Computer Science and Artificial Intelligence Laboratory

Technical Report

MIT-CSAIL-TR-2003-016
AIM-2003-019

August 28, 2003

# Permutation Tests for Classification

Sayan Mukherjee, Polina Golland, and Dmitry Panchenko

massachusetts institute of technology, cambridge, ma 02139 usa — www.csail.mit.edu

## Abstract

We introduce and explore an approach to estimating statistical significance of classification accuracy, which is particularly useful in scientific applications of machine learning where high dimensionality of the data and the small number of training examples render most standard convergence bounds too loose to yield a meaningful guarantee of the generalization ability of the classifier. Instead, we estimate statistical significance of the observed classification accuracy, or the likelihood of observing such accuracy by chance due to spurious correlations of the high-dimensional data patterns with the class labels in the given training set. We adopt permutation testing, a non-parametric technique previously developed in classical statistics for hypothesis testing in the generative setting (i.e., comparing two probability distributions). We demonstrate the method on real examples from neuroimaging studies and DNA microarray analysis and suggest a theoretical analysis of the procedure that relates the asymptotic behavior of the test to the existing convergence bounds.

**Keywords:** Classification, Permutation testing, Statistical significance, Non-parametric tests, Rademacher processes.

## Acknowledgments

The authors would like thank Pablo Tamayo, Vladimir Koltchinskii, Jill Mesirov, Todd Golub and Bruce Fischl for useful discussions. Sayan Mukherjee was supported by a SLOAN/DOE grant. Polina Golland was supported in part by NSF IIS 9610249 grant and Athinoula A. Martinos Center for Biomedical Imaging collaborative research grant. Dmitry Panchenko was partially supported by AT&T Buchsbaum grant.

The authors would like to acknowledge Dr. M. Spiridon and Dr. N. Kanwisher for providing the fMRI data, Dr. R. Buckner for providing the cortical thickness data and Dr. D. Greve and Dr. B. Fischl for help with registration and feature extraction in the MRI experiments discussed in this paper. Dr. Kanwisher would like to acknowledge EY 13455 and MH 59150 grants. Dr. Buckner would like to acknowledge the assistance of the Washington University ADRC, James S McDonnell Foundation, the Alzheimer's Association, and NIA grants AG05682 and AG03991. Dr. B. Fischl would like to acknowledge NIH R01 RR16594-01A1 grant. The Human Brain Project/Neuroinformatics research is funded jointly by the NINDS, the NIMH and the NCI (R01-NS39581). Further support was provided by the NCRR (P41-RR14075 and R01-RR13609).

## 1. Introduction

Many scientific studies involve detection and characterization of predictive patterns in high dimensional measurements, which can often be reduced to training a binary classifier or a regression model. We will use two examples of such applications to illustrate the techniques in this paper: medical image studies and gene expression analysis. Image-based clinical studies of brain disorders attempt to detect neuroanatomical changes induced by diseases, as well as predict development of the disease. The goals of gene expression analysis include classification of the tissue morphology and prediction of the treatment outcome from DNA microarray data. In both fields, training a classifier to reliably label new examples into the healthy population or one of the disease sub-groups can help to improve screening and early diagnostics, as well as provide an insight into the nature of the disorder. Both imaging data and DNA microarray measurements are characterized by high dimensionality of the input space (thousands of features) and small datasets (tens of independent examples), typical of many biological applications.

For statistical learning to be useful in such scientific applications, it must provide an estimate of the significance of the detected differences, i.e., a guarantee of how well the results of learning describe the entire population. machine learning theory offers two types of such guarantees, both based on estimating the expected error of the resulting classifier function. The first approach is to estimate the test error on a hold-out set – or by applying a cross-validation procedure, such as a jackknife or bootstrap (Efron, 1982) – which, in conjunction with a variance-based convergence bound, provides a confidence interval (i.e., the interval that with high probability contains the true value) for the expected error. Small sample sizes render this approach ineffective as the variance of the error on a hold-out set is often too large to provide a meaningful estimate on how close we are to the true error. Applying variance-based bounds to the cross-validation error estimates produces misleading results as the cross-validation iterations are not independent, causing us to underestimate the variance. An alternative, but equally fruitless, approach is to use bounds on the convergence of the empirical training error to the expected test error. For very high dimensional data, the training error is always zero and the bounds are extremely loose. Thus we are often forced to conclude that, although the classification accuracy looks promising, we need significantly more data (several orders of magnitude more than currently available) before the standard bounds provide a meaningful confidence interval for the expected error (Guyon et al., 1998). Since collecting data in these applications is often expensive in terms of time and resources, it is desirable to obtain a quantitative indicator of how robust is the observed classification accuracy, long before the asymptotic bounds apply, for the training set sizes in tens to hundreds examples. This is particularly relevant since the empirical results often indicate that efficient learning is possible with far fewer examples than predicted by the convergence bounds.

In this paper, we demonstrate how a weaker guarantee, that of statistical significance, can still be provided for classification results on a small number of examples. We consider the question on the differences between the two classes, as measured by the classifier performance on a test set, in the framework of hypothesis testing traditionally used in statistics. Intuitively, statistical significance is a measure of how likely we were to obtain the observed test accuracy by chance, only because the training algorithm identified some pattern in the

high-dimensional data that happened to correlate with the class labels as an artifact of a small data set size. Our goal is to reject the null hypothesis, namely that a given family of classifiers cannot learn to accurately predict the labels of a test point given a training set. The empirical estimate of the expected test error can serve as a test statistic that measures how different the two classes are with respect to the family of classifiers we use in training. We adopt permutation tests (Good, 1994, Kendall, 1945), a non-parametric technique developed in the statistics literature, to estimate the probability distribution of the statistic under the null hypothesis from the available data. We employ the permutation procedure to construct an empirical estimate of the error distribution and then use this estimate to assign a significance to the observed classification error.

In the next section, we provide the necessary background on hypothesis testing and permutation tests. In Section 3, we extend the permutation procedure to estimate statistical significance of classification results. Section 4 demonstrates the application of the test on two detailed examples, reports results for several studies from the fields of brain imaging and gene expression analysis and offers practical guidelines for applying the procedure. In Section 5, we suggest a theoretical analysis of the procedure that leads to convergence bounds governed by similar quantities to those that control standard empirical error bounds, closing with a brief discussion of open questions.

## 2. Background. Hypothesis Testing and Permutations

In two-class comparison hypothesis testing, the differences between two data distributions are measured using a dataset statistic

$$\mathcal{T} : (\mathbb{R}^n \times \{-1, 1\})^* \mapsto \mathbb{R},$$

such that for a given dataset $\{(\mathbf{x}_k, y_k)\}_{k=1}^l$, where $\mathbf{x}_k \in \mathbb{R}^n$ are observations and $y_k \in \{-1, 1\}$ are the corresponding class labels, $\mathcal{T}(\mathbf{x}_1, y_1, \ldots, \mathbf{x}_l, y_l)$ is a measure of the similarity of the subsets $\{\mathbf{x}_k | y_k{=}1\}$ and $\{\mathbf{x}_k | y_k{=}{-}1\}$. The null hypothesis typically assumes that the two conditional probability distributions are identical, $p(\mathbf{x}|y{=}1) = p(\mathbf{x}|y{=}{-}1)$, or equivalently, that the data and the labels are independent, $p(\mathbf{x}, y) = p(\mathbf{x})p(y)$. The goal of the hypothesis test is to reject the null hypothesis at a certain level of significance $\alpha$ which sets the maximal acceptable probability of false positive (declaring that the classes are different when the null hypothesis is true). For any value of the statistic, the corresponding *p-value* is the highest level of significance at which the null hypothesis can still be rejected. In classical statistics, the data are often assumed to be one-dimensional ($n = 1$).

For example, in the two-sample t-test, the data in the two classes are assumed to be generated by one-dimensional Gaussian distributions of equal variance. The null hypothesis is that the distributions have the same mean. The distribution of the *t-statistic*, the difference between the sample means normalized by the standard error, under the null hypothesis is the Student's distribution (Sachs, 1984, Student, 1908). If the integral of Student's distribution over the values higher than the observed t-statistic is smaller than the desired significance level $\alpha$, we reject the null hypothesis in favor of the alternative hypothesis that the means of the two distributions are different.

In order to perform hypothesis testing, we need to know the probability distribution of the selected statistic under the null hypothesis. In general, the distribution for a particular

statistic cannot be computed without making strong assumptions on the generative model of the data. Non-parametric techniques, such as permutation tests, can be of great value if the distribution of the data is unknown. Permutation tests were first introduced as a non-parametric alternative to the one-dimensional t-test and have been used to replace Student's distribution when the normality of the data distribution could not be assured. Here, we describe the general formulation of the test as applied to any statistic $\mathcal{T}$.

Suppose we have chosen an appropriate statistic $\mathcal{T}$ and the acceptable significance level $\alpha$. Let $\Pi_l$ be a set of all permutations of indices $1, \ldots, l$, where $l$ is the number of independent examples in the dataset. The permutation test procedure that consists of $M$ iterations is defined as follows:

- Repeat $M$ times (with index $m = 1, \ldots, M$):
  - sample a permutation $\boldsymbol{\pi}^m$ from a uniform distribution over $\Pi_l$,
  - compute the statistic value for this permutation of labels
$$t^m = \mathcal{T}(\mathbf{x}_1, y_{\pi_1^m}, \ldots, \mathbf{x}_l, y_{\pi_l^m}).$$
- Construct an empirical cumulative distribution
$$\hat{P}(T \leq t) = \frac{1}{M} \sum_{m=1}^{M} \Theta(t - t^m),$$
where $\Theta$ is a step-function ($\Theta(x) = 1$, if $x \geq 0$; 0 otherwise).
- Compute the statistic value for the actual labels, $t_0 = \mathcal{T}(\mathbf{x}_1, y_1, \ldots, \mathbf{x}_l, y_l)$ and its corresponding p-value $p_0$ under the empirical distribution $\hat{P}$. If $p_0 \leq \alpha$, reject the null hypothesis.

The procedure computes an empirical estimate of the cumulative distribution of the statistic $\mathcal{T}$ under the null hypothesis and uses it for hypothesis testing. Since the null hypothesis assumes that the two classes are indistinguishable with respect to the selected statistic, all the training datasets generated through permutations are equally likely to be observed under the null hypothesis, yielding the estimates of the statistic for the empirical distribution. An equivalent result is obtained if we choose to permute the data, rather than the labels. Ideally, we would like to use the entire set of permutations $\Pi_l$ to construct the empirical distribution $\hat{P}$, but it might be not feasible for computational reasons. Instead, we resort to sampling from $\Pi_l$. It is therefore important to select the number of sampling iterations $M$ to be large enough to guarantee accurate estimation. One solution is to monitor the rate of change in the estimated distribution and stop when the changes are below an acceptable threshold. The precision of the test therefore depends on the number of iterations and the number of distinct labelings of the training set.

To better understand the difference between the parametric approach of the t-test and permutation testing, observe that statistical significance does not provide an absolute measure of how robust the observed differences are, but it is rather contingent upon certain assumptions about the data distribution in each class $p(\mathbf{x}|y)$ being true. The t-test assumes that the distribution of data in each class is Gaussian, while the permutation test assumes that the data distribution is adequately represented by the sample data. Neither estimates how well the sample data describe the general population, which is one of the fundamental questions in statistical learning theory and is outside the scope of this paper.

## 3. Permutations Tests for Classification

Permutation tests can be used to assess statistical significance of the classifier and its performance using an empirical estimate of the test error as a statistic that measures dissimilarity between two populations. Depending on the amount of the available data, the test error can be estimated on a large hold-out set or using cross-validation in every iteration of the permutation procedure. The null hypothesis assumes that the relationship between the data and the labels cannot be learned reliably by the family of classifiers used in the training step. The alternative hypothesis is that we can train a classifier with small expected error.

We use permutations to estimate the empirical cumulative distribution of the classifier error under the null hypothesis. For any value of the estimated error $e$, the appropriate p-value is $\hat{P}(e)$ (i.e., the probability of observing classification error lower than $e$). We can reject the null hypothesis and declare that the classifier learned the (probabilistic) relationship between the data and the labels with a risk of being wrong with probability of at most $\hat{P}(e)$.

The permutation procedure is equivalent to sampling new training sets from a probability distribution that can be factored into the original marginals for the data and the labels, $\tilde{p}(\mathbf{x}, y) = p(\mathbf{x})p(y)$ because it leaves the data unchanged and maintains relative frequencies of the labels through permutation, while destroying any relationship between the data and the labels. The test evaluates the likelihood of test error estimate on the original dataset relative to the empirical distribution of the error on data sets sampled from $\tilde{p}(\mathbf{x}, y)$.

To underscore the point made in the previous section, the test uses only the available data examples to evaluate the complexity of the classification problem, and is therefore valid only to the extent that the available dataset represents the true distribution $p(\mathbf{x}, y)$. Unlike standard convergence bounds, such as bounds based on VC-dimension, the empirical probability distribution of the classification error under the null hypothesis says nothing about how well the estimated error rate will generalize. Thus permutation tests provide a weaker guarantee than the convergence bounds, but they can still be useful in testing if the observed classification results are likely to be obtained by chance, due to a spurious pattern correlated with the labels in the given small data set.

Note that the estimated empirical distribution also depends on the classifier family and the training algorithm used to construct the classifier function. It essentially estimates the expressive power of the classifier family with respect to the training dataset. The variance of the empirical distribution $\hat{P}$ constructed by the permutation test is a function two quantities: the randomness due to the small sample size and the difficulty of the classification problem. The varaince decreases with more samples and easier classification problems.

## 4. Application of the Test

We use permutation testing in our work to assess the significance of the observed classification accuracy before we conclude that the results obtained in the cross-validation procedure are robust, or decide that more data are needed before we can trust the detected pattern or trend in the biological data. In this section, we first demonstrate the procedure in detail on two different examples, a study of changes in the cortical thickness due to Alzheimer's disease using MRI scans for measurement and a discrimination between two types of leukemia

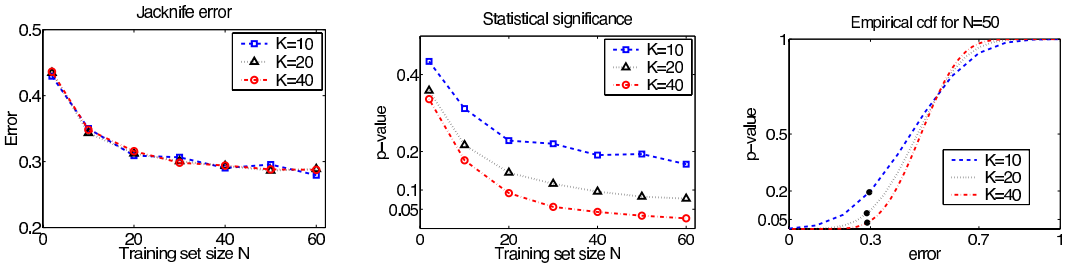


Figure 1: Estimated test error (left) and statistical significance (middle) computed for different training set sizes $N$ and test set sizes $K$, and empirical error distribution (right) constructed for $N = 50$ and different test set sizes $K$ in the cortical thickness study. Filled circles on the right graph indicate the classifier performance on the true labels ($K = 10$: $e = .30$, $p = .19$; $K = 20$: $e = .29$, $p = .08$; $K = 40$: $e = .29$, $p = .03$).

based on DNA microarray data. These studies involve very different types of data, but both share small sample size and high dimensionality of the original input space, rendering the convergence bounds extremely loose. We then report experimental results on more examples from real biological studies and offer practical guidelines on application of the test.

In all experiments reported in this section, we used linear Support Vector Machines (Vapnik, 1998) to train a classifier, and jackknifing (i.e., sampling without replacement) for cross-validation. The number of cross-validation iterations was $1,000$, and the number of permutation iterations was $10,000$.

### 4.1 Detailed Examples

The first example compares the thickness of the cortex in 50 patients diagnosed with dementia of the Alzheimer type and 50 normal controls of matched age. The cortical sheet was automatically segmented from each MRI scan, followed by a registration step that brought the surfaces into correspondence by mapping them onto a unit sphere while minimizing distortions and then aligning the cortical folding patterns (Fischl et al., 1999, Fischl and Dale, 2000). The cortical thickness was densely sampled on a 1mm grid at corresponding locations for all subjects, resulting in over 300,000 thickness measurements. The measurements in neighboring locations are highly correlated, as both the pattern of thickness and the pattern of its change are smooth over the surface of the cortex, leading us to believe that learning the differences between the two groups might be possible with a reasonable number of examples.

We start by studying the behavior of the estimated error and its statistical significance as a function of training set size and test set size, reported in Figure 1. Every point in the first two graphs is characterized by a corresponding training set size $N$ and test set size $K$, drawn from the original dataset. In permutation testing, the labels of the training data are permuted prior to training. It is not surprising that increasing the number of training examples improves the robustness of classification as exhibited by both the accuracy and the significance estimates. By examining the left graph, we conclude that at approximately

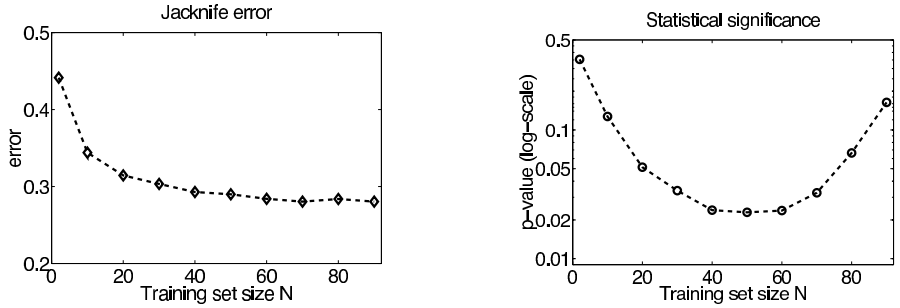


Figure 2: Estimated test error and statistical significance for different training set sizes $N$ for the cortical thickness study. Unlike the experiments in Figure 1, all of the examples unused in training were used to test the classifier. The p-values are shown on a logarithmic scale.

$N = 40$, the accuracy of the classification saturates at 71% ($e = .29$). In testing for significance, we typically expect to work in this region of relatively slow change in the estimated test error.

Increasing the number of independent examples on which we test the classifier in each iteration does not significantly affect the estimated classification error, but substantially improves statistical significance of the same error value, as to be expected: when we increase the test set size, a classifier trained on a random labeling of the training data is less likely to maintain the same level of testing accuracy. The right graph in Figure 1 illustrates this point for a particular training set size of $N = 50$ (well within the saturation range for the expected error estimates). It shows the empirical distribution $\hat{P}(e)$ curves for the test set sizes $K = 10, 20, 40$. The filled circles represent classification performance on the true labels and the corresponding p-values. We note again that the three circles represent virtually the same accuracy, but substantially different p-values. For this training set size, if we set the significance threshold at $\alpha = .05$, testing on $K = 40$ achieves statistical significance ($p = .03$, i.e., 3% chance of observing better than 71% accuracy of cross-validation if the data and the labels in this problem are truly independent).

In the experiments described above, most iterations of cross-validation and permutation testing did not use all available examples (i.e., $N + K$ was less than the total number of examples). These were constructed to illustrate the behavior of the test for the same test set size but varying training set sizes. In practice, one should use all available data for testing, as it will only improve the significance. Figure 2 shows the estimated classification error and the corresponding p-values that were estimated using all of the examples left out in the training step for testing the classifier. And while the error graph looks very similar to that in Figure 1, the behavior of significance estimates is quite different. The p-values originally decrease as the training set size increases, but after a certain point, they start growing. Two conflicting factors control p-value estimates as the number of training examples increases: improved accuracy of the classification, which causes the point of interest to slide to the left – and as a result, down – on the empirical cdf curve, and the decreasing number of test examples, which causes the empirical cdf curve to become more shallow. In our experience, the minimum of the p-value often lies in the region of slow change in error values. In this

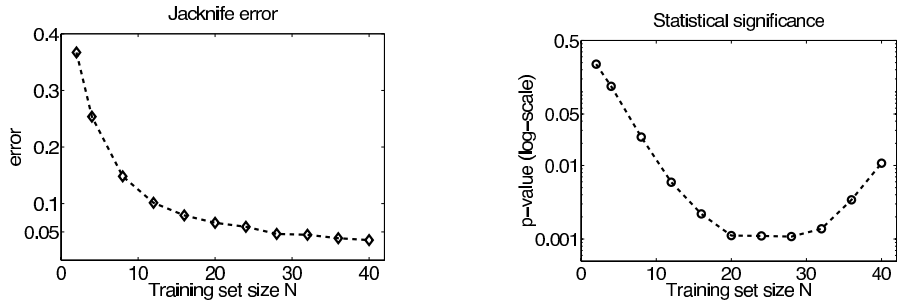


Figure 3: Estimated test error and statistical significance for different training set sizes $N$ for the leukemia morphology study. All of the examples unused in training were used to test the classifier. The p-values are shown on a logarithmic scale.

particular example, the minimum of p-value, $p = 0.02$, is achieved at $N = 50$. It is smaller than the numbers reported earlier because in this experiment because we use more test examples ($K = 50$) in each iteration of permutation testing.

Before proceeding to the summary reporting of the experimental results for other studies, we demonstrate the technique on one more detailed example. The objective of the underlying experiment was to accurately discriminate acute myeloid leukemia (AML) from acute lymphoblastic leukemia using DNA microarray expression data. The data set contains 48 samples of AML and 25 samples of ALL. Expression levels of $7,129$ genes and expressed sequence tags (ESTs) were measured via an oligonucleotide microarray for each sample (Golub et al., 1999, Slonim et al., 2000). Figure 3 shows the results for this study. Small number of available examples forces us to work with much smaller range of training and test set sizes, but, in contrast to the previous study, we achieve statistical significance with substantially fewer data points. The cross-validation error reduces rapidly as we increase the number of training examples, dropping below 5% at $N = 26$ training examples. The p-values also decrease very quickly as we increase the number of training examples, achieving minimum of .001 at $N = 28$ training examples. Similarly to the previous example, the most statistically significant result lies in the range of relatively slow error change.

As both examples demonstrate, it is not necessarily best to use most of the data for training. Additional data might not improve the testing accuracy in a substantial way and could be much more useful in obtaining more accurate estimates of the error and the p-value. In fact, a commonly used leave-one-out procedure that only uses one example to test in each cross-validation iteration procures extremely noisy estimates. In all the experiments reported in the next section, we use at least 10 examples to test in each iteration.

### 4.2 Experimental Results

We report experimental results for nine different studies involving classification of biological data derived from imaging of two different types, as well as microarray measurements. Table 1 summarizes the number of features, which determines the input space dimensionality, the number of examples in each class for each study and the results of statistical analysis. The studies are loosely sorted in the ascending strength of the results (decreasing error and p-values).

| Experiment | features | pos | neg | p-value < .05 | | | lowest p-value | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | $N/2$ | $e$ | $p$ | $N/2$ | $e$ | $p$ | $e_{\min}$ |
| Lymphoma outcome | 7,129 | 32 | 26 | | – | | 15 | .47 | .47 | .47 |
| Brain cancer outcome | 7,129 | 22 | 28 | | – | | 17 | .46 | .47 | .45 |
| Breast cancer outcome | 24,624 | 44 | 34 | | – | | 15 | .39 | .15 | .38 |
| MRI | 327,684 | 50 | 50 | 15 | .30 | .03 | 25 | .29 | .02 | .28 |
| Medullo *vs.* glioma | 7,129 | 45 | 15 | 6 | .12 | .04 | 6 | .12 | .04 | .11 |
| AML *vs.* ALL | 7,129 | 48 | 25 | 4 | .15 | .02 | 14 | .05 | .001 | .04 |
| fMRI full | 303,865 | 15 | 15 | 6 | .14 | .02 | 8 | .08 | .01 | .06 |
| fMRI reduced | 95,122 | 15 | 15 | 4 | .08 | .02 | 8 | .009 | .007 | .003 |
| Tumor *vs.* norm | 16.063 | 190 | 90 | 10 | .30 | .008 | 45 | .16 | $10^{-6}$ | .14 |

Table 1: Summary of the experimental data and results. The first four columns describe the study: the name of the experiment, the number of features and the number of examples in each class. Columns 5-7 report the number of training examples from each class $N/2$, the jackknife error $e$ and the p-value $p$ for the smallest training set size that achieves significance at $\alpha = .05$. Columns 8-10 report the results for the training set size that yields the smallest p-value. The last column contains the lowest error $e$ observed in the experiment.

**Imaging Data.** In addition to the MRI study of cortical thickness described above ("MRI" in Table 1), we include the results of two fMRI (functional MRI) experiments that compare the patterns of brain activations in response to different visual stimuli in a single subject. We present the results of comparing activations in response to face images to those induced by house images, as these categories are believed to have special representation in the cortex. The feature extraction step was similar to that of the MRI study, treating the activation signal as the measurement to be measured over the cortical surface. The first experiment ("fMRI full") used the entire cortical surface for feature extraction, while the second experiment ("fMRI reduced") considered only the visually active region of the cortex. The mask for the visually active voxels was obtained using a separate visual task. The goal of using the mask was to test if removing irrelevant voxels from consideration improves the classification performance.

**Microarray Data.** In addition to the leukemia morphology study ("AML *vs.* ALL"), we include the results of five other expression datasets where either the morphology or the treatment outcome was predicted (Mukherjee et al., 2003). Three studies involved predicting treatment outcome: survival of lymphoma patients, survival of patients with brain cancer and predicting metastisis of breast cancers. Three other studies involved predicting morphological properties of the tissue: medulloblastomas (medullo) *vs.* glioblastomas (glio)[1] AML *vs.* ALL and tumor tissue *vs.* normal.

For each experiment, we analyzed the behavior of the cross-validation error and the statistical significance similarly to the detailed examples presented earlier. Table 1 summarizes the results for three important events: the first time the p-value plot crosses .05 threshold (thus achieving statistical significance at that level), the point of lowest p-value,

1. Gliobastomas are tumors of glial cells in the brain while medulloblastomas are tumors of neural tissue.

and the lowest classification error. For the first two events, we report the number of training examples, the error and the p-value. The lowest error is typically achieved for the largest training set size and is shown here mainly for comparison with the other two error values reported. We observe that the error values corresponding to the lowest p-values are very close to the smallest errors reported, implying that the p-values bottom out in the region of a relatively slow change in the error estimates.

The first three studies in the table did not produce statistically significant results. In the first two studies, the rest of the indicators are extremely weak[2]. In the third study, predicting whether a breast tumor will metastisize, the error stabilizes fairly early (training on 30 examples leads to 39% error, while the smallest error observed is 38%, obtained by training on 58 examples), but the p-values are too high. This leads us to believe that more data could help establish the significance of the result, similarly to the MRI study. Unfortunately, the error in these two studies is too high to be useful in a diagnostic application. The rest of the studies achieve relatively low errors and p-values. The last study in the table predicting cancerous tissue from normal tissue yields a highly significant result ($p < 10^{-6}$), with the error staying very stable for training on 90 examples to training on 170 examples. We also observe that the significance threshold of .05 is probably too high for these experiments, as the corresponding error values are significantly higher than the ones reported for the smallest p-value. A more stringent threshold of .01 would cause most experiments to produce a more realistic estimates of the cross-validation error attainable on the given data set.

### 4.3 Summary And Heuristics

In this section, we show how permutation testing in conjunction with cross-validation can be used to analyze the quality of classification results on scientific data. Here, we provide a list of practical lessons learned from the empirical studies that we hope will be useful to readers applying this methodology.

1. *Interpreting the p-value.* Two factors affect statistical significance: the separation between the classes and the amount of data we have to support it. We can achieve low p-values in a situation where the two classes are very far apart and we have a few data points from each group, or when they are much closer, but we have substantially more data. P-value by itself does not indicate which of these two situations is true. However, looking at both the p-value and the classifier accuracy gives us an indication of how easy it is to separate the classes.

2. *Size of the holdout set.* In our experience, small test sizes in cross-validation and permutation testing leads to noisy estimates of the classification accuracy and the significance. We typically limit the training set size to allow at least 10 test examples in each iteration of the resampling procedures.

3. *Size of the training set.* Since we are interested in a robust estimates of the test error, one should utilize sufficient number of training examples to be working in the region where the cross-validation error does not vary much. Cross-checking this with the

2. In (Mukherjee et al., 2003), a gene selection procedure led to greater accuracy and smaller p-values.

region of lowest p-values provides another useful indication of the acceptable training set size. In our experiments with artificial data (not shown here), the number of training examples at which the p-values stop decreasing dramatically remains almost constant as we add more data. This could mean that acquiring more experimental data will not change the "optimal" training set size substantially, only lower the resulting p-values.

4. *Performance on future samples.* As we pointed out in earlier sections, the permutation test does not provide a guarantee on how close the observed classification error is to the true expected error. Thus we might achieve statistical significance, but still have an inaccurate estimate of the expected error. This brings us back to the main assumption made by most non-parametric procedures, namely that the available examples capture the relevant properties of the underlying probability distribution with adequate precision. The variance-based convergence bounds and the VC-style generalization bounds are still the only two ways known to us to obtain a distribution-free guarantee on the expected error of the classifier. The next section relates permutation testing to the generalization bounds and offers a theoretical justification for the procedure.

## 5. A Theoretical Motivation for the Permutation Procedure

The purpose of the permutation test is to show that it is very unlikely that a permuted dataset will achieve the same cross-validation error as the cross-validation error on the unpermuted dataset. In this paper, we restrict our proofs to leave-one-out cross-validation, with future plans to extend this work to the general case. Ideally, we would like to show that for a reasonably constrained family of classifiers (for example a VC class) the following two facts hold: the leave-one-out error of the classifier is close to the error rate of one of the best classifiers in the class and the leave-one-out error of the permuted dataset is close to the smaller of the prior probabilities of the two classes. In Section 5.1, we prove that the training error on the permuted data concentrates around the smaller prior probability. In Section 5.2, we combine previously known results to state that the leave-one-out error of the classifier is close to the error rate of one the best classifiers in the class. In Section 5.3, we extend a result from (Mukherjee et al., 2002) about the leave-one-out procedure to relate the training error of permuted data to the leave-one-out error. We close with some comments relating the theoretical results to the empirical permutation procedure.

### 5.1 The Permutation Problem

We are given a class of concepts $\mathcal{C}$ and an unknown target concept $c_0$. Without loss of generality we will assume that $P(c_0) \leq 1/2$ and $\emptyset \in \mathcal{C}$. For a permutation $\boldsymbol{\pi}$ of the training data, the smallest training error on the permuted set is

$$\begin{aligned} e_l(\boldsymbol{\pi}) &= \min_{c \in \mathcal{C}} P_l(c \triangle c_0) \qquad (1) \\ &= \min_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c, \mathbf{x}_i^{\pi} \notin c_0) + I(\mathbf{x}_i \notin c, \mathbf{x}_i^{\pi} \in c_0) \right], \end{aligned}$$

where $\mathbf{x}_i$ is the $i$-th sample and $\mathbf{x}_i^\pi$ is the $i$-th sample after permutation. For a fixed concept $c \in \mathcal{C}$ the average error is

$$\mathbb{E}P_l(c \triangle c_0) = P(c)(1 - P(c_0)) + (1 - P(c))P(c_0),$$

and it is clear that since $P(c_0) \leq 1/2$ taking $c = \emptyset$ will minimize the average error which in that case will be equal to $P(c_0)$. Thus, our goal will be to show that under some complexity assumptions on class $\mathcal{C}$ the smallest training error $e_l(\boldsymbol{\pi})$ is close to $P(c_0)$.

Minimizing (1) is equivalent to the following maximization problem

$$\max_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)(2I(\mathbf{x}_i^\pi \in c_0) - 1) \right],$$

since

$$e_l(\boldsymbol{\pi}) = P_l(\mathbf{x} \in c_0) - \max_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)(2I(\mathbf{x}_i^\pi \in c_0) - 1) \right],$$

and $P_l(\mathbf{x} \in c_0)$ is the empirical measure of the random concept. We would like to show that $e_l(\boldsymbol{\pi})$ is close to the random error $P(\mathbf{x} \in c_0)$ and give rates of convergence. We will do this by bounding the process

$$G_l(\boldsymbol{\pi}) = \sup_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)(2I(\mathbf{x}_i^\pi \in c_0) - 1) \right]$$

and using the fact that, by Chernoff's inequality, $P_l(\mathbf{x} \in c_0)$ is close to $P(\mathbf{x} \in c_0)$:

$$\mathbb{P}\left( P(\mathbf{x} \in c_0) - P_l(\mathbf{x} \in c_0) \leq \sqrt{\frac{2P(c_0)(1 - P(c_0))t}{l}} \right) \geq 1 - e^{-t}. \tag{2}$$

**Theorem 1** *If the concept class $\mathcal{C}$ has VC dimension $V$ then with probability $1 - Ke^{-t/K}$*

$$G_l(\boldsymbol{\pi}) \leq K \min\left( \sqrt{\frac{V \log l}{l}}, \frac{V \log l}{l(1 - 2P(c_0))^2} \right) + \sqrt{\frac{Kt}{l}}.$$

**Remark**
The second quantity in the above bound comes from the application of Chernoff's inequality similar to (2) and, thus, has a "one dimensional nature" in a sense that it doesn't depend on the complexity (VC dimension) of class $\mathcal{C}$. An interesting property of this result is that if $P(c_0) < 1/2$ then first term that depends on the VC dimension $V$ will be of order $V \log l / l$ which, ignoring the "one dimensional terms", gives the zero-error type rate of convergence of $e_l(\boldsymbol{\pi})$ to $P(\mathbf{x} \in c_0)$. Combining this theorem and equation (2) we can state that with probability $1 - Ke^{-t/K}$

$$P(\mathbf{x} \in c_0) \leq P_l(\mathbf{x} \in c_0) + K \min\left( \sqrt{\frac{V \log l}{l}}, \frac{V \log l}{l(1 - 2P(c_0))^2} \right) + \sqrt{\frac{Kt}{l}}.$$

In order to prove Theorem 1, we require several preliminary results. We first prove the following useful lemma.

**Lemma 1** *It is possible to construct on the same probability space two i.i.d Bernoulli sequences* $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)$ *and* $\varepsilon' = (\varepsilon'_1, \ldots, \varepsilon'_n)$ *such that* $\varepsilon$ *is independent of* $\varepsilon'_1 + \ldots + \varepsilon'_n$ *and* $\sum_{i=1}^n |\varepsilon_i - \varepsilon'_i| = |\sum_{i=1}^n \varepsilon_i - \sum_{i=1}^n \varepsilon'_i|$.

**Proof**

For $k = 0, \ldots, n$, let us consider the following probability space $\mathcal{E}_k$. Each element $w$ of $\mathcal{E}_k$ consists of two coordinates $w = (\varepsilon, \pi)$. The first coordinate $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)$ has the marginal distribution of an i.i.d. Bernoulli sequence. The second coordinate $\pi$ implements the following randomization. Given the first coordinate $\varepsilon$, consider a set $\mathcal{I}(\varepsilon) = \{i : \varepsilon_i = 1\}$ and denote its cardinality $m = \text{card}\{\mathcal{I}(\varepsilon)\}$. If $m \geq k$, then $\pi$ picks a subset $\mathcal{I}(\pi, \varepsilon)$ of $\mathcal{I}(\varepsilon)$ with cardinality $k$ uniformly, and if $m < k$, then $\pi$ picks a subset $\mathcal{I}(\pi, \varepsilon)$ of the complement $I^c(\varepsilon)$ with cardinality $n - k$ also uniformly. On this probability space $\mathcal{E}_k$, we construct a sequence $\varepsilon' = \varepsilon'(\varepsilon, \pi)$ in the following way. If $k \leq m = \text{card}\{\mathcal{I}(\varepsilon)\}$ then we set $\varepsilon'_i = 1$ if $i \in \mathcal{I}(\pi, \varepsilon)$ and $\varepsilon'_i = -1$ otherwise. If $k > m = \text{card}\{\mathcal{I}(\varepsilon)\}$ then we set $\varepsilon'_i = -1$ if $i \in \mathcal{I}(\pi, \varepsilon)$ and $\varepsilon'_i = 1$ otherwise. Next, we consider a space $\mathcal{E} = \cup_{k \leq n} \mathcal{E}_k$ with probability measure $\mathbb{P}(\mathcal{A}) = \sum_{k=0}^n B(n, p, k) \mathbb{P}(\mathcal{A} \cap \mathcal{E}_k)$, where $B(n, p, k) = \binom{n}{k} p^k (1-p)^{n-k}$. On this probability space the sequence $\varepsilon$ and $\varepsilon'$ will satisfy the conditions of the lemma. First of all, $X = \varepsilon'_1 + \ldots + \varepsilon'_n$ has binomial distribution since by construction $\mathbb{P}(X = k) = \mathbb{P}(\mathcal{E}_k) = B(n, p, k)$. Also, by construction, the distribution of $\varepsilon'$ is invariant under the permutation of coordinates. This, clearly, implies that $\varepsilon'$ is i.i.d. Bernoulli. Also, obviously, $\varepsilon$ is independent of $\varepsilon'_1 + \ldots + \varepsilon'_n$. Finally, by construction $\sum_{i=1}^n |\varepsilon_i - \varepsilon'_i| = |\sum_{i=1}^n \varepsilon_i - \sum_{i=1}^n \varepsilon'_i|$.

■

**Definition 1** *Let* $u > 0$ *and let* $\mathcal{C}$ *be a set of concepts. Every finite set of concepts* $c_1, ..., c_n$ *with the property that for all* $c \in \mathcal{C}$ *there is a* $c_j$ *such that*

$$\frac{1}{l} \sum_{i=1}^{l} |c_j(x_i) - c(x_i)|^2 \leq u$$

*is called a* $u$*-cover with respect to* $\|\cdot\|_{L_2(\mathbf{x}_l)}$. *The covering number* $\mathcal{N}(\mathcal{C}, u, \{\mathbf{x}_1, ..\mathbf{x}_l\})$ *is the smallest number for which the above holds.*

**Definition 2** *The uniform metric entropy is* $\log \mathcal{N}(\mathcal{C}, u)$ *where* $\mathcal{N}(\mathcal{C}, u)$ *is the smallest integer for which*

$$\forall l, \quad \forall (\mathbf{x}_1, ..., \mathbf{x}_l), \quad \mathcal{N}(\mathcal{C}, u, \{\mathbf{x}_1, ..\mathbf{x}_l\}) \leq \mathcal{N}(\mathcal{C}, u).$$

**Theorem 2** *The following holds with probability greater than* $1 - 4e^{-t/4}$

$$G_l(\boldsymbol{\pi}) \leq \sup_r \left[ K \frac{1}{\sqrt{l}} \int_0^{\sqrt{\mu_r}} \sqrt{\log \mathcal{N}(u, \mathcal{C})} du - \frac{\mu_r}{2} (1 - 2P(c_0)) + \sqrt{\frac{\mu_r (t + 2\log(r+1))}{l}} \right] + 2\sqrt{\frac{2tP(c_0)(1 - P(c_0))}{l}},$$

*where* $\mu_r = 2^{-r}$ *and* $\log \mathcal{N}(\mathcal{C}, u)$ *is the uniform metric entropy for the class* $\mathcal{C}$.

**Proof**

The process

$$G_l(\boldsymbol{\pi}) = \sup_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)(2I(\mathbf{x}_i^{\pi} \in c_0) - 1) \right].$$

can be rewritten as

$$G_l(\boldsymbol{\pi}) = \sup_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)\varepsilon_i \right],$$

where $\varepsilon_i = 2I(\mathbf{x}_i^{\pi} \in c_0) - 1 = \pm 1$ are Bernoulli random variables with $P(\varepsilon_i = 1) = P(c_0)$. Due to permutations the random variables $(\varepsilon_i)$ depend on $(\mathbf{x}_i)$ only through the cardinality of $\{\mathbf{x}_i \in c_0\}$. By lemma 1 we can construct a random Bernoulli sequence $(\varepsilon_i')$ that is independent of $\mathbf{x}$ and for which

$$G_l(\boldsymbol{\pi}) \leq \sup_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)\varepsilon_i' \right] + \left| \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i - \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i' \right|.$$

We first control the second term

$$\left| \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i - \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i' \right| \leq \left| \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i' - (2P(c_0) - 1) \right| + \left| \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i - (2P(c_0) - 1) \right|,$$

then using Chernoff's inequality twice we get with probability $1 - 2e^{-t}$

$$\left| \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i - \frac{1}{l} \sum_{i=1}^{l} \varepsilon_i' \right| \leq 2\sqrt{\frac{2tP(c_0)(1 - P(c_0))}{l}}.$$

We block concepts in $\mathcal{C}$ into levels

$$\mathcal{C}_r = \left\{ c \in \mathcal{C} : \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \in (2^{-r-1}, 2^{-r}] \right\}$$

and denote $\mu_r = 2^{-r}$. We define the processes

$$R(r) = \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)\varepsilon_i' \right],$$

and obtain

$$\sup_{c \in \mathcal{C}} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)\varepsilon_i' \right] \leq \sup_{r} R(r).$$

By Talagrand's inequality on the cube (Talagrand, 1995), we have for each level $r$

$$\mathbb{P}_{\varepsilon'} \left( R(r) \leq \mathbb{E}_{\varepsilon'} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)\varepsilon_i' \right] + \sqrt{\frac{\mu_r t}{l}} \right) \geq 1 - e^{-t/4}.$$

Note that for this inequality to hold, the random variables $(\varepsilon')$ need only be independent, they do not need to be symmetric. This bound is conditioned on a given $\{\mathbf{x}_i\}_{i=1}^l$ and therefore,

$$\mathbb{P}\left( R(r) \leq \mathbb{E}_{\varepsilon'} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \varepsilon_i' \right] + \sqrt{\frac{\mu_r t}{l}} \right) \geq 1 - e^{-t/4}.$$

If, for each $r$, we set $t \to t + 2\log(r+1)$, we can write

$$\begin{aligned}
&\mathbb{P}\left( \forall r \;\; R(r) \leq \mathbb{E}_{\varepsilon'} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \varepsilon_i' \right] + \sqrt{\frac{\mu_r (t + 2\log(r+1))}{l}} \right) \\
&\geq 1 - \sum_{r=0}^{\infty} \frac{1}{(r+1)^2} e^{-t/4} \geq 1 - 2e^{-t/4}.
\end{aligned}$$

Using standard symmetrization techniques we add and subtract an independent sequence $\varepsilon_i''$ such that $\mathbb{E}\varepsilon_i'' = \mathbb{E}\varepsilon_i' = (2P(c_0) - 1)$:

$$\begin{aligned}
&\mathbb{E}_{\varepsilon'} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \varepsilon_i' \right] \\
&\leq \mathbb{E}_{\varepsilon'} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \varepsilon_i' - \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \mathbb{E}\varepsilon_i'' + \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)(2P(c_0) - 1) \right] \\
&\leq \mathbb{E}_{\varepsilon' \varepsilon''} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c)(\varepsilon' - \varepsilon'') \right] - (1 - 2P(c_0)) \inf_{c \in \mathcal{C}_r} \left( \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \right) \\
&\leq 2\mathbb{E}_{\eta_i} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \eta_i \right] - \frac{\mu_r (1 - 2P(c_0))}{2},
\end{aligned}$$

where $\eta_i = (\varepsilon_i' - \varepsilon_i'')/2$ takes values $\{-1, 0, 1\}$ with probability $P(\eta_i = 1) = P(\eta_i = -1)$. One can easily check that the random variables $\eta_i$ satisfy the inequality

$$\mathbb{P}\left( \sum_{i=1}^{l} \eta_i a_i > t \right) \leq e^{-\frac{t^2}{2\sum_{i=1}^{l} a_i^2}}$$

which is the only prerequisite for the chaining method. Thus, one can write Dudley's entropy integral bound (van der Vaart and Wellner, 1996) as

$$\mathbb{E}_{\eta_i} \sup_{c \in \mathcal{C}_r} \left[ \frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in c) \eta_i \right] \leq K \frac{1}{\sqrt{l}} \int_0^{\sqrt{\mu_r}} \sqrt{\log \mathcal{N}(u, \mathcal{C})} du.$$

We finally get

$$\begin{aligned}
&\mathbb{P}\left( \forall r \;\; R(r) \leq K \frac{1}{\sqrt{l}} \int_0^{\sqrt{\mu_r}} \sqrt{\log \mathcal{N}(u, \mathcal{C})} du + \sqrt{\frac{\mu_r (t + 2\log(r+1))}{l}} - \frac{\mu_r (1 - 2P(c_0))}{2} \right) \\
&\geq 1 - 2e^{-t/4}.
\end{aligned}$$

This completes the proof of Theorem 2.
■

**Proof of Theorem 1**
For a class with VC dimension $V$, it is well known that (van der Vaart and Wellner, 1996)

$$\frac{1}{\sqrt{l}} \int_0^{\sqrt{\mu_r}} \sqrt{\log \mathcal{N}(u, \mathcal{C})} du \leq K \sqrt{\frac{V \mu_r \log \frac{2}{\mu_r}}{l}}.$$

Since without loss of generality we only need to consider $\mu_r > 1/l$, it remains to apply Theorem 2 and notice that

$$\sup_r \left[ K \sqrt{\frac{V \mu_r \log l}{l}} - \frac{\mu_r}{2} (1 - 2P(c_0)) \right] \leq K \min \left( \sqrt{\frac{V \log l}{l}}, \frac{V \log l}{l(1 - 2P(c_0))^2} \right).$$

All other terms that do not depend on the VC dimension $V$ can be combined to give $\sqrt{Kt/l}$.
■

### 5.2 Convergence of the Leave-one-out Error to the Best in the Class

Now we turn our attention to the cross-validation error and show that the leave-one-out error of the classifier is close to the error rate of one the best classifiers in the class.

Given a training set generated by an unknown concept $c_0$ and a class of concepts $\mathcal{C}$, the training step selects concept $\hat{c}$ from this class via empirical risk minimization

$$\hat{c} \in \arg\min_{c \in \mathcal{C}} P_l(c \triangle c_0),$$

where $P_l(c \triangle c_0)$ is the empirical symmetric difference observed on the $l$ points in the dataset $S = \{(\mathbf{x}_i, y_i)\}_{i=1}^l$. This error is called the training error. The best classifiers in the class are ones for which

$$c_{opt} \in \arg\min_{c \in \mathcal{C}} P(c \triangle c_0).$$

It is well known that if $\mathcal{C}$ has VC dimension $V$ then with probability $1 - e^{-t}$ (Vapnik, 1998)

$$P(\hat{c} \triangle c_0) - P(c_{opt} \triangle c_0) \leq K \sqrt{\frac{V \log l}{l}}, \tag{3}$$

so the expected error of the empirical minimizer approaches the error rate of optimal classifiers in the class.

We now relate the the leave-one-out error to the expected error of the empirical minimizer. The classifier $\hat{c}_{S^i}$ is the empirical minimizer of the dataset with the $i$-th point left out and we write the the leave-one-out error as

$$\frac{1}{l} \sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i}).$$

Theorem 4.2 in (Kearns and Ron, 1999) states that for empirical risk minimization on a VC class with probability $1-\delta$

$$\left|\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i}) - P(\hat{c}\triangle c_0)\right| \leq K\sqrt{\frac{V\log l}{l}}\Big/\delta.$$

The above bound coupled with the bound in equation (3) ensures that with high probability the leave-one-out error approaches the error rate of one of the best classifiers in the class.

### 5.3 Relating the Training Error to the Leave-one-out Error

In the two previous subsections we demonstrated that the leave-one-out error approaches the error rate of one of the best classifiers in the class and that the training error on the permuted data concentrates around the smaller prior probability. However, in our statistical test we use the cross-validation error of the permuted data rather than the training error of the permuted data to build the empirical distribution function. So we have to relate the leave-one-out procedure to the training error for permuted data.

In the case of empirical minimization on VC classes, the training error can be related to the leave-one-out error by the following bound (Kearns and Ron, 1999, Mukherjee et al., 2002) (see also appendix A)

$$\mathbb{E}_S\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i}) - P(\hat{c}\triangle\tilde{c}_0)\right] \leq \Theta\left(\sqrt{\frac{V\log l}{l}}\right),$$

where $\mathbb{E}_S$ is the expectation over datasets $S$, $V$ is the VC dimension, and $\hat{c}_{S^i}$ is the empirical minimizer of the dataset with the $i$-th point left out. This bound is of the same order as the deviation between the empirical and expected errors of the empirical minimizer

$$\mathbb{E}_S\left[P_l(\hat{c}\triangle\tilde{c}_0) - P(\hat{c}\triangle\tilde{c}_0)\right] \leq \Theta\left(\sqrt{\frac{V\log l}{l}}\right).$$

This inequality implies that, on average, the leave-one-out error is not a significantly better estimate of the test error than the training error.

In the case of the permutation procedure we show that a similar result holds.

**Theorem 3** *If the family of classifiers has VC-dimension $V$ then*

$$\mathbb{E}_{S,\pi}\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i^{\pi} \in \hat{c}_{S^i,\pi}) - P(\mathbf{x} \in c_0)\right] \leq K\sqrt{\frac{V\log l}{l}},$$

*where $\mathbb{E}_{S,\pi}$ is the expectation over datasets $S$ and permutations $\boldsymbol{\pi}$, $\hat{c}_{S^i,\pi}$ is the empirical minimizer of the permuted dataset with the $i$-th point left out, $\mathbf{x}_i^{\pi}$ is the $i$-th point after the permutation.*

The proof of this theorem is in appendix A. From Section 5.1 we have that

$$\mathbb{E}_{S,\boldsymbol{\pi}}\left[P_l(\hat{c}\triangle c_0) - P(\mathbf{x} \in c_0)\right] \leq K\sqrt{\frac{V\log l}{l}}.$$

Therefore, one can conclude that for the permutation procedure on average the leave-one-out error is not a significantly better estimate of the random error than the training error.

### 5.4 Comments on the Theoretical Results

The theoretical results in this section were to give an analysis and motivation for the permutation tests. The bounds derived are not meant to replace the empirical permutation procedure. Similarly to VC-style generalization bounds, these would require amounts of data far beyond the range of samples realistically attainable in the real experiments to obtain practically useful deviations (of the order of a few percent), which is precisely the motivation for the empirical permutation procedure.

Moreover, we state that leave-one-out procedure is not a significantly better estimate of error that the training error. This statement must be taken with a grain of salt since it was derived via a worst case analysis. In practice, the leave-one-out estimator is almost always a better estimate of the test error than the training error and, for the empirical risk minimization, the leave-one-out error is never smaller than the training error: (Mukherjee et al., 2002)

$$\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i}) \geq \frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_S).$$

Also, it is important to note that the relation we state between the leave-one-out error and training error is in expectation. Ideally, we would like to relate these quantities in probability with a similar rates. It is an open question whether this holds.

## 6. Conclusion And Open Problems

This paper describes and explores an approach to estimating statistical significance of a classifier given a small sample size based on permutation testing. The following is the list of open problems related to this methodology:

1. *Size of the training/test set.* We provide a heuristic to select the size of the training and the holdout sets. A more rigorous formulation of this problem might suggest a more principled methodology for setting the training set size. This problem is clearly an example of the ubiquitous bias-variance tradeoff dilemma.

2. *Leave-one-out error and training error.* In the theoretical motivation, we relate the leave-one-out error to the training error in expectation. The theoretical motivation would be much stronger if this relation was made in probability. A careful experimental and theoretical analysis of the relation between the training error and leave-n-out error for these types permutation procedures would be of interest.

3. *Feature selection.* Both in neuroimaging studies and in DNA microarray analysis, finding the features which most accurately classify the data is very important. Permutation procedures similar to the one described in this paper have been used to address this problem (Golub et al., 1999, Slonim et al., 2000, Nichols and Holmes, 2001). The analysis of permutation procedures for selecting discriminative features seems to be more difficult than the analysis of the permutation procedure for classification. It would be very interesting to extend the type of analysis here to the feature selection problem.

4. *Multi-class classification.* Extending the methodology and theoretical motivation for the multi-class problem has not been done.

To conclude, we hope other researchers in the community will find the technique useful in assessing statistical significance of observed results when the data are high dimensional and are not necessarily generated by a known distribution.

## Appendix A. Proof of Theorem 3

In this appendix, we prove Theorem 3 from section 5.3. We first define some terms. A concept will be designated $c_0$. A dataset $S$ is made up of $l$ points $\{\mathbf{z}_i\}_{i=1}^{l}$ where $\mathbf{z} = (\mathbf{x}, y)$. When the i-th point is left out of $S$ we have the set $S^i$. The empirical minimizer on $S$ is $\hat{c}_S$ and the empirical minimizer on $S^i$ is $\hat{c}_{S^i}$. The empirical error on a set $S$ is

$$P_l(\hat{c}_S \triangle c_0),$$

and the empirical error on a set $S^i$ is

$$P_{l-1}(\hat{c}_{S^i} \triangle c_0).$$

If we perform empirical risk minimization on a VC class, then with high probability $(1 - e^{-t/K})$

$$|P(\hat{c}_S \triangle c_0) - P_l(\hat{c}_S \triangle c_0)| \leq K\sqrt{\frac{V \log l}{l}}. \tag{4}$$

Similarly, with high probability

$$|P(\hat{c}_{S^i} \triangle c_0) - P_{l-1}(\hat{c}_{S^i} \triangle c_0)| \leq K\sqrt{\frac{V \log(l-1)}{(l-1)}}$$

and

$$|P_{l-1}(\hat{c}_{S^i} \triangle c_0) - P_l(\hat{c}_{S^i} \triangle c_0)| \leq \frac{K}{l}.$$

We turn the probability into an expectation

$$\mathbb{E}_S\left[P(\hat{c}_{S^i} \triangle c_0) - P_l(\hat{c}_S \triangle c_0)\right] \leq K\sqrt{\frac{V \log l}{l}}. \tag{5}$$

One can check that the expectation over $S$ of the leave-one-out error is equal to the expectation over $S$ of the expected error of $\hat{c}_{S^i}$ (Mukherjee et al., 2002, Bousquet and Elisseeff, 2002):

$$\mathbb{E}_S\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i})\right] = \mathbb{E}_S P(\hat{c}_{S^i} \triangle c_0).$$

Combining the above equality with inequalities (4) and (5) gives us the following bounds

$$\mathbb{E}_S\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i}) - P_l(\hat{c}_S \triangle c_0)\right] \leq K\sqrt{\frac{V \log l}{l}},$$

$$\mathbb{E}_S\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i \in \hat{c}_{S^i}) - P(\hat{c}_S \triangle c_0)\right] \leq K\sqrt{\frac{V \log l}{l}}.$$

So for the nonpermuted data, the leave-one-out error is not significantly closer to the test error in expectation than the training error.
We want to show something like the above for the case where the concept $c_0$ is random and the empirical minimizer is constructed on a dataset with labels randomly permuted. Let's denote $\hat{c}_{S^i,\pi}$ the empirical minimizer of the permuted dataset with the $i$th point left out. It's leave-one-out error is

$$\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i^\pi \in \hat{c}_{S^i,\pi}).$$

If we can show that

$$\mathbb{E}_{S,\pi}\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i^\pi \in \hat{c}_{S^i,\pi})\right] = \mathbb{E}_{S,\pi} P(\hat{c}_{S^i,\pi} \triangle c_0),$$

then we can use the same argument we used for the nonrandom case.
We start by breaking up the expectations

$$\begin{aligned}\mathbb{E}_{S,\pi}\left[\frac{1}{l}\sum_{i=1}^{l} I(\mathbf{x}_i^\pi \in \hat{c}_{S^i,\pi})\right] &= \mathbb{E}_\pi \mathbb{E}_S \mathbb{E}_{\mathbf{z}_i} I(\mathbf{x}_i^\pi \in \hat{c}_{S^i,\pi}) \\ &= \mathbb{E}_\pi \mathbb{E}_S \mathbb{E}_{\mathbf{x}_i} \mathbb{E}_{y_i} I(\hat{c}_{S^i,\pi}(\mathbf{x}_i^\pi) = y_i),\end{aligned}$$

The second line holds because for a random concept $p(\mathbf{x}, y) = p(y)p(\mathbf{x})$. One can easily check the following hold

$$\mathbb{E}_{y_i} I(\hat{c}_{S^i,\pi}(\mathbf{x}_i^\pi) = y_i) = \min(P(y=-1), P(y=1)) = P(\mathbf{x} \in c_0),$$

and

$$\mathbb{E}_\pi \mathbb{E}_S \mathbb{E}_{\mathbf{x}_i} \min(P(y=-1), P(y=1)) = P(\mathbf{x} \in c_0).$$

Similarly it holds that

$$\mathbb{E}_{S,\pi} P(\hat{c}_{S^i,\pi} \triangle c_0) = \min(P(y=-1), P(y=1)) = P(\mathbf{x} \in c_0).$$

■

## References


O. Bousquet and A. Elisseeff. Stability and generalization. *Journal Machine Learning Research*, 2:499–526, 2002.

B. Efron. *The Jackknife, The Bootstrap, and Other Resampling Plans*. SIAM, Philadelphia, PA, 1982.

B. Fischl and A.M. Dale. Measuring the thickness of the human cerebral cortex from magnetic resonance images. *PNAS*, 26:11050–11055, 2000.

B. Fischl, M.I. Sereno, R.B.H. Tootell, and A.M. Dale. High-resolution intersubject averaging and a coordinate system for the cortical surface. *Human Brain Mapping*, 8:262–284, 1999.

T.R. Golub, D. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, H. Coller, M.L. Loh, J.R. Downing, M.A. Caligiuri, C. D. Bloomfield, and E. S. Lander. Molecular Classification of Cancer: Class discovery and class prediction by gene expression monitoring. *Science*, 286:531–537, 1999.

P. Good. *Permutation Tests: A Practical Guide to Resampling Methods for Testing Hypothesis*. Springer-Verlag, 1994.

I. Guyon, J. Makhoul, R. Schwartz, and V. Vapnik. What size test set gives good error estimates? *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:52–64, 1998.

M. Kearns and D. Ron. Algorithmic stability and sanity-check bounds for leave-one-out cross-validation. *Neural Computation*, 11:1427–1453, 1999.

M.G. Kendall. The treatment of ties in ranking problems. *Biometrika*, 33:239–251, 1945.

S. Mukherjee, P. Niyogi, T. Poggio, and R. Rifkin. Statistical learning: Stability is necessary and sufficient for consistency of empirical risk minimization. AI Memo 2002-023, Massachusetts Institute of Technology, 2002.

S. Mukherjee, P. Tamayo, S. Rogers, R. Rifkin, A. Engle, C. Campbell, T.R. Golub, and J.P. Mesirov. Estimating dataset size requirements for classifying dna microarray data. *Journal Computational Biology*, 10(2):119–142, 2003.

T.E. Nichols and A.P. Holmes. Nonparametric permutation tests for functional neuroimaging: A primer with examples. *Human Brain Mapping*, 15:1–25, 2001.

L. Sachs. *Applied Statistics: A Handbook of Techniques*. Springer Verlag, 1984.

D. Slonim, P. Tamayo, J.P. Mesirov, T.R. Golub, and E. Lander. Class prediction and discovery using gene expression data. In *Proceedings of the Fourth Annual Conference on Computational Molecular Biology (RECOMB)*, pages 263–272, 2000.

Student. The probable error of a mean. *Biometrika*, 6:1–25, 1908.

M. Talagrand. Concentration of measure and isoperimetric inequalities in product spaces. *Publications Mathématiques de l'I.H.E.S.*, 81:73–205, 1995.

A. van der Vaart and J. Wellner. *Weak convergence and Empirical Processes With Applications to Statistics*. Springer-Verlag, 1996.

V.N. Vapnik. *Statistical Learning Theory*. John Wiley & Sons, 1998.