Data-Driven Decision Making in Online and Offline Retail

by

Divya Singhvi

B.S., Cornell University (2015)

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2020

Author .....
Sloan School of Management
July 26, 2020

Certified by .....
Georgia Perakis
William F. Pounds Professor of Management Science
Co-director, Operations Research Center
Thesis Supervisor

Accepted by .....
Patrick Jaillet
Dugald C. Jackson Professor, Department of Electrical Engineering and Computer Science
Co-director, Operations Research Center

Data-Driven Decision Making in Online and Offline Retail

by

Divya Singhvi

Submitted to the Sloan School of Management
on July 26, 2020, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Operations Research

Abstract

Retail operations have experienced a transformational change in the past decade with the advent and adoption of data-driven approaches to drive decision making. Granular data collection has enabled firms to make personalized decisions that improve customer experience and maintain long-term engagement. In this thesis we discuss important problems that retailers face in practice, *before*, *while* and *after* a product is introduced in the market.

In Chapter 2, we consider the problem of estimating sales for a new product before retailers release the product to the customer. We introduce a joint clustering and regression method that jointly clusters existing products based on their features as well as their sales patterns while estimating their demand. Further, we use this information to predict demand for new products. Analytically, we show an out-of-sample prediction error bound. Numerically, we perform an extensive study on real world data sets from Johnson & Johnson and a large fashion retailer and find that the proposed method outperforms state-of-the-art prediction methods and improves the WMAPE forecasting metric between 5%-15%.

Even after the product is released in the market, a customer's decision of purchasing the product depends on the right recommendation personalized for her. In Chapter 3, we consider the problem of personalized product recommendations when customer preferences are unknown and the retailer risks losing customers because of irrelevant recommendations. We present empirical evidence of customer disengagement through real-world data. We formulate this problem as a user preference learning problem. We show that customer disengagement can cause almost all state-of-the-art learning algorithms to fail in this setting. We propose modifying bandit learning strategies by constraining the action space upfront using an integer optimization model. We prove that this modification can keep significantly more customers engaged on the platform. Numerical experiments demonstrate that our algorithm can improve customer engagement with the platform by up to 80%.

Another important decision a retailer needs to make for a new product, is its pricing. In Chapter 4, we consider the dynamic pricing problem of a retailer who does not have any information on the underlying demand for the product. An important feature we incorporate is the fact that the retailer also seeks to reduce the amount of price experimentation. We consider the pricing problem when demand is non-parametric and construct a pricing algorithm that uses piecewise linear approximations of the unknown demand function and establish when the proposed policy achieves a near-optimal rate of regret $(\tilde{\mathcal{O}})(\sqrt{T})$, while making $\mathcal{O}(\log\log T)$ price changes. Our algorithm allows for a considerable reduction in price changes from the previously known $\mathcal{O}(\log T)$ rate of price change guarantee found in the literature.

Finally, once a purchase is made, a customer's decision to return to the same retailer depends on the product return polices and after-sales services of the retailer. As a result, in Chapter 5, we focus on the problem of reducing product returns. Closely working with one of India's

largest online fashion retailers, we focus on identifying the effect of delivery gaps (total time that customers have to wait for the product they ordered to arrive) and customer promise dates on product returns. We perform an extensive empirical analysis and run a large scale Randomized Control Trial (RCT) to estimate these effects. Based on the insights from this empirical analysis, we then develop an integer optimization model to optimize delivery speed targets.

Thesis Supervisor: Georgia Perakis
Title: William F. Pounds Professor of Management Science

Acknowledgements

First and foremost, I would like to thank my advisor Georgia Perakis for her tremendous support throughout my time at MIT. I consider myself very lucky to have found an advisor who is as caring and loving as Georgia. PhD can be like a roller coaster ride and it was no different for me. I went through many ups and downs, both professionally and personally, and Georgia stood strongly behind me, always supporting and encouraging me to take challenges head on. Her faith in me, and her optimistic attitude towards research is what got me through these years. I would also like to thank her for providing me with the opportunity to collaborate with excellent researchers. Georgia's dedication towards her work, and especially her students is unparalleled. I truly hope that one day I can become at least half as good a mentor and researcher as she is today.

Next, I would like to thank Vivek Farias, Jonas Jonasson, and Huseyin Topaloglu, members of my thesis committee. Their help and guidance during my time on the academic job market, and advice on the thesis have been truly invaluable. I would also like to thank Retsef Levi, and Karen Zheng, thesis advisors to Somya. I have thoroughly enjoyed my frequent conversations and mentorship throughout these years. Thank you also to Nikos Trichakis for his guida. Special thanks also to Dimitris Bertsimas and Patrick Jaillet for leading the ORC during my time at MIT and making it the amazing place that it is today. Thank you also to Laura Rose and Andrew Carvalho for ensuring the well being of the ORC students all the time.

I would also like to thank my undergraduate mentors Peter Frazier, Shane Henderson, Andreea Minca, Eoin O'Mahony, David Shmoys and Dawn Woodard for inspiring me to pursue graduate research. It was your mentorship and faith in me that got me to MIT. I will always be grateful to you for shaping my initial years as a researcher. Sincere thanks also to my collaborators Lennart Baardman, Hamsa Bastani, Pavithra Harsha, Igor Levin, Yiannis Spanditakis, Omar Skali-Lami, Leann Thayaparan and Amine Anoun. I have learnt tremendously from each one of you during my time at MIT. Special thanks to Lennart, for your astounding patience and

guidance during my early years as a PhD student; and to Hamsa and Pavithra, for being amazing collaborators and mentors. Thank you also to the MIT UROPs, MBAn and the industry collaborators that I had the chance to work with during my time at MIT.

I would also like to thank my incredible friends Shwetha & Paul, Deeksha & Pradeep, Tamar & Jonathan, Mukund, In-Young and Eduardo. MIT can be a stressful place, but thanks to the amazing friends that I made over the past five years, it was fun and exciting at the same time. Thank you for all the food, fun, laughter and trips around the world. I am grateful to have found a family away from home and to have formed such lasting friendships. Thank you also to Kushagra, Nidhi, Rajan, Ricardo, Swati & Tushar, Nishanth, Velibor, Brad, Peter, Andrew, Matthew, Michael, Elizabeth, Steven, Daniel, Mohit, Rajesh, Rakshith, Chinmay, Vaishnavi, the ORC family and the coffee hour family at the Ashdown house for all the interesting conversations throughout my time at MIT.

Lastly, I would like to show my deepest gratitude towards my family and elders without whose blessings and support, I would not be here. To my parents for raising me to be what I am today; to Somya for being my strongest support and oldest companion; to Bhavya & Samta for your perspective in life; to my in-laws for being so supportive and welcoming and finally to my wife, Dhwani who has seen me in the most stressful phase of my PhD life but has been a pillar of support, and has made this journey all the more fun.

I dedicate this thesis to my family, to whom I owe everything in this life.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

Operations management has undergone a paradigm shift in the last decade due to the emergence of new data driven tools and practices to guide decision making. Granular data collection combined with increased computational power has ushered an era where anything and everything is recorded. A Forbes report from 2018 states that more than 90% of all data was generated the previous two years. This paradigm shift has also led to a real need to look at classical operations management problems through a data-driven lens, in order to make sure they connect well with current practice. As a result, in this thesis we consider various decision problems related to retail management, and propose novel methods that are both practically relevant and theoretically strong.

To motivate the different decision problems considered in this thesis, we take a holistic view of retail management in both the online and the offline setting. Retail operations for products involve decision that are made prior to the launch of the product, decisions that are made when the product is on the market and finally decisions that are made after the product is sold to the customers.

Pre-launch decision involve deciding whether to release a new product, how much inventory to produce, amongst many other decisions. Demand predictions play a central role in guiding all these decisions. But predicting demand for new products remains one of the most challenging problems. Each year, firms spend billions of dollars on new product launches but with little success (Willemot et al. 2015). In fact a recent survey states that more than 72% of all new products launched in the market do not meet their revenue targets. This leads to considerable

bottom line losses for retailers. Hence, the first problem that we consider is that of predicting sales for a new product. While accurate sales forecasts are instrumental for various operational decisions, current practice has considerable difficulties in predicting the success of a new product. Standard forecasting models rely on using past sales data, but new product sales forecasting happens far in advance without any availability of sales history. This results in firms investing in costly and time consuming quantitative methods such as surveys and expert opinions (Kahn 2002). Nevertheless, the use of analytics for predicting sales remains limited and as much as 80% of companies that use analytics are not satisfied with their approach (Cecere 2013, Kahn 2014).

Similarly, once the product is launched on the market, its eventual success is determined based on whether it is recommended to the the right set of customers. But with an explosion in product variety and considerable heterogeneity in customer taste, making relevant recommendations remains a challenging problem. Popular techniques use prior data on customer preferences to personalize recommendations but since there is no data for new customers, making personalized recommendations becomes complex. To solve this, current practice involves recommending different products to estimate individual customer preferences. But oftentimes, this experimentation leads to customer disengagement, on account of poor recommendations. In fact, a recent survey indicated that as much as 80% of the customers opt out of marketing emails because of irrelevant recommendations.

Even if the customer is recommended the products that are relevant for her, the eventual purchase decision depends on whether it is appropriately priced. Naturally, pricing decisions play an important role in ensuring a product's success (Huang et al. 2007). Since there is no sales data to estimate price elasticity, dynamic pricing is used to experiment with prices and estimate demand. Nevertheless, many practical constraints make the problem challenging. For example, traditional learning policies involve frequent price experimentation which is infeasible in many retail settings because of the negative effects of frequent price changes (Netessine 2006, PK Kannan 2001). Hence retailers often resort to pre-decided pricing policies and risk loosing considerable revenues due to poor pricing decisions (Carmichael 2014).

Finally, long-term engagement with customers is very important for retailers. And whether a customer will return to the same retailer depends on after-sales services such as delivery speed and product return polices. In fact, returns remains one of the largest problems in retail. Return rates in online retail can be as high as 30% of the total orders. Hence, the problem of returns

has been rightly referred to as a "ticking time bomb" estimated to be as much as $ 260 Billion per year (Reagan 2016). In summary, the thesis aims to develop tools and techniques that can be used to solve the aforementioned problems of practical relevance.

In summary, there are many challenges in using analytics for decision making in retail management.

1.2 Contributions

The main contribution of this thesis is in developing analytical tools and methods for practical problems in revenue management through close collaboration with industry practitioners.

From a practical stand point, the work in this thesis is an outcome of close collaboration with industry partners. This close collaboration has ensured that the solutions we develop remain practically relevant. This is particularly important, in light of previously cited reports that discuss how very few retailers have adopted analytical tools for driving decisions. For example, the second chapter discusses the creation of an excel tool, developed in collaboration with Johnson & Johnson, that can be used by managers to forecast sales for new products using sales data from comparable products. The third chapter is devoted to developing a practical recommendations method whose performance is tested on real world movieLens data. Similarly the fourth chapter discusses a pricing algorithm that ensures that the practical business constraint of very limited price changes is satisfied. Finally the fifth chapter discusses a Randomized Control Trial that was run in collaboration with one of the largest e-fashion retailers in India to understand the causes of product returns and to reduce them. In summary, the thesis contributes by developing practical solutions to important problems in retail management.

From a methodological stand-point, the thesis contributes in two main ways. First, we develop new algorithms with provable analytical guarantees for problems in sequential-decision making related to online and offline retail. Second, we introduce and analyze novel applications of classical problems that are relevant due to the recent advances in personalized data generation and collection. For example, the second chapter on demand estimation introduces new machine learning and optimization methods in order to develop a demand estimation method with provable out-of-sample prediction analytical guarantees. The third and the fourth chapter develop Bandit learning algorithms in order to learn the demand for new products with provable regret guarantees. Finally, the last chapter focuses on using an empirical analysis combined

with optimization techniques to develop practical insights into the problem of product returns. The underlying theme in each case is to develop tractable methods that can enable retailers to leverage existing data to solve problems when there is little data to drive "good" decisions. We discuss the contributions of each of the chapters in more detail next.

Chapter 2 investigates how to forecast sales of new products when there is no prior data to estimate the demand for the product. Collaborating with Johnson & Johnson as well as a large fashion retailer, we find that these estimates are crucial for many decisions including those related to production, pricing and logistics among others. Yet, the problem of predicting sales for new products is complex since no prior sales data is available to fit prediction models. Hence, we devise a joint clustering and regression method that jointly clusters existing products based on their features as well as sales patterns while estimating their demand. Intuitively, this approach uses data from comparable past products to estimate the demand of the new product. Analytically, we prove in-sample and out-of-sample prediction error guarantees in the LASSO regularized linear regression case to account for over-fitting due to high dimensional data. We show that as the size of the training data from comparable products increases, the prediction error of the new product decreases. Numerically we perform an extensive comparative study on real world data sets from Johnson & Johnson and a large fast fashion retailer. We show that the proposed algorithm outperforms state-of-the-art prediction methods and improves the WMAPE forecasting metric between 5%-15%. Furthermore, since the proposed method is inspired from the intuition of our practitioner collaborators, the method is more interpretable. We also provided a data-driven tool for forecasting sales that can guide practitioners in other operational decisions.

Chapter 3 considers the problem of personalized recommendations, another important lever that retailers and service providers use in order to increase demand. This problem becomes particularly relevant in the current era where recommendations are omnipresent: from personal emails to social media and news feeds. We study the problem of personalized product recommendations when customer preferences are unknown and the retailer risks losing customers because of irrelevant recommendations. We present empirical evidence of customer disengagement through real-world data from a major airline carrier who offers a sequence of ad campaigns. Our findings suggest that customers decide to stay on the platform based on the relevance of the recommendations they are offered. We formulate the problem as a user preference learning problem with the notable difference that the customer's total time on the platform is a function

of the relevance of past recommendations. We show that this seemingly obvious phenomenon can cause almost all state-of-the-art learning algorithms to fail in this setting. For example, we find that classical bandit learning as well as greedy algorithms provably over-explore. Hence, they risk loosing all customers from the platform. We propose modifying bandit learning strategies by constraining the action space upfront using an integer optimization model. We prove that this modification allows us to keep significantly more customers on the platform. Numerical experiments on real movie recommendations data demonstrate that our algorithm can improve customer engagement with the platform by up to 80%.

Chapter 4 investigates the problem of pricing of new products for a retailer who does not have any information on the underlying demand for a product. The retailer aims to maximize cumulative revenue collected over a finite time horizon by balancing two objectives: *learning* demand and *maximizing* revenue. The retailer also seeks to reduce the amount of price experimentation because of the potential costs associated with price changes. Existing literature solves this problem in the case where the unknown demand is parametric. We consider the pricing problem when demand is non-parametric. We introduce a new pricing algorithm that uses piecewise linear approximations of the unknown demand function and establish when the proposed policy achieves near-optimal rate of regret, $\tilde{\mathcal{O}}(\sqrt{T})$, while making $\mathcal{O}(\log \log T)$ price changes. Hence, we show considerable reduction in price changes from the previously known $\mathcal{O}(\log T)$ rate of price change guarantee in the literature. We also perform extensive numerical experiments to show that the algorithm substantially improves over existing methods in terms of the total price changes, with comparable performance on the cumulative regret metric.

Finally in Chapter 5, we focus on the problem of reducing product returns, one of the key challenges that retailers face worldwide. We investigate this problem through a supply chain lens. Closely working with one of India's largest online fashion retailers, we focus on identifying the effect of delivery gaps (total time that customers have to wait for the item to arrive) and customer promise dates on product Returns To Origin (RTO): the setting where the customer refuses to accept the package when delivered at their door and returns it back to the retailer. Our empirical analysis reveals that an increase in delivery gaps causes an increase in product RTO. We estimate that a 2-day reduction in the delivery gap from the current average can lead to annual cost savings of up to \$1.5 million just from RTO reduction for the retailer. To estimate the effect of delivery promise on product returns, we conduct a Randomized Control Trial. We find that in regions where product deliveries are expedited, beating the customer promise

date by overshooting the promise can further lead to a reduction in product RTO. Based on the insights from this empirical analysis, we then develop an integer optimization model that mimics managers' decision-making process in selecting delivery speed targets. Our integer optimization formulation can account for various business constraints that might be relevant in practice. In order to make the optimization model solve fast, we propose a linear optimization relaxation-based method and show, both analytically as well as through simulations, that the method's performance is near-optimal.

Chapter 2

Leveraging Comparables for New Product Sales Forecasting

2.1 Introduction

Business analytics enables firms to improve their operational decision making through its data-driven techniques. Analytics transform data to decisions by applying techniques from statistics, machine learning, and optimization. Progress in data storage and computation power has benefited analytics significantly. These advances have enabled algorithms to handle more complex datasets in a faster manner. Additionally, analytics has benefited from improvements to the algorithms themselves. Novel models are able to describe data more accurately, and hence take decisions more optimally. In this chapter, we propose a new approach for predictive analytics when facing new entities (e.g., new customers or new products). In addition, we apply this new algorithm to an important problem in the space of retail, namely that of new product sales forecasting.

Sales forecasting is a central activity in a firm's operations. Most operational decision making tools incorporate models describing product sales. Particularly, the sales forecasts of new products guide many of the operational decisions made during product development (e.g., production, inventory, and pricing). Making the right decisions is key to the success of a product launch, and therefore, it is important to forecast the sales of a product accurately. The difficulty in doing this varies considerably between industries, even for existing products. When historical sales data is available, regularly purchased products (e.g., fast moving consumer goods) are easier to predict than temporary products (e.g., fashion clothing). Several studies illustrate this

with a mean absolute error (MAE) of roughly 5 units and a mean absolute percentage error (MAPE) of 11% to 19% for grocery brands (Ali et al. 2009, Cohen et al. 2017), which worsens for fashion retailers where it is around 13 units and 68% to 93% (Ferreira et al. 2016). Regular consumption of a product reduces the variability of its sales over time, which means that its historical sales will be a good indicator of future sales.

Within this area, we focus on new product sales forecasting, where typically no historical sales data is available. The interest in this particular problem originates from our collaboration with Johnson & Johnson Consumer Companies Inc., a major fast moving consumer goods manufacturer, and was later verified by the interest of a large fast fashion retailer in our approach. Both industry partners introduce new products to update their assortment frequently: monthly in the fast moving consumer goods industry, and weekly or sometimes daily in the fast fashion industry. Before and during a new product launch, each firm needs to make many decisions that affect the success of the product. These decisions span the entire range of operations: capacity planning, procurement, production scheduling, inventory control, distribution planning, marketing promotions and pricing. As each of these decisions are guided by forecasts, an accurate sales forecasting model is key to a successful product launch for both industry partners and others. The importance of this success has grown tremendously over the past decade, as Cecere (2013) estimates that on average new product costs have increased four times over a period of five years.

Current practice has considerable difficulties with predicting new product success. Standard forecasting models use past sales data to predict on the short term. However, predictions for new products need to be made far in advance without any sales history. As a result, many firms, our industry partners included, resort to costly and time-consuming qualitative methods. Kahn (2002) suggests that surveys, expert opinions, and average sales of comparable products are the most widespread techniques for predicting demand of new products. These methods are popular due to their interpretability. This is an essential characteristic, as Armstrong et al. (2015) argue that practitioners should be overly conservative when they do not understand the forecasting procedures. Possibly for this reason, Kahn (2002) observes that at the time only 10% of companies make use of some form of analytics. More recently, Cecere (2013) and Kahn (2014) argue that the usage of analytics is still limited. At the same time only 20% of companies are satisfied with their approach. Altogether, this literature shows that there are opportunities to improve the new product sales forecasting process significantly, particularly using analytical

techniques from optimization, statistics, and machine learning.

In this chapter, we develop an accurate, scalable, and interpretable forecasting method calibrated with our industry partners' data. These characteristics of the approach are important to both industry partners. For interpretability, we draw inspiration from the practice of our industry collaborators who use comparable products to predict sales of new products. Currently, many practitioners manually determine which products are similar to the new product and use their average sales as a prediction. We propose a Cluster-While-Regress model that mimics this, but simultaneously creates clusters of comparable past products and creates forecasting models for each cluster. Adding to the interpretability, the model also includes feature selection methods from statistics and machine learning. Incorporating regularization allows each cluster to prioritize different variables as the most important predictors of sales. As an example, customers might be more price sensitive for generic brands than for premium brands. To address scalability, we devise a fast optimization algorithm whose steps mimic industry practice. The accuracy of the estimated model is established both in theory and in practice. Theoretically, we prove bounds on the in-sample and out-of-sample forecasting error of the estimated model that holds with high probability. From a practical standpoint, we estimate our model on data from our two industry partners, and observe significant improvements in out-of-sample forecasting metrics. From a broader standpoint, we remark that our proposed analytics model can be used more generally to predict for new entities.

2.1.1 Contributions

Our main contribution is the development of a new forecasting approach based on ideas from optimization, statistics, and machine learning. We estimate a clustered forecasting model using data of comparable products, and show strong results on the forecasting accuracy for new products introduced by our industry partners. To summarize our contributions:

1. *Interpretable predictive analytics model using clustering and regression:* We propose a novel approach to predicting outcomes for new entities. Motivated by the practice in new product sales forecasting, we say that sales of different product clusters are generated by different sales models. In our model formulation, comparable products are clustered together and share a forecasting model that can be any regression model. As this Cluster-While-Regress (CWR) model is grounded in current practice, it is easy to use for managers who need to understand the model in order to trust it and use it. Further-

more, the approach is general and can account for changes in drivers of product sales such as marketing budget or distribution decisions. Section 5.2 uses the data of our industry partners to motivate our general model, which is formulated in Section 2.3.1.

2. *Tractable Cluster-While-Regress algorithm:* The forecasting process associated with our proposed model takes a stepwise approach. First, we estimate the clustered forecasting model, then, we estimate a cluster assignment model, and finally, we assign the new product to a cluster and compute the correct forecast. Altogether, this forms the basis of the CWR algorithm. For the clustered forecasting model, we propose to solve a mixed integer non-linear optimization model, while for the cluster assignment model, we propose to fit a multiclass classication model. This method is flexible, as it allows using any regression model (e.g., linear regression, generalized linear models, regression trees, etc.) in the clustered forecasting model, and any classification model (e.g., multinomial logistic regression, support vector machines, classification trees, etc.) in the cluster assignment model. In cases where solving the mixed integer non-linear optimization problem is computationally intractable, we propose an approximate CWR algorithm. This algorithm approximates the mixed integer non-linear optimization problem using an iterative optimization procedure. Section 2.3.2 discusses the clustered forecasting problem, and Section 2.3.3 describes the cluster assignment problem.

3. *Application to regularized linear regression:* In our applications, we focus on the case of regularized linear regression, because of its clarity to practitioners coming from its interpretable coefficients and meaningful clusters. We formulate the linear version of our model, and adapt the CWR algorithm to the linear case. For the linear model, we prove that the forecasting error of the algorithm's solution is bounded with high probability both in-sample and out-of-sample. Section 2.4 formulates the linear model, discusses its estimation, and presents these results.

4. *Strong performance on experimental and real data:* To test the practical performance of our algorithm, we run a variety of computational experiments as well as tests on real data. In our computations, the CWR algorithm significantly outperforms the benchmark algorithms including random forests and gradient boosted trees. Working in collaboration with two large industry partners, we also show that our algorithm results in at least 5%-15% WMAPE improvement on their data. These results are particularly robust to

external changes in the market as we also incorporate competitor's data in the estimation of our prediction model, which results in a better understanding of how the market responds to new product releases. In order to further check the robustness of our results, we test our algorithm on various product categories and observe similar improvements. Section 2.5 compares our algorithm's performance against the benchmark algorithms on experimental data. In Section 2.6, we describe the results on fast moving consumer goods data. Furthermore, Section 2.7 tests the robustness of the model with fast fashion retail data.

5. *Accessible forecasting tool for practitioners:* Out-of-sample results encouraged our fast moving consumer goods manufacturing partner, Johnson & Johnson, to employ the forecasting approach. Our model can be estimated offline, which allows us to code the estimated model into Excel. In this tool, our partners can experiment with a product by changing product features, which immediately gives a report on expected sales, trends in sales over time, and the most important constituents of predicted sales. This allows the managers to identify the key drivers of demand for own and competitive products, in turn this allows them to optimize their new product launch and outperform competitive launches through scenario planning. Positive feedback has encouraged further development of the tool. Section 2.6 discusses the forecasting tool.

2.1.2 Literature Review

Our work relates to the literature on both sales forecasting and product innovation which have been studied extensively in the operations management and marketing literature. More specifically, our work lies in the intersection of three different streams of literature: product diffusion and innovation in marketing, new products and high dimensional models in operations management, and clustered regression models in machine learning and statistics.

First of all, product diffusion and innovation has been widely studied in the marketing literature. As the seminal paper in this area, Bass (1969) develops a simple yet strong model that estimates how a new product diffuses through a population. The Bass model predicts lifetime sales based on a few parameters: market size, coefficient of innovation, and coefficient of imitation. Over the years, this model has been extended substantially and Bass (2004) discusses some of the most important extensions, namely how successive generations of products diffuse and how contextual features such as pricing can be included in the model. The model is still

widely used, for example, the inclusion of online reviews (Fan et al. 2017b), predicting adoption of new automotive technologies (Massiani and Gohs 2015), the use of personal health records (Ford et al. 2016). For reviews on product innovation and diffusion in marketing we refer to Chandrasekaran and Tellis (2007) and Fan et al. (2017a). The Bass model, while widely used, makes lifecycle predictions. Hence, it might make significant prediction error in the introductory sales prediction, the focus of this chapter. In comparison, we propose a model that is grounded in practice and is data-driven by incorporating machine learning and statistical tools.

A second stream of literature comes from operations management. In this area, the interest in studying operational decision making for new products is growing. In particular, recent studies have considered production, inventory, and pricing of new products. Previously, Fisher and Raman (1996) show that decisions can be improved significantly through accurate sales forecasts. While almost all decision making models in operations management include a forecasting component, there are very few papers that carefully study the sales forecasting problem itself. Recently, Hu et al. (2016) use a two step approach to forecasting the sales of new products and show that mean absolute errors reduce by around 2-3%. Their forecasting model fits lifecycle curves to products, then clusters these products, and aggregates the predictions. In contrast to estimating the entire lifecycle of a product, our forecasting problem focuses on the product's introductory period. After this period, the acquired sales data can be used by existing models (Ali et al. 2009, Huang et al. 2014) to generate better forecasts. Inherently, this means our approach deals with the more complex and most uncertain period during a product's lifecycle. Furthermore, instead of the two step approach, we propose an algorithm that estimates clusters and forecasting models jointly, while allowing each cluster's forecasting model to be any machine learning or statistical regression model. Specifically, this allows our model to incorporate other features such as pricing, marketing, and distribution into the prediction models.

With regards to new product pricing, more attention has recently been placed on pricing when the demand curve is unknown. Specifically, a new product is released into the market and dynamic pricing is used as a tool to understand the underlying demand. These studies set prices carefully to maximize revenue while balancing exploration and exploitation. Keskin and Zeevi (2014) study asymptotically optimal policies for pricing a product with linear demand, but assume either no or limited data is available. Ban and Keskin (2017) extend to the setting where customer characteristics are available and pricing policies can be personalized. In certain industries, experimenting with the price of a new product is not desired or allowed. This problem

is analyzed by Cohen et al. (2015) who propose a simple pricing policy based on linear demand curves that performs well for many parametric forms of unknown demand curves. To contrast with this literature, we assume that historical data is available for comparable products and focus on the forecasting problem.

Concerning production and inventory management of new products, several models have been developed to improve decision making before the product launch. In Fisher and Raman (1996), the quick response system changes production based on the sales forecast of the new product. The analysis shows that responding with accurate forecasts can increase profits by up to 60% in the fashion industry. Several studies followed along the same lines, for example, Caro and Gallien (2010), Gallien et al. (2015), Chen et al. (2017b), and Ban et al. (2017). These studies optimize production and inventory decisions assuming a particular structure on the demand for the new products. This contrasts with our focus on improving the sales forecasting model itself. In turn, our forecasts can also be used to improve these operational decision models.

From a theoretical point of view, this work is related to the recent surge in operational models that involve high dimensional features about people or products. This increase in data enables personalized or product-specific policies. Regularization is used to control model complexity and ensures that decisions can generalize to when new people or products arrive. Among others, Bastani and Bayati (2015), Javanmard and Nazerzadeh (2016), Ban and Keskin (2017) analyze various operations management problems, such as pricing and healthcare delivery, from a high dimensional perspective. Our work differs in that regularization is used to improve the accuracy of predictions instead of prescriptions. Datasets have grown in both observations (e.g., number of people and products) and features (e.g., information on people and products). Therefore, our models involve high dimensional data, and for the aforementioned reason, we use regularization to avoid overfitting. Naturally, the model is to be used to enhance operational models, but our main focus is the forecasting problem itself.

Finally, the model that was applied to the data of our industry collaborators uses a LASSO regularized regression model for each cluster. In this setting, our model is related to clusterwise regression. Clusterwise linear regression models cluster observations and fit linear regression models to these clusters simultaneously. The objective is to find different clusters of observations whose data generating mechanisms follows significantly different correlation patterns. The framework of clusterwise linear regression has been applied to various application domains

including market segmentation (Brusco et al. 2002), income prediction (Chirico 2013), rainfall prediction (Bagirov et al. 2017) and others.

Mainly, research has focused on the computational aspects of combining clustering with regression. One of the first algorithms to solve this problem was a heuristic proposed by Späth (1979), which, in each iteration, reassigned a single point to a different cluster if it would result in a reduction of the prediction error. Since then, several algorithms have been proposed to solve the clusterwise regression problem. In particular, clusterwise linear regression problems are solved by DeSarbo et al. (1989) using simulated annealing, by DeSarbo and Cron (1988) using maximum likelihood estimation and expectation-maximization, and by Viele and Tong (2002) using Gibbs sampling while also providing theoretical consistency results on the posterior sampling distribution. More recently, new aspects of the clusterwise regression problem have been studied, such as fuzzy regression and fuzzy clustering by D'Urso et al. (2010), and robust regression by Schlittgen (2011). Mathematical programming based approaches have also been proposed. For example, Lau et al. (1999) propose to solve a nonlinear formulation, Bertsimas and Shioda (2007) propose to solve a compact mixed-integer linear formulation, and Carbonneau et al. (2011, 2012), Park et al. (2016) propose a heuristic based on column generation for an integer linear formulation.

Our work differs from these studies in several important ways. Methodologically, our work is complimentary to these papers because it provides a new technique for out-of-sample predictions, which is still considered a challenge in clusterwise regression, even in the linear setting. In particular, any new observation needs to be assigned to a cluster before its prediction can be made. Two possible approaches that have been proposed previously are the following: fit a cluster assignment function during the estimation process with which a new observation can be clustered and a prediction can be made (Manwani and Sastry 2015), or to provide weights to each cluster for a new observation and a weighted average becomes the prediction (Bagirov et al. 2017). Our approach combines these ideas by developing a novel out-of-sample prediction method that uses a data-driven function based on logistic regression weights in order to weight the forecasts of each cluster. Additionally, in contrast to previous literature, we expand the clusterwise regression setting to the high dimensional setting by considering regularized linear regression models for each cluster. Hence, we consider the more realistic setting where our model has access to many features, but it might need to discard those that are relevant. Analytically, we contrast the literature by providing an out-of-sample prediction error bound, which shows

that our model is statistically consistent. Our analysis is based on the cluster compatibility condition, which is related to the classical compatibility condition if the number of misclassifications during the estimation process is limited. Finally, our model extends beyond the linear case of clusterwise regression, as our estimation algorithm functions even when the forecasting models of a cluster take nonlinear forms such as nonlinear regression models, regression trees, or random forests.

2.2 Motivation and Data from Practice

Before introducing our model, we describe the problem faced by our two industry partners, we discuss their current approaches to new product sales forecasting, and describe the challenges of these approaches. Finally, we use the data of our industry partners to motivate the clustered forecasting model proposed in this chapter.

Our industry partners, and firms more generally, invest millions of dollars in innovation every year. The success of a new product is partially dependent on making the right operational decisions surrounding the product launch. These decisions are guided by sales forecasts, and therefore, our industry partners note the importance of accurate forecasting. Though, accuracy is not the only metric of importance. It is also important that the forecasting tool is interpretable and scalable. The model needs to be interpretable because new products are surrounded by large uncertainty and models that are not easy to understand will receive less usage from practitioners. In certain industries, such as fast moving consumer goods and fast fashion, new products are introduced frequently and scalability is important. Their extensive product assortment leads to large datasets to fit forecasting models on. Additionally, due to frequent product introductions, these firms need to forecast often. Table 2.1 shows the number of products that our partners introduced in several consumer goods and fashion categories during the corresponding periods.

Table 2.1: Description of available datasets from two partners with multiple categories

Industry	**Category**	**Time Period**	**Number of New Products**
Consumer Goods	Baby care	2013-2017	122
	Body care	2013-2017	71
	Facial care	2012-2016	219
Fashion	1	April 2016-June 2016	75
	2	April 2016-June 2016	66

Over a period of five years and three different product categories, the consumer goods manufacturer released a total of 412 new products. At an even faster rate, the fashion retailer

introduced 143 new products over just three months and two categories. The faster pace of innovation also corresponds to products with shorter lifetimes. This does not change the fundamental problem of forecasting sales of new products, but it has an effect on the scale of the problem. For example, yearly forecasts might be adequate for production, procurement, and inventory decisions at a manufacturer, while weekly forecasts are needed for distribution and pricing decisions at a retailer. In particular, where our consumer goods partner is interested in forecasting first year sales in an entire country, the fashion retail partner needs accurate forecasts for the first half-week at a store or regional level.

Forecasting new product sales is a clear challenge for both partners. Often, errors are too large, namely over 50% off. In discussions, our partners explained that using these forecasts would lead to wrong decisions. As an example, applying current forecasting methods to a subcategory, only 16 out of 31 product introductions were predicted accurately enough to aid in effective decision making. To illustrate the difficulty, Figure 2.1 shows the actual sales over the first six months of two new products that were released by the consumer goods manufacturer. The two products belong to the same product subcategory, yet behave very differently. While the first product shows an increasing trend in sales over the introductory horizon, the sales of the second product decline over the same months after release.

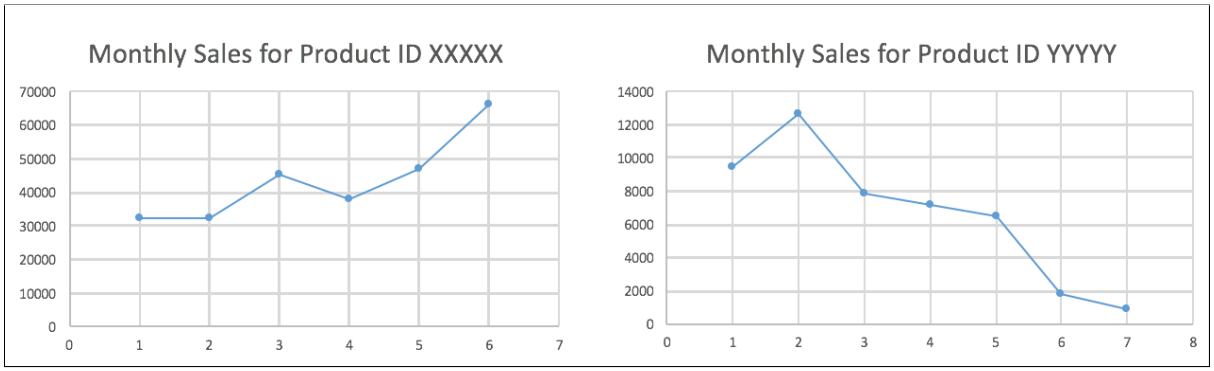

Figure 2.1: Actual sales data of two new products over the first six months after introduction

Evidently, consumer response to the two new products was very different. The cause of this difference can be attributed to many factors: pricing, promotions, distribution, product attributes (such as size, packaging, and color), and other latent factors. From Johnson & Johnson Consumer Companies Inc., we have access to five year long datasets in which each observation describes the monthly sales of a SKU (Stock Keeping Unit) in an entire trade channel. Interestingly, this dataset includes the sales data from competitor's products at the same level of granularity. Available product features include the date of product introduction,

the product's categorization, price, promotional events, sizing, packaging, chemical composition, claimed benefits, as well as distribution measures such as the number of stores selling the product and %ACV (Percentage of All Commodity Volume), which is a measure of distribution across stores. The fast fashion retailer gave us access to a dataset containing three months of half-weekly store sales for individual SKUs. The data contains product features such as date of introduction in both the brick-and-mortar and the online channel, product categorization, price, color, style, and prior clickstream data if the product was introduced online earlier (e.g., cumulative views online, cumulative add-to-carts online, cumulative remove-from-carts online). More information on the data can be found in Sections 2.6 and 2.7. Unfortunately, even when accounting for all these product features and analyzing a specific subcategory, a single model to predict sales of new products is often unable to capture certain hidden factors. One visible example of these latent factors is the upwards or downwards sales trend in Figure 2.1. This trend can be hard to predict before a product launch, as no historical sales data is available.

As a remedy, our partners and many other firms in the FMCG and fast fashion retailing business, use prediction tools that combine market surveys, expert opinions, and comparable products. Apart from being expensive, these research approaches are time consuming. Moreover, the final product often differs from the prototype product during market research, which undermines the accuracy of these forecasts. Hence, high costs and long prediction lead times, make these tools unpractical for most of the smaller and even medium sized new product launches. This problem has been more pronounced in recent times when the frequency of new product launches has been increasing. Therefore, sometimes, firms have to resort to more qualitative methods to forecast new product sales. One collaborator has established a forecasting technique where product managers use their expertise to find products comparable to the new product and then use their actual sales as a forecast. The hope is that these clusters of comparable products capture shared latent factors. Thus, if the right comparables have been selected, the effects of latent factors on sales should be captured by the actual sales of the clustered products. For this reason and the ease that practitioners have with forecasting using comparables, in the remainder of the chapter, we develop an algorithm and subsequently a tool which clusters products and simultaneously fits demand forecasting models for these clusters.

2.3 Cluster-While-Regress Model

In what follows, we introduce our sales forecasting model for new products. As described, many firms use a *Cluster-Then-Regress* (CTR) model in which experts select past products similar to the new product and use their average sales as a forecast. Unfortunately, initial tests on data from our industry partners showed that this approach produces weak results even when using data-driven clustering (see Sections 2.6 and 2.7). Nonetheless, our new product sales forecasting model is related to this practice, namely we propose a *Cluster-While-Regress* (CWR) model that clusters products and fits sales forecasting models to each cluster simultaneously. The main difference between the two approaches is that the data-driven version of the CTR model clusters based on product feature similarity, fixes these clusters, and then forecasts the cluster's sales. Instead, our CWR model clusters products and estimates forecasting models simultaneously, thereby clustering on the similarity in terms of both product features as well as sales behavior of the products.

2.3.1 General Model

Formalizing our approach, we are interested in predicting sales of a new product, $y_0 \in \mathbb{R}$, based on m product features that are available before introduction, $x_0 \in \mathbb{R}^m$. Examples of these product features include the aforementioned data such as the product's regular price, brand, and sizing. Naturally, firms also use historical sales data to forecast future sales of existing products, but lack this data for new products. As a result, the challenge is to provide an accurate sales forecast for the new product without an indication of its rough sales potential. Through our industry partners we have access to data on past products that were once new. By $y_i \in \mathbb{R}$ and $x_i \in \mathbb{R}^m$, $i = 1, \ldots, n$, we denote the sales and product feature data for n past new products.

Using this data, we determine a model describing the sales of these past products, which can then be used to predict sales for the new product. Specifically, we consider ℓ clusters of products, each with a different sales generating model, but where the products within a cluster share the same model. To be precise, we propose the following sales generating model:

$$y_i = \sum_{k=1}^{\ell} z_{ik} f_k(x_i) + \epsilon_i, \quad i = 0, 1, \ldots, n, \tag{2.1}$$

where $z_{ik} \in \{0, 1\}$ indicates whether product i belongs to cluster k, $f_k(x_i)$ is the sales forecasting

model for a product in cluster k with features x_i (i.e., a particular functional form to estimate the conditional expected sales for a product in cluster k with features x_i), and ϵ_i is assumed to be a zero-mean random noise. The conditional expected sales of each cluster can be a highly non-linear function of the available features. Thus, our estimation approach needs to be able to incorporate a large variety of cluster forecasting models. Examples of f_k include linear regression models (used in our application), generalized linear models, non-linear regression models, regression trees, and random forests.

In addition, our approach needs to allow for estimating sparse models. To exemplify this necessity, consider the case where f_k is a linear regression model. The total number of parameters of model (2.1) is then not just the number of product features, m, but rather a multiple of the number of product features and the number of clusters, $m\ell$. This means that the dimension of the model can grow quickly as the number of clusters grows. Additionally, some of the features that are included in the model might not affect the sales of certain groups of products. The goal of the regularizer is to guard against fitting an overly complex model. The estimation procedure is likely to find a model that only uses the most important predictors of a cluster's sales. To account for sparsity, we add a regularization penalty to the objective, which has the added benefit of producing a model that is robust to measurement error (Bertsimas and Copenhaver 2017).

Now, if we want to forecast sales in accordance with model (2.1), we need to estimate each cluster forecasting model as well as to which cluster each product is assigned. Our CWR algorithm runs as follows:

1. In the first step, we estimate each cluster forecasting model by using past products' sales data and feature data, y_i and x_i. In this stage, we assign past products to clusters, $\widehat{z}_{ik}$, and determine the parameters of each cluster forecasting model, $\widehat{f}_k$.

2. In the second step, we estimate the cluster assignment model by using past products' cluster assignments and feature data, $\widehat{z}_{ik}$ and x_i. In this phase, we determine the parameters of the cluster assignment model, $\widehat{p}_k$.

3. In the final step, we plug the features of the new product, x_0, into each cluster forecasting model as well as the cluster assignment model, and combine these cluster forecasts and cluster assignments to predict the sales of the new product, $\widehat{y}_0$.

The details of each stage depend on the proposed sales generating model. In what follows, we expand on what happens in each of these steps.

2.3.2 Estimation for Past Products

Before we can forecast sales, we have to describe the estimation procedure of the sales generating model. For model (2.1), we need to decide in which cluster a product lies, $\widehat{z}_{ik}$, and estimate each cluster forecasting model, $\widehat{f}_k$. The general CWR problem is formulated as the following mixed integer non-linear optimization problem (P):

$$\min_{z_{ik}, f_k} \sum_{i=1}^{n} L\left(y_i, \sum_{k=1}^{\ell} z_{ik} f_k(x_i)\right) + \lambda R(f_1, \ldots, f_\ell) \tag{2.2a}$$

$$\text{s.t.} \sum_{k=1}^{\ell} z_{ik} = 1, \quad i = 1, \ldots, n \tag{2.2b}$$

$$z_{ik} \in \{0, 1\}, \quad i = 1, \ldots, n, \quad k = 1, \ldots, \ell. \tag{2.2c}$$

The objective (2.2a) represents the minimization of regularized prediction error. For each past product i, we observe y_i sales and forecast $\sum_{k=1}^{\ell} z_{ik} f_k(x_i)$ sales. For any error in this prediction we incur a loss $L\left(y_i, \sum_{k=1}^{\ell} z_{ik} f(x_i)\right)$. In addition, we regularize the cluster forecasting models through a regularizer $R(f_1, \ldots, f_\ell)$ and a penalty parameter $\lambda \geq 0$ that balances the loss and regularizer. The form of the loss, L, and regularizer, R, largely depends on the forecasting model, f_k, that is chosen. For example, to estimate the parameters of a LASSO regularized linear regression model (as in our application), we use the squared error loss with a regularizer that sums the absolute values of the regression parameters. Together, the constraints (2.2b) and (2.2c) ensure that each product gets assigned to exactly one cluster.

2.3.3 Forecasting for New Products

Having estimated the sales generating model (2.1), we can now forecast sales for the new product, y_0. However, we still need to decide to which cluster the new product belongs, as past products are clustered based on actual sales data which is unavailable for the new product.

For this, we can use any multiclass classification method such as a multinomial logistic regression (which we use in our application), support vector machines, classification trees, and random forests. We propose to train this cluster assignment model by using the cluster assign-

ments for past products, $\widehat{z}_{ik}$, as the dependent variable, and the features of past products, x_i, as the independent variables. This creates a mapping $\widehat{p}_k$ from the product feature space (excluding sales) to the clusters. Depending on the classification method, its predicted assignment for the new product, $\widehat{z}_{0k} = \widehat{p}_k(x_0)$, will either be in the form of an assignment to a cluster or probabilities of assignment to clusters. Both cases can be captured by letting $\widehat{p}_k(x_0)$ give the probability that the new product with features x_0 lies in cluster k. In the case where the classification method gives a pure cluster assignment the cluster's corresponding probability is set to 1, while others are set to 0. In either case, the new product sales forecast is given by

$$\widehat{y}_0 = \sum_{k=1}^{\ell} \widehat{z}_{0k} \widehat{f}_k(x_0) = \sum_{k=1}^{\ell} \widehat{p}_k(x_0) \widehat{f}_k(x_0). \tag{2.3}$$

When $\widehat{p}_k(x_0)$ assigns probabilities to clusters, the sales forecast in (2.3) is a weighted average of the cluster forecasts weighted by the cluster probabilities.

2.4 Application of Linear Cluster-While Regress Model

In this section, we specify the cluster forecasting model, f_k, and cluster assignment model, p_k, that we used in collaboration with our industry partners. In the real-world applications considered in this work, we consider f_k to be a LASSO regularized linear regression model, and we use a multinomial logistic regression to estimate p_k. This section will cover the linear model, the linear CWR algorithm, and an analysis of the forecasting error of our model and algorithm.

2.4.1 Linear Model

Formally, we consider the following linear cluster forecasting model $f_k(x_i) = \sum_{j=1}^{m} \beta_{kj} x_{ij}$ where β_{kj} is the linear regression parameter associated with product feature j in cluster k. The sales generating model (2.1) can then be rewritten as the following linear sales generating model:

$$y_i = \sum_{k=1}^{\ell} z_{ik} \sum_{j=1}^{m} \beta_{kj} x_{ij} + \epsilon_i, \quad i = 0, 1, \ldots, n, \tag{2.4}$$

where we assume that $\epsilon_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$ for all $i = 0, 1, \ldots, n$. We note that any of our results can be extended to the case where ϵ_i follows a Subgaussian distribution. The linear sales generating

model can also be given by the following vector representation:

$$y = (Z * X)\beta + \epsilon, \tag{2.5}$$

Here, $y = (y_1, \ldots, y_n) \in \mathbb{R}^n$ is a column vector of product sales, $X = (x_{ij}) \in \mathbb{R}^{n \times m}$ is a block matrix whose blocks are rows of product features, $Z = (z_{ik}) \in \{0, 1\}^{n \times \ell}$ is a block matrix whose blocks are rows of cluster assignments, $\beta = (\beta_{11}, \beta_{12}, \ldots, \beta_{1m}, \beta_{21}, \ldots, \beta_{\ell m}) \in \mathbb{R}^{\ell m}$ is a column vector that stacks the regression coefficients (this vector is s-sparse if at most s elements of β are non-zero), and $\epsilon = (\epsilon_1, \ldots \epsilon_n) \in \mathbb{R}^n$ is a column vector of zero-mean random error. In this representation, we use the Khatri-Rao product $Z * X$, which is defined as follows:

$$Z * X = \begin{pmatrix} z_{11} & z_{12} & \ldots & z_{1\ell} \\ \hline z_{21} & z_{22} & \ldots & z_{2\ell} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline z_{n1} & z_{n2} & \ldots & z_{n\ell} \end{pmatrix} * \begin{pmatrix} x_{11} & x_{12} & \ldots & x_{1\ell} \\ \hline x_{21} & x_{22} & \ldots & x_{2\ell} \\ \hline \vdots & \vdots & \ddots & \vdots \\ \hline x_{n1} & x_{n2} & \ldots & x_{n\ell} \end{pmatrix} =$$

$$\begin{pmatrix} z_{11}x_{11} & z_{11}x_{12} & \ldots & z_{11}x_{1m} & z_{12}x_{11} & z_{12}x_{12} & \ldots & z_{1\ell}x_{1m} \\ \hline z_{21}x_{21} & z_{21}x_{22} & \ldots & z_{21}x_{2m} & z_{22}x_{21} & z_{22}x_{22} & \ldots & z_{2\ell}x_{2m} \\ \hline \vdots & \vdots & \ddots & \vdots & & & & \\ \hline z_{n1}x_{n1} & z_{n1}x_{n2} & \ldots & z_{n1}x_{nm} & z_{n2}x_{n1} & z_{n2}x_{n2} & \ldots & z_{n\ell}x_{nm} \end{pmatrix}.$$

Under the assumption that the conditional expected sales is also a linear function of the dependent variables, we use $Z^* = (z^*_{ik})$ and $\beta^* = (\beta^*_{kj})$ to denote the cluster assignments and regression parameters of the true model, while our estimates are denoted by $\widehat{Z} = (\widehat{z}_{ik})$ and $\widehat{\beta} = (\widehat{\beta}_{kj})$. Additionally, we consider the case where the true model might be sparse. Specifically, we let $S \subset \{(1,1), (1,2), \ldots, (1,m), (2,1), \ldots, (\ell, m)\}$ contain the indices of the s non-zero regression parameters in the true model, and let β_S denote the vector that contains β_{kj} for the indices $(k, j) \in S$ and 0 otherwise.

2.4.2 Linear CWR Algorithm

In what follows, we adapt our CWR algorithm to the linear model. In the first step, the algorithm uses the LASSO regularized linear regression model to estimate each cluster forecasting model. In the second step, the algorithm uses a multinomial logistic regression model to estimate a cluster assignment model. As a result, the algorithm can determine the sales forecast

for the new product.

First, we have to describe how we estimate the cluster forecasting models based on the data of past products. As we consider each cluster forecasting model to follow a linear regression model, we use the ordinary least squares estimation method, which implies that $L(y_i, \widehat{y}_i) = (y_i - \widehat{y}_i)^2$. To this, we add LASSO regularization on the linear regression coefficients, which means that $R(\beta_1, \ldots, \beta_\ell) = \sum_{k=1}^{\ell} \sum_{j=1}^{m} |\beta_{kj}|$. Here, we consider LASSO regularization, but we note that the results below can be extended to the case of ridge regularization. In preliminary experimentation, we estimated the ridge regularized model and it produced worse results, plausibly due to overfitting. Generally, ridge regularization is not able to exclude unimportant variables while LASSO regularization can generate sparse models (Bühlmann and van de Geer 2011).

Having specified f_k, L, and R, we adapt problem (P) to formulate the linear CWR problem that estimates the linear sales generating model (2.4) as the following mixed integer non-linear optimization problem (P_L):

$$\min_{z_{ik}, \beta_{kj}} \quad \sum_{i=1}^{n} \left(y_i - \sum_{k=1}^{\ell} z_{ik} \sum_{j=1}^{m} \beta_{kj} x_{ij} \right)^2 + \lambda \sum_{k=1}^{\ell} \sum_{j=1}^{m} |\beta_{kj}| \tag{2.6a}$$

$$\text{s.t.} \quad \sum_{k=1}^{\ell} z_{ik} = 1, \quad i = 1, \ldots, n \tag{2.6b}$$

$$z_{ik} \in \{0, 1\}, \quad i = 1, \ldots, n, \quad k = 1, \ldots, \ell. \tag{2.6c}$$

In this problem, we minimize the LASSO objective by determining the right cluster assignments and regression parameters. Earlier, we mentioned that problem (P) is hard due to its integer decision variables and possibly non-linear objective. Here, problem (P_L) shows that even a linear regression model without regularization has a non-linear objective, namely biquadratic. This makes the problem hard to solve, even for commercial solvers. In fact, in Megiddo and Tamir (1982), problem (P_L) is proven to be NP-hard for the case where $\lambda = 0$. Clearly, adding regularization generalizes the problem, which therefore remains NP-hard. Nonetheless, in Proposition 2.4.1, we show that problem (2.6) can be reformulated as a mixed-integer quadratic problem that commercial solvers are able to solve.

Proposition 2.4.1. The linear CWR problem can be reformulated as the following mixed-

integer quadratic optimization problem (P_{LR}), where M is a big constant:

$$\min_{z_{ik},\beta_{kj},q_{ikj},r_{kj}} \quad \sum_{i=1}^{n}\left(y_i - \sum_{k=1}^{\ell}\sum_{j=1}^{m} q_{ikj}x_{ij}\right)^2 + \lambda\sum_{k=1}^{\ell}\sum_{j=1}^{m} r_{kj} \tag{2.7a}$$

$$\text{s.t.} \quad \sum_{k=1}^{m} z_{ik} = 1, \quad i = 1,\dots,n \tag{2.7b}$$

$$-M(1-z_{ik}) \le q_{ikj} - \beta_{kj} \le M(1-z_{ik}), \quad i = 1,\dots,n, \quad k = 1,\dots,\ell, \quad j = 1,\dots,m \tag{2.7c}$$

$$-Mz_{ik} \le q_{ikj} \le Mz_{ik}, \quad i = 1,\dots,n, \quad k = 1,\dots,\ell, \quad j = 1,\dots,m \tag{2.7d}$$

$$r_{kj} \ge \beta_{kj}, \quad k = 1,\dots,\ell, \quad j = 1,\dots,m \tag{2.7e}$$

$$r_{kj} \ge -\beta_{kj}, \quad k = 1,\dots,\ell, \quad j = 1,\dots,m \tag{2.7f}$$

$$z_{ik} \in \{0,1\}, \quad i = 1,\dots,n, \quad k = 1,\dots,\ell. \tag{2.7g}$$

Proof. Proof. See Appendix A.1. □ □

Second, we can now describe how we estimate the cluster assignment model based on our previous cluster assignments of past products. The cluster assignment model is estimated by fitting a multinomial logistic regression of the past product cluster assignments $\widehat{z}_{ik}$, on the available product features x_i (also available for the new product x_0). In particular, this estimation results in the multinomial logistic regression coefficients $\widehat{\gamma}_{kj}$ which define the following cluster assignment probabilities of the new product:

$$\widehat{z}_{0k} = \frac{\exp\left(\sum_{j=1}^{m}\widehat{\gamma}_{kj}x_{0j}\right)}{\sum_{k'=1}^{\ell}\exp\left(\sum_{j=1}^{m}\widehat{\gamma}_{k'j}x_{0j}\right)}. \tag{2.8}$$

Finally, plugging (2.8) into (2.3), the new product forecast becomes

$$\widehat{y}_0 = \sum_{k=1}^{\ell}\frac{\exp\left(\sum_{j=1}^{m}\widehat{\gamma}_{kj}x_{0j}\right)}{\sum_{k'=1}^{\ell}\exp\left(\sum_{j=1}^{m}\widehat{\gamma}_{k'j}x_{0j}\right)}\sum_{j=1}^{m}\widehat{\beta}_{kj}x_{0j}. \tag{2.9}$$

Thus, to forecast sales described by model (2.4), we combine these stages in our linear CWR algorithm that takes the following approach:

1. In the first step, we solve (P_{LR}) using the past products' sales data and feature data, y_i

and x_i, to find cluster assignments, $\widehat{z}_{ik}$, and linear regression parameters, $\widehat{\beta}_{kj}$.

2. In the second step, we fit a multinomial logistic regression using past products' cluster assignments and feature data, $\widehat{z}_{ik}$ and x_i, to find the logistic regression parameters, $\widehat{\gamma}_{kj}$.

3. In the final step, we plug the features of the new product, x_0, into each cluster forecasting model as well as the cluster assignment model, and combine these cluster forecasts and cluster assignments to predict the sales of the new product, $\widehat{y}_0$.

This process assumes that an initial number of clusters ℓ and the penalty parameter λ have been chosen. As we described before, we tune these parameters through the train-validate-test split. By running the algorithm on a training set and selecting those parameters that result in the best out-of-sample forecasting metrics on a validation set, we obtain the model to be analyzed on the test set. Note that we rescale our data such that $||X_j||_2 = 1$ in order to avoid implementation issues.

Unfortunately, in some cases, solving (P_{LR}) can be a time-consuming process. For example, when using Gurobi 7.0.2 to solve 50 randomized instances of (P_{LR}) programmed in Julia/JuMP (Dunning et al. 2017) on an Intel Core i5-4690K @ 3.5GHz CPU and 8 GB RAM, the average running time is 0.19 seconds when $n = 10$, $m = 5$, $\ell = 2$, but it grows to over 10 minutes when $n = 100$, $m = 5$, $\ell = 2$. In the following, we describe the approximate linear CWR algorithm that on average takes 0.02 seconds and 0.04 seconds to solve the same instances. In fact, it scales well to larger instances as its running time is 8.74 seconds for $n = 10000$, $m = 100$, $\ell = 10$, and 27.72 seconds for $n = 10000$, $m = 200$, $\ell = 20$.

In the approximate linear CWR algorithm, the second and final step remain the same, but we use an approximate method in the first step. Instead of solving a mixed-integer quadratic optimization problem, this approximate method take ideas from iterative greedy algorithms. The approximation initializes by assigning each past product to a random cluster, after which it can fit a LASSO regularized linear regression model for these randomized cluster. Then, in each iteration, it assigns a past product to the cluster whose LASSO regularized linear regression model gives the best forecast for that product, and it fits a LASSO regularized linear regression model for the updated clusters. To be more specific, the approximate linear CWR algorithm has the following procedure:

1. In the first step, we approximate (P_{LR}) using the past products' sales data and feature data, y_i and x_i, through the following approximate method:

   a. Initialize the assignment of products to clusters $\widehat{z}_{ik}^{(0)}$ randomly.

   b. Iteratively re-estimate the cluster forecasting model and re-cluster the products. For iteration $t = 1, \ldots, T$:

      i. For cluster $k = 1, \ldots, \ell$, fit a LASSO regression of sales on the product features of products in cluster k, i.e., fit LASSO regression of y_i on x_i for i such that $\widehat{z}_{ik}^{(t-1)} = 1$ to obtain $\widehat{\beta}_{kj}^{(t)}$.

      ii. For product $i = 1, \ldots, n$, compute the distance between product i's sales and each cluster's forecast and assign product i to the closest cluster, i.e., for $k = \arg\min_{k'}(y_i - \sum_{j=1}^{m} \widehat{\beta}_{k'j}^{(t)} x_{ij})^2$ set $\widehat{z}_{ik}^{(t)} = 1$ and $\widehat{z}_{ik'}^{(t)} = 0$ for $k' \neq k$.

      iii. Terminate with $\widehat{z}_{ik}$ and $\widehat{\beta}_{kj}$ if $t = T$ or $\widehat{z}_{ik}^{(t)} = \widehat{z}_{ik}^{(t-1)}$, otherwise proceed to $t + 1$.

2. In the second step, we fit a multinomial logistic regression using past products' cluster assignments and feature data, $\widehat{z}_{ik}$ and x_i, to find the logistic regression parameters, $\widehat{\gamma}_{kj}$.

3. In the final step, we plug the features of the new product, x_0, into each cluster forecasting model as well as the cluster assignment model, and combine these cluster forecasts and cluster assignments to predict the sales of the new product, $\widehat{y}_0$.

This gives us two methods to forecast sales of new products that follow the linear CWR model. The first linear CWR algorithm optimally solves the linear CWR problem, while the second approximate linear CWR algorithm finds a good approximation to the linear CWR problem. On the other hand, the optimal algorithm takes longer to run than the approximate algorithm. After developing these algorithms, we want to test their theoretical and practical performance.

2.4.3 Forecasting Error Analysis

In what follows, we present theoretical guarantees on the forecasts of the linear CWR algorithm. We would like to note that these guarantees extend to the approximate linear CWR algorithm in many cases. Specifically, our computational results indicate that the guarantees effectively hold in 85% to 92% of simulated instances depending on the parameter settings of these instances.

For the linear CWR algorithm, we prove that the forecasting error is bounded, both in-sample for past products in the train data as well as out-of-sample for new products in a test set. In proving these probabilistic guarantees, we use Lemma 2.4.2 to show that the forecasting error coming from noisy measurements is small with high probability.

Lemma 2.4.2. If $||X_j||_2 = 1$ and $\lambda = 4\sigma\sqrt{\frac{2}{n}\log\left(\frac{2nm\ell}{\delta}\right)}$, then for any allowable Z and $0 < \delta < 1$,

$$\mathbb{P}\left(\frac{1}{n}||\epsilon^T(Z * X)||_\infty \leq \frac{\lambda}{4}\right) \geq 1 - \delta.$$

Proof. Proof. See Appendix A.1. □ □

In order to tighten the results, we require the train data X to follow the cluster compatibility condition, which specifically puts a condition on the minimum eigenvalue of X^TX. To state the definition, let $\beta^\Delta = \widehat{\beta} - \beta^*$ and $Z^\Delta = \hat{Z} - Z^*$ denote the estimation error in the regression parameters and in the cluster assignments. Additionally, recall that $S \subset \{(1,1),(1,2),\ldots,(1,m),(2,1),\ldots,(\ell,m)\}$ consists of the indices corresponding to the s non-zero regression parameters in the true model, as well as that β_S consists of β_{kj} if $(k,j) \in S$ and 0 otherwise.

Cluster Compatibility Condition *The cluster compatibility condition is satisfied if for all β^Δ and Z^Δ satisfying $\|\beta^\Delta_{S^C}\|_1 \leq 3\|\beta^\Delta_S\|_1 + 2\|\beta^*\|$ there exists $\theta > 0$ such that*

$$\|\beta^\Delta_S\|_1 \leq \frac{s}{n\theta^2}\left(\|(Z^* * X)\beta^\Delta\|_2^2 - \|(Z^\Delta * X)\beta^\Delta\|_2^2 - 2\|(Z^* * X)\beta^*\|_2^2\right) \tag{2.10}$$

Both the cluster compatibility condition in (2.10) and the regular compatibility condition of Bühlmann and van de Geer (2011) are related to the minimum eigenvalue of the train data. In particular, if there are no incorrect cluster assignments, then the second term of (2.10) goes to 0, which reduces the cluster compatibility condition to the general compatibility condition. Nevertheless, our condition remains stronger due to an additional constant term. This accounts for the fact that our algorithm needs to learn clusters for both old and new products.

Interestingly, we can show that instead of imposing the cluster compatibility condition, we can instead assume the regular compatibility condition and a bound on the number of incorrect cluster assignments. Proposition 2.4.3 proves that the regular compatibility condition is equivalent to the cluster compatibility condition under a limit on incorrect cluster assignments.

Proposition 2.4.3. If $||X_j||_2 = 1$, and the number of incorrect cluster assignments $r < \left(\frac{n}{2m}\right)\left(\frac{\beta^{\Delta}_{min}}{\beta^{\Delta}_{max}}\right)^2$, where $\beta^{\Delta}_{min} = \min_{k,j}|\beta^{\Delta}_{kj}|$ and $\beta^{\Delta}_{max} = \max_{k,j}|\beta^{\Delta}_{kj}|$, then there exists $\kappa > 0$ such that

$$\|(Z^* * X)\left(\beta^* - \widehat{\beta}\right)\|_2 - \|(Z^{\Delta} * X)(\widehat{\beta} - \beta^*)\|_2 = \kappa\|(Z^* * X)\left(\beta^* - \widehat{\beta}\right)\|_2 .$$

In addition, if for all β^{Δ} and Z^{Δ} satisfying $\|\beta^{\Delta}_{S^C}\|_1 \leq 3\|\beta^{\Delta}_{S}\|_1 + 2\|\beta^*\|$ there exists $\eta > 0$ such that

$$\|\beta^{\Delta}_{S}\| \leq \frac{s}{n\eta^2}\left(\|(Z^* * X)\beta^{\Delta}\|_2^2 - \frac{2}{\kappa}\|(Z^* * X)\beta^*\|_2^2\right),$$

then there exists $\theta > 0$ such that

$$\|\beta^{\Delta}_{S}\| \leq \frac{s}{n\theta^2}\left(\|(Z^* * X)\beta^{\Delta}\|_2^2 - \|(Z^{\Delta} * X)\beta^{\Delta}\|_2^2 - 2\|(Z^* * X)\beta^*\|_2^2\right) .$$

Proof. See Appendix A.1. □

2.4.4 Bound on In-Sample Forecasting Error

First, we investigate the in-sample performance of our estimated model against the true model. Specifically, we present a bound on the in-sample mean squared forecasting error:

$$\frac{1}{n}||y^* - \widehat{y}||_2^2 = \frac{1}{n}||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2, \tag{2.11}$$

where y^* and $\widehat{y}$ contain the expected sales of each past product under the true and estimated linear CWR models, respectively. As we consider the in-sample error, X is the train data, $\widehat{Z}$ is the cluster assignments given by the linear CWR algorithm, and $\widehat{\beta}$ is the regression parameters given by the linear CWR algorithm.

The mean squared forecasting error is a natural measure for the difference between the forecasts from our estimated model and the true model. This difference is caused by the difficulty in estimating z^*_{ik} and β^*_{kj} exactly, which comes from the fact that we gather noisy measurements (due to ϵ_i) instead of the true conditional expected sales. Our theoretical results show that, with high probability, the in-sample mean squared forecasting error (2.11) is bounded. Theorem 2.4.4 proves this probabilistic guarantee on the in-sample mean squared forecasting error (2.11),

which shows that the linear CWR algorithm produces statistically consistent forecasts.

Theorem 2.4.4. Consider the linear sales generating model (2.5) and let $\widehat{Z}$ and $\widehat{\beta}$ be the estimates of Z^* and β^* generated by the linear CWR algorithm. Let $||X_j||_2 = 1$ and $\lambda = 4\sigma\sqrt{\frac{2}{n}\log\left(\frac{2nm\ell}{\delta}\right)}$, then the following probabilistic bound holds for any $0 < \delta < 1$,

$$\mathbb{P}\left(\frac{1}{n}||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2 \leq \frac{5}{2}\lambda||\beta^*||_1\right) \geq 1 - \delta.$$

In addition, under the cluster compatibility condition,

$$\mathbb{P}\left(\frac{1}{n}||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2 + \lambda||\widehat{\beta} - \beta^*||_1 \leq 4\lambda^2\frac{s}{\theta^2} + 2\lambda||\beta^*||_1\right) \geq 1 - \delta.$$

Proof. See Appendix A.1. □

For the first bound in Theorem 2.4.4, the only required assumption is that $||X_j||_2 = 1$, which can always be achieved by simply rescaling the columns of X. For the second bound in Theorem 2.4.4, we require an additional assumption, namely the cluster compatibility condition. However, this condition grants us an additional bound on the estimation error between the estimated regression parameters $\widehat{\beta}$ and the true regression parameters β^*. Generally, the bounds in Theorem 2.4.4 show that the forecasts of the linear CWR algorithm are consistent, even without the cluster compatibility condition, because $\log(n)/n$ converges to 0 as n increases.

To illustrate and compare these results, Figure 2.2 presents the bounds as a function of the number of observations n. The figure on the left shows the case where the bounds hold with 90% probability ($\delta = 0.10$) and the figure on the right for 99% probability ($\delta = 0.01$). The solid curves represent the first bound, whereas the dotted curves illustrate the second bound. The red curves form the bound for $\ell = 2$ clusters, and the blue curves form the bound for $\ell = 10$ clusters. The other parameters are given by $m = 10$, $||\beta^*||_1 = 10$, $\sigma = 1$, $s = 1$, and $\theta = 2$.

The main observation is that each forecasting error bound declines as the number of observations for past products n increases. Initially, the bound decreases rapidly, which indicates that the algorithm is accurate even for small datasets. We also observe the consistency of our algorithm's estimates. As more and more data becomes available the estimated model converges to the true model, and hence, the prediction error converges to 0. Additionally, we observe that the algorithm's forecasting error is nearly identical when there are two or more clusters. Finally, comparing the two figures, the forecasting error is similar when the probability with which the

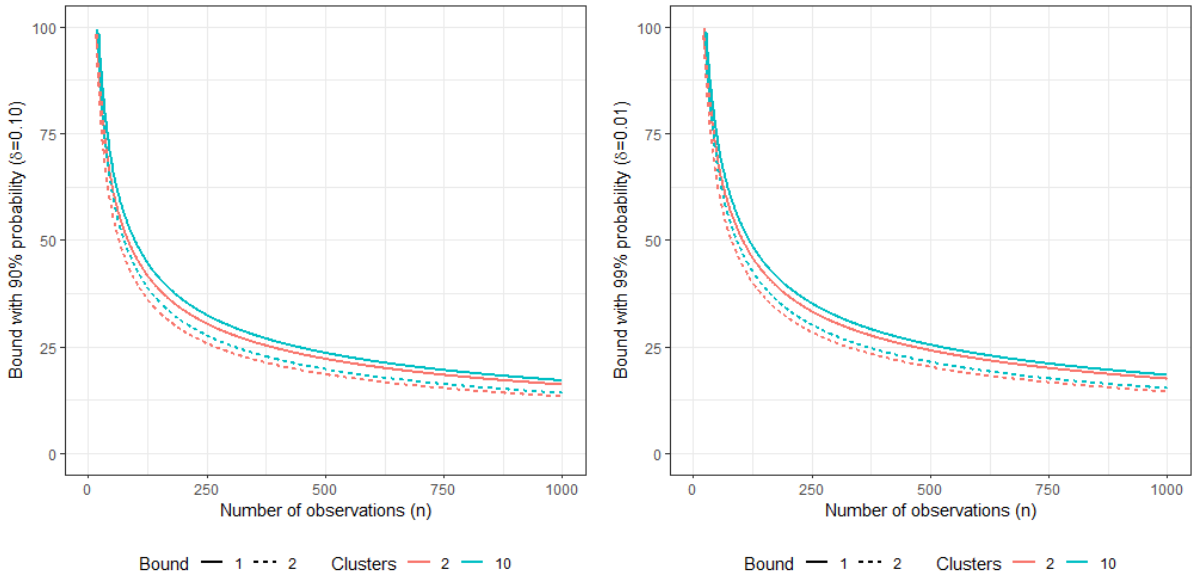

Figure 2.2: Probabilistic bounds on the mean squared forecasting error as the number of observations (n) changes for several numbers of clusters (ℓ) and probability to exceed the bound ($\delta = 0.10$ on the left, $\delta = 0.01$ on the right)

bound holds is increased. Each of the bounds on the left ($\delta = 0.10$) increase by at most 20% on the right ($\delta = 0.01$).

2.4.5 Bound on Out-of-Sample Forecasting Error

Next, we analyze the out-of-sample performance of the proposed algorithm. In what follows, we present a bound on the out-of-sample absolute forecasting error:

$$||y_0^* - \widehat{y}_0||_1 = ||\sum_{k=1}^{\ell} z_{0k}^* \sum_{j=1}^{m} \beta_{kj}^* x_{0j} - \sum_{k=1}^{\ell} \widehat{z}_{0k} \sum_{j=1}^{m} \widehat{\beta}_{kj} x_{0j}||_1, \tag{2.12}$$

where y_0^* and $\widehat{y}_0$ are expected sales of the new product under the true and estimated linear CWR models, respectively. Note that for the out-of-sample error, x_{0j} will be the new product's data, $\widehat{z}_{0k}$ will represent a forecasted cluster assignment, and $\widehat{\beta}_{kj}$ will remain the regression parameters given by the linear CWR algorithm. Regarding the value of $\widehat{Z}$, our forecasting approach uses cluster assignment probabilities based on multinomial logistic regression, as in equation (2.8). Instead, for our out-of-sample analysis, we will use cluster assignment probabilities that are equal for each cluster, as in $\widehat{z}_{0k} = 1/\ell$. Lastly, we note that the new product's true cluster assignment is denoted by k^*, as in $z_{0k^*}^* = 1$ and $z_{0k}^* = 0$ for $k \neq k^*$.

The absolute forecasting error measures the difference between forecasts, but to prove a bound, we relate it to the estimation error. In particular, we rewrite the out-of-sample fore-

casting error in terms of the in-sample forecasting error, after which we can use the result in Theorem 2.4.4 to show that the out-of-sample forecasting error (2.12) is bounded with high probability. Theorem 2.4.5 proves this probabilistic guarantee on the out-of-sample absolute forecasting error (2.12).

Theorem 2.4.5. Consider the linear sales generating model (2.5) and let $\widehat{Z}$ and $\widehat{\beta}$ be the estimates of Z^* and β^* generated by the linear CWR algorithm. Let $\|X_j\|_2 = 1$, let $C = \max_{k',k''} \|\beta^*_{k'} - \beta^*_{k''}\|_1$, and $\lambda = 4\sigma\sqrt{\frac{2}{n}\log\left(\frac{2nm\ell}{\delta}\right)}$, then under the cluster compatibility condition, the following probabilistic bound holds for any new product feature vector x_0 and any $0 < \delta < 1$,

$$\mathbb{P}\left(||\sum_{k=1}^{\ell} z^*_{0k}\sum_{j=1}^{m}\beta^*_{kj}x_{0j} - \sum_{k=1}^{\ell}\widehat{z}_{0k}\sum_{j=1}^{m}\widehat{\beta}_{kj}x_{0j}||_1 \leq C + \frac{2}{\ell}\|\beta^*\|_1 + \frac{4}{\ell}\lambda\frac{s}{\theta^2}\right) \geq 1-\delta.$$

Proof. See Appendix A.1. □

Theorem 2.4.5 that the out-of-sample error decreases as the number of training samples n increases. Nevertheless, due to the uniform probability of assigning a test point to any of the clusters, the worst case prediction error will remain a function of how far true parameters of different clusters are (i.e. parameter C in Theorem 2.4.5).

2.5 Computational Experiments

In this section, we evaluate the results of our computational experiments. We analyze the performance of our linear Cluster-While-Regress (CWR) algorithm. We compare it against several benchmark algorithms including Regularized Linear Regression (LASSO), Cluster-Then-Regress (CTR), Random Forests (RF), and Gradient Boosted Trees (GBT). Our algorithm was coded in Python, and the benchmark algorithms were imported from the scikit-learn library for Python.

2.5.1 Benchmark Algorithms

First, we describe the various benchmark algorithms that we used to compare our approach against. The first two benchmarks are based on our linear CWR algorithm, the third benchmark is a recently developed algorithm for the clusterwise regression problem, while the last two benchmarks come from advanced machine learning. For any benchmark, we use the same train-validate-test split as for our linear CWR algorithm. In particular, we fit each method to the

train data, tune the model's tuning parameters on the validation data, and analyze the results on the test data.

Regularized Linear Regression (LASSO): Regularized linear regression, specifically LASSO, is closely related to our linear CWR algorithm. While our linear CWR algorithm dynamically clusters and regresses sales for different products, another simpler approach is to assume that all products belong to the same single cluster. In the linear case, this simplifies our problem to the well understood LASSO regression. Therefore, we consider LASSO regression as one of the benchmark algorithms. In our application, we tune the LASSO regularization parameter.

Cluster-Then-Regress (CTR): The Cluster-Then-Regress algorithm follows a stepwise method to clustering and regression. In the first step, we can cluster products using any clustering method such as k-means or randomized clustering which allows us to use the multiple restart method. After clustering, one can then fit separate demand models such as LASSO regression to each cluster. The forecast comes from weighting the forecast of each cluster, similar to our linear CWR algorithm. We consider this algorithm as one of the benchmark algorithms, because both our industry partners used slight modifications of this stepwise approach. In applying this model, we initialize using randomized clustering, where we tune the seed of this initial clustering, as well as the number of clusters, and the LASSO regularization parameter.

Column-Generation for Clusterwise Regression (CGCR): Clusterwise Regression is a problem for which algorithms have been developed previously. Recently, Park et al. (2016) have developed a heuristic method that uses column-generation to solve the linear CWR problem without regularization. For a fair assessment, we adapt their algorithm to account for regularization. We include this column-generation based algorithm in order to compare our linear CWR algorithm against previously developed methods for our problem. Our application tunes the number of clusters, and the LASSO regularization parameter.

Random Forests (RF): The Random Forest algorithm is a machine learning method that fits many randomized regression trees to a set of data. Each regression tree takes a random subset of the data and features, and determines the best way to split this data into groups that have as similar outcomes as possible. To produce a forecast, the average is taken of each tree's forecast for the new product. We compare against this algorithm due to its historical strengths

in practice. In our application, we tune the number of trees, the maximum depth of each tree, the minimum number of observations per leaf, and the minimum number of observations required to split a node.

Gradient Boosted Trees (GBT): The Gradient Boosted Tree algorithm is another approach from machine learning that successively refits regression trees to a set of data. Initially, a regression tree is fit and the error for each data point is computed. Afterwards, iteratively, new regression trees are fit that specifically weigh the previously erroneous data points more heavily. The new product's forecast comes from averaging the successively built trees. We use this algorithm as a benchmark as it is an advanced machine learning algorithm. In the application of this model, we tune the maximal depth of the tree, and the stepsize of the algorithm.

2.5.2 Data Generation

Next, we describe the data that was generated for the computational experiments. In our computations, we consider multiple parameter settings. For the number of past products n we experiment with the values 100 and 200, while we consider 5 and 10 for the number of features m and number of clusters ℓ. The sales of each product i are generated according to the initially proposed sales generating model in (2.4):

$$y_i = \sum_{k=1}^{\ell} z_{ik} \sum_{j=1}^{m} \beta_{kj} x_{ij} + \epsilon_i, \quad i = 0, 1, \ldots, n,$$

where the errors of the model ϵ_i are drawn from a normal distribution with mean 0 and standard deviation 50, the product features x_{ij} are drawn from a uniform distribution on $[0, 1]$, the cluster regression parameters β_{kj} are fixed, and the cluster assignments z_{ik} are such that each cluster contains a $1/\ell$ fraction of the products. For each parameter setting, we draw 1000 random instances of the dataset, run the algorithms over each dataset, and average the results. In these parameter settings, we fix the cluster regression parameters to be the same across instances, which allows us to specifically generate sparse models (i.e., some of the β_{kj} are 0) in accordance with our initial modeling assumptions.

2.5.3 Results

As our first forecasting metric, we use the Mean Absolute Percentage Error (MAPE). The MAPE measures the relative difference between the actual and predicted sales, which means

that a lower MAPE implies better performance. If we assume that there are n products in our test set and that y_i denotes the actual sales while $\hat{y}_i$ denotes the predicted sales for the i'th product, then the MAPE of a set of predictions is given by:

$$\text{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|. \tag{2.13}$$

In Table 2.2, we present the MAPE of our algorithm as well as the benchmark algorithms when we fix the number of iterations and the number of restarts of the CWR algorithm to both equal 10.

Table 2.2: MAPE comparison of algorithms on experimental settings

n	**m**	ℓ	**CWR**	**LASSO**	**CTR**	**CGCR**	**RF**	**GBT**
100	5	5	0.1782	0.2026	0.1870	0.1811	0.2146	0.1889
100	10	10	0.1200	0.1561	0.1396	0.1622	0.1356	0.1356
200	5	5	0.1215	0.1494	0.1382	0.1297	0.1371	0.1249
200	10	10	0.1122	0.1645	0.1453	0.1438	0.1508	0.1253

We note that all algorithms have a small MAPE on these computational experiments, indicating their small error and good performance. Due to the structured sales generating model, we would expect statistics and machine learning algorithms to yield strong results. However, we note that our algorithm performs best among all algorithms. In particular, it outperforms random forests and gradient boosted trees, which are advanced machine learning tools for non-linear environments such as the one we encounter. This shows that by specifically exploiting the model structure, as our algorithm does, we can obtain significantly better forecasts. Furthermore, the CWR model is inspired from the current industry practice and the CWR which makes it easier to use by industry practitioners. Additionally, we observe that our linear CWR algorithm outperforms previously developed methods for the linear CWR problem without regularization, particularly the CGCR algorithm adapted to regularization.

For the second forecasting metric, we use the Weighted Mean Absolute Percentage Error (WMAPE). The WMAPE is similar to the MAPE except for the different weighing of observations. Instead of weighing each product in the test set equally, the WMAPE weighs products based on the magnitude of their sales. This means that products with higher sales have a higher weight. Using the same notation as before, the WMAPE of a prediction method is given by:

$$\text{WMAPE} = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right|\frac{y_i}{\sum_{i=1}^{n} y_i}. \tag{2.14}$$

In Table 2.3, we show the WMAPE of our algorithm against the benchmark algorithms when the number of iterations and the number of restarts of the linear CWR algorithm are also fixed to 10.

Table 2.3: WMAPE comparison of algorithms on experimental settings

n	**m**	ℓ	**CWR**	**LASSO**	**CTR**	**CGCR**	**RF**	**GBT**
100	5	5	0.1065	0.1236	0.1147	0.1135	0.1180	0.1166
100	10	10	0.0982	0.1264	0.1140	0.1319	0.1170	0.1087
200	5	5	0.1010	0.1237	0.1144	0.1080	0.1088	0.1036
200	10	10	0.0844	0.1253	0.1099	0.1106	0.1003	0.0888

In this table, we see that each algorithm has a small WMAPE in our experiments, which indicates a strong performance for any algorithm. This confirms the previous results, yet we see that the WMAPE is generally smaller than the MAPE. This indicates that our models have an especially strong forecasting performance for high selling products, which are often deemed more important.

Next, we test how robust our algorithm is to changes in its parameters. Table 2.4 presents both the MAPE and WMAPE of our algorithm when we vary both the number of iterations and the number of restarts of the linear CWR algorithm to be 5, 10, and 20.

Table 2.4: MAPE and WMAPE of CWR algorithm on experimental parameters

Iterations	**Restarts**	**MAPE**	**WMAPE**
5	5	0.1854	0.1073
5	10	0.1774	0.1065
5	20	0.1819	0.1053
10	5	0.1753	0.1070
10	10	0.1782	0.1065
10	20	0.1806	0.1056
20	5	0.1754	0.1071
20	10	0.1780	0.1065
20	20	0.1805	0.1055

From this table, we observe that the results of our algorithm are robust to changes to its parameters. Though the MAPE does not change predictably, we observe a slight improvement in the WMAPE whenever the number of restarts increases. The running time of the algorithm increases as these parameters increase, and hence, these results indicate that a good performance can be obtained without the need for a long running time.

2.6 Case Study: Johnson & Johnson Consumer Companies Inc.

In this section, we give a detailed description of our collaboration with Johnson & Johnson Consumer Companies Inc., one of the largest consumer goods manufacturers in the world. While we introduced the collaboration in Section 5.2 briefly, we will give an extensive description of the data collected, segments tested and performance metrics used. Afterwards, we discuss the forecasting tool that was developed for our collaborators.

2.6.1 Data Description

As mentioned before, Johnson & Johnson is highly invested in innovation of its product segments and releases new products frequently. As our partner, they provided us with sales and feature data of new products released in the past. In our analysis, we focus on data from product segments where most of the innovation occurred (i.e., the highest number of new product releases). This leads us to the following categories of products: facial care, body care, and baby care products.

For each category, we have access to monthly sales and feature data for a period of roughly 4 years. Through interactions with our industry partner, we realized that a product is considered new for the first 12 months of its lifecycle. Additionally, products that last less than four months are characterized as promotional versions of existing products. Thus, we subset our dataset to first remove products with a lifecycle of less than four months. We further subset the data to include only the first twelve months of a product's sales information. Next, we split the dataset into a train set containing the new products in the first two years of data, a validation set containing the new products in the third year of data, and a test set containing the new products in the last year of data. Then, we fit our prediction models using the linear CWR algorithm on the train set, use the validation set to tune the model parameters (regularization parameter λ and number of clusters ℓ), and analyze the models on the test set.

In our application, we use the logarithm of sales as our dependent variable and all features as our independent variables. In a preliminary analysis, we compared fitting sales as well as the logarithm of sales on the features, which showed substantially improved forecasting metrics when taking the logarithm of sales. As an additional benefit, this means that sales are an exponential function of the features, and hence, the sales forecasts are non-negative. Another interesting fact is that while the provided datasets contain sales data for the manufacturer's

brand, they also contain sales data for competing brands. This means, we can train our model on both the manufacturer's as well as the competitors' data and then test models on collaborator specific brand data.

The available features include product features such as average unit price, brand, claimed benefits, form (e.g., lotion, foam, powder, etc.), and time-based features such as the time since product introduction, month, season. Additionally, we have distribution information such as %ACV (Percentage of All Commodity Volume), number of stores selling the product, and we have promotion information such as usage of display promotions for the product, usage of feature promotions for the product.

It is important to note how we engineer our features related to promotion and distribution. While promotion and distribution are important drivers of product sales, these decisions are possibly dependent on customer response. For instance, a well received new product might see a jump in promotion budget and increased distribution amongst a variety of channels. In contrast, firms might reduce spending on promotion and distribution for products that have seen tepid response from consumers. As a consequence, correctly assessing promotion and distribution features for new products is almost as hard as forecasting sales for the new product itself. Therefore, we engineer features that give a rough indication of the level of promotion or distribution that a product receives. It is often easier to assess whether the budget for a new product will be low or high compared to the budgets that were allocated to past products.

Consider the display promotion feature, which describes whether the product in question was promoted using a display promotion in a given month. Given that we have historical feature and sales data, we have full hindsight information on when this product was promoted during the months after introduction. Clearly, monthly prediction of such features is hard. Hence, we will transform this feature into a new feature describing the intensity of display promotion usage over the first year of introduction. For each product we check whether it falls below the 33rd percentile, between the 33rd and 67th percentile, or above the 67th percentile of display promotion usage. Depending on where the product falls, we classify it as Low/Medium/High on the intensity of display promotions used. Thus, we have simplified the task of predicting monthly display based promotion to calculating an yearly promotion intensity indicator. Not only does this transformation let us use feature information that could not have been directly used, it can also provide insights on how display intensity can impact sales of the new product. Furthermore, this is a more stable feature in comparison to monthly features that can change

over the course of a product's introduction and hence might be hard to predict. We use similar transformation for all features related to promotion and distribution.

2.6.2 Results

To compare the performance of our linear CWR algorithm against the natural benchmark algorithms from practice, we fit these algorithms to the train data, tune their parameters on the validation data, and compute the forecasting metrics on the test data. Table 2.5 presents WMAPE of each algorithm when applied to the three different segments.

Table 2.5: WMAPE comparison of algorithms on segments of consumer goods products

Segment	**CWR**	**LASSO**	**CTR**	**CGCR**	**RF**	**GBT**
Baby Care	0.5869	0.6616	0.7925	1.7055	0.6846	0.6119
Body Care	0.4976	0.6592	0.5759	0.6339	0.5942	0.5894
Facial Care	0.4704	0.5174	0.5213	3.7895	0.4957	0.5117

Not only do we notice that our method yields the best performance, we also see that it beats the other methods by a considerable margin. Furthermore, the WMAPE lie below 0.6 regardless of the dataset considered. In turn, this indicates that our method is robust to different data settings. We explore the question of robustness further in Section 2.7. The two benchmarks that we consider are inspired by the current industry practice of either fitting a single regression model to the sales of products introduced in the past or using a two step approach of sequentially clustering products and fitting demand models. Our results clearly show that a simultaneous approach of jointly clustering and regressing results in considerable improvements in sales estimation.

As an additional forecasting metric, we consider the Bull's Eye Metric, which is commonly used by our partners. This measure compares the predicted sales with the actual sales by bucketing products together based on their percent accuracy with respect to the actual sales. Table 2.6 shows the Bull's Eye metric for the CWR method on six different datasets. The end points for different buckets have been created based on consultation with our industry partner.

Table 2.6: Bull's Eye Metric of CWR algorithm on segments of consumer goods products

Segment	**<50%**	**50-70%**	**70-130%**	**130-150%**	**>150%**
Baby Care	2	1	11	2	0
Body Care	3	7	11	1	3
Facial Care	4	6	19	2	9

The number 11 in the 70-130 bucket for Baby Care represents that our predictions were

within 70-130% of the actual sales for 11 out of the 16 products in the dataset. Ideally, the higher the number in the middle bucket, the better the prediction method is. Notice that most of the predictions lie within the desired bracket of 70-130%, which demonstrates the accuracy of our method for each individual product in these subsegments.

2.6.3 Implementation in Practice

Next, we describe the pilot implementation of the CWR algorithm at Johnson & Johnson. Our pilot tool was created with the objective of simplifying the sales forecasting of new products for managers while giving fast, easy to use and reliable sales predictions. In Figure 2.3, we describe the workflow of the pilot tool. We first apply the CWR algorithm on historical sales data of comparable products to create optimal clusters and prediction models. Then, we use Excel to make a user friendly prediction interface which can be used to make final sales predictions. In the first step, the CWR model works with high dimensional feature data and selects important features for sales predictions. Afterwards, we use these features in the Excel tool as input to make predictions. Figure 2.4 shows a screen-shot of the Excel tool.

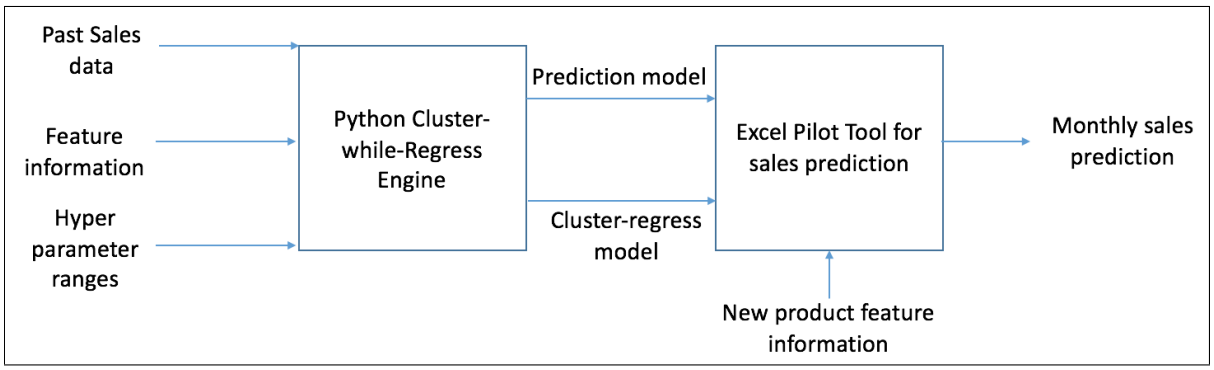

Figure 2.3: Workflow for live pilot testing

In this section we describe user related inputs that are needed for the Excel tool. The user is asked to provide new product feature information such as the brand, packaging size and unit price of the new product. We provide a range of values that all the input product features can take. This serves dual purposes: first, to provide the user an idea of the kind of values that the input can take, and second, to make sure that our predictions stay reliable and we do not extrapolate our linear models. We also ask the user to provide information on product marketing and the promotion budget as the marketing effort is a decision that can have a large impact on the eventual sales of the new product.

This includes deciding on display promotions, feature promotions, and display and feature

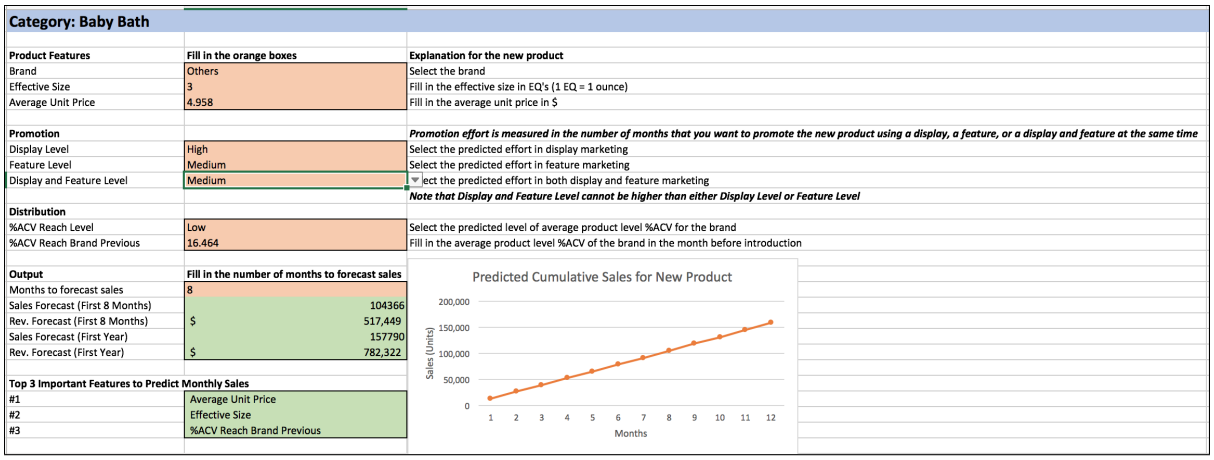

Figure 2.4: Screenshot of the Excel tool developed for live testing

promotions which includes periods when a product is promoted both through display as well as feature advertising. As explained in the previous section, all these are transformed into intensity indicator variables that compare the level of promotion or distribution in comparison to other products within the same brand. The user inputs Low/Medium/High levels for these features comparing the anticipated levels with those of existing products within the same brand. Similar transformation is also done for distribution related features such as %ACV and others.

We next use the feature information and the already generated clusters and prediction models to make monthly predictions. Note that our task here is to make accurate first year predictions. Nevertheless, in order to make the predictions more interpretable, our tool also illustrates the monthly predictions generated from our model (Figure 2.5).

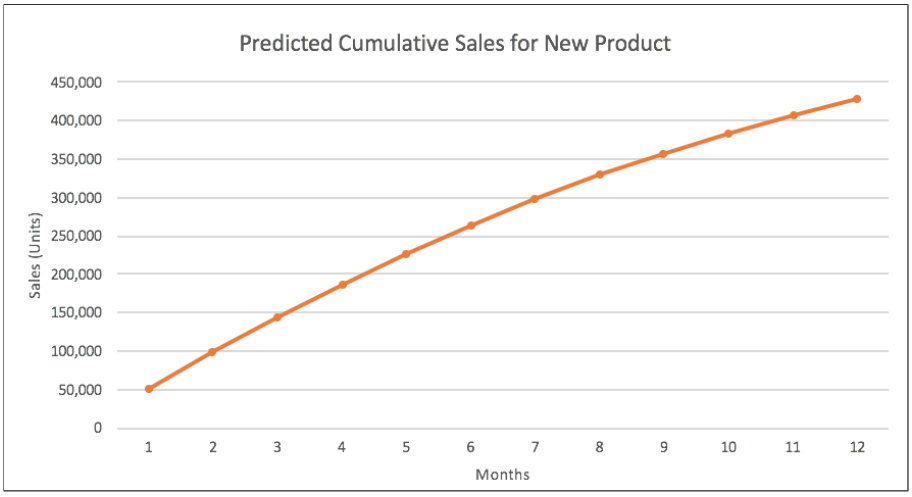

Figure 2.5: Sample output for a new product with user input features. The model predicts that the sales will slow down as we come close to the end of the introductory period.

As a by product of our prediction model, we also create a list of the predictors that are most predictive of sales for the given input feature values of the new product. This is created

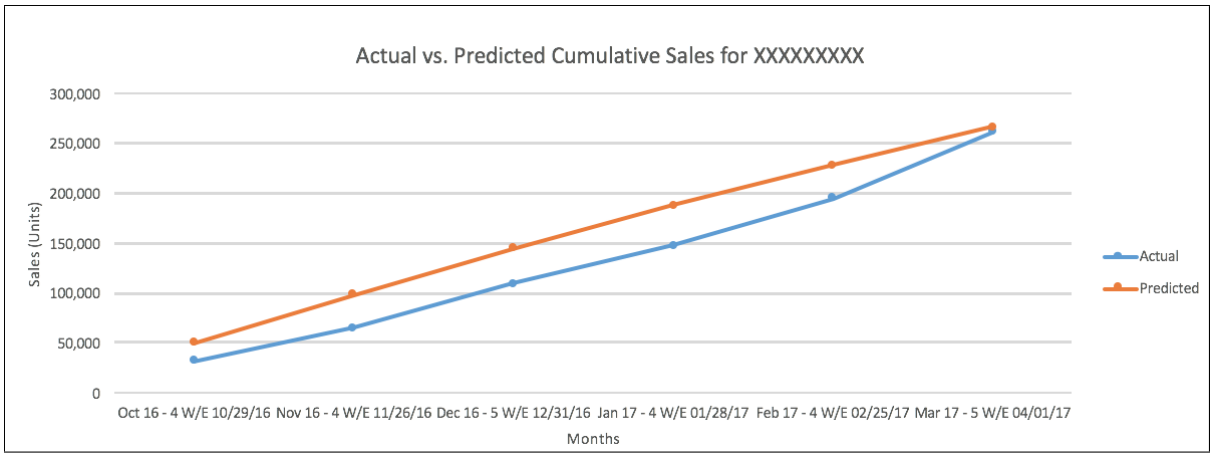

Figure 2.6: Sample output comparing predicted and actual monthly sales

by evaluating the derivative of the forecasting model (2.9) presented in Section 2.4. This list is useful in guiding important decisions such as pricing or packaging of the product before its introduction to the market. We note that the Excel tool created is very simple to use, descriptive and can guide decision making in various operations issues in new product management. While the creation of cluster and regression model is computationally more expensive, given that new product sales data is weekly. Hence, the Excel tool can be very easily updated to incorporate new features and data into the prediction interface.

We asked our industry partner to use the pilot tool and present some of the feedback as well as results from the live pilot. In Figure 2.5 we present an illustrative figure created from the Excel tool. Note that since these product features do not exist in the market yet it was impossible to compare our predictions with actual numbers. Hence, our industry partner tried other products from the test set and compared the final prediction numbers in order to see how well the tool and the models were performing.

Figure 2.6 presents results from one such product. Our prediction model performed well on the product with an error of only 1% of overall sales. We also received positive feedback on other features of the tool such as the list of important predictors and the movement of sales with changing promotion or distribution levels. Overall, our industry partners appreciated the ease of use of the tool as well as a simplified approach of incorporating new data generated through product releases in the model. In all, the feedback that we received on the tool was very positive and the industry partner is now trying to incorporate the tool in the decision making and sales estimation process of new products.

2.7 Case Study: Fashion Retailer

In order to test the robustness of our approach, and test whether the results of this work applies to other retail settings, we also collaborated with a fashion retailer. Our industry partner is one of the world's largest fashion retailer. As in the previous case, this fashion retailer also invests considerable resources in new product releases. In what follows, we describe the data and results for forecasting sales of new products for the fast fashion retailer.

2.7.1 Data Description

The fast fashion retail industry is characterized by fast paced innovation and product with very short life cycle. For instance, Gallien et al. (2015) states that in 2011, Zara released 8000 new fashion products in the span of just one year. In this case, the products sold by our partner also have a very short life cycle (at most 6 weeks).

Our industry partner classifies products into segments and subsegments (e.g., a particular subsegment could include all female trousers or all male dress shirts). As with the consumer goods data, we estimated and tested our model at the subsegment level. For each product we again divided the data into train-validation-test set. As noted in Section 5.2, we have access data for different subsegments between April to June of 2016. In consultation with our industry partners, we realized that a product in fast fashion retail is considered new for the first half week of its sales and hence we subset the data to only include the first half week of sales for each product.

We are provided with sales and feature data from different new products at the store level. For each product, we have access to product features such as color, price, segment, subsegment, and time-based features such as the time since introduction, seasonality.Moreover, we have distribution information such as total capacity allocated, store capacity allocated, and we have e-commerce data such as number of user clicks on the product, number of times the product was added in the cart, number of times the product was bought online, and other online specific features. We used all these features as potential predictors of sales of the new product.

2.7.2 Results

In Table 2.7, we compare the results of our linear CWR algorithm against the two benchmark methods on two different segments of fashion products. We present the forecasting metrics at

both the individual store level as well as an aggregated store level (where sales of products are aggregated over stores). Both these prediction numbers are important for the firm and can aid different decision making problems. For instance, the aggregate predictions can guide overall initial production of the product while the individual store level prediction can guide the distribution of the product.

Table 2.7: WMAPE comparison of algorithms on segments of fashion products

Segment	Store Level	CWR	LASSO	CTR
1	Individual	0.553	0.699	1.031
	Aggregated	0.291	0.463	0.824
2	Individual	0.660	0.777	1.598
	Aggregated	0.370	0.587	0.895

In all instances, our method considerably improves over the other benchmarks. Moreover, this improvement is independent of the aggregation level as well as the subsegment chosen. We note that in absolute terms, predictions at the aggregate level are better. This is expected as prediction at aggregated level is easier than individual store level due to higher variance from store to store.

2.8 Conclusion

In this chapter, we propose a new sales forecasting model that can accurately predict sales of new products by efficiently using available data on comparable products. The forecasting model proposed is general as it is able to estimate a variety of standard demand models for unknown clusters of products. Specifying the linear case for our real-world implementation, we develop an optimization algorithm to forecast new product sales. This algorithm estimates the optimal forecasting model with analytical guarantees on its forecasting error, but it is computationally hard to run. Hence, we also propose an approximate version of the algorithm that is scalabale due to its lower running time. We then use our algorithm to forecast sales of new products for a consumer goods manufacturer and a fashion retailer. We show robust results on real datasets from various segments and subsegments that significantly improve the prediction error over other benchmarks. Finally, we create and test an Excel pilot tool with our consumer goods manufacturing partner, and observe that its accurate, robust, and fast prediction process considerably simplifies the task of forecasting new product sales for practitioners.

Chapter 3

Personalized Product Recommendations with Customer Disengagement

3.1 Introduction

Personalized customer recommendations are a key ingredient to the success of platforms such as Netflix, Amazon and Expedia. Product variety has exploded, catering to the heterogeneous tastes of customers. However, this has also increased search costs, making it difficult for customers to find products that interest them. Platforms add value by learning a customer's preferences over time, and leveraging this information to match her with relevant products.

The personalized recommendation problem is typically formulated as an instance of collaborative filtering (Sarwar et al. 2001, Linden et al. 2003). In this setting, the platform observes different customers' past ratings or purchase decisions for random subsets of products. Collaborative filtering techniques use the feedback across all observed customer-product pairs to infer a low-dimensional model of customer preferences over products. This model is then used to make personalized recommendations over unseen products for any specific customer. While collaborative filtering has found industry-wide success (Breese et al. 1998, Herlocker et al. 2004), it is well-known that it suffers from the "cold start" problem (Schein et al. 2002). In particular, when a new customer enters the platform, no data is available on her preferences over *any* products. Collaborative filtering can only make sensible personalized recommendations for the new customer after she has rated at least $\mathcal{O}(d \log n)$ products, where d is the dimension

of the low-dimensional model learned via collaborative filtering and n is the total number of products. Consequently, bandit approaches have been proposed in tandem with collaborative filtering (Bresler et al. 2014, Li et al. 2016, Gopalan and Maillard 2016) to tackle the cold start problem using a combination of exploration and exploitation. The basic idea behind these algorithms is to sequentially offer random products to a customer during an exploration phase, learn the customer's low-dimensional preference model, and then exploit this model to make good recommendations.

A key assumption underlying this literature is that the customer is patient, and will remain on the platform for the entire (possibly unknown) time horizon T regardless of the goodness of the recommendations that have been made thus far. However, this is a tenuous assumption, particularly when customers have strong outside options (*e.g.,* a Netflix user may abandon the platform for Hulu if they receive a series of bad entertainment recommendations). We demonstrate this effect using customer panel data on a series of ad campaigns from a major commercial airline. Specifically, we find that a customer is far more likely to click on a suggested travel product in the current ad campaign if the previous ad campaign's recommendation was relevant to her. In other words, customers may *disengage* from the platform and ignore new recommendations entirely if past recommendations were irrelevant. In light of this issue, we introduce a new formulation of the bandit product recommendation problem where customers may disengage from the platform depending on the rewards of past recommendations, *i.e.,* the customer's time horizon T on the platform is no longer fixed, but is a function of the platform's actions thus far.

Customer disengagement introduces a significant difficulty to the dynamic learning or bandit literature. We prove lower bounds that show that any algorithm in this setting achieves regret that scales linearly in T (the customer's time horizon on the platform if they are given good recommendations). This hardness result arises because no algorithm can satisfy *every* customer early on when we have limited knowledge of their preferences; thus, no matter what policy we use, at least some customers will disengage from the platform. The best we can hope to accomplish is to keep a large fraction of customers engaged on the platform for the entire time horizon, and to match these customers with their preferred products.

However, classical bandit algorithms perform particularly badly in this setting – we prove that *every* customer disengages from the platform with probability one as T grows large. This is because bandit algorithms *over-explore*: they rely on an early exploration phase where cus-

tomers are offered random products that are likely to be irrelevant for them. Thus, it is highly probable that the customer receives several bad recommendations during exploration, and disengages from the platform entirely. This exploration is continued for the entire time horizon, T, under the principal of optimism. This is not to say that learning through exploration is a bad strategy. We show that a greedy exploitation-only algorithm also under-performs by either over-exploring through natural exploration, or under-exploring by getting stuck in sub-optimal fixed points. Consequently, the platform misses out on its key value proposition of learning customer preferences and matching them to their preferred products.

Our results demonstrate that one needs to more carefully balance the exploration-exploitation tradeoff in the presence of customer disengagement. We propose a simple modification of classical bandit algorithms by constraining the space of possible product recommendations upfront. We leverage the rich information available from existing customers on the platform to identify a diverse subset of products that are palatable to a large segment of potential customer types; all recommendations made by the platform for new customers are then constrained to be in this set. This approach guarantees that mainstream customers remain on the platform with high probability, and that they are matched to their preferred products over time; we compromise on tail customers, but these customers are unlikely to show up on the platform, and catering recommendations to them endangers the engagement of mainstream customers. We formulate the initial optimization of the product offering as an integer program. We then prove that our proposed algorithm achieves sublinear regret in T for a large fraction of customers, *i.e.,* it succeeds in keeping a large fraction of customers on the platform for the entire time horizon, and matches them with their preferred product. Numerical experiments on synthetic and real data demonstrate that our approach significantly improves both regret and the length of time that a customer is engaged with the platform compared to both classical bandit and greedy algorithms.

3.1.1 Main Contributions

We highlight our main contributions below:

1. *Evidence of disengagement:* We first present empirical evidence of customer disengagement using panel data from a sequence of ad campaigns from a major airline carrier. Our results strongly suggest that customers decide to stay on the platform based on the quality of recommendations.

2. *Disengagement model:* A linear bandit is the classical formulation for learning product recommendations for new customers. Motivated by our empirical results on customer disengagement, we propose a novel formulation, where the customer's horizon length is endogenously determined by past recommendations, *i.e.,* the customer may exit if given poor recommendations.

3. *Hardness & classical approaches:* We first show that no algorithm can perform well (*i.e.,* achieve sublinear regret) on *every* customer in this setting; however, we can hope to perform well on a subset of customers. Unfortunately, we show that existing state-of-art classical bandit and greedy algorithms over-explore and fail to keep *any* customer engaged on the platform, suggesting that platforms should be careful to avoid over-exploration when learning personalized recommendations.

4. *Algorithm:* We propose the Constrained Bandit algorithm, which modifies standard bandit strategies by constraining the product set upfront using a novel integer programming formulation. The integer program leverages information on other customers on the platform to select a subset of products that are likely to be relevant for the incoming customer. Unlike classical approaches, the Constrained Bandit achieves sublinear regret for a significant fraction of customers.

5. *Numerical experiments:* Extensive numerical experiments on synthetic and real world movie recommendation data (we use the publicly available MovieLens data by Harper and Konstan 2016) demonstrate that the Constrained Bandit significantly improves both regret and the length of time that a customer is engaged with the platform. In particular, our approach increases mean customer engagement time on MovieLens by up to 80% over classical bandit and greedy algorithms.

3.1.2 Related Literature

Personalized decision-making is increasingly a topic of interest, and a central problem is that of learning customer preferences and optimizing the resulting recommendations. However, customer disengagement can introduces significant difficulty to traditional learning algorithms that have been proposed in the literature.

Personalized Recommendations: The value of personalizing the customer experience has been recognized for a long time (Surprenant and Solomon 1987). We refer the readers to Murthi and Sarkar (2003) for an overview of personalization in operations and revenue management applications. Recently, Besbes et al. (2015), Demirezen and Kumar (2016), and Farias and Li (2017) have proposed novel methods for personalization in online content and product recommendations. We take the widely-used collaborative filtering framework (Sarwar et al. 2001, Su and Khoshgoftaar 2009) as our point of departure. However, all these methods suffer from the cold start problem (Schein et al. 2002). When a new customer enters the platform, no data is available on her preferences over any products, making the problem of personalized recommendations challenging.

Bandits: Consequently, bandit approaches have been proposed in tandem with collaborative filtering (Bresler et al. 2014, Li et al. 2016, Gopalan and Maillard 2016) to tackle the cold start problem using a combination of exploration and exploitation. The basic idea behind these algorithms is to offer random products to customers during an exploration phase, learn the customer's preferences over products, and then exploit this model to make good recommendations. Relatedly, Lika et al. (2014) and Wei et al. (2017) use machine learning techniques such as similarity measures and deep neural networks to alleviate the cold start problem. In this chapter, we consider the additional challenge of customer disengagement, which introduces a significant difficulty to the dynamic learning or bandit literature. In fact, we show that traditional bandit approaches over-explore, and fail to keep any customer engaged on the platform in the presence of disengagement.

At a high level, our work also relates to the broader bandit literature, where a decision-maker must dynamically collect data to learn and optimize an unknown objective function. For example, many have studied the problem of dynamically pricing products with unknown demand (see, *e.g.,* den Boer and Zwart 2013, Keskin and Zeevi 2014). Agrawal et al. (2016) analyze the problem of optimal assortment selection with unknown user preferences. Johari et al. (2017) learn to match heterogeneous workers (supply) and jobs (demand) on a platform. Kallus and Udell (2016) use online learning for personalized assortment optimization. These studies rely on optimally balancing the exploration-exploitation tradeoff under bandit feedback. Relatedly, Shah et al. (2018) study bandit learning where the platform's decisions affects the arrival process of new customers; interestingly, they find that classical bandit algorithms can

perform poorly due to under-exploration. Closer to our findings, Russo and Van Roy (2018) argue that bandit algorithms can over-explore when an approximately good solution suffices, and propose constraining exploration to actions with sufficiently uncertain rewards. A key assumption underlying this literature is that the time horizon T is fixed and independent of the goodness of the decisions made by the decision-maker. We show that this is a tenuous assumption for recommender systems, since customers may disengage from the platform when offered poor recommendations. Thus, the customer's time horizon T is endogenously determined by the platform's actions, necessitating a novel analysis.

Customer Disengagement: Customer disengagement and its relation to service quality have been extensively studied. For instance, Venetis and Ghauri (2004) use a structural model to establish that service quality contributes to long term customer relationship and retention. Bowden (2009) models the differences in engagement behavior across new and repeat customers. Sousa and Voss (2012) study the impact of e-service quality on customer behavior in multi-channel services.

Closer to our work, Fitzsimons and Lehmann (2004) use a large-scale experiment on college students to demonstrate that poor recommendations can have a considerably negative impact on customer engagement. We find similarly that poor recommendations result in customer disengagement on airline campaign data. Relatedly, Tan et al. (2017) empirically find that increasing product variety on Netflix *increases* demand concentration around popular products; this is surprising since one may expect that increasing product variety would cater to the long tail of customers, enabling more nuanced customer-product matches. However, increasing product variety also increases customer search costs, which may cause customers to cluster around popular products or disengage from the platform entirely. Our proposed algorithm, the Constrained Bandit, makes a similar tradeoff — we constrain our recommendations upfront to a set of popular products that cater to mainstream customers. This approach guarantees that mainstream customers remain engaged with high probability; we compromise on tail customers, but these customers are unlikely to show up, and catering recommendations to them endangers the engagement of mainstream customers.

There are also several papers that study service optimization to improve customer engagement. For example, Davis and Vollmann (1990) develop a framework for relating customer wait times with service quality perception, while Lu et al. (2013) provide empirical evidence

of changes in customer purchase behavior due to wait times. Kanoria et al. (2018) model customer disengagement based on the goodwill model of Nerlove and Arrow (1962). In their work, a service provider has two options: a low-cost service level with high likelihood of customer abandonment, or a high-cost service level with low likelihood of customer abandonment. Similarly, Aflaki and Popescu (2013), model the customer disengagement decision as a deterministic known function of service quality. None of these papers study learning in the presence of customer disengagement.

A notable exception is Johari and Schmit (2018), who study the problem of learning a customer's tolerance level in order to send an appropriate number of marketing messages without creating customer disengagement. Here, the decision-maker's objective is to learn the customer's tolerance level, which is a scalar quantity. Similar to our work, the customer's disengagement decision is endogenous to the platform's actions (*e.g.,* the number of marketing messages). However, in our work, we seek to learn a low-dimensional model of the customer's preferences, *i.e.,* a complex mapping of unknown customer-specific latent features to rewards based on product features. The added richness in our action space (product recommendations rather than a scalar quantity) necessitates a different algorithm and analysis. Our work bridges the gap between state-of-the-art machine learning techniques (collaborative filtering and bandits) and the extensive modeling literature on customer disengagement and service quality optimization.

3.2 Motivation

We use customer panel data from a major commercial airline, obtained as part of client engagement at IBM, to provide evidence for customer disengagement. The airline conducted a sequence of ad campaigns over email to customers that were registered with the airline's loyalty program. Our results suggest that a customer indeed disengages with recommendations if a past recommendation was irrelevant to her. This finding motivates our problem formulation described in the next section.

Data. The airline conducted 7 large-scale non-targeted ad campaigns over the course of a year. Each campaign involved emailing loyalty customers destination recommendations hand-selected by a marketing team at discounted rates. Importantly, these recommendations were made uniformly across customers regardless of customer-specific preferences.

Our sample consists of 130,510 customers. For each campaign, we observe whether or not

the customer clicked on the link provided in the email after viewing the recommendations. We assume that a click signals a positive reaction to the recommendation, while no click could signal either (i) a negative reaction to the recommendation, or (ii) that the customer is already disengaged with the airline campaign and is no longer responding to recommendations.

Empirical Strategy. Since recommendations were not personalized, we use the heterogeneity in customer preferences to understand customer engagement in the current campaign as a function of the customer-specific quality of recommendations in previous campaigns. To this end, we use the first 5 campaigns in our data to build a score that assesses the relevance of a recommendation to a particular customer. We then evaluate whether the quality of the recommendation in the 6^{th} (previous) campaign affected the customer's response in the 7^{th} (current) campaign after controlling for the quality of the recommendation in the 7^{th} (current) campaign. Our reasoning is as follows: in the absence of customer disengagement, the customer's response to a campaign should depend only on the quality of the current campaign's recommendations; if we instead find that the quality of the previous campaign's recommendations plays an additional negative role in the likelihood of a customer click in the current campaign, then this strongly suggests that customers who previously received bad recommendations have disengaged from the airline campaigns.

We construct a personalized relevance score of recommendations for each customer using click data from the first 5 campaigns. This score is trained using the standard collaborative filtering package available in Python, and achieves an in-sample RMSE of 10%. We note that a version of this score was later implemented in a live pilot by the airline for making personalized recommendations to customers in similar ad campaigns.

Regression Specification. We perform our regression over the 7^{th} (current) campaign's click data. Specifically, we wish to understand if the quality of the recommendation in the 6^{th} (previous) campaign affected the customer's response in the current campaign after controlling for the quality of the current campaign's recommendation. For each customer i, we use the collaborative filtering model to evaluate the relevance score $prev_i$ of the previous campaign's recommendations and the relevance score $curr_i$ of the current campaign's recommendation. We then perform a simple logistic regression as follows:

$$y_i = f(\beta_0 + \beta_1 \cdot prev_i + \beta_2 \cdot curr_i + \varepsilon_i)\,,$$

where f is the logistic function and y_i is the click outcome for customer i in the current campaign, and ε_i is i.i.d. noise. We fit an intercept term β_0, the effect of the previous campaign's recommendation quality on the customer's click likelihood β_1, and the effect of the current campaign's recommendation quality on the customer's click likelihood β_2. We expect β_2 to be positive since better recommendations in the current campaign should yield higher click likelihood in the current campaign. Our null hypothesis is that $\beta_1 = 0$, and a finding that $\beta_1 < 0$ would suggest that customers disengage from the campaigns if previous recommendations were of poor quality.

Results. Our regression results are shown in Table 3.1. As expected, we find that customers are more likely to click if the current campaign's recommendation is relevant to the customer, *i.e.,* $\beta_2 > 0$ (p-value $= 0.02$). More importantly, we find evidence for customer disengagement since customers are less likely to click in the current campaign if the *previous* campaign's recommendation was not relevant to the customer, *i.e.,* $\beta_1 > 0$ (p-value $= 7 \times 10^{-9}$). In fact, our point estimates suggest that the disengagement effect dominates the value of the current campaign's recommendation since the coefficient β_1 is roughly three times the coefficient β_2. In other words, it is much more important to have offered a relevant recommendation in the previous campaign (*i.e.,* to keep customers engaged with the campaigns) compared to offering a relevant recommendation in the current campaign to get high click likelihood. These results motivate the problem formulation in the next section explicitly modeling customer disengagement.

Variable	**Point Estimate**	**Standard Error**
(Intercept)	-3.62^{***}	0.02
Relevance Score of Previous Ad Campaign	0.06^{***}	0.01
Relevance Score of Current Ad Campaign	0.02^{**}	0.01

$^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$

Table 3.1: Regression results from airline ad campaign panel data.

3.3 Problem Formulation

We embed our problem within the popular product recommendation framework of collaborative filtering (Sarwar et al. 2001, Linden et al. 2003). In this setting, the key quantity of interest is a matrix $A \in \mathbb{R}^{m \times n}$, whose entries A_{ij} are numerical values rating the relevance of product j to customer i. Most of the entries in this matrix are missing since a typical customer has only evaluated a small subset of available products. The key idea behind collaborative filtering is to

use a low-rank decomposition

$$A = U^\top V\,,$$

where $U \in \mathbb{R}^{m\times d}, V \in \mathbb{R}^{d\times n}$ for some small value of d. The decomposition can be interpreted as follows: each customer $i \in \{1, ..., m\}$ is associated with some low-dimensional vector $U_i \in \mathbb{R}^d$ (row i of the matrix U) that models her preferences; similarly, each product $j \in \{1, ..., n\}$ is associated with a low-dimensional vector $V_j \in \mathbb{R}^d$ (given by column j of the matrix V) that models its attributes. Then, the relevance or utility of product j to customer i is simply $U_i^\top V_j$. We refer the reader to Su and Khoshgoftaar (2009) for an extensive review of the collaborative filtering literature. We assume that the platform has a large base of existing customers from whom we have already learned good estimates of the matrices U and V. In particular, all existing customers are associated with known vectors $\{U_i\}_{i=1}^m$, and similarly all products are associated with known vectors $\{V_j\}_{j=1}^n$.

Now, consider a single new customer that arrives to the platform. She forms a new row in A, and all the entries in her row are missing since she is yet to view any products. Like the other customers, she is associated with some vector $U_0 \in \mathbb{R}^d$ that models her preferences, *i.e.,* her expected utility for product $j \in \{1, ..., n\}$ is $U_0^\top V_j$. However, U_0 is unknown because we have no data on her product preferences yet. We assume that $U_0 \sim \mathcal{P}$, where $\mathcal{P}$ is a known distribution over new customers' preference vectors; typically, $\mathcal{P}$ is taken to be the empirical distribution of known preference vectors associated with the existing customer base $\{U_1, ..., U_m\}$. For ease of exposition and analytical tractability, we will take $\mathcal{P}$ to be a multivariate normal distribution $\mathcal{N}(0, \sigma^2 I_d)$ throughout the rest of the paper.

At each time t, the platform makes a single product recommendation $a_t \in \{V_1, ..., V_n\}$, and observes a noisy signal of the customer's utility

$$U_0^\top a_t + \varepsilon_t\,,$$

where ε_t is zero-mean ξ-subgaussian noise. For instance, platforms often make recommendations through email marketing campaigns (see Figure 3.1 for example emails from Netflix and Amazon), and observe noisy feedback from the customer based on their subsequent click/view/purchase behavior. We seek to learn U_0 through the customer's feedback from a series of product recommendations in order to eventually offer her the best available product

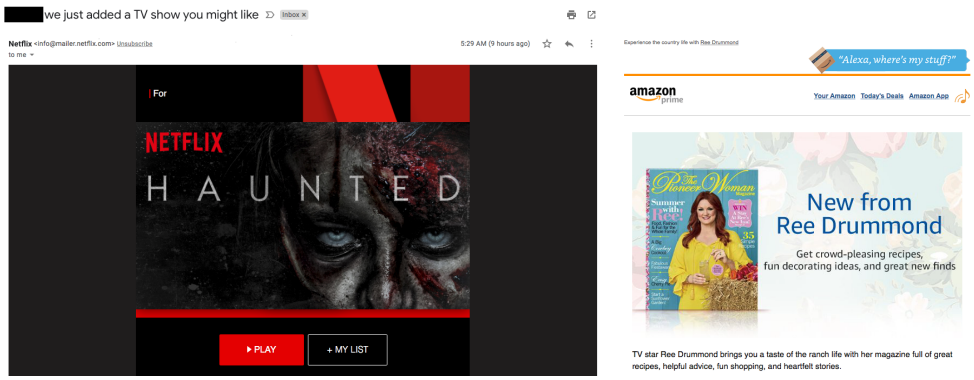

Figure 3.1: Examples of personalized recommendations through email marketing campaigns from Netflix (left) and Amazon Prime (right).

on the platform

$$V_* = \arg \max_{V_j \in \{V_1, \ldots, V_n\}} U_0^\top V_j \, .$$

We impose that $U_0^\top V_* > 0$, *i.e.,* the customer receives positive utility from being matched to her most preferred product on the platform; if this is not the case, then the platform is not appropriate for the customer. We further assume that the product attributes V_i are bounded, *i.e.,* there exists $L > 0$ such that $\|V_i\|_2 \leq L \;\; \forall i$. The problem of learning U_0 now reduces to a classical linear bandit (Rusmevichientong and Tsitsiklis 2010), where we seek to learn an unknown parameter U_0 given a discrete action space $\{V_j\}_{j=1}^n$ and stochastic linear rewards. However, as we describe next, our formulation as well as our definition of regret departs from the standard setting by modeling customer disengagement.

3.3.1 Disengagement Model

Let T be the time horizon for which the customer will stay on the platform if she remains engaged throughout her interaction with the platform. Unfortunately, poor recommendations can cause the customer to disengage from the platform. In particular, at each time t, upon viewing the platform's product recommendation a_t, the customer makes a choice $\Upsilon_t \in \{0, 1\}$, where $\Upsilon_t = 1$ signifies that the customer has disengaged (and receives zero utility for the remainder of the time horizon T)[1] and $\Upsilon_t = 0$ signifies that the customer has chosen to remain engaged for the next time step.

There are many ways to model disengagement. Our model follows the experimental findings of Fitzsimons and Lehmann (2004), who study customer reactions to poor or inconsistent recommendations. In particular, through a series of behavioral experiments, they observed that

[1]We later relax this assumption to allow disengaged customers to return to the platform after some time.

irrelevant recommendations could lead to customers completely ignoring future recommendations. Therefore, we consider the following model: each customer has a tolerance parameter $\rho > 0$ and a disengagement propensity $p \in [0, 1]$. Then, the probability that the customer disengages at time t (assuming she has been engaged until now) upon receiving recommendation a_t is:

$$\Pr[\Upsilon_t = 1 \mid a_t] = \begin{cases} 0 & \text{if } \; u_0^\top a_t \; \geq \; u_0^\top V_* - \rho \,, \\ p & \text{otherwise.} \end{cases}$$

In other words, each customer is willing to tolerate a utility reduction of up to ρ from a recommendation with respect to her utility from her (unknown) optimal product V_*. If the platform makes a recommendation that results in a utility reduction greater than ρ, the customer will disengage with probability p. Note that we recover the classical linear bandit formulation (with no disengagement) when $p = 0$ or $\rho \to \infty$. We discuss alternative disengagement models in the next subsection.

We seek to construct a sequential decision-making policy $\pi = \{a_1, \cdots, a_T\}$ that learns U_0 over time to maximize the customer's utility on the platform. We measure the performance of π by its *cumulative expected regret*, where we modify the standard metric in the analysis of bandit algorithms (Lai and Robbins 1985) to accommodate customer disengagement. In particular, we compare the performance of our policy π against an oracle policy π^* that knows U_0 in advance and always offers the customer her preferred product V_*. At time t, we define the instantaneous expected regret of the policy π for a new customer with realized latent attributes $U_0 = u_0$:

$$r_t^\pi(\rho, p, u_0) = \begin{cases} u_0^\top V_* & \text{if } \; \Upsilon_{t'} = 1 \;\; \text{for any } \;\; t' < t \,, \\ u_0^\top V_* - u_0^\top a_t & \text{otherwise.} \end{cases}$$

This is simply the expected utility difference between the oracle's recommendation and our policy's recommendation, accounting for the fact that the customer receives zero utility for all future recommendations after she disengages. The expectation is taken with respect to ε_t, the ξ-subgaussian noise in realized customer utilities that was defined earlier. The cumulative expected regret for a given customer is then simply

$$\mathcal{R}^\pi(T, \rho, p, u_0) = \sum_{t=1}^{T} r_t^\pi(\rho, p, u_0) \,. \tag{3.1}$$

Our goal is to find a policy π that minimizes the cumulative expected regret for a new customer whose latent attributes U_0 is a random variable drawn from the distribution $\mathcal{P} = \mathcal{N}(0, \sigma^2 I_d)$. We will show in the next section that no policy can hope to achieve sublinear regret for *all* realizations of U_0; however, we can hope to perform well on likely realizations of U_0, *i.e.,* mainstream customers.

We note that our algorithms and analysis assume that ρ (the tolerance parameter) and p (the disengagement propensity) are known. In practice, these may be unknown parameters that need to be estimated from historical data, or tuned during the learning process. We discuss one possible estimation procedure of these parameters from historical movie recommendation data in our numerical experiments (see §3.6).

3.3.2 Alternative Disengagement Models

We present the simplest possible disengagement model above; this allows for a simpler, more intuitive exposition in the next two sections. However, our results easily extend to alternative, more complex models of disengagement, *e.g.,*

1. In some settings, the customer may not have any beliefs about the utility $u_0^\top V_*$ that she will derive from her (apriori unknown) optimal match, making it difficult to model her disengagement decision around this value. In this case, the customer may instead choose to disengage (with some probability) if she does not receive at least a baseline utility of $\tilde{\rho}$.

2. The customer's disengagement probability p may not be a constant. It could depend on the current time step t (*e.g.,* capturing the customer's loyalty to the platform), or on the utilities derived from the recommendations thus far $\{u_0^\top a_i\}_{i=1}^t$ (*e.g.,* a poor recommendation at time t may be less likely to cause disengagement if past recommendations have been relevant).

We can easily incorporate the above by updating the customer's disengagement decision to be:

$$\Pr[\Upsilon_t = 1 \mid a_t] = \begin{cases} 0 & \text{if } u_0^\top a_t \geq \tilde{\rho}\,, \\ p(t, u_0, a_1, \ldots a_t) & \text{otherwise.} \end{cases}$$

Here, $\tilde{\rho} < u_0^\top V_*$, *i.e.,* there is at least one product on the platform that is acceptable to the customer. We further impose that the disengagement probability is uniformly bounded below by a positive constant, *i.e.,* $p(t, u_0, a_1, \ldots a_t) \geq \tilde{c} > 0$ for all $t, u_0, \{a_i\}_{i=1}^t$; this ensures that

disengagement is always a salient feature on the platform. All the forthcoming results (lower and upper bounds) can be easily extended to the more general setting above; we defer the details to Appendix B.4.

Finally, in some settings, customers may only disengage *temporarily*, rather than for the entire horizon T; we discuss how our results extend to this setting in Appendix B.4.

3.4 Classical Approaches

We now prove lower bounds that demonstrate (i) no policy can perform well on every customer in this setting, and (ii) bandit algorithms and greedy Bayesian updating can fail for all customers.

3.4.1 Preliminaries

We restrict ourselves to the family of non-anticipating policies $\Pi : \pi = \{\pi_t\}$ that form a sequence of random functions π_t that depend only on observations collected until time t. In particular, if we let $H_t = (a_1, Y_1, a_2, Y_2, ...a_{t-1}, Y_{t-1})$ denote the vectorized history of product recommendations and corresponding utility realizations and $\mathcal{F}_t$ denote the σ-field generated by H_t, then π_{t+1} is $\mathcal{F}_t$ measurable. All policies assume full knowledge of the tolerance parameter ρ, the disengagement propensity p, and the distribution of latent customer attributes $\mathcal{P}$.

Next, we define a general class of bandit learning algorithms that achieve sublinear regret in the standard setting with no disengagement.

Definition 3.4.1. *A policy π belongs in the class of consistent bandit algorithms Π^C if for all u_0, there exists $\nu \in [0, 1)$ and $\mathcal{R}(T, \rho, p = 0, u_0) = \mathcal{O}(T^\nu)$. This is equivalent to the following condition*:

$$\lim_{T\to\infty} \sup \frac{\log\left(\mathcal{R}(T, \rho, p = 0, u_0)\right)}{\log(T)} = \nu\,,$$

where the supremum is taken over all feasible realizations of the unknown customer feature vector u_0. As discussed before, when $p = 0$, our regret definition reduces to the classical bandit regret with no disengagement. The above definition implies that a policy π is consistent if its rate of cumulative regret is sublinear in T. The consistent policy class Π^C includes the well-studied UCB (*e.g.,* Auer 2002, Abbasi-Yadkori et al. 2011), Thompson Sampling (*e.g.,* Agrawal and Goyal 2013, Russo and Van Roy 2014), and other bandit algorithms. Our definition of consistency is inspired by Lattimore and Szepesvari (2016), but encompasses a larger class

of policies. We will show that any algorithm in Π^C fails to perform well in the presence of disengagement.

Notation: For any vector $V \in \mathbb{R}^d$ and positive semidefinite matrix $X \in \mathbb{R}^{d\times d}$, $\|V\|_X$ refers to the operator norm of V with respect to matrix X given by $\sqrt{V^\top XV}$. Similarly, for any set S, $S\backslash i$ for some $i \in S$ refers to the set S without element i. I_d refers to the $d \times d$ identity matrix for some $d \in \mathbb{Z}$. For any series of scalars (vectors), $Y_1, ...Y_t$, $Y_{1:t}$ refers to the column vector of the scalars (vectors) Y_1,..,Y_t. Next, we define the set $\mathcal{S}(u_0, \rho)$ of products that are tolerable to the customer, *i.e.,* recommending any product from this (unknown) set will not cause disengagement:

Definition 3.4.2. *Let $\mathcal{S}(u_0, \rho)$ be the set of products, among all products, that satisfy the tolerance threshold for the customer with latent attribute vector, u_0. More specifically, when $p > 0$,*

$$\mathcal{S}(u_0, \rho) := \{i : u_0^\top V_i \ \geq \ u_0^\top V_* - \rho, \forall i = 1, .., n\}\,. \tag{3.2}$$

Note that in the classical bandit setting, this set contains all products, $|\mathcal{S}(u_0, \rho)| = n$. When $\mathcal{S}(u_0, \rho)$ is large, exploration is less costly, but as the customer tolerance threshold ρ decreases, $|\mathcal{S}(u_0, \rho)|$ decreases as well.

Finally, we consider the following simplified latent product features to enable a tractable analysis.

Setting 1. We assume that there are d total products in $\mathbb{R}^d$, and the latent product features $V_i = e_i$, the i^{th} basis vector. We also take $p > 0$, *i.e.,* customers may disengage.

3.4.2 Lower bounds

We first show an impossibility result that no non-anticipating policy can obtain sublinear regret over *all* customers. We consider the worst-case regret of any non-anticipating policy over all feasible customer tolerance parameters ρ. Proofs for all results in this section are deferred to Appendix B.1.

Theorem 3.4.3 (Hardness Result)**.** Under the assumptions of Setting 1, any non anticipating

policy $\pi \in \Pi$ achieves regret that scales linearly with T:

$$\inf_{\pi \in \Pi} \sup_{\rho > 0} \mathbb{E}_{u_0 \sim \mathcal{P}} \left[\mathcal{R}^{\pi}(T, \rho, p, u_0) \right] = C \cdot T = \mathcal{O}(T) \,,$$

where $C \in \mathbb{R}$ is a constant independent of T but dependent on other problem parameters.

Theorem 3.4.3 shows that the expected worst case regret is linear in T. In other words, regardless of the policy chosen, there exists a subset of customers (with positive measure under $\mathcal{P}$) who incur linear regret in the presence of disengagement. The proof relies on showing that there is always a positive probability that the customer (i) will not be offered her preferred product in the first time step, and consequently, (ii) for sufficiently small ρ, will disengage from the platform immediately. Thus, in expectation, any non-anticipating policy is bound to incur linear regret.

Theorem 3.4.3 shows that product recommendation with customer disengagement requires making a trade-off over the types of customers that we seek to engage. No policy can keep *all* the users engaged without knowing the user's preference apriori. Nevertheless, since Theorem 3.4.3 only characterizes the worst case *expected* regret, this poor performance can be caused by a very small fraction of customers. Hence, another approach could be to ensure that at least a large fraction of customers (mainstream customers) are engaged, while potentially sacrificing the engagement of customers with niche preferences (tail customers).

In Theorem 3.4.4, we show that consistent bandit learning algorithms fail to achieve engagement even for mainstream customers throughout the time horizon. Thus, in contrast to showing that the worst case *expected* regret is linear (Theorem 3.4.3), we show that the worst case regret is linear for *any* customer realization u_0.

Theorem 3.4.4 (Failure of Bandits)**.** Let u_0 be any realization of the latent user attributes from $\mathcal{P}$. Under the assumptions of Setting 1, any consistent bandit algorithm $\pi \in \Pi^C$ achieves regret that scales linearly with T for this customer as $T \to \infty$. That is,

$$\inf_{\pi \in \Pi^C} \sup_{\rho > 0} \mathcal{R}^{\pi}(T, \rho, p, u_0) = C_1 \cdot T = \mathcal{O}(T) \,,$$

where $C_1 \in \mathbb{R}$ is a constant independent of T but dependent on other problem parameters.

Theorem 3.4.4 shows that the worst case regret of consistent bandit policies is linear for *every* customer realization (including mainstream customers). We note that this result is worse than

what we may have hoped for given the earlier hardness result (Theorem 3.4.3), since the linearity of regret applies to all customers rather than a subset of customers. The proof of Theorem 3.4.4 considers the case when the size of the set of tolerable products $|\mathcal{S}(u_0, \rho)| < d$, which occurs for sufficiently small ρ. Clearly, exploring outside this set can lead to customer disengagement. However, since $|\mathcal{S}(u_0, \rho)| < d$, this set of products cannot span the space $\mathbb{R}^d$, implying that one cannot recover the true customer latent attributes u_0 without sampling products outside of the set. On the other hand, consistent bandit algorithms require convergence to u_0, *i.e.,* they will sample outside the set $\mathcal{S}(u_0, \rho)$ infinitely many times (as $T \to \infty$) at a rate that depends on their corresponding regret bound. Yet, it is clear to see that offering infinitely many recommendations outside the customer's set of tolerable products $\mathcal{S}(u_0, \rho)$ will eventually lead to customer disengagement (when $p > 0$) with probability 1. This result highlights the tension between avoiding incomplete learning (which requires exploring products outside the tolerable set) and avoiding customer disengagement (which requires restricting our recommendations to the tolerable set). Thus, we see that the design of bandit learning strategies fundamentally relies on the assumption that the time horizon T is exogenous, making exploration inexpensive. State-of-the-art techniques such as UCB and Thompson Sampling perform particularly poorly by over-exploring in the presence of customer disengagement.

Recent literature has highlighted the success of greedy policies in bandit problems where exploration may be costly (see, *e.g.,* Bastani et al. 2017). One may expect that the natural exploration afforded by greedy policies may enable better performance in settings where exploration can lead to customer disengagement. Therefore, we now shift our focus to Greedy Bayesian Updating policy (Algorithm 1) below. We use a Bayesian policy since we wish to make full use of the known prior $\mathcal{P}$ over latent customer attributes. Unfortunately, we find that, similar to consistent bandit algorithms, the greedy policy also incurs worst-case linear regret for *every* customer. Furthermore, the greedy policy can perform poorly even when there is no disengagement.

The greedy Bayesian updating policy begins by recommends the most commonly preferred product based on the $\mathcal{P}$. Then, in every subsequent time step, it observes the customer response, updates its posterior on the customer's latent attributes using Bayesian linear regression, and then offers the most commonly preferred product based on the updated posterior. The form of the resulting estimator $\hat{u}_t$ of the customer's latent attributes is similar to the well-known ridge regression estimator with regularization parameter $\frac{\xi^2}{\sigma^2 \bar{l}}$, where we regularize towards the mean

of the prior $\mathcal{P}$ over latent customer attributes (which we have normalized to 0 here).

Algorithm 1 Greedy Bayesian Updating (GBU)

Initialize and recommend a randomly selected product.
for $t \in [T]$ **do**
  Observe customer utility, $Y_t = u_0^\top a_t + \varepsilon_t$.
  Update customer feature estimate, $\hat{u}_{t+1} = \left(a_{1:t}^\top a_{1:t} + \frac{\xi^2}{\sigma^2} I\right)^{-1} \left(a_{1:t}^\top Y_{1:t}\right)$.
  Recommend product $a_{t+1} = \arg\max_{i=1\ldots,n} \hat{u}_{t+1}^\top V_i$.

In Theorem 3.4.5, we show that the greedy policy also fails to achieve engagement even for mainstream customers throughout the time horizon. In essence, the *free exploration* induced by greedy policies (see, *e.g.,* Bastani et al. 2017, Qiang and Bayati 2016) is in theory as problematic as the optimistic exploration by bandit algorithms. Furthermore, Theorem 3.4.6 shows that even when exploration is not costly (there is no disengagement), the greedy policy can get stuck at suboptimal fixed points, and fail to produce a good match.

Theorem 3.4.5 (Failure of Greedy)**.** Let u_0 be any realization of the latent user attributes from $\mathcal{P}$. Under the assumptions of Setting 1, the GBU policy achieves regret that scales linearly with T for this customer as $T \to \infty$. That is,

$$\sup_{\rho>0} \mathcal{R}^{GBU}(T, \rho, p, u_0) = C_2 \cdot T = \mathcal{O}(T)\,,$$

where $C_2 \in \mathbb{R}$ is a constant independent of T but dependent on other problem parameters.

Similar to our result for consistent bandit algorithms in Theorem 3.4.4, Theorem 3.4.5 shows that the worst case regret of the greedy policy is linear for *every* customer realization (including mainstream customers). While intuition may suggest that greedy algorithms avoid over-exploration, they still involve natural exploration due to the noise in customer feedback, which may cause the algorithm to over-explore and choose irrelevant products. Although Theorems 3.4.4 and 3.4.5 are similar, it is worth noting that over-exploration is *empirically* much less likely with the greedy policy than with a consistent bandit algorithm that is designed to explore. This difference is exemplified in our numerical experiments in §3.6; however, we will see that one is still better off (both theoretically and empirically) constraining exploration by restricting the product set upfront.

The proof of Theorem 3.4.5 has two cases: tail and mainstream customers. For tail customers (this set is determined by the choice of ρ), the first offered product (the most commonly preferred product across customers given the distribution $\mathcal{P}$) may not be tolerable, and so they disengage

immediately with some probability p, yielding linear expected regret for these customers. Note that this is true for any algorithm, including the Constrained Bandit. The more interesting case is that of mainstream customers, who *do* find the first offered product tolerable. In this case, since customer feedback is noisy, the greedy policy may subsequently erroneously switch to a product outside of the tolerable set, which again results in immediate customer disengagement with probability p. Note that this effect is exactly the natural exploration that allows the greedy policy to sometimes yield rate-optimal convergence in classical contextual bandits (Bastani et al. 2017). Putting these two cases together, we find that the greedy policy achieves linear regret for every customer.

It is also worth considering the performance of the greedy policy when there is no disengagement and exploration is not costly. In Theorem 3.4.6, we show that the greedy policy may under-explore and fail to converge in the other extreme, *i.e.,* when there is no customer disengagement. Unlike the previous results, this result is under the case of $p = 0$ (otherwise, the setting of Setting 1 applies).

Theorem 3.4.6 (Failure of Greedy without Disengagement)**.** Let $\rho \to \infty$ or $p = 0$, *i.e.,* no customer disengagement. The GBU policy achieves regret that scales linearly with T. That is,

$$\mathbb{E}_{u_0 \sim \mathcal{P}} \left[\mathcal{R}^{GBU}(T, \rho, p = 0, u_0) \right] = C_3 \cdot T = \mathcal{O}(T) \,,$$

where $C_3 \in \mathbb{R}$ is a constant independent of T but dependent on other problem parameters.

Theorem 3.4.6 shows that the greedy policy fails with some probability even in the classical bandit learning setting when there is no customer disengagement. The proof follows from considering the subset of customers for whom the most commonly preferred product is *not* their preferred product. We show that within this subset, the greedy policy continues recommending this suboptimal product for the remaining time horizon T with positive probability. This illustrates that a greedy policy can get "stuck" on a suboptimal product due to incomplete learning (see, *e.g.,* Keskin and Zeevi 2014) even when customers never disengage. Thus, we see that the greedy policy can also fail due to *under-exploration*. In contrast, a consistent bandit policy is always guaranteed to converge to the preferred product when there is no disengagement; the Constrained Bandit will trivially achieve the same guarantee since we will not restrict the product set when there is no disengagement.

These results illustrate that there is a need to constrain exploration in the presence of

customer disengagement; however, naively adopting a greedy policy does not achieve this goal. This is because, intuitively, the greedy policy constrains the *rate* of exploration rather than the *size* of exploration. The proof of Theorem 3.4.4 clearly demonstrates that the key issue is to constrain exploration to be within the set of tolerable products $\mathcal{S}(u_0, \rho)$. The challenge is that this set is unknown since the customer's latent attributes u_0 are unknown. However, our prior $\mathcal{P}$ gives us reasonable knowledge of which products lie in $\mathcal{S}(u_0, \rho)$ for mainstream customers. In the next section, we will leverage this knowledge to restrict the product set upfront in the Constrained Bandit. As we saw from Theorem 3.4.3, we may as well restrict our focus to serving the subset of mainstream customers, since we cannot hope to do well for all customers.

Remark 3.4.7. In these lower bounds, we have taken $\rho \to 0$ for simplicity. However, the proofs and results hold even when ρ is sizeable. In particular, we only require that there exists at least a single product that is not tolerable for every customer realization, *i.e.*, $|\mathcal{S}(u_0, \rho)| < d$ for all u_0.

3.5 Constrained Bandit Algorithm

We have so far established that both classical bandit algorithms and the greedy algorithm may fail to perform well on *any* customer. We now propose a two-step procedure, where we play a bandit strategy after constraining our action space to a restricted set of products that are carefully chosen using an integer program. In §3.5.3, we will prove that this simple modification guarantees good performance on a significant fraction of customers.

3.5.1 Intuition

As shown in Theorem 3.4.4, classical bandit algorithms fail because of over-exploration. Bandit algorithms rely on an early exploration phase where customers are offered random products; the feedback from these products is then used to infer the customer's low-dimensional preference model, in order to inform future (relevant) recommendations during the exploitation phase. However, in the presence of customer disengagement, the algorithm doesn't get to reap the benefits of exploitation since the customer likely disengages from the platform during the exploration phase after receiving several irrelevant recommendations. This is not to say that learning through exploration is a bad strategy. Theorem 3.4.5 shows that greedy exploitation-only algorithm also under-perform by under-exploring, and getting stuck in sub-optimal fixed points.

This can be harmful since the platform misses out on its key value proposition of learning customer preferences and matching them to their preferred products.

These results suggest that a platform can only succeed by avoiding poor early recommendations. Since we don't know the customer's preferences, this is impossible to do in general; however, our key insight is that a probabilistic approach is still feasible. In particular, the platform has knowledge of the distribution of customer preferences $\mathcal{P}$ from past customers, and can transfer this knowledge to avoid products that do not meet the tolerance threshold of most customers. We formulate this product selection problem as an integer program, which ensures that any recommendations within the optimal restricted set are acceptable to most customers. After selecting an optimal restricted set of products, we follow a classical bandit approach (*e.g.,* linear UCB by Abbasi-Yadkori et al. 2011). Under this approach, if our new customer is a mainstream customer, she is unlikely to disengage from the platform even during the exploration phase, and will be matched to her preferred product. However, if the new customer is a tail customer, her preferred product may not be available in our restricted set, causing her to disengage. This result is shown formally in Theorem 3.5.5 in the next section. Thus, we compromise performance on tail customers to achieve good performance on mainstream customers. Theorem 3.4.3 shows that such a tradeoff is necessary, since it is impossible to guarantee good performance on *every* customer.

We introduce a set diameter parameter γ in our integer program formulation. This parameter can be used to tune the size of the restricted product set based on our prior $\mathcal{P}$ over customer preferences. Larger values of γ increase the risk of customer disengagement by introducing greater variability in product relevance, but also increase the likelihood that the customer's preferred product lies in the set. On the other hand, smaller values of γ decrease the risk of customer disengagement *if* the customer's preferred product is in the restricted set, but there is a higher chance that the customer's preferred product is not in the set. Thus, appropriately choosing this parameter is a key ingredient of our proposed algorithm. We discuss how to choose γ at the end of §3.5.3.

3.5.2 Constrained Exploration

We seek to find a restricted set of products that cater to a large fraction of customers (which is measured with respect to the distribution $\mathcal{P}$ over customer attributes), but are not too "far" from each other (to limit exploration). Before we describe the problem, we introduce notation

that captures the likelihood of a product being relevant for the new customer:

Definition 3.5.1. $\mathcal{C}_i(\rho)$ is the probability of product i satisfying the new customer's tolerance level:

$$\mathcal{C}_i(\rho) = \mathbb{P}_{u_0 \sim \mathcal{P}}(i \in \mathcal{S}(u_0, \rho)),$$

where $\mathcal{S}(u_0, \rho)$ is given by Definition 3.4.2.

Recall that $\mathcal{S}(u_0, \rho)$ is the set of tolerable products for a customer with latent attributes u_0. Given that u_0 is unknown, $\mathcal{C}_i(\rho)$ captures the probability that product i is relevant to the customer with respect to the distribution $\mathcal{P}$ over random customer preferences. In the presence of disengagement, we seek to explore over products that are likely to satisfy the new customer's tolerance level. For example, mainstream products may be tolerable for a large probability mass of customers (with respect to $\mathcal{P}$) while niche products may only be tolerable for tail customers. Thus, $\mathcal{C}_i(\rho)$ translates our prior on customer latent attributes to a likelihood of tolerance over the space of products. Computing $\mathcal{C}_i(\rho)$ using Monte Carlo simulation is straightforward: we generate random customer latent attributes according to $\mathcal{P}$, and count the fraction of customers for which product i was within the customer's tolerance threshold of ρ from the customer's preferred product V_*.

As discussed earlier, a larger product set increases the likelihood that the new customer's preferred product is in the set, but it also increases the likelihood of disengagement due to poor recommendations during the exploration phase. However, the key metric here is not the number of products in the set, but rather the similarity of the products in the set. In other words, we wish to restrict product diversity in the set to ensure that all products are tolerable to mainstream customers. Thus, we define

$$D_{ij} = \|V_i - V_j\|_2,$$

the Euclidean distance between the (known) features of products i and j, *i.e.,* the similarity between two products. We seek to find a subset of products such that the distance between any pair of products is bounded by the set diameter γ. Let $\phi_{ij}(\gamma)$ be an indicator function that

determines whether $D_{ij} \leq \gamma$. Hence,

$$\phi_{ij}(\gamma) = \begin{cases} 1 & \text{if } D_{ij} \leq \gamma\,, \\ 0 & \text{otherwise}\,. \end{cases}$$

Note that γ and ρ are related. When the customer tolerance ρ is large, we will choose larger values of the set diameter γ and vice-versa. We specify how to choose γ at the end of §3.5.3.

The objective is to select a set of products, which together have a high likelihood of containing the customer's preferred match under the distribution over customer preferences $\mathcal{P}$ (*i.e.,* high $\mathcal{C}_i(\rho)$), with the constraint that no two products are too dissimilar from each other (*i.e.,* pairwise distance greater than γ). We propose solving the following product selection integer program:

$$\mathbf{OP}(\gamma) = \max_{\mathbf{x},\mathbf{z}} \sum_{i=1}^{n} C_i(\rho) x_i \tag{3.3a}$$

$$\text{s.t. } z_{ij} \leq x_i, \quad i = 1, \ldots, n, \tag{3.3b}$$

$$z_{ij} \leq x_j, \quad j = 1, \ldots, n, \tag{3.3c}$$

$$z_{ij} \geq x_i + x_j - 1, \quad i = 1, \ldots, n, \quad j = 1, \ldots, n, \tag{3.3d}$$

$$z_{ij} \leq \phi_{ij}(\gamma), \quad i = 1, \ldots, n, \quad j = 1, \ldots, n, \tag{3.3e}$$

$$x_i \in \{0, 1\} \quad i = 1, \ldots, n. \tag{3.3f}$$

The decision variables in the above problem are $\{x_i\}_{i=1}^n$ and $\{z_{i,j}\}_{i,j=1}^n$. In particular, x_i in $\mathbf{OP}(\gamma)$ defines whether product i is included in the restricted set, and $z_{i,j}$ is an indicator variable for whether both products i and j are included in the restricted set. Constraints (3.3b) – (3.3e) ensure that only products that are "close" to each other are selected.

Solving $\mathbf{OP}(\gamma)$ results in a set of products (products for which the corresponding x_i is 1) that maximizes the likelihood of satisfying the new customer's tolerance level, while ensuring that every pair is within γ distance from each other.

Algorithm 2 presents the Constrained Bandit (CB) algorithm, where the second phase follows the popular linear UCB algorithm (Abbasi-Yadkori et al. 2011). There are two input parameters: λ (the standard regularization parameter employed in the linear bandit literature, see, *e.g.,* Abbasi-Yadkori et al. 2011) and γ (the set diameter). We discuss the selection of γ and the corresponding tradeoffs in the next subsection and in Appendix B.3. As discussed

Algorithm 2 Constrained Bandit(λ,γ)

Step 1: Constrained Exploration:

Solve $\mathbf{OP}(\gamma)$ to get Ξ, the constrained set of products to explore over. Let a_1 be a randomly selected product to recommend in Ξ.

Step 2: Bandit Learning:

for $t \in [T]$ **do**

Observe customer utility, $Y_t = u_0^\top a_t + \varepsilon_t$.

Let $\hat{u}_t = (a_{1:t}^\top a_{1:t} + \lambda I)^{-1} a_{1:t} Y_{1:t}$, and,

$$\mathcal{Q}_t = \left\{ u \in \mathbb{R}^d : \|\hat{u}_t - u\|_{\bar{X}_t} \leq \left(\xi \sqrt{d \log\left(\frac{1 + tL^2}{\delta} \right)} + \sqrt{\lambda} \frac{\rho}{\gamma} \right) \right\}.$$

Let $(u_{opt}, a_t) = \arg\max_{\{i \in \Xi, u \in \mathcal{Q}_t\}} u^\top V_i$.

Recommend product a_t at time t if the customer is still engaged. Stop if the customer disengages from the platform.

earlier, we employ a two-step procedure. In the first step, the action space is restricted to the product set given by $\mathbf{OP}(\gamma)$. This step ensures that subsequent exploration is unlikely to cause a significant fraction of customers to disengage. Then, a standard bandit algorithm is used to learn the customer's preference model and match her with her preferred product through repeated interactions. The main idea remains simple: in the presence of customer disengagement, the platform should be cautious while exploring. Since we are uncertain about the customer's preferences, we optimize exploration for mainstream customers who are more likely to visit the platform.

Remark 3.5.2. The Constrained Bandit uses a fixed exploration set for the entire horizon T. One could alternatively consider updating this set dynamically based on customer feedback, *i.e.,* update our posterior on the customer's preference vector U_0 using noisy observations of the customer utilities, and resolve $\mathbf{OP}(\gamma)$ at each time t. While the latter does not lend a tractable regret analysis, it may yield improved empirical performance. Note that when the disengagement propensity p is high, dynamically updating the exploration set is unlikely to be helpful, since the customer will likely disengage immediately if the initial product set is not relevant (*i.e.,* before we obtain sufficient observations to form a posterior that is significantly different from the prior $\mathcal{P}$).

3.5.3 Theoretical Guarantee

We now show that the Constrained Bandit performs well and incurs regret that scales sublinearly in T over a fraction of customers. We begin by defining $L_{t,\rho,p}$, an indicator variable that captures

whether the customer is still engaged at time t:

Definition 3.5.3. Let,

$$L_{t,\rho,p} = \begin{cases} 1 & \text{Customer engaged until time t}\,, \\ 0 & \text{otherwise.} \end{cases}$$

Clearly, $\mathbb{1}\{L_{T,\rho,p} = 1\} = \Pi_{t=1}^{T} \mathbb{1}\{\Upsilon_t = 0\}$, where we recall that Υ_t is the disengagement decision of the customer at time t. We first show that as $T \to \infty$, $L_{T,\rho,p} = 1$ for some customers, *i.e.,* they remain engaged. Next, we show that most engaged customers are eventually matched to their preferred product. Proofs for all results in this section are deferred to Appendix B.2.

Theorem 3.5.4 shows that the worst-case regret of the Constrained Bandit scales sublinearly in T for a positive fraction of customers. In particular, regardless of the customer tolerance parameter ρ, we can match some subset of customers to their preferred products. Note that this is in stark contrast with both bandit and greedy algorithms (Theorems 3.4.4 and 3.4.5).

Theorem 3.5.4 (Matching Upper Bound for Constrained Bandit)**.** Let u_0 be any realization of the latent user attributes from $\mathcal{P}$. Under the assumptions of Setting 1, the Constrained Bandit with set diameter $\gamma = 1/\sqrt{2}$ achieves zero regret with positive probability. In particular, there exists a set $\mathcal{W}_{\lambda,\gamma=\frac{1}{\sqrt{2}}}$ of realizations of customer latent attributes with positive measure under $\mathcal{P}$, *i.e.,*

$$\mathbb{P}\left(\mathcal{W}_{\lambda,\gamma=\frac{1}{\sqrt{2}}}\right) > 0\,,$$

such that, for all $u_0 \in \mathcal{W}_{\lambda,\gamma=\frac{1}{\sqrt{2}}}$, the worst-case regret of the Constrained Bandit algorithm is

$$\sup_{\rho>0} \mathcal{R}^{\mathrm{CB}\left(\lambda,\gamma=\frac{1}{\sqrt{2}}\right)}(T,\rho,p,u_0) = 0\,.$$

Note that this result holds for any value of ρ, *i.e.,* customers can be arbitrarily intolerant of products that are not their preferred product V_*. Thus, the only way to make progress is to immediately recommend their preferred product. This can trivially be done by restricting our product set to a single product, which at the very least caters to *some* customers. This is exactly what we do in Theorem 3.5.4: the choice of $\gamma = 1/\sqrt{2}$ and the product space given in Setting 1 ensures that only a single product will be in our restricted set Ξ. By construction of $\mathbf{OP}(\gamma)$, this will be the most popular preferred product. $\mathcal{W}$ denotes the subset of customers for whom this product is optimal, and this set has positive measure under $\mathcal{P}$ by construction since

we have a discrete number of products. Note that these customers are immediately matched to their preferred product, so it immediately follows that we incur zero regret on this subset of customers.

Theorem 3.5.4 shows that there is nontrivial value in restricting the product set upfront, which cannot be obtained through either bandit or greedy algorithms. However, it considers the degenerate case of constraining exploration to only a single product, which is clearly too restrictive in practice, especially when customers are relatively tolerant (*i.e.,* ρ is not too small). Thus, it does not provide useful insight into how much the product set should be constrained as a function of the customer's tolerance parameter. To answer this question, we move away from the setting described in the simplified setting and consider a fluid approximation of the product space. Since the nature of $\mathbf{OP}(\gamma)$ is complex, letting the product space be continuous $V = [-1, 1]^d$ will help us cleanly demonstrate the key tradeoff in constraining exploration: a larger product set has a higher probability of containing customers' preferred products, but also a higher risk of disengagement. Furthermore, for algebraic simplicity, we shift the mean of the prior over the customer's latent attributes, so $\mathcal{P} = \mathcal{N}(\bar{u}, \frac{\sigma^2}{d} I_d)$, where $\|\bar{u}\|_2 = 1$. This ensures that our problem is not symmetric, which again helps us analytically characterize the solution of $\mathbf{OP}(\gamma)$.

Theorem 3.5.5 shows that the Constrained Bandit algorithm can achieve sublinear regret for a fraction of customers under this albeit stylized setting. More importantly, it yields insights into how we might choose the set diameter γ as a function of the customer's tolerance parameter ρ. In §3.6, we demonstrate the strong empirical performance of our algorithm on real data.

Theorem 3.5.5 (Guarantee for Constrained Bandit Algorithm)**.** Let $\mathcal{P} = \mathcal{N}(\bar{u}, \frac{\sigma^2}{d^2} I_d)$. Also consider a continuous product space $V = [-1, 1]^d$. There exists a set $\mathcal{W}$ of latent customer attribute realizations with positive probability under $\mathcal{P}$, *i.e.,*

$$\mathbb{P}(\mathcal{W}) \geq w \;=\; \left(1 - 2d \exp\left(-\frac{1}{\sigma}\left(1 - \sqrt{1 - \frac{\gamma^2}{4}}\right)\right)\right)\left(1 - 2d \exp\left(-\frac{1}{\sigma^2}\left(\frac{\rho}{\gamma} - \sum_{i=1}^{i=d} \bar{u}_i\right)^2\right)\right),$$

such that for all $u_0 \in \mathcal{W}$ the cumulative regret of the Constrained Bandit is

$$\begin{aligned}\mathcal{R}^{CB(\lambda,\rho)}(T, \rho, p, u_0) &\leq 5\sqrt{Td \log\left(\lambda + \frac{TL}{d}\right)}\left(\sqrt{\lambda}\frac{\rho}{\gamma} + \xi\sqrt{\log(T) + d\log\left(1 + \frac{TL}{\lambda d}\right)}\right) \\ &= \tilde{\mathcal{O}}\left(\sqrt{T}\right).\end{aligned}$$

This result explicitly characterizes the fraction of customers that we successfully serve as a function of the customer tolerance parameter ρ and the set diameter γ. Thus, given a value of ρ, we can choose the set diameter γ to optimize the probability w of this set.

The proof of Theorem 3.5.5 follows in three steps. First, we lower bound the probability that the constrained exploration set Ξ contains the preferred product for a new customer whose attributes are drawn from $\mathcal{P}$. Next, conditioned on the previous event, we lower bound the probability that the customer remains engaged for the entire time horizon T when recommendations are made from the restricted product set Ξ. Lastly, conditioned on the previous event, we can apply standard self-normalized martingale techniques (Abbasi-Yadkori et al. 2011) to bound the regret of the Constrained Bandit algorithm for the customer subset $\mathcal{W}$.

Again, as in Theorem 3.5.4, we see that there can be significant value in restricting the product set upfront that cannot be achieved by classical bandit or greedy approaches. We further see that the choice of the set diameter γ is an important consideration to ensure that the new customer is engaged and matched to her preferred product with as high a likelihood as possible. As discussed earlier, larger values of γ increase the risk of customer disengagement by introducing greater variability in product relevance, but also increase the likelihood that the customer's preferred product lies in the set. On the other hand, smaller values of γ decrease the risk of customer disengagement *if* the customer's preferred product is in the restricted set, but there is a higher chance that the customer's preferred product is not in the set. In other words, we wish to choose γ to maximize w. While there is no closed form expression for the optimal γ, we propose the following approximately optimal choice based on a Taylor series approximation (see details in Appendix B.3):

$$\gamma^* \in \left\{\gamma : \rho = \frac{\sqrt{\sigma}\gamma^2}{2\left(4-\gamma^2\right)^{1/4}} \text{ and } \gamma > 0\right\}.$$

Numerical experiments demonstrate that this approximate value of γ is typically within 1% of the value of γ that maximizes the expression for w given in Theorem 3.5.5; the resulting values of w are also very close (see Appendix B.3). This expression yields some interesting comparative statics: we should choose a smaller set diameter γ when customers are less tolerant (ρ is small) and customer feedback is noisy (σ is large). In practice, we can tune the set diameter through cross-validation.

3.6 Numerical Experiments

We now compare the empirical performance of the Constrained Bandit with the state-of-the-art Thompson sampling (which is widely considered to empirically outperform other bandit algorithms, see, *e.g.,* Chapelle and Li 2011, Russo and Van Roy 2014) and a greedy Bayesian updating policy. We present two sets of empirical results evaluating our algorithm on both synthetic data (§3.6.1), and on real movie recommendation data (§3.6.2).

Benchmarks: We compare our algorithm with (i) linear Thompson Sampling (Russo and Van Roy 2014) and (ii) the greedy Bayesian updating (Algorithm 1) referred to as MLE.

Constrained Thompson Sampling (CTS): To ensure a fair comparison, we consider a Thompson Sampling version of the Constrained Bandit algorithm (see Algorithm 3 below). Recall that our approach allows for any bandit strategy after obtaining a restricted product set based on our (algorithm-independent) integer program $\mathbf{OP}(\gamma)$. We use the same implementation of linear Thompson sampling (Russo and Van Roy 2014) as our benchmark in the second step. Thus, any improvements in performance can be attributed to restricting the product set.

Algorithm 3 Constrained Thompson Sampling (λ,γ)

Step 1: Constrained Exploration:
Solve $\mathbf{OP}(\gamma)$ to get the constrained set of products to explore over, $S_{constrained}$. Let $\hat{u}_1 = \bar{u}$.
Step 2: Bandit Learning:
for $t \in [T]$ **do**
  Sample $u(t)$ from distribution $\mathcal{N}\left(\hat{u}_t, \sigma^2 I_d\right)$.
  Recommend $a_t = \arg\max_{\{i \in S_{constrained}\}} u(t)^\top V_i$ if the customer is still engaged.
  Observe customer utility, $Y_t = U_0^\top a_t + \varepsilon_t$, and update $\hat{u}_t = (V_{a_1:a_t}^\top V_{a_1:a_t} + \lambda I)^{-1} V_{a_1:a_t} Y_{1:t}$
  Stop if the customer disengages from the platform.

3.6.1 Synthetic Data

We generate synthetic data and study the performance of all three algorithms as we increase the customer's disengagement propensity $p \in [0, 1]$. A low value of p implies that customer disengagement is not a salient concern, and thus, one would expect Thompson sampling to perform well in this regime. On the other hand, a high value of p implies that customers are extremely intolerant of poor recommendations, and thus, all algorithms may fare poorly. We find that Constrained Thompson Sampling performs comparably to vanilla Thompson Sampling when p is low, and offers sizeable gains over both benchmarks when p is medium or large.

Data generation: We consider the standard collaborative filtering problem (described earlier) with 10 products. Recall that collaborative filtering fits a low rank model of latent customer preferences and product attributes; we take this rank[2] to be 2. We generate product features from a multivariate normal distribution with mean $[1, 5]^\top \in \mathbb{R}^2$ and variance $0.3 \cdot I_2 \in \mathbb{R}^{2\times 2}$, where we recall that I_d is the $d \times d$ identity matrix. Similarly, latent user attributes are generated from a multivariate normal with with mean $[2, 2]^\top \in \mathbb{R}^2$ and variance $2 \cdot I_2 \in \mathbb{R}^{2\times 2}$. These values ensure that, with high probability for every customer, there exists a product on the platform that generates positive utility. Note that the product features are known to the algorithms, but the latent user attributes are unknown. Finally, we take our noise $\epsilon \sim \mathcal{N}(0, 5)$, the customer tolerance ρ to be generated from a truncated $\mathcal{N}(0, 1)$ distribution, and the total horizon length $T = 1000$. All algorithms are provided with the distribution of customer latent attributes, the distribution of the customer tolerance ρ, and the horizon length T. They are not provided with the noise variance, which needs to be estimated over time. Finally, we consider several values of the disengagement propensity, *i.e.,* $p \in \{1\%, 10\%, 50\%, 100\%\}$, to capture the value of restricting the product set with varying levels of customer disengagement.

Engagement Time: We use average customer engagement time (*i.e.,* the average time that a customer remains engaged with the platform, up to time T) as our metric for measuring algorithmic performance. As we have seen in earlier sections, customer engagement is necessary to achieve low cumulative regret. Furthermore, it is a more relevant metric from a managerial perspective since higher engagement is directly related with customer retention and loyalty, as well as the potential for future high quality/revenue customer-product matches.

Results: Figure 3.2 shows the customer engagement time averaged over 1000 randomly generated users (along with the 95% confidence intervals) for all three algorithms as we vary the disengagement propensity p from 1% to 100%. As expected, when $p = 1\%$ (*i.e.,* customer disengagement is relatively insignificant), TS performs well, and CTS performs comparably. However, greedy Bayesian updating is likely to converge to a suboptimal product outside of the customer's relevance set, and continues to recommend this product until the customer eventually disengages. As we increase p, all algorithms achieve worse engagement, since customers become considerably more likely to leave the platform. As expected, we also see that CTS starts

[2]We choose a small rank based on empirical experiments showing that collaborative filtering models perform better in practice with small rank (Chen and Chi 2018). Our results remain qualitatively similar with higher rank values.

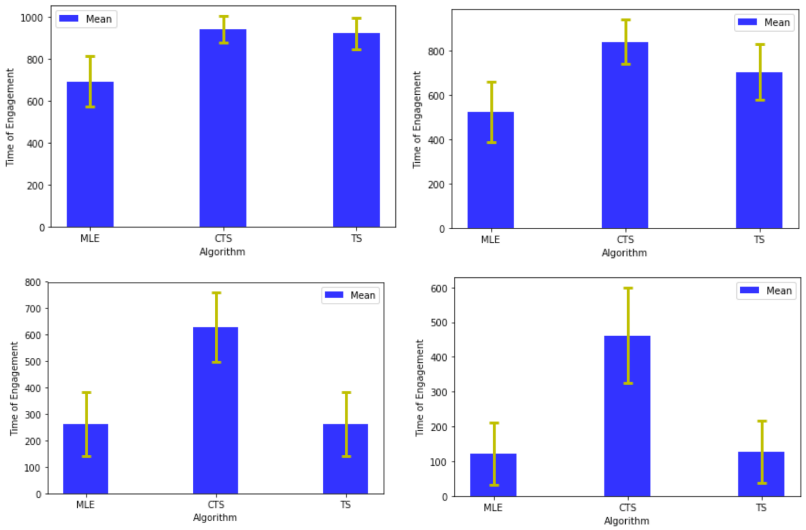

Figure 3.2: Time of engagement and 95% confidence intervals averaged over 1000 randomly generated customers for disengagement propensity p values of 1%, 10%, 50%, and 100% respectively.

to significantly outperform the other two benchmark algorithms as p increases. For instance, the mean engagement time of CTS improves over the engagement time of the benchmark algorithms by a factor of 2.2 when $p = 50\%$ and by a factor or 4.4 when $p = 100\%$. Thus, we see that restricting the product set is critical when customer disengagement is a salient feature on the platform.

A recent report by Smith (2018) notes that an average worker receives as many as 121 emails on average per day. Furthermore, the average click rate for retail recommendation emails is as low as 2.5%. These number suggest that customer disengagement is becoming increasingly salient, and we argue that constraining exploration on these platforms to quickly match as many customers as possible to a tolerable product is a key consideration in recommender system design.

3.6.2 Case Study: Movie Recommendations

We now compare CTS to the same benchmarks on MovieLens, a publicly available movie recommendations data collected by GroupLens Research. This dataset is widely used in the academic community as a benchmark for recommendation and collaborative filtering algorithms (Harper and Konstan 2016). Importantly, we no longer have access to the problem parameters (*e.g.*, ρ) and must estimate them; we discuss simple heuristics for estimating these parameters.

Data Description & Parameter Estimation

The MovieLens dataset contains over 20 million user ratings based on personalized recommendations of 27,000 movies to 138,000 users. We use a random sample (provided by MovieLens) of 100,000 ratings from 671 users over 9,066 movies. Ratings are made on a scale of 1 to 5, and are accompanied by a time stamp for when the user submitted the rating. The average movie rating is 3.65.

The first step in our analysis is identifying likely disengaged customers in our data. We will argue that the number of user ratings is a proxy for disengagement. In Figure 3.3, we plot the histogram of the number of ratings per user. Users provide an average of 149 ratings, and a median of 71 ratings. Clearly, there is high variability and skew in the number of

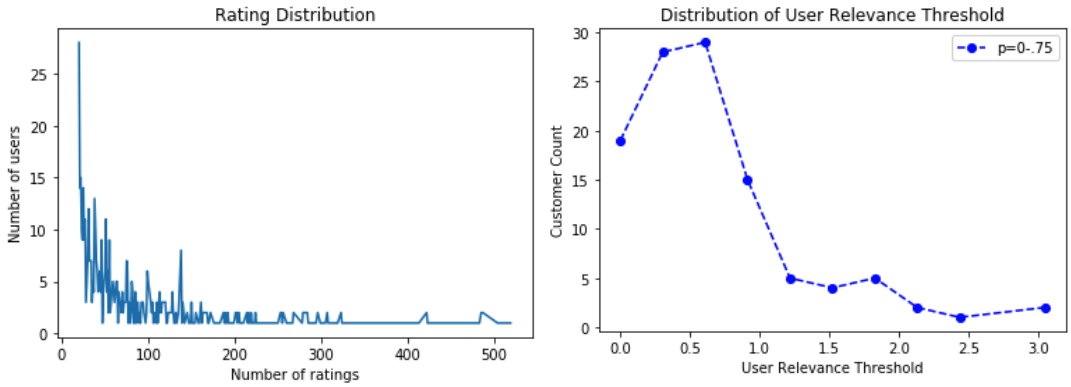

Figure 3.3: On left, the histogram of user ratings in MovieLens data. On right, the empirical distribution of ρ, the customer-specific tolerance parameter, across all disengaged users for a fixed customer disengagement propensity $p = .75$. This distribution is robust to any choice of $p \in (0, .75]$

ratings that users provide. We argue that there are two primary reasons why a customer may stop providing ratings: (i) satiation and (ii) disengagement. Satiation occurs when the user has exhausted the platform's offerings that are relevant to her, while disengagement occurs when the user is relatively new to the platform and does not find sufficiently relevant recommendations to justify engaging with the platform. Thus, satiation applies primarily to users who have provided many ratings (right tail of Figure 3.3), while disengagement applies primarily to users who have provided very few ratings (left tail of Figure 3.3).

Accordingly, we consider the subset of users who provided fewer that 27 ratings (bottom 15% of users) as *disengaged* users. We hypothesize that these users provided a low number of ratings because they received recommendations that did not meet their tolerance threshold. This hypothesis is supported by the ratings. In particular, the average rating of disengaged users is 3.56 (standard error of 0.10) while the average rating of the remaining (engaged) users is 3.67

(standard error of 0.04). A one-way ANOVA test (Welch 1951) yields a F-statistic of 29.23 and a p-value of 10^{-8}, showing that the difference is statistically significant and that disengaged users dislike their recommendations more than engaged users. This finding relates to our results in §3.2, *i.e.,* disengagement is related to the customer-specific quality of recommendations made by the platform.

Estimating latent user and movie features: We need to estimate the latent product features $\{V_i\}_{i=1}^n$ as well as the distribution $\mathcal{P}$ over latent user attributes from historical data. Thus, we use low rank matrix factorization (Ekstrand et al. 2011) on the ratings data (we find that a rank of 5 yields a good fit) to derive $\{U_i\}_{i=1}^m$ and $\{V_i\}_{i=1}^n$. We fit a normal distribution $\mathcal{P}$ to the latent user attributes $\{U_i\}_{i=1}^m$, and use this to generate new users; we use the latent product features as-is.

Estimating the tolerance parameter ρ: Recall that ρ is the maximum utility reduction (with respect to the utility of the unknown optimal product V_*) that a customer is willing to tolerate before disengaging with probability p. In our theory, we have so far assumed that there is a single known value of ρ for all customers. However, in practice, it is likely that ρ may be a random value that is sampled from a distribution (*e.g.,* there may be natural variability in tolerance among customers), and further, the distribution of ρ may be different for different customer types (*e.g.,* tail customer types may be more tolerant of poor recommendations since they are used to having higher search costs for niche products). Thus, we estimate the distribution of ρ as a function of the user's latent attributes u_0 using maximum likelihood estimation, and sample different realizations for different incoming customers on the platform. We detail the process of this estimation next.

In order to estimate ρ for a user, we consider the time series of ratings provided by a single user with latent attributes u_0 in our historical data. Clearly, disengagement occurred when the user provided the last rating to the platform, and this decision was driven by both the user's disengagement propensity p, and tolerance parameter ρ. For a given p and ρ, let t^{leave} denote the last rating of the user, and $a_1, ...a_{t_{leave}}$ be the recommendations made to the user until time t_{leave}. Then, the likelihood function of the observation sequence is:

$$\mathcal{L}(p,\rho) = p(1-p)^{\left(t^{leave} - \sum_{i=1}^{(t^{leave}-1)} \mathbb{1}\{a_i \in \mathcal{S}(u_0,\rho)\}\right)},$$

where we recall that $\mathcal{S}(u_0, \rho)$ defines the set of products that the user considers tolerable. Since u_0 and V_i are known apriori (estimated from the low rank model), $\mathcal{S}(u_0, \rho)$ is also known apriori for any given value of ρ. Hence, for any given value of p, we can estimate the most likely user-specific tolerance parameter ρ using the maximum likelihood estimator of $\mathcal{L}(p, \rho)$. In Figure 3.3, we also plot the overall estimated empirical distribution of ρ for our subset of disengaged users. We see that more than 88% of disengaged users have an estimated tolerance parameter of less than 1.2, *i.e.,* they consider disengagement if the recommendation is more than 1 star away from what they would rate their preferred movie. As we may expect, very few disengaged users have a high estimated value of ρ, suggesting that they have high expectations on the quality of recommendations.

One caveat of our estimation strategy is that we are unable to identify both p and ρ simultaneously; instead, we estimate the user-specific distribution of ρ and perform our simulations for varying values of the disengagement propensity p. Empirically, we find that our estimation of ρ is robust to different values of p, *i.e.,* for any value of $p \in (0, .75]$, we observe that our estimated distribution of ρ distribution does not change. Thus, we believe that this strategy is sound.

Results

Similar to §3.6.1, we compare Constrained Thompson Sampling against our two benchmarks (Thompson Sampling and greedy Bayesian updating) based on average customer engagement time. We use a random sample of 200 products, and take our horizon length $T = 100$.

Figure 3.4 shows the customer engagement time averaged over 1000 randomly generated users (along with the 95% confidence intervals) for all three algorithms as we vary the disengagement propensity p from 1% to 100%. Again, we see similar trends as we saw in our numerical experiments on synthetic data (§3.6.1). When $p = 1\%$ (*i.e.,* customer disengagement is relatively insignificant), all algorithms perform well, and CTS performs comparably. As we increase p, all algorithms achieve worse engagement, since customers become considerably more likely to leave the platform. As expected, we also see that CTS starts to significantly outperform the other two benchmark algorithms as p increases. For instance, the mean engagement time of CTS improves over the engagement time of the benchmark algorithms by a factor of 1.26 when $p = 10\%$, by a factor of 1.66 when $p = 50\%$ and by a factor or 1.8 when $p = 100\%$. Thus, our main finding remains similar on real movie recommendation data: restricting the product set

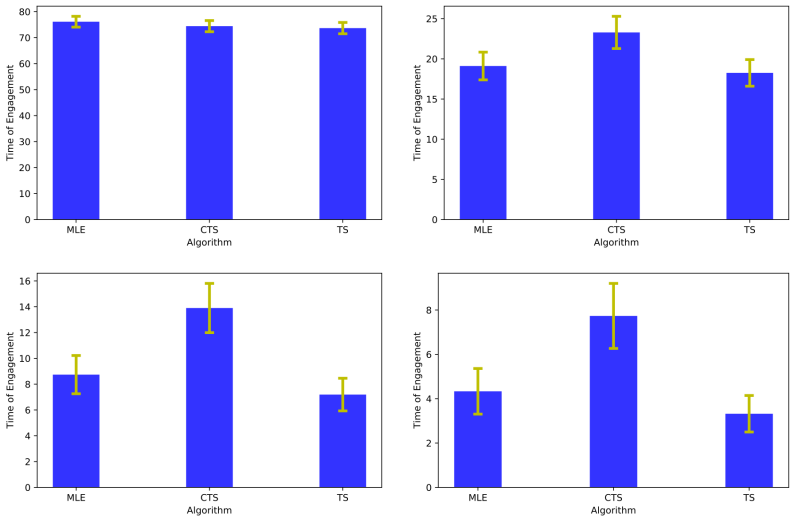

Figure 3.4: Time of engagement and 95% confidence intervals on MovieLens data averaged over 1000 randomly generated customers for disengagement propensity p values of 1% (top left), 10% (top right), 50% (bottom left), and 100% (bottom right) respectively.

is critical when customer disengagement is a salient feature on the platform.

3.7 Conclusions

We consider the problem of sequential product recommendation when customer preferences are unknown. First, using a sequence of ad campaigns from a major airline carrier, we present empirical evidence suggesting that customer disengagement plays an important role in the success of recommender systems. In particular, customers decide to stay on the platform based on the quality of recommendations. To the best of our knowledge, this issue has not been studied in the framework of collaborative filtering, a widely-used machine learning technique. We formulate this problem as a linear bandit, with the notable difference that the customer's horizon length is a function of past recommendations. Our formulation bridges two disparate literatures on bandit learning in recommender systems, and customer disengagement modeling.

We then prove that this problem is fundamentally hard, *i.e.,* no algorithm can keep all customers engaged. Thus, we shift our focus to keeping a large number of customers (*i.e.,* mainstream customers) engaged, at the expense of tail customers with niche preferences. Our results highlight a necessary tradeoff with clear managerial implications for platforms that seek to make personalized recommendations. Unfortunately, we find that classical bandit learning algorithms as well as a simple greedy Bayesian updating strategy perform poorly, and can fail to keep any customer engaged. To solve this problem, we propose modifying bandit learning

strategies by constraining the action space upfront using an integer program. We prove that this simple modification allows our algorithm to perform well (*i.e.,* achieve sublinear regret) for a significant fraction of customers. Furthermore, we perform extensive numerical experiments on real movie recommendations data that demonstrate the value of restricting the product set upfront. In particular, we find that our algorithm can improve customer engagement with the platform by up to 80% in the presence of significant customer disengagement.

Chapter 4

Dynamic Pricing with Unknown Non-Parametric Demand and Limited Price Changes

4.1 Introduction

Firms constantly innovate and introduce new products in order to compete and better position themselves in a rapidly changing business environment. Each year, billions of dollars are invested on product innovation and new product launches (Willemot et al. 2015). Unfortunately, not all new product launches succeed. In particular, Willemot et al. (2015) states that almost 15% of the total new products launched in the market each year are unsuccessful and are taken off shelves before the end of their life cycle. In fact, a recent survey states that more than 72% of all new products do not meet their revenue targets and attributes such failures to pricing (Carmichael 2014, Huang et al. 2007). While on one hand, dynamic pricing gives retailers the opportunity to learn price elasticity, on the other hand, uninformed pricing can have unexpected consequences and can lead to product failures.

One potential solution towards informed pricing decisions for new products is to incorporate demand learning within the dynamic pricing framework. Given the ubiquitous nature of pricing for new products, it is not surprising that many researchers have studied the combined pricing and learning problem. Since the introduction of the combined problem by Rothschild (1974), various extensions and algorithms have been proposed. The central tradeoff in this problem setting is that of *exploration-exploitation*. While the retailer can *learn* demand re-

sponse by *exploring* prices, maximizing revenue implies *exploiting* from the already explored prices. Near optimal algorithms ensure that the two competing goals of *learning* and *earning* are appropriately balanced.

Both the analysis as well as the performance of the proposed algorithms depend on the underlying assumptions made, particularly about the demand as well as the error in demand realizations. The unknown demand is generally parameterized by a set of parameters that are learned over time (Bertsimas and Perakis 2006, den Boer and Zwart 2013, Keskin and Zeevi 2014, Cheung et al. 2015 amongst others). Common algorithms use Maximum Likelihood Estimation (MLE) techniques to ensure fast convergence to the unknown optimal price. But choosing a parametric form, particularly when demand is unknown can be challenging. In order to alleviate these concerns, few studies (see for e.g., Besbes and Zeevi 2009, 2015) have focused on assuming non-parametric demand and have proposed near optimal learning and pricing algorithms. Nevertheless, to the best of our knowledge, all of the non-parametric pricing and learning studies assume that prices can be potentially changed in every period. This becomes a tenuous assumption, especially in the offline retail setting where changing prices involves changing labels, which can be operationally costly (Netessine 2006). Moreover, even in the online setting, the negative effect of frequent price changes on consumer trust has been well established (Garbarino and Lee 2003). For example, if consumers observe frequent price changes and observe price discrimination that is based on latent customer attributes unknown to the customers, they are less likely to visit the store again. In light of these issues, we focus on the dynamic pricing and demand learning problem with no parametric assumptions on the demand when retailers prefer very limited price changes.

Non-parametric demand with limited price changes adds substantial difficulty to the pricing problem. While changing prices from one time period to another could be optimal for the *learning and earning* objectives, pricing constraints could imply that one has to keep a fixed price (even when it is suboptimal) for a group of customers before prices can be changed again. Furthermore, since demand is non-parametric, exact MLE-based estimation of parameters becomes effectively infeasible. Following these constraints, we focus on the class of demand functions that satisfy general smoothness assumptions and propose the Stochastic Limited Price Experimentation (*SLPE*) policy that uses three key ideas. First, since price changes are limited and demand is stochastic, once a price is chosen to experiment upon, it is fixed until a high probability estimate of demand at that price can be obtained. Second, since experimentation

prices can potentially remain fixed for a long time period, future experimentation prices are carefully selected based on a high probability estimate of the optimal price using previous price experimentations. Third, over time as the high probability region containing the optimal price becomes smaller and smaller, we can use the same price for longer time periods, thereby ensuring very low cumulative price changes, without incurring extra revenue loss.

We analyze the proposed pricing policy both analytically and numerically to demonstrate its effectiveness. We show that when the unknown demand function is locally linear infinitesimally close to the optimal price, the demand and the revenue functions satisfy commonly imposed continuity assumptions and the error in demand observations at two different prices satisfy *two-point bandit feedback*, the rate of regret of the *SLPE* policy is $\tilde{\mathcal{O}}(\sqrt{T})$, and the rate of price changes is $\mathcal{O}(\log\log T)$. The two-point bandit feedback structure was introduced by Agarwal and Dekel (2010) and independently by Nesterov (2011) to analyze zero-order optimization methods when a direct gradient calculation is either infeasible or ill-posed. Subsequently, various researchers have constructed optimal algorithms under the two-point feedback assumption in various settings (see for example, Ghadimi and Lan 2013, Duchi et al. 2015, Shamir 2017). Our work is also part of this growing literature. We establish that the rate of regret of the policy is near optimal (upto a logarithmic factor), while making very limited price changes. We note that the previous best known price change guarantee is $\mathcal{O}(\log T)$. Hence, our analysis reveals a class of non-parametric functions for which the best known price change guarantee improves from $\mathcal{O}(\log T)$ to $\mathcal{O}(\log\log T)$. To the best of our knowledge, we are also the first to leverage two-point bandit feedback to reduce the total number of price changes while maintaining near optimal regret guarantee.

We also perform extensive numerical experiments to empirically test the performance of the proposed method. Examples on synthetic data show that the proposed algorithm reduces the number of price changes substantially over the best performing benchmark. While one would expect that this reduction in price changes could lead to an increase in terms of regret, we find that our policy performs at-par with benchmark methods.

4.1.1 Literature Review

Dynamic pricing under demand uncertainty has been widely studied in operations research and operations management. While various learning methods have been proposed, non-parametric demand coupled with limited price changes lead to significant challenges in revenue maximiza-

tion.

Pricing and Learning: Pricing has a rich history. Since its introduction by Rothschild (1974), numerous extensions have been proposed. Review papers and references therein (see for e.g., Aviv and Vulcano 2012 and den Boer 2015) provide a comprehensive overview of the current advancements in the field. While many of the early works tackled pricing problems assuming that the underlying demand is known, recent studies have focused on the problem of *learning* the demand while *earning* revenue. Assumptions on the demand, particularly assuming a parametric vs non-parametric form, substantially changes the learning and pricing problem and the subsequent analysis.

Parametric models for pricing and learning: Many researchers have proposed dynamic pricing algorithms under various assumptions on the unknown parametric demand. Under this setting, a parametric model of demand (often linear in price) is assumed and the unknown parameters of the demand model are learned in a sequential manner. den Boer and Zwart (2013) assume a linear price demand relationship and propose a controlled variance pricing policy that accomplishes sufficient learning by introducing variance in the dynamically chosen prices. Keskin and Zeevi (2014) find sufficient conditions under which a pricing policy is optimal in terms of its rate of regret in the linear demand setting. Handel and Misra (2015) and Bertsimas and Vayanos (2017) use techniques from robust optimization to solve the dynamic pricing with unknown demand parameters. More recently, Cohen et al. (2016), Qiang and Bayati (2016), Javanmard and Nazerzadeh (2016), Ban and Keskin (2017), Elmachtoub et al. (2018), Bastani et al. (2019) and others have used parametric models of demand that not only include price but other product related covariates for optimal pricing decisions. None of these papers explicitly account for price changes. Hence, they could potentially lead to linear rate of increase in the number of price changes with respect to the total time horizon.

Non-parametric models for pricing and learning: Dynamic pricing with non-parametric demand has also been extensively studied. Under this setting, demand is assumed to belong to a family of demand curves characterized by some structural properties of the revenue function such as its concavity and unimodality. Besbes and Zeevi (2015) construct a misspecified algorithm that uses linear models of demand to optimize subsequent prices. Somewhat surprisingly, they find that even though misspecified, their constructed policy is near optimal. Following their intuition, our proposed policy also constructs linear interpolations of the unknown demand and

uses these approximations for future pricing. Nevertheless, since their proposed algorithm does not account for price changes, they incur at least $\mathcal{O}(\log T)$ price changes, substantially higher than the proposed the rate of price change of the propsed algorithm in this chapter: $\mathcal{O}(\log \log T)$. Others such as Besbes and Zeevi (2009), Lei et al. (2014), Dokka Venkata Satyanaraya et al. (2018), Chen et al. (2017a) and Chen and Gallego (2018) also construct non-parametric pricing policies but do not account for price changes. As a result, the proposed algorithm and the theoretical lower bounds we establish in this paper are fundamentally different from existing work.

Dynamic Pricing with Limited Price Experimentation: The effects of dynamic pricing on consumer behavior have also been extensively studied. For example, PK Kannan (2001) hypothesized that extensive price changes based on dynamic pricing policies that discriminate between consumers can lead to mistrust amongst consumers. These claims were substantiated through experiments conducted by Garbarino and Lee (2003) and Haws and Bearden (2006). These studies point out the inherent tension between frequent price changes and revenue maximization.

In the operations management literature, constraints on price changes are not new. Feng and Gallego (1995) consider the optimal timing of a single price change to maximize revenue. Similarly, Bitran and Mondschein (1997) optimize dynamic prices given a prespecified schedule of price changes. Netessine (2006) considers the problem of optimal dynamic pricing with infrequent price changes and inventory constraints. Nevertheless, these papers do not account for demand learning and hence, they are substantially different from the current work. Closer to our work, Broder (2011) first formulated the demand learning problem with limited price changes. Focusing on the parametric demand setting, the authors construct a pricing algorithm based on the Maximum Likelihood Estimation method that incurs $\tilde{\mathcal{O}}(\sqrt{T})$ regret with $\mathcal{O}(\log T)$ price changes. More recently, Cheung et al. (2015) focus on a demand learning problem with limited price changes in a parametric setting where the actual demand is one among finite known demand curves. Similarly, Chen and Chao (2017) focus on a parametric family of unknown demand functions, but with the added complication of censored demand. Finally, Cohen et al. (2015) analyze how a single price (average price) can be near optimal in many parametric demand settings. Nevertheless, since no price changes are allowed, the single price policy they propose would earn a linear rate of regret in our setting.

All the studies cited above differ substantially from the current work due to their parametric demand assumption. For example, the MLE based algorithm of Broder (2011) works with a known parametric demand form. Similarly, the pricing policy proposed by Cheung et al. (2015) uses a data-driven approach to construct candidate demand curves that are known to the retailer. In contrast, in this chapter, we do not impose any parametric assumptions on the demand. In addition, we do not assume that historical sales data is available. Our proposed *SLPE* policy uses a fundamentally different intuition. Our price selection procedure uses piecewise linear approximations to provide near optimal prices for experimentation. We ensure limited price changes by fixing a selected price until demand at that price cannot be estimated with high certainty. Our proposed policy incurs $\tilde{\mathcal{O}}(\sqrt{T})$ rate of regret with $\mathcal{O}(\log\log T)$ price changes. We further show superior empirical performance through an extensive numerical study.

Bandit convex optimization and two-point bandit feedback: Our work is also related to the Continuum Armed Bandit problem, an extension to the classical bandit learning introduced by Lai and Robbins (1985). In this setting, decisions (arms) are a subset of $\mathbb{R}$ and rewards are a continuous function of the decision (Agrawal 1995). Many extensions, particularly based on changing the underlying assumption of the reward function have been proposed. See, for example, Kleinberg (2005) and Auer et al. (2007). Nevertheless, these algorithms fundamentally differ from the current algorithm because they rely on discretization of the decision space and are robust to the adversarial setting. For example, the CAB1 algorithm of Kleinberg (2005) incurs a regret of $\tilde{\mathcal{O}}(T^{2/3})$. More recently, Agarwal et al. (2011) have proposed a bandit learning algorithm that sequentially reduces the continuous action space to converge to the optimal decision. Similar to our chapter's policy, their algorithm also constructs high probability bounds around sampled decision points in order to discard suboptimal decision regions. Nevertheless, their proposed decision point selection is considerably different from the decision point selection of this chapter's algorithm and is based on stochastic bisection search. Their algorithm and analysis leverages convexity. Instead, we focus on smoothness and consider the two-point bandit feedback case. Hence, we obtain a strong numerical as well as analytical performance. The *SLPE* algorithm we introduce in this chapter, provably reduces the number of price changes to $\mathcal{O}(\log\log T)$ from the $\mathcal{O}(\log T)$ price change of the proposed algorithm of Agarwal et al. (2011) and shows better empirical performance on all metrics in numerical experiments. Finally, limited price changes can also be associated with batched learning (Somerville 1954). In this setting,

learning progresses with batched outcomes and switching between batches is associated with costs. The objective is to find optimal batch sizes that minimizes the cost of switching between batches while maximizing an unknown objective. Perchet et al. (2016) solve the batched learning problem in the context of clinical trials. Similar to the current chapter, they find that very few batches lead to optimal regret bounds. More recently, Simchi-Levi and Xu (2019) explore the relation between switching costs and phase transitions. Similarly, Simchi-Levi et al. (2019) also study network revenue management with switching costs. Nevertheless, all these papers focus on discrete arm bandits instead of continuum armed bandits studied in this chapter. This leads to fundamentally different analysis techniques between the current work and the papers referenced above. For instance, the regret bound that we establish in this chapter, relies on the smoothness of the unknown objective revenue function. Instead, Perchet et al. (2016) use the gap between the optimal arm and the second best arm in order to control the batch size and the regret. By construction, since our setting includes continuous arms, this gap can be arbitrarily small and hence the same techniques cannot be applied in our analysis.

Finally our work is also related to derivative free optimization, and in particular recent studies on two-point bandit feedback. As mentioned, the two-point bandit feedback structure was introduced by Agarwal and Dekel (2010), and independently by Nesterov (2011). This structure posits that the objective function can be evaluated at two different query points with identical stochastic error. This assumption becomes particularly useful in the derivative free optimization setting where the decision maker has access to function evaluations only to make optimal decisions. Both papers use randomized gradient estimates that are constructed using the two-point bandit feedback, and then fed into gradient search similar to first order methods. Since then, researchers have analyzed optimal regret rates under various settings. In particular, Duchi et al. (2015) consider the class of strongly smooth functions and show that the optimal rate of regret is $\mathcal{O}(\sqrt{T})$. More recently, Shamir (2017) extends this analysis to include convex and Lipschitz continuous functions. All these papers consider the multi-dimensional functional minimization case, and focus on the dependence of the rate of regret on the dimensions of the decision space. Evidently, they update decisions in every round, thereby having a $\mathcal{O}(T)$ price change guarantee. Instead, the focus of the current chapter is on the single dimension case, but with very limited price changes. Hence, the analysis and the algorithm are both considerably different from the aforementioned papers. Furthermore, we also establish how to extend the results under a relaxed version of the two-point bandit feedback assumption that can be of

independent interest.

4.1.2 Contributions

- *Dynamic pricing for non-parametric unknown demand with limited price changes:* We analyze the problem of dynamic pricing and demand learning with limited price changes under non-parametric demand, when the feasible prices belong to a continuous price range. While prior researchers have investigated separately, the problem of non parametric demand learning as well as the one with limited price changes, to the best of our knowledge, this chapter is the first that studies these two problems together. In many settings, dynamic pricing and learning can be tenuous with either of the two assumptions. For example, in offline retail, the parametric form of the demand for a new product could be hard to select a-priori. Similarly, prices cannot be changed for every incoming customer. Furthermore, choosing a predefined price ladder for a new product can be a hard problem by itself. For example, if the price ladder is too coarse, the optimal price learned from the discrete price ladder can be far from the real optimal price, which can lead to a linear rate of regret.

- *Pricing algorithm with piece-wise linear estimates and provable regret guarantee:* We propose the *Stochastic Limited Price Experimentation (SLPE)* policy that uses linear interpolations of the unknown demand in order to generate future prices. We show that the rate of regret for the *SLPE* policy is $\tilde{\mathcal{O}}(\sqrt{T})$ and the total number of price changes is $\mathcal{O}(\log\log T)$. The upper bound on the rate of regret of the *SLPE* policy matches the lower bound up-to a constant and logarithmic factor. Furthermore, the rate of price changes is the best known price change guarantee for a class of non-parametric demand functions. Hence, our proposed policy provably achieves a near optimal regret rate (upto a constant and logarithmic factor) while incurring very limited price changes.

- *Proof technique and results:* Our proof technique leverages Lipschitz continuity, local linearity and the two-point bandit feedback of demand to reduce the cumulative number of price changes. While Lipschitz continuity and local linearity is used to control the gradient estimation error due to finite differences, two-point bandit feedback is important for controlling the estimation error due to stochastic demand realizations. We also extend all the results to a relaxed version of the two-point bandit feedback assumption, to show

that our results are robust relative to the assumptions.

- *Strong numerical performance:* We perform extensive numerical experiments to investigate the empirical performance of the proposed policy. First, we present a practical approach to elicit demand observations with near-identical stochastic error. Then, we compare our policy to benchmark pricing policies and show that the *SLPE* policy considerably outperforms the benchmark algorithms in terms of regret as well as in terms of the total number of price changes. The proposed policy uses 80% less price changes than the best performing benchmark policy. This leads us to conclude that our proposed method exhibits strong numerical performance.

4.2 Model and Performance Metrics

In this section, we formulate the dynamic pricing and learning problem and formalize the notion of regret, limited price experimentation and limited price changes.

4.2.1 Model

We consider the pricing problem of a retailer offering a single new product with unlimited inventory. The retailer is allowed to choose prices $p_1, p_2, ...p_T$ such that $p_i \in [p^L, p^U]$, $\forall\ i = 1, .., T$. Each price results in a realized demand $Y_1, Y_2, ...$ generated according to the following demand specification with additive noise (similar to Agarwal et al. 2011, den Boer and Zwart 2013 etc.):

$$Y_t = d(p_t) + \varepsilon_t, \quad \forall t = 1, 2, ..., \tag{4.1}$$

where $d : \mathbb{R}^+ \in [p^L, p^U] \to \mathbb{R}^+$ is the unknown fixed demand. We let d be any non-parametric demand function which is non-increasing in price and smooth (See §4.2.3). Similarly, $\{\epsilon_t$, t=1,2,...$\}$ is the σ-subgaussian noise term. Given any price $p \in [p^L, p^U]$, the retailer's expected single period revenue function is given by:

$$r(p) := pd(p), \tag{4.2}$$

where the expectation is taken over ε_t, the σ-subgaussian noise in the demand realization. Then, the revenue maximizing price, p^*, is defined as

$$p^* := \arg\max\{r(p) : p \in [p^L, p^U]\}. \tag{4.3}$$

The retailer's objective is to maximize revenue. Indeed, if demand function d was known a-priori, the retailer would always charge p^*, the revenue maximizing price. Nevertheless, function d is unknown and the retailer has to *learn* the demand function while concurrently *earning* revenue.

Feasible Pricing Policies: We will restrict ourselves to the family of non anticipating policies Π. A non anticipating policy, $\pi \in \Pi : \pi = \{\pi_t\}$, is a sequence of random functions $\pi_t : \mathbb{R}^{2t} \to [p^L, p^U]$, which indicate what price to charge at time t, such that π_t depends only on demand observations collected until time t. Thus, if we let $H_t = (p_1, Y_1, p_2, Y_2, ...p_{t-1}, Y_{t-1})$ denote the vector of the history of prices and corresponding demand realizations until time t and $\mathcal{F}_t$ denote the σ-field generated by H_t, then π_{t+1} is $\mathcal{F}_t$- measurable.

4.2.2 Performance Metrics

We use two different performance metrics in order to compare feasible pricing policies. While cumulative regret measures the revenue gap of a policy from the clairvoyant's optimal policy, the LPC metric measures the number of price changes for any policy π. We discuss each of them next.

Cumulative Regret:

$\mathcal{R}^\pi(T)$, for any feasible pricing policy π, is defined as the expected cumulative revenue loss incurred until time T, when using policy π instead of the revenue optimal price, p^*. That is,

$$\mathcal{R}^\pi(T) = \sum_{t=1}^{T} \left(r(p^*) - r(p_t^\pi)\right). \tag{4.4}$$

The cumulative regret compares any policy π with a clairvoyant who has full knowledge of the demand and hence chooses the optimal price, p^*. The classical pricing and learning literature minimizes (4.4). Nevertheless, as noted before, practical pricing policies additionally aim to make only a few price changes chosen over a very small set of prices.

Limited Price Changes (Broder (2011)):

Limited Price Change, $\text{LPC}^{\pi}(T)$, for any feasible pricing policy π, measures the total number of price changes made by policy π, until time T. That is,

$$\text{LPC}^{\pi}(T) = 1 + |\{2 \leq t \leq T : p_t^{\pi} \neq p_{t-1}^{\pi}\}| \,. \tag{4.5}$$

High $LPC^{\pi}(T)$ would imply potentially high price change costs and vice versa. As before, since the clairvoyant's optimal policy is a constant pricing policy, the optimal price change metric is also 1.

Our goal is to find a policy π, that performs well on both the metrics. Policies with high LPC could result in high price change costs that could out-weigh the benefits of learning the optimal price. Furthermore, high LPC could also lead to customer dissatisfaction (PK Kannan 2001). Finally, high cumulative regret would imply lost revenue potential due to sub-optimal learning. Nevertheless, allowing for unlimited price changes, with a large price menu aids in learning the unknown demand. Hence optimally balancing both the metrics can be challenging.

Notation: Throughout the chapter, upper case letters refer to random variables, and lower case letters refer to deterministic variables. For any random variable, X, $\bar{X}_n$ refers to the average over n i.i.d. realizations of the X. Similarly, $X_{1,..,n}$ denotes a vector of n independent realizations of any random variable X. We will suppress n in the average notation wherever it is self-explanatory for ease of exposition. For any vector $\mathbf{V} \in \mathbb{R}^k$, V^i denotes the i^{th} entry of the vector $\mathbf{V}$. Furthermore, we also normalize the demand so that $p^L = 0$ and $p^U = 1$. For any twice differentiable real-valued function *f(x)*, $f'(x) = \frac{d}{dx}(f(x))$ denotes the derivative of f with respect to x. Similarly, $f''(x) = \frac{d}{dx}\left(\frac{d}{dx}(f(x))\right)$ denotes the second derivative of f with respect to x. More generally, for a k-differentiable real-valued function, $f^{(i)}(x), \forall i = 1, .., k$ denotes the i^{th}-derivative of of f with respect to x.

4.2.3 Assumptions

Preliminaries

Definition 4.2.1 (Lipschitz Continuity)**.** A real valued function $f(x) : \mathbb{R} \to \mathbb{R}$ is ψ-Lipschitz continuous if for any x_1, x_2 in the domain of f,

$$|f(x_1) - f(x_2)| \leq \psi |x_1 - x_2| \,,$$

for some constant $\psi > 0$.

Assumption 4.2.2 (Demand gradient bounded away from 0)**.** Let $p^* > 0$ be the revenue maximizing price and p be any other feasible price. Then, $\exists \kappa > 0$ s.t.

$$d(p^*) - \kappa(p - p^*)^2 \leq -d'(p)p^* \,,$$

Assumption 4.2.2 bounds the gradient of the demand to be away from 0. Assumption 4.2.2 can be used to relate the gradient of the demand at any price with the gradient of the demand at the optimal price. This statement is made more precise in Lemma 4.2.3.

Lemma 4.2.3. Consider a demand function d that satisfies Assumption 4.2.2. Then, $\forall p \in [0, 1]$, $\exists \kappa > 0$ s.t.

$$p^*(d'(p) - d'(p^*)) \leq \kappa(p - p^*)^2 \,.$$

Proof. See Appendix C.2. □ □

Lemma 4.2.3 shows that for demand functions that satisfy Assumption 4.2.2, the demand gradient at any price is close to the demand gradient at the unknown optimal price. Moreover the gradient at the optimal price (p^*) can be expressed as a function of the gradient at any price (p) plus an extra error factor (quadratic in $(p - p^*)^2$).

Having given some interpretation to Assumption 4.2.2, next we establish some general properties of the demand function that satisfy Assumption 4.2.2. In particular, our assumptions rely on the Lipschitz-continuity of the gradient of the unknown demand and the corresponding revenue function.

Lemma 4.2.4. Let the double derivative of the revenue function r'' and the derivative of demand d' be Ψ and $\bar{\Psi}$ Lipschitz continuous and the demand function d be thrice differentiable.

Assume also that $d''(p^*) = 0$ and let $K_1 = \max_{i\leq 3, p\in[0,1]} |d^{(i)}(p)|$. Then, d satisfies the condition of Assumption 4.2.2. That is, the following holds for $\kappa = \Psi + \bar{\Psi} + 2K_1$:

$$p^*(d'(p) - d'(p^*)) \leq \kappa(p - p^*)^2 \,.$$

Proof. See Appendix C.2. □ □

Lemma 2 shows that along with continuity of demand and revenue functions, if the second derivative of the demand is 0 around the optimal price, implying that the demand function is locally linear around the optimal price (see Figure 4.1), then the demand function satisfies Assumption 4.2.2.

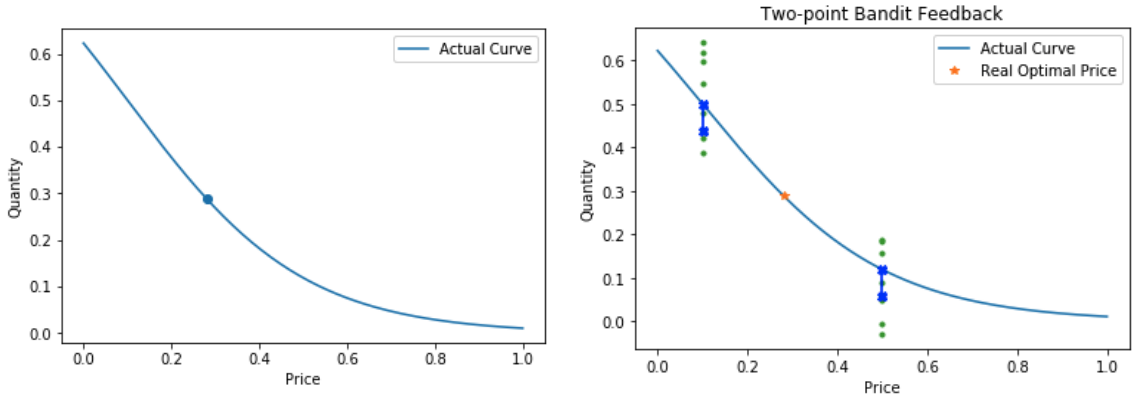

Figure 4.1: On the left figure, we plot the demand function that satisfies the smoothness assumptions of Lemma 4.2.4. The demand function is a modified Logit function since the demand gradient around the optimal price is 0. On the right figure, stochastic demand observations at two arbitrary chosen prices. Notice that one of the demand observations at each price is displaced by the same amount; hence it satisfies the two-point bandit feedback assumption. See §4.4.1 for details on how to relax this assumption.

Assumption 4.2.5 (Two Point Bandit Feedback (Agarwal and Dekel 2010, Nesterov 2011))**.** Let $p_1, p_2 \in [0, 1]$ be any two prices charged by the retailer. Also let $D_{1,..,n}(p_1)$ and $D_{1,..,n}(p_2)$ denote the vector of n random demand realizations each at p_1 and p_2 respectively given by

$$D_i(p_1) = d(p_1) + \epsilon_i^1 \quad \& \quad D_j(p_2) = d(p_2) + \epsilon_j^2, \ \forall i = 1, .., n; j = 1, .., n \,,$$

where ϵ_j^i are the σ-subgaussian noise terms. Then, $\forall n, \exists i^* \leq n$ and $j^* \leq n$ such that $\epsilon_{i^*}^1 = \epsilon_{j^*}^2$. Furthermore, i^* and j^* can be estimated from data.

Assumption 2 posits that when prices are changed, the error in the demand realization among at least one of the n demand realizations is identical (Figure 4.1). This is referred to

as the two-point bandit feedback assumption and becomes important for estimating the error in the gradient of the unknown demand curve at various prices. In particular, since demand realizations are stochastic, estimating derivatives with zero-order demand information could lead to merely subtracting noise. As discussed before, a similar assumption has also been used in recent studies in bandit convex optimization, (see, for example, Agarwal and Dekel (2010), Nesterov (2011), Duchi et al. (2015) and the references there in).

In what follows we present the algorithm and the analysis under Assumption 4.2.5. Nevertheless, to show that the result is robust to the assumption, in §4.4.1 we consider various relaxations of the assumption. In particular, we consider the case when the error at the two demand observations i^* and j^* at the two prices is not the same. We show that all the results continue to hold when this difference between the error terms is bounded and decreases as the number of observations at each price increases (n). Furthermore, we also consider the case when not all prices in the feasible price range satisfy this assumption. Instead, we show that the results continue to hold even when the assumption is satisfied when considering demand observations at prices only in a closed neighborhood around the optimal price.

Finally, we also discuss a practical strategy to compute demand pairs that satisfy Assumption 4.2.5 from observed demand data. In practice, demand for products can have cyclic shocks. For example in offline retail, if customers follow a fixed routine to shop for products over a week, then the demand shocks on any given day of the week could be similar. That is, $\epsilon_{Monday}(p_1) \approx \epsilon_{Monday}(p_2)$. Hence demand data for two different prices on a Monday would satisfy the conditions of Assumption 4.2.5.

We also assume that the unknown demand function d is continuous and w-differentiable for some $w > 0$. Here, w-differentiablity implies that the w-th derivative of the function exists. Finally, the revenue function r has a unique maximizer that is strictly positive. Note that we do not assume that the revenue function is concave since many common demand functions such as Logit demand functions do not lead to concave revenue functions (Figure 4.2).

4.3 Proposed Algorithm

4.3.1 Preliminaries

Linear Interpolations: Let $\mathbf{P} \in \mathbb{R}^k$ and $\mathbf{G} \in \mathbb{R}^{k-1}$ be k dimensional vectors of demand and price points and an estimate of the change in demand at these prices, respectively. Each

component of $\mathbf{P}$ is a tuple (p_i, q_i) that contains the price and estimated demand at that price. For example, given n independent realizations of demand at a price p_i, q_i can be the average of the n demand observations. Assume that $\mathbf{P}$ are arranged in an increasing order of its pricing component. That is, $p_1 < p_2, ..., < p_k$. Then, the linear interpolated approximation of the demand, $\hat{d} : \mathbb{R} \to \mathbb{R}$ is defined as:

$$\hat{d}(\boldsymbol{P}, \boldsymbol{G}, p) = \begin{cases} q_2 + G_1 \left(p - p_2\right), \forall p \leq p_2 \,, \\ q_i + G_i \left(p - p_i\right), \forall p \in [p_i, p_{i+1}], \;\; i = 2, .., k-1 \,. \end{cases}$$

Similarly, $\hat{r} : \mathbb{R} \to \mathbb{R}$, the revenue approximation from such a linear interpolation, is given by,

$$\hat{r}(\boldsymbol{P}, \boldsymbol{Q}, p) = p \cdot \hat{d}(\boldsymbol{P}, \boldsymbol{Q}, p) \,.$$

Finally, let $\zeta \subset \mathbb{R}$, be a subset of feasible prices over the domain of $\hat{d}$. Then, $\hat{p}$, the approximated optimal price from the interpolated revenue approximation is given by:

$$\hat{p}\left(\boldsymbol{P}, \boldsymbol{Q}, \zeta\right) = \underset{p \in \zeta}{\arg\max} \;\; \hat{r}(\boldsymbol{P}, \boldsymbol{Q}, p) \,. \tag{4.6}$$

Problem (4.6) involves solving at most $k-1$ concave and quadratic revenue maximization problems that relate to the $k-1$ linear demand pieces. Nevertheless, each of these problems has a closed form solution. Hence, the overall complexity of the problem is linear in terms of k, the number of linear pieces.

In Figure 4.2, we plot such approximations of the unknown demand curve for a fixed $\mathbf{P}$ and $\mathbf{G}$. Note that the construction of these approximations are fairly simple. They are obtained from linearly interpolating between the price and quantity observations.

4.3.2 Pricing Algorithm

The Stochastic Limited Price Experimentation (*SLPE*) algorithm (Algorithm 4) takes as an input three parameters: T, ρ and μ. While T is the total length of the time horizon, ρ and μ are tuning parameters that govern the dispersion amongst experimentation prices as well as the amount of demand observations allowed at each of these prices. High value of μ ensures that the selected prices in each *round* are well dispersed in the feasible pricing range. Similarly, a high ρ leads to a higher number of sampled demand points at each of the selected prices. In §4.4, we

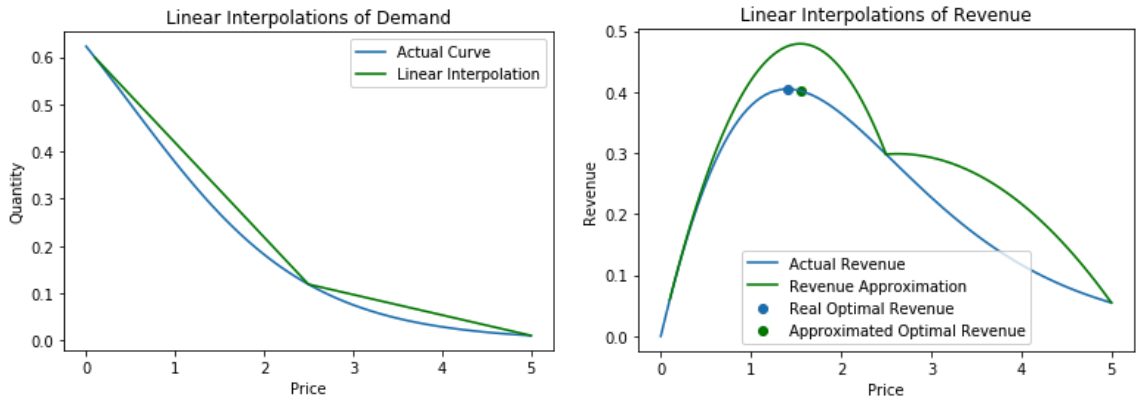

Figure 4.2: Linearly interpolated demand and the corresponding revenue approximation. In this case the optimal price of the approximation is very close to the actual optimal price.

Algorithm 4 SLPE(T,μ,ρ)

Let $p_1^L = 0$, $p_1^H = 1$, $i = 1$, $t = 0$, $\Delta_1 = .5$, $\mathbf{P} = \{\}$, $\mathbf{G} = \{\}$ and $i = 1$.
while $t \leq T$ **do**

Let $n_i = 2\rho^4 \frac{\log(T)}{\Delta_i^4}$ and $t = t + 3n_i$.

Price for n_i periods each at p_i^L, $p_i^M = \frac{p_i^L + p_i^H}{2}$ and p_i^H, respectively.

Let $\mathbf{P} = \{(p_i^L, \bar{D}_{n_i}(p_i^L)), (p_i^M, \bar{D}_{n_i}(p_i^M)), (p_i^H, \bar{D}_{n_i}(p_i^H))\}$

Let $\mathbf{G} = \left\{ \frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{\Delta_i}, \frac{D_{H^*}(p_i^H) - D_{M_2^*}(p_i^M)}{\Delta_i} \right\}$ (pairs satisfying Assumption 4.2.5).

Optimize over piecewise-linear demand estimate with $\mathbf{P}$ and $\mathbf{G}$ (see §4.3.1) to get $\tilde{p}_i^*$.

Let $p_{i+1}^L = \tilde{p}_i^* - \mu\Delta_i^2$, $p_{i+1}^H = \tilde{p}_i^* + \mu\Delta_i^2$, $\Delta_{i+1} = \mu\Delta_i^2$ and i=i+1.

discuss how the choice of μ and ρ govern the theoretical guarantees of the algorithm and in §4.5, we discuss practical choices of μ and ρ that show substantially improved numerical performance over benchmark algorithms. Finally, while we assume that T is known in advance, we note that the theoretical analysis of the *SLPE* algorithm (see §4.4) is independent of knowing the specific value of T. The analysis can be extended using the well known "Doubling Trick" frequently used in the analysis of online algorithms (see, for example, Besson and Kaufmann 2018).

Initially, the algorithm starts by experimenting at the end points, as well as the mid point of the initial price range (prices 0, 0.5 and 1 respectively). The subsequent price selection is determined by optimizing over the piecewise linear approximation of the demand function created by the demand observations obtained in the previous round. In each subsequent round, we select the optimal approximated price and two prices around that optimal approximated price. Prices are selected so that in each round, the overall price range around the approximated

optimal price contains the real unknown optimal price with high probability (see §4.4).

Interestingly, while we use the average of the observed demand, $\bar{D}$ to estimate the unknown demand at any price (see **P** in Algorithm 4), the gradient is estimated using the central-difference estimator from demand observations satisfying two-point bandit feedback, $D_{M_1^*}$ and D_{L^*} (see **G** in Algorithm 4).

While Algorithm 4 assumes that at least one pair of demand observations that satisfy Assumption 4.2.5 are known, in §4.4.1 we discuss how to estimate such demand pairs that could potentially have the same error. This method can be used as a sub-routine to first find such pairs, and then use the procedure that is proposed in Algorithm 4. In fact, the numerical algorithm in §4.5 uses this sub-routine and still outperforms other benchmarks, showing practical applicability of the proposed method.

Intuition:

To develop some intuition on the proposed pricing policy, we embed the pricing problem within the framework of bandit convex optimization (Kleinberg 2005). In this setting, the quantity of interest is a certain scalar p^* that maximizes a certain well behaved function (see §4.2.3), d, which is unknown. While d is unknown, noisy observations of d at selected decision points p_i, $(D(p_i))$, are available; the objective is to select various decision points, p_i, such that they converge to the optimal point, p^*. Over the years, various algorithms have been proposed to solve this problem. In particular, as we previously mentioned, one popular approach is to use stochastic bisection search methods (Agarwal et al. 2011, Jasin et al. 2015) to find the optimal price. These algorithms depend on an underlying concavity assumption on the revenue function. Unfortunately, this concavity property is often not satisfied (see Figure 4.2). Similarly, another approach is to approximate the unknown demand function, for example with linear functions (Besbes and Zeevi 2015) that incurs $\tilde{\mathcal{O}}(\sqrt{T})$ with $\mathcal{O}(\log T)$ price changes.

Instead, the SLPE policy uses piecewise linear interpolations for approximating the unknown non-parametric demand function. Gradients are estimated using a pair of demand observations that have identical error (see Assumption 4.2.5). This ensures that not only is the estimate of the demand at any price "good" enough, but also that the estimate of the first-derivative of the demand at any price is also "good" enough. To see this, let $\hat{d}$ be an approximation of the unknown demand function d, and let $\hat{p}^*$ the revenue maximizing price for the approximated unknown demand curve. Also denote by $g(p) = d(p) + d'(p)p$ and $\hat{g}(p) = \hat{d}(p) + \hat{d}'(p)p$, the first

order condition for the revenue maximization problem. Then, $g(p^*) = 0$ and $\hat{g}(\hat{p}^*) = 0$. Notice that while g is unknown, $\hat{g}$ is known. Furthermore,

$$\begin{aligned} \hat{g}(p^*) - g(p^*) &= (\hat{d}(p^*) - d(p^*)) + (\hat{d}'(p^*) - d'(p^*))p^* \\ \implies \hat{g}(p^*) - \hat{g}(\hat{p}^*) &= (\hat{d}(p^*) - d(p^*)) + (\hat{d}'(p^*) - d'(p^*))p^* \end{aligned}$$

Now if, for example, $\hat{g}$ is a linear function of the form $\hat{g}(p) = \alpha + \beta p$, for some $\beta \neq 0$, it follows that

$$|p^* - \hat{p}^*| \leq \frac{1}{\beta} \left(\underbrace{|\hat{d}(p^*) - d(p^*)|}_{\text{A}} + \underbrace{|\hat{d}'(p^*) - d'(p^*)|}_{\text{B}} p^* \right) \tag{4.7}$$

While (A) in (4.7) represents the estimation error of demand at the optimal price, (B) in (4.7) represents the estimation error of the gradient at the optimal price. Hence, to get an upper bound on the estimation error between the estimated and the unknown optimal price, we need to bound the error in estimating the demand at the optimal price, as well as the error in estimating the gradient at the optimal price. Since the optimal price is unknown, SLPE ensures that both these estimation errors are "small" at any selected price. This is accomplished by controlling two sources of error. First, we have to control for the error due to demand misspecification on account of the unknown non-parametric structure of the demand function d. Second, since demand realizations are stochastic, we also incur an estimation error due to the noise in demand observations at selected prices. Notice that both misspecification and demand stochasticity contribute to error in estimating demand and the gradient of demand at any given price. n_i in each round is selected so that the error due to stochastic demand realizations is of $\mathcal{O}(\Delta_i^2)$ for any price. Similarly, the gradient estimation error is also of $\mathcal{O}(\Delta_i^2)$ since we use the finite difference of demand observations with identical error. To contrast this with existing approaches that use average demand at any two prices as an estimate of the gradient, we have that for any price p,

$$\hat{d}'(p) = \frac{\bar{D}(p) - \bar{D}(p-\Delta)}{\Delta} = \frac{d(p) - d(p-\Delta)}{\Delta} + \frac{1}{n\Delta} \left(\sum_{i=1}^{n} \epsilon_i - \sum_{j=1}^{n} \tilde{\epsilon}_j \right). \tag{4.8}$$

Here ϵ and $\tilde{\epsilon}$ are error realizations in demand (see §4.2.1), n denotes the total demand realizations at any of these price points and $\bar{D}(p)$ denotes the average of demand observations at price p. The first part on the RHS of (4.8) denotes the error due to using finite differences in

demand observations to estimate the demand gradient. The second part is error incurred due to stochastic demand observations. Importantly, this stochastic error is not present when gradients are estimated in the SLPE policy. Hence, when n is tuned appropriately, the error in the average based estimator (4.8) decays at the rate of $\mathcal{O}(\Delta)$, but the gradient estimated in the SLPE policy decays at the faster rate of $\mathcal{O}(\Delta^2)$. This faster convergence of the gradient estimate to the true gradient leads to improved estimation of the optimal price. In every round, we can estimate a high probability region of size $\mathcal{O}(\Delta^2)$ around the approximated optimal price that contains the optimal price. We make these statements precise in what follows.

4.4 Analytical Results

The main result of this section is a $\tilde{\mathcal{O}}(\sqrt{T})$ bound on the regret of the SLPE policy that we present in Theorem 4.4.1.

Theorem 4.4.1. Let the unknown real demand function $d(p)$ satisfy Assumptions (1) and (2). Also assume that $|d'(p^*)| \geq \left(\frac{K_1}{24} + \frac{\kappa}{4}\right)\frac{1}{4} + \frac{c}{2}$, for some positive constant c and κ as defined in Assumption 4.2.2 and $K_1 = \max_{i\leq w, p\in[0,1]} |d^{(i)}(p)|$, where $d(p)$ is w-differentiable. Then if $2 > \mu \geq \left(\frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + 4\kappa\right)$, the regret of the SLPE pricing policy is

$$\mathcal{R}^{SLPE(T)}(T) \leq \left(C_1 + C_2 \log\left(\log\left(\frac{T}{2\mu\rho^4 \log(T)}\right)\right)\right)\sqrt{T}\log(T) = \mathcal{O}\left(\log(T)\log(\log(T))\sqrt{T}\right),$$

where C_1 and C_2 are constants independent of T.

The proof follows through a series of lemmas that we discuss next.

First note that the SLPE algorithm is a round based policy. Hence, we start by showing that with high probability, in any round i, the optimal price is contained within the upper and lower bound on optimal price (p^H and p^L) for that round.

In Lemma 4.4.2, we start by showing that the selection of number of samples in each round i of the SLPE algorithm (n_i) ensures that the finite difference estimator of the gradient of demand has a small approximation error of $\mathcal{O}(\Delta_i^2)$, where $\Delta_i := p_i^H - p_i^M = p_i^M - p_i^L$ is the size of the interpolation of round i. The result is based on Taylor series expansion that leads to an upper bound on the error due to linear interpolations, and the Lipschitz continuity of demand and revenue functions.

Lemma 4.4.2. Consider the three experimental prices p_i^L, p_i^M and p_i^H of prices experimented in round i of Algorithm (4). Then,

$$\left| \frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right) \right| \leq \frac{1}{2} K_1 \frac{\Delta_i^2}{12}\,, \tag{4.9}$$

and

$$\left| \frac{D_{H^*}(p_i^H) - D_{M_2^*}(p_i^M)}{\Delta_i} - d'\left(\frac{p_i^M + p_i^H}{2}\right) \right| \leq \frac{1}{2} K_1 \frac{\Delta_i^2}{12}\,, \tag{4.10}$$

where $K_1 = \max_{i \leq w, p \in [0,1]} |d^{(i)}(p)|$, and $d^{(i)}(p)$ denotes the i^{th} derivative of demand at any price p. Furthermore, $(D_{L^*}(p_i^L), D_{M_1^*}(p_i^M))$ and $(D_{M_2^*}(M_i), D_{H^*}(H_i))$ are pair of demand realizations that satisfy Assumption (4.2.5).

Proof. See Appendix C.2. □

We combine the result of Lemma 4.4.2 to show in Lemma 4.4.3 that the "small" approximation error in the derivatives also leads to a "small" gap between the approximated optimal price and the unknown optimal price. The proof relies on using the first order condition that the optimal price satisfies and comparing it with the approximated first order condition that can be estimated using the piecewise linear approximation. We show that the optimal price and the approximated price for all rounds of the algorithm are contained in a region of size $\mathcal{O}(\Delta_i^2)$ with high probability.

Lemma 4.4.3. Consider the SLPE pricing policy of Algorithm 4 and let the unknown demand function follow all assumptions as in Theorem 4.4.1. Then for any round i,

$$|p^* - \tilde{p}_i^*| \leq M \Delta_i^2\,,$$

for $M = \left(\frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + 4\kappa\right)$ with probability at least $1 - \frac{1}{T^2}$, where p^* is the real unknown optimal price and $\tilde{p}_i^*$ is the approximated optimal price from the piecewise linear interpolated demand curve of round i.

Proof. The proof follows in two main steps. In the first step, we use the first order condition to relate the error in the unknown optimal and the approximated optimal price to the estimation error in the demand and the gradient. Then in the second step, we bound the estimation error respectively.

Step 1: Relating the error in the approximated optimal price with the estimation error:

Let $g(p) := r'(p)$ be the first order equation of the unknown revenue function. Then, by the optimality of p^*, we have that $g(p^*) = 0$. Similarly, we have that $\tilde{p}^*$ is the estimated optimal price from the piecewise linear demand curve constructed using demand observations at p^L, p^M and p^H. In particular, recall that

$$\tilde{p}^* = \arg\max_{p\in[p^L,p^H]} p d^{est}(p)\,,$$

where we let

$$d^{est}(p) := \begin{cases} \bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(p - p^M\right), \forall p \le p^M\,, \\ \bar{D}(p^M) + \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}\left(p - p^M\right), \forall p > p^M\,. \end{cases}$$

Recall by definition that $\tilde{p}^*$ is the approximated optimal price that is revenue maximizing for the approximated demand $d^{est}(p)$. Since $d^{est}(p)$ is a piecewise-linear function we have two cases to analyze: (i) $\tilde{p}^* \le p^M$ or (ii) $\tilde{p}^* > p^M$. Assume without loss of generality that $\tilde{p}^* \le p^M$. Since $\tilde{p}^*$ is the revenue maximizing price, it is a solution to the following (approximate) first order condition:

$$d^{est}(p) + d'^{est}(p)p = 0\,.$$

Hence $d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = 0$. In order to compare the approximated optimal price with the real optimal price, we evaluate the optimal price at the approximate first order condition.Note though that the approximate first order condition is also a piecewise function. Hence, we have to analyze two cases: (i) if $p^* < p^M$ or (ii) $p^* \ge p^M$.

Case (i) $p^* < p^M$: Consider the approximate first order condition evaluated at p^*,

$$d^{est}(p^*) + d'^{est}(p^*)p^* = \bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(p^* - p^M\right) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}p^*\,. \tag{4.11}$$

Similarly,

$$d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = \bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(\tilde{p}^* - p^M\right) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\tilde{p}^* = 0\,, \tag{4.12}$$

where the last equality follows from the optimality of $\tilde{p}^*$ for the approximate demand. Hence subtracting (4.12) from (4.11), we have that:

$$d^{est}(p^*) + d'^{est}(p^*)p^* - (d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^*) = 2\left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)(p^* - \tilde{p}^*)\,.$$

Also note that $g(p^*) = 0$. Hence, $d(p^*) + d'(p^*)p^* = 0$. Furthermore,

$$\begin{aligned} d^{est}(p^*) + d'^{est}(p^*)p^* &= d^{est}(p^*) + d'^{est}(p^*)p^* - (d(p^*) + d'(p^*)p^*) \\ &= (d^{est}(p^*) - d(p^*)) + (d'^{est}(p^*) - d'(p^*))p^*\,. \end{aligned}$$

Hence, combining the two equalities above, we get that

$$\begin{aligned} &2\left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)(p^* - \tilde{p}^*) = (d^{est}(p^*) - d(p^*)) + (d'^{est}(p^*) - d'(p^*))p^* \\ &2\Big|\left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)\Big||p^* - \tilde{p}^*| \le |d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^* \qquad (4.13) \\ \implies &|p^* - \tilde{p}^*| \le \frac{1}{2|d'^{est}(p^*)|}\left(\underbrace{|d^{est}(p^*) - d(p^*)|}_{\text{A}} + \underbrace{|d'^{est}(p^*) - d'(p^*)|p^*}_{\text{B}}\right), \end{aligned}$$

where $d'^{est}(p^*) := \left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)$. To bound the estimation error in the optimal price, we need to bound terms (A) and (B). (A) denotes the estimation error in demand at the optimal price and (B) denotes the estimation error in the gradient. In what follows, we will bound both these errors.

Step 2: Bounding the estimation error in the demand and the gradient:

We proceed by independently bounding (A) and (B) from (4.13).

Bounding $|d^{est}(p^*) - d(p^*)|$: By definition, we have that

$$\begin{aligned} &|d^{est}(p^*) - d(p^*)| = \left|\bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(p^* - p^M\right) - d(p^*)\right| \\ &= \left|\bar{D}(p^M) \pm d(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(p^* - p^M\right) - d(p^*) \pm \left(\frac{d(p^M) - d(p^L)}{p^M - p^L}(p^* - p^L)\right)\right| \\ &\le \left|\bar{D}(p^M) - d(p^M) + \frac{D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)}{p^L - p^M}\left(p^* - p^M\right)\right| + \\ &\left| - d(p^*) + d(p^M) + \frac{d(p^M) - d(p^L)}{p^M - p^L}(p^* - p^M)\right|. \end{aligned}$$

Now let $p^* = \lambda p^L + (1-\lambda)p^M$, for some $\lambda \in [0,1]$. Then,

$$\left|\bar{D}(p^M) - d(p^M) + \frac{D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)}{p^L - p^M}\left(p^* - p^M\right)\right| \leq$$
$$|\bar{D}(p^M) - d(p^M)| + \lambda|(D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L))|\,.$$

But by Assumption 4.2.5, we have that

$$(D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)) = d(p^M) + \epsilon^* - d(p^M) + d(p^L) - d(p^L) - \epsilon^* = 0\,.$$

Hence, w.p at least $1 - \frac{1}{T^2}$

$$\left|\bar{D}(p^M) - d(p^M) + \frac{D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)}{p^L - p^M}\left(p^* - p^M\right)\right| \leq |\bar{D}(p^M) - d(p^M)|$$
$$\leq \frac{\Delta^2}{\rho^2}\,.$$

The last inequality follows through Hoeffding's inequality for sub-gaussian random variables.

Next, to bound $\left|-d(p^*)+d(p^M)+\frac{d(p^M)-d(p^L)}{p^M-p^L}(p^*-p^M)\right|$, we can apply the linear interpolation error bound (see Chapter 6 of Süli and Mayers (2003)) and get that

$$\left|d(p^L) + \frac{d(p^L) - d(p^M)}{p^L - p^M}(p^* - p^L) - d(p^*)\right| \leq \frac{K_1}{8}\Delta^2\,,$$

where recall that $K_1 = \max_{p\in[0,1], i\leq w} |d^i(p)|$. Hence,

$$|d^{est}(p^*) - d(p^*)| \leq \left(\frac{1}{\rho^2} + \frac{K_1}{8}\right)\Delta_i^2\,. \tag{4.14}$$

Now we focus on bounding term (B) of (4.13), that is $|d'^{est}(p^*) - d'(p^*)|$.

Bounding $|d'^{est}(p^*) - d'(p^*)|$: First recall, by definition that $d'^{est}(p^*) = \left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)$. Hence,

$$|d'^{est}(p^*) - d'(p^*)| = \left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} + d'\left(\frac{p^M + p^L}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right|$$
$$\leq \left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} - d'\left(\frac{p^M + p^L}{2}\right)\right| + \left|d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right|. \tag{4.15}$$

Lemma 4.4.2 implies that

$$\left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} - d'\left(\frac{p^M + p^L}{2}\right)\right| \leq \frac{1}{2}K_1\frac{\Delta_i^2}{12}\,.$$

Similarly, to bound $\left|d'\left(\frac{p^M+p^L}{2}\right) - d'(p^*)\right|$, we use Assumption 4.2.2 and get that

$$\left|d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right| \leq \kappa\left(p^* - \frac{p^M + p^L}{2}\right)^2 \leq \frac{\kappa\Delta_i^2}{4}\,,$$

where the last inequality follows because $p^* \leq p^M$. Hence, combining the above two results and using (4.15), we have that

$$|d'^{est}(p^*) - d'(p^*)| \leq \left(\frac{K_1}{24} + \frac{\kappa}{4}\right)\Delta_i^2\,, \tag{4.16}$$

Hence using (4.14) and (4.16), it follows that

$$\begin{aligned}|p^* - \tilde{p}^*| &\leq \frac{1}{2|d'^{est}(p^*)|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)p^*|\right)\\ &\leq \frac{1}{2|d'^{est}(p^*)|}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.\end{aligned}$$

Finally, to bound $d'^{est}(p^*)$, note that

$$|d'^{est}(p^*)| \geq |d'(p^*)| - |d'^{est}(p^*) - d'(p^*)| \geq |d'(p^*)| - \left(\frac{K_1}{24} + \frac{\kappa}{4}\right)\Delta_i^2 \geq |d'(p^*)| - \left(\frac{K_1}{24} + \frac{\kappa}{4}\right)\frac{1}{4} \geq \frac{c}{2}\,,$$

where the last inequality follows from the assumption that the derivative of demand at the optimal price is bounded away from 0. Hence, we get that

$$|p^* - \tilde{p}^*| \leq \frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.$$

So far, we assumed that both p^* and $\tilde{p}^*$ are less than the mid point of the current interpolation. Next, consider case (ii) when $p^* > p^M$ but as before $\tilde{p}^* \leq p^M$. In this case, we have to account for a larger approximation error in the demand and the gradient of demand at the optimal price.

Case (ii) $p^* > p^M$: Consider the first order condition on the approximated demand evaluated

at p^*,

$$\begin{aligned} d^{est}(p^*) + d'^{est}(p^*)p^* =& \bar{D}(p^M) + \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}\left(p^* - p^M\right) \\ &+ \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}p^M - p^L p^* \,. \end{aligned} \tag{4.17}$$

Similarly, evaluating the first order equation at the optimal price calculated using the approximated demand, we get that

$$d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = \bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(\tilde{p}^* - p^M\right) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\tilde{p}^* = 0\,, \tag{4.18}$$

where the difference is due to the fact that $p^* > p^M$ but $\tilde{p}^* \leq p^M$. Subtracting (4.18) from (4.17), and letting $m_1 = \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}$ and $m_2 = \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}$, for ease of notation, we get that

$$\begin{aligned} &d^{est}(p^*) + d'^{est}(p^*)p^* - \left(d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^*\right) = m_1(p^* - p^M) + m_1 p^* - m_2(\tilde{p^*} - p^M) - m_2 p^* \\ &= m_1(p^* - p^M + \tilde{p}^* - \tilde{p}^*) - m_2(\tilde{p}^* - p^M) + m_1(p^* - \tilde{p}^*) - m_2\tilde{p}^* \\ &= 2m_1(p^* - \tilde{p}^*) + m_1(\tilde{p}^* - p^M) - m_2(\tilde{p}^* - p^M) + (m_1 - m_2)\tilde{p}^* \\ &= 2m_1(p^* - \tilde{p}^*) + (m_1 - m_2)(2\tilde{p}^* - p^M)\,. \end{aligned}$$

We follow the same analysis as before and arrive at the following:

$$|p^* - \tilde{p}^*| \leq \underbrace{\frac{1}{2|m_1|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^*\right)}_{\text{A}} + \underbrace{|m_1 - m_2|\left(2\tilde{p}^* - p^M\right)}_{\text{B}}\,. \tag{4.19}$$

Notice that (A) in the equation above is the same as before (case (i) when $p^* \leq p^M$). Hence, an identical analysis yields that

$$\frac{1}{2|m_1|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^*\right) \leq \frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.$$

Focusing on (B), we get that

$$\begin{aligned}
|m_1 - m_2| &= \left| m_1 - m_2 + d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^H}{2}\right) \right| \\
&= \left| m_1 - m_2 + d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^H}{2}\right) + d'\left(\frac{p^M + p^L}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right) \right| \\
&\leq \left| m_1 - d'\left(\frac{p^M + p^H}{2}\right) \right| + \left| d'\left(\frac{p^M + p^L}{2}\right) - m_2 \right| + \left| d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right) \right| \\
&\leq \left(\frac{K_1}{12} + 4\kappa\right) \Delta_i^2 ,
\end{aligned}$$

where the last inequality follows by Lemma 4.4.2, Assumption 4.2.2 and $p^* d'(p) \leq d(p^*), \forall p \in [0,1]$. Hence, we have that

$$|p^* - \tilde{p}^*| \leq \left(\frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + 4\kappa\right) \Delta_i^2 ,$$

hence, letting $M = \left(\frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + 4\kappa\right)$ proves the final result.

□

We are finally in the position to prove that the regret of the *SLPE* policy is near optimal and it scales sublinearly with T (that is, $\tilde{\mathcal{O}}(\sqrt{T})$). The proof uses Lemma 4.4.2 and Lemma 4.4.3 to first show that the regret is bounded as we move from one round to the next. In particular, we use the Mean Value Theorem to first relate regret from any price to the distance of this price from the optimal price. Then Lemma 4.4.3 is used to bound the distance of prices selected in any round to the optimal price.

Proof. Proof of Theorem 4.4.1: We will show that the regret is bounded in two steps. In Step 1, we will first bound the regret from any round of the policy. Then, in Step 2, we will bound the overall regret of the policy. Our focus will be the high probability events that happen with probability at least $1 - 1/T^2$ since regret from the low probability event over the whole time horizon would be of a constant factor.

Step 1 (Bounding regret loss in any round): In any round $i > 1$, the SLPE policy makes 3 price changes, at p_i^H, p_i^L and at $\tilde{p}_{i-1}^*$. Hence, we need to first estimate the revenue loss due to pricing at these price points.

First note that

$$r(p^*) - r(\tilde{p}_{i-1}^*) \leq K_1 (p^* - \tilde{p}_{i-1}^*)^2 ,$$

where recall that $K_1 = \max_{p\in[0,1],i\leq w} |d^i(p)|$. This follows by a direct application of (i) the Mean Value Theorem and (ii) the boundedness of second derivative of the demand function and the revenue optimality of the unknown optimal price p^*. Next, in order to bound $(p^* - \tilde{p}^*_{i-1})$, notice that by Lemma 4.4.3

$$|p^* - \tilde{p}^*_{i-1}| \leq M\Delta^2_{i-1} \leq \Delta_i\,,$$

which holds by by construction since $\Delta_i = \mu\Delta^2_{i-1}$ and $M \leq \mu$. Hence, we have that

$$r(p^*) - r(\tilde{p}^*_{i-1}) \leq K_1\Delta^2_i\,.$$

Similarly, considering the error from p^H_i,

$$\begin{aligned} r(p^*) - r(p^H_i) \leq K_1(p^* - p^H_i)^2 &= K_1(p^* - p^H_i + \tilde{p}^*_{i-1} - \tilde{p}^*_{i-1})^2 \\ &\leq K_1(|p^* - \tilde{p}^*_{i-1}| + |\tilde{p}^*_{i-1} - p^H_i|)^2 \\ &\leq K_1(\Delta_i + 2\Delta_i)^2 \\ &\leq 9K_1\Delta^2_i\,, \end{aligned}$$

where the last inequality follows because $|p^* - \tilde{p}^*_{i-1}| \leq \Delta_i$ and $|\tilde{p}^*_{i-1} - p^H_i| \leq 2\Delta_i$. An identical analysis yields that

$$r(p^*) - r(p^L_i) \leq 9K_1\Delta^2_i\,.$$

Now recall that at each of these three prices, the number of demand realizations, by design, is $n_i = \frac{2\rho^4\log(T)}{\Delta^4_i}$. Hence, the upper bound on overall regret from this round is

$$\begin{aligned} n_i(r(p^*) - r(p^L_i) + r(p^*) - r(p^H_i) + r(p^*) - r(\tilde{p}^*_{i-1}+)) &\leq n_i\left(19K_1\Delta^2_i\right) \\ &= \frac{2\rho^4\log(T)}{\Delta^4_i}\left(19K_1\Delta^2_i\right) \\ &= \frac{38K_1\rho^4\log(T)}{\Delta^2_i}\,. \end{aligned}$$

But, recall that there are in total T samples. Hence, the samples at any given price cannot exceed the total available samples. That is,

$$n_i = \frac{2\rho^4\log(T)}{\Delta^4_i} \leq T \implies \frac{38K_1\rho^4\log(T)}{\Delta^2_i} \leq 38K_1\rho^2\log(T)\sqrt{T}\,. \tag{4.20}$$

Hence, in each round after the first round, the regret scales with $\tilde{\mathcal{O}}(\sqrt{T})$. Simple algebra also yields that the total regret from the first round is upper bounded by $24K_1\rho^4$. Next, to bound the total regret, we bound the total number of rounds in the SLPE policy.

Step 2: Bounding the total number of rounds in SLPE: For any round $i \leq i_{max}$, recall that

$$n_i = \frac{2\rho^4 \log(T)}{\Delta_i^4},$$

steps. Also note that for any $i > 1$, $\Delta_i = \mu\Delta_{i-1}^2$. Hence, in any round $i \leq i_{max}$,

$$\Delta_i = \mu\Delta_{i-1}^2 = \mu\left(\mu\Delta_{i-2}^2\right)^2 = ... = \mu^{2^{i-1}}\left(\frac{1}{2}\right)^{2^i}.$$

Furthermore, since the total number of demand realizations are upper bounded by T, we also have that

$$\frac{2\rho^4 \log(T)}{\Delta_i^4} \leq T.$$

$$\begin{aligned}
\frac{2\rho^4 \log(T)}{\Delta_i^4} \leq T &\implies \frac{1}{\Delta_i^4} \leq \frac{T}{2\rho^4 \log(T)} \implies \frac{1}{\Delta_i} \leq \left(\frac{T}{2\rho^4 \log(T)}\right)^{\frac{1}{4}} \implies \frac{2^{2^i}}{\mu^{2^i}} \leq \frac{1}{\mu}\left(\frac{T}{2\rho^4 \log(T)}\right)^{\frac{1}{4}} \\
&\implies 2^i \log\left(\frac{2}{\mu}\right) \leq \log\left(\frac{1}{\mu}\left(\frac{T}{2\rho^4 \log(T)}\right)^{\frac{1}{4}}\right) \\
&\implies 2^i \leq \frac{1}{\log\left(\frac{2}{\mu}\right)}\left(\log\left(\frac{T}{2\mu\rho^4 \log(T)}\right)\right),
\end{aligned}$$

where the second to last inequality follows since, $\mu \leq 2$. Furthermore, since the inequality above holds for all i, we have that

$$\begin{aligned}
&\implies i_{max} \leq \log\left(\frac{1}{\log\left(\frac{2}{\mu}\right)}\left(\log\left(\frac{T}{2\mu\rho^4 \log(T)}\right)\right)\right) \\
&\implies i_{max} \leq \log\left(\frac{1}{\log\left(\frac{2}{\mu}\right)}\right) + \log\left(\log\left(\frac{T}{2\mu\rho^4 \log(T)}\right)\right).
\end{aligned}$$

Step 3: Bounding the overall regret: We are now in a position to combine the maximum number of rounds and the regret from each round to get an upper bound on the cumulative

regret of the SLPE policy. First note that

$$\begin{aligned}
\mathcal{R}^{SLPE(T)} &= \sum_{t=1}^{T} r(p^*) - r(p_t^\pi) \\
&= \sum_{i=1}^{i_{max}} n_i \left((r(p^*) - r(\tilde{p}_i^*)) + (r(p^*) - r(p_i^H)) + (r(p^*) - r(p_i^L)) \right) \\
&\leq n_1 K_1 \frac{1}{4} + \sum_{i=2}^{i_{max}} 38 K_1 \rho^2 \log(T) \sqrt{T} \\
&\leq 24 K_1 \rho^4 + 38 K_1 \rho^2 \log(T) \sqrt{T} i_{max} \,,
\end{aligned} \tag{4.21}$$

where the second to last inequality follows because of (4.20). Thus in order to bound the total regret, we need to bound i_{max}. Substituting back in (4.21), we get that

$$\begin{aligned}
\mathcal{R}^{SLPE(T)}(T) &= 24 K_1 \rho^4 + 38 K_1 \rho^2 \log(T) \sqrt{T} i_{max} \\
&= 24 K_1 \rho^4 + 38 K_1 \rho^2 \log(T) \sqrt{T} \left(\log \left(\frac{1}{\log\left(\frac{2}{\mu}\right)} \right) + \log \left(\log \left(\frac{T}{2\mu\rho^4 \log(T)} \right) \right) \right) \\
&\leq \left(C_1 + C_2 \log \left(\log \left(\frac{T}{2\mu\rho^4 \log(T)} \right) \right) \right) \sqrt{T} \log(T) \\
&= \mathcal{O}\left(\sqrt{T} \log(\log(T)) \log(T) \right) \,,
\end{aligned}$$

where $C_1 = \log \left(\frac{1}{\log\left(\frac{2}{\mu}\right)} \right) C_2 + 24 K_1 \rho^4$ and $C_2 = 38 K_1 \rho^2$. This proves the final result. □

Theorem 4.4.1 shows that the regret of the SLPE policy is $\mathcal{O}(\log T \log(\log T) \sqrt{T})$. Keskin and Zeevi (2014) have already shown that the lower bound for the regret of this class of policies is $\mathcal{O}(\sqrt{T})$ (Theorem 1). Hence, the proposed policy is near optimal. As a part of the proof, we also bound the total number of rounds in the policy. Since each round entails a total of 3 price changes, this naturally results in an upper bound on the total number of price changes (see §4.2.2). Furthermore, the above result also shows that the number of price changes are of the order of $\log \log(T)$. Corollary 4.4.4 makes this statement more precise.

Corollary 4.4.4. Let the unknown real demand function $d(p)$ satisfy Assumptions (1) and (2). Also assume that $2 > \mu \geq M$, where M is defined in Lemma 4.4.3. Then the total number of price changes of the SLPE policy,

$$LPC^{SLPE}(T) \leq 3\tilde{C} + 3 \log \left(\log \left(\frac{T}{2\mu\rho^4 \log(T)} \right) \right) \,,$$

where $\tilde{C} = \log\left(\frac{1}{\log\left(\frac{2}{\mu}\right)}\right)$.

Proof. Proof: The proof follows directly from the analysis of Theorem 1. □

We note that Broder (2011) proved a $\mathcal{O}(\log T)$ price change bound in the parametric demand case. Nevertheless, our results do not contradict their lower bound since Assumption 4.2.5 allows us to come up with improved gradient estimates, even when demand observations are stochastic. In-fact, our work bridges the gap between discrete and the continuous armed bandit cases by discovering a class of continuous armed problems that have identical price change guarantee as that of the discrete case.

We have so far shown that the analytical performance of the algorithm is near optimal in terms of regret, and the total number of price changes and price experimentation are also very low. In what follows, we revisit Assumption 4.2.5 that is important in our analysis. We will show that the results follow even when the two-point bandit feedback is relaxed. Furthermore, we also discuss practical implications of the assumption and propose an algorithm for eliciting pair of demand points that poses the two-point feedback property.

4.4.1 Relaxing the Two Point Bandit Feedback Assumption

Previously we saw that two-point bandit feedback plays an important role in reducing the number of price changes from $\mathcal{O}(\log T)$ to $\mathcal{O}(\log\log T)$. In this section, we posit the question of how robust the results are to this assumption. In particular, we relax this assumption and still get the same guarantees as before. Indeed, as mentioned before, without any structure on the error, Broder (2011) have already shown a lower bound on the price changes that is $\mathcal{O}(\log T)$ in a parametric demand setting. Hence, completely removing this assumption is futile if we want to reduce the number of price changes further. Instead, we consider a relaxed version of Assumption 4.2.5. For the sake of completeness, we start by restating Assumption 4.2.5.

Assumption 2: Let $p_1, p_2 \in [0,1]$ be any two prices charged by the retailer. Also let $D_{1,..,n}(p_1)$ and $D_{1,..,n}(p_2)$ denote the vector of n random demand realizations each at p_1 and p_2 respectively given by

$$D_i(p_1) = d(p_1) + \epsilon_i^1 \quad \& \quad D_j(p_2) = d(p_2) + \epsilon_j^2, \ \forall i = 1,..,n; j = 1,..,n\,, \tag{4.22}$$

where ϵ_j^i are the σ-subgaussian noise terms. Then, $\forall n, \exists i^* \leq n$ and $j^* \leq n$ such that $\epsilon_{i^*}^1 = \epsilon_{j^*}^2$.

Furthermore, i^* and j^* can be estimated from data.

In particular, Assumption 4.2.5 imposes the existence of two stochastic demand realizations (at different prices) with the same error realization throughout the price range. We relax this assumption and instead impose the following assumption:

2A *Let $p_1, p_2 \in [p^* - \sigma, p^* + \sigma]$ be any two subsequent prices charged by the retailer in the neighborhood of the unknown optimal price. Also let $D_{1,..,n}(p_1)$ and $D_{1,..,n}(p_2)$ denote n random demand realizations each at p_1 and p_2 respectively given by (4.22). Then, $\forall n, \exists i^* \leq n$ and $j^* \leq n$ such that*

$$|\epsilon^1_{i^*} - \epsilon^2_{j^*}| \leq f(n)\,,$$

where $f(n)$ is a decreasing function of n. Furthermore, the indices i^* and j^* can be estimated from data. Assumption 4.4.1 relaxes Assumption 4.2.5 in several ways. In particular,

1. Assumption 4.4.1 allows the error realizations at the two selected prices to be different from each other, in comparison to being the same as in Assumption 4.2.5. $f(n)$ captures the bound on this difference between the error. In Theorem 4.4.5 we show that when the decay rate of *f(n)* is of $\mathcal{O}(n^{-\delta})$, for $\delta \geq 3/4$, we get the same price change and regret guarantees as before.

2. Assumption 4.4.1 relaxes the existence of such demand pairs that have bounded difference in the error realization from throughout the feasible price range ($p \in [0, 1]$), to only a closed neighborhood around p^* ($p \in [p^* - \sigma, p^* + \sigma]$) (Figure 4.3).

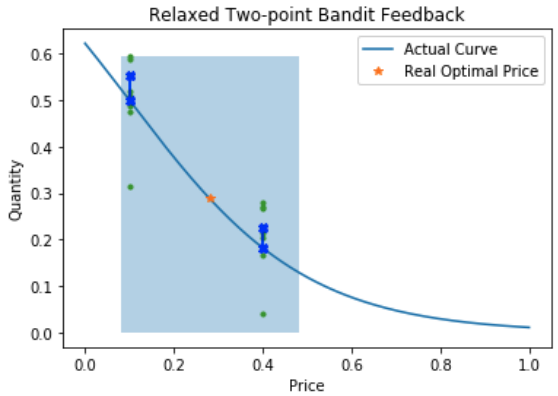

Figure 4.3: The two point bandit feedback is restricted to the shaded region around the unknown optimal price. The error in demand realization at the two prices is not the same. Its difference is bounded and decreasing with the number of demand realization.

In Theorem 4.4.5 we show that the price change and regret guarantees remain the same, albeit with a slightly modified SLPE Algorithm, Algorithm SLPE-Ext (Algorithm 5) presented

in Appendix C.4.

Theorem 4.4.5. Let the unkown real demand function $d(p)$ satisfy Assumptions (1) and (2A) with $f(n) = \frac{1}{n^\delta}$ and $\delta > 3/4$ and that $|d'(p^*)| \geq \left(\frac{K_1}{24} + \frac{\kappa}{4} + \frac{3}{\rho^2}\right)\frac{1}{4} + \frac{c}{2}$, for some $c > 0$. Also assume that $2 \geq \mu > M$, where M is defined in Lemma C.4.2 of Appendix C.4. Then the Regret of the SLPE-Ext pricing policy (Algorithm 5 of Appendix C.4) is,

$$\mathcal{R}^{SLPE(T)}(T) \leq \left(C_1 + C_2 \log\left(\log\left(\frac{T}{2\mu\rho^4 \log(T)}\right)\right)\right)\sqrt{T}\log(T) = \mathcal{O}\left(\log(T)\log(\log(T))\sqrt{T}\right),$$

where C_1 and C_2 are constants independent of T.

Proof. See Appendix C.4. □

The proof of Theorem 4.4.5 follows by proving results that are analogous to Lemma 4.4.2 and Lemma 4.4.3. These proofs, along with the proof of Theorem 4.4.5 are presented in Appendix C.4. We refer the interested reader to Appendix C.4 for further discussion on the algorithm as well as the proof.

We have so far discussed different relaxations of the two-point bandit feedback assumption. In what follows, we discuss a practical algorithm to elicit such pairs from stochastic demand realizations.

Finding feasible demand pairs satisfying two-point bandit feedback assumption:

The SLPE algorithm assumes that the decision maker has access to a pair of demand observations that satisfy the two point bandit feedback assumption (either Assumption 4.2.5 or 4.4.1). Naturally, two questions arise: (i) when is Assumption 4.4.1 satisfied, and (ii) how does the retailer estimate the pair of demand observations that satisfy this assumption? In what follows, we answer both these questions by first describing why the assumption is satisfied in many settings. Then, we describe a heuristic algorithm that finds the pair of demand observations that have the minimum difference between the error realizations.

Recall that the retailer has access to a series of demand observations $D_{1,..,n}(p_1)$ and $D_{1,..,n}(p_2)$. Each demand observation has an additive error: $D_i(p_j) = d(p_j) + \epsilon_i^j$. The objective is to find a pair of demand realizations that minimize the error:

$$\underset{i=1,..,n;j=1,..,n}{\arg\min} \quad |\epsilon_i^1 - \epsilon_j^2|\,. \tag{OP}$$

Problem (OP) is easy to solve if the real demand function, *d(p)* is known. Indeed, when $d(p)$ is known, the retailer can simply use the known error realizations to find the pair that minimizes the error. Furthermore, the optimal solution would also satisfy Assumption 4.4.1. Assumption 2A posits that the optimal value of OP is bounded above by $\mathcal{O}(n^{-3/4})$ as a function of n. To see the intuition behind this, consider the case when ϵ_i is uniformly distributed over a bounded space. Then as the number of samples n increases, the distance between the error realizations will decrease with a rate of at least $1/n$. In fact, similar intuition also holds in the case of bounded errors with a heavy tailed distribution. Hence, if OP could be solved, the optimal solution would satisfy the assumption.

But how should one solve OP when in the more practical setting, *d(p)* is unknown? One strategy is to use an estimate of the demand, $\hat{d}(p)$ instead of the unknown real demand. Then, problem OP becomes:

$$\underset{i=1,..,n;j=1,..,n}{\arg\min} \left| D_i(p_1) - D_j(p_2) + \hat{d}(p_2) - \hat{d}(p_1) \right|. \tag{OP-Approx}$$

It is easy to notice that while OP is not solvable, problem OP-Approx can be solved using data that the retailer has access to. In particular, $D_i(p)$ are known and $\hat{d}$ can be constructed using demand observations. Furthermore, notice that even if $\hat{d}(p)$ is not close to $d(p)$ and the approximation error is high, as long as the error at both p_1 and p_2 is close, then the solution of Problem OP-Approx is close to the solution of OP. Hence, in what follows, we propose a heuristic that ensures that the error in the estimation remains the same with high probability.

To motivate our proposed approach, we start by noting that in most practical situations, the error in demand observations is bounded. Hence, we will consider the case of bounded random variables with heavy tails. In such a setting, as the number of demand observations increase, the likelihood of at least one demand observation for any price, taking its maximum value is high. Hence, we simply let

$$\hat{d}(p) = \max_{i=1,..,n} D_i(p), \forall p\,.$$

Then, it is easy to notice that the optimal solution of OP-Approx is 0 and is achieved for $i^* = \arg\max_i D_i(p_1)$ and $j^* = \arg\max_j D_j(p_2)$. Nevertheless, this pair of demand observations might not be the optimal solution of OP. Moreover, it might not satisfy the $\mathcal{O}(n^{-3/4})$ upper bound on the error.

Finally, the proposed model is not the only model to estimate $\hat{d}$. In fact, a simpler model

could be to simply use average demand realizations at each price (p_1 and p_2). Then, OP can be written as

$$\underset{i=1,..,n;j=1,..,n}{\arg\min} \left| D_i(p_1) - D_j(p_2) + \bar{D}(p_2) - \bar{D}(p_1) \right|. \tag{OP-Average}$$

We now compare both OP-Average and OP-Approx with maximum demand as an estimate of the unknown demand OP-Max. We are interested in the performance of the two proposed methods, as the size of the data increases. The comparison will be based on how good the method approximates the real unknown demand difference, $d(p_1) - d(p_2)$. We generate n stochastic demand observations at prices p_1 and p_2. Each demand observation is a function of price. As n increases, the distance between p_1 and p_2 decreases, as is the case in the SLPE algorithm.

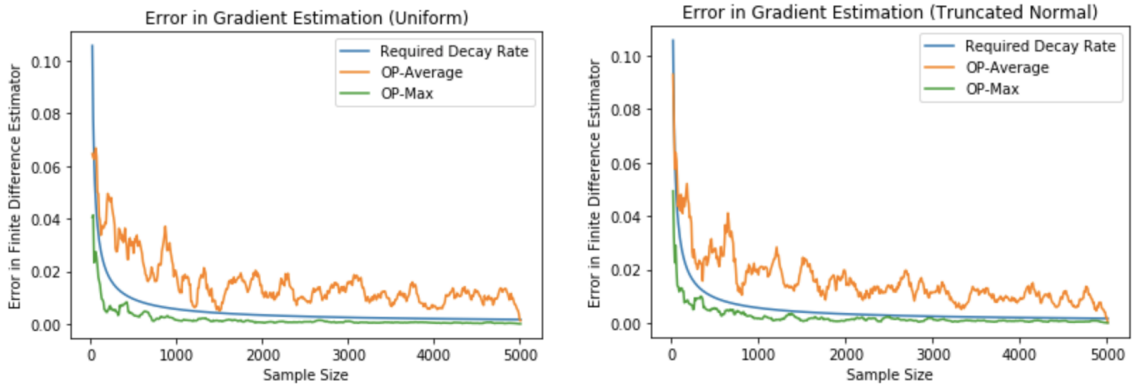

Figure 4.4: Comparison of OP-Average and OP-Approx with maximum demand. On the Y-axis we plot the error in estimation of the difference in demand at p_1 and p_2 as the number of stochastic demand observations, sample size, increases. On the left, we restrict the error to be Uniform [-1,1] and on the right, we restrict the error to be truncated Normal [-1,1] with 0 mean and unit variance. In both cases, the proposed heuristic outperforms the average based estimator. Furthermore, it consistently remains below the required decay rate necessary for price change reduction (see Theorem 4.4.5).

In Figure 4.4, we plot the difference between the estimated and actual difference between $d(p_1) - d(p_2)$ as we scale n and compare it with the minimum decay rate needed for Theorem 4.4.5 to hold (at least $n^{-3/4}$, see Theorem 4.4.5). Notice that the OP-Max solution consistently outperforms the average-based method. Furthermore, it is consistently below the minimum decay rate. This result continues to hold when the underlying error follows a uniform or a truncated normal distribution. This shows that our proposed heuristic method of using maximum demand observations at any price to estimate demand gradient is both computationally efficient and satisfies Assumption 2A. Hence, in the next section, when we compare the numerical performance of the algorithm, we use this heuristic to estimate demand gradient.

4.5 Numerical Study

In this section, we perform numerical experiments to compare the performance of the proposed SLPE algorithm over other algorithms. The purpose is two-fold: (i) to discuss practical implications of how to select different input parameters; and (ii) to compare the numerical performance of the algorithm with other state-of-the-art algorithms for this setting. In what follows, we discuss these implications through an extensive numerical study and start by discussing the various benchmarks.

Benchmarks: We consider two benchmark algorithms: (i) the single dimensional convex bandit optimization algorithm (Algorithm 1) of Agarwal et al. (2011), (referred to as *Bandit Convex (BC)*) (ii) the misspecified pricing scheme of Besbes and Zeevi (2015) (referred to as *Misspecified Pricing (MP)*). These two benchmarks are selected because neither make any parametric assumptions on the objective function. While neither of these limit price changes, both have comparable theoretical regret guarantees. Since a limited number of price changes can be a by-product of good regret performance, it is worthwhile to compare these algorithms with the *SLPE* algorithm. The *BC* policy is modified for the pricing problem under consideration. This ensures a fair comparison amongst benchmark algorithms. Chen et al. (2015) explicitly model a constraint on the number of price experimentation points. Nevertheless, since demand in their case is among several known parametric demand curves, their algorithm is considerably different from the current proposed policy and cannot be applied without further assumptions on the demand.

Performance metrics: We will compare all three algorithms in terms of 3 metrics: *LPC*, $\mathcal{R}(T)$ (see §4.2.2) and *LPE* that we discuss next.

Limited Price Experimentation, $\text{LPE}^{\pi}(T)$, for any feasible pricing policy π, measures the total number of unique prices used by the pricing policy until time T. That is,

$$\text{LPE}^{\pi}(T) = 1 + \left|\{2 \leq t \leq T : p_t^{\pi} \notin \{p_1^{\pi}, .., p_{t-1}^{\pi}\}\}\right|. \tag{4.23}$$

Since the clairvoyant knows the optimal price, p^*, the optimal price ladder size (LPE) is 1.

Remark 4.5.1 (Connection between LPE and LPC)**.** Both the LPE and LPC metrics are related to how a pricing policy switches between different prices. A low LPC ensures that the

policy does not switch between prices very often. In contrast, a low LPE ensures that the pricing policy only experiments with a small set of prices. While a low LPC naturally implies a low LPE, a low LPE can still lead to a high LPC. For example, the LPE of a 2-menu price policy would be 2 but if the policy switches at each time period, LPC of the same policy would be T. In Example 4.5.2, we make this connection clear by comparing two feasible pricing policies in a 5 period example.

Example 4.5.2 (Pricing policy comparison over different metrics)**.** Consider a retailer selling a single product with infinite inventory over 5 time periods. Under consideration are two pricing policies with the following period specific prices:

$$\pi^1 : \{p_1 = \$100,\ p_2 = \$90,\ p_3 = \$80,\ p_4 = \$75,\ p_5 = \$50\}\,,$$

$$\pi^2 : \{p_1 = \$100,\ p_2 = \$80,\ p_3 = \$70,\ p_4 = \$70,\ p_5 = \$80\}\,.$$

π^1 changes prices from one period to the other and charges a unique price in each period. π^2 changes prices between periods 1-2, 2-3 and 4-5. Furthermore, over the 5 time periods π^2 switches between a set of 3 unique prices. Hence, comparing π^1 and π^2 over LPE and LPC metrics, we have that $\text{LPC}^{\pi^1}(5) = 5$ and $\text{LPC}^{\pi^2}(5) = 4$ and $\text{LPE}^{\pi^1}(5) = 5$ and $\text{LPE}^{\pi^2}(5) = 3$.

As described before, each of these metrics relate to operational considerations of different pricing algorithms. Particularly, while LPE and LPC track the operational costs of the pricing policy under consideration, the cumulative regret tracks the revenue generation from the pricing policy. Policies that have low LPE, LPC and regret are the best performing since they would incur minimal operational costs while ensuring revenue maximization.

4.5.1 Synthetic Data

We generate synthetic data and evaluate the performance of all three algorithms under different demand realizations arising from the same parametric demand structure.

Data generation: We consider the following Logit demand model due to its wide applicability and use by both the academic community and practitioners. See, for example, Besbes and Zeevi (2015).

$$d(p) = \frac{\exp(\alpha - \beta p)}{1 + \exp(\alpha - \beta p)},\ \alpha \in [\underline{\alpha}, \bar{\alpha}], \beta \in [\underline{\beta}, \bar{\beta}], \textit{where } [\underline{\alpha}, \bar{\alpha}] = [0, 10] \textit{ and } [\underline{\beta}, \bar{\beta}] = [0.5, 10].$$

where $\underline{\alpha}$,$\bar{\alpha}$, $\underline{\beta}$ and $\bar{\beta} \in \mathbb{R}$ are such that the optimal price is between 0 and 1. The Logit demand model is an S shaped demand function with varying price elasticity across the feasible price range. For suitably chosen parameter values, the Logit demand function can model concave, convex or both demand models as well.

We consider 20 draws of the parameters α and β that are sampled according to a uniform distribution on $[\underline{\alpha},\bar{\alpha}]$ and $[\underline{\beta},\bar{\beta}]$. Each sample determines the underlying true demand model which is known to the clairvoyant. We compare all the three algorithms against the clairvoyant's optimal policy in each round. We denote by σ, the standard deviation of the error in the idiosyncratic demand response, fixed to be 0.1. Similarly, the total time horizon length (T) is fixed to be 5,000. In all cases the feasible price range is fixed to be the interval [0,1]. The *MP* policy has three parameters: block length, historical data length and a tuning parameter. We let the block length to be 2^i for round i, and tuning parameter to be 0.75. Finally we use all the historical data length to be the full history of collected data until that time. Note that this set of parameters ensure that the number of price changes remain $\mathcal{O}(\log T)$. Selecting block length to be 1, as is done in Besbes and Zeevi (2015) can further reduce regret at the expense of making more price changes. Since our focus is on price changes, hence our choice of parameters. There are no tuning parameters for the *BC* algorithm. Finally, *SLPE* has two tuning parameters: μ and ρ (see §4.3). While μ controls the size of exploration around the estimated approximate optimal price, ρ controls the depth of exploration. For example, increasing μ ensures that a larger pricing region around the approximated optimal price is explored in the future rounds. Similarly, a higher ρ results in more demand realizations being sampled at selected prices in each iteration of the simulation. We have already discussed the dependence of these parameters on the theoretical performance of the algorithm in §4.3. But, as mentioned previously, in more practical settings, the tuning of these parameters can be complex. Hence, in what follows, we circumvent this issue by fixing both $\rho = 1$ and $\mu < 1$. This modification does not change the regret guarantee of the policy. Nevertheless, fixing tuning parameters could lead to sub-optimal algorithmic performance. Finally, we use the SLPE-Ext algorithm which works under the relaxed assumption (see Assumption 4.4.1) and use the heuristic proposed in §4.4.1 to estimate demand observations satisfying two point bandit feedback. In what follows we compare different algorithms and show that the SLPE algorithm continues to outperform benchmark algorithms even when input parameters are fixed.

Results: Figure 4.5 shows the cumulative price changes (LPC) and the price ladder size (LPE) averaged over 20 different trials (along with the 95% confidence intervals) for all three algorithms. The proposed *SLPE* policy considerably outperforms both the *MP* and the *BC* policy. On average, the proposed SLPE policy makes only 4 price changes in comparison to more than 20 price changes by the *MP* policy and more than 35 price changes by the *BC* policy. This translates to almost an 80% reduction in the total price changes from the better performing *MP* policy. It directly translates to a reduction in the cost incurred by the retailer and can be detrimental to the success of a new product. This reduction in price changes can be directly attributed to the price selection process of the *SLPE* policy. Since the approximate piecewise demand yields a very good approximation of the unknown demand, the approximated optimal price is estimated with very good accuracy. Hence when this price is fixed for a large fraction of customers, it leads to very few total price changes without incurring considerable revenue loss (see Figure 4.6).

Similar improvements are also observed in the overall price ladder size (LPE). In particular, while the *SLPE* selects 4 unique prices, thereby selecting a new price every time a price change is made, both the *MP* and the *BC* policy repeat previously selected prices. While the MP policy selects from amongst 8 unique prices, the BC policy selects prices from 10 unique prices for the 5000 customers. This again translates to a 50% reduction in the size of the price ladder. This improvement is crucial in light of the negative behavioral effects of frequent price changes established by researchers (see PK Kannan 2001). A small price ladder ensures that customers do not think that they are discriminated against, based on unknown latent information independent of the product utility (see PK Kannan 2001).

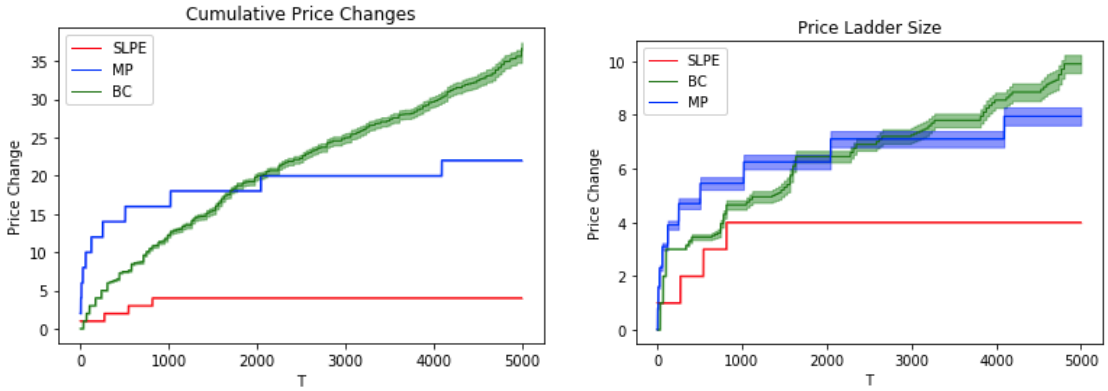

Figure 4.5: Cumulative price change (on the left) and cumulative price experimentation (on the right) with 95% confidence intervals for Logit demand specification.

We have so far shown that *SLPE* performs well in terms of the LPE and the LPC metrics,

outperforming the benchmark algorithms. Next, we focus on the regret metric of §4.2.2 that compares the revenue of a pricing policy with respect to the clairvoyant's optimal price. A priori, since limited price experimentation and price changes can slow down *learning*, one might expect that the *MP* or the *BC* policy outperforms *SLPE* policy. In Figure 4.6, we plot the cumulative regret and the 95% confidence intervals of all three algorithms. *SLPE* policy considerably outperforms *MP* and marginally improves over the regret of the *BC* policy. To begin with, we first note that the *MP* policy's regret performance can improve substantially at the expense of more price changes (see Besbes and Zeevi 2015). Hence, we change the block length parameter to 1 and rerun the policy to compare its regret. As expected, the regret performance improves substantially (right of Figure 4.6) but this leads to a linear increase in price changes. Hence, it might not be applicable in many offline retail settings.

The improvement over *BC* can again be directly attributed to the price selection of the *SLPE* policy. The *BC* policy selects experimental prices based on bisection search. Instead, the *SLPE* policy is fundamentally driven by a different intuition. Instead of merely using the demand observations to determine optimal price region in the next round, as in the case of the *BC* policy, the structure of the revenue maximization objective is used to guide price selection as well. Estimation of piecewise linear approximation leads to improved price point selection. Similarly, while the *MP* policy outperforms *SLPE* initially, since *MP* policy is forced to explore around the myopic estimated optimal price in each round, *SLPE* overtakes the *MP* since *SLPE* only explores when a suboptimal price region is identified and the policy moves to the next round.

Overall, we find that the proposed SLPE policy outperforms other benchmark methods in terms of the price change and price experimentation metrics with comparable performance in terms of the regret metric.

4.6 Conclusions

We consider the dynamic pricing problem of a retailer selling a single product when the underlying demand is unknown and non-parametric. The retailer seeks to reduce the amount of price experimentation due to the associated operational costs of price experimentation. To the best of our knowledge, the non-parametric demand setting with limited price experimentation has not been considered in the pricing literature so far. Limited price changes add another dimension to

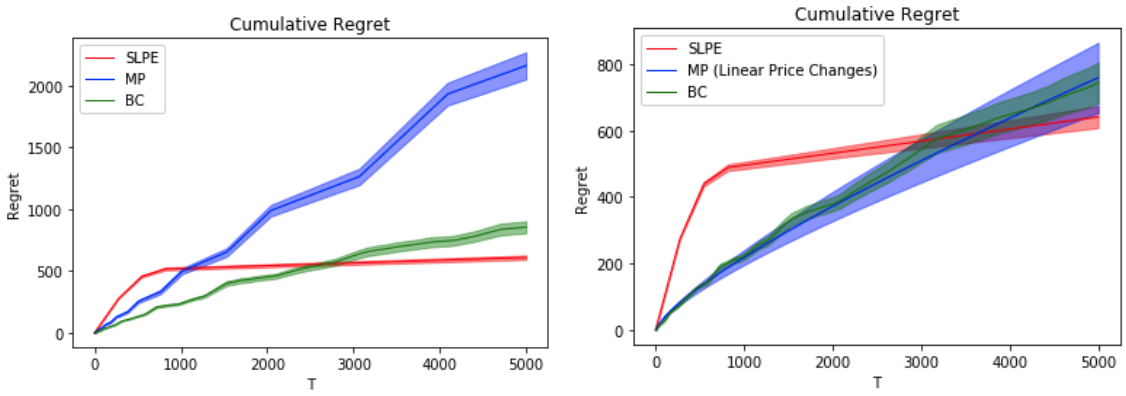

Figure 4.6: Cumulative regret and 95 % confidence intervals for Logit demand specification. On the left, the *MP* policy's performance with $\mathcal{O}(\log T)$ price changes. On the right, the performance of the *MP* policy improves substantially when frequent price changes are allowed ($\mathcal{O}(T)$). In both cases, the proposed *SLPE* policy performance of the revenue metric is comparable to that of the *BC* and the *MP* policy that make more frequent price changes.

the *exploration-exploitation* trade-off since learning and earning objectives might lead to price changes in every time period which are not desired. We construct a dynamic pricing policy that uses piecewise linear approximations of the non-parametric demand in order to generate future prices. Our proposed policy performs well both analytically and numerically. We show that the policy incurs $\tilde{\mathcal{O}}(\sqrt{T})$ rate of regret while the number of price changes grow at $\mathcal{O}(\log \log T)$ for a class of non-parametric demand functions. Evaluation on synthetic examples demonstrate that the policy reduces the number of price changes considerably while obtaining comparable maximum revenue.

Chapter 5

First Delivery Gaps: A Supply Chain Lever to Reduce Product Returns in Online Retail

5.1 Introduction

Online retail has become ubiquitous to shopping in recent years. More than 13% ($453.46 billion) of the overall retail purchases in 2017 came through online sales (Zaroban 2018), and e-commerce is growing at an average annual rate of 56%. This rise of e-retail is not restricted to the developed world. For example, the online industry in India alone is expected to grow by a staggering 1200% to more than $100 billion by 2020 (Ahmad 2018). But this exponential growth in developing countries comes with its own set of unique challenges.

A large portion of the population is new to online retail. Hence, getting traditional off-line customers accustomed to online shopping involves unique marketing, pricing, and operational strategies. A recent study by Goldman Sachs states that e-commerce players in India spend more than 30% of their overall budget on discounts (Ramnath 2016). Another challenge is that of bringing supply chain efficiency to e-retailers. With limited resources and subpar infrastructure, managing timely delivery of products becomes very challenging. Alyoubi (2015) states that logistical problems act as one of the biggest barriers in the growth of online retail. Finally, yet another significant and related challenge is that of product returns. The problem of returns is indeed a double-edged sword for retailers: it is a cost burner due to the two-way shipping costs that companies experience on the returned orders. Furthermore, the negative experience of

first-time customers could potentially lead to their complete disengagement with the e-retailer due to unmet expectations.

Given the negative impact that product returns can have on a company's bottom line, it is not surprising that many researchers have looked at the problem of returns and proposed operational strategies to reduce them (see Petersen and Kumar 2009 for a review). Nevertheless, very few researchers have analyzed probable supply chain levers, such as delivery gap, as potential causes of returns. In this chapter, we focus on a particular form of return: Returns to Origin (RTO). RTO is prevalent in India and other developing economies. An RTO product is one that has been shipped to the customer who then refused to accept it and sent it back. RTO products are different from usual returns mainly because they are not directly associated with product defects or mismatched product quality expectations. As noted by Bandi et al. (2017), lenient payment policies such as "cash on delivery (COD)" have led to a further increase in RTO. Although for the customer, a product RTO is a "zero cost" process, it causes further stress on the retailer's supply chain. The retailer not only incurs double cost of shipment, but COD orders that result in an RTO also leads to extra strain on the retailer's cash flow and finances.

The focus of the current work is to analyze the reasons for RTO and mitigate it by examining the process through a supply chain lens. In particular, we focus on the following key research questions: (i) *What is the impact of expediting deliveries on product RTO?;* (ii) *how does customer delivery promise drive product RTO?* (particularly, is it better to provide an exact estimate of delivery gap or should one be more robust in the delivery promises?); and, finally (iii) *given the operational costs attached to expediting deliveries, how can firms optimize on delivery gaps at an order level?* We answer the first two questions by estimating an econometric model of customers' RTO decisions and performing a large-scale RCT. In fact, this work is a result of an industry collaboration with Myntra, one of the largest online fashion retailers in India. Using the company's rich order-level transactional data set, we are also able to provide unique insights into customer RTO decisions. These insights lead to important managerial implications that, we conjecture, have wider applicability. Particularly, we conservatively estimate that a two-day reduction in delivery gaps from the current average can result in an overall cost savings of \$1.5 million per year for the industry partner from RTO reduction. We also find that for faster deliveries with little scope of further delivery improvement, *beating* the customer promise date by a larger margin is better for RTO reduction. Finally, we answer the third

question by proposing a joint bilevel optimization problem that constitutes the strategic and the tactical delivery expediting optimization problems, that can optimally balance delivery costs with potential savings from RTO reduction. The objective in these problems is to reduce the delivery gap while accounting for delivery improvement costs, which can be substantial. Using data from our industry partner, we show that even in regions where delivery gaps are considerably fast, our proposed delivery threshold recommendation could lead to cost savings of up to 2.7%, accounting for RTO and delivery improvement costs.

5.1.1 Contributions

In this chapter, we analyze the problem of RTO reduction through a supply chain lens. We answer the question of what causes RTO (increased delivery gaps) and what can be done to reduce RTO (through optimizing the joint Optimal Delivery Thresholding Problem (ODTP) and the Optimal Delivery Expediting Problem (ODEP)).

- *Relation of RTO and the First Delivery Gap (FDG):* We investigate the hypothesis that an increase in delivery gap could lead to an increase in product RTO. Using data from a large fashion e-retailer in India, we estimate an econometric model of a customer's RTO decision and establish that a reduction in the gap between order placement and delivery attempted date can have a positive impact on RTOs.

- *Relation of RTO and delivery promise date:* We conduct a large-scale RCT in a region where product deliveries are *fast*, and establish that *beating* customer promise date by overshooting it can lead to further reduction in RTO.

  We propose the joint strategic and tactical delivery optimization problems to effectively balance delivery improvement costs with potential cost savings from RTO reduction.

- *Optimal Delivery Thresholding:* We introduce the strategic Optimal Delivery Thresholding Problem (ODTP) to choose an optimal delivery threshold for the retailer so that delivery of all orders is attempted within that threshold. The optimization formulation we introduce is data driven (in fact, its inputs can be estimated using transaction-level data). The formulation balances the cost of delivery improvements with RTO costs while accounting for uncertainty in the delivery times. We establish that the objective function of the ODTP is neither concave nor convex. Furthermore, it is not even unimodal. Nevertheless, by characterizing the regions of convexity of the objective function, we are able

to determine the unique optimal threshold solution.

- *Optimal Delivery Expediting:* To operationalize the strategic threshold of ODTP, we formulate the multi-product Optimal Delivery Expediting Problem (ODEP) under budgetary constraints and characterize its optimal solution. We propose an integer programming (IP) formulation for the ODEP, and use a Linear Programming (LP) relaxation-based heuristic solution for the ODEP. We also establish that our heuristic approach has a very small optimality gap in comparison to the computationally intensive IP solution. We illustrate the applicability of the model by recommending an optimal threshold and tactical delivery expediting levels for our industry partner. Furthermore, we estimate that the proposed threshold could reduce costs by as much as 2.7% for our industry partner.

5.1.2 Literature Review

Reducing returns is increasingly becoming an important operational problem, and researchers have looked at both, understanding the causes of returns through econometric studies as well as operational strategies for reducing returns.

Econometric studies for understanding causes of returns: Prior research in returns has focused on answering two major questions: *why* customers return products and *what* is the value of such returns to the customer (Rao et al. 2014). Different behavioral reasons have been attributed to why product returns happen. Particularly, customer satisfaction and cognitive dissonance have been found to be important drivers of product returns (Powers and Jack 2013). Both these factors are hugely impacted by the overall transaction and post-purchase experience of the customer. Post-purchase experience is driven by product defects, quality, compatibility, and physical distribution services (Anderson et al. 2009, Gallino and Moreno 2018). Nevertheless, not many researchers have looked at the problem of returns through a supply chain and operations perspective, especially in the online retail setting. Rao et al. (2014) use online transaction data to show that physical distribution service plays an important role in customer returns. Like the current work, they find that customer satisfaction, driven by the reliability of delivery service, drives customer returns. However, we use different identification strategies (instrumental variables and an RCT) to measure the effect of faster deliveries on RTO. Furthermore, analysis on larger data sets ensure more robust findings. Similarly, Bandi et al. (2017) find that observed post-purchase price drop is another cause of returns as it

gives rise to opportunistic returns. Because they do not focus on delivery services as a cause of returns, their work is considerably different from the current work. Finally, Fisher et al. (2016) provides an empirical estimate on the revenue impact due to improvement in delivery speeds. Like the current work, they use transaction data from an e-retailer to show that after accounting for different delivery improvement costs, the net effect of such improvements on the revenue is still positive. Nevertheless, they do not estimate the impact of such improvements on returns. Hence, the current work can be considered complementary to their work. Somewhat related, researchers have also analyzed how customers react to better (and worse) service quality. For example, Smith and Bolton (1998) analyze the effect of service failure on the customer's overall assessment of the service provider. More recently, Proserpio and Zervas (2017) show that targeted response to customer online reviews can lead to an overall increase in customer ratings. Finally, Cohen et al. (2018) show how to use promotions effectively in case of unmet service expectations in ride sharing. Because the focus of the current chapter is on returns, the current work substantially differs from the previously cited studies.

Operational models for reducing returns: Two main stream of literature consider the problem of optimizing return operations. One looks at optimizing the reverse logistics of suppliers for supply chain efficiency (see Rogers and Tibben-Lembke (2001) for a review), while the other looks at reducing returns by optimizing retailer's return policies and other factors that affect customer returns. Because the current work relates to the latter, we detail the literature in this stream next. Davis et al. (1995) were the first to recognize the effect of full refund policies on returns. Since then, many researchers have recognized the effect of return polices on returns, including Chen et al. (2008b), Su (2009), Chen and Chen (2017), and Nageswaran et al. (2017), among others. Nevertheless, because these works focus on optimizing return policies instead of delivery gaps, they differ considerably from the current work. Somewhat related, the importance of minimizing delivery lead time in the area of e-commerce and getting the product to the costumer as soon as possible has been identified as a key characteristic of success for online retailers (see, for example, Keeney (1999), Swaminathan and Tayur 2003). Following this, researchers have also analyzed various delivery expediting policies. For instance, Li (2013) considers the optimal logistics network design problem with expedited delivery option to minimize lead times and other costs. Similarly, Chen et al. (2008a) consider the optimal network design problem when deliveries happen at particular time intervals. These studies differ consid-

erably from the current work because we consider the case when the logistics network is a given, and we instead focus on optimizing delivery times. Particularly, our optimization framework optimally prioritizes orders for delivery so that RTO can be reduced. Finally, another stream of literature considers optimal delivery policies in multichannel retail (see Guide Jr et al. (2006) for a review). To the best of our knowledge, however, prior papers have not analyzed the problem of minimizing delivery times in the context of returns reduction. The objective function as well as the cost structure considered in the current work makes it considerably different from the prior work.

5.2 Motivation from an Online Fashion Retailer

Our industry collaborator, Myntra, is one of India's largest fashion e-retailers. The retailer has annual revenue on the order of a billion dollars and ships more than 150,000 items to its customers on average every day. Furthermore, the retailer sold more than 1.8 million unique products (SKUs) in a 1-year period (spanning 2017-2018). The retailer sells both in-house products (products manufactured, marketed, and sold by the e-retailer) and products from other sellers. Products sold on the platform include apparel, footwear, and other accessories. Customers are allowed to pay online or in cash at the time of delivery (COD), and deliveries happen through a complex supply chain network. Because the current work focuses on reducing product RTO, we start by providing some descriptive statistics on RTO and the retailer's logistics process.

Figure 5.1 shows the revenue contribution and RTO rates among different product categories. We find that the RTO rate is significant, and is a cause of concern for the retailer. When an order is returned, it is shipped back to its originating warehouse, where it goes through a quality-check (QC) process before becoming part of the forward inventory. Returned shipments traverse through different nodes of the supply chain in a reverse direction, starting from the customer's doorstep and eventually ending up at the warehouse. Naturally, and as a result, these returned orders incur double the costs for the retailer. Hence, reducing returns is an important problem to address for our industry collaborator.

Although product returns are often attributed to unmet expectations in terms of the *quality and other product-specific characteristics*, reasons for RTOs are hard to pinpoint. For starters, RTOs are product returns that happen when customers order the product of their own *free will*

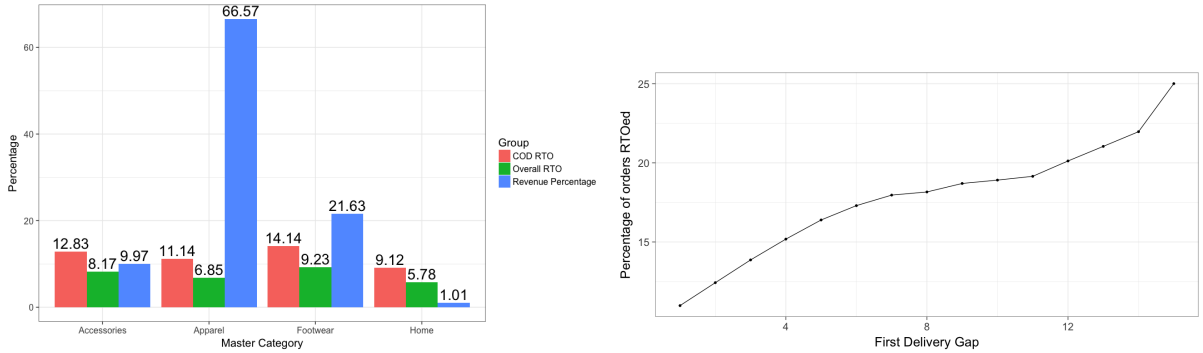

Figure 5.1: On left, RTO rates (in %) across different categories for both online and COD orders. On right, change in RTO orders with change in the FDG.

but decide to return it without opening the package when it eventually reaches their doorstep.

A good starting point to understand the causes of RTOs could be to examine the problem through a supply chain efficiency lens. Particularly, we pose the hypothesis that delays in delivery could lead to increased RTO orders. Customer reddress-seeking behavior due to the lack of service fulfillment satisfaction has been well demonstrated and studied (see, e.g., Berry et al. 1994, Weiner 2000). Unlike traditional retailers, for whom location plays an important role in customer satisfaction, online retailers rely on delivery fulfillment quality for customer satisfaction (Rabinovich and Bailey 2004, Rao et al. 2014). Whereas the effect of service quality has been well studied in the offline setting, very few researchers have studied its effect in the context of online retail. Following this literature, we expect that an increase in delivery gaps could lead to an increase in RTO.

Hypothesis 1 (H1)**.** An increase in days between the order placement date and the first delivery attempted date leads to an increase in product RTO.

Figure 5.1 shows the percentage of RTO orders versus the gap (in terms of days) when the order was placed relative to the first time delivery of the order was attempted (FDG). Notice that there is a clear positive trend associated with RTOs and delivery gaps. If one finds further evidence of **H**1, an important intervention would be to decrease the delivery time for customers. However, delivery time is driven by the supply chain structure of the industry collaborator. Hence, it is crucial to understand the current order fulfillment process of the retailer. Orders are fulfilled using two different models: *marketplace* and *inventory*. Products fulfilled through the *inventory* model are stored in the retailer's warehouses, whereas products fulfilled through the *marketplace* model are stored in the seller's warehouses. When an order is placed, the respective items are first brought to one of the retailer's warehouses. Items are

then shipped to the customer's location. Each order goes through a series of line halls, regional hubs, and delivery centers (DC) before it is finally picked up by the last mile delivery personnel for delivery attempts. Figure D.1 in Appendix D.4 describes the overall supply chain structure of the retailer. The four main steps of the delivery process can be described as follows:

1. An item involved in an order is identified as either *inventory* or *marketplace*, depending on where it is stored in the supply chain.

2. The retailer finds the warehouse that is closest to the customer's delivery address, and the item is shipped from the current warehouse to the closest warehouse.

3. The item goes through a QC process at the shipping warehouse and is then packed to be shipped to the last mile DC.

4. The item is then delivered to the customer's doorstep via last mile delivery personnel.

Many interventions can lead to expedited product deliveries. In particular, one intervention can be to implement a policy of attempting all deliveries within a threshold period from the day of order placement. While such a change could bring RTO rates down, it could also lead to substantial delivery improvements costs. Hence, balancing these costs in itself can be a complex problem, particularly due to the retailer's large supply chain network.

Nevertheless, even with a lot of cost investments, customer deliveries cannot be brought down below a threshold. For example, in regions where the collaborator's supply chain is already very efficient, and delivery times have been minimized (one-day deliveries), improving delivery times further is nearly infeasible. In these regions, we ask if *customer promise* can be used as a tool to reduce product RTO. Customer promise is an estimate of the delivery date that is made to the customer immediately after an order is placed on the retailer's online platform. In regions where delivery is *fast* and less variable, the industry collaborator has two options: (i) either *meet* the customer promise; that is, deliver the order on the date of customer promise; (ii) or *beat* the customer promise, that is, deliver the order before the customer promise date. For example, giving a customer promise date of 1 day from the order placement date and delivering within 1 day would mean meeting the promise. But instead, if the customer promise is 4 days and the order is delivered within 1 day, it would imply *beating* the promise. The Expectation Confirmation Theory (ECT) of Oliver (1980) posits that exceeding customer expectation results in positive disconfirmation, which leads to customer satisfaction (Rao et al. 2014). Following

this theory, we hypothesize that for faster deliveries, *beating* the promise can lead to further reduction in product RTO.

Hypothesis 2 (H2)**.** When delivery attempt is made within 1 day of order placement and before the customer promise date, an increase in days between the customer promise date and the delivery attempted date leads to a decrease in product RTO.

If we find evidence supporting **H2**, retailers can increase customer promise in regions where there is little scope of delivery improvements or where deliveries are already expedited. This change in customer promise could lead to a further reduction in product RTO.

In the remainder of this chapter, we will first tease out the effect of the FDG on RTO and show evidence supporting **H1**. We will then describe an RCT that was conducted in collaboration with Myntra to test **H2**, and provide further evidence supporting **H1**. Finally, we will describe the strategic and tactical optimization problems and discuss cost savings if deliveries are expedited based on the proposed recommendations.

5.3 Empirical Analysis

In this section, we discuss the details of the empirical approach used to test the hypothesis developed in §5.2. We start in §5.3.1 by providing descriptive statistics related to the data set. Then, in §5.3.2, we discuss the empirical approach and the potential challenges related to the approach. In §5.3.3, we discuss results from the empirical analysis.

5.3.1 Data and Descriptive Statistics

As mentioned before, we use a comprehensive transaction-level data set from Myntra to answer our empirical research questions. In total, the data set includes more than 56 million transactions that occurred over a 12-month period (2017-18). We also have access to other information associated with each transaction, such as product- and customer-related features. This information creates a unique advantage because we are able to control for various factors that could not be controlled otherwise due to data size issues. To make the analysis more tractable and insightful, we focus on *footwear* orders with COD payment. The RTO rate among orders with the COD payment type is significantly higher than online payments. Furthermore, footwear is the second-largest category in terms of revenues and, at the same time, has the highest RTO

rates among different product categories (Figure 5.1). In what follows, we present descriptive statistics corresponding to the footwear category.

Footwear category at a glance.

Footwear is one of the most popular product types on Myntra's online platform. More than 7 million footwear orders were made over the 1-year period of the study. These products were associated with more than 400 brands. Out of these, 87 brands contributed more than 95% of the overall revenue. Hence, we focus our analysis on transactions from these brands. Transaction-level decisions, such as price charged and discount offered, are dynamically generated and vary from order to order. For example, we find that the prices of footwear products are highly variable with a mean-to-variance ratio of 0.001. Products are highly discounted: more than 50% of orders receive a discount of more than 50% over the retail price. This discount further corroborates the general trend of heavy discounting in e-commerce retailers (Ramnath 2016). Because product features such as price and discount can affect RTO decisions, we provide some summary statistics and underlying product RTO implications next. Particularly, we focus on (i) the type of product, (ii) the maximum retail price (MRP) charged, (iii) the selling price after excluding all discounts, and (iv) the discount offered on the product.

- *Article type:* Myntra sells 10 different types of footwear products. The top three article types in the footwear category that comprise 70% of the overall orders are *casual shoes (44.8%), flip flops (11.9%), and sports shoes (15.54%).* The RTO rate among these different groups varies significantly. For example, 16.6% of casual shoe orders were RTOed, and 17.4% of sports shoes orders were RTOed. For flip flops, the RTO rate was 14.7%, almost 3% lower than the other two groups. This shows significant heterogeneity in RTO decisions based on the product article type.

- *MRP:* The maximum retail price (MRP) is the price of the product. The retailer often offers discounts over this price. We find that there is high variability in the MRP of footwear products sold. The average MRP is Rs. 2862 with a standard deviation of Rs. 1648. This variability also leads to differences in RTO decisions. As the MRP increases, the likelihood of RTO also increases. The RTO rate of orders in the bottom 25% quantile in terms of the MRP was 12.5% compared to an RTO rate of 17.5% for orders with an MRP in the top 25% quantile.

- *Price paid:* Price paid for an order is the MRP minus discounts offered on the order. Because the price paid is highly correlated with the MRP, it is also highly variable. The mean price paid for a footwear product is Rs. 1610 with a standard deviation of Rs. 1025. The significant difference between the MRP and price paid shows that products are highly discounted (analyzed next). We find empirical evidence of positive correlation between price paid and RTO. Comparing the bottom 25% quantile of orders with the top 25% quantile of orders in terms of *price paid*, we find that the RTO rates differ by almost 6.2% (12.2% vs. 18.4%).

- *Discount:* Footwear products on Myntra are heavily discounted. The average discount offered on products is 47.8% with more than 50% of products being offered at a higher than 50% discount on the MRP. The effect of discounts on RTO decisions is hard to anticipate. While a high discount could mean a lower price paid, it could also be related to the perceived product quality being poor. Nevertheless, we find evidence of negative correlation between discount and RTO rates: the RTO rate of all orders with discounts below 40% (bottom 25% quantile) is 15.6% as compared to that of a 14.17% RTO rate for orders with discounts above 60% (top 25% quantile).

In summary, we find that different order-level features play a key role when customers decide on whether to RTO a product. While the focus of the current work is on supply chain features (the FDG and the difference between actual and promised deliveries), the above discussion provides an intuition about other features that could affect RTO decisions that need to be controlled for in an econometric analysis.

5.3.2 Econometric Specification

The dependent variable in our analysis is rto_i, the RTO decision associated with order i. The rto_i is 1 if order i is RTOed and 0 otherwise. As noted before, the RTO decision of an order depends on various order-level factors, such as the price, discount offered, and delivery experience. Let $\boldsymbol{C_i}$ denote all these order-level controls. We are particularly interested in the effect of the FDG on RTO decisions. Hence, let

$$FDG_i = \textit{First Delivery Attempted Date}_i - \textit{Customer Order Date}_i$$

denote the FDG of order i. Similarly, to control for potential effects of customer promise with respect to the actual delivery attempt date, we let Actual vs Promised Delivery (APD_i) gap of order i be

$$APD_i = \textit{First Delivery Attempted Date}_i - \textit{Customer Promise Date}_i\,.$$

Notice that a positive APD_i implies that the promise of an order was not met, meaning that the first delivery was attempted after the customer promise, and vice versa for the negative APD. Because the customer response to meeting the promise versus not meeting the promise can be very different, we let

$$APD_i^+ = \begin{cases} 0, & \text{if } APD_i \leq 0 \\ APD_i, & \text{otherwise} \end{cases} \quad \text{and} \quad APD_i^- = \begin{cases} 0, & \text{if } APD_i \geq 0 \\ APD_i, & \text{otherwise} \end{cases},$$

denote the positive and negative parts of APD_i. Then,

$$rto_i = \alpha FDG_i + \beta^+ APD_i^+ + \beta^- APD_i^- + \gamma^T \boldsymbol{C_i} + \epsilon_i \tag{5.1}$$

where ϵ_i is the idiosyncratic zero mean noise term associated with transaction i that is uncorrelated with $\boldsymbol{C_i}$. A positive value of α would imply that an increase in the FDG would result in an increase in RTO, supporting **H1**. Similarly, positive β^+ would imply that whenever customer promises are not met, an increase in the delivery gap would lead to an increase in the product RTO. Finally, a positive β^- would imply that if customer promises are met, it is better to *beat* the customer promise by a larger margin. Next, we discuss potential challenges of the above econometric approach and discuss methods to overcome these.

Empirical challenges.

Because the econometric analysis presented below is based on observational data, the analysis is prone to the usual pitfalls associated with inference of this kind. We discuss some of these issues in detail next.

- *Potential endogeneity of the FDG:* As the negative effects of RTO are considerable, retailers usually enforce periodic review policies to keep RTO in check. This raises concerns about potential reverse causality. For example, at our industry collaborator, as well as at

other online retailers, inventory decisions that affect the FDG can be based occasionally on the previous period's RTO. At Myntra, last mile deliveries go through DCs that are managed by supply chain managers. Each DC is responsible for order fulfillment in a small geographical region. Furthermore, monthly reviews ensure that critical customer service metrics, such as RTO rates, remain under control. When deliveries in a particular region are RTOed more than other regions, managers try to push for faster last mile deliveries, which, in-turn, leads to reduced FDGs. Indeed, such interventions (without controls) would result in an endogeneity issue in our panel data analysis as RTO causes the FDG to change and not the other way round, as conjectured. To control for such interventions, we add month-DC level fixed effects to our base model. Because we have data from one full year, these fixed effects would account for any potential intervention from month to month at any DC.

- *Omitted variable bias:* Although we have access to a rich data set, RTO decisions can be driven by other unobservables that we cannot control for. In the case of omitted variables, estimation of α based on (5.1) would be biased. To assuage such concerns, we perform an instrumental variable (IV) analysis (Imbens 2014). Specifically, we instrument the FDG with a *warehouse time* metric that captures the effect of travel time between different zip codes and warehouses on delivery gap. Details of this analysis are presented in §5.3.3.

- *Customer and product heterogeneity:* Product RTO decisions can be dependent both on the customers ordering the products and the products being ordered. Although, limitations in terms of the data set size constrain the number of controls in the econometric model, access to a large data set provides a unique advantage. Particularly, we control for customer level fixed effects in our model specification, which allows us to account for customer-level heterogeneity in RTO decisions. We also account for various product-level features such as brand, article type, price, and discount offered apart from other variables in our panel analysis. These controls are further detailed in §5.3.2.

Controls.

Equation (5.1) includes several control variables. At the product level, brand-article type fixed effects let us control for distinct characteristics of each article type of every brand. We also control for the price paid and discount offered for an order i. At the customer level, we add

fixed effects for every individual customer to control for invariant customer characteristics. In order to account for reverse causality and seasonal effects, we add month-delivery center level fixed effects. Finally, to account for delivery quality that can also drive RTO decisions, we add courier partner and supply type fixed effects.

5.3.3 Results

In this section, we present the results of the econometric analysis and show evidence that validates **H1**. We start by detailing the results of the base model panel analysis with different controls. We then discuss the IV analysis and perform several statistical tests to check for robustness of our findings.

Base level panel analysis.

Column (1) of Table 5.1 reports the estimation results from the panel analysis with no instruments. We find strong evidence supporting **H1**. Particularly, the coefficient of the FDG is positive and significant at the 99% significance level. This implies that when controlling for various order-, customer-, and product-level features, an increase in delivery gap leads to an increase in RTO. We also find the coefficient of APD^+ to be positive and significant which implies that whenever customer promises are not met, an increase in the delivery gap results in increased chances of product RTO. The coefficient of discount is negative and significant, and the coefficient of price is positive and significant. Hence, RTO increases with the price of the product and decrease with the discount offered for the product. Although the coefficient of APD^- is positive, it is not significant. Hence, no inference can be drawn on the dependence of APD^- on RTO decisions. Nevertheless, the significant difference between the coefficients of APD^+ and APD^- is further proof of the heterogeneous effects of meeting versus not meeting delivery promise on product returns.

Variable	**OLS**	**IV**	**Single Zip Code**
FDG	0.007*** (0.000)	0.014*** (0.001)	0.011*** (0.002)
APD^+	0.006*** (0.000)	−0.001 (0.001)	−0.000 (0.008)
APD^-	0.000 (0.000)	−0.001** (0.000)	0.000 (0.001)
price	+0.000*** (0.000)	+0.000*** (0.000)	0.000*** (0.000)
product discount	-0.079*** (0.002)	−0.100*** (0.006)	0.004 (0.022)
Observations	2,392,061	1,662,175	20,998

*$p < 0.10$, **$p < 0.05$, ***$p < 0.01$

Table 5.1: Estimation results from different regression models. In column (1), we present the results from the base-level panel analysis; in column (2), we present the results from the IV analysis; and in column (3), the results from the regression analysis based on transactions from a single zip code.

Instrumental variable (IV) analysis.

While the panel analysis shows strong evidence supporting **H1**, coefficient estimates can be biased because of potential omitted variables in the model specification (see §5.3.2). To assuage these concerns, we propose an IV analysis (For a comprehensive review of the technique, we refer the interested readers to Imbens (2014).) A valid instrument for the FDG is a variable/s that satisfies the following two validity conditions: (i) the instrument is uncorrelated with the error term of equation (5.1), and (ii) the instrument is highly correlated with the variable of interest, FDG. After a valid instrument is found, standard two-stage procedure can be used to get an unbiased estimate of the endogenous variable. Before we propose the instrument, we describe in greater detail the supply chain structure of the industry collaborator that drives our intuition about the instrument.

Myntra fulfills most orders through inventories stored in three large warehouses that are located in Mumbai, Delhi and Bengaluru, which are large metro cities located in the west, north, and south of the country. The product then traverses through a complex supply chain network before arriving at the customer's doorstep (see §5.2). Naturally, because most products originate from one of the three warehouses, the overall delivery gap of an order is driven by the time that it takes for the product to leave the warehouse and get to the customer's location. Furthermore, inventory storage decisions are exogenous to the RTO decisions of the customer and are driven by capacity and other logistic considerations at the country-level. Let Z_i and w_i define the zip code, warehouse, respectively, associated with transaction i. In addition, let $d_{kl} \in \mathcal{R}^+$ denote the travel time between two locations k and l, calculated using the Google Maps application programming interface (API). Then, $d_{Z_i,M}$ defines the travel time between zip code Z_i and Mumbai, and $d_{Z_i,B}$ and $d_{Z_i,D}$ are defined analogously. Finally, the warehouse travel time of an order can be defined as

$$\textit{warehouse time}_i = d_{Z_i,B}\mathbb{1}\{w_i = B\} + d_{Z_i,D}\mathbb{1}\{w_i = D\} + d_{Z_i,M}\mathbb{1}\{w_i = M\}.$$

We use the *warehouse time* to instrument the FDG. We argue that it is a valid instrument because it satisfies both the IV validity conditions. Particularly, because the order was sourced from the corresponding warehouse, such a time metric should be highly correlated with the FDG of the order and satisfy the second condition of being a valid instrument. Similarly, after controlling for various covariates, we do not expect the *warehouse time* metric to be correlated with the idiosyncratic error of (5.1). Such a correlation would imply that the customer order or RTO decisions are correlated with the distance of the warehouse from which they are getting

served. Nevertheless, as we noted before, there is no a priori reason to believe that such a correlation would exist. We perform various statistical tests to check the validity of the instrument, which we discuss next.

We start by presenting the results obtained from the first stage of the two-staged least squares (2SLS) estimation for the endogenous variable, FDG. The model specification, including controls, remains the same as before. In particular, we control for customer-level heterogeneity by adding customer-level fixed effects to the model. Product-level heterogeneity is controlled by article-type and brand-level fixed effects along with price, discount, and customer promise controls (see (5.1)). In Table D.1 of Appendix D.4, we report the estimation results from the first-stage analysis. Particularly, the coefficient of *warehouse time* is significant (at the 99% significance level) and positive. Furthermore, the adjusted R^2 of the first-stage regression is 0.77 and the within R^2 of 0.44. The partial R^2 of the *warehouse time* instrument is 0.21 with an F statistic above 10^6, showing the strong predictive power of the proposed instrument. Moreover, we also perform statistical tests to check if the proposed instrument is weak or under-identified. We find evidence of neither. Particularly, the the Kleibergen-Paap rk Wald F statistic for weak identification is 527.10, beating the Stock-Yogo weak ID test critical value of 16.38 by a considerable margin. Similarly, the Kleibergen-Paap rk LM statistic that tests for under-identification is 21.09. All these results continue to hold at a coarser level of error clustering (see §5.3.3).

In column (2) of Table 5.1, we present the results obtained from the second stage of the 2SLS estimation. The coefficient of the FDG remains positive and significant at the 99% significance level. Hence, accounting for potential omitted variables and endogeneity, we still see a strong effect of the FDG on RTO. As before, the directional insights with respect to price and discount remain consistent: RTO increases with an increase in price and decreases with a decrease in product discount. Although the coefficient of APD^+ is negative, it is not significant.

Robustness checks and instrument validity.

In the previous section, we provided intuition behind the proposed instrument and backed its validity with the results from various statistical tests. However our identification strategy is driven by variation in the travel time of an order from a warehouse to a particular zip code. Hence, if a large fraction of zip codes are always served from the same warehouse, then the *warehouse time* metric could be correlated with unobservable confounders associated with that

zip code region. Similarly, if particular products are always served from the same warehouse, then the *warehouse time* metric would be correlated with order level features, which would make the coefficient estimates of the FDG biased. Finally, if other unobservables affect both the RTO and order decisions of the customer, as well as the *warehouse time*, it would again invalidate the instrument. In what follows, we provide further empirical evidence of instrument validity by performing various robustness checks.

First, we perform a zip code level analysis to analyze the distribution of warehouses that serve a particular zip code. To this effect, for each zip code, we find the percentage of total orders that were served by the corresponding closest warehouse. We compare this percentage with orders served from the second-closest warehouse and the farthest warehouse. On average, only 51% of the orders were served from the closest warehouse of a zip code. Furthermore, more than 20% of the orders were served from the farthest warehouse. These statistics attest to the claim that inventory fulfillment decisions are very dynamic and are affected by various country-level factors that include the capacity of the warehouse, the safety stock, the available logistics capacity for inventory movement, and others. These factors bring significant heterogeneity to the *warehouse time* metric, which we then exploit to identify the effect of the FDG on RTO.

Second, we investigate whether orders of particular footwear products are always served from the same sourcing warehouse. We find that order fulfillment at the level of brand and article type is relatively homogeneous among the three warehouses. On average, 27% of orders of a selected brand were served from the Bengaluru warehouse, 41% were served from the Delhi warehouse, and 32% from the Mumbai warehouse. Similarly, on average, 29% of orders of a selected footwear type (article type; see §5.3.1 for more details) were sourced from Bengaluru, 38% from Delhi, and 31% from Mumbai.

Third, to assuage concerns related to potential heterogeneity in customer population and other socioeconomic factors that could affect RTO decisions and are unobserved, we first run a region-level panel analysis by focusing on a single zip code. A significant effect of the FDG would provide further evidence for **H1**. This panel analysis on a subset of the data is performed over the zip code with the highest number of orders (33,926 orders). We use the same model specification and controls as before (see §5.3.2). In column (3) of Table 5.1, we present the results of the single zip code panel analysis. The effect of the FDG on RTO is positive and significant at the 99% significance level. We run the same analysis on five randomly selected zip codes and find that the effect of FDG on RTO is robust and persistent: the coefficient of the

FDG is consistently positive and significant above the 90% significance level. We also run the IV analysis on a subset of the data set: only orders from zip codes that are close (less than 2 days in travel time) to all three warehouses are included in this analysis. Since zip codes close to bigger metropolis regions should be relatively comparable in terms of unobserved socioeconomic factors, this analysis would provide further evidence of the effect of the FDG on RTO decisions. We find that the effect of the FDG on RTO continues to be significant and positive, and the instrument passes all validity tests. Furthermore, the effect continues to be persistent as we change the subset to only include orders from zip codes that are at most 1 day away from all warehouses or 3 days away from all the warehouses (see Table D.2 of Appendix D.2).

Finally, we run the 2SLS model with different levels of error clustering. Particularly, we cluster errors at two different coarser geographical levels, namely zip code and district (a district is comprised of several zip codes). Cameron and Miller (2015) have noted that coarser error clustering leads to weaker effects. The effect of the first delivery gap on RTO continues to remain significant at the district-level error clustering, implying that the effect is indeed robust. In fact, the results in column (2) of Table 1 are at the district-level error clustering. Furthermore, the instrument also passes statistical tests for instrument validity at the coarser error clustering level. Finally, as an alternate model, we use a probit model specification due to the binary nature of the response variable. We perform the Probit analysis with 10 data sets comprising a 10% randomly generated sample of the overall data set. In each case, the effect of the FDG on RTO continues to be positive and significant at the 99% significance level. The average coefficient of the FDG over the 10 samples is 0.092 with a standard deviation of 0.003. This confirms the robustness of the effect of the FDG on RTO.

We have so far presented strong empirical evidence of the effect of the FDG on product RTO. Nevertheless, delivery improvement becomes significantly difficult after a certain level. To understand if the delivery promise plays a role in product RTO decisions, we run a large-scale RCT. We discuss the RCT and the results next.

5.4 Live Experiment for Hypothesis Testing

Goal and Potential Outcomes The goal of the pilot is to understand the effect of *difference in promised versus actual delivery gap* on RTO decisions for fast deliveries. Recall that **H2** hypothesizes that for fast deliveries, *beating* customer promise by a larger margin leads to a

reduction in product RTO. **H2** is particularly important for fast delivery regions where deliveries are already expedited. Hence, reduction in RTO orders has to be driven by other measures, such as customer promise. We also want to further check the relation between the FDG and RTO, to test **H1**.

Experimental Design: We conduct our experiment over a 3-week period and consider all orders originating in the Bengaluru metropolis region, which was selected for our experiment based on Mnytra's (i) significant customer base in this region (see §5.7 for details), which ensures enough data collection; (ii) relatively smaller geographical region, which ensures customer homogeneity; and (iii) very strong fulfillment network in the region, which ensures potentially fast deliveries of orders. Finally, because Myntra's central office is in Bengaluru, this selection ensured easier implementation of the pilot and subsequent data collection.

The Bengaluru metropolis region consists of 98 zip codes. Orders originating from this region are fulfilled through a network of 14 delivery centers. Fulfillment happens on a first come, first serve basis where order preference is based on the *promise* date made to the customer at the time of order placement. For example, if order A has a 1-day delivery promise and order B has a 2-day delivery promise, then order A is preferred over order B for fulfillment. Recall that we are interested in randomizing the difference between actual versus promised delivery. We can accomplish this by randomizing delivery promises for all orders with a fixed 1-day FDG. The current design of the retailer enforces that on a given day, the customer promise for all orders originating from a zip code remains fixed. This design is driven by operational considerations. Particularly, customer promises are manually selected on a daily basis at the zip code level. Hence, changing promises for every order is infeasible. Instead, we select zip code as our unit of randomization. On all weekdays (Monday through Friday) during the experiment, random delivery promises (between 2 and 4 days) are made to the customers. Randomization occurs across weeks over all zip codes. That is, we randomly select an ordering of numbers 2-3-4 for every zip code and every weekday. This becomes the random customer promise sequence on a given day over the 3-week period.

In summary, the experiment affected a total of 65,187 product shipments across 98 zip codes. 30.68% (20,001) were COD, which had an overall RTO rate of 6.72%. The RTO rate of online orders was 0.48%, considerably below the RTO rate for COD orders, showing the strong influence that payment type has on RTO decisions. In the subsequent analysis, we subset the data to include COD orders.

Out of 20,001 COD orders, 32.5% of orders received a promised delivery of 2 days, 34.9% promised delivery of 3 days and 32.5% received a promised delivery of 4 days. The difference in the size of the randomized groups is due to the random number of orders made on different days in different zip codes. We also find that the retailer's supply chain network in this region is very efficient: 84.6% of the orders were delivered within a day of the order placement, and less than 1% of orders had an FDG of more than 4 days.

Results: To test **H2**, we look at all orders that were attempted to be delivered 1 day after order placement, which constitutes 15,765 COD orders (78.8% of all COD orders). Among these, 32.5% of orders had a promised delivery of 2 days, 35% of orders had a promised delivery of 3 days, and 32.5% of orders had a promised delivery of 4 days. This results in three groups with randomly assigned actual versus promised delivery differences (-1, -2, and -3 days). The RTO rate for orders with a -1 day difference was 6.6% versus 6.76% for orders with -2 day difference and 5.51% for orders with a -3 day difference. Performing Pearson Chi-squared test (Agresti 2018) to compare the mean RTO rate of different randomized groups, we find that the RTO rates from the three groups are different at the 95% confidence level. Hence, delivery promises do have an effect on product RTO decisions.

Nevertheless, the above analysis does not account for product-level or customer-level heterogeneity. More specifically, because the randomization happened at the zip code level, different zip codes can be biased based on the kind of products being ordered. This heterogeneity in products or customers can drive RTO, instead of the difference in actual versus promised delivery. To assuage these concerns, we perform a regression analysis controlling for various product and user features.

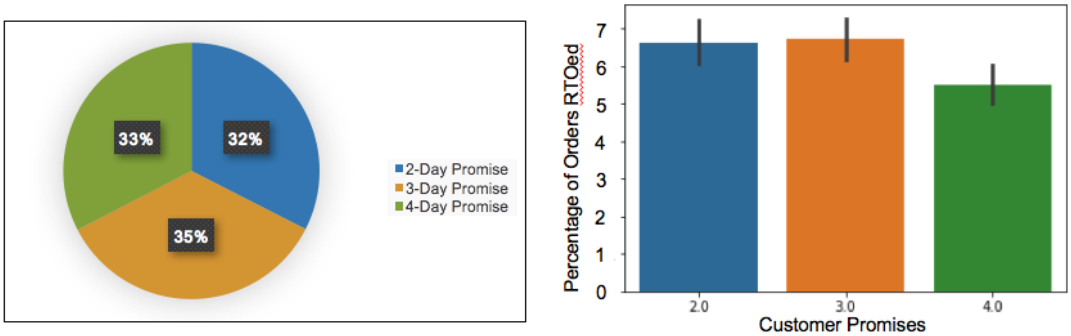

Figure 5.2: Summary statistics from the RCT. On left, we plot the percentage of total orders in different treatment groups. The different treatment groups are On right, we plot the RTO percentages in different treatment groups. The RTO rate si significantly lower in the 4-day promise treatment group.

Regression Analysis: For any order i, let rto_i denote the dependent variable (whether order i

was a RTO). rto_i can be driven by two factors: (i) the customer associated with order i, and (ii) the product associated with order i. Controlling for these factors, we are interested in the effect of the difference in actual versus promised delivery gap on RTO decisions.

To control for customer heterogeneity, ideally we would like to estimate a customer-level fixed effect model. Nevertheless, such an estimation would require multiple orders from the same customer during the experimentation phase, which is a rarity, especially in the fashion retail setting. Therefore, we work under the assumption that accounting for gender, customers within the same zip code are homogeneous. This is modeled by considering a zip code level fixed effect model with gender level indicators at the order level. Product heterogeneity is controlled by accounting for the master category and article type of the product, the price of the product, and the discount offered for the product on that order. Because the orders included in the experiment were all serviced through Myntra's in-house delivery personnel, delivery quality is homogeneous across orders and is absorbed in the other fixed effect variables. We do not need to control for the FDG since the analysis is restricted to orders that had a realized delivery gap of 1. Hence,

$$rto_i = \beta_{-2} \cdot \mathbb{1}\{APD_i^- = -2\} + \beta_{-3} \cdot \mathbb{1}\{APD_i^- = -3\} + \gamma \cdot price_i + \delta \cdot discount_i + z_i + at_i + g_i + dow_i + mc_i + \epsilon_i \,,$$

where order i belongs to zip code z_i, customer gender is denoted by g_i, dow_i is the day of the week of order i, mc_i is the master category of product i and at_i is the article type of product i. In addition, ϵ_i is the idiosyncratic noise term assumed to be independent. (We also consider a cluster-error model. The results and insights continue to remain consistent.)

We report the results of the OLS regression in Table 5.2. We find that the coefficient of APD^- is positive and significant at the 99% significance level. Hence, controlling for product-level and customer-level heterogeneity, an increase in APD^- causes an increase in the RTO chances of the order. Particularly, promising more robust delivery dates can lead to a reduction in the RTO rate. For example, if Myntra is certain that it can attempt deliveries within 1 day of order placement, it is more beneficial to promise a 3-day delivery than a 1-day delivery promise. This would ensure that the difference between actual and promised delivery is higher, which can lead to a reduction in RTO orders. Running the same regression with a logistic specification shows that the coefficient of APD^- continues to remain significant at the 95% level.

One concern with the analysis above is that zip code and gender-level effects might not account for all the customer-level heterogeneity. To assuage these concerns, we run another panel analysis based on transactions from repeat customers; that is, we subset the data to include only orders that were placed by customers who had made purchases on Myntra prior to the start of the RCT. We control for customer-level heterogeneity by accounting for the total number of orders, the overall discount availed, and the average price paid prior to the start of the RCT for each customer. We find that the coefficient of APD^- is again positive (0.005) and significant at the 90% confidence level, providing further evidence for the claim. The results of this analysis are presented in Table D.3 of Appendix D.2.

Variable	**Point Estimate**	**Standard Error**
$\mathbb{1}\{APD^- = -2\}$	0.013**	0.002
$\mathbb{1}\{APD^- = -3\}$	0.014**	0.002
price	+0.000***	0.000
product discount	−0.000	0.000
Observations	15,748	

$^*p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$

Table 5.2: Regression results from the RCT. The coefficient of APD^- is significant and positive showing that it is better to overshoot promise and beat it by a wider margin for RTO reduction.

Effect of the FDG: Although we do not directly randomize FDGs, they are driven by customer promises, as noted before; therefore, randomizing customer promises leads to a randomization of delivery gaps. Because orders are fulfilled in a first in, first out sequence based on customer promises, panel analysis based on the RCT data can provide further evidence on the increasing effect that the FDG can have on RTO decisions. As before, we perform a panel analysis with the model specification of

$$rto_i = \alpha FDG_i + \beta^+ APD_i^+ + \beta^- APD_i^- + \gamma \cdot price_i + \delta \cdot discount_i + z_i + at_i + g_i + dow_i + mc_i + \epsilon_i \,,$$

where the different controls are defined as before. In Table D.4 of Appendix D.2, we present the results of the panel analysis. Consistent with our hypothesis, we again find evidence that an increase in the FDG leads to an increase in RTO. Particularly, the coefficient of the FDG is 0.010 which is significant at the 99% level. Furthermore, consistent with the results of the above analysis, we find the coefficient of APD^- to be positive and significant at the 99% significance level. Somewhat surprisingly, the coefficient of APD^+ is negative and insignificant. However, note that APD^+ is the actual vs promised difference for all orders where the promise of delivery was not met. These orders constitute less than 1.5% of the overall orders. Therefore, the coefficient estimates turn out to be insignificant.

We end this section by remarking on two interesting outcomes from the RCT that relate to the overall variability in RTO rates across treatment groups and the impact of increasing promises on overall orders.

- **Insignificant change in RTO due to changing customer promise from 2-days to 3-days:** While we have seen that an increase in actual versus promised delivery difference from -1 to -3 days results in a reduction in RTO orders, there is no significant decrease when this difference goes from -1 to -2 days. One potential reason cold be that customers have a threshold for *exceptional* service (i.e. beating delivery promise). Furthermore, the positive effects of improved services only start to play a significant role above this threshold. For example, in Bengaluru's case, this threshold could be *beating* delivery promises by more than 2 days. In this case, RTOs would start to decrease only when the retailer beats the promised delivery date by more than 2 days. While a more detailed analysis of how customers form this service threshold is out of the scope of this chapter, it opens interesting new directions for future research.

- **Effect of increase in delivery promise on total orders:** As discussed before, we find that the total number of RTO orders decrease as we *beat* customer promise by a wider margin (4-day promise treatment group). But one potential concern could be that this change could be due to a reduction in the overall orders on the e-retailer's platform due to customers ordering from other platforms that promise faster deliveries. Nevertheless, in what follows, we show empirical evidence of no such effect.

  In particular, our RCT design ensures that for each zipcode, we are able to observe the total number of orders placed in all the three treatment group (2-3-4 days promise). Hence, in Figure 5.3 on the left, we plot the average orders placed from different zipcodes across different treatment groups. We also plot the standard errors of the total orders to check if the total orders decrease from a zipcode, as we increase the customer promise. As is evident, we find no evidence of such a change. In particular, the 90% confidence intervals around average orders overlaps across different treatments, showing that there is no statistical difference in total orders as we increase customer promises from 2 to 4.

  Our RCT design ensures that we can also perform a day level analysis of total orders across different zipcodes. In Figure 5.3, on the right, we plot the average orders as we vary the week of the day and the treatment (customer promise). Interestingly, we find that except

for Friday, total orders across different treatment groups remain similar. One potential explanation for such a difference could be that customers start to notice and account for promises if products are not promised to be delivered by the weekend. Nevertheless, a more detailed analysis of this phenomenon is out of the scope of this chapter.

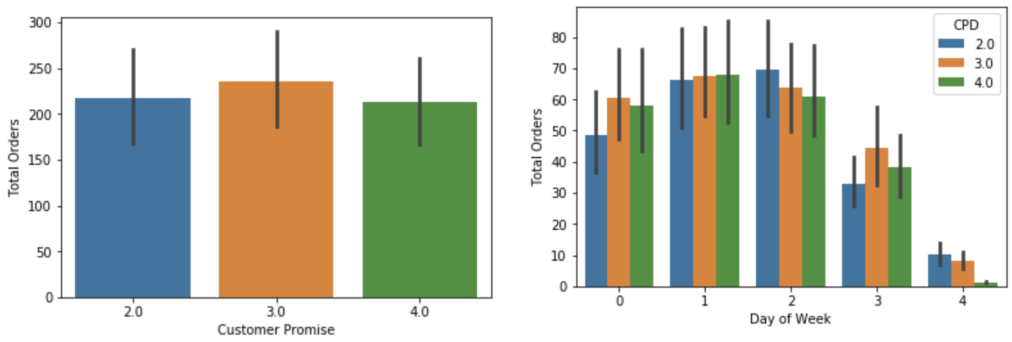

Figure 5.3: Average orders across different zipcodes under different treatment groups. On left, we plot average orders from different zipcodes as we increase customer promises. On right, we plot the average orders on a day (Monday to Friday) as we change customer promises. There is no statistically significant decrease in orders due to an increase in customer promise, except for Fridays.

5.5 Managerial Insights

In this section, we present the managerial insights we gained from the analysis performed above.

- *Expediting deliveries can lead to significant cost savings due to RTO reduction:* We find that reduction in the order delivery gap leads to a significant reduction in RTO orders. Particularly, we estimate that a 2-day reduction in the FDG of order shipments from its current average of 4.66 results in a 1.5% reduction in the probability of product RTO. For Myntra, which ships 150,000 products per day, this translates into 2,250 fewer orders being RTOed. Singh (2015) states that the average cost of each shipment in India is around Rs. 67.5. Hence, we conservatively estimate that Myntra would reap cost savings of as much as $1.5 million from cost savings due to a reduction in RTO orders. Given the robustness of the effect, we expect that the positive effects of RTO are not restricted to Myntra but extend to other e-retailers.

- *Beating customer promise is better than meeting customer promise at locations with fast order fulfillment:* For regions where delivery gaps are minimal (1-day delivery gaps), we find that *beating* customer promise can act as a proxy to faster deliveries for RTO reduction. We recommend that retailers should make 2- or 3- day promise, while still

fulfilling orders in 1 day. This would lead to a positive effect of *beating* customer promise which could lead to a reduction in product RTO.

- *Expediting deliveries has a heterogeneous effect on RTO:* While an increase in the FDG leads to an increase in product RTO, there is heterogeneity in the effect based on the product and the discount offered. Particularly, we find that the coefficient of the FDG for casual shoes is 0.008 as compared to 0.006 for sports shoes. Although it is significant at the 99% significance level, the difference in the magnitude of the effect shows that customers are less likely to RTO casual shoes for the same improvement in delivery gap, as compared to sports shoes. Running the panel analysis with the FDG and discount interaction terms, we find that the coefficient of the interaction term is negative and significant at the 99% significance level.

- *E-retailers should promote online payments for RTO reduction:* Product RTO is significantly lower for orders with online payment versus COD orders. This is in line with Bandi et al. (2017) who also find that product returns are significantly higher when a customer decides to use the COD payment method. Therefore, one effective way of reducing product RTO could be to promote and incentivize the usage of online payment methods (see Appendix D.3 for details on the analysis of the heterogeneity in COD usage).

In summary, we have seen that while the effect of the FDG on RTO is consistent, it is heterogeneous across orders. Furthermore, because delivery improvements come at a considerable cost in complex supply chains, we could use this heterogeneity to prioritize deliveries in order to reduce RTOs. We discuss the details of this optimization problem next.

5.6 Optimizing Deliveries: A Joint Strategic and Tactical Decision

We have so far established that reducing delivery gaps can lead to a reduction in RTO orders, which can lead to substantial cost savings for retailers. Nevertheless, considerable costs are associated with expediting deliveries. In this section, we take both a system-level view and a tactical operational view to optimize delivery gaps to reduce returns. Our modeling strategy is motivated by two important decisions that supply chain managers have to make repeatedly.

- *Strategic delivery threshold:* Supply chain managers have to decide on a delivery threshold

target for the coming month. These targets are usually region specific and are tracked by the retailer. Choosing a threshold for the overall supply chain is a strategic decision. A fast delivery gap target, relative to the current practice of the retailer, could result in substantial delivery improvement costs. However, not pushing for faster deliveries could instead lead to high RTO costs. Furthermore, a considerable uncertainty is related to realized costs because the threshold decision is made well in advance of the actual order arrivals, which can change from day to day. Therefore, managers need to take a system-level view of the overall supply chain to make threshold delivery target decisions.

- *Tactical delivery expediting:* In addition, when an order arrives, supply chain managers have to decide how to ship the order among the various available transportation options. More specifically, they need to decide if there is a need to *expedite* the order to minimize the RTO and delivery expediting costs. These decisions are order based and hence tactical. Furthermore, managers operate under delivery improvement budgets and multiple product orders, meaning they need to decide on how to allocate their budget among the available orders.

Indeed, both of these problems are connected. For example, if the overall supply chain delivery threshold is very low, all the delivery improvement budget would be used to comply with the delivery threshold provided. In contrast, if the delivery threshold is not changed, important system-level service quality parameters, such as the average delivery gap, would be high and could affect customer satisfaction on the retailer's platform. In what follows,we make this connection precise and formal.

Consider a retailer who is deciding on a system-level delivery threshold (y^*) such that all orders will be attempted to be delivered within this threshold. Orders delivered through the retailer's current supply chain have a delivery gap of Z days, where $Z \sim f$ is a known probability density function that can be estimated from the retailer's data. In addition, let $F(x) = \mathbb{P}(Z \leq x)$ denote the cumulative distribution function (CDF) of Z, and let $\mathbb{E}[Z] = \mu$ denote the current mean delivery gap. Delivery gap from the supply chain is modeled as a random variable because of the randomness in delivery locations, which are unknown in advance. However, prior data can be used to estimate the distribution of order locations (f) that guide the distribution of delivery gaps.

The retailer also has the option of delivering orders through an outside supplier, $\bar{S}(y)$ (e.g.

Fedex, DHL etc.), which would deliver orders with a delivery gap of $\bar{Z}(y)$ (assumed to be random) with $\mathbb{E}[\bar{Z}(y)] = y$. Expediting deliveries using outside suppliers would involve delivery improvement costs (DIC). Let $c_{DIC}(w) : \mathbb{R} \to \mathbb{R}$ denote the DIC of expediting a delivery by w days.

Finally, expediting deliveries would have an effect on the customer RTO decision. Particularly, let $R(x) \in \{0, 1\}$ denote the RTO decision of the customer if the delivery gap of the current order is x days. Each RTO order costs the retailer c_{RTO} in delivery and reverse logistics costs. A customer's RTO decision is modeled as a random variable with $\mathbb{E}[R(x)] = r(x)$. The expected RTO rate function, $r(x) : \mathbb{R} \to [0, 1]$, can be estimated using historical supply chain data (Figure 5.1).

Consider an order arrival that happens in the future, and let y be any chosen delivery threshold target. In addition, let $\mathrm{I}(y)$ denote whether the order was assigned to an outside option. $\mathrm{I}(y)$ is random because if the retailer can fulfill the delivery within y days using the current supply chain ($Z < y$), there is no need to use an outside option for the delivery. Hence, for any delivery threshold, y, the delivery gap is given by

$$\tilde{Z}(y) = \begin{cases} Z, & \text{if } \mathrm{I}(y) = 0, \\ \bar{Z}(y), & \text{otherwise,}. \end{cases} \tag{5.2}$$

and the combined RTO and delivery costs are given by

$$\mathcal{C}(y) = c_{DIC}(Z - y)\mathrm{I}(y) + c_{RTO}\mathrm{R}\left(\tilde{Z}(y)\right) . \tag{5.3}$$

The model above takes a system-level view of the supply chain. In contrast, the tactical decision of expediting deliveries is a more immediate decision. Managers have to decide which orders to expedite and by how much in order to meet the strategic threshold levels while minimizing RTO rates. Hence, letting $i = 1, .., n$ be the orders to be delivered tomorrow, z_i (sampled from f) be the delivery times of these orders (had they been delivered from S), and B be the delivery expediting budget, the bi-level problem of jointly selecting a critical supply chain threshold and

deciding expediting levels, w_i, is given by

$$\text{Min}_{w_1,...,w_n} \sum_{i=1}^{n} \mathbb{E}[R_i(\bar{Z}(z_i - w_i))]\,, \tag{5.4a}$$

$$\text{s.t} \sum_{i=1}^{n} c_{DIC}(w_i) \leq B, \tag{5.4b}$$

$$0 \leq z_i - y^* \leq w_i,\ \forall i = 1, ..., n,\ \text{and} \tag{5.4c}$$

$$y^* = \arg\min_y \mathbb{E}\left[\mathcal{C}(y)\right]\,. \tag{5.4d}$$

The upper-level problem (5.4a) is that of *minimizing* the total expected RTO orders from the most recent orders under the budget constraint (5.4b) and the supply chain threshold constraint (5.4c). The threshold constraint is determined by the lower-level problem of minimizing the strategic expected RTO costs and DIC (5.4d). Notice that a very low y^* could make the upper-level problem infeasible; therefore, the two problems have to be solved jointly. Nevertheless, both the upper-level and the lower-level problems are independently of interest to retailers. For example, the solution from the lower level problem can be part of the retailer's service compliance plans whereas the upper-level problem can guide managers to accomplish these service-level targets with optimal budget utilization. In what follows, we describe solution strategies for each of the problems independently. We come back to the joint problem at the end of the section.

Assumptions: For the remainder of this section, we assume that $r(.)$, the expected return rate function, is linear in x. That is, we let $r(x) = rto_{min} + \beta x$, where rto_{min} represents the baseline RTO rate if orders are shipped with 0 days of the delivery gap. The baseline RTO rate captures the effect of price, discount, product type, and other order-level features that have an effect on RTO. β defines the rate of RTO change due to a change in the FDG. This functional form is also validated by our empirical analysis, where we show that RTO rates increase almost linearly with increasing delivery gaps (Figure 5.1). We also assume the DIC function ($c_{DIC}(.)$) to be piecewise constant. Particularly, the cost function has k different pieces, defined by $k+1$ end points, $d_i, i = 1..., k+1$. Hence, the cost associated with an improvement of $d^k < w_i \leq d^{k+1}$ would be $\bar{C}_k$. The piecewise constant assumption is again motivated from practice: delivery improvement of up to a day can be accomplished using ground transport, but 2-day improvements could lead to the use of air transport, in which case, the cost function

would be piecewise constant. Finally, we also let f, the empirical delivery gap distribution, to be exponential. This assumption is also motivated from discussions with the industry partner and empirically justified (see Figure D.2 in Appendix D.4 and §5.7).

5.6.1 Lower-Level Optimal Delivery Thresholding Problem

We define the Optimal Delivery Thresholding Problem (ODTP) as the lower-level problem. In this problem, the goal is to select a strategic delivery threshold by minimizing expected delivery and RTO costs (5.4d) as

$$\text{Min}_{y \geq 0}\ g(y), \tag{ODTP}$$

where $g(y) = \mathbb{E}\left[\mathcal{C}(y)\right]$, is given by (5.3). For ease of exposition and simplicity of notation, we start by letting $c_{DIC}(y)$ be constant for all y and refer to it as $\bar{C}_{DIC}$. We also assume WLOG that rto_{min} is 0. We come back to the more general case of piecewise constant costs at the end of the section. The objective function of ODTP simplifies to

$$\begin{aligned}
\mathbb{E}\left[\mathcal{C}(y)\right] &= c_{DIC}\left(1 - F(y)\right) + c_{RTO}\left(\mathrm{r}\left(y + F(y)\left(\mathbb{E}[Z|Z < y] - y\right)\right)\right) \\
&= \bar{C}_{DIC} e^{\frac{-y}{\mu}} + c_{RTO}\beta\left(y + \left(1 - e^{\frac{-y}{\mu}}\right)\left(\mathbb{E}[Z|Z < y] - y\right)\right) \\
&= \bar{C}_{DIC} e^{\frac{-y}{\mu}} + \mu c_{RTO}\beta e^{-\frac{2y}{\mu}}\left(1 - 2e^{\frac{y}{\mu}} + e^{\frac{2y}{\mu}} + \frac{y}{\mu}\right).
\end{aligned}$$

The objective of ODTP is to *minimize* the total costs by choosing an optimal threshold for choosing an outside option for expedited deliveries. Solving the problem over $y > \mu$ would lead to an increase in RTO costs with no change in DIC. Hence, we consider $y \leq \mu$. If the objective function of ODTP is convex or unimodal, the problem can be solved using the first-order conditions for the optimal threshold, y^*. Unfortunately, one can show, by generating simple counter examples, that the objective is not well behaved. That is, it is neither a concave nor a convex function. Furthermore, the objective function of ODTP is not even unimodal in general (Figure D.3 in Appendix D.4). Nevertheless, in what follows, we characterize the optimal solution of the ODTP.

Theorem 5.6.1. Let y^* be the optimal delivery threshold of the retailer. Then,

- If $\frac{\bar{C}_{DIC}}{c_{RTO}} > 2\beta\mu$,

– the objective function of ODTP is strictly convex, and y^* is the unique solution to:

$$c_{RTO}\beta e^{\frac{-2y}{\mu}}\left(2e^{\frac{y}{\mu}}-2\frac{y}{\mu}-1\right)-\frac{\bar{C}_{DIC}e^{\frac{-y}{\mu}}}{\mu}=0, \tag{5.5}$$

– y^* decreases with β and c_{RTO} and increases with $\bar{C}_{DIC}$ and μ.

- If $\frac{\bar{C}_{DIC}}{c_{RTO}} < \left(1-\frac{2}{e}\right)2\beta\mu$, then the objective function of ODTP is concave, and the optimal solution is $y^*=0$.

- If $2\beta\mu \geq \frac{\bar{C}_{DIC}}{c_{RTO}} \geq \left(1-\frac{2}{e}\right)2\beta\mu$, then the objective function of ODTP is neither concave nor convex, and

$$y^* = \arg\min\left\{g(0), g(z^*\mu), g(\bar{y})\right\},$$

where z^* is the solution to

$$\frac{e^z}{z} = \frac{2c_{RTO}\beta - \frac{\bar{C}_{DIC}}{\mu}}{4c_{RTO}\mu\beta}, z \in [0,1],$$

and $\bar{y}$ is the solution of (5.5) between $[z^*\mu, \mu]$.

We relegate the proof of Theorem 5.6.1 to Appendix D.1 but discuss the intuition of the proof next.

To characterize the optimal solution, we first characterize the convexity (concavity) of the objective function. Whenever the ratio of the DIC relative to the RTO costs is sufficiently large (namely, higher than twice the mean RTO costs, $2\beta\mu$), the objective function is convex. Intuitively, for very small values of the threshold (y), the DICs dominates over the RTO costs. As y increases, the RTO costs start to dominate. The optimal solution in this case is determined through the first-order conditions (5.5). We also characterize how the optimal threshold (y^*) changes as other problem parameters change. As the RTO cost or the slope of the RTO function increases, the optimal y^* decreases due to increased RTO costs, which start to dominate DICs. Similarly, as the current mean delivery gap of the supply chain (μ) increases, y^* again increases because there is a higher percentage of orders above any chosen threshold.

When the ratio of the DICs and RTO costs is not sufficiently high (lower than $\left(1-\frac{2}{e}\right)2\beta\mu$), the objective cost function can be shown to be concave. In this region, the DIC is so low that the RTO costs always dominate in the ODTP objective cost function. The optimal solution, as a result, is to choose the lowest threshold (0) and reduce the RTO rate to 0 to incur minimum

RTO costs.

Finally, when neither of the conditions discussed above hold, the function is neither concave nor convex. Furthermore, z^* (defined in Theorem 1) defines the convexity of the objective function. The optimal solution in this case is obtained from comparing the boundary points (0, $z^*\mu$) and $\bar{y}$. We note that in practice, retailers fall into the case where DICs are high, that is, the case where the objective is convex or both concave and convex (see §5.7).

Note that in the analysis of this subsection, we investigated the case of constant DICs. Nevertheless, the case of piecewise constant DIC also follows in a similar fashion. For example, consider the case when c_{DIC} consists of two pieces. We can solve two ODEP subproblem corresponding to the DICs of the two pieces. A comparison of the optimal costs from the two subproblems would result in the overall optimal threshold. We omit the analysis for the sake of brevity.

5.6.2 Upper Level Tactical Delivery Expediting Problem

Next, we consider the upper level ODEP (5.4a), which operationalizes the strategically chosen delivery threshold target, y^*, under budgetary constraints:

$$\begin{aligned} \text{Min}_{w_1,\ldots,w_n} \ & \sum_{i=1}^{n} \mathbb{E}[R_i(\bar{Z}(z_i - w_i))] \\ \text{s.t} \ & \sum_{i=1}^{n} c_{DIC}(w_i) \leq B, \\ & 0 \leq z_i - y^* \leq w_i, \ \forall i = 1, \ldots, n. \end{aligned} \tag{ODEP}$$

Recall that $R_i(\bar{Z}(.))$ denotes the random variable that models a customer's RTO decision, while $\bar{Z}(w_i)$ denotes the random delivery gap when the retailer expedites the delivery of order i by w_i days. The ODEP objective can be reformulated as

$$\text{Min}_{w_1,\ldots,w_n} \sum_{i=1}^{i=n} \mathbb{E}[R_i(\bar{Z}(z_i - w_i))] = \text{Min}_{w_1,\ldots,w_n} \sum_{i=1}^{i=n} r_i(z_i - w_i)\,. \tag{ODEP}$$

The reformulation above follows from Jensen's inequality and the fact that r is an affine function. Furthermore, because the list of orders is known, we now incorporate the order-level heterogeneity in the RTO-FDG relation by letting the RTO function $r_i(x) = rto^i_{min} - \beta_i \cdot x, \forall i = 1, \ldots, n$.

A priori, it is hard to characterize the optimal solution of the ODEP with piecewise constant costs because a small shift in the amount of improvement in delivery speed, can result in a

substantial change in the overall cost. Nevertheless, in Proposition 5.6.2, we reformulate the ODEP with piecewise constant costs as a mixed integer optimization problem. We start by taking a continuous approximation of the cost function and assume that the cost increases linearly between d_i and $d_{i+1} + \epsilon$ for a very small ϵ. This ensures that we have a continuous DIC function (Figure D.4 in Appendix D.4). With a slight abuse of notation, we will assume that the DIC function is continuous for all d_j, $j = 1..k+1$. Finally, we assume that the rate of change of the DICs at different constant levels is at least 1; That is, the higher the improvement level, the more it will cost to further expedite deliveries. Finally, we assume that the budget, B, is high enough so that a feasible solution to the problem exists.

Proposition 5.6.2. The ODEP with piecewise constant DIC can be reformulated as the following mixed-integer optimization problem:

$$\text{Max}_{w,\lambda,\tau} \quad \sum_{i=1}^{i=n} \beta_i w_i \tag{5.6a}$$

$$\text{s.t} \quad \sum_{i=1}^{i=n} \sum_{j=1}^{j=k} \bar{C}_i \lambda_j^i \leq B, \tag{5.6b}$$

$$\sum_{j=1}^{j=k} \lambda_i^j d_i^j = w_i, \ \forall i = 1, .., n \tag{5.6c}$$

$$0 \leq z_i - y^* \leq w_i, \ \forall i = 1, ..., n. \tag{5.6d}$$

$$\lambda_i^1 \leq \tau_i^1, \ \forall i = 1, .., n \tag{5.6e}$$

$$\lambda_i^j \leq \tau_i^{j-1} + \tau_i^j, \ \forall i = 1, .., n, \ j = 2..., k \tag{5.6f}$$

$$\lambda_i^{k+1} \leq \tau_i^k, \ \forall i = 1, .., n \tag{5.6g}$$

$$\sum_{j=1}^{j=k+1} \lambda_j^i = 1, \ \forall i = 1, .., n \tag{5.6h}$$

$$\sum_{j=1}^{j=k} \tau_j^i = 1, \ \forall i = 1, .., n \tag{5.6i}$$

$$\tau_i^j \in \{0, 1\}, \ \forall i = 1, .., n, \ j = 1..., k \tag{5.6j}$$

$$\lambda_i^j \geq 0, \ \forall i = 1, .., n, \ j = 1..., k+1 \tag{5.6k}$$

We relegate the proof of Proposition 5.6.2 to Appendix D.1. The reformulation uses similar ideas as in Vielma et al. (2010). Because the RTO rate is increasing in the FDG (decreasing

in delivery improvement), the objective can be reformulated as *maximizing* the RTO rate reduction over the n orders. To model the piecewise constant cost structure of DIC, we introduce two sets of new variables τ_j^i and λ_j^i, where τ_j^i characterizes where the selected expediting level, w_i, lies, and λ_j^i represents w_i as the convex combination of the end points of the region where it lies. This convex combination of the end points is then used to effectively model the budget constraint associated with DICs.

The number of variables increases linearly with the number of orders in the above formulation. Hence, the IP solution can be computationally expensive to obtain, especially when the order size is very high. In what follows, we consider the LP relaxation of the ODEP-IP problem and construct a feasible ODEP solution. We show that the optimality gap of the solution constructed from the LP relaxation is bounded from above by the maximum rate of change of RTO (β_i) and is independent of n. Notice that in the process, we construct a polynomial time algorithm to find the ODEP solution with a very small optimality gap.

Theorem 5.6.3. Consider the LP relaxation of ODEP-IP (ODEP-LP) where z_j^i are relaxed to be nonnegative continuous variables, and let $(\bar{w}, \bar{\lambda}, \bar{\tau})$ be the optimal solution of ODEP-LP. Then, there exists at most one order i (i_{NI}) such that $\bar{\tau}_i^j$ is nonintegral for $j = 1, ...k$.

Let $w_{feasible} = \arg\max_{j=1...k} \{\bar{C}_j \leq B - \sum_{i=1,..n, i\neq NI} \sum_{j=1,..,k+1} \bar{C}_j \bar{\lambda}_i^j\}$. Consider $(\tilde{w}, \tilde{\lambda}, \tilde{\tau})$ such that

1. $(\tilde{w}, \tilde{\lambda}, \tilde{\tau}) = (\bar{w}, \bar{\lambda}, \bar{\tau}), \forall i \neq i_{NI}$;
2. $\tilde{\tau}_{i_{NI}}^{w_{feasible}} = 1, \lambda_{i_{NI}}^{w_{feasible}} = 1,\ w_{i_{NI}} = d_{w_{feasible}}$; and
3. $\tilde{\tau}_{i_{NI}}^{j} = 0, \lambda_{i_{NI}}^{j} = 0,\ \forall j \neq w_{feasible}$.

Then $(\tilde{w}, \tilde{\lambda}, \tilde{\tau})$ is a feasible solution to ODEP-IP and the optimality gap of $(\tilde{w}, \tilde{\lambda}, \tilde{\tau})$ with optimal solution $(w^{opt}, \lambda^{opt}, \tau^{opt})$ is

$$\textstyle\sum_{i=1}^{i=n} \beta_i w_i^{opt} - \sum_{i=1}^{i=n} \beta_i \tilde{w}_i \leq \bar{w}_{IN}\beta_{IN} \leq \ \max_{i=1,..n} \bar{w}_{IN}\beta_i \leq 1\,.$$

We relegate the proof of Theorem 5.6.3 to Appendix D.1. The heuristic solution constructed in Theorem 5.6.3 differs from the LP solution only for a single order. This allows us to characterize the optimality gap of the heuristic solution relative to the optimal IP solution. First, we show feasibility of the constructed solution. We then show that the objective of the heuristic solution is "close" to the optimal LP objective, which, in turn, bounds the optimality gap relative

to the optimal IP solution. Note that because the optimality gap is independent of n, it also implies that as the number of orders increases, the objective value of the heuristic converges to the IP optimal value.

Solving the joint problem: Now that we have discussed the solutions of both the upper-level and the lower-level problems, we come back to the joint bi-level problem of deciding the threshold gap as well as the tactical expediting levels. Given an optimal threshold (y^*), the upper-level solution might turn out to be infeasible due to limited budgets. In case of infeasibility, the space of possible thresholds (y) can be updated, and the lower-level problem can be resolved in the restricted region to get an updated y^* which then can be used in the upper-level problem to get optimal tactical decisions. We note that operationally, feasibility can be also achieved by increasing the budget to ensure that all orders are delivered within the selected threshold, which could lead to optimal solutions for both subproblems jointly.

Remark 5.6.4. The constructed IP for the upper-level problem assumes that all orders need to meet the strategic threshold, y^*. Another relaxed version of the problem could be that the mean delivery time of order, on any given day, cannot be more that the chosen strategic threshold, y^*. The IP formulation presented in Proposition 5.6.2 can be easily updated to account for such a constraint or order-specific constraints that are important in practice but we omitted here due to space limitations.

5.7 Impact in Practice

In this section, we detail the implications of the FDG-optimization framework of §5.7 on our industry collaborator. As we have discussed previously, our industry partner ships more than 150,000 orders every day across India. Because its supply chain decisions are based on geographic considerations, we focus our analysis on the Bengaluru metropolis region, which lies in the southern Indian state of Karnataka. This region consists of 98 zip codes, with a total population of 12.34 million and a geographical area of 741 sq kms. It also contributes substantially to the industry partner's business. From 2017-18, the e-retailer shipped more than 4 million items in eight master categories to more than 600,000 unique users in the region. These packages were served through a network of 14 DCs spread across the region. Recall that after an order is placed, the item goes through various nodes in the supply chain, eventually arriving at the last

mile DC. From there, the package is then taken to the customer's doorstep by last mile delivery personnel.

Figure D.2 in Appendix D.4 shows the percentage of orders with respect to their FDGs in the Bengaluru region. The average FDG for orders in this region is 1.886, which is considerably lower than the national average. This difference can be attributed to the proximity of this region to the Bengaluru warehouse and the relatively improved infrastructure of the region. Similarly, Figure D.5 in Appendix D.4 shows the overall RTO rate with respect to the FDG for footwear orders in this region. Note that an increase in FDG leads to an increase in RTO, replicating the general trend.

The objective of the retailer is to jointly (i) select a strategic delivery gap threshold that all orders should meet, and (ii) optimize budget allocations on different orders to minimize the chances of an RTO occurring on any given day. Because delivery improvement costs drive the trade-off between RTO costs and faster shipment, we discuss the associated costs in detail next.

The retailer takes 8 hours of initial processing time after an order is placed on the retailer's online platform. This involves initial QC packaging and bagging, after which the package is ready to be shipped from the warehouse to the last mile DC. Because a majority of the orders are delivered from the local warehouse, using air transportation to deliver products from different areas to the Bengaluru warehouse is not useful. For transportation between the warehouse and the last mile DC, the company uses trucks that run on a particular schedule two times per day. These trucks leave the warehouse at 6 AM and 12:01 PM every day. If a product gets ordered between midnight and 4 AM, then there is a chance that it will be delivered within the same day, thus resulting in 0 days FDG. If a product is ordered after 4 AM, then a delivery attempt is made at least 1 day after the order is placed.

To decrease the FDG of delivered orders, the retailer has the option to increase the frequency of trips between warehouse and the DC. Nevertheless, as noted above, the cost of expediting the process of delivery by w days does not increase linearly with w. For example, in the Bengaluru warehouse, one could employ more trucks to ensure that almost all the products reach the last mile DC within 1 day of order placement. But a 0 day FDG (2-day improvement from the current average delivery gap) would mean that the company must also hire additional warehouse employees and run the warehouse overnight at capacity. This has a substantial overhead cost and further implies that the DIC function is indeed piecewise constant. In Figure D.5 of Appendix D.4, we plot the DIC of the retailer for 1- and 2-day improvements. To reduce

the mean FDG to 1 day in Bengaluru, our industry partner has to employ more trucks. Each truck can transfer up to 500 kg of orders for a price of Rs. 1300. Because an average package weighs around 0.7 kg, this adds an extra cost of around Rs. 1.9 per order. Similarly, to ensure express delivery within the same day, the total cost rises to Rs. 100 per order. This includes all the overhead costs attached with the warehouse and extra workers.

Having described the DIC, we discuss the estimation of the empirical delivery gap distribution. Because a very small fraction of the total orders were served within a day of the order placement, we assume that the delivery time for deliveries within 1 day is uniformly distributed. We use a mixture of shifted exponential distributions to estimate the delivery gap of orders where the FDG is above 1 day (Figure D.2 in Appendix D.4). Note that a mixture of exponentials shows a very good fit to the order-delivery gap distribution. Finally, the expected RTO-FDG function can also be estimated using a linear fit (Figure D.5 of Appendix D.4). Having estimated all the problem parameters, now we discuss the optimal threshold and expediting decisions in the Bengaluru region.

The first-stage optimal ODTP threshold for Bengaluru is a 1 day delivery: the retailer should target to deliver all orders within 1-day of order placement. Difference in the DICs make deliveries faster than 1 day economically unviable for the retailer. In Figure D.6 of Appendix D.4, we also plot the objective function of the ODTP for the retailer. As noted in Theorem 5.6.1, the cost function is both concave and convex, depending on the DICs. An overall threshold target of 1-day delivery would imply an average delivery improvement of 0.86 days from the current average. We estimate that by expediting the mean FDG, the company can reduce the RTO rate by 13.30% from the current mean RTO rate of the region. Finally, to check the robustness of this improvement, we also perform a comparative analysis of the change in improvement as DIC changes and increase the cost to Rs 2.85 for a 1-day delivery improvement, 50% more than the previously considered DIC cost (Figure D.7 in Appendix D.4). An increase in the DIC leads to an increase in the optimal threshold (from 1 to 1.1) and a cost improvement of 1.05%. Although the improvement is smaller than before (from 2.7% to 1.05%), given the magnitude this is still a significant improvement.

Next, we discuss the second stage tactical problem of allocating a budget for expediting deliveries of selected orders. The tactical ODEP optimally assigns a delivery improvement budget over multiple orders. We consider all orders in the Bengaluru metropolis region on a randomly selected day (1,563 total orders). We first estimate the change in RTO due to a

change in the FDG at the master category, brand, and article type levels using OLS regression (430 total combinations). Then, we consider the optimal ODEP solution over the n orders with varying budget (B) and plot the percentage decrease in RTO orders due to expedited deliveries (Figure D.8 in Appendix D.4). We assume that the budget allocation is such that all deliveries satisfy the ODTP delivery threshold, and the budget can be used for further delivery improvements. Whereas the LP-based heuristic computes the optimal solution within seconds, the IP solver in Gurobi is considerably slower in solving . Note that the bound in Theorem 5.6.3 shows that the optimality gap of the LP-based heuristic is at most 0.30. Nevertheless, our computational study shows that the realized bound is indeed below 1.0 x 10^{-5}, establishing the near optimality of the heuristic. We further find that the RTO rate can be reduce by as much as 10% with optimal allocation of a reasonable budget (e.g., less than Rs. 2 per order). Hence, an optimal allocation of budget for expediting deliveries can lead to substantial changes in product RTOs.

5.7.1 Conclusions

Product returns pose a big challenge to online retailers around the world; Therefore, reducing returns is an important problem. Working with one of India's largest e-fashion retailers, we show the causal relation of delivery gaps on RTO, thereby identifying a supply chain lever to control returns. We also perform an RCT to analyze the effect of delivery promise on the RTO rates of fast delivery orders. We conservatively estimate that a 2-day reduction in average delivery time could lead to potential savings of as much as $1.5 million due to RTO costs reduction. We also introduce the joint strategic and tactical delivery optimization problems that carefully balance the reduction in RTO costs with DIC. Using our industry partner's data, we estimate that this improvement in delivery gaps can lead to a reduction of 2.7% by optimizing delivery times.

Chapter 6

Conclusions

Developing data-driven methods to drive operational decisions in retail management remains a major challenge. In this thesis we consider various decision problems that retailers face before and during a product is launched on the market, as well as after it is sold to the customer. This thesis presents formulations, algorithms and analysis of operational problems in retail management, that are developed in close collaboration with industry collaborators.

First, we start by discussing the problem of predicting demand for new products before they are launched on the market. We devise a joint clustering and regression method that jointly clusters existing products based on their features as well as sales patterns while estimating their demand. Analytically, we prove in-sample and out-of-sample prediction error guarantees in the LASSO regularized linear regression case to account for over-fitting due to high dimensional data. Numerically we perform an extensive comparative study on real world data sets to show that the proposed algorithm outperforms state-of-the-art prediction methods and improves the WMAPE forecasting metric between 20-60%.

Second, we consider the problem of making personalized product recommendations when customer preferences are unknown and the retailer risks losing customers because of irrelevant recommendations. We present empirical evidence of customer disengagement through real-world data from a major airline carrier who offers a sequence of ad campaigns. We formulate the problem as a user preference learning problem and show that this seemingly obvious phenomenon can cause almost all state-of-the-art learning algorithms to fail in this setting. We propose modifying bandit learning strategies by constraining the action space upfront using an integer optimization model. We prove that this modification allows us to keep significantly more customers on the platform. Numerical experiments on real movie recommendations data

demonstrate that our algorithm can improve customer engagement with the platform by up to 80%.

Third, we investigates the problem of pricing of new products for a retailer who does not have any information on the underlying demand for a product. The retailer also seeks to reduce the amount of price experimentation because of the potential costs associated with price changes. We construct a pricing algorithm and establish when the proposed policy achieves near-optimal rate of regret, $\tilde{\mathcal{O}}(\sqrt{T})$, while making $\mathcal{O}(\log \log T)$ price changes. Hence, we show considerable reduction in price changes from the previously known $\mathcal{O}(\log T)$ rate of price change guarantee in the literature.

Fourth, we focus on the problem of reducing product returns and investigate it through a supply chain lens. Closely working with one of India's largest online fashion retailers, we focus on identifying the effect of delivery gaps (total time that customers have to wait for the item to arrive) and customer promise dates on product Returns To Origin (RTO). Our empirical analysis reveals that an increase in delivery gaps causes a substantial increase in product RTO. Furthermore, we also perform a RCT in to estimate the effect of delivery promise on product returns. Based on the insights from this empirical analysis, we then develop an integer optimization model that mimics managers' decision-making process in selecting personalized delivery speed targets.

In summary, the thesis develops data-driven practical techniques, in close collaboration with industry practitioners, that can have substantial impact on retailer's bottom line.

Appendix A

Appendix of Chapter 2

A.1 Proofs of Section 2.4

Proof. Proof of Proposition 2.4.1. Consider problem (P_L) where q_{ikj} substitutes $z_{ik}\beta_{kj}$ and r_{kj} substitutes $|\beta_{kj}|$, namely,

$$\begin{aligned}\min_{z_{ik},\beta_{kj}} \quad & \sum_{i=1}^{n}\left(y_i - \sum_{k=1}^{\ell}\sum_{j=1}^{m} q_{ikj}x_{ij}\right)^2 + \lambda\sum_{k=1}^{\ell}\sum_{j=1}^{m} r_{kj} \\ \text{s.t.} \quad & \sum_{k=1}^{\ell} z_{ik} = 1, \quad i = 1,\dots,n \\ & z_{ik} \in \{0,1\}, \quad i = 1,\dots,n, \quad k = 1,\dots,\ell.\end{aligned}$$

This substituted problem is not identical to (P_L) as it does not specify the link between z_{ik}, β_{kj}, q_{ikj}, and r_{kj}. For this, we need to add constraints that define how q_{ikj} and r_{kj} depend on z_{ik} and β_{kj}. First, the following constraints defines that $q_{ikj} = \beta_{kj}$ when $z_{ik} = 1$,

$$-M(1 - z_{ik}) \leq q_{ikj} - \beta_{kj} \leq M(1 - z_{ik}), \quad i = 1,\dots,n, \quad k = 1,\dots,\ell, \quad j = 1,\dots,m.$$

When $z_{ik} = 1$, the inequality states that $0 \leq q_{ikj} - \beta_{kj} \leq 0$, and hence that $q_{ikj} = \beta_{kj}$. On the other hand, when $z_{ik} = 0$, the inequality leaves q_{ikj} and β_{kj} unconstrained. Second, the following constraint defines that $q_{ikj} = 0$ when $z_{ik} = 0$,

$$-Mz_{ik} \leq q_{ikj} \leq Mz_{ik}, \quad i = 1,\dots,n, \quad k = 1,\dots,\ell, \quad j = 1,\dots,m.$$

When $z_{ik} = 0$, the inequality becomes $0 \leq q_{ikj} \leq 0$, which means that $q_{ikj} = 0$. In the case

where $z_{ik} = 1$, this constraint is essentially eliminated. Thus, together these two constraints define that $q_{ikj} = z_{ik}\beta_{kj}$. Finally, to ensure that $r_{kj} = |\beta_{kj}|$ we add the following constraints,

$$r_{kj} \geq \beta_{kj}, \quad k = 1, \ldots, \ell, \quad j = 1, \ldots, m,$$
$$r_{kj} \geq -\beta_{kj}, \quad k = 1, \ldots, \ell, \quad j = 1, \ldots, m.$$

In the objective, λr_{kj} appears as an additive term and $\lambda \geq 0$. Hence, since we are minimizing the objective, the optimal value of r_{kj} is as small as allowed. In the case where $\beta_{kj} \geq 0$, the first constraint will set this minimal value to be β_{kj}, while in the case where $\beta_{kj} \leq 0$, the second constraint will mean it is $|\beta_{kj}|$. Therefore, the two constraints together define $r_{kj} = |\beta_{kj}|$. Hence, after adding these 4 sets of constraints to the substituted problem it forms (P_{LR}) and is equivalent to (P_L). □ □

Proof. Proof of Lemma 2.4.2. To bound this probability, we use the probability complement, the union bound, Hölders inequality, and the tail bound on standard normal random variables as follows,

$$\begin{aligned}
\mathbb{P}\left(\frac{1}{n}||\epsilon^T(Z * X)||_\infty \leq \frac{\lambda}{4}\right) &= \mathbb{P}\left(\max_{c=1,\ldots,m\ell} \left|\epsilon^T(Z * X)_c\right| \leq \frac{n\lambda}{4}\right) \\
&= 1 - \mathbb{P}\left(\max_{c=1,\ldots,m\ell} \left|\epsilon^T(Z * X)_c\right| > \frac{n\lambda}{4}\right) \\
&\geq 1 - \sum_{c=1}^{m\ell} \mathbb{P}\left(\left|\epsilon^T(Z * X)_c\right| > \frac{n\lambda}{4}\right) \\
&\geq 1 - \sum_{c=1}^{m\ell} \mathbb{P}\left(||\epsilon^T||_\infty||(Z * X)_c||_1 > \frac{n\lambda}{4}\right) \\
&\geq 1 - \sum_{c=1}^{m\ell} \mathbb{P}\left(\max_{i=1,\ldots,n} |\epsilon_i| > \frac{\sqrt{n}\lambda}{4}\right) \geq 1 - \sum_{c=1}^{m\ell}\sum_{i=1}^{n} \mathbb{P}\left(\left|\frac{\epsilon_i}{\sigma}\right| > \frac{\sqrt{n}\lambda}{4\sigma}\right) \\
&\geq 1 - \sum_{c=1}^{m\ell}\sum_{i=1}^{n} 2\exp\left(-\frac{1}{2}\left(\frac{\sqrt{n}\lambda}{4\sigma}\right)^2\right) = 1 - \delta.
\end{aligned}$$

In particular, in the third inequality, we use the fact that for $x \in \mathbb{R}^n$, $||x||_1 \leq \sqrt{n}||x||_2$, and hence, $||(Z * X)_c||_1 \leq \sqrt{n}||(Z * X)_c||_2 \leq \sqrt{n}$. The last inequality follows from the tail bound on standard normal random variables. □ □

Proof. Proof of Proposition 2.4.3: Recall, by definition that

$$\|(Z^\Delta * X)(\widehat{\beta} - \beta^*)\|_2 = \|(Z^\Delta * X)\beta^\Delta\|_2 .$$

Furthermore, $Z^{\Delta} = \hat{Z} - Z^*$, which implies

$$Z^{\Delta} = \begin{pmatrix} \widehat{z_{11}} & \widehat{z_{12}} & \dots & \widehat{z_{1\ell}} \\ \widehat{z_{21}} & \widehat{z_{22}} & \dots & \widehat{z_{2\ell}} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{z_{n1}} & \widehat{z_{n2}} & \dots & \widehat{z_{n\ell}} \end{pmatrix} - \begin{pmatrix} z^*_{11} & z^*_{12} & \dots & z^*_{1\ell} \\ z^*_{21} & z^*_{22} & \dots & z^*_{2\ell} \\ \vdots & \vdots & \ddots & \vdots \\ z^*_{n1} & z^*_{n2} & \dots & z^*_{n\ell} \end{pmatrix} = \begin{pmatrix} \widehat{z_{11}} - z^*_{11} & \widehat{z_{12}} - z^*_{12} & \dots & \widehat{z_{1\ell}} - z^*_{1\ell} \\ \widehat{z_{21}} - z^*_{21} & \widehat{z_{22}} - z^*_{22} & \dots & \widehat{z_{2\ell}} - z^*_{2\ell} \\ \vdots & \vdots & \ddots & \vdots \\ \widehat{z_{n1}} - z^*_{n1} & \widehat{z_{n1}} - z^*_{n2} & \dots & \widehat{z_{n\ell}} - z^*_{n\ell} \end{pmatrix}$$

Now assume without loss of generality that the first r points are incorrectly clustered. Then, for $i = 1, ...r$, $Z^{\Delta}_{i,c^*_i} = -1$ and $Z^{\Delta}_{i,mc_i} = 1$, where c^*_i denotes the true unknown cluster of point i and mc_i denotes the incorrect cluster to which point i was assigned. All other entries of the Z^{Δ} matrix are 0 by definition. Also note that Hence,

$$\begin{aligned}
\|(Z^{\Delta} * X)\beta^{\Delta}\|_2 &= \sqrt{\sum_{i=1}^{n} \left[\sum_{j=1}^{m} \sum_{k=1}^{\ell} \left(\widehat{z_{ik}} - z^*_{ik}\right) \left(x_{ij}\beta^{\Delta}_{kj}\right) \right]^2} \\
&\leq \sqrt{\sum_{i=1}^{r} \left[\sum_{j=1}^{m} \sum_{k=1}^{\ell} |x_{ij}\beta^{\Delta}_{kj} z^{\Delta}_{ik}| \right]^2} \\
&= \sqrt{\sum_{i=1}^{r} \left[\sum_{j=1}^{m} |x_{ij}\beta^{\Delta}_{c^*_i j} z^{\Delta}_{ic^*_i}| + |x_{ij}\beta^{\Delta}_{mc_i j} z^{\Delta}_{imc_i}| \right]^2} \\
&\leq \sqrt{\sum_{i=1}^{r} \left[\sum_{j=1}^{m} 2 \max_{ijk} |x_{ij}\beta^{\Delta}_{kj}| \right]^2} \\
&\leq \sqrt{\sum_{i=1}^{r} \left[\sum_{j=1}^{m} 2 \max_{ijk} |x_{ij}||\beta^{\Delta}_{kj}| \right]^2} \\
&\leq \sqrt{\sum_{i=1}^{r} \left[\sum_{j=1}^{m} 2 \max_{ijk} |\beta^{\Delta}_{kj}| \right]^2} \\
&= 2m\sqrt{r}\beta^{\Delta}_{max}
\end{aligned}$$

A similar analysis on $\|(Z^* * X)\left(\beta^* - \widehat{\beta}\right)\|_2$ yields that

$$\begin{aligned}
\|(Z^* * X)\beta^{\Delta}\|_2 &= \sqrt{\sum_{i=1}^{n}\left[\sum_{j=1}^{m}\sum_{k=1}^{\ell} z^*_{ik}x_{ij}\beta^{\Delta}_{kj}\right]^2} \\
&\geq \beta^{\Delta}_{min}\sqrt{\sum_{i=1}^{n}\left[\sum_{j=1}^{m}\sum_{k=1}^{\ell} z^*_{ik}x_{ij}\right]^2} \\
&\geq \beta^{\Delta}_{min}\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}\sum_{k=1}^{\ell} z^*_{ik}x^2_{ij}} \\
&= \beta^{\Delta}_{min}\sqrt{\sum_{i=1}^{n}\|x_i\|_2} \\
&= \sqrt{n}\beta^{\Delta}_{min}
\end{aligned}$$

where we have assumed that $\|x_i\|_2 = 1\ \forall i = 1$. Also, by assumption, $r < \left(\frac{n}{2m}\right)\left(\frac{\beta^{\Delta}_{min}}{\beta^{\Delta}_{max}}\right)^2$. Hence,

$$\|(Z^{\Delta} * X)\left(\beta^* - \widehat{\beta}\right)\|_2 < \|(Z^* * X)\left(\beta^* - \widehat{\beta}\right)\|_2 ,$$

which proves the existence of $\kappa > 0$ such that

$$\|(Z^* * X)\beta^{\Delta}\|_2^2 - \|(Z^{\Delta} * X)\beta^{\Delta}\|_2^2 = \kappa\|(Z^* * X)\beta^{\Delta}\|_2^2 .$$

Now we are in a position to prove the final statement. First note that we have β^{Δ} such that

$$\begin{aligned}
\|\beta^{\Delta}_S\| &\leq \frac{s}{n\eta^2}\left(\|(Z^* * X)\beta^{\Delta}\|_2^2 - \frac{2}{\kappa}\|(Z^* * X)\beta^*\|_2^2\right) \\
\implies \|\beta^{\Delta}_S\| &\leq \frac{s}{n\kappa\eta^2}\left(\kappa\|(Z^* * X)\beta^{\Delta}\|_2^2 - 2\|(Z^* * X)\beta^*\|_2^2\right) \\
\implies \|\beta^{\Delta}_S\| &\leq \frac{s}{n\kappa\eta^2}\left(\|(Z^* * X)\beta^{\Delta}\|_2^2 - \|(Z^{\Delta} * X)\beta^{\Delta}\|_2^2 - 2\|(Z^* * X)\beta^*\|_2^2\right)
\end{aligned}$$

Hence, letting $\theta = \kappa\eta$ gives the desired result. □

Proof. Proof of Theorem 2.4.4. This proof can be divided into six steps. The first three steps lead to a probabilistic bound without the cluster compatibility condition, and the last three steps extend this under the cluster compatibility condition. First, we bound using the chance that the solution of the CWR algorithm has a smaller LASSO objective than the true model. Afterwards, we bound the forecasting error that is due to noisy measurements caused by the

systematic error ϵ. Then, we can establish our first bound. Next, we rewrite the cluster compatibility condition into a useful inequality. Additionally, we use the sparseness of the true regression parameters to provide another helpful inequality. Finally, these inequalities together allow us to construct the second bound.

(Step 1) Consider $\widehat{Z}$ and $\widehat{\beta}$ and note that the objective at the estimated parameters is smaller than the objective at Z^* and β^*, which means that,

$$\frac{1}{n}||y-(\widehat{Z}*X)\widehat{\beta}||_2^2+\lambda||\widehat{\beta}||_1 \leq \frac{1}{n}||y-(Z^**X)\beta^*||_2^2+\lambda||\beta^*||_1.$$

Plugging $y=(Z^**X)\beta^*+\epsilon$ into this inequality, we obtain

$$\frac{1}{n}||(Z^**X)\beta^*+\epsilon-(\widehat{Z}*X)\widehat{\beta}||_2^2 \leq \frac{1}{n}||(Z^**X)\beta^*+\epsilon-(Z^**X)\beta^*||_2^2+\lambda(||\beta^*||_1-||\widehat{\beta}||_1),$$

which we can rewrite into

$$\frac{1}{n}||(Z^**X)\beta^*-(\widehat{Z}*X)\widehat{\beta}||_2^2 \leq \frac{2}{n}\epsilon^T((\widehat{Z}*X)\widehat{\beta}-(Z^**X)\beta^*)+\lambda(||\beta^*||_1-||\widehat{\beta}||_1).$$

To prove the probabilistic bound, we need to analyze the right-hand side of this event.

(Step 2) For the first term on the right-hand side, we add and subtract $(\widehat{Z}*X)\beta^*$ to the product's second term, use Hölder's inequality, and use the triangle inequality to obtain

$$\begin{aligned}
&\epsilon^T((\widehat{Z}*X)\widehat{\beta}-(Z^**X)\beta^*)=\epsilon^T((\widehat{Z}*X)\widehat{\beta}-(\widehat{Z}*X)\beta^*+(\widehat{Z}*X)\beta^*-(Z^**X)\beta^*)\\
&=\epsilon^T(\widehat{Z}*X)(\widehat{\beta}-\beta^*)+\epsilon^T(\widehat{Z}*X-Z^**X)\beta^*\\
&\leq |\epsilon^T(\widehat{Z}*X)(\widehat{\beta}-\beta^*)|+|\epsilon^T(\widehat{Z}*X-Z^**X)\beta^*|\\
&\leq ||\epsilon^T(\widehat{Z}*X)||_\infty||\widehat{\beta}-\beta^*||_1+||\epsilon^T(\widehat{Z}*X-Z^**X)||_\infty||\beta^*||_1\\
&\leq ||\epsilon^T(\widehat{Z}*X)||_\infty||\widehat{\beta}-\beta^*||_1+||\epsilon^T(\widehat{Z}*X)||_\infty||\beta^*||_1+||\epsilon^T(Z^**X)||_\infty||\beta^*||_1.
\end{aligned}$$

Applying Lemma 2.4.2 this yields with probability at least $1-\delta$ that

$$\frac{2}{n}\epsilon^T((\widehat{Z}*X)\widehat{\beta}-(Z^**X)\beta^*) \leq \frac{\lambda}{2}(||\widehat{\beta}-\beta^*||_1+2||\beta^*||_1).$$

Merging with the result of step 1 this implies that

$$\begin{aligned}\frac{1}{n}||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2 &\leq \frac{\lambda}{2}(||\widehat{\beta} - \beta^*||_1 + 2||\beta^*||_1) + \lambda(||\beta^*||_1 - ||\widehat{\beta}||_1) \\ &= \frac{\lambda}{2}||\widehat{\beta} - \beta^*||_1 + 2\lambda||\beta^*||_1 - \lambda||\widehat{\beta}||_1\end{aligned}$$

(Step 3) For our first probabilistic bound, we use the result of step 2, the triangle inequality, and $\lambda = 4\sigma\sqrt{\frac{2}{n}\log\left(\frac{2nm\ell}{\delta}\right)}$ to find that

$$\frac{1}{n}||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2 \leq \frac{\lambda}{2}||\widehat{\beta} - \beta^*||_1 + 2\lambda||\beta^*||_1 - \lambda||\widehat{\beta}||_1 \leq \frac{5\lambda}{2}||\beta^*||_1.$$

The guarantee is conditional on the event E, which implies our first probabilistic bound

$$\begin{aligned}&\mathbb{P}\left(||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2 \leq \frac{5\lambda}{2}||\beta^*||_1\right) \geq \\ &\mathbb{P}\left(||(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}||_2^2 \leq \frac{5\lambda}{2}||\beta^*||_1 | E\right)\mathbb{P}(E) \geq (1-\delta)\zeta.\end{aligned}$$

(Step 4) To create a further bound under the cluster compatibility condition, we observe that for some $\theta > 0$,

$$\begin{aligned}\frac{1}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 = & \\ &\frac{1}{n}\|(Z^* * X)(\beta^* - \widehat{\beta}) - ((\widehat{Z} - Z^*) * X)(\widehat{\beta} - \beta^*) - ((\widehat{Z} - Z^*) * X)\beta^*\|_2^2 \\ &\geq \frac{1}{n}\left(\|(Z^* * X)\beta^{\Delta}\|_2^2 - \|(Z^{\Delta} * X)\beta^{\Delta}\|_2^2 - \|(Z^{\Delta} * X)\beta^*\|_2^2\right) \\ &\geq \frac{1}{n}\left(\|(Z^* * X)\beta^{\Delta}\|_2^2 - \|(Z^{\Delta} * X)\beta^{\Delta}\|_2^2 - 2\|(Z^* * X)\beta^*\|_2^2\right) \\ &\geq \frac{\theta^2}{s}\|\beta_S^{\Delta}\|_1^2,\end{aligned}$$

which implies that

$$\|\widehat{\beta}_S - \beta_S^*\|_1^2 \leq \frac{s}{n\theta^2}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 .$$

(Step 5) To construct a helpful inequality, we first rearrange $||\widehat{\beta}||_1$ to find that

$$\begin{aligned}\|\widehat{\beta}\|_1 &= \|\widehat{\beta}_S\|_1 + \|\widehat{\beta}_{S^C}\|_1 \\ &= \|\beta^*_S + \widehat{\beta}_S - \beta^*_S\|_1 + \|\widehat{\beta}_{S^C}\|_1 \\ &\geq \|\beta^*_S\|_1 - \|\widehat{\beta}_S - \beta^*_S\|_1 + \|\widehat{\beta}_{S^C}\|_1 \\ \implies \|\widehat{\beta}_{S^C}\|_1 &\leq \|\widehat{\beta}\|_1 + \|\widehat{\beta}_S - \beta^*_S\|_1 - \|\beta^*_S\|_1.\end{aligned}$$

Merging with the result of step 2 yields that

$$\begin{aligned}\frac{2}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + 2\lambda\|\widehat{\beta}_{S^C}\|_1 \leq& \\ \frac{2}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + 2\lambda\|\widehat{\beta}\|_1 + 2\lambda\|\widehat{\beta}_S - \beta^*_S\|_1 - 2\lambda\|\beta^*_S\|_1 \,.& \\ \leq \lambda\|\widehat{\beta}_S - \beta^*_S\|_1 + \lambda\|\widehat{\beta}_{S^C}\|_1 + 4\lambda||\beta^*||_1 + 2\lambda\|\widehat{\beta}_S - \beta^*_S\|_1 - 2\lambda\|\beta^*_S\|_1& \\ = 3\lambda\|\widehat{\beta}_S - \beta^*_S\|_1 + 2\lambda\|\beta^*_S\|_1 + \lambda\|\widehat{\beta}_{S^C}\|_1.&\end{aligned}$$

Using this inequality shows that

$$\begin{aligned}\frac{2}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + \lambda\|\widehat{\beta} - \beta^*\|_1 =& \\ \frac{2}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + \lambda\|\widehat{\beta}_S - \beta^*_S\|_1 + \lambda\|\widehat{\beta}_{S^C}\|_1& \\ \leq 3\lambda\|\widehat{\beta}_S - \beta^*_S\|_1 + 2\lambda\|\beta^*_S\|_1 + \lambda\|\widehat{\beta}_S - \beta^*_S\|_1& \\ = 4\lambda\|\widehat{\beta}_S - \beta^*_S\|_1 + 2\lambda\|\beta^*_S\|_1 \,.&\end{aligned}$$

Following the inequality $4ab \leq a^2 + 4b^2$ this yields that

$$\begin{aligned}\frac{2}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + \lambda\|\widehat{\beta} - \beta^*\|_1 \leq& \\ 4\lambda\sqrt{\frac{s}{n\theta^2}}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2 + 2\lambda\|\beta^*_S\|_1& \\ \leq 4\lambda^2\frac{s}{\theta^2} + \frac{1}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + 2\lambda\|\beta^*_S\|_1 \,.&\end{aligned}$$

(Step 6) For our second probabilistic bound, we rewrite the result of step 5 to find that

$$\frac{1}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + \lambda\|\widehat{\beta} - \beta^*\|_1 \leq 4\lambda^2\frac{s}{\theta^2} + 2\lambda\|\beta^*_S\|_1 \,.$$

Hence,

$$\mathbb{P}\left(\frac{1}{n}\|(Z^* * X)\beta^* - (\widehat{Z} * X)\widehat{\beta}\|_2^2 + \lambda\|\widehat{\beta} - \beta^*\|_1 \leq 4\lambda^2\frac{s}{\theta^2} + 2\lambda\|\beta_S^*\|_1\right) \geq (1-\delta).$$

□

Proof. Proof of Theorem 2.4.5. Note that $\widehat{z}_{0k} = 1/\ell$ for all k, and $z^*_{0k^*} = 1$ and $z^*_{0k} = 0$ for $k \neq k^*$. Given these cluster assignments, bounding yields that

$$\begin{aligned}
\|\sum_{k=1}^{\ell} z^*_{0k}\sum_{j=1}^{m}\beta^*_{kj}x_{0j} - \sum_{k=1}^{\ell}\widehat{z}_{0k}\sum_{j=1}^{m}\widehat{\beta}_{kj}x_{0j}\|_1 &= \|\sum_{j=1}^{m}\beta^*_{k^*j}x_{0j} - \frac{1}{\ell}\sum_{k=1}^{\ell}\sum_{j=1}^{m}\widehat{\beta}_{kj}x_{0j}\|_1 \\
&\leq \frac{1}{\ell}\sum_{k=1}^{\ell}\|\beta^*_{k^*} - \widehat{\beta}_k\|_1\|x_0\|_\infty \\
&\leq \frac{1}{\ell}\sum_{k=1}^{\ell}\|\beta^*_{k^*} - \widehat{\beta}_k + \beta^*_k - \beta^*_k\|_1 \\
&\leq \frac{1}{\ell}\sum_{k=1}^{\ell}\|\beta^*_{k^*} - \beta^*_k\|_1 + \frac{1}{\ell}\sum_{k=1}^{\ell}\|\widehat{\beta}_k - \beta^*_k\|_1 \\
&= C + \frac{1}{\ell}\|\widehat{\beta} - \beta^*\|_1
\end{aligned}$$

Applying Theorem 2.4.4, we have on an event with at least probability $(1-\delta)\zeta$ that

$$\|\widehat{\beta} - \beta^*\|_1 \leq 4\lambda\frac{s}{\theta^2} + 2\|\beta^*\|_1,$$

which implies our probabilistic bound

$$\mathbb{P}\left(||\sum_{k=1}^{\ell} z^*_{0k}\sum_{j=1}^{m}\beta^*_{kj}x_{0j} - \sum_{k=1}^{\ell}\widehat{z}_{0k}\sum_{j=1}^{m}\widehat{\beta}_{kj}x_{0j}||_1 \leq C + \frac{2}{\ell}\|\beta^*\|_1 + \frac{4}{\ell}\lambda\frac{s}{\theta^2}\right) \geq (1-\delta)\zeta.$$

□

Appendix B

Proofs of Chapter 3

B.1 Lower Bound for Classical Approaches

Proof. Proof of Theorem 3.4.3: Consider the case when $d = 2$. Then, $u_0 \sim \mathcal{N}(0, \sigma^2 I_2)$. Furthermore, $V_1 = [1, 0]$ and $V_2 = [0, 1]$. Clearly, Product 1 is optimal when $u_{0_1} > u_{0_2}$ and vice versa. Consider the following events: $\mathcal{E}_1 = \{u_{0_1} < u_{0_2} - \rho\}$, and $\mathcal{E}_2 = \{u_{0_2} < u_{0_1} - \rho\}$. Then on $\mathcal{E}_1$, recommending product 1 leads to customer disengagement with probability p and on $\mathcal{E}_2$, recommending product 2 leads to customer disengagement with probability p. Next, we will characterize the probability of events $\mathcal{E}_1$ and $\mathcal{E}_2$. $\mathbb{P}(\mathcal{E}_1) = \mathbb{P}(u_{0_1} < u_{0_2} - \rho) = \mathbb{P}\left(\frac{u_{0_1} - u_{0_2}}{\sqrt{2}\sigma} < \frac{-\rho}{\sqrt{2}\sigma}\right) = \mathbb{P}\left(Z < \frac{-\rho}{\sqrt{2}\sigma}\right) = C$, such that $C \in (0, 1)$. Symmetrically, $\mathbb{P}(\mathcal{E}_2) = \mathbb{P}(u_{0_2} < u_{0_1} - \rho) = C$. Any policy π has two options at time 1: either to recommend product 1 or to recommend product 2. First consider the case when $a_1 = 1$ and notice that

$$\mathbb{E}_{u_0 \sim \mathcal{P}}\left[\mathcal{R}^{\pi}(T, \rho, p, u_0)\right] \geq \sum_{t=1}^{T} r_t(\rho, p, u_0 \in \mathcal{E}_1).\mathbb{P}(\mathcal{E}_1) \geq T \cdot \mathbb{P}(\mathcal{E}_1) \cdot p = CpT\,.$$

Similarly, when $a_1 = 2$, $\mathbb{E}_{u_0 \sim \mathcal{P}}\left[\mathcal{R}^{\pi}(T, \rho, p, u_0)\right] \geq CpT$. Hence,

$$\inf_{\pi \in \Pi} \sup_{\rho > 0} \mathbb{E}_{U_0 \sim \mathcal{P}}\left[\mathcal{R}^{\pi}(T, \rho, p, U_0)\right] = C \cdot p \cdot T = \mathcal{O}(T)\,,$$

The proof follows similarly for any $d > 2$ since the probability of disengagement continues to be strictly positive in the initial round. □

Before we prove Theorem 3.4.4, we prove an important Lemma that relates the confidence width of the mean reward of product V ($\|V\|^2_{X_t^{-1}}$) and shows that this width shrinks at a rate faster than the confidence width of the estimation of the gap between reward from V and the

optimal product (Δ_V).

Lemma B.1.1. Let π be a consistent policy and let $a_1, .., a_t$ be actions taken under policy π. Let $u_0 \in R^d$ be a realization of the random user vector, $U_0 \sim \mathcal{P}$, such that there is a unique optimal product, V_* amongst the set of feasible products. Then $\forall\ V \in \{V_1, ...V_n\}/\ V_*$, $\limsup_{t\to\infty} \log(t)\|V\|^2_{X_t^{-1}} \leq \frac{\Delta_V^2}{2(1-\nu)}$, where $\Delta_V = u_0^\top V_* - u_0^\top V$ and $X_t = \mathbb{E}\left[\sum_{l=1}^T a_l a_l'\right]$.

Proof. The proof strategy is similar to that of Theorem 3.4.4 in Lattimore and Szepesvari (2016) with two main steps. In Step 1, we show that $\limsup_{t\to\infty} \log(t)\|V - V_*\|^2_{X_t^{-1}} \leq \frac{\Delta_V^2}{2(1-\nu)}$. Then, in Step 2, we connect this result to the matrix norm on the features of V which leads to the final result. We skip the details for the sake of brevity and refer the interested readers to Lattimore and Szepesvari (2016). □

Proof. Proof of Theorem 3.4.4: We will prove that whenever $|S(u_0, \rho)| < d$, any consistent policy, π, recommends products outside of the customer's feasibility set infinitely often. Note that for any realization of u_0, one can reduce ρ and make it sufficiently small so that $|S(u_0, \rho)| < d$. Customer disengagement thus follows directly since there is a positive probability, p, of customer leaving the platform whenever a product outside the customer's feasibility set is offered.

Let us assume, by contradiction, that there exists a policy π that is consistent and offers products inside the feasible set infinitely often. This implies that there exists $\bar{t}$ such that $\forall t > \bar{t},\ a_t \in \mathcal{S}(u_0, \rho)$. Now under the stated assumptions of the simplified setting, there are d products in total ($n = d$) and the feature vector of the i^{th} product is the i^{th} basis vector. Further let u_o, the unknown consumer feature vector, and ρ, the tolerance threshold parameter be such that WLOG, $\mathcal{S}(u_0, \rho) = \{2, 3...d\}$ (follows by Definition (3.2)). That is, only the first product is outside of the feasible set. Also let, $R_t^\pi = \begin{bmatrix} T_1^\pi(t) & 0 & \dots \\ \vdots & \ddots & \\ 0 & & T_d^\pi(t) \end{bmatrix}$, where $T_j(t) = \mathbb{E}\left[\sum_{f=1}^t \mathbb{1}\{a_f^\pi = j\}\right]$. $T_j(n)$ is the total number of times the j^{th} product is offered until

time t under policy π. Next consider the following:

$$\begin{aligned}
\limsup_{t\to\infty} \log(t)\|e_1\|^2_{X_t^{-1}} &= \limsup_{t\to\infty} \log(t) e_1^\top X_t^{-1} e_1 \\
&= \limsup_{t\to\infty} \log(t) e_1^\top \mathbb{E}\left[\sum_{f=1}^{t} a_f a_f^\top\right]^{-1} e_1 \\
&= \limsup_{t\to\infty} \log(t) e_1^\top \left[R_t\right]^{-1} e_1 \\
&\geq \limsup_{t\to\infty} \log(t) \left(\frac{1}{T_1(t)}\right) \\
&\geq \limsup_{t\to\infty} \log(t) \left(\frac{1}{T_1(\bar{t})}\right) = \infty\,.
\end{aligned} \tag{B.1}$$

Where the second to last inequality follows from the fact that $\forall t > \bar{t}$, π recommends products inside the feasible set, $\mathcal{S}(u_0, \rho)$, which does not contain product 1. Furthermore, $T_1(\bar{t}) = T_1(\bar{t}+1) = T_1(\bar{t}+2) = ... = \lim_{n\to\infty} T_1(\bar{t}+n)$. For any finite Δ_{V_1}, and $0 < \nu < 1$, we have that, $\limsup_{t\to\infty} \log(t)\|e_1\|^2_{X_t^{-1}} \geq \frac{\Delta_1^2}{2(1-\nu)}$. which implies that $\exists a_i$ in the action space such that the condition of Lemma B.1.1 is not satisfied. Hence, we have show that there exists no consistent policy that recommends products inside of the feasible set of products infinitely often. Now since ρ is small and p is positive, customers are guaranteed to disengage from the platform eventually. This leads to a linear rate of regret for all customers. □

Proof. Proof of Theorem 3.4.5: We prove the result in two parts. In the first part, we consider latent attribute realizations for which the optimal apriori product, which is chosen by the GBU policy in the initial round, is not optimal. In this case, if we take the tolerance threshold parameter to be small, there is a positive probability that the customer leaves at the beginning of the time period, which leads to linear regret over this set of customers. In the second part, we consider those customers for which the apriori product is indeed optimal. For these customers, we again take the case when ρ is sufficiently small and reduce the leaving time to the probability of shifting from the first arm to another arm. Since the switched arm is suboptimal and outside of the user threshold, the customer leaves with a positive probability resulting in linear regret for this set of customers. Recall, by the simplified setting, there are d total products and attribute of the i^{th} product is the i^{th} basis vector. Furthermore, the prior is uninformative. That is, the first recommended product is selected at random. Let us assume, WLOG, that the GBU policy picks product 1 to recommend. We have two cases to analyze: (i) product 1 is sub optimal for the realized latent attribute vector, u_0, (ii) product 1 is optimal for the realized

latent attribute vector, u_0. Let us consider case (i) when product 1 is suboptimal. In this case, if we let ρ to be smaller than the difference between the utility of the optimal product and product 1 ($\rho < u_0^\top(V_* - V_1)$), then the customer leaves with probability p in the current round. Hence, for all such customers $\mathcal{R}^\pi(T, \rho, p, u_0) \geq T \cdot p = pT$. Next, we consider the customers for which product 1 is optimal. In this case, the customer leaves with probability p when the greedy policy switches from the initial recommendation to some other product. Again, at any such time, t, if we let ρ to be small such that the chosen product is outside of the customer threshold, then we will have disengagement with a constant probability p in that round. This would again lead to a linear rate of regret. Let $E_i^t = \{V_1^\top \hat{u}_t - V_i^\top \hat{u}_t > 0\}$. E_i^t denotes the event that the initially picked product is indeed better than the i^{th} product in the product assortment at time t. Similarly, let G^t to be the event that the GBU policy switches to some other product from product 1 by time t. Then,

$$\mathbb{P}(G^t) = \mathbb{P}\left(\cup_{i=2..n} \cup_{j=1..t} (E_i^j)^c\right) \geq \mathbb{P}\left((E_i^j)^c\right), \ \forall i = 2, .., n, \forall j = 1, .., t\,.$$

We will lower bound the probability of product 1 not being the optimal product for any time t under the GBU policy. Since we are dynamically updating the estimated latent customer feature vector, the probability of switching depends on the realization of ε_t, the idiosyncratic noise term that governs the customer response. We will first consider the case of two products ($d = 2$). Furthermore, we will analyze the probability of switching from product 1 to product 2 after round 1 ($(E_2^1)^c$). First note that, $E_i^t = \{V_1^t \hat{u}_t - V_i^\top \hat{u}_t \geq 0\}$, which implies

$$(E_i^t)^c = \{V_i^t \hat{u}_t - V_1^\top \hat{u}_t > 0\} = \{(V_i - V_1)^\top (\hat{u}_t - u_0) > \Delta_i\}\,.$$

where $\Delta_i = V_1^\top u_0 - V_i^\top u_0$. Now, note that

$$\hat{u}_t = \left[\sum_{f=1}^{t} a_f a_f^\top + \frac{\xi^2}{\sigma^2} I_d\right]^{-1} [a_{1:t}]^\top Y_{f=1:t}\,.$$

Hence,

$$\hat{u}_1 = \begin{bmatrix} 1 + \frac{\xi^2}{\sigma^2} & 0 \\ 0 & \frac{\xi^2}{\sigma^2} \end{bmatrix}^{-1} \begin{bmatrix} Y_1 & 0 \\ 0 & 0 \end{bmatrix} = \left[\frac{\sigma^2 Y_1}{\sigma^2 + \xi^2}, 0\right]$$

Therefore, we are interested in the event

$$\left\{\frac{\sigma^2 Y_1}{\sigma^2+\xi^2}<0\right\}=\{Y_1<0\}=\{u_{0_1}+\varepsilon_1<0\}=\{u_{0_1}+\varepsilon_1<0\}\,.$$

Now note that for any realization of u_0, there is a positive probability of the event above happening. Hence, let $\mathbb{P}(\varepsilon_1<-u_{0_1})=C_4>0$. This implies that $\mathbb{P}(G^t)\geq C_4$. Following the same regret argument as before, we have that for all such customers, $\mathcal{R}^{\mathrm{GBU}}(T,\rho,p,u_0)=C_4\cdot T$. The argument for $d>2$ follows similarly since with positive probability, the GBU policy would either get stuck at a sub-optimal arm or would switch to a sub-optimal arm. Hence, we have shown that regardless of the realization of the latent user attribute, u_0 the GBU policy incurs linear regret on the customers. That is,

$$\forall u_0, \sup_{\rho>0}\ \mathcal{R}^{\mathrm{GBU}}(T,\rho,p,u_0)=C_2\cdot T=\mathcal{O}(T)\,.$$

□

Proof. Proof of Theorem 3.4.6: We will use the same strategy as in the proof of Theorem 3.4.5 with two main exceptions; (i) because this is the case of no disengagement, we cannot select ρ to be appropriately small, (ii) since the result is on the expectation of regret over all possible latent attribute realizations, we need to show the result only for a set of customer attributes with positive measure. Noting (ii) above, we focus on customers for which the first recommended product is suboptimal and show that with positive probability the greedy policy gets "stuck" on this product and keeps on recommending this product. This leads to a linear rate of regret for these customers.

Step 1 (Lower bound on selecting an initial suboptimal product): WLOG assume that product 1 was recommended and consider the set of customers for which product 1 is suboptimal. Note that since u_0 is multivariate normal, there is a positive measure of such customers.

Step 2 (Upper bound on the probability of switching from the current product to a different product during the later periods:) Now that we have selected a suboptimal product, we will bound the probability that the GBU policy continues to offer the same product until the end of the horizon. We will use the same notation as before. Recall that $E_i^t=\{V_1^\top\hat{u}_t-V_i^\top\hat{u}_t>0\}$. E_i^t denotes the event that the initially picked product is indeed better than the i^{th} product in the product assortment at time t. Similarly, G^t denotes the event that the GBU policy switches

to some other product from product 1 by time t. Then, we are interested in lower bounding the event that the GBU policy never switches from the product 1 and gets stuck. That is, $\mathbb{P}((G^t)^c) \geq 1 - \sum_{j=1..t} \sum_{i=1..n, i\neq i^*} \mathbb{P}((E_i^j)^c)$. As before, first we consider the case when there are only 2 products. In this case, if we start by recommending product 1, we want to calculate the probability of continuing with Product 1 through out the time horizon. First note that using the same calculation, one can show that if until time t, we continue with only recommending product 1, then the latent attribute estimate at time t is given by $\hat{u}_t = \left[\frac{\sigma^2 \sum_{f=1}^{t} Y_f}{t\sigma^2+\xi^2}, 0\right]$. For any time t, we claim that the GBU policy continues to recommend the same product as before if the utility realization at time t is positive. That is, if $Y_{t-1} > 0$ and the GBU policy offered product 1 in rounds $1, ..t-1$, then it will continue recommending product 1 in round t. We prove this claim using induction. Note that the base case of $t = 2$ was proved in the previous proof (reversing the argument in the second part of Theorem 3.4.5 results in the base case) and we omit the details here. Now by induction hypothesis, we have that the GBU policy offered product 1 at time $t-1$ because $Y_1, ...Y_{t-2}$ were all positive. Now consider time t let $Y_{t-1} > 0$, Then we have that $\hat{u}_{t-1} = \left[\frac{\sigma^2 \sum_{f=1}^{t-1} Y_f}{t\sigma^2+\xi^2}, 0\right]$. We will select product 1 if $\frac{\sigma^2 \sum_{f=1}^{t-1} Y_f}{t\sigma^2+\xi^2} > 0$ which implies $\frac{\sigma^2 \sum_{f=1}^{t-2} Y_f}{t\sigma^2+\xi^2} + \frac{\sigma^2 Y_{t-1}}{t\sigma^2+\xi^2} > 0$. But note that by induction hypothesis, the first term of the sum above is positive. Hence, GBU selects product 1 at least when $\frac{\sigma^2 Y_{t-1}}{t\sigma^2+\xi^2} > 0$ which proves the claim. Now note that for any time t, the probability Y_i being positive is independent across time periods. Furthermore,

$$\mathbb{P}\left(\frac{\sigma^2 Y_{t-1}}{t\sigma^2+\xi^2} > 0\right) = \mathbb{P}\left(Y_{t-1} > 0\right) = \mathbb{P}\left(u_{0_1} + \varepsilon_t > 0\right).$$

For any t, probability of not switching from the first product is at least

$$\begin{aligned}
\mathbb{P}((G^t)^c) = 1 - \mathbb{P}((G^t)^c) &\geq 1 - \sum_{j=1..t} \sum_{i=1..n, i\neq i^*} \mathbb{P}((E_i^j)^c) \\
&= 1 - \sum_{j=1..t} \mathbb{P}\left(\varepsilon_t > -u_{0_1}\right) \\
&= 1 - t\mathbb{P}\left(\varepsilon > -u_{0_1}\right)
\end{aligned}$$

Now for any t, if we consider all realizations of u_0 such that $\mathbb{P}\left(\varepsilon > -u_{0_1}\right) < \frac{1}{t}$, then we have that the above probability is always positive. Note that Product 1 was not optimal, hence, over these customers, the GBU policy incurs linear regret which results in an expected linear regret.

That is,

$$\mathbb{E}_{u_0\sim\mathcal{P}}\left[\mathcal{R}^{GBU}(T,\rho,0,u_0)\right] = C_3\cdot T = \mathcal{O}(T)\,.$$

The proof for the case when $d > 2$ follows similarly since the GUB policy would continue to switch to a sub-optimal arm, or get "stuck" at an optimal arm. □

B.2 Upper Bound for Constrained Bandit

Proof. Proof of Theorem 3.5.4: Consider any feasible $\rho > 0$ and let $\tilde{\gamma}$ be such that only a single product remains in the constrained exploration set. Note that a feasible γ that ensures that only a single product is chosen for exploration is $\gamma < \frac{1}{\sqrt{2}}$. Such a selection would ensure that $\mathbf{OP}(\gamma)$ picks a single product $(\tilde{i})$ in the exploration phase. Now let $\tilde{\gamma} = \frac{1}{\sqrt{2}}$ and consider $\mathcal{W}_{\lambda,\tilde{\gamma}} := \{u_0 : V_{\tilde{i}}^\top u_0 > \max_{i-1,..,n,i\neq\tilde{i}} V_i^\top u_0\}$. Then we have that $\forall u_0 \in \mathcal{W}_{\lambda,\tilde{\gamma}}$, customers are going to continue engaging with the platform since the recommended product is the corresponding optimal product. Next, since the prior is a multivariate normal, we have that $\mathbb{P}(\mathcal{W}_{\lambda,\tilde{\gamma}}) > 0$. This holds because by assumption since V_i is the i^{th} basis vector and u_0 is multivariate normal with prior mean of 0 across all dimensions. So, the probability of sampling a u_0 such that $u_{0_{\tilde{i}}} > u_{0_j},\ \forall j = 1,..,d,\ j \neq \tilde{i}$ has a positive measure under the prior assumption. We claim that for any ρ, the regret incurred from this policy will be optimal. Consider two cases: (i) When ρ is such that their is more than 1 product within the customer's relevance threshold. That is, $|\mathcal{S}(u_0,\rho)| > 1$ (ii) When there is a single product within the customer's tolerance threshold, ρ. That is, $|\mathcal{S}(u_0,\rho)| = 1$. In both cases, $\tilde{i}$, which is the only product in the exploration phase, is contained in $|\mathcal{S}(u_0,\rho)|$. That is, $\forall u_0 \in \mathcal{W}_{\lambda,\tilde{\gamma}},\ \tilde{i} \in \mathcal{S}(u_0,\rho)$. Hence, there are no chances of customer disengagement if product $\tilde{i}$ is offered to the customer. Furthermore, regret over all such customers is in fact 0 since the platform recommends their optimal product. This proves the result. □

Proof. Proof of Theorem 3.5.5: We will prove the above result in three steps. In the first step we will lower bound the probability that the constrained exploration set, Ξ, contains the optimal product for an incoming vector. In the second step we will lower bound the probability of customer engagement over the constrained set. Finally, in the last step, we use the above lower bounds on probabilities to upper bound regret from the Constrained Bandit algorithm.

Step 1 (Lower bounding the probability of not choosing the optimal product for an incoming customer in the constrained set): Let, $\mathcal{E}_{no-optimal}$, be the event that the optimal product, V_*

for the incoming user is not contained in Ξ. Also let $\tilde{i} = \arg\max_{V\in[-1,1]^d} \bar{u}^\top V$, denote the attributes of the prior optimal product. Notice that $V_{\tilde{i}} = \bar{u}$ since $\|\bar{u}\|_2 = 1$. Also recall that $V_* = \arg\max_{V\in[-1,1]^d} u_0^\top V$, denotes the current optimal product which is unknown because of unknown customer latent attributes. We are interested in $\mathbb{P}(\mathcal{E}_{no-optimal}) = \mathbb{P}(V_* \notin \Xi)$. In order to characterize the above probability, we focus on the structure of the constrained set, Ξ. Recall that Ξ is the outcome of Step 1 of Constrained Bandit (Algorithm 2) and uses $\mathbf{OP}(\gamma)$ to restrict the exploration space. It is easy to observe that Ξ in the continuous feature space case would be centred around the prior optimal product vector ($\bar{u}$) and will contain all products that are at most γ *away* from each other. We are interested in characterizing the probability of the event that $u_0 \notin [\bar{u}_l, \bar{u}_r]$ where $\bar{u}_l$ and $\bar{u}_r$ denote the attributes of the farthest products inside a γ constrained sphere. Simple geometric analysis yields that $\bar{u}$ and $\bar{u}_l$ are $\bar{d} = \sqrt{2\left(1-\sqrt{(1-\gamma^2/4)}\right)}$ apart. The distance between $\bar{u}$ and $\bar{u}_r$ follows symmetrically. Having calculated the distance between $\bar{u}$ and $\bar{u}_l$, we are now in a position to characterize the probability of $\mathcal{E}_{no-optimal}$. But $\mathbb{P}(\mathcal{E}_{no-optimal}) = \mathbb{P}(V_* \notin \Xi) = \mathbb{P}\left(\|u_0 - \bar{u}\|_2 \geq \bar{d}\right)$. Note by Holder's inequality that, $\bar{d} \leq \|u_0 - \bar{u}\|_2 \leq \|u_0 - \bar{u}\|_1$, which implies that,

$$\mathbb{P}(\mathcal{E}_{no-optimal}) = \mathbb{P}\left(\|u_0 - \bar{u}\|_2 \geq \bar{d}\right) \leq \mathbb{P}\left(\|u_0 - \bar{u}\|_1 \geq \bar{d}\right).$$

Note that $u_0 \sim \mathcal{N}(\bar{u}, \frac{\sigma^2}{d^2} I_d)$. Using Lemma B.5.1 in Appendix B.5, we have that, $\mathbb{P}\left(\|u_0 - \bar{u}\|_1 \leq \bar{d}\right) \geq 1 - 2d\exp\left(-\left(1-\sqrt{(1-\gamma^2/4)}/\sigma\right)\right)$, which results in a lower bound.

Step 2 (Lower bounding the probability of customer disengagement due to relevance of the recommendation): Recall that customer disengagement decision is driven by the relevance of the recommendation and the tolerance threshold of the customer. Hence,

$$\begin{aligned}\mathbb{P}(u_0^\top V_* - u_0^\top V_i < \rho) &= \mathbb{P}(u_0^\top u_0 - u_0^\top u_i < \rho | u_0, u_i \in \Xi) = \mathbb{P}(u_0^\top (u_0 - u_i) < \rho | u_0, u_i \in \Xi), \\ &\geq \mathbb{P}\left(\|u_0\|_2 < \frac{\rho}{\gamma} | u_0, u_i \in \Xi\right) \geq \left(1 - 2d\exp\left(-\left(\rho/\gamma - \sum_{i=1}^{i=d} \bar{u}_i\right)^2 / \sigma^2\right)\right).\end{aligned}$$

where the second to last inequality follows by Cauchy-Schwarz ineqaulity and the last inequality follows by Lemma B.5.1. This in-turn shows that with probability at least

$$\left(1 - 2d\exp\left(-\left(\rho/\gamma - \sum_{i=1}^{i=d} \bar{u}_i\right)^2 / \sigma^2\right)\right),$$

customers will not leave the platform because of irrelevant product recommendations. We let such latent attribute realizations be denoted by the event $\mathcal{E}_{relevant}$.

Step 3 (Sub-linearity of Regret): Recall, by definition, that

$$r_t(\rho, p, u_0) = (u_0^\top V_* - u_0^\top a_t)\mathbb{1}\{L_{t,\rho,p} = 1\} + u_0^\top V_* \mathbb{1}\{L_{t,\rho,p} = 0\} = (u_0^\top V_* - u_0^\top a_t) + u_0^\top a_t(1 - \Pi_{t=1}^\top \mathbb{1}\{\Upsilon_t = 0\})$$

Next, focusing on cumulative regret and taking expectation over the random customer response on quality feedback (ratings), we have that,

$$\begin{aligned}
\mathbb{E}_{U_0\sim\mathcal{P}}\left[\mathcal{R}^{CB}(T,\rho,p,u_0)\right] &= \mathbb{E}_{U_0\sim\mathcal{P}}\left[\sum_{t=1}^{T} r_t(\rho,p,u_0)\right] \\
&\leq \mathbb{E}\left[\sum_{t=1}^{\top}(u_0^\top V_* - u_0^\top a_t) + u_0^\top a_t(1 - \Pi_{t=1}^\top \mathbb{1}\{\Upsilon_t = 0\})\right] \\
&= \sum_{t=1}^{T}\mathbb{E}\left[(u_0^\top V_* - u_0^\top a_t)\right] + \mathbb{E}\left[u_0^\top a_t\left(1 - \Pi_{t=1}^\top \mathbb{1}\{\Upsilon_t = 0\}\right)\right].
\end{aligned}$$

Note that conditional on fraction w of customers, we have that these customers would never disengage from the platform due to irrelevant personalized recommendations. Hence, $1 - \Pi_{t=1}^\top \mathbb{1}\{\Upsilon_t = 0\} = 0$, Hence, $\mathcal{R}^{CB(\lambda,\gamma)}(T,\rho,p,u_0|u_0 \in \mathcal{E}_{relevant}) = \sum_{t=1}^{T}(u_0^\top V_* - u_0^\top a_t)$. Now notice that for any realization of u_0, Theorem 2 in Abbasi-Yadkori et al. (2011) shows that

$$\mathcal{R}^{CB(\lambda,\gamma)}(T,\rho,p,u_0|u_0 \in \mathcal{E}_{relevant}) \leq 4\sqrt{Td\,\log\left(\lambda + \frac{TL}{d}\right)}\left(\sqrt{\lambda}S + \xi\sqrt{2\log\frac{1}{\delta} + d\log\,(1 + \frac{TL}{\lambda d})}\right),$$

with probability at least 1-δ if $\|u_0\|_2 \leq S$. From Step 2, we have that all w fraction of customers have $\|u_0\|_2 \leq \frac{\rho}{\gamma}$. Hence first we replace S with $\frac{\rho}{\gamma}$. Finally, letting $\delta = \frac{1}{\sqrt{T}}$, we get that

$$\begin{aligned}
&\mathcal{R}^{CB(\lambda,\gamma)}(T,\rho,p,u_0|u_0 \in \mathcal{E}_{relevant}) \leq \\
&4\sqrt{Td\log\left(\lambda + \frac{TL}{d}\right)}\left(\sqrt{\lambda}\frac{\rho}{\gamma} + \xi\sqrt{\log(T) + d\log\left(1 + \frac{TL}{\lambda d}\right)}\right) + \frac{1}{\sqrt{T}}T \\
&= \tilde{\mathcal{O}}\left(\sqrt{T}\right).
\end{aligned}$$

Rearranging the terms above gives the final answer. □

B.3 Selecting set diameter γ

In the previous section, we proved that the Constrained Bandit algorithm achieves sublinear regret for a large fraction of customers. This fraction depends on the constrained threshold

tuning parameter γ and other problem parameters (see Theorem 3.5.5). In this section, we explore this dependence in more detail and provide intuition on the selection of γ that maximizes this .

Recall, from Theorem 3.5.5, that the fraction of customers who remain engaged with the platform is lower bounded by w. This fraction comprises of two parts.

The first part, $\left(1 - 2d \exp\left(-\left(1 - \sqrt{1-\gamma^2/4}\right)/\sigma\right)\right)$, denotes the fraction of customers for which the corresponding optimal product is contained in the constrained exploration set, Ξ. Notice that the fraction of customers for which the optimal product is contained in the constrained set increases as the constraint threshold, γ, increases. This follows since a larger γ implies a larger exploration set and more customer that can be served with their most relevant recommendation. Similarly, the second part, $\left(1 - 2d \exp\left(-\left(\rho/\gamma - \sum_{i=1}^{i=d} \bar{u}_i\right)^2/\sigma^2\right)\right)$, denotes the fraction of customers who will not disengage from the platform due to irrelevant recommendations in the learning phase. Contrary to the previous case, as the constraint threshold γ increases, the fraction of customers guaranteed to engage decreases. Intuitively, as the exploration set becomes larger, there is wider range of offerings with more variability in the relevance of the recommendations for a particular customer. This wider relevance in turn leads to a decrease in the probability of engagement of a customer. Hence, γ can either increase or decrease the fraction of engaged customers based on the other problem parameters.

In Figure B.1, we plot the fraction of customers who will remain engaged with the platform as a function of the set diameter, γ, for different values of tolerance threshold, ρ. As noted earlier, the fraction of engaged customers is not monotonically increasing in γ. When γ is small, the constrained set for exploration (from Step 1 of Algorithm 2) is over constrained. Hence, increasing γ leads to an increase in the fraction of engaged customers. Nevertheless, increasing it above a threshold implies that customers are more likely to disengage from the platform due to irrelevant recommendations. Hence, increasing γ further leads to a decrease in the fraction of engaged customers. We also note that as customers become less quality concious (small ρ), the fraction of engaged customers increases for any chosen value of γ. This again follows from the fact that a higher value of ρ implies a higher probability of customer engagement in the learning phase. This increase in engagement probability during the learning phase encourages less conservative exploration (larger γ). The above discussion alludes to the fact that the optimal γ that maximizes the fraction of engaged customers is a function of different problem parameters and is hard to optimize in general. Nevertheless, simple algebra

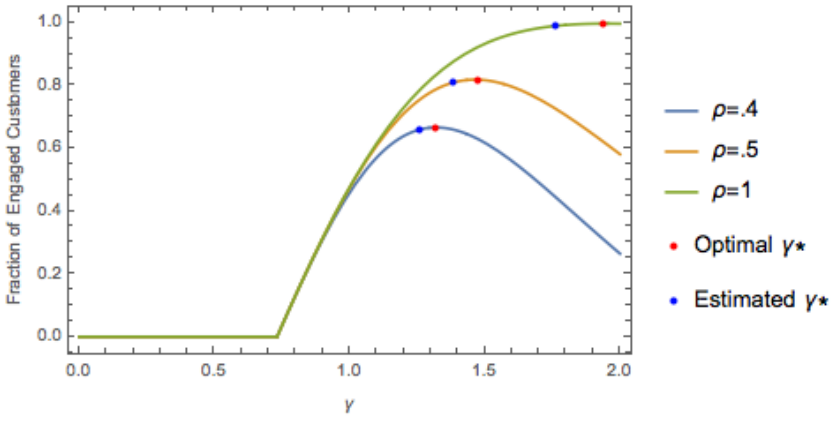

Figure B.1: Fraction of engaged customer as a function of the set diameter γ for different values of tolerance threshold, ρ. A higher ρ implies that the customer is less quality conscious. Hence, for any γ, this ensures higher chance of engagement. We also plot the optimal γ that ensures maximum engagement and an approximated γ that can be easily approximated. The approximated γ is considerably close to the optimal γ and ensures high level of engagement.

yields that $w \approx \frac{1}{2d^2} - \frac{1}{2d}\exp\left(-\frac{1}{\sigma}\left(1-\sqrt{1-\frac{\gamma^2}{4}}\right)\right) - \frac{1}{2d}\exp\left(-\frac{1}{\sigma^2}\left(\frac{\rho}{\gamma} - \sum_{i=1}^{i=d}\bar{u}_i\right)^2\right)$. Hence, in order to maximize w, we have to solve the following minimization problem:

$$\min_{\gamma} \ \exp\left(-\left(1-\sqrt{1-\gamma^2/4}\right)/\sigma\right) + \exp\left(-\left(\rho/\gamma - \sum_{i=1}^{i=d}\bar{u}_i\right)^2/\sigma^2\right). \tag{B.2}$$

While Problem (B.2) has no closed form solution, we consider the following problem:

$$\min_{\gamma} \ 1/\sigma\sqrt{(1-\gamma^2/4)} - \rho^2/(\gamma^2\sigma^2). \tag{B.3}$$

Note that (B.3) is an approximation of (B.2) based on the Taylor series expansion of the exponent function and assuming that the joint term in the second exponent will be sufficiently small. Solving (B.3) using FOC conditions, a suitable choice of γ yields the following: $\gamma^* \in \left\{\gamma : \rho = \frac{\sqrt{\sigma}\gamma^2}{2(4-\gamma^2)^{1/4}} \text{ and } \gamma > 0\right\}$. While γ^* is not optimal, it provides directional insights to managers on suitable choices of γ. For example, as ρ increases the estimated optimal γ also increases. Furthermore, it decreases with the prior variance, σ. A lower variance yields better understanding of the unknown customer and leads to lower size of the optimal exploration set. Similarly, as the latent vector dimension, d, increases, there are higher chances of not satisfying customer relevance thresholds in the learning phase. This leads to a more constrained exploration.

In order to analyze the estimated optimal γ, we compare the estimated optimal γ with the numerically calculated optimal γ for different values of ρ, the customer tolerance threshold. In Table B.1, we show the gap in the lower bound of engaged customers from choosing the optimal

γ vs the estimated γ. Note that the approximated optimal γ performs well in terms of the fraction of engaged customers. More specifically, the estimated γ loses at most 1% customers because of the approximation.

Tolerance Threshold (ρ)	***Optimal γ^****	***Estimated γ^****	***% Gap in Engagement***
0.4	1.31	1.25	1.1%
0.5	1.47	1.37	1.1%
1.0	1.93	1.76	0.07%

Table B.1: Optimal vs. estimated γ threshold for different values of customer tolerance threshold, ρ. Note that the % gap between the lower bound on engaged customers is below 1.1% showing that the estimated γ is near optimal.

B.4 Results for extensions of the disengagement model

We now discuss how our results extend under the alternative disengagement model described in §3.3.2:

$$Pr[\Upsilon_t = 1 \mid a_t] = \begin{cases} 0 & \text{if } u_0^\top a_t \geq \tilde{\rho}, \\ p(t, u_0, a_1, \ldots a_t) & \text{otherwise.} \end{cases}$$

Recall that $p(t, u_0, a_1, \ldots a_t) \geq \tilde{c} > 0$ for all $t, u_0, \{a_i\}_{i=1}^t$.

Proof. Proof of Theorem 3.4.3: We use the same setting as before with two products, whose feature vectors are the basis vectors in $\mathbb{R}^2$, and customer feature vectors $u_0 \sim \mathcal{N}(0, \sigma^2 I_2)$. At $t = 1$, any policy π has to either recommend product 1 or product 2. Note that when product 1 is recommended, the customer disengages immediately with probability at least $\tilde{c} \cdot \mathbb{P}(u_{0_1} \leq \tilde{\rho}) > 0$. The rest of the argument follows as before.

Proof. Proof of Theorem 3.4.4: We use the same setting as before and let $\tilde{\rho}$ be large enough so that at least one product i is not acceptable to the customer, *i.e.,* $U_0^\top V_i < \tilde{\rho}$. As before, it then follows that bandit algorithms offer product i infinitely often. Since $p(t, u_0, a_1, \ldots a_t) \geq \tilde{c} > 0$, the customer eventually disengages from the platform with probability 1. The rest of the argument follows as before.

Proof. Proof of Theorem 1: Consider any feasible value of $\tilde{\rho}$ and let $\tilde{\gamma}$ be such that only a single product remains in the constrained exploration set. The rest of the proof follows identically as before: there is some subset of customers (with positive measure under $\mathcal{P}$) for whom this product is relevant/optimal, yielding 0 regret.

Proof. Proof of Theorem 2: Steps 1 and 3 follow identically as before. We focus on Step 2, which characterizes the probability of disengagement due to poor recommendations. Let $C_1 = \frac{d^2}{2\sigma}\left(\gamma + \sqrt{\gamma^2 + 4\tilde{\rho}}\right)$. Then,

$$\begin{aligned}
\mathbb{P}(u_0^\top a_i \geq \tilde{\rho}) &= \mathbb{P}(u_0^\top u_0 - u_0^\top u_0 + u_0^\top u_i \geq \tilde{\rho}) = \mathbb{P}(u_0^\top (u_0 - u_i) < u_0^\top u_0 - \tilde{\rho}) \\
&\geq \mathbb{P}\left(\|u_0\|_2 < \frac{u_0^\top u_0 - \tilde{\rho}}{\gamma} \mid u_0, u_i \in \Xi\right) \\
&= \mathbb{P}\left(\|u_0\|_2^2 - \gamma\|u_0\|_2 - \tilde{\rho} > 0 \mid u_0, u_i \in \Xi\right) \\
&= \mathbb{P}\left(2\|u_0\|_2 \geq \gamma + \sqrt{\gamma^2 + 4\tilde{\rho}} \mid u_0, u_i \in \Xi\right) \\
&\geq \mathbb{P}\left(2\|u_0\|_1 \geq \sqrt{d}\left(\gamma + \sqrt{\gamma^2 + 4\tilde{\rho}}\right) \mid u_0, u_i \in \Xi\right) \\
&\geq 2\mathbb{P}\left(2u_0^1 \geq \sqrt{d}\left(\gamma + \sqrt{\gamma^2 + 4\tilde{\rho}}\right) \mid u_0, u_i \in \Xi\right) \\
&\geq 2\mathbb{P}\left(2d(u_0^1 - \bar{u}^1) \geq d^2\left(\gamma + \sqrt{\gamma^2 + 4\tilde{\rho}}\right) \mid u_0, u_i \in \Xi\right) \\
&\geq \frac{1}{\sqrt{2\pi}C_1}\exp\left(-C_1^2/2\right),
\end{aligned}$$

where the last inequality follows by the lower bound on tail probabilities of standard normal random variables. Hence, the probability of engagement changes to

$$\mathbb{P}(\mathcal{W}) \geq w \;=\; \left(1 - 2d\exp\left(-\left(1 - \sqrt{1 - \gamma^2/4}\right)/\sigma\right)\right)\left(\frac{1}{\sqrt{2\pi}C_1}\exp\left(\frac{-C_1^2}{2}\right)\right).$$

The regret guarantee remains the same.

Returning Customers: As noted earlier, in some settings, customers may disengage *temporarily* rather than for the entire horizon T. With slight abuse of notation, let Υ_t denote the total time of disengagement due to the recommendation at time t. Then, we can propose the following customer disengagement model:

$$\Upsilon_t \mid a_t = \begin{cases} 0 & \text{if } \; u_0^\top a_t \;\geq\; \tilde{\rho}\,, \\ f(t, u_0, a_1, \ldots a_t) & \text{otherwise}\,, \end{cases}$$

where $f(t, u_0, a_1, \ldots a_t)$ denotes the *total* time that the customer is disengaged due to all recommendations made until time t. Consider the case where $f(t, u_0, a_1, \ldots a_t) = T^\delta$, for some $\delta \leq 1$. Our previous models imposed $\delta = 1$ (customer does not return for remaining horizon), while $\delta \to -\infty$ models the classical bandit setting with no disengagement. Our results can be

straightforwardly extended to this more general setting. First, we can show a lower bound on regret for all consistent policies is $\widetilde{O}(T^{\max\{\frac{1}{2},\min\{1,\delta+\frac{1}{2}\}\}})$. In other words, one can safely use classical bandit policies without any loss when customers disengage for no more than a constant period of time (*i.e.,* $\delta \leq 0$); however, when poor recommendations can result in substantial periods of disengagement (*i.e.,* $\delta > 0$), we can show that constraining exploration is strictly better than using classical bandit algorithms. Particularly, while classical bandit algorithms would risk temporary disengagement of *all* customers, a modification of the constrained bandit policy would incur $\widetilde{O}(\sqrt{T})$ regret for the fraction of customers for whom the constrained set of products is *relevant*, and match the performance of classical bandit algorithms on the remaining customers. We skip the details of this analysis due to space constraints.

B.5 Supplementary Results

Lemma B.5.1. Let $X \in \mathbb{R}^d \sim \mathcal{N}(\mu, \sigma^2 I)$ be a multivariate normal random variable with mean vector $\mu \in \mathbb{R}^d$. Let $S \in \mathbb{R}^d$ be such that $S \geq \sum_{i=1}^{i=d} \mu_i$. Then, $\mathbb{P}(\|X\|_1 \leq S) \geq 1 - 2d \exp\left(-\left(\frac{S-\sum_{i=1}^{i=d}\mu_i}{d\sigma}\right)^2\right)$

Proof. Proof: The proof follows from simple application of the Pigeon Hole Principle and tail bounds on multivariate normal variables. We skip the details for the sake of brevity. □

Appendix C

Appendix of Chapter 4

C.1 Summary of Notation

Variables	***Description***
T	Total time horizon.
μ	Price dispersion parameter.
ρ	Price depth parameter.
$\mathbf{P}$	Vector of tuples where the first entry of the tuple is a price and the second entry is the average of demand observations at that price.
$\mathbf{G}$	Vector of gradients estimated using finite differences from demand observations that satisfy the two point Bandit feedback assumption.
p_i^L	Minimum price that is used for experimentation in round i.
p_i^H	Maximum price that is used for experimentation in round i.
p_i^M	Average of the minimum and the maximum price that is used for experimentation in round i.
$\tilde{p}_i^*$	Optimal price approximated from the linearly-interpolated demand in round i.
Δ_i	Size of the linear interpolation for demand estimation in round i.

Table C.1: Notation Table

C.2 Proofs from §4.2.3

Proof. Proof of Lemma 4.2.3: We have that

$$
\begin{aligned}
& d(p^*) - \kappa(p - p^*)^2 \leq -d'(p)p^* \\
\implies & d(p^*) + p^* d'(p^*) - \kappa(p - p^*)^2 \leq -d'(p)p^* + p^* d'(p^*) \\
\implies & d'(p)p^* - p^* d'(p^*) \leq \kappa(p - p^*)^2 - (d(p^*) + p^* d'(p^*)) \\
\implies & p^* \left(d'(p) - d'(p^*)\right) \leq \kappa(p - p^*)^2 ,
\end{aligned}
$$

where the last inequality follows because p^* is the revenue maximizing price. That is, $d(p^*) + p^* d'(p^*) = 0$. This proves the final result. □

Proof. Proof of Lemma 4.2.4: The proof follows in two steps. We first show using Lipschitz continuity that the following holds:

$$r'(p_1) - r'(p_2) \leq r''(p_2)(p_1 - p_2) + \Psi(p_1 - p_2)^2 \,, \tag{C.1}$$

and

$$d(p^*) - d(p) \leq d'(p)(p^* - p) + \bar{\Psi}(p^* - p)^2 \,. \tag{C.2}$$

We focus on (C.1) first and note by assumption that $\forall p_1, p_2 \in [0, 1]$

$$r''(p_1) - r''(p_2) \leq \Psi(p_1 - p_2)^2 \,.$$

Now let $g(t) := r'(p_1 + t(p_2 - p_1)), \forall t \in [0, 1]$. Furthermore,

$$g'(t) - g'(0) = \left(r''(p_1 + t(p_2 - p_1)) - r''(p_1)\right)(p_2 - p_1) \leq t\Psi(p_1 - p_2)^2 \,.$$

Hence, integrating from $t = 0$ to $t = 1$,

$$\begin{aligned} r'(p_2) = g(1) &= g(0) + \int_0^1 g'(t)dt \\ &\leq g(0) + g'(0) + g'(0)\frac{\psi}{2}(p_2 - p_1)^2 \\ &= r'(p_1) + r'(p_1)(p_2 - p_1) + \frac{\psi}{2}(p_2 - p_1)^2 \,. \end{aligned}$$

Interchanging p_1 with p_2 and p_2 with p_1 gives the final result. An identical argument can be used for proving (C.2).

In the second step, we use these conditions with respect to the unknown optimal price and

any other price. In particular, letting $p_1 = p$ and $p_2 = p^*$, we get that

$$\begin{aligned}
& r'(p) - r'(p^*) \leq r''(p^*)(p - p^*) + \Psi(p - p^*)^2 \,, \\
\implies & pd'(p) + d(p) - (p^*d'(p^*) + d(p^*)) \leq r''(p^*)(p - p^*) + \Psi(p - p^*)^2 \\
\implies & p^*d'(p) - p^*d'(p) + pd'(p) + d(p) - (p^*d'(p^*) + d(p^*)) \leq r''(p^*)(p - p^*) + \Psi(p - p^*)^2 \\
\implies & p^*(d'(p) - d'(p^*)) \leq -d'(p)(p - p^*) + d(p^*) - d(p) + r''(p^*)(p - p^*) + \Psi(p - p^*)^2 \,.
\end{aligned}$$

where we have used the definition of $r'(p) = pd'(p) + d(p)$. Next, note that $r''(p) = 2d'(p) + p^*d''(p)$. And also by (C.2), we have that

$$d(p^*) - d(p) \leq d'(p)(p^* - p) + \bar{\Psi}(p^* - p)^2 \,.$$

Hence, replacing $r''(p^*)$, and using above, we have that

$$\begin{aligned}
& p^*(d'(p) - d'(p^*)) \leq \\
& - d'(p)(p - p^*) + d'(p)(p^* - p) + \bar{\Psi}(p^* - p)^2 + (2d'(p^*) + p^*d''(p^*))(p - p^*) + \Psi(p - p^*)^2 \\
\implies & p^*(d'(p) - d'(p^*)) \leq 2(p - p^*)(d'(p) - d'(p^*)) + p^*d''(p^*)(p - p^*) + (\Psi + \bar{\Psi})(p^* - p)^2 \\
\implies & p^*(d'(p) - d'(p^*)) \leq 2(p - p^*)(d'(p) - d'(p^*)) + (\Psi + \bar{\Psi})(p^* - p)^2 \,,
\end{aligned}$$

where the last inequality follows by the assumption that $d''(p^*) = 0$. Finally, since d' is assumed to be continuous and differentiable, by a direct application of the Mean Value Theorem, we have that

$$d'(p) - d'(p^*) \leq K_1(p - p^*) \,,$$

where $K_1 = \max_{i \leq w, p \in [0,1]} |d^{(i)}(p)|$. Hence,

$$\begin{aligned}
p^*(d'(p) - d'(p^*)) &\leq 2K_1(p - p^*)^2 + (\Psi + \bar{\Psi})(p^* - p)^2 \\
&= (\Psi + \bar{\Psi} + 2K_1)(p^* - p)^2 \,.
\end{aligned}$$

This proves the final result. □

C.3 Proofs of results from §4.4.1

Proof. Proof of Lemma 4.4.2: We prove this result using a direct application of triangular

inequality and Taylor series expansion of the unknown demand function around the $\frac{p_i^M+p_i^L}{2}$.

Consider (4.9) and note that

$$\begin{aligned}
&\left|\frac{D_{M_1^*}(p_i^M)-D_{L^*}(p_i^L)}{p_i^M-p_i^L}-d'\left(\frac{p_i^M+p_i^L}{2}\right)\right| = \\
&\left|\frac{D_{M_1^*}(p_i^M)-D_{L^*}(p_i^L)}{p_i^M-p_i^L}-d'\left(\frac{p_i^M+p_i^L}{2}\right)+\frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L}-\frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L}\right| \\
&\leq \underbrace{\left|\frac{D_{M_1^*}(p_i^M)-D_{L^*}(p_i^L)}{p_i^M-p_i^L}-\frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L}\right|}_{\text{A}}+\underbrace{\left|\frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L}-d'\left(\frac{p_i^M+p_i^L}{2}\right)\right|}_{\text{B}}.
\end{aligned} \tag{C.3}$$

We will bound (A) and (B) in (C.3) separately. In particular, while (A) corresponds to error due to stochastic realizations, (B) corresponds to error due to finite difference approximation of the gradient. In what follows, we will suppress the dependence on i for ease of notation.

Step 1: Bounding error due to stochastic realizations: Recall, by Assumption (4.2.5) that,

$$D_{M_1^*}(p^M)=d(p^M)+\epsilon^* \;\;\&\;\; D_{L^*}(p^L)=d(p^L)+\epsilon^*\,.$$

In particular the error in the two pairs of demand observations is the same. Hence,

$$\frac{D_{M_1^*}(p^M)-D_{L^*}(p^L)}{p^M-p^L}-\frac{d(p^M)-d(p^L)}{p^M-p^L}=\frac{d(p^M)+\epsilon^*-(d(p^L)+\epsilon^*)}{p^M-p^L}-\frac{d(p^M)-d(p^L)}{p^M-p^L}=0\,.$$

Step 2: Bounding error due to linear interpolation: To bound $\left|\frac{d(p^M)-d(p^L)}{p^M-p^L}-d'\left(\frac{p^M+p^L}{2}\right)\right|$, consider the Taylor series expansion of $d(p^M)$ around $\frac{p^M+p^L}{2}$. Let $p_{mid}=\frac{p^M+p^L}{2}$. Then, we have that

$$d(p^M)=d(p_{mid})+d'(p_{mid})\frac{\Delta_i}{2}+\frac{1}{2}d''(p_{mid})\frac{\Delta_i^2}{4}+\frac{1}{6}d'''(p_{mid})\frac{\Delta_i^3}{8}+\ldots$$

Similarly, considering the Taylor series expansion of $d(p^L)$ around p_{mid},

$$d(p^L)=d(p_{mid})-d'(p_{mid})\frac{\Delta_i}{2}+\frac{1}{2}d''(p_{mid})\frac{\Delta_i^2}{4}-\frac{1}{6}d'''(p_{mid})\frac{\Delta_i^3}{8}+\ldots$$

Hence, we get that

$$\left|\frac{d(p^M)-d(P^L)}{\Delta_i}-d'(p_{mid})\right|\leq K_1\frac{\Delta_i^2}{24}\,,$$

where $K_1=\max_{i\leq q,p\in[0,1]}|d^{(i)}(p)|$, and recall that $d^{(i)}(p)$ denotes the i^{th} derivative of demand

at any price p.

Finally, substituting back in (C.3), we have that

$$\left|\frac{D_{H^*}(p_i^H) - D_{M_2^*}(p_i^M)}{\Delta_i} - d'\left(\frac{p_i^M + p_i^H}{2}\right)\right| \leq \frac{1}{2}K_1\frac{\Delta_i^2}{12}\,.$$

The proof for (4.10) follows identically and hence we skip the details for the sake of brevity. □

C.4 SLPE-Extended Algorithm to account for relaxed assumptions:

In §4.4.1 we discussed various relaxations of the two-point bandit feedback assumption and claimed that an extension of the SLPE algorithm with similar ideas can be implemented to get comparable guarantees on regret and price changes. In this section, we make these statements more precise.

First we present the Extended SLPE algorithm (Algorithm 5) with the relaxed assumptions (2A). Instead of using demand observations that satisfy two point bandit feedback throughout the pricing space, the algorithm splits the pricing decisions in two different phases. In the first phase, when the size of interpolation is large in comparison to the standard deviation of the error ($\sigma \leq \Delta$), we use the average demand at each selected price point to estimate the region of the optimal price. Then, when the feasible pricing region becomes small, making the overall interpolation region small as well ($\Delta < \sigma$), we switch to using demand realizations that satisfy the two point bandit feedback assumption, to estimate the gradient of demand.

C.4.1 Theoretical Guarantees

The proofs follow similar intuition as before. In particular, we will first show that the approximated optimal price can be used to estimate the region containing the optimal price with high probability. This translates to bounded regret in each round of SLPE-Ext and in-turn leads to bounded regret with very limited price change.

We start by showing that the point wise gradient estimate of demand at any price is a *good* estimate.

Lemma C.4.1. Consider the three experimental prices p_i^L, p_i^M and p_i^H of prices experimented in round i of Algorithm (5). Assume that $f(n)$ described in Assumption 4.4.1 is such that

Algorithm 5 SLPE-Ext(T,μ,ρ,$f(n)$)

Let $p_1^L = 0$, $p_1^H = 1$, $i = 1$, $t = 0$, $\Delta_1 = .5$, $\mathbf{P} = \{\}$, $\mathbf{G} = \{\}$ and $i = 1$.
while $t \leq T$ **do**
  if $\sigma \leq \Delta_i$ **then**
    Let $n_i = 2\rho^4 \frac{\log(T)}{\Delta_i^4}$ and $t = t + 3n_i$.
    Price for n_i rounds each at p_i^L, $p_i^M = \frac{p_i^L + p_i^H}{2}$ and p_i^H, respectively.
    Let $\mathbf{P} = \{(p_i^L, \bar{D}_{n_i}(p_i^L)), (p_i^M, \bar{D}_{n_i}(p_i^M)), (p_i^H, \bar{D}_{n_i}(p_i^H))\}$
    Let $\mathbf{G} = \left\{\frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{\Delta_i}, \frac{\bar{D}_{n_i}(p_i^H) - \bar{D}_{n_i}(p_i^M)}{\Delta_i}\right\}$.
    Optimize over piecewise-linear demand estimate with $\mathbf{P}$ and $\mathbf{G}$ to get $\tilde{p}_i^*$.
    Let $p_{i+1}^L = \tilde{p}_i^* - \mu\Delta_i^2$, $p_{i+1}^H = \tilde{p}_i^* + \mu\Delta_i^2$, $\Delta_{i+1} = \mu\Delta_i^2$ and *i=i+1*.
  if $\sigma > \Delta_i$ **then**
    Let $n_i = 2\rho^4 \frac{\log(T)}{\Delta_i^4}$ and $t = t + 3n_i$.
    Price for n_i rounds each at p_i^L, $p_i^M = \frac{p_i^L + p_i^H}{2}$ and p_i^H, respectively.
    Let $\mathbf{P} = \{(p_i^L, \bar{D}_{n_i}(p_i^L)), (p_i^M, \bar{D}_{n_i}(p_i^M)), (p_i^H, \bar{D}_{n_i}(p_i^H))\}$
    Let $\mathbf{G} = \left\{\frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{\Delta_i}, \frac{D_{H^*}(p_i^H) - D_{M_2^*}(p_i^M)}{\Delta_i}\right\}$.
    Optimize over piecewise-linear demand estimate with $\mathbf{P}$ and $\mathbf{G}$ to get $\tilde{p}_i^*$.
    Let $p_{i+1}^L = \tilde{p}_i^* - \mu\Delta_i^2$, $p_{i+1}^H = \tilde{p}_i^* + \mu\Delta_i^2$, $\Delta_{i+1} = \mu\Delta_i^2$ and *i=i+1*.

$\delta > 3/4$. Then, for any round i such that $\sigma \leq \Delta_i$,

$$\left| \frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right) \right| \leq \left(\frac{K_1}{24} + \frac{2}{\rho^2}\right)\Delta_i^2\,, \tag{C.4}$$

and

$$\left| \frac{\bar{D}_{n_i}(p_i^H) - \bar{D}_{n_i}(p_i^M)}{\Delta_i} - d'\left(\frac{p_i^M + p_i^H}{2}\right) \right| \leq \left(\frac{K_1}{24} + \frac{2}{\rho^2}\right)\Delta_i^2\,. \tag{C.5}$$

Similarly, for all rounds i such that $\sigma > \Delta_i$,

$$\left| \frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right) \right| \leq \Delta_i^2\left(\frac{K_1}{24} + \frac{1}{\rho^2}\right)\,, \tag{C.6}$$

and

$$\left| \frac{D_{H^*}(p_i^H) - D_{M_2^*}(p_i^M)}{\Delta_i} - d'\left(\frac{p_i^M + p_i^H}{2}\right) \right| \leq \Delta_i^2\left(\frac{K_1}{24} + \frac{1}{\rho^2}\right)\,, \tag{C.7}$$

where $K_1 = \max_{i \leq q, p \in [0,1]} |d^{(i)}(p)|$, and $d^{(i)}(p)$ denotes the i^{th} derivative of demand at any price p. Furthermore, $(D_{L^*}(p_i^L), D_{M_1^*}(p_i^M))$ and $(D_{M_2^*}(M_i), D_{H^*}(H_i))$ are pair of demand realizations that satisfy Assumption (4.2.5).

Proof. Proof: We prove this result using a direct application of triangular inequality, Taylor

series expansion of the unknown demand function around the $\frac{p_i^M+p_i^L}{2}$, and Hoeffding's inequality for tail bounds on sub-gaussian random variables.

Consider the case when $\sigma \leq \Delta_i$. We will start by (C.4) and note that

$$\begin{aligned}
&\left|\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L} - d'\left(\frac{p_i^M+p_i^L}{2}\right)\right| = \\
&\left|\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L} - d'\left(\frac{p_i^M+p_i^L}{2}\right) + \frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L} - \frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L}\right| \\
&\leq \underbrace{\left|\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L} - \frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L}\right|}_{\text{A}} + \underbrace{\left|\frac{d(p_i^M)-d(p_i^L)}{p_i^M-p_i^L} - d'\left(\frac{p_i^M+p_i^L}{2}\right)\right|}_{\text{B}}.
\end{aligned} \tag{C.8}$$

We will bound (A) and (B) in (C.8) separately. In particular, while (A) corresponds to error due to stochastic realizations, (B) corresponds to error due to finite difference approximation of the gradient. In what follows, we will suppress the dependence on i for ease of notation.

Step 1: Bounding error due to stochastic realizations: Recall, by Assumption (4.2.5) that,

$$\bar{D}_{n_i}(p_i^M) = d(p^M) + \frac{1}{n_i}\sum_{i=1}^{n_i}\epsilon_i \quad \& \quad \bar{D}_{n_i}(p_i^L) = d(p^L) + \frac{1}{n_i}\sum_{i=1}^{n_i}\tilde{\epsilon}_i\,.$$

Hence,

$$\begin{aligned}
&\frac{\bar{D}_n(p^M)-\bar{D}_n(p^L)}{p^M-p^L} - \frac{d(p^M)-d(p^L)}{p^M-p^L} = \\
&\frac{1}{p^M-p^L}\left(d(p^M) + \frac{1}{n}\sum_{i=1}^{n}\epsilon_i - \left(d(p^L) + \frac{1}{n_i}\sum_{i=1}^{n}\tilde{\epsilon}_i\right)\right) - \frac{d(p^M)-d(p^L)}{p^M-p^L} \\
&= \frac{1}{n(p^M-p^L)}\left(\sum_{i=1}^{n}\epsilon_i - \sum_{i=1}^{n}\tilde{\epsilon}_i\right)
\end{aligned}$$

Hence,

$$\left|\frac{\bar{D}_n(p^M)-\bar{D}_n(p^L)}{p^M-p^L} - \frac{d(p^M)-d(p^L)}{p^M-p^L}\right| \leq \frac{1}{n(p^M-p^L)}\left|\sum_{i=1}^{n}\epsilon_i - \sum_{i=1}^{n}\tilde{\epsilon}_i\right| \leq \frac{2}{n(p^M-p^L)}\left|\sum_{i=1}^{n}\epsilon_i\right|.$$

Next, we have by a direct application of Hoeffding's inequality that for any x

$$\mathbb{P}\left(\frac{1}{n}\left|\sum_{i=1}^{n}\epsilon_i\right| > x\right) \leq 2\exp\left(-\frac{nx^2}{2\sigma^2}\right).$$

Hence, letting $n = 2\frac{\rho^4}{\Delta^4}\log(T)$ and $x = \frac{\sigma\Delta^2}{\rho^2}$, we have that

$$\mathbb{P}\left(\frac{1}{n}\left|\sum_{i=1}^{n}\epsilon_i\right| > \frac{\sigma\Delta^2}{\rho^2}\right) \leq \frac{1}{T^2}\,.$$

This implies that with probability at least $1 - \frac{1}{T^2}$,

$$\left|\frac{\bar{D}_n(p^M) - \bar{D}_n(p^L)}{p^M - p^L} - \frac{d(p^M) - d(p^L)}{p^M - p^L}\right| \leq \frac{2}{n(p^M - p^L)}\left|\sum_{i=1}^{n}\epsilon_i\right| = \frac{2\sigma\Delta^2}{\Delta\rho^2} \leq \frac{2\Delta^2}{\rho^2}\,,$$

where the last inequality follows because $\sigma \leq \Delta$.

Step 2: Bounding error due to linear interpolation: To bound $\left|\frac{d(p^M)-d(p^L)}{p^M-p^L} - d'\left(\frac{p^M+p^L}{2}\right)\right|$, consider the Taylor series expansion of $d(p^M)$ around $\frac{p^M+p^L}{2}$. Let $p_{mid} = \frac{p^M+p^L}{2}$. Then, we have that

$$d(p^M) = d(p_{mid}) + d'(p_{mid})\frac{\Delta_i}{2} + \frac{1}{2}d''(p_{mid})\frac{\Delta_i^2}{4} + \frac{1}{6}d'''(p_{mid})\frac{\Delta_i^3}{8} + \dots$$

Similarly, considering the Taylor series expansion of $d(p^L)$ around p_{mid},

$$d(p^L) = d(p_{mid}) - d'(p_{mid})\frac{\Delta_i}{2} + \frac{1}{2}d''(p_{mid})\frac{\Delta_i^2}{4} - \frac{1}{6}d'''(p_{mid})\frac{\Delta_i^3}{8} + \dots$$

Hence, we get that

$$\left|\frac{d(p^M) - d(P^L)}{\Delta_i} - d'(p_{mid})\right| \leq K_1\frac{\Delta_i^2}{24}\,,$$

where $K_1 = \max_{i\leq w, p\in[0,1]}|d^{(i)}(p)|$, and recall that $d^{(i)}(p)$ denotes the i^{th} derivative of demand at any price p.

Finally, substituting back in (C.8), we have that

$$\left|\frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right)\right| \leq K_1\frac{\Delta_i^2}{24} + \frac{2\Delta_i^2}{\rho^2} = \Delta_i^2\left(\frac{K_1}{24} + \frac{2}{\rho^2}\right)\,.$$

The proof for (C.5) follows identically and hence we skip the details for the sake of brevity.

Next, we focus on the case when $\Delta_i \leq \sigma$ and start by considering (C.6)

$$\begin{aligned}
&\left|\frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right)\right| = \\
&\left|\frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right) + \frac{d(p_i^M) - d(p_i^L)}{p_i^M - p_i^L} - \frac{d(p_i^M) - d(p_i^L)}{p_i^M - p_i^L}\right| \qquad \text{(C.9)} \\
&\leq \underbrace{\left|\frac{D_{M_1^*}(p_i^M) - D_{L^*}(p_i^L)}{p_i^M - p_i^L} - \frac{d(p_i^M) - d(p_i^L)}{p_i^M - p_i^L}\right|}_{\text{A}} + \underbrace{\left|\frac{d(p_i^M) - d(p_i^L)}{p_i^M - p_i^L} - d'\left(\frac{p_i^M + p_i^L}{2}\right)\right|}_{\text{B}}.
\end{aligned}$$

We will bound (A) and (B) in (C.9) separately. In particular, while (A) corresponds to error due to stochastic realizations, (B) corresponds to error due to finite difference approximation of the gradient. In what follows, we will suppress the dependence on i for ease of notation.

Step 1: Bounding error due to stochastic realizations: Recall, by Assumption 2A that,

$$D_{M_1^*}(p^M) = d(p^M) + \epsilon_{m_n^*} \quad \& \quad D_{L^*}(p^L) = d(p^L) + \epsilon_{l_n^*}\,,$$

where $|\epsilon_{m_n^*} - \epsilon_{l_n^*}| \leq f(n)$. In particular the error in the two pairs of demand observations is the same. Hence,

$$\begin{aligned}
&\left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} - \frac{d(p^M) - d(p^L)}{p^M - p^L}\right| \\
&= \frac{d(p^M) + \epsilon_m^* - (d(p^L) + \epsilon_l^*)}{p^M - p^L} - \frac{d(p^M) - d(p^L)}{p^M - p^L} = \left|\frac{\epsilon_m^* - \epsilon_l^*}{\Delta}\right| \leq \frac{f(n)}{\Delta}\,.
\end{aligned}$$

Hence, if $f(n) = n^{-\delta}$, we get that

$$\left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} - \frac{d(p^M) - d(p^L)}{p^M - p^L}\right| \leq \frac{\Delta^{4\delta}}{\rho^{4\delta}\Delta} \leq \frac{\Delta^{4\delta-1}}{\rho^{4\delta}} \leq \frac{\Delta^2}{\rho^2}\,,$$

where in the last inequality, we have assumed that $\delta \geq 3/4$.

Step 2: Bounding error due to linear interpolation: This follows identically as the proof and we get that

$$\left|\frac{d(p^M) - d(P^L)}{\Delta_i} - d'(p_{mid})\right| \leq K_1 \frac{\Delta_i^2}{24}\,,$$

where K_1 was defined before. Finally, substituting back in (C.9), we have that

$$\left|\frac{D_{H^*}(p_i^H) - D_{M_2^*}(p_i^M)}{\Delta_i} - d'\left(\frac{p_i^M + p_i^H}{2}\right)\right| \leq K_1 \frac{\Delta_i^2}{24} + \frac{\Delta^2}{\rho^2} = \Delta_i^2\left(\frac{K_1}{24} + \frac{1}{\rho^2}\right).$$

The proof for (C.7) follows identically and hence we skip the details for the sake of brevity. Hence, this proves the final result. □

Lemma C.4.2. Consider the SLPE pricing policy of Algorithm 5 and let the unknown demand function be such that $|d'(p^*)| \geq \left(\frac{K_1}{24} + \frac{\kappa}{4} + \frac{3}{\rho^2}\right)\frac{1}{4} + \frac{c}{2}$, for some positive constant c. Then, for any round i,

$$|p^* - \tilde{p}_i^*| \leq M\Delta_i^2\,,$$

for $M = \left(\frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + 4\kappa\right)$ with probability at least $1 - \frac{1}{T^2}$, where p^* is the real unknown optimal price and $\tilde{p}_i^*$ is the approximated optimal price from the piecewise linear interpolated demand curve of round i.

Proof. Proof: The proof follows in two main steps. In the first step, we use the first order conditions to relate the error in the unknown optimal and the approximated optimal price to the estimation error in demand and gradient. Then in the second step, we bound the estimation error respectively. Since the algorithm is split in two cases depending on the size of the interpolation,

Step 1: Relating error in approximated optimal price with estimation error:

Let $g(p) := r'(p)$ be the first order equation of the unknown revenue function. Then, by the optimality of p^*, we have that $g(p^*) = 0$. Similarly, we have that $\tilde{p}^*$ is the estimated optimal price from the piecewise linear demand curve constructed using demand observations at p^L, p^M and p^H. In particular, recall that

$$\tilde{p}^* = \arg\max_{p\in[p^L,p^H]} p d^{est}(p)\,,$$

where we note that for all rounds such that $\sigma \leq \Delta_i$

$$d^{est}(p) := \begin{cases} \bar{D}(p^M) + \frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L}\left(p - p^M\right), \forall p \leq p^M\,, \\ \bar{D}(p^M) + \frac{\bar{D}_{n_i}(p_i^H) - \bar{D}_{n_i}(p_i^M)}{p_i^H - p_i^M}\left(p - p^M\right), \forall p > p^M\,. \end{cases}$$

And for all rounds such that $\sigma \geq \Delta_i$,

$$d^{est}(p) := \begin{cases} \bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(p - p^M\right), \forall p \leq p^M\,, \\ \bar{D}(p^M) + \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}\left(p - p^M\right), \forall p > p^M\,. \end{cases}$$

Recall by definition that $\tilde{p}^*$ is the approximated optimal price that is revenue maximizing for the approximated demand $d^{est}(p)$. Since $d^{est}(p)$ is a piecewise-linear function we have two cases to analyze: (i)$\tilde{p}^* \leq p^M$ or (ii) $\tilde{p}^* > p^M$. Assume without loss of generality that $\tilde{p}^* \leq p^M$. Since $\tilde{p}^*$ is the revenue maximizing price, it is a solution to the following (approximate) first order condition:

$$d^{est}(p) + d'^{est}(p)p = 0\,.$$

Hence $d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = 0$. In order to compare the approximated optimal price with the real optimal price, we evaluate the optimal price at the approximate first order condition. But note that the approximate first order condition is also a piecewise function and depends on the size of the interpolation. Hence, we have to analyze all rounds such that $\Delta_i \leq \sigma$ and then $\Delta_i \geq \sigma$. In each of these there are two cases to analyze: (i) if $p^* < p^M$ or (ii) $p^* \geq p^M$.

All rounds such that $\Delta_i > \sigma$:

Case (i) $p^* < p^M$: As before, consider the first order condition evaluated at p^*,

$$\begin{aligned} &d^{est}(p^*) + d'^{est}(p^*)p^* = \\ &\bar{D}(p^M) + \frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L}\left(p^* - p^M\right) + \frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L}p^*\,. \end{aligned} \tag{C.10}$$

Similarly,

$$\begin{aligned} &d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = \\ &\bar{D}(p^M) + \frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L}\left(\tilde{p}^* - p^M\right) + \frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L}p^M - p^L\tilde{p}^* = 0\,, \end{aligned} \tag{C.11}$$

where the last equality follows from the optimality of $\tilde{p}^*$ for the approximate demand. Hence subtracting (C.11) from (C.10), we have that:

$$d^{est}(p^*) + d'^{est}(p^*)p^* - (d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^*) = 2\left(\frac{\bar{D}_{n_i}(p_i^M) - \bar{D}_{n_i}(p_i^L)}{p_i^M - p_i^L}\right)(p^* - \tilde{p}^*)\,.$$

Also note that $g(p^*) = 0$. Hence, $d(p^*) + d'(p^*)p^* = 0$. Furthermore,

$$d^{est}(p^*)+d'^{est}(p^*)p^* = d^{est}(p^*)+d'^{est}(p^*)p^*-(d(p^*)+d'(p^*)p^*) = (d^{est}(p^*)-d(p^*))+(d'^{est}(p^*)-d'(p^*))p^*\,.$$

Hence, combining the two above, we get that

$$
\begin{aligned}
&2\left(\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)(p^*-\tilde{p}^*) = (d^{est}(p^*)-d(p^*)) + (d'^{est}(p^*)-d'(p^*))p^* \\
&2\left|\left(\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)\right||p^*-\tilde{p}^*| \le |d^{est}(p^*)-d(p^*)| + |d'^{est}(p^*)-d'(p^*)|p^* \qquad \text{(C.12)} \\
\implies &|p^*-\tilde{p}^*| \le \frac{1}{2|d'^{est}(p^*)|}\left(\underbrace{|d^{est}(p^*)-d(p^*)|}_{\text{A}} + \underbrace{|d'^{est}(p^*)-d'(p^*)|p^*}_{\text{B}}\right),
\end{aligned}
$$

where $d'^{est}(p^*) := \left(\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)$. Hence, to bound the estimation error in the optimal price, we need to bound terms (A) and (B). Recall by definition that $d^{est}(p^*)$ and $d(p^*)$ denote the estimated demand at the optimal price and the real unknown demand at the optimal price, respectively. Hence, (A) denotes the estimation error in demand at the optimal price. Similarly, $d'^{est}(p^*)$ and $d'(p^*)$ denote the approximate and the real unknown gradient of demand at the optimal price. Therefore, (B) denotes the estimation error in the gradient. In what follows, we will bound both these errors.

Step 2: Bounding estimation error in demand and gradient:

We proceed by independently bounding (A) and (B) from (4.13).

Bounding $|d^{est}(p^*)-d(p^*)|$: By definition, we have that

$$
\begin{aligned}
&|d^{est}(p^*)-d(p^*)| = \left|\bar{D}(p^M) + \left(\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)(p^*-p^M) - d(p^*)\right| \\
&= \left|\bar{D}(p^M) \pm d(p^M) + \left(\frac{\bar{D}_{n_i}(p_i^M)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)(p^*-p^M) - d(p^*) \pm \left(\frac{d(p^M)-d(p^L)}{p^M-p^L}(p^*-p^L)\right)\right| \\
&\le \left|\bar{D}(p^M) - d(p^M) + \left(\frac{\bar{D}_{n_i}(p_i^M)-d(p^M)+d(p^L)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)(p^*-p^M)\right| + \\
&\left| - d(p^*) + d(p^M) + \frac{d(p^M)-d(p^L)}{p^M-p^L}(p^*-p^M)\right|.
\end{aligned}
$$

Now let $p^* = \lambda p^L + (1-\lambda)p^M$, for some $\lambda \in [0,1]$. Then,

$$
\begin{aligned}
&\left|\bar{D}(p^M) - d(p^M) + \left(\frac{\bar{D}_{n_i}(p_i^M)-d(p^M)+d(p^L)-\bar{D}_{n_i}(p_i^L)}{p_i^M-p_i^L}\right)(p^*-p^M)\right| \le \\
&\qquad (1-\lambda)|\bar{D}(p^M)-d(p^M)| + \lambda|d(p^L)-\bar{D}_{n_i}(p^L)| \\
&\qquad \le |\bar{D}(p^M)-d(p^M)|.
\end{aligned}
$$

Hence, with probability at least $1 - \frac{1}{T^2}$,

$$\left|\bar{D}(p^M) - d(p^M) + \frac{\bar{D}(p^M) - d(p^M) + d(p^L) - \bar{D}(p^L)}{p^L - p^M}\left(p^* - p^M\right)\right| \leq |\bar{D}(p^M) - d(p^M)| \leq \frac{\Delta^2}{\rho^2}\,.$$

where the last inequality follows by a direct application of Hoeffding's inequality for sub-gaussian random variables.

Next, to bound $\left| - d(p^*) + d(p^M) + \frac{d(p^M)-d(p^L)}{p^M-p^L}(p^* - p^M)\right|$, we can directly apply the linear interpolation error bound (see Chapter 6 of Süli and Mayers (2003)) and get that

$$\left|d(p^L) + \frac{d(p^L) - d(p^M)}{p^L - p^M}(p^* - p^L) - d(p^*)\right| \leq \frac{K_1}{8}\Delta^2\,,$$

where recall that $K_1 = \max_{p\in[0,1], i\leq W} |d^i(p)|$. Hence,

$$|d^{est}(p^*) - d(p^*)| \leq \left(\frac{1}{\rho^2} + \frac{K_1}{8}\right)\Delta_i^2\,. \tag{C.13}$$

Now we focus on bounding term B of (C.12), and hence consider $|d'^{est}(p^*) - d'(p^*)|$.

Bounding $|d'^{est}(p^*) - d'(p^*)|$: First recall, by definition that $d'^{est}(p^*) = \left(\frac{\bar{D}(p^M)-\bar{D}(p^L)}{p^M-p^L}\right)$. Hence,

$$\begin{aligned}|d'^{est}(p^*) - d'(p^*)| &= \left|\frac{\bar{D}(p^M) - \bar{D}(p^L)}{p^M - p^L} + d'\left(\frac{p^M + p^L}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right| \\ &\leq \left|\frac{\bar{D}(p^M) - \bar{D}(p^L)}{p^M - p^L} - d'\left(\frac{p^M + p^L}{2}\right)\right| + \left|d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right|.\end{aligned} \tag{C.14}$$

But by Lemma C.4.1, we have that

$$\left|\frac{\bar{D}(p^M) - \bar{D}p^L)}{p^M - p^L} - d'\left(\frac{p^M + p^L}{2}\right)\right| \leq \Delta_i^2\left(\frac{K_1}{24} + \frac{2}{\rho^2}\right).$$

Similarly, to bound $\left|d'\left(\frac{p^M+p^L}{2}\right) - d'(p^*)\right|$, we use Assumption 4.2.2. This results in

$$\left|d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right| \leq \kappa\left(p^* - \frac{p^M + p^L}{2}\right)^2 \leq \frac{\kappa\Delta_i^2}{4}\,,$$

where the last inequality follows because $p^* \leq p^M$. Hence, combining the above two results and using (C.14), we have that

$$|d'^{est}(p^*) - d'(p^*)| \leq \left(\frac{K_1}{24} + \frac{2}{\rho^2} + \frac{\kappa}{4}\right)\Delta_i^2\,, \tag{C.15}$$

Hence using (C.13) and (C.15), we have that

$$\begin{aligned}|p^* - \tilde{p}^*| &\leq \frac{1}{2|d'^{est}(p^*)|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)p^*|\right)\\ &\leq \frac{1}{2|d'^{est}(p^*)|}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.\end{aligned}$$

Finally, to bound $d'^{est}(p^*)$, note that

$$\begin{aligned}&|d'^{est}(p^*)| \geq |d'(p^*)| - |d'^{est}(p^*) - d'(p^*)| \geq\\ &|d'(p^*)| - \left(\frac{K_1}{24} + \frac{\kappa}{4} + \frac{3}{\rho^2}\right)\Delta_i^2 \geq\\ &|d'(p^*)| - \left(\frac{K_1}{24} + \frac{\kappa}{4} + \frac{3}{\rho^2}\right)\frac{1}{4} \geq \frac{c}{2}\,,\end{aligned} \tag{C.16}$$

where the last inequality follows by our assumption on the derivative of demand at the optimal price bounded away from 0. Hence, we get that

$$|p^* - \tilde{p}^*| \leq \frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.$$

So far, we assumed that both p^* and $\tilde{p}^*$ are below the mid point of the current interpolation. Next, consider case (ii) when $p^* > p^M$ but as before $\tilde{p}^* \leq p^M$. In this case, we have to account for a larger approximation error in demand and gradient of demand at the optimal price.

Case (ii) $p^* > p^M$: Consider the approximate first order condition evaluated at p^*,

$$d^{est}(p^*) + d'^{est}(p^*)p^* = \bar{D}(p^M) + \frac{\bar{D}(p^H) - \bar{D}(p^M)}{p^H - p^M}\left(p^* - p^M\right) + \frac{\bar{D}(p^H) - \bar{D}(p^M)}{p^H - p^M}p^M - p^L p^*\,. \tag{C.17}$$

Similarly, evaluating the first order equation at the approximated optimal price, we get that

$$d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = \bar{D}(p^M) + \frac{\bar{D}(p^M) - \bar{D}(p^L)}{p^M - p^L}\left(\tilde{p}^* - p^M\right) + \frac{\bar{D}(p^M) - \bar{D}(p^L)}{p^M - p^L}\tilde{p}^* = 0\,, \tag{C.18}$$

where note that the difference in the evaluation function is due to $p^* > p^M$ but $\tilde{p}^* \leq p^M$. Subtracting (C.18) from (C.17), and letting $m_1 = \frac{\bar{D}(p^H)-\bar{D}(p^M)}{p^H-p^M}$ and $m_2 = \frac{\bar{D}(p^M)-\bar{D}(p^L)}{p^M-p^L}$, for ease of notation, we get that

$$\begin{aligned}
&d^{est}(p^*) + d'^{est}(p^*)p^* - \left(d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^*\right) = m_1(p^* - p^M) + m_1 p^* - m_2(\tilde{p^*} - p^M) - m_2 p^* \\
&= m_1(p^* - p^M + \tilde{p}^* - \tilde{p}^*) - m_2(\tilde{p}^* - p^M) + m_1(p^* - \tilde{p}^*) - m_2\tilde{p}^* \\
&= 2m_1(p^* - \tilde{p}^*) + m_1(\tilde{p}^* - p^M) - m_2(\tilde{p}^* - p^M) + (m_1 - m_2)\tilde{p}^* \\
&= 2m_1(p^* - \tilde{p}^*) + (m_1 - m_2)(2\tilde{p}^* - p^M)\,.
\end{aligned}$$

We follow the same analysis as before and arrive at the following:

$$|p^* - \tilde{p}^*| \leq \underbrace{\frac{1}{2|m_1|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^*\right)}_{\text{A}} + \underbrace{|m_1 - m_2|\left(2\tilde{p}^* - p^M\right)}_{\text{B}}\,. \quad \text{(C.19)}$$

Notice that (A) in the equation above is the same as before (case (i) when $p^* \leq p^M$). Hence, an identical analysis yields that

$$\frac{1}{2|m_1|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^*\right) \leq \frac{1}{c}\left(\frac{1}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.$$

Focusing on (B), we get that

$$\begin{aligned}
|m_1 - m_2| &= \left|m_1 - m_2 + d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^H}{2}\right)\right| \\
&= \left|m_1 - m_2 + d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^H}{2}\right) + d'\left(\frac{p^M + p^L}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right)\right| \\
&\leq \left|m_1 - d'\left(\frac{p^M + p^H}{2}\right)\right| + \left|d'\left(\frac{p^M + p^L}{2}\right) - m_2\right| + \left|d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right)\right| \\
&\leq \left(\frac{K_1}{12} + \frac{4}{\rho^2} + 4\kappa\right)\Delta_i^2\,,
\end{aligned}$$

with probability at least $1 - 1/T^2$, where the last inequality follows by Lemma C.4.1 and Assumption 4.2.2. Hence, we have that

$$|p^* - \tilde{p}^*| \leq \left(\frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + \frac{4}{\rho^2} + 4\kappa\right)\Delta_i^2\,,$$

hence, letting $M = \left(\frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + 4\kappa\right)$ proves the final result.

We have so far considered all rounds such that $\Delta_i \geq \sigma$. Next, we consider the case when $\Delta_i \leq \sigma$. Since the proof follows very similarly as above, we will skip some details for the sake of brevity.

All rounds such that $\Delta_i \leq \sigma$:

As before we analyze two cases: (i) $p^* < p^M$ and (ii)$p^* \geq p^M$.

Case (i) $p^* < p^M$: Consider the approximate first order condition evaluated at p^*, and following the same analysis as before, we have that

$$\begin{aligned} 2\left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)(p^* - \tilde{p}^*) &= (d^{est}(p^*) - d(p^*)) + (d'^{est}(p^*) - d'(p^*))p^* \\ 2\left|\left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)\right||p^* - \tilde{p}^*| &\leq |d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^* \qquad \text{(C.20)} \\ \implies |p^* - \tilde{p}^*| \leq \frac{1}{2|d'^{est}(p^*)|}&\left(\underbrace{|d^{est}(p^*) - d(p^*)|}_{\text{A}} + \underbrace{|d'^{est}(p^*) - d'(p^*)|p^*}_{\text{B}}\right), \end{aligned}$$

where $d'^{est}(p^*) := \left(\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\right)$. Hence, to bound the estimation error in the optimal price, we need to bound terms (A) and (B).

Step 2: Bounding estimation error in demand and gradient:

We proceed by independently bounding (A) and (B) from (C.20).

Bounding $|d^{est}(p^*) - d(p^*)|$: By definition, we have that

$$\begin{aligned} |d^{est}(p^*) - d(p^*)| &= \left|\bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}(p^* - p^M) - d(p^*)\right| \\ &= \left|\bar{D}(p^M) \pm d(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}(p^* - p^M) - d(p^*) \pm \left(\frac{d(p^M) - d(p^L)}{p^M - p^L}(p^* - p^L)\right)\right| \\ &\leq \left|\bar{D}(p^M) - d(p^M) + \frac{D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)}{p^L - p^M}(p^* - p^M)\right| + \\ &\left| - d(p^*) + d(p^M) + \frac{d(p^M) - d(p^L)}{p^M - p^L}(p^* - p^M)\right|. \end{aligned}$$

Now let $p^* = \lambda p^L + (1-\lambda)p^M$, for some $\lambda \in [0,1]$. Then,

$$\begin{aligned} &\left|\bar{D}(p^M) - d(p^M) + \frac{D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)}{p^L - p^M}(p^* - p^M)\right| \leq \\ &\qquad |\bar{D}(p^M) - d(p^M)| + \lambda|(D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L))|. \end{aligned}$$

But by Assumption 4.4.1, we have that

$$(D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)) = d(p^M) + \epsilon_m^* - d(p^M) + d(p^L) - d(p^L) - \epsilon_l^* \leq f(n)\,.$$

Hence,

$$\begin{aligned}\left|\bar{D}(p^M) - d(p^M) + \frac{D_{M_1^*}(p^M) - d(p^M) + d(p^L) - D_{L^*}(p^L)}{p^L - p^M}\left(p^* - p^M\right)\right| &\leq |\bar{D}(p^M) - d(p^M)| + \lambda f(n) \\ &\leq \frac{\Delta^2}{\rho^2} + \frac{\Delta^3}{\rho^3}\,,\end{aligned}$$

where the last inequality follows by a direct application of Hoeffding's inequality for sub-gaussian random variables and by the assumption that $\delta > 3/4$.

Next, to bound $\left| - d(p^*) + d(p^M) + \frac{d(p^M)-d(p^L)}{p^M-p^L}(p^* - p^M)\right|$, we use the same argument as before and get that

$$\left|d(p^L) + \frac{d(p^L) - d(p^M)}{p^L - p^M}(p^* - p^L) - d(p^*)\right| \leq \frac{K_1}{8}\Delta^2\,,$$

where recall that $K_1 = \max_{p\in[0,1],i\leq w} |d^i(p)|$. Hence,

$$|d^{est}(p^*) - d(p^*)| \leq \left(\frac{1}{\rho^2} + \frac{\Delta}{\rho} + \frac{K_1}{8}\right)\Delta_i^2\,. \tag{C.21}$$

Now we focus on bounding term B of (C.20), and hence consider $|d'^{est}(p^*) - d'(p^*)|$.

Bounding $|d'^{est}(p^*) - d'(p^*)|$: First recall, by definition that $d'^{est}(p^*) = \left(\frac{D_{M_1^*}(p^M)-D_{L^*}(p^L)}{p^M-p^L}\right)$. Hence,

$$\begin{aligned}|d'^{est}(p^*) - d'(p^*)| &= \left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} + d'\left(\frac{p^M + p^L}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right| \\ &\leq \left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} - d'\left(\frac{p^M + p^L}{2}\right)\right| + \left|d'\left(\frac{p^M + p^L}{2}\right) - d'(p^*)\right|.\end{aligned} \tag{C.22}$$

But by Lemma C.4.1, we have that

$$\left|\frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L} - d'\left(\frac{p^M + p^L}{2}\right)\right| \leq \Delta^2\left(\frac{K_1}{24} + \frac{1}{\rho^2}\right)\,.$$

Similarly, to bound $\left|d'\left(\frac{p^M+p^L}{2}\right) - d'(p^*)\right|$, we use Assumption 4.2.2. This results in

$$\left|d'\left(\frac{p^M+p^L}{2}\right) - d'(p^*)\right| \leq \kappa\left(p^* - \frac{p^M+p^L}{2}\right)^2 \leq \frac{\kappa\Delta_i^2}{4},$$

where the last inequality follows because $p^* \leq p^M$. Hence, combining the above two results and using (C.22), we have that

$$|d'^{est}(p^*) - d'(p^*)| \leq \left(\frac{K_1}{24} + \frac{1}{\rho^2} + \frac{\kappa}{4}\right)\Delta_i^2, \tag{C.23}$$

Hence using (C.21) and (C.23), we have that

$$\begin{aligned}|p^* - \tilde{p}^*| &\leq \frac{1}{2|d'^{est}(p^*)|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)p^*|\right)\\ &\leq \frac{1}{2|d'^{est}(p^*)|}\left(\frac{2}{\rho^2} + \frac{\Delta}{\rho} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2.\end{aligned}$$

Finally, to bound $d'^{est}(p^*)$, note that previously, we assumed that the derivative of demand at the optimal price is bounded away from 0

$$|d'(p^*)| - \left(\frac{K_1}{24} + \frac{\kappa}{4} + \frac{3}{\rho^2}\right)\frac{1}{4} \geq \frac{c}{2}.$$

Hence, since $\Delta \leq \frac{1}{\rho}$

$$|p^* - \tilde{p}^*| \leq \frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2.$$

Notice that so far we assumed that both p^* and $\tilde{p}^*$ are below the mid point of the current interpolation. Next, consider case (ii) when $p^* > p^M$ but as before $\tilde{p}^* \leq p^M$. In this case, we have to account for a larger approximation error in demand and gradient of demand at the optimal price.

Case (ii) $p^* > p^M$: Consider the approximate first order condition evaluated at p^*,

$$d^{est}(p^*) + d'^{est}(p^*)p^* = \bar{D}(p^M) + \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}\left(p^* - p^M\right) \tag{C.24}$$

$$+ \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}p^M - p^L p^*. \tag{C.25}$$

Similarly, evaluating the first order equation at the approximated optimal price, we get that

$$d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^* = \bar{D}(p^M) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\left(\tilde{p}^* - p^M\right) + \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}\tilde{p}^* = 0\,, \tag{C.26}$$

where note that the difference in the evaluation function is due to $p^* > p^M$ but $\tilde{p}^* \leq p^M$. Subtracting (C.26) from (C.24), and letting $m_1 = \frac{D_{H^*}(p^H) - D_{M_2^*}(p^M)}{p^H - p^M}$ and $m_2 = \frac{D_{M_1^*}(p^M) - D_{L^*}(p^L)}{p^M - p^L}$, for ease of notation, we get that

$$d^{est}(p^*) + d'^{est}(p^*)p^* - \left(d^{est}(\tilde{p}^*) + d'^{est}(\tilde{p}^*)\tilde{p}^*\right) = 2m_1(p^* - \tilde{p}^*) + (m_1 - m_2)(2\tilde{p}^* - p^M)\,.$$

We follow the same analysis as before and arrive at the following:

$$|p^* - \tilde{p}^*| \leq \underbrace{\frac{1}{2|m_1|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^*\right)}_{\text{A}} + \underbrace{|m_1 - m_2|\left(2\tilde{p}^* - p^M\right)}_{\text{B}}\,. \tag{C.27}$$

Notice that (A) in the equation above is the same as before (case (i) when $p^* \leq p^M$). Hence, an identical analysis yields that

$$\frac{1}{2|m_1|}\left(|d^{est}(p^*) - d(p^*)| + |d'^{est}(p^*) - d'(p^*)|p^*\right) \leq \frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right)\Delta_i^2\,.$$

Focusing on (B), we get that

$$\begin{aligned}
|m_1 - m_2| &= \left|m_1 - m_2 + d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^H}{2}\right)\right| \\
&= \left|m_1 - m_2 + d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^H}{2}\right) + d'\left(\frac{p^M + p^L}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right)\right| \\
&\leq \left|m_1 - d'\left(\frac{p^M + p^H}{2}\right)\right| + \left|d'\left(\frac{p^M + p^L}{2}\right) - m_2\right| + \left|d'\left(\frac{p^M + p^H}{2}\right) - d'\left(\frac{p^M + p^L}{2}\right)\right| \\
&\leq \left(\frac{K_1}{12} + \frac{2}{\rho^2} + 4\kappa\right)\Delta_i^2\,,
\end{aligned}$$

where the last inequality follows by Lemma C.4.1 and Assumption 4.2.2. Hence, we have that

$$|p^* - \tilde{p}^*| \leq \left(\frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{K_1}{12} + \frac{2}{\rho^2} + 4\kappa\right)\Delta_i^2\,,$$

hence, letting $M = \left(\frac{1}{c}\left(\frac{3}{\rho^2} + \frac{K_1}{6} + \frac{\kappa}{4}\right) + \frac{2}{\rho^2} + \frac{K_1}{12} + 4\kappa\right)$ proves the final result for all rounds

such that $\Delta_i \leq \sigma$. □

Proof. Proof of Theorem 4.4.5: The proof of Theorem 4.4.5 follows identically as the proof of Theorem 4.4.1, where we use Lemma C.4.1 and Lemma C.4.2 instead of Lemma 4.4.2 and Lemma 4.4.3. We skip the details for the sake of brevity. □

Appendix D

Appendix of Chapter 5

D.1 Proofs of technical results

Proof. Proof of Theorem 5.6.1: By definition, we have that

$$\begin{aligned}\mathbb{E}\left[\mathcal{C}(y)\right] &= \mathbb{E}\left[c_{DIC}\mathrm{I}(y) + c_{RTO}\mathrm{R}\left(\tilde{Z}(y)\right)\right] \\ &= c_{DIC}\mathbb{E}\left[\mathrm{I}(y)\right] + c_{RTO}\mathbb{E}\left[\mathrm{R}\left(\tilde{Z}(y)\right)\right] .\end{aligned}$$

Note that $\mathbb{E}[\mathrm{I}(y)] = 1 - F(y)$. This follows directly from the definition of I(y). Similarly,

$$\mathbb{E}\left[\mathrm{R}\left(\tilde{Z}(y)\right)\right] = \mathbb{E}\left[\mathrm{r}(\tilde{Z}(y))\right] = \mathrm{r}\left(\mathbb{E}\left[\tilde{Z}(y)\right]\right) ,$$

where the first equality follows by definition of R, and the second equality follows using Jensen's inequality and the assumption that r is a simple affine function (Kuczma 2009). Finally,

$$\begin{aligned}\mathbb{E}[\tilde{Z}(y)] &= \mathbb{E}[\tilde{Z}(y)|\mathrm{I}(y) = 1]\mathbb{P}(\mathrm{I}(y) = 1) + \mathbb{E}[\tilde{Z}(y)|\mathrm{I}(y) = 0]\mathbb{P}(\mathrm{I}(y) = 0) \\ &= y(1 - \mathrm{F}(y)) + \mu\mathrm{F}(y) = y + \mathrm{F}(y)\left(\mu - y\right) .\end{aligned}$$

Hence,

$$\mathbb{E}\left[\mathrm{R}\left(\tilde{Z}(y)\right)\right] = \mathrm{r}\left(y + \mathrm{F}(y)\left(\mu - y\right)\right) . \tag{D.1}$$

$$\begin{aligned}\mathbb{E}\left[\mathcal{C}(y)\right] &= c_{DIC}\left(1-F(y)\right) + c_{RTO}\left(\mathrm{r}\left(y+F(y)\left(\mathbb{E}[Z|Z<y]-y\right)\right)\right)\\ &= \bar{C}_{DIC}e^{\frac{-y}{\mu}} + c_{RTO}\beta\left(y+\left(1-e^{\frac{-y}{\mu}}\right)\left(\mathbb{E}[Z|Z<y]-y\right)\right)\\ &= \bar{C}_{DIC}e^{\frac{-y}{\mu}} + \mu c_{RTO}\beta e^{-\frac{2y}{\mu}}\left(1-2e^{\frac{y}{\mu}}+e^{\frac{2y}{\mu}}+\frac{y}{\mu}\right),\end{aligned}$$

where the first inequality follows because $F(y)$ for an exponential random variable is $1-e^{\frac{-y}{\mu}}$. The second inequality follows because $\mathbb{E}[Z|Z<y] = \mu\left(1-e^{-\frac{y}{\mu}}\left(1+\frac{y}{\mu}\right)\right)$.

Next, to analyze the cost function, we consider the second derivative of the objective function:

$$\begin{aligned}\frac{\partial\mathbb{E}\left[\mathcal{C}(y)\right]}{\partial y} &= -\frac{\bar{C}_{DIC}e^{\frac{-y}{\mu}}}{\mu} + \beta c_{RTO}e^{-\frac{2y}{\mu}}\left(-1+2e^{\frac{y}{\mu}}-2\frac{y}{\mu}\right)\\ \frac{\partial^2\mathbb{E}\left[\mathcal{C}(y)\right]}{\partial y^2} &= \frac{\bar{C}_{DIC}e^{\frac{-y}{\mu}}}{\mu^2} - 2\frac{c_{RTO}e^{\frac{-2y}{\mu}}\beta\left(e^{\frac{y}{\mu}}-\frac{2y}{\mu}\right)}{\mu}.\end{aligned}$$

Then, for any y, the objective cost function is strictly convex if $\frac{\partial^2\mathbb{E}[\mathcal{C}(y)]}{\partial y^2} > 0$. Hence,

$$\begin{aligned}&> 0 \implies \frac{e^{\frac{-2y}{\mu}}}{\mu}\left(\frac{\bar{C}_{DIC}e^{\frac{y}{\mu}}}{\mu} - 2c_{RTO}\beta\left(e^{\frac{y}{\mu}}-\frac{2y}{\mu}\right)\right) > 0\\ \implies &\frac{\bar{C}_{DIC}e^{\frac{y}{\mu}}}{\mu} + 4c_{RTO}\beta\frac{y}{\mu} > 2c_{RTO}\beta\left(e^{\frac{y}{\mu}}\right) \implies \frac{\bar{C}_{DIC}}{2c_{RTO}\beta\mu} + 2\frac{\frac{y}{\mu}}{e^{\frac{y}{\mu}}} > 1,\\ \implies &\frac{\frac{y}{\mu}}{e^{\frac{y}{\mu}}} > \frac{1}{2}\left(1-\frac{\bar{C}_{DIC}}{2c_{RTO}\beta\mu}\right) \implies \frac{z}{e^z} > \left(\frac{2c_{RTO}\beta\mu-\bar{C}_{DIC}}{4c_{RTO}\beta\mu}\right)\end{aligned} \tag{D.2}$$

where $z = \mu y \in [0,1]$. Now note that if $2c_{RTO}\beta\mu < \bar{C}_{DIC}$, the RHS of the above equation is negative. Furthermore, the minimum of the LHS in the above equation is 0. Therefore, the equation above holds for any value of z and the function is strictly convex. Furthermore, the optimal solution of ODTP is given by the first order condition (FOC). This proves that the objective is strictly convex when $2c_{RTO}\beta\mu < \bar{C}_{DIC}$.

To characterize how the optimal solution changes with different problem parameters ($\beta, c_{RTO}, \bar{C}_{DIC}$

and μ), we use the Implicit Function Theorem. Hence,

$$\frac{\partial y^*}{\partial c_{RTO}} = -\beta \left(\frac{2e^{\frac{y}{\mu}} - 2\frac{y}{\mu} - 1}{\frac{\bar{C}_{DIC} e^{\frac{-y}{\mu}}}{\mu^2} - 2\frac{c_{RTO} e^{\frac{-2y}{\mu}} \beta \left(e^{\frac{y}{\mu}} - \frac{2y}{\mu} \right)}{\mu}} \right) < 0,$$

$$\frac{\partial y^*}{\partial \beta} = -c_{RTO} \left(\frac{2e^{\frac{y}{\mu}} - 2\frac{y}{\mu} - 1}{\frac{\bar{C}_{DIC} e^{\frac{-y}{\mu}}}{\mu^2} - 2\frac{c_{RTO} e^{\frac{-2y}{\mu}} \beta \left(e^{\frac{y}{\mu}} - \frac{2y}{\mu} \right)}{\mu}} \right) < 0$$

$$\frac{\partial y^*}{\partial \mu} = -\left(\frac{-2c_{RTO}\beta y \left(e^{\frac{y}{\mu}} - 2\frac{y}{\mu} \right) + \bar{C}_{DIC} e^{\frac{y}{\mu}} \left(\frac{y}{\mu} - 1 \right)}{\frac{\bar{C}_{DIC} e^{\frac{-y}{\mu}}}{\mu^2} - 2\frac{c_{RTO} e^{\frac{-2y}{\mu}} \beta \left(e^{\frac{y}{\mu}} - \frac{2y}{\mu} \right)}{\mu}} \right) > 0,$$

$$\frac{\partial y^*}{\partial \bar{C}_{DIC}} = \left(\frac{e^{\frac{y}{\mu}}}{\frac{\bar{C}_{DIC} e^{\frac{-y}{\mu}}}{\mu} - 2c_{RTO} e^{\frac{-2y}{\mu}} \beta \left(e^{\frac{y}{\mu}} - \frac{2y}{\mu} \right)} \right) > 0\,,$$

where the above inequalities follow from the fact that the denominator in each case is positive (because of the convexity of the curve) and the sign of the numerator drives the sign of the overall expression.

Next, consider $2c_{RTO}\beta\mu \geq \bar{C}_{DIC}$ and note that $\frac{z}{e^z} \leq \frac{1}{e}$. Then, if

$$\begin{aligned} \frac{1}{e} &\leq \left(\frac{2c_{RTO}\beta\mu - \bar{C}_{DIC}}{4c_{RTO}\beta\mu} \right) \\ &\implies 4c_{RTO}\beta\mu \leq e\left(2c_{RTO}\beta\mu - \bar{C}_{DIC}\right) \\ &\implies \frac{\bar{C}_{DIC}}{c_{RTO}} \leq 2\beta\mu \left(1 - \frac{2}{e} \right), \end{aligned}$$

then the objective cost function is concave. Furthermore, because ODTP has a minimum cost objective, the optimal solution lies on the boundary points. Hence, checking the value of the objective function at the boundary points, we find that the objective is $\bar{C}_{DIC}$ when the threshold is 0 and $\frac{\bar{C}_{DIC}}{e} + \frac{c_{RTO}\mu\beta(2-2e+e^2)}{e^2}$ when the threshold is μ. It is easy to check that the objective cost at 0 always dominates the other cost, and hence the optimal solution is to always choose the minimum threshold possible at the $\bar{C}_{DIC}$ cost, which is 0. This proves the second part of the Theorem.

Finally, if $2\beta\mu \geq \frac{\bar{C}_{DIC}}{c_{RTO}} \geq \left(1 - \frac{2}{e}\right) 2\beta\mu$, then the objective is neither concave nor convex.

Furthermore, the second derivative of the objective function is 0 when,

$$\frac{z}{e^z} = \left(\frac{2c_{RTO}\beta\mu - \bar{C}_{DIC}}{4c_{RTO}\beta\mu}\right),$$

where $z = \frac{y}{\mu}$. Let z^* be the solution to the above equation. The objective cost function is concave $\forall x \in [0, z^*\mu]$ and convex $\forall x \in [z^*\mu, \mu]$. Hence, the optimal solution lies either at the boundary points of the concave region (0 or $z^*\mu$) or is the interior point solution in the part of the region where the objective is convex (i.e., when it lies between $[z^*\mu, \mu]$). Checking and comparing the objective value at all these end points gives the optimal solution of the ODTP. □

Proof. Proof of Proposition 5.6.2: In order to prove this proposition, we need to show that the budget constraint is also satisfied with the IP formulation. We introduce auxiliary variables τ_i^j and λ_i^j in order to enforce appropriate DIC cost accounting. Consider any order i and let w_i be such that $d_i^j \leq w_i \leq d_i^{j+1}$. Then w_i can be represented as a convex combination of d_i^j and d_i^{j+1}. The convex weights are denoted by continuous variables λ_i^j. While (9g) ensures that the weights sum up to 1, binary auxiliary variables τ_i^j ensure that only the corresponding λ_i^j variables are chosen to represent w_i. Furthermore, because DIC cost function is continuous and piecewise linear, the DIC cost associated with decision w_i is $\sum_{i=1,..k+1} \lambda_i^j d_i^j$. This is formulated in the budget constraint 9(b). Hence, we have that the ODEP problem with piecewise linear costs can be formulated as a mixed integer optimization problem.

Proof. Proof of Theorem 5.6.3: We prove the Theorem in two steps.

1. Showing feasibility of the constructed solution.
2. Showing that the heuristic solution is very close to the optimal LP solution in terms of the objective value.

This in turn results in a optimality gap bound from the optimal IP solution.

Feasibility of the heuristic solution: We split the orders in two disjoint sets: O_1 contains all orders for which the optimal expediting decision $\bar{w}$, is not at one of the end points of the DICs function (d_j), and O_2 contains the rest of the orders not in O_1. Note that for all orders in O_2, the corresponding $\bar{z}$ variables are already integral. Hence, the optimal expediting LP variable is already feasible for the IP formulation. Next, note that for orders in O_2, the optimal solution is d_j and not $d_j + \epsilon$. Indeed, as noted before, increasing $\bar{w}_i$ from $d_j + \epsilon$ to d_{j+1} leads

to no change in the cost function but an improvement in the objective value. Having shown that w_i are all on the right end of our constructed continuous DIC function, we next show the integrality of the corresponding τ_i^j variables. WLOG assume that for order $i \in O_2$, $w_i = d_j^*$.

Now notice that the corresponding constraints associated with τ and λ of order i are

$$\lambda^1 \leq \tau^i, \lambda^j \leq \tau^j + \tau^{j-1}, \forall j = 2, ...k \text{ and } \lambda^{k+1} \leq \tau^k.$$

Note that the integrality constraint on z ensures that any w is a convex combination of at most two points, the end points of the region where w lies. Without the integrality constraint, w can now be a convex combination of any number of points. Note that the convex combination that is chosen to represent w does not affect the objective. Nevertheless, it indeed affects the budget constraint. An optimal convex combination would be one that chooses the minimum cost associated with the improvement w.

Next, we will show that the discontinuity parameter ϵ can be tuned such that representing d_j^* with a single nonzero λ would lead to a minimum cost. Indeed, if λ is integral, then so is τ which will prove the integrality of the solution. To prove this, let us assume by contradiction that d_j^* is represented as $\sum_{j=1,..,k+1,j\neq j^*} \lambda_j d_j$. If j^* is part of the convex combination, then the solution is already integral, and we are done. If it is not, then the convex combination contains two sets:

$$c^+ = \{d_j : d_j > d_j^*\} \quad \text{and} \quad c^- = \{d_j : d_j < d_j^*\}.$$

Note that the cost associated with all entries in c^+ is more than the cost of c_{j^*} ,and vice versa. Next, consider the following optimization problem:

$$\begin{aligned} Min_{\lambda_j, j=1,..k+1} \quad & \sum_{j=1,..k+1, k\neq j^*} \lambda_j \bar{C}_j \\ \text{s.t.} \quad & \sum_{j=1,..k+1, k\neq j^*} \lambda_j = 1, \\ & \sum_{j=1,..k+1, k\neq j^*} \lambda_j d_j = d_j^* \text{ and } \lambda_j \geq 0. \end{aligned}$$

The objective of the constructed optimization problem is to find a convex representation of d_j^* with minimum DIC. If we can show that the optimal solution of the constructed LP is $\bar{C}_j^*$, the LP relaxation already satisfies the integrality constraints. This happens because the continuous problem will correctly account for the associated costs, and will have integral z variables.

We start by considering the optimal solution of the constructed LP and note by LP duality that the optimal solution has at least $(k-1)$ λ variables equal to 0. If the convex representation has only one positive λ, we are done. If not, and one of the two end points contain ϵ, it can be tuned such that the objective value of the solution would always be greater than $\bar{C}_{j^*}$. Next, assume that the representation contains only end points without an ϵ, and let the representation be

$$d_j^* = \lambda^+ d^+ + (1-\lambda^+)d^- \,,$$

where $d^+ \in c^+$ and $d^- \in c^-$. Simple algebra yields that $\lambda^+ = \frac{d_j^* - d^-}{d_j^+ - d^-}$. Furthermore, the corresponding objective value is $\bar{C}_- + \lambda^+(\bar{C}_+ - \bar{C}_-)$. Next, we show that the objective is at least $\bar{C}_{j^*}$. Consider

$$\bar{C}_- + \lambda^+(\bar{C}_+ - \bar{C}_-) \geq \bar{C}_{j^*} \implies \lambda^+(\bar{C}_+ - \bar{C}_-) \geq \bar{C}_{j^*} - \bar{C}_- \implies \frac{\bar{C}_+ - \bar{C}_-}{\bar{C}_{j^*} - \bar{C}_-} \geq \frac{1}{\lambda^+} \,.$$

But by assumption, the costs change at a rate faster than a linear cost change (linear slope of 1). Hence the inequality holds and the objective value is always less than $\bar{C}_{j^*}$. Because the proof holds for any arbitrary j^*, we are done with all orders in O_2.

Next, consider all orders in O_1. By definition, the LP optimal solution for these orders is not one of d_j. We first show that O_1 contains at most one order. Let us prove this by contradiction and let O_1 contain two orders. Then, because it is a feasible solution, the sum of the costs for both of these orders satisfies the budget constraint. Let the corresponding RTO slopes (objective coefficients) be β_1 and β_2, respectively, and WLOG assume $\beta_1 \leq \beta_2$. Then, using more budget on order 2 yields greater improvement in the objective. Hence, the optimal solution should use more budget for order 2, pushing order 2's optimal solution to either the next DIC range (if the budget constraint is not tight) or pushing it to the right end point of the current cost level. In either case, it is pushed to O_2. Hence, we are left with a single order in O_1. Let this order be NI and note that the updated heuristic solution constructed for order NI is IP feasible by construction.

Optimality gap of the constructed heuristic: Having constructed a feasible IP solution, we finally discuss the optimality gap of the constructed solution. First, note that

$$Obj(IP_{opt}) \leq Obj(LP_{opt}) \implies Obj(IP_{opt}) - Obj(LP_{heuristic}) \leq Obj(LP_{opt}) - Obj(LP_{heuristic}) \,.$$

Hence, consider $Obj(LP_{opt}) - Obj(LP_{heuristic})$, and note that the heuristic solution is different from the LP optimal solution only for a single order. Furthermore, because the optimal improvement in RTO rate from a single order is bounded above by 1, the optimality gap follows.

D.2 Results from the Econometric Analysis of §5.3.3

	FDG	
warehouse time	0.001***	(0.000)
APD^+	1.102***	(0.038)
APD^-	0.206***	(0.011)
price	-0.000***	(0.000)
product discount	1.12***	(0.036)
N	1,662,175	

Standard errors in parentheses

* $p < 0.05$, ** $p < 0.01$, *** $p < 0.001$

Table D.1: First-stage regression results for IV analysis with various controls. The Adj. R^2 is 0.77. Furthermore, the partial R^2 of the *warehouse time* instrument is 0.21. Note that we also control for brand, article type, supply type, courier, partner, month, and DC level fixed effects in our specification.

Variable	**IV-1 day**	**IV-2 days**	**IV-3 days**
FDG	0.009*** (0.002)	0.014*** (0.000)	0.014*** (0.000)
APD^+	0.001 (0.002)	−0.002 (0.000)	−0.001 (0.000)
APD^-	0.000 (0.000)	−0.001** (0.000)	−0.001*** (0.000)
price	+0.000*** (0.000)	+0.000*** (0.000)	0.000*** (0.000)
product discount	-0.090*** (0.008)	−0.103*** (0.003)	−0.101*** (0.003)
Observations	196,765	1,489,067	1,653,729

*$p < 0.10$, **$p < 0.05$, ***$p < 0.01$

Table D.2: Estimation results from the second stage of the IV analysis with different levels of subsetting of transaction data. In column (1), we present the results when the data set includes transactions from zip codes that are less than 1 day away from each of the warehouses. In column (2), the subset includes transactions from zip codes that are less than 2 days away from all the warehouses. Finally, in column (3), we includes all zip codes that are less than 3 days away from all the warehouses.

Variable	**Point Estimate**	**Standard Error**
APD^-	0.005*	0.002
price	0.000***	0.000
product discount	-0.000	0.000
average discount	-0.000	0.000
total orders	0.000**	0.000
average price	0.000	0.000
Observations	9,138	

*$p < 0.10$, **$p < 0.05$, ***$p < 0.01$

Table D.3: Regression results from the RCT with data from repeat customers. The coefficient of APD^- is significant and positive, showing that an increase in delivery gap results in an increase in product RTO.

Variable	Point Estimate	Standard Error
FDG	0.010***	0.004
APD^{+}	−0.002	0.008
APD^{-}	0.005***	0.002
price	+0.000***	0.000
product discount	-0.000	0.000
Observations	19,983	

$^{*}p < 0.10$, $^{**}p < 0.05$, $^{***}p < 0.01$

Table D.4: Regression results from the RCT with data from all customers with variable delivery gaps. The coefficient of the FDG is significant and positive, showing that an increase in delivery gap results in an increase in product RTO.

D.3 Analysis of the Usage of the COD Payment Method

We perform two different analyses to understand the heterogeneity in the COD usage. (i) We perform a district-level analysis, where we empirically check for correlation between the COD usage, and socioeconomic factors such as percentage of literate population, percentage of urban population and percentage of population working in agriculture and related activities. Using data from 255 districts from 19 states for which socioeconomic data was present, we find that the COD usage is

- *Negatively correlated* with the percentage of literate population. Districts with literacy rate of less than 56.06% (bottom 25% quantile) used the COD payment method for 70.22% orders, as compared to 52.4% orders for districts with literacy rate higher than 68.74% (top 25% quantile).

- *Negatively correlated* with the percentage of urban population. Districts with urban population of less than 16.25% (bottom 25% quantile) used the COD payment method for 71.09% orders, as compared to 53.79% orders for districts with urban population higher than 33.96% (top 25% quantile).

- *Positively correlated* with the percentage of population related with agriculture and allied activities. Districts where less than 6.06% (bottom 25% quantile) population was involved in agriculture used the COD payment method for 52.04% orders, as compared to 71.30% orders for districts where more than 16.1% (top 25% quantile) population was involved in agriculture.

We also perform a city-level analysis of the COD usage. Myntra ranks each city between one to three, based on infrastructure development and other socioeconomic factors. A Tier 1 city is

economically more advanced than Tier 2 city and so on. We find that COD usage in Tier 1 cities is lower (49.5% of all orders), compared to Tier 2 (65.24% of all orders) and Tier 3 (59.96% of all orders) cities. Nevertheless, COD payment method is very popular among Myntra customers across cities. Since we have city tier information on all zipcodes, this analysis is performed on all orders and hence has no sample bias.

The above analysis shows that the COD payment method is used heterogeneously among different districts and is correlated with socioeconomic indicators of an area. A simple RTO reduction strategy could be to focus on delivery improvement efforts only in districts with high COD usage. This is similar to the order-level tactical problem that optimizes budget utilization among different orders with base level RTO rates higher for regions where there is high COD utilization.

D.4 Supplementary Figures

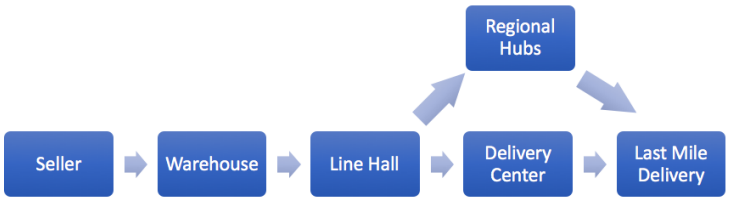

Figure D.1: Supply chain structure at the fashion e-retailer.

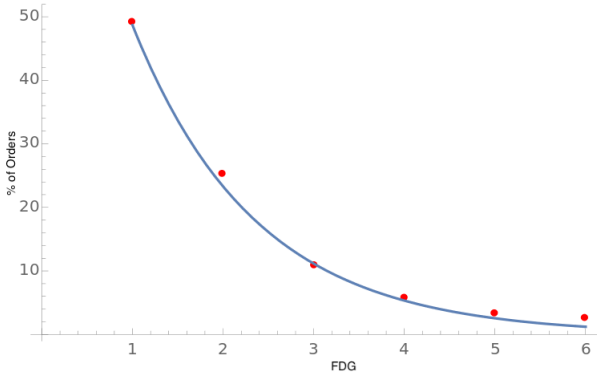

Figure D.2: Empirical f distribution fitted with a mixture of shifted exponentials. The exponential distribution shows a very good fit for values greater than 1.

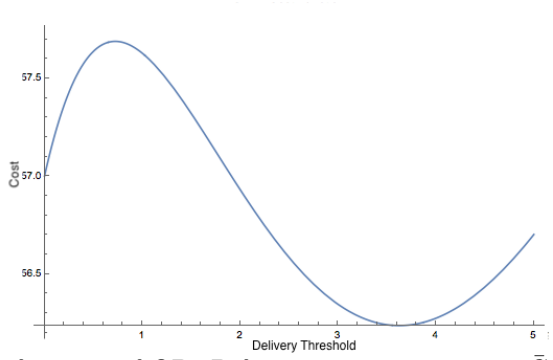

Figure D.3: The objective function of ODTP for $\mu = 4$, $c_{RTO} = 165$, $\bar{C}_{DIC} = 57$, and $\beta = 0.10$. The objective is neither concave nor convex and not even unimodal. Nevertheless, it is concave until a critical value (μz^*) and becomes convex afterwards.

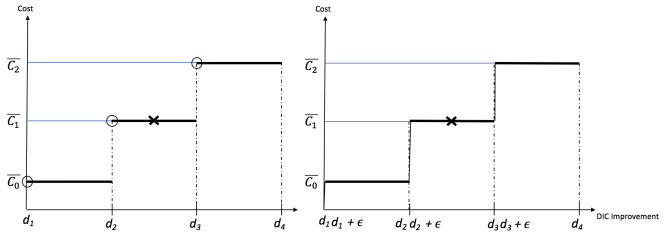

Figure D.4: Piecewise constant DIC function. To formulate ODEP as an IP, we first convert the cost function into a piecewise-linear cost function by connecting the discontinuous pieces.

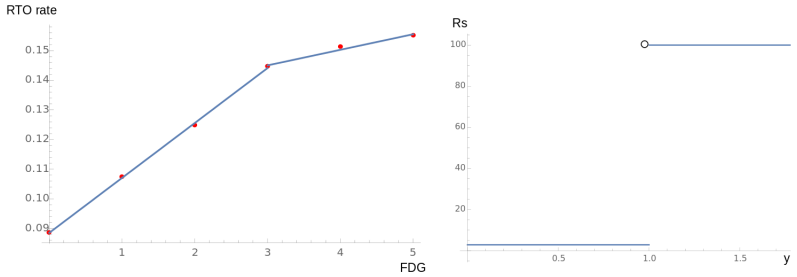

Figure D.5: On the left, RTO function for Bengaluru around the mean. On the right, the cost in rupees (Rs) of expediting FDG by y number of days.

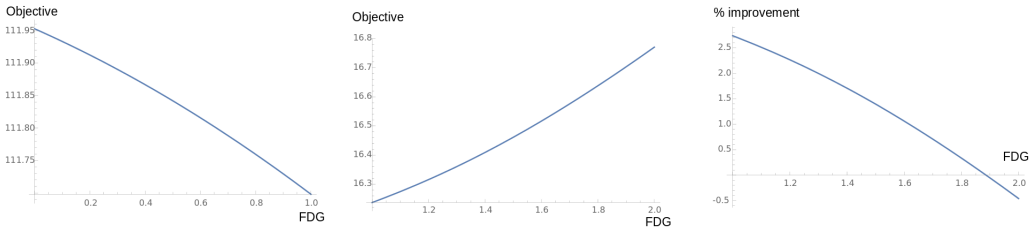

Figure D.6: ODTP objective cost function for FDG <1 and $FDG \geq 1$ on the left, and the middle plots. On the right, we plot the RTO cost improvement for various threshold levels at $\bar{C}_{DIC} = 1.9$ Rs

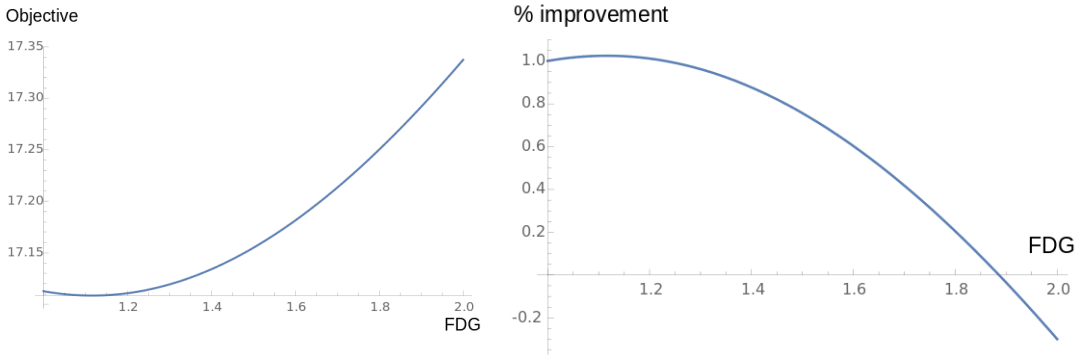

Figure D.7: The objective function and the % improvement for C_{DIC}=Rs. 2.85.

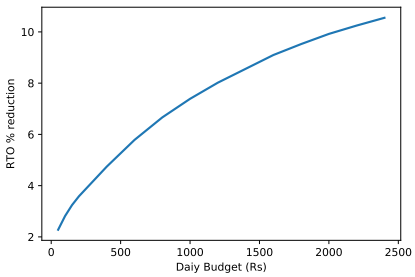

Figure D.8: RTO rate improvement due to optimal budget allocation (ODEP) as a function of the daily budget (B) in rupees.

Bibliography

Abbasi-Yadkori, Y, D Pál, C Szepesvári. 2011. Improved algorithms for linear stochastic bandits. *NIPS*.

Aflaki, S, I Popescu. 2013. Managing retention in service relationships. *Management Science* **60**(2).

Agarwal, Alekh, Ofer Dekel. 2010. Optimal algorithms for online convex optimization with multi-point bandit feedback. *COLT*. Citeseer, 28–40.

Agarwal, Alekh, Dean P Foster, Daniel J Hsu, Sham M Kakade, Alexander Rakhlin. 2011. Stochastic convex optimization with bandit feedback. *Advances in Neural Information Processing Systems*. 1035–1043.

Agrawal, Rajeev. 1995. The continuum-armed bandit problem. *SIAM journal on control and optimization* **33**(6) 1926–1951.

Agrawal, S, V Avadhanula, V Goyal, A Zeevi. 2016. A near-optimal exploration-exploitation approach for assortment selection. *Proceedings of the 2016 ACM Conference on Economics and Computation*. ACM, 599–600.

Agrawal, Shipra, Navin Goyal. 2013. Further optimal regret bounds for thompson sampling. *Artificial Intelligence and Statistics*.

Agresti, Alan. 2018. *An introduction to categorical data analysis*. Wiley.

Ahmad, Samreen. 2018. Indian e-commerce industry is expected to cross $100 billion mark by 2020 .

Ali, Özden Gür, Serpil Sayın, Tom Van Woensel, Jan Fransoo. 2009. Sku demand forecasting in the presence of promotions. *Expert Systems with Applications* **36** 12340–12348.

Alyoubi, Adel A. 2015. E-commerce in developing countries and how to develop them during the introduction of modern systems. *Procedia Computer Science* **65** 479–483.

Anderson, Eric T., Karsten Hansen, Duncan Simester. 2009. The option value of returns: Theory and empirical evidence. *Marketing Science* **28**(3) 405–423. doi:10.1287/mksc.1080.0430. URL https://doi.org/10.1287/mksc.1080.0430.

Armstrong, J Scott, Kersten C Green, Andreas Graefe. 2015. Golden rule of forecasting: Be conservative. *Journal of Business Research* **68** 1717–1731.

Auer, P. 2002. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research* **3**.

Auer, Peter, Ronald Ortner, Csaba Szepesvári. 2007. Improved rates for the stochastic continuum-armed bandit problem. *International Conference on Computational Learning Theory*. Springer, 454–468.

Aviv, Yossi, Gustavo Vulcano. 2012. Dynamic list pricing. *The Oxford handbook of pricing management*.

Bagirov, Adil M, Arshad Mahmood, Andrew Barton. 2017. Prediction of monthly rainfall in victoria, australia: Clusterwise linear regression approach. *Atmospheric research* **188** 20–29.

Ban, Gah-Yi, Jérémie Gallien, Adam Mersereau. 2017. Dynamic procurement of new products with covariate information: The residual tree method .

Ban, Gah-Yi, N Bora Keskin. 2017. Personalized dynamic pricing with machine learning .

Bandi, Chaithanya, Antonio Moreno, Zhiji Xu. 2017. The hidden costs of dynamic pricing: Empirical evidence from online retailing in emerging markets .

Bass, Frank M. 1969. A new product growth for model consumer durables. *Management Science* **15**(5) 215–227.

Bass, Frank M. 2004. Comments on "a new product growth for model consumer durables the bass model". *Management Science* **50**(12) 1833–1840.

Bastani, H, M Bayati, K Khosravi. 2017. Mostly exploration-free algorithms for contextual bandits. *arXiv preprint arXiv:1704.09011* .

Bastani, Hamsa, Mohsen Bayati. 2015. Online decision-making with high-dimensional covariates .

Bastani, Hamsa, David Simchi-Levi, Ruihao Zhu. 2019. Meta dynamic pricing: Learning across experiments. *Available at SSRN 3334629* .

Berry, Leonard L, Anantharanthan Parasuraman, Valerie A Zeithaml. 1994. Improving service quality in america: lessons learned. *Academy of Management Perspectives* **8**(2) 32–45.

Bertsimas, Dimitris, Martin S. Copenhaver. 2017. Characterization of the equivalence of robustification and regularization in linear and matrix regression. *European Journal of Operational Research* .

Bertsimas, Dimitris, Georgia Perakis. 2006. Dynamic pricing: A learning approach. *Mathematical and computational models for congestion charging*. Springer, 45–79.

Bertsimas, Dimitris, Romy Shioda. 2007. Classification and regression via integer optimization. *Operations Research* **55**(2) 252–271.

Bertsimas, Dimitris, Phebe Vayanos. 2017. Data-driven learning in dynamic pricing using adaptive optimization .

Besbes, O, Y Gur, A Zeevi. 2015. Optimization in online content recommendation services: Beyond click-through rates. *Manufacturing & Service Operations Management* .

Besbes, Omar, Assaf Zeevi. 2009. Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Operations Research* **57**(6) 1407–1420.

Besbes, Omar, Assaf Zeevi. 2015. On the (surprising) sufficiency of linear models for dynamic pricing with demand learning. *Management Science* **61**(4) 723–739.

Besson, Lilian, Emilie Kaufmann. 2018. What doubling tricks can and can't do for multi-armed bandits. *arXiv preprint arXiv:1803.06971* .

Bitran, Gabriel R, Susana V Mondschein. 1997. Periodic pricing of seasonal products in retailing. *Management science* **43**(1) 64–79.

Bowden, J L. 2009. The process of customer engagement: A conceptual framework. *Journal of Marketing Theory and Practice* **17**.

Breese, J S, D Heckerman, C Kadie. 1998. Empirical analysis of predictive algorithms for collaborative filtering. *UAI*. Morgan Kaufmann Publishers Inc.

Bresler, G, G H Chen, D Shah. 2014. A latent source model for online collaborative filtering. *NIPS*. 3347–3355.

Broder, Josef Meinrad. 2011. Online algorithms for revenue management. Ph.D. thesis, Cornell University.

Brusco, Michael J, J Dennis Cradit, Stephanie Stahl. 2002. A simulated annealing heuristic for a bicriterion partitioning problem in market segmentation. *Journal of Marketing Research* **39**(1) 99–109.

Bühlmann, Peter, Sara van de Geer. 2011. *Statistics for High-Dimensional Data: Methods, Theory and Applications*. Springer, New York.

Cameron, A Colin, Douglas L Miller. 2015. A practitioner's guide to cluster-robust inference. *Journal of Human Resources* **50**(2) 317–372.

Carbonneau, Réal A, Gilles Caporossi, Pierre Hansen. 2011. Globally optimal clusterwise regression by mixed logical-quadratic programming. *European Journal of Operational Research* **212**(1) 213–222.

Carbonneau, Réal A, Gilles Caporossi, Pierre Hansen. 2012. Extensions to the repetitive branch and bound algorithm for globally optimal clusterwise regression. *Computers & Operations Research* **39**(11) 2748–2762.

Carmichael, Sarah Green. 2014. The silent killer of new products: Lazy pricing URL https://hbr.org/2014/09/the-silent-killer-of-new-products-lazy-pricing.

Caro, Felipe, Jérémie Gallien. 2010. Inventory management of a fast-fashion retail network. *Operations Research* **58**(2) 257–273.

Cecere, Lora. 2013. New products: More costly and more important. URL https://www.forbes.com/sites/loracecere/2013/12/11/new-products-more-costly-and-more-important.

Chandrasekaran, Deepa, Gerard J Tellis. 2007. A critical review of marketing research on diffusion of new products. *Review of marketing research*. Emerald Group Publishing Limited, 39–80.

Chapelle, O, L Li. 2011. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*.

Chen, Bintong, Jing Chen. 2017. When to introduce an online channel, and offer money back guarantees and personalized pricing? *European Journal of Operational Research* **257**(2) 614–624.

Chen, Boxiao, Xiuli Chao. 2017. Data-based dynamic pricing and inventory control with censored demand and limited price changes. *Available at SSRN 2700747* .

Chen, Boxiao, Xiuli Chao, Cong Shi. 2017a. Nonparametric algorithms for joint pricing and inventory control with lost-sales and censored demand URL https://pdfs.semanticscholar.org/db4e/2e0e61ac587afa36c9ea02d2d2dda8298d1b.pdf.

Chen, Hui, Ann Melissa Campbell, Barrett W Thomas. 2008a. Network design for time-constrained delivery. *Naval Research Logistics (NRL)* **55**(6) 493–515.

Chen, Kay-Yut, Murat Kaya, Özalp Özer. 2008b. Dual sales channel management with service competition. *Manufacturing & Service Operations Management* **10**(4) 654–675.

Chen, Li, Adam J Mersereau, Zhe Wang. 2017b. Optimal merchandise testing with limited inventory. *Operations Research* .

Chen, Ningyuan, Guillermo Gallego. 2018. A primal-dual learning algorithm for personalized dynamic pricing with an inventory constraint. *Available at SSRN* .

Chen, Qi, Stefanus Jasin, Izak Duenyas. 2015. Real-time dynamic pricing with minimal and flexible price adjustment. *Management Science* .

Chen, Y, Y Chi. 2018. Harnessing structures in big data via guaranteed low-rank matrix estimation. *arXiv preprint arXiv:1802.08397* .

Cheung, Wang Chi, David Simchi-Levi, He Wang. 2015. Dynamic pricing and demand learning with limited price experimentation. *Available at SSRN 2457296* .

Chirico, Paolo. 2013. A clusterwise regression method for the prediction of the disposal income in municipalities. *Classification and Data Mining*. Springer, 173–180.

Cohen, Maxime, Michael-David Fiszer, Baek Jung Kim. 2018. Frustration-based promotions: Field experiments in ride-sharing. *Available at SSRN 3129717* .

Cohen, Maxime, Ilan Lobel, Renato Paes Leme. 2016. Feature-based dynamic pricing. *Available at SSRN 2737045* .

Cohen, Maxime C, Ngai-Hang Zachary Leung, Kiran Panchamgam, Georgia Perakis, Anthony Smith. 2017. The impact of linear optimization on promotion planning. *Operations Research* **65**(2) 446–468.

Cohen, Maxime C, Georgia Perakis, Robert S Pindyck. 2015. Pricing with limited knowledge of demand .

Davis, M M, T E Vollmann. 1990. A framework for relating waiting time and customer satisfaction in a service operation. *Journal of Services Marketing* **4**(1).

Davis, Scott, Eitan Gerstner, Michael Hagerty. 1995. Money back guarantees in retailing: Matching products to consumer tastes. *Journal of Retailing* **71**(1) 7–22.

Demirezen, E M, S Kumar. 2016. Optimization of recommender systems based on inventory. *Production and Operations Management* **25**(4).

den Boer, Arnoud V. 2015. Dynamic pricing and learning: historical origins, current research, and new directions. *Surveys in operations research and management science* **20**(1) 1–18.

den Boer, Arnoud V, Bert Zwart. 2013. Simultaneously learning and optimizing using controlled variance pricing. *Management Science* **60**(3) 770–783.

DeSarbo, Wayne S, William L Cron. 1988. A maximum likelihood methodology for clusterwise linear regression. *Journal of classification* **5**(2) 249–282.

DeSarbo, Wayne S, Richard L Oliver, Arvind Rangaswamy. 1989. A simulated annealing methodology for clusterwise linear regression. *Psychometrika* **54**(4) 707–736.

Dokka Venkata Satyanaraya, Trivikram, Peter Jacko, Waseem Aslam. 2018. Non-parametric dynamic pricing: a non-adversarial robust optimization approach .

Duchi, John C, Michael I Jordan, Martin J Wainwright, Andre Wibisono. 2015. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory* **61**(5) 2788–2806.

Dunning, Iain, Joey Huchette, Miles Lubin. 2017. Jump: A modeling language for mathematical optimization. *SIAM Review* **59**(2) 295–320.

D'Urso, Pierpaolo, Riccardo Massari, Adriana Santoro. 2010. A class of fuzzy clusterwise regression models. *Information Sciences* **180**(24) 4737–4762.

Ekstrand, M D, J T Riedl, J A Konstan, et al. 2011. Collaborative filtering recommender systems. *Foundations and Trends® in Human–Computer Interaction* **4**.

Elmachtoub, Adam N, Vishal Gupta, Michael Hamilton. 2018. The value of personalized pricing. *Available at 3127719* .

Fan, Tingting, Peter N. Golder, Donald R. Lehmann. 2017a. *Handbook of Marketing Decision Models*, *International Series in Operations Research & Management Science*, vol. 254, chap. 3. Springer, Cham, 79–116.

Fan, Zhi-Ping, Yu-Jie Che, Zhen-Yu Chen. 2017b. Product sales forecasting using online reviews and historical sales data: A method combining the bass model and sentiment analysis. *Journal of Business Research* **74** 90–100.

Farias, V F, A A Li. 2017. Learning preferences with side information .

Feng, Youyi, Guillermo Gallego. 1995. Optimal starting times for end-of-season sales and optimal stopping times for promotional fares. *Management Science* **41**(8) 1371–1391.

Ferreira, Kris Johnson, Bin Hong Alex Lee, David Simchi-Levi. 2016. Analytics for an online retailer: Demand forecasting and price optimization. *Manufacturing & Service Operations Management* **18**(1) 69–88.

Fisher, Marshall, Santiago Gallino, Joseph Xu. 2016. The value of rapid delivery in online retailing .

Fisher, Marshall, Ananth Raman. 1996. Reducing the cost of demand uncertainty through accurate response to early sales. *Operations Research* **44**(1) 87–99.

Fitzsimons, G J, D R Lehmann. 2004. Reactance to recommendations: When unsolicited advice yields contrary responses. *Marketing Science* **23**(1).

Ford, Eric W., Bradford W. Hesse, Timothy R. Huerta. 2016. Personal health record use in the united states: Forecasting future adoption levels. *Journal of Medical Internet Research* **18**(3) e73.

Gallien, Jérémie, Adam J Mersereau, Andres Garro, Alberte Dapena Mora, Martín Nóvoa Vidal. 2015. Initial shipment decisions for new products at zara. *Operations Research* **63**(2) 269–286.

Gallino, Santiago, Antonio Moreno. 2018. The value of fit information in online retail: Evidence from a randomized field experiment. *Manufacturing & Service Operations Management* **20**(4) 767–787. doi:10.1287/msom.2017.0686. URL https://doi.org/10.1287/msom.2017.0686.

Garbarino, Ellen, Olivia F Lee. 2003. Dynamic pricing in internet retail: effects on consumer trust. *Psychology & Marketing* **20**(6) 495–513.

Ghadimi, Saeed, Guanghui Lan. 2013. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization* **23**(4) 2341–2368.

Gopalan, A, M Maillard, Oand Zaki. 2016. Low-rank bandits with latent mixtures. *arXiv preprint arXiv:1609.01508* .

Guide Jr, V Daniel R, Gilvan C Souza, Luk N Van Wassenhove, Joseph D Blackburn. 2006. Time value of commercial product returns. *Management Science* **52**(8) 1200–1214.

Handel, Benjamin R, Kanishka Misra. 2015. Robust new product pricing. *Marketing Science* **34**(6) 864–881.

Harper, F M, J A Konstan. 2016. The movielens datasets: History and context. *Acm transactions on interactive intelligent systems (tiis)* **5** 19.

Haws, Kelly L, William O Bearden. 2006. Dynamic pricing and consumer fairness perceptions. *Journal of Consumer Research* **33**(3) 304–311.

Herlocker, J L, J A Konstan, L G Terveen, J T Riedl. 2004. Evaluating collaborative filtering recommender systems. *TOIS* **22**(1).

Hu, Kejia, Jason Acimovic, Francisco Erize, Douglas J Thomas, Jan A Van Mieghem. 2016. Forecasting product life cycle curves: Practical approach and empirical analysis .

Huang, Hong-Zhong, Zhi-Jie Liu, DNP Murthy. 2007. Optimal reliability, warranty and price for new products. *Iie Transactions* **39**(8) 819–827.

Huang, Tao, Robert Fildes, Didier Soopramanien. 2014. The value of competitive information in forecasting fmcg retail product sales and the variable selection problem. *European Journal of Operational Research* **237**(2) 738 – 748.

Imbens, Guido. 2014. Instrumental variables: An econometrician's perspective. Tech. rep., National Bureau of Economic Research.

Jasin, Stefanus, Yanzhe Lei, Amitabh Sinha. 2015. Multidimensional bisection search for constrained optimization with noisy observations doi:10.13140/RG.2.1.3429.3204.

Javanmard, Adel, Hamid Nazerzadeh. 2016. Dynamic pricing in high-dimensions .

Johari, R, V Kamble, Y Kanoria. 2017. Matching while learning. *Proceedings of the 2017 ACM Conference on Economics and Computation*. ACM.

Johari, Ramesh, Sven Schmit. 2018. Learning with abandonment. *arXiv preprint arXiv:1802.08718* .

Kahn, Kenneth B. 2002. An exploratory investigation of new product forecasting practices. *The Journal of Product Innovation Management* **19** 133–143.

Kahn, Kenneth B. 2014. Solving the problems of new product forecasting. *Business Horizons* **57**(5) 607–615.

Kallus, N, M Udell. 2016. Dynamic assortment personalization in high dimensions. *arXiv* .

Kanoria, Y, I Lobel, J Lu. 2018. Managing customer churn via service mode control .

Keeney, Ralph L. 1999. The value of internet commerce to the customer. *Management Science* **45**(4) 533–542. doi:10.1287/mnsc.45.4.533. URL https://doi.org/10.1287/mnsc.45.4.533.

Keskin, N Bora, Assaf Zeevi. 2014. Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Operations Research* **62**(5) 1142–1167.

Kleinberg, Robert D. 2005. Nearly tight bounds for the continuum-armed bandit problem. *Advances in Neural Information Processing Systems*. 697–704.

Kuczma, Marek. 2009. *An introduction to the theory of functional equations and inequalities: Cauchy's equation and Jensen's inequality*. Springer Science & Business Media.

Lai, Tze Leung, Herbert Robbins. 1985. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics* **6**(1) 4–22.

Lattimore, T, C Szepesvari. 2016. The end of optimism? an asymptotic analysis of finite-armed linear bandits. *arXiv* .

Lau, Kin-nam, Pui-lam Leung, Ka-kit Tse. 1999. A mathematical programming approach to clusterwise regression model and its extensions. *European Journal of Operational Research* **116**(3) 640–652.

Lei, Yanzhe Murray, Stefanus Jasin, Amitabh Sinha. 2014. Near-optimal bisection search for nonparametric dynamic pricing with inventory constraint. *Available at SSRN 2509425* .

Li, S, A Karatzoglou, C Gentile. 2016. Collaborative filtering bandits. *SIGIR*. ACM.

Li, Xiaopeng. 2013. An integrated modeling framework for design of logistics networks with expedited shipment services. *Transportation Research Part E: Logistics and Transportation Review* **56** 46–63.

Lika, B, K Kolomvatsos, S Hadjiefthymiades. 2014. Facing the cold start problem in recommender systems. *Expert Systems with Applications* **41**(4) 2065–2073.

Linden, G, B Smith, J York. 2003. Amazon. com recommendations: Item-to-item collaborative filtering. *IEEE Internet computing* (1).

Lu, Y, Andrés Musalem, M Olivares, A Schilkrut. 2013. Measuring the effect of queues on customer purchases. *Management Science* **59**(8).

Manwani, Naresh, PS Sastry. 2015. K-plane regression. *Information Sciences* **292** 39–56.

Massiani, Jérôme, Andreas Gohs. 2015. The choice of bass model coefficients to forecast diffusion for innovative products: An empirical investigation for new automotive technologies. *Research in Transportation Economics* **50** 17–28.

Megiddo, Nimrod, Arie Tamir. 1982. On the complexity of locating linear facilities in the plane. *Operations research letters* **1**(5) 194–197.

Murthi, BPS, S Sarkar. 2003. The role of the management sciences in research on personalization. *Management Science* .

Nageswaran, Leela, Soo-Haeng Cho, Alan Andrew Scheller-Wolf. 2017. Consumer return policies in omnichannel operations .

Nerlove, M, K J Arrow. 1962. Optimal advertising policy under dynamic conditions. *Economica* .

Nesterov, Yu. 2011. Random gradient-free minimization of convex functions .

Netessine, Serguei. 2006. Dynamic pricing of inventory/capacity with infrequent price changes. *European Journal of Operational Research* **174**(1) 553–580.

Oliver, Richard L. 1980. A cognitive model of the antecedents and consequences of satisfaction decisions. *Journal of marketing research* **17**(4) 460–469.

Park, Young Woong, Yan Jiang, Diego Klabjan, Loren Williams. 2016. Algorithms for generalized cluster-wise linear regression. *arXiv preprint arXiv:1607.01417* .

Perchet, Vianney, Philippe Rigollet, Sylvain Chassang, Erik Snowberg, et al. 2016. Batched bandit problems. *The Annals of Statistics* **44**(2) 660–681.

Petersen, J Andrew, V Kumar. 2009. Are product returns a necessary evil? antecedents and consequences. *Journal of Marketing* **73**(3) 35–51.

PK Kannan, Praveen K Kopalle. 2001. Dynamic pricing on the internet: Importance and implications for consumer behavior. *International Journal of Electronic Commerce* **5**(3) 63–83.

Powers, Thomas L, Eric P Jack. 2013. The influence of cognitive dissonance on retail product returns. *Psychology & marketing* **30**(8) 724–735.

Proserpio, Davide, Georgios Zervas. 2017. Online reputation management: Estimating the impact of management responses on consumer reviews. *Marketing Science* **36**(5) 645–665.

Qiang, Sheng, Mohsen Bayati. 2016. Dynamic pricing with demand covariates. *Available at SSRN 2765257* .

Rabinovich, Elliot, Joseph P Bailey. 2004. Physical distribution service quality in internet retailing: service pricing, transaction attributes, and firm attributes. *Journal of Operations Management* **21**(6) 651–672.

Ramnath, N S. 2016. Indian e-commerce in 10 charts. URL http://www.foundingfuel.com/slideshow/indian-ecommerce-in-10-charts/.

Rao, Shashank, Elliot Rabinovich, Dheeraj Raju. 2014. The role of physical distribution services as determinants of product returns in internet retailing. *Journal of Operations Management* **32**(6) 295–312.

Reagan, Courtney. 2016. A $ 260 billion 'ticking time bomb': The costly business of retail returns. *A $ 260 billion 'ticking time bomb': The costly business of retail returns* URL https://www.cnbc.com/2016/12/16/a-260-billion-ticking-time-bomb-the-costly-business-of-retail-returns.html.

Rogers, Dale S, Ronald Tibben-Lembke. 2001. An examination of reverse logistics practices. *Journal of business logistics* **22**(2) 129–148.

Rothschild, Michael. 1974. A two-armed bandit theory of market pricing. *Journal of Economic Theory* **9**(2) 185–202.

Rusmevichientong, P, J N Tsitsiklis. 2010. Linearly parameterized bandits. *Mathematics of Operations Research* **35**(2).

Russo, D, B Van Roy. 2014. Learning to optimize via posterior sampling. *Mathematics of Operations Research* **39**(4).

Russo, D, B Van Roy. 2018. Satisficing in time-sensitive bandit learning. *arXiv* .

Sarwar, B, G Karypis, J Konstan, J Riedl. 2001. Item-based collaborative filtering recommendation algorithms. *WWW*. ACM.

Schein, A I, A Popescul, L H Ungar, D M Pennock. 2002. Methods and metrics for cold-start recommendations. *SIGIR*. ACM.

Schlittgen, Rainer. 2011. A weighted least-squares approach to clusterwise regression. *AStA Advances in Statistical Analysis* **95**(2) 205–217.

Shah, V, J Blanchet, R Johari. 2018. Bandit learning with positive externalities .

Shamir, Ohad. 2017. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *The Journal of Machine Learning Research* **18**(1) 1703–1713.

Simchi-Levi, David, Yunzong Xu. 2019. Phase transitions and cyclic phenomena in bandits with switching constraints. *Advances in Neural Information Processing Systems*. 7521–7530.

Simchi-Levi, David, Yunzong Xu, Jinglong Zhao. 2019. Network revenue management with limited switches: Known and unknown demand distributions. *Available at SSRN 3479477* .

Singh, Shweta. 2015. Cod logistic service providers. URL https://indianonlineseller.com/2015/11/cod-logistic-service-providers-for-your-ecommerce-business/.

Smith, Amy K, Ruth N Bolton. 1998. An experimental investigation of customer reactions to service failure and recovery encounters: paradox or peril? *Journal of service research* **1**(1) 65–81.

Smith, C. 2018. 90 interesting email statistics and facts. URL https://expandedramblings.com/index.php/email-statistics/.

Somerville, Paul N. 1954. Some problems of optimum sampling. *Biometrika* **41**(3/4) 420–429.

Sousa, R, C Voss. 2012. The impacts of e-service quality on customer behaviour in multi-channel e-services. *Total Quality Management & Business Excellence* **23**.

Späth, Helmuth. 1979. Algorithm 39: Clusterwise linear regression. *Computing* **22**(4) 367–373.

Su, X, T M Khoshgoftaar. 2009. A survey of collaborative filtering techniques. *Advances in artificial intelligence* **2009**.

Su, Xuanming. 2009. Consumer returns policies and supply chain performance. *Manufacturing & Service Operations Management* **11**(4) 595–612.

Süli, Endre, David F Mayers. 2003. *An introduction to numerical analysis*. Cambridge university press.

Surprenant, C F, M R Solomon. 1987. Predictability and personalization in the service encounter. *the Journal of Marketing* .

Swaminathan, Jayashankar, Sridhar Tayur. 2003. Models for supply chains in e-business **49** 1387–1406.

Tan, T F, S Netessine, L Hitt. 2017. Is tom cruise threatened? an empirical study of the impact of product variety on demand concentration. *Information Systems Research* **28**(3).

Venetis, K A, P N Ghauri. 2004. Service quality and customer retention: building long-term relationships. *European Journal of marketing* **38**.

Viele, Kert, Barbara Tong. 2002. Modeling with mixtures of linear regressions. *Statistics and Computing* **12**(4) 315–330.

Vielma, Juan Pablo, Shabbir Ahmed, George Nemhauser. 2010. Mixed-integer models for nonseparable piecewise-linear optimization: Unifying framework and extensions. *Operations research* **58**(2) 303–315.

Wei, J, J He, K Chen, Y Zhou, Z Tang. 2017. Collaborative filtering and deep learning based recommendation system for cold start items. *Expert Systems with Applications* **69** 29–39.

Weiner, Bernard. 2000. Attributional thoughts about consumer behavior. *Journal of Consumer research* **27**(3) 382–387.

Welch, B L. 1951. On the comparison of several mean values: an alternative approach. *Biometrika* .

Willemot, Nicolas, Mike Booker, Kara Gruver, Amélie Dumont. 2015. Innovation in consumer goods: Heroes to the rescue URL https://www.bain.com/insights/innovation-in-consumer-goods-heroes-to-the-rescue/.

Zaroban, Stefany. 2018. U.s. e-commerce sales grow 16.0% in 2017. https://www.digitalcommerce360.com/article/us-ecommerce-sales/. Accessed: 2018-09-07.