CENTER FOR
Brains
Minds+
Machines

CBMM Memo No. 63 May 26, 2017

Symmetry Regularization

Fabio Anselmi[1,2*], **Georgios Evangelopoulos**[1*], **Lorenzo Rosasco**[1,2] **Tomaso Poggio**[1,2]

(* denotes equal contribution)

1: Center for Brains, Minds, and Machines | McGovern Institute for Brain Research at MIT, Cambridge, MA, USA
2: Laboratory for Computational and Statistical learning (LCSL)-Istituto Italiano di Tecnologia, Genova, Italy

Abstract

The properties of a representation, such as smoothness, adaptability, generality, equivariance/invariance, depend on restrictions imposed during learning. In this paper, we propose using data symmetries, in the sense of equivalences under transformations, as a means for learning symmetry-adapted representations, i.e., representations that are equivariant to transformations in the original space. We provide a sufficient condition to enforce the representation, for example the weights of a neural network layer or the atoms of a dictionary, to have a group structure and specifically the group structure in an unlabeled training set. By reducing the analysis of generic group symmetries to permutation symmetries, we devise an analytic expression for a regularization scheme and a permutation invariant metric on the representation space. Our work provides a proof of concept on why and how to learn equivariant representations, without explicit knowledge of the underlying symmetries in the data.

This material is based upon work supported by the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216.

Symmetry Regularization

Fabio Anselmi[1,2*] **Georgios Evangelopoulos**[1*], **Lorenzo Rosasco**[1,2], **Tomaso Poggio**[1,2]
1: Center for Brains, Minds, and Machines — McGovern Institute for Brain Research at MIT, Cambridge, MA, USA 2: Laboratory for Computational and Statistical learning (LCSL)-Istituto Italiano di Tecnologia, Genova, Italy (equal contribution)*

Abstract

The properties of a representation, such as smoothness, adaptability, generality, equivariance/invariance, depend on restrictions imposed during learning. In this paper, we propose using data symmetries, in the sense of equivalences under transformations, as a means for learning symmetry-adapted representations, i.e., representations that are equivariant to transformations in the original space. We provide a sufficient condition to enforce the representation, for example the weights of a neural network layer or the atoms of a dictionary, to have a group structure and specifically the group structure in an unlabeled training set. By reducing the analysis of generic group symmetries to permutation symmetries, we devise an analytic expression for a regularization scheme and a permutation invariant metric on the representation space. Our work provides a proof of concept on why and how to learn equivariant representations, without explicit knowledge of the underlying symmetries in the data.

1. Introduction

Symmetry is ubiquitous from subatomic particles to natural patterns, human design and artistic production. The core idea of this work is that data symmetries can be learned and used in the context of machine learning to derive data representations [5] that are equivariant and invariant [46, 3, 30] to unknown, symmetry-generating transformations. Such properties are important for learning efficient and statistically relevant weight sharing schemes and reducing the complexity of supervised learning, with respect to transformations that preserve the data distribution and prediction function [46, 18].

Data representation is a classic concept in harmonic analysis and signal processing where representations are typically designed on the basis of prior, available information. In machine learning, methods for learning the data representation in a supervised or unsupervised way, are conceptually an attempt to learn such priors or features, directly from the data. Examples of unsupervised representation learning methods include dictionary learning [54], autoencoders [4] and metric learning techniques [58]. In the supervised learning regime, the representation is learned jointly with the classifier, for example, by parameterizing functions form a hypothesis space through the lower layers of deep neural networks [28].

A fundamental structure in the input space is imposed by the *symmetries of data points* with respect to transformations that, for a task of learning from labeled instances, are irrelevant or even *nuisance factors* [46]. In visual object recognition, for example, the identity of the object is preserved under geometric transformations. In speech recognition, the phonetic or lexical category is preserved through different speakers or speaking conditions.

Symmetries are relevant for representation learning from the perspective of a) learning symmetries and b) employing them for symmetry-adapted representations for transformation equivariance or invariance. In this paper we focus on the former, building on existing theoretical frameworks for constructing equivariant and invariant representations.

Transformation symmetries define equivalence classes that capture part of the intraclass variability of the categorization task. They also imply a quotient space where points are equivalent up to transformations and a representation that is invariant, in the sense that it preserves the output (labeling) distribution. Restriction of the hypothesis space to this quotient representation space is crucial for learning from high-dimensional data [2, 6, 46, 31]. by reducing the sample complexity of learning (the size of the labeled training set) [1, 3, 16].

Representations that are symmetry-adapted can also be seen as an effort to a) generalize Convolutional Neural Networks (CNNs) [28] to weight sharing schemes and invariances beyond translations by explicitly *learning the symmetry or convolution group* from data and b) derive algorithms for learning optimal network architectures and feature map properties, such as sparsity, locality and weight-sharing topologies, instead of designing them.

Deep CNNs have an explicit parameterization for translation equivariance and robustness using convolutions and pooling. These computations, sharing principles from neuroscience [2, 37], involve filtering with localized kernels, weight re-sharing and pooling over local regions. However, data symmetries will extend to other transformations that can be described, in the simplest scenario, by geometric changes such as scaling, rotation or affine maps. A representation that does not account for more generic, arbitrary and unknown symmetries, will have to *compensate using a larger number of parameters*, which in turn will result in an increased size of the labeled training set for learning. This is in fact the case for many of the impressive results of deep learning, for example in vision, speech and natural language, which rely on massive amounts of labeled training data [28], for training convolutional models with a number of parameters of the same order [25, 44, 51], data augmentation and adaptation to known transformation models [20, 24].

The contributions of this work are the following: a) We outline the principles of a family of methods for learning the symmetries in the data, and learning equivariant representations, without explicit knowledge of the symmetry group. As opposed to *learning with known symmetries*, like many of the existing methods, we propose to *learn the symmetries*; b) We reduce the analysis of generic group symmetries to permutation symmetries; c) We devise an analytic expression for a regularization term acting on the matrix of weights (dictionary or neural network parametrization) that enforces a group structure and specifically the group structure in an unlabeled training set. d) We derive a permutation invariant metric on the representation space for evaluating permutation relationships and symmetry-adapted representations under our framework.

The paper is organized as follows. We summarize related work in Sec. 2 and provide background on on equivariant representations and group orbits in Sec. 3. Then in Sec. 4 we state the problem of symmetry-adapted representation learning and provide a general principle in Sec. 5 for designing orbit regularization. The main theoretical contributions of the paper are stated in Sec. 6 along with algorithmic implementations, through the concept of permutation maximal invariants (6.1), and a computable, analytic form for an orbit regularization term and a term sensitive to exact data symmetries (6.2). A summary of the results and their use for learning and evaluating symmetry-adapted representations are provided in Sec. 7. As a proof of concept, we formulate an unsupervised dictionary learning problem in Sec. 8 and provide results on learning from data with exact, analytic, group-transformations and show that minima can correspond to dictionaries that result in equivariant maps. We conclude in Sec. 9 by discussing open theoretical and algorithmic challenges. Most of the mathematical derivations can be found in the Appendix.

2. Related work

Understanding and using data symmetries has In an early work, using symmetries for learning was framed as categorizing symmetry groups (mirror, roto-translation) in random patterns [41]. In the direction of learning symmetries, various methods have been proposed for learning infinitesimal generators of Lie groups [38, 35, 48, 57, 9]. Encoding symmetries has also been implicitly approached as learning transformations between images [33]. Symmetries in dictionary design and learning have been almost exclusively considered through shifts of localized atoms [22], equivalent to imposing banded and circular (Toeplitz) dictionary structures. Convolutional or shift-invariant sparse coding schemes [17] were proposed for translation-invariant representations, trained on larger input supports.

In unsupervised representation learning, convolutional deep belief networks [29] and convolutional autoencoder [32] variants were proposed to enforce shift-invariance in the feature maps. Equivariant Boltzmann machines were explored for global (rotation) [24] and local (rotation, translation scale) [49] filter transformations, using *given transformation matrices*. Specifically, in [49] linear transformations of the weights are incorporated in learning and encoding to achieve invariance, through explicit max pooling over the set. Extensions to autoencoder and sparse coding objectives were also discussed. Transforming autoencoder [19] learn features with parameters, such as pose or position, that are equivariant w.r.t. to global input transformations. The result is a composition of *capsules* that transform parts, trained with *transformation supervision*.

Invariance to transformations has been used as a key property in representation learning, for example: studying the properties of convolutional networks [31, 7]; deriving analytical expressions based on minimal sufficient statistics [47]; formalizing hierarchical, compositional maps for learning with small complexity [3]; forming similarity-based loss functions [52].

Extensions of invariance in CNNs beyond translations were explored with scale-space pooling [42], convolutional maxout networks for pooling over translations and feature channels [55], pooling over neighboring values and similar filters [23] and tiled CNNs [27]. Also wavelet *scattering networks* compute a translation invariant, deformation stable representation [6], extended to scale and rotation in[43]. Spatial transformer networks [20] are modules that manipulate the data, instead of pooling for invariance, by applying a transformation, whose parameters are conditioned on the input. The modules are purely discriminatively trained, requiring however an explicit, parametric transformation model.

Symmetry networks [16] are discriminatively trained architectures that employ feature maps generated from groups other than translations. However the symmetries, i.e., the transformation group, is *known and parameterized* in order to define symmetry spaces, neighborhoods, pooling, with parameters efficiently sampled for tractable representations. Single and double-layer affine symnets were used to demonstrate the decrease in sample complexity w.r.t. standard (translational) convolution layers for image classification.

Group-convolutional networks [60] proposed generalized group convolutions and invariance to transformations [3] that model perceptual changes such as speaker variability in speech. The representation relies on pooling jointly over translations and transformations for sets of transformed inputs; for unitary transformations this is equivalent to pooling over transformations of the weights. Analogously, additional operations to the inputs were proposed in [12], for partial equivariance and invariance to a small group of four rotations, isomorphic to cyclic group C_4. *Group equivariant* networks [10, 11] in turn, employ explicit filter transformations using small groups (four rotations, reflection), for equivariance through group convolutions. The equivariance properties of tied-weight neural network layers for a *discrete, known group* are theoretically treated also in [39].

The relations between group-based regularization, i.e. constraints on orbit equivalence relationships, for known, simple groups and well known regularization schemes (e.g., l_1, l_2, l_∞, nuclear, spectral) have been explored in [36]. In [40] the notion of maximal invariants has been employed in analytic functions for shape classification up to permutation transformations.

3. Background: Data representations with symmetries

3.1 Representation learning

Definition 1. *A representation learning algorithm $\mathcal{A}$ is a map from a training set S_N to a finite set of weights or dictionary elements (atoms)*

$$W = \{w_i\}_{i=1}^m = \mathcal{A}(S_N), \quad S_N \subset \mathcal{X}, w_i \in \mathcal{X}, \quad N \in \mathbb{N}. \tag{1}$$

Definition 2. *A representation $\Phi(x) \in \mathcal{F}$ of $x \in \mathcal{X}$, given W is given by the set of measurements or coefficients $\phi_i(x) = \{\sigma(\langle w_i, x\rangle)\}_{i=1}^m$, where $\sigma : \mathbb{R} \to \mathbb{R}$ a linear or non-linear function, for example $\Phi : \mathcal{X} \to \mathcal{F}$, with $\Phi(x) = (\phi_1(x), \dots, \phi_m(x))$ and $\mathcal{X} = \mathbb{R}^d, \mathcal{F} = \mathbb{R}^m$.*

A way to mathematically define the question of learning W with specific structure is to pose learning as a variational problem. Suppose that by the learning algorithm, we are trying to minimize some objective function $\mathcal{L} : \mathcal{X} \to \mathbb{R}$ that specifies the representation criterion, for example reconstruction or classification accuracy. We can include a priori information on the target by adding a regularization term to restrict the set of possible solutions to those that have a particular structure, i.e to choose a hypothesis space. More general, we can formulate the learning problem as that of minimizing a sum of two functions:

$$\min_W \left(\mathcal{L}(W, S_N) + \beta \mathcal{J}(W)\right), \quad \beta \in \mathbb{R}_+ \tag{2}$$

where $\mathcal{L}$ is a data dependent loss function that assigns, for example, an empirical classification or reconstruction error using W and $\mathcal{J} : \mathbb{R}^{d\times m} \to \mathbb{R}$ is a regularization term, controlled by β, incorporating expectations, assumptions or prior knowledge, e.g. imposing that the solutions are in the class of smooth functions or that the resulting reconstruction is parsimonious (sparsity constraints).

3.2 Equivariant and invariant representations from group symmetries

We briefly recall elements from the theoretical and computational framework of group orbits [3, 2, 60, 52] towards representations that are equivariant [30, 10] or invariant [31, 47] with respect to transformations described by group symmetries. Such properties are useful for predicting the response to transformed inputs, efficient weight sharing schemes and reducing the sample complexity of a supervised learning task.

Let the input space be a vector space endowed with dot product $\langle \cdot, \cdot \rangle$, e.g. $\mathcal{X} = \mathbb{R}^d$. We denote the transformations of a point (signal) $x \in \mathcal{X}$ through a finite group $\mathcal{G}$, of order $|\mathcal{G}| < \infty$ in d dimensions, by gx where $g \in \mathcal{G}$ is a $d \times d$ matrix. Examples include the set of cyclic symmetries of a polygon, the matrices for planar image rotations, or the rotation-reflection symmetries of images or shapes (dihedral symmetry). The *group orbit* of $x \in \mathcal{X}$ is the set of transformed signals

$$O_x = \{gx \in \mathcal{X} |\, g \in \mathcal{G}\}. \tag{3}$$

In this way, the action of the group, and the associated orbit definition, induces a partition of $\mathcal{X}$ into orbit sets. More precisely, it defines an equivalence relation

$$x \sim x' \Leftrightarrow \exists g \in \mathcal{G} \colon x' = gx \tag{4}$$

that separates points in $\mathcal{X}$ on the basis of x being a transformation of x'.

Orbits can be used to derive a representation $\Phi : \mathcal{X} \to \mathcal{F}$ selected from some hypothesis space of maps with a specific parametrization. A parametrization that can result in equivariant (or invariant as a special case), and selective, representations is based on nonlinear measurements of the projections of a signal on orbits [2, 3]. More specifically given a set of orbits

$$\mathcal{W} = \{W_j = (g_1 t_j, \ldots, g_{|\mathcal{G}|} t_j)\},\ t_j \in \mathcal{X} \tag{5}$$

we consider the nonlinear projections, using a set of nonlinear functions $\{\sigma_\alpha : \mathbb{R} \to \mathbb{R}\}$ with a scalar parameter α, of x onto the orbit elements

$$\phi_{j,\alpha}(x) = \sigma_\alpha(W_j^T x) = (\sigma_\alpha(\langle g_1 t_j, x \rangle), \ldots, \sigma_\alpha(\langle g_{|\mathcal{G}|} t_j, x \rangle)), \tag{6}$$

with $j = 1, \ldots, |\mathcal{W}|$ and $\alpha = 1, \ldots, |\{\sigma_\alpha\}|$. Each $\phi_{j,\alpha} : \mathcal{X} \to \mathbb{R}^{|\mathcal{G}|}$, corresponding to a single orbit, is a *permutation-equivariant* representation w.r.t. the transformations in $\mathcal{G}$ i.e.:

$$\phi_{j,\alpha}(gx) = P_g \phi_{j,\alpha}(x),\ g \in \mathcal{G}, \forall j, \alpha \tag{7}$$

where P_g is a permutation matrix that depends on the transformation g. The representation of a transformed gx is simply a permuted version of the untransformed x. In Sec. (7.1), we provide a way to quantify equivalence using such permutation-equivariance representations.

By summing the components of $\phi_{j,\alpha}(x)$, we obtain an *invariant* representation $\bar{\phi}_{j,\alpha} : \mathcal{X} \to \mathbb{R}$ over transformation in $\mathcal{G}$:

$$\bar{\phi}_{j,\alpha}(x) = \sum_{i=1}^{|\mathcal{G}|} \sigma_\alpha(\langle g_i t_j, x \rangle) = \bar{\phi}_{j,\alpha}(gx),\ g \in \mathcal{G}, \forall j, \alpha \tag{8}$$

as transforming x amounts to simply re-ordering the terms in the sum, i.e. $\bar{\phi}_{j,\alpha}(gx) = \mathbf{1}_{|\mathcal{G}|}^T \phi_{j,\alpha}(gx) = \mathbf{1}_{|\mathcal{G}|}^T P_g \phi_{j,\alpha}(x) = \mathbf{1}_{|\mathcal{G}|}^T \phi_{j,\alpha}(x) = \bar{\phi}_{j,\alpha}(x)$. Equivalently, for unitary groups this can be seen as forming the group-average of $\sigma_\alpha(\langle t, gx \rangle)$ over $\mathcal{G}$. Moreover a map $\Phi : \mathcal{X} \to \mathbb{R}^{|\mathcal{W}| \times |\{\sigma_\alpha\}|}$, such that

$$\Phi(x) = \left(\bar{\phi}_{j,\alpha}(x)\right), \tag{9}$$

i.e. the concatenation of a finite number of invariant maps, corresponding to a finite number of orbit, can be proven to also be approximately *selective* w.r.t. the partition of $\mathcal{X}$ induced by $\mathcal{G}$, i.e. sufficient for separating the equivalence classes in Eq. (4) (see [2, 3] for details).

4. Symmetry-adapted representation learning

4.1 Motivation

A requirement for the construction of an equivariant (or invariant) signature as in Eq. (9) is having an orbit set, of an an unknown transformation group, as the set of representation weights. An orbit is a set of instances that have a transformation relationship, and is symmetric under a group transformation. We describe a number of principles that can be used for designing symmetry-adapted representation learning algorithms, where the learned weights reflect the symmetries in the training set. This can also be seen as obtaining orbits, directly from data, through a learning process. Learning is unsupervised, as no orbit membership is used, although it is not a clustering approach, as the learned orbits are not extracted from the training set instances. Note that for orbit selective representations, multiple orbits would be required [3], which can be the result of different initializations or local minima, or multiple orbit learning schemes.

4.2 Problem formulation

Before proceeding, we make a simplifying assumption on the training set distribution which we will maintain for the rest of the paper:

Assumption 1. *The input data set $S_N = \{x_i\}_{i=1}^N \subset \mathcal{X}$ is a finite collection of Q orbits w.r.t. a finite group $\mathcal{G}$*

$$S_N = \{x_i\}_{i=1}^N = \{gx_j,\ \forall g \in \mathcal{G}\}_{j=1}^Q,\ \ x_i, x_j \in \mathcal{X},\ \ N = |\mathcal{G}|Q \tag{10}$$

where $|\mathcal{G}|$ is the group cardinality and we assume $\mathcal{X} = \mathbb{R}^d$.

For symmetry-adapted representation learning, under this assumption for the training set, we will formulate the learning problem as

$$\min_W \left(\mathcal{L}(W, S_N) + \mathcal{L}'(W, S_N) + \beta \mathcal{J}(W)\right), \quad \beta \in \mathbb{R}_+ \tag{11}$$

where $\mathcal{L}(W, S_N)$ is a loss for representation learning[1], that can be dictated by a downstream task, e.g. reconstruction, discriminability or empirical error; $\mathcal{J} : \mathbb{R}^{d \times |\mathcal{G}|} \to \mathbb{R}$ is a regularization term that enforces the representation to be symmetry-adapted, meaning the weights W to form an *orbit of a finite group*; $\mathcal{L}'(W, S_N)$ an additional loss or data-dependent regularization term that will further restrict the group in W to be the same as the latent group in the training set distribution.

[1] Typically, an average over points of a loss $l(W, x)$, $l : \mathbb{R}^{d \times |\mathcal{G}|} \times \mathbb{R}^d \to \mathbb{R}_+$, i.e. $\mathcal{L}(W, S_N) = \frac{1}{N} \sum_i l(W, x_i)$.

The contribution of this paper is an analytic formulation for $\mathcal{L}'$ and $\mathcal{J}$ based on a general framework for learning W with symmetries from a finite, unknown group. Our roadmap for Sections 5 and 6 is the following:

- In Sec. 5 we describe a simple principle for constructing of regularization terms $\mathcal{J}$, by looking at the structure of the Gramian matrix W^TW induced by an orbit structure in W (Observation 1).

- In Sec. 6.1 we show how to use a maximal invariant for the permutation group to derive an analytical expression for $\mathcal{J}$ that enforces the desired Gramian matrix structure described in Obs 1.

- In Sec. 6.2 we propose a loss term $\mathcal{L}'$ that penalizes symmetries in W different from the ones in S_N, to further guide the orbit structure enforced by $\mathcal{J}$, using the symmetry in the data and weight covariances (Lemma 1).

5. Structure of the Gram matrix of orbits

In this Section we describe a simple principle for constructing a regularization term $\mathcal{J}(W)$ for representation learning that drives the weights to form an orbit of a *template* vector $t \in \mathbb{R}^d$ w.r.t. a finite group $\mathcal{G}$. More formally, we want the representation to have the following property:

$$\{w_i\}_{i=1}^{|\mathcal{G}|} = \{gt, \forall g \in \mathcal{G}\}, \;\; w_i, t \in \mathbb{R}^d. \tag{12}$$

For the time being, we pose no further requirements on the type of group or the correspondence to observations from a training set.

If the weight vectors form an orbit of some t w.r.t. to some $\mathcal{G}$, then the matrix of weights is, up to permutations by $\{g_i\}$:

$$W = (g_1 t, \ldots, g_{|\mathcal{G}|} t) \in \mathbb{R}^{d \times |\mathcal{G}|}, \;\; g_i \in \mathcal{G}, t \in \mathbb{R}^d, \tag{13}$$

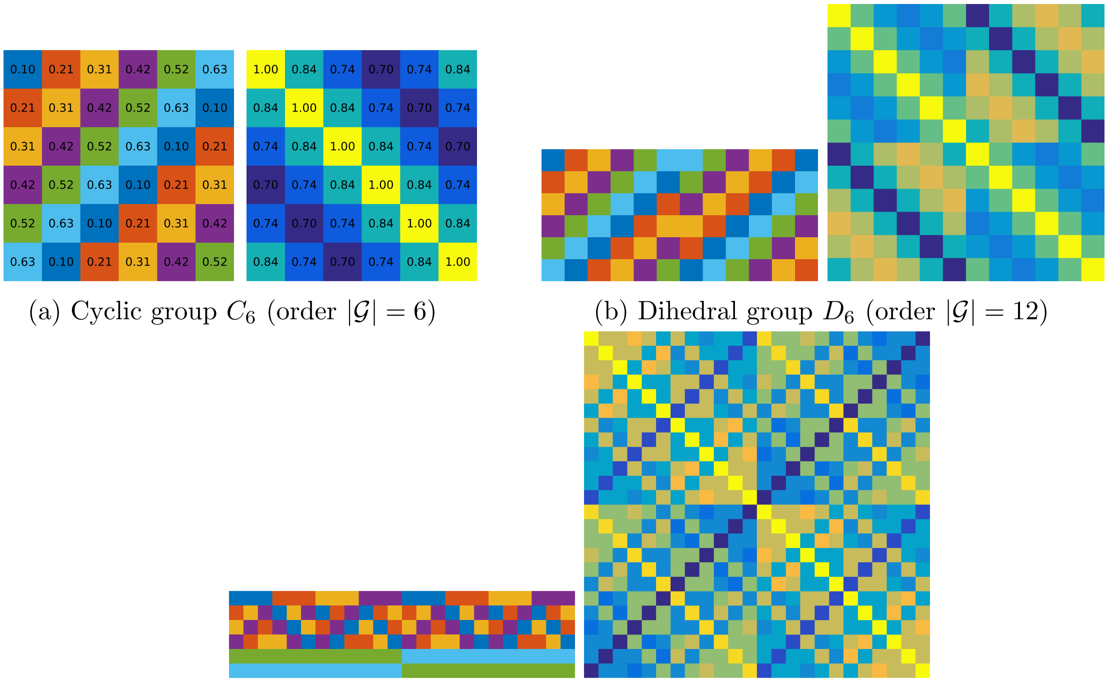
and the associated Gramian matrix $G = W^TW \in \mathbb{R}^{|\mathcal{G}| \times |\mathcal{G}|}$ has entries of the form:

$$(G)_{ij} = \langle g_i t, g_j t \rangle = \langle t, g_i^* g_j t \rangle \,, \tag{14}$$

where g_i^* the conjugate transpose of g_i. A matrix is called *permuted* if its columns are permutations of a single vector (Appendix, Sec. A and [14]); assuming a unitary group, i.e. $g^* = g^{-1}$ (a hypothesis that can be relaxed), and using the closure property of the group composition we can formulate the simple key observation of this work:

Observation 1. *If a set of vectors is an orbit w.r.t. a finite group then their Gramian is a permuted matrix.*

The property follows from the fact that the Gramian matrix is a function of the multiplication table of the finite group $\mathcal{G}$. More precisely, by *Cayley's theorem* [8] the permutations associated to a finite group multiplication table (often called a *Cayley table*) form a group, a subgroup of the symmetric group of $|\mathcal{G}|$ objects $S_{|\mathcal{G}|}$, that is isomorphic to $\mathcal{G}$ (see Appendix, Sec. C).



(a) Cyclic group C_6 (order $|\mathcal{G}| = 6$)

(b) Dihedral group D_6 (order $|\mathcal{G}| = 12$)

(c) Pyritohedral group T_h (order $|\mathcal{G}| = 24$)

Figure 1: Examples of group action transformations, for groups of orders $|\mathcal{G}| = \{6, 12, 24\}$, on a unit-norm vector in $\mathbb{R}^d, d = 6$. Each subfigure shows the $d \times |\mathcal{G}|$ matrix W of orbit elements arranged columnwise (left), and the $|\mathcal{G}| \times |\mathcal{G}|$ Gramian matrix $W^T W$ (right).

Example 1. *[Cyclic group (order 3)] Let $\mathbb{Z}_3 = \{0, 1, 2\}$ with law $i \circ j = (i + j \pmod 3)$. Each column in the multiplication table associated to the elements of $\mathbb{Z}_3$ is a permutation of the others:*

$\circ$	***0***	***1***	***2***
0	*0*	*1*	*2*
1	*1*	*2*	*0*
2	*2*	*0*	*1*

Therefore any injective function of the multiplication table entries will give a permuted matrix. In our case this function, $\nu_t : \mathcal{G} \times \mathcal{G} \to \mathbb{R}$, is given by the scalar product $\nu_t(g_i, g_j) = \langle t, g_i^{-1} g_j t \rangle$.

Remark 1. *For the case of permutation groups, taking any elementwise function of the orbit, $f((W)_{ij}), f : \mathbb{R} \to \mathbb{R}$, the induced Gramian with entries $(G)_{ij} = \langle f(g_i t), f(g_j t) \rangle$ is also a permuted matrix. In fact the action of the function on the orbit W is equivalent to a change of the template t since $(f(g_i t))_q = (g_i f(t))_q, \quad \forall\ i, q$. This observation will be useful for restricting the set of possible solutions W of a learning process (Sec. 8) driven by imposing symmetry constraints (Theorem 2).*

6. Analytic expressions for regularization

In this section we provide closed-form expression for two conditions: a) the *permuted matrix codition* from Observation 1 that enforces the group orbit structure on a representation matrix, based on the notion of maximal invariants [46, 59] and b) the *same-symmetry condition* for imposing such group to be the same as that generating the training set.

Definition 3. *A maximal invariant h w.r.t. a group $\mathcal{G}$ is a function such that:*

$$h(u) = h(v) \;\Leftrightarrow\; \exists g \in \mathcal{G} : u = gv, \;\; u, v \in \mathbb{R}^d \tag{15}$$

In our context, a maximal invariant is the same on all elements of an orbit but different on elements of different orbits, i.e. it separates classes of equivalence.

Since we are interested in enforcing the condition of permuted matrices we consider maximal invariants of the permutation group for vectors $u \in \mathbb{R}^d$. In particular maximal invariants of the form:

$$h(u, \lambda) = \sum_{i=1}^{d} f_\lambda(u_i), \;\; \lambda \in \mathbb{R}. \tag{16}$$

where $f_\lambda : \mathbb{R} \to \mathbb{R}$ a function with a free scalar parameter $\lambda \in \mathbb{R}$, applied elmentwise on u. Thus $u, v \in \mathbb{R}^d$ are equivalent up to permutations if-and-only-if $h(u, \lambda) = h(v, \lambda), \; \forall \lambda \in \mathbb{R}$.

6.1 Maximal invariant for a permuted Gramian

A special case of a maximal invariant is the probability distribution of the components of the vector, or more precisely, since we consider a finite vector space, the histogram of the components. A simple way of building the component histogram for $v \in \mathbb{R}^d$ is to use the delta function $f_\lambda(\cdot) = \delta(\cdot - \lambda)$ in Eq. (16)

$$h(v, \lambda) = h_v(\lambda) = \sum_{i=1}^{d} \delta(v_i - \lambda) \tag{17}$$

The value $h_v(\lambda)$ corresponds to the frequency of the value λ in the d components of vector v. We can then use a distance, e.g. the Euclidean, between histograms to quantify if two vectors are equivalent up to permutations: $v \sim u$ iff the distance of functions h_v, h_u is zero:

$$\|h_u - h_v\|_2^2 = 0 \implies \int d\lambda \; (h_u(\lambda) - h_v(\lambda))^2 = 0. \tag{18}$$

In the context of Observation 1, we consider the Gramian matrix $G = W^T W$. A condition to guarantee that each column of G is a permutation of a single vector can therefore be written as

$$\sum_{ij=1}^{|\mathcal{G}|} \Big(\int d\lambda \; (h_i(\lambda) - h_j(\lambda)) \Big)^2 = 0, \;\; h_i(\lambda) = \sum_{k=1}^{|\mathcal{G}|} \delta(G_{ki} - \lambda), \tag{19}$$

with $G_{ki} = (G)_{ki}$ the element in row k and column i of G. In words, Eq. (19) calculates the histogram of the column value distribution for different columns and imposes the distance to be null. The corresponding maximal invariant $h_i(\lambda) = h(G_{:i}, \lambda)$ is the distribution function of the Gramian columns (rows) $G_{:i}, i = 1 \dots |\mathcal{G}|$. This condition, which forces the distribution functions to be the same, is a necessary and sufficient condition for the rows (columns) of the Gramian to be permutations of a single vector.

Theorem 1 (Symmetry regularization)**.** *Let matrix $W \in \mathbb{R}^{d \times |\mathcal{G}|}$ and $r : \mathbb{R}^{|\mathcal{G}| \times |\mathcal{G}|} \to \mathbb{R}_+$ with*

$$r(W^T W) = \tau^T \delta(C vec(G)), \;\; G = W^T W \tag{20}$$

where G is the Gramian matrix, C is a constant matrix calculating all differences of the components of vector $vec(G) \in \mathbb{R}^{|\mathcal{G}|^2}$, the function δ acts elementwise ($\delta(a) = 1$ if $a = 0$, otherwise 0), and τ is a constant weight vector. If $r(W^T W) = 0$, then G is a permuted matrix and viceversa (details and proof in Appendix, Sec. B).

Other choices for the nonlinearity in Eq. (16), or the choice of the distance in Eq. (18), will give rise to different ways to enforce orbit regularization (see Appendix, Sec. F). It is worth noting that for the case of permutation groups, and using the property in Remark 1, Theorem 1 can be generalized to derive a class of regularization terms. In particular the following holds:

Corollary 1. *In the hypothesis of Theorem 1 for any permutation group $\mathcal{G}$ and any $f : \mathbb{R} \to \mathbb{R}$ acting pointwise on the matrix W if $W^T W$ is a permuted matrix then $r(f(W^T) f(W)) = 0$.*

Remark 2 (On using the Gramian)**.** *Theorem 1 specifies a symmetry constrain for a matrix W by using the Gramian matrix $G = W^T W$. Imposing constrains on G instead of directly on W is crucial for dealing with arbitrary groups. In effect, working with G implies dealing with the permutation group only. We can then avoid the need to consider group-specific regularization strategies, and deal with a variety of groups through permutations.*

6.2 Same-symmetry condition

The condition in Observation 1 is clearly necessary but not sufficient for a set of weights to be an orbit w.r.t. a finite group. Furthermore, it does not depend on the data structure i.e. its generating group could be different from the one that generated the training data S_N.

One possible way to constrain the set of solutions to have the same symmetries comes from the analysis of the covariance matrix of a collection of orbits. If $X = (x_1, \dots, x_N) \in \mathbb{R}^{d \times N}$ is the matrix storing the training set S_N in (10), we have:

$$X X^T = \sum_{i=1}^{|\mathcal{G}|} g_i T T^T g_i^T \tag{21}$$

where $T = (x_1, \dots, x_Q)$ is the $d \times Q$ matrix of the representatives for each orbit in S_N. The matrix in 21 has a high degree of symmetry that can be used to prove the following (see Appendix, Sec. C for details). Let $[A, B] = AB - BA$ the commutator of matrices A, B:

Lemma 1. *Let W be an orbit w.r.t. a finite group $\tilde{\mathcal{G}}$ and $X = (x_1, \dots, x_N) \in \mathbb{R}^{d \times N}, x_i \in S_N$ as in (10) with $|\mathcal{G}| = |\tilde{\mathcal{G}}|$. If $[X X^T, W W^T] = \mathbf{0}$ then $\tilde{\mathcal{G}} = \mathcal{G}$ (up to a constant unitary matrix conjugation).*

7. Learning with symmetry regularization

The main theoretical result of the paper is then summarized by the following Theorem, which puts together Observation 1, Theorem 1 and Lemma 1. The proof and details are provided in the Appendix, Sec. C.

Theorem 2 (Symmetry adapted regularization)**.** *Let $X \in \mathbb{R}^{d \times Q|\mathcal{G}|}$ a matrix whose columns are all elements from the union of orbits of Q vectors in $\mathbb{R}^d$ w.r.t. a finite group $\mathcal{G} = \{g_i,\ i = 1, \ldots, |\mathcal{G}|\}$, where $g_i \in \mathbb{R}^{d \times d}$ is a matrix representation of the group action and $|\mathcal{G}|$ indicates the cardinality of the group. Let $W \in \mathbb{R}^{d \times |\mathcal{G}|}$ a matrix whose columns are the elements of an orbit of a vector $w \in \mathbb{R}^d$ w.r.t. a finite group $\tilde{\mathcal{G}}$ with $|\mathcal{G}| = |\tilde{\mathcal{G}}|$. Then W^TW is permuted and $r(W^TW) = 0$. Further if $[XX^T, WW^T] = \mathbf{0}$ then $\tilde{\mathcal{G}} = \mathcal{G}$ (up to a constant unitary matrix conjugation).*

Incorporating both conditions in Theorem 2, the general form for a symmetry-adapted representation learning problem in Eq. (11) is of the form

$$\min_{W} \left(\mathcal{L}(W, S_N) + \gamma \left\| [XX^T, WW^T] \right\|_F^2 + \beta r(W^TW) \right), \quad \beta, \gamma \in \mathbb{R}_+ \tag{22}$$

where we used $\mathcal{L}'(W, S_N) = \left\| [XX^T, WW^T] \right\|_F^2$, $\mathcal{J}(W) = r(W^TW)$ and the loss term $\mathcal{L}(W, S_N)$ depends on the problem. In the following, we use a smooth (differentiable) version of Eq. (20) substituting the δ function with a Gaussian function $g_\sigma : \mathbb{R} \to \mathbb{R}$ of $\sigma \ll 1$ width. The regularizer is then the form $r(W^TW) = \tau^T g_\sigma(C\text{vec}(G))$. Details on C and τ, along with the analytic form of the gradients for $\mathcal{L}(W)'$ and $\mathcal{J}(W)$, for use in gradient-based updating rules for minimizing Eq. (22) are provided in Appendix, Sec. B.

Remark 3 (Loss and data-dependent terms)**.** *For the loss term, one could consider for example reconstruction or distance-preservation (for unsupervised learning), similarity or discriminability (for weakly supervised learning) or prediction accuracy (for supervised learning). Note that the commutator term $\mathcal{L}(W, S_N)'$ is also data-dependent and can be seen either as a data-dependent regularization or an additional loss function that penalizes different-symmetry solutions. In the context of supervised or weakly supervised learning, the pseudo-distance defined below in Eq. (23) can be a valid loss $\mathcal{L}(W, S_N)$.*

7.1 Permutation invariant pseudometric for testing solutions

As outlined in Sec. 3.2, a representation of the form of Eq. (6) is permutation-equivariant. To quantify equivalence under permutations, and in our context equivalence under general symmetries and thus symmetry structure in the representation W, we can use a modified version of the regularizer function in Eq. (20), where the input to r is a 2-column instead of a symmetric matrix. The motivation comes from the fact that the representations $\Phi(x), \Phi(x')$ of two elements $x, x' \in \mathbb{R}^d$ of the same orbit O_x w.r.t. an orbit dictionary W (generated by the same group) are permuted vectors. Given a representation matrix $W \in \mathbb{R}^{d \times m}$, and the corresponding coefficients through linear projections $\Phi(x) = W^Tx = (\langle w_1, x \rangle, \ldots \langle w_m, x \rangle)$, such that $\Phi : \mathbb{R}^d \to \mathbb{R}^m$, we define the following pseudometric

$$D(x, x') = r((\Phi(x), \Phi(x'))) = r(W^T(x, x')) = \tau^T \delta(C(\Phi(x), \Phi(x'))),\ \Phi(x) \in \mathbb{R}^m \tag{23}$$

where the input to r is now the $m \times 2$ matrix $(\Phi(x), \Phi(x'))$. The quantity $r : \mathbb{R}^{m \times 2} \to \mathbb{R}_+$ is a pseudometric[2] and

$$\Phi(x) = P_g \Phi(x') \iff r((\Phi(x), \Phi(x'))) = 0. \tag{24}$$

Suppose now that W is learned by the minimization in Eq. (22), which implies $r(W^T W) = 0$ and $\left\|[XX^T, WW^T]\right\|_F^2 = 0$ are satisfied, and still $\Phi(x) = W^T x$ with $m = |\mathcal{G}|$. We then have:

Lemma 2 (Pseudometric)**.** *If W satisfies the constrains in Theorem 2 then:*

$$r(W^T(x, x')) = 0 \iff x = gx' \quad \forall\, x, x' \in \mathbb{R}^d,\ g \in \mathcal{G}.$$

For the direct implication note that the Gramian matrix $W^T W$ identifies W up to an arbitrary (but fixed) unitary transformation U; together with the regularization constrain $r(W^T W) = 0$ we thus have that W can be written as $W = UW_o$ where W_o is an orbit matrix (see Supp. Mat., Sec. E for the full proof). This result translates to r being zero for elements of the same orbit (inter-orbit) and large for elements of different orbits (intra-orbit).

Thus values of r for pairs of vectors x, x' can then be used to test the following: a) are $\Phi(x)$ and $\Phi(x')$ are permuted vectors (from Eq. (24))? and b) do x, x' belong to the same orbit, for $\Phi(x) = W^T x$ and W learned or satisfying the orbit regularization conditions (from Lemma 2)? And as an extension, does W provide an equivariant map, i.e. is it an orbit of the same group?

8. Experimental results on unsupervised learning

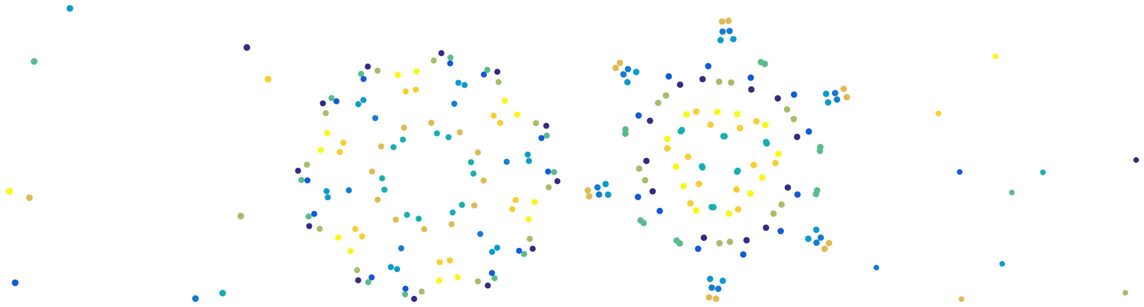
As a proof-of-concept, the proposed regularization conditions are evaluated for the unsupervised learning of a symmetry-adapted representation from orbit data. The problem is formulated as follows: *given an unlabeled training set which is a union of orbits from an unknown group $\mathcal{G}$, i.e. $S_N = \{gx_j,\ \forall g \in \mathcal{G}\}_{j=1}^{Q}$, $x_j \in \mathbb{R}^d, N = |\mathcal{G}|Q$ as in Eq. (10), find a single orbit W (of an arbitrary vector $t \in \mathbb{R}^d$) of the (latent) group $\mathcal{G}$ that generated the data, i.e. the structure of the columns of learned W should follow Eq. (12).*

In the absence of additional requirements for W, the term $\mathcal{L}(W, S_N)$ in Eq. (22) is not used. Moreover, to strengthen the (necessary but not sufficient) orbit condition imposed by minimizing $r(W^T W)$ (Theorem 2), we use a sum of J regularizers; each term is derived by applying a pointwise nonlinearity on W, $f_\alpha((W)_{ij}) = \sigma((W)_{ij} - \alpha), \sigma : \mathbb{R} \to \mathbb{R}$ and computing the regularizer on the generalized Gramian $G_\alpha : (G_\alpha)_{ij} = \langle f_\alpha(w_i), f_\alpha(w_j) \rangle$. If W is an orbit matrix, G_α should also be a permuted matrix (Remark 1). The minimization problem for learning W takes the following form

$$\min_W \left(\left\|[XX^T, WW^T]\right\|_F^2 + \frac{\beta}{J} \sum_{j=1}^{J} r(\sigma(W^T - \alpha_j)\sigma(W - \alpha_j)) \right), \quad \beta \in \mathbb{R}_+ \tag{25}$$

where X is the $d \times N$ matrix storing the training set S_N, $\sigma(x - \alpha_j) = \max(x - \alpha_j, 0)$ is the rectifier nonlinearity with parameter α_j, J is a hyper-parameter on the number of

[2] It is easy to see that it fulfills the same properties as a metric but allows the distance between two different points to be zero (when the two vectors are permuted).



(a) 10 seeds, MDS (b) orbits, MDS (c) orbits, tSNE (d) orbits, MDS $r(x, x')$

Figure 2: Visualization of seed vectors and generated orbits (encoded by different colors) from the training data distribution (Dihedral group D_6 (order 12)). (a) Seed vectors Q uniformly sampled from the unit ball, shown via 2d metric Multidimensional Scaling (MDS). Orbit samples ($Q \times 12$) are shown as 2d MDS and 2d t-SNE projections in (b) and (c) respectively. Applying MDS with the proposed $r(x, x')$ permutation test instead of the Euclidean distance (d) collapses the permutation orbits back into single points.

rectifiers or α_j values, and β is a constant controlling the trade-off between the two terms. The regularization terms r, that aim for a symmetry structure in the columns of W, act on the data-dependent, covariances commutator term which promotes W with the exact $\mathcal{G}$-symmetries in X. Note that the single $r(W^T W)$ term in Eq. (22) is a special case of the used regularization sum, derived by setting $J = 1$ and $\alpha_1 << 0$.

8.1 Synthetic group data

We use synthetic data generated from known, finite mathematical groups: cyclic group C_6 of order $|\mathcal{G}| = 6$ (an Abelian group), dihedral group D_6 of order $|\mathcal{G}| = 12$ and pyritohedral group T_h of order $|\mathcal{G}| = 24$ (non Abelian groups). All groups are acting on $\mathbb{R}^6$. Figure 1 visualizes the action of the groups on a sample vector. Seed vectors, uniformly sampled at random from the unit ball in $\mathbb{R}^6$, to generate train and validation sets, of size $Q = 1000$ and $Q = 200$ orbits respectively, amounting to $Q * |\mathcal{G}|$ points, where $|\mathcal{G}|$ the order of the group. Examples of 2D projections from the training set distribution of the Dihedral group are shown in Fig. 2. In (a) and (b) we plot in 2D the seeds (templates) and generated D_6 orbits using metric Multidimensional Scaling (MDS) on the matrix of cosine distances. We also visualize the orbits using t-SNE in 2D (c) and MDS on the pseudometric $r(x, x')$ of Eq. (24) in (d). Note how the latter collapses the permutation group orbits into single points.

8.2 Training

For the reported results, we tried schemes with the hyper-parameter J set to 100 and 300, by selecting α_j uniformly spaced in $[-1, 1]$. In addition, we use an explicit $r(W^T W)$ term. For the value of β, that controls the amount of regularization, we tried a range of orders of magnitude, namely $\log_{10} \beta = \{0, \ldots, -7\}$. Minimization of Eq. (25) was performed by quasi-Newton iterative optimization with a cubic line search, using Broyden–Fletcher–Goldfarb–Shanno (BFGS) for the Hessian matrix approximation. The training

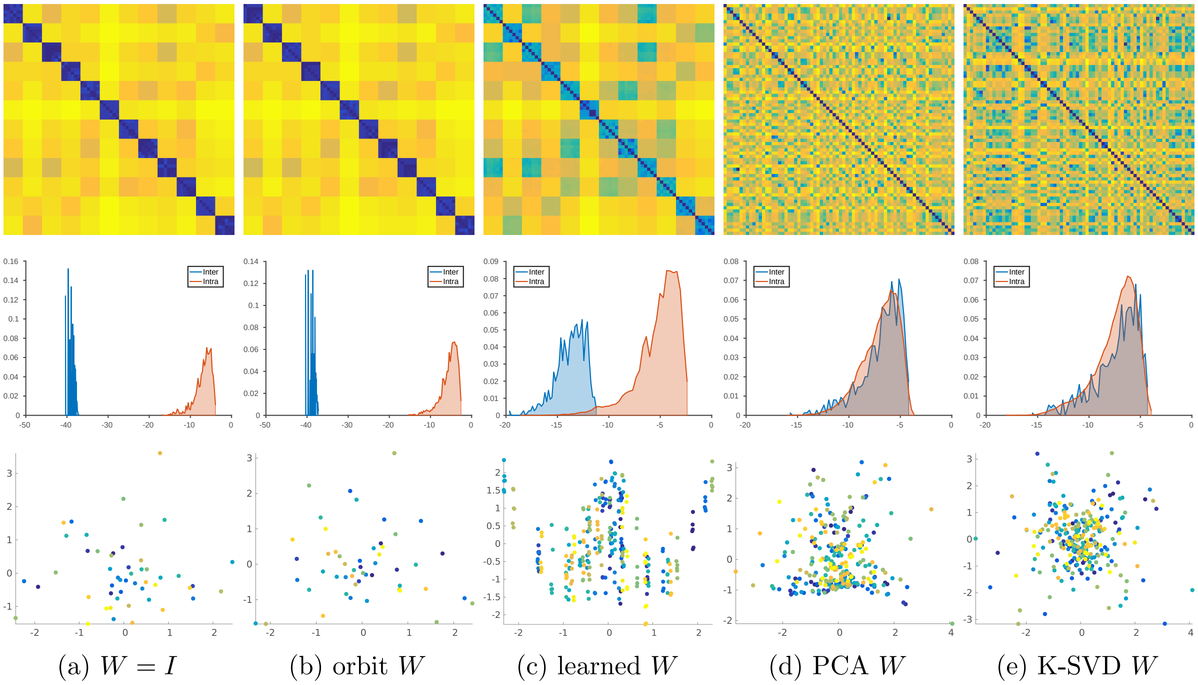
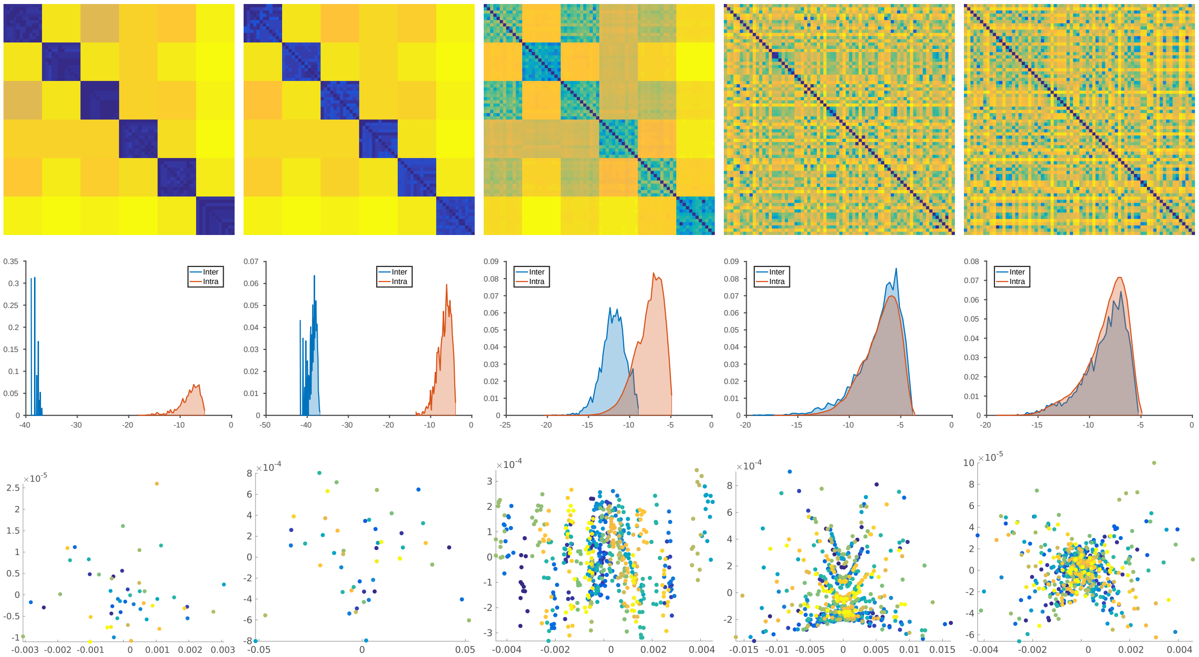


Figure 3: **Cyclic group C_6 (order 6)**: permutation pseudometric Eq. (23) $r(\Phi(x), \Phi(x'))$, with $\Phi(x) = W^T x$ and W a 6×6 dictionary. (a) input space, i.e. $W = I$, (b) true orbit from validation set, (c) learned solution, (d) PCA on the training set, (e) k-SVD with minimal sparsity. *top*: distance matrices for 3 orbits (arranged in 12 blocks of 6 orbit elements); *middle*: inter-/intra-orbit distributions for 50 orbits of the validation set; *bottom*: 2D output of metric multidimensional scaling on the matrix of distances $r(\Phi(x), \Phi(x'))$(shown as z-score standardized coordinates).

data X are shuffled and the dictionary matrix W is initialized at random. Note that the size of the target orbit $|\mathcal{G}|$, and thus the column size of W ($d \times |\mathcal{G}|$) is assumed given. We initialize and run the minimization process M times to produce M orbit solutions for each value of β, corresponding possibly to different local minima of the loss function. Examples of the output of the minimization, in terms of W and $G = W^T W$ and the dependency of the solutions on the β and J values are shown in Appendix, Sec. H.

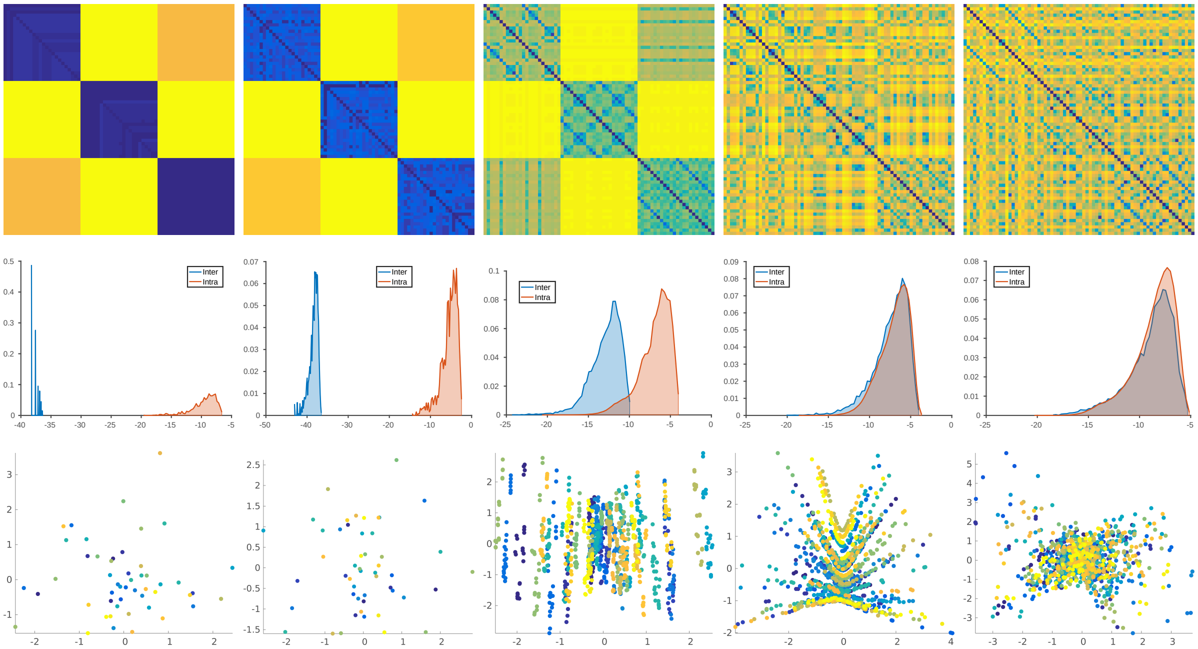
8.3 Quantitative comparisons

To test the validity of solutions, and thus the approximation of the underlying data symmetries in the training set, we use the pseudometric on the permutation invariant quotient space in Eq. (23). Given a representation matrix W and assuming $\Phi(x) = W^T x$ we compute $r(\Phi(x), \Phi(x'))$ for pairs (x, x') in $\mathcal{X}$ the validation set. *We can then check if W induces an equivariant representation and for the case of the learned matrices with the proposed method, if the equivariance is induced by a matrix with the symmetries in the data.* The latter is also true for any representation W that satisfies the constraints in Theorem 2, according to Lemma 2, which are necessary but not sufficient; examples are matrices that give an



(a) $W = I$ (b) orbit W (c) learned W (d) PCA W (e) K-SVD W

Figure 4: **Dihedral group** D_6 **(order 12)**: permutation pseudometric Eq. (23) $r(\Phi(x), \Phi(x'))$, with $\Phi(x) = W^T x$ and W a 6×12 dictionary. (a) input space, i.e. $W = I$, (b) true orbit from validation set, (c) learned solution, (d) PCA on the training set, (e) k-SVD with minimal sparsity. *top*: distance matrices for 3 orbits (arranged in 6 blocks of 12 orbit elements); *middle*: inter-/intra-orbit distributions for 50 orbits of the validation set; *bottom*: 2D output of metric multidimensional scaling on the matrix of distances $r(\Phi(x), \Phi(x'))$(shown as z-score standardized coordinates).



(a) $W = I$ (b) orbit W (c) learned W (d) PCA W (e) K-SVD W

Figure 5: **Pyritohedral group** T_h **(order 24)**: permutation pseudometric Eq. (23) $r(\Phi(x), \Phi(x'))$, with $\Phi(x) = W^T x$ and W a 6×24 dictionary. (a) input space, i.e. $W = I$, (b) true orbit from validation set, (c) learned solution, (d) PCA on the training set, (e) k-SVD with minimal sparsity. *top*: distance matrices for 3 orbits (arranged in 3 blocks of 24 orbit elements); *middle*: inter-/intra-orbit distributions for 50 orbits of the validation set; *bottom*: 2D output of metric multidimensional scaling on the matrix of distances $r(\Phi(x), \Phi(x'))$(shown as z-score standardized coordinates).

orthonormal Gramian, implying a trivial permutation orbit in W. For the analysis and evaluation we assume that the validation set has labeled orbits, i.e. we know all inter- and intra-orbit pairs.

We present the same analysis using different types of W: a) $W = I_{d\times|\mathcal{G}|}$, i.e. the I matrix in $\mathbb{R}^{d\times|\mathcal{G}|}$ such that $\Phi(x) = x$. This serves as a sanity check for the pseudodistance as, for the case of permutation groups, each orbit is composed from permuted vectors. b) W is an orbit selected from the same distribution as the training set, i.e. a ground truth W; this serves as a sanity check that the same symmetry will indeed induce equivariance in $\Phi(x)$. c) W is a learned representation via minimization of Eq. (25), using a single β and the min-loss in convergence out of M initializations. d) W are the eigenvectors of the data covariance (PCA). e) W is the result of a sparse dictionary learning algorithm (K-SVD [54]). The last two are included as baseline comparisons in case symmetries can be discovered through high-variance directions and sparsity/reconstruction constraints.

Figures 3, 4 and 5 show a part of the distance matrices for the three groups corresponding to 12, 6 and 3 orbits respectively, and the probability distributions of pairwise distances for the validation set (50 orbits and in total 300, 600 and 1200 points resp.), estimated for inter- and intra-orbit pairs separately. All representations, except K-SVD (d), satisfy the requirements of Lemma 2 and thus can be used to test if the symmetries in W are the symmetries in the data X. The block diagonal structure of the distance matrices in (a), (b), (c) are in favor of the main claims of the paper (as stated by Eq. (23) and Theorem 2), i.e., only permuted matrices give small distances. In addition exact symmetry, as in trivial orbit in (a), true orbit in (b) and learned solution in (c), induces permutations that approximate an equivariant map. Similar, the separability of the inter- and intra- distributions in (c), suggests that the learned W implies a quotient representation w.r.t. the unknown group of transformations. The figures also show 2D MDS projections. Note how (a) and (b) collapse orbits to single points (perfect permutations) and how (c) contracts and separates orbits, visible as small y-distance of intra-orbit points and densities in vertical lines respectively.

From these examples is is obvious that, for permutation groups, a distance such as Eq. (23) would be sufficient for separability, i.e. the representation space coincides with the input space; however this would only be meaningful for low-dimensional, permutation groups. A symmetry-adapted representation would be crucial for dealing with real, high-dimensional, generic group-transformed data.

9. Discussion and future work

We studied the problem of learning data symmetries as good prior on data structure, motivated by the need to reduce the sample complexity of a learning problem. In particular we explored mathematical conditions that can enforce a representation for the data, learned in an unsupervised way, to reflect the symmetries in the training set by being equivariant/invariant to transformations. The work is particularly relevant for data in high-dimensional perceptual spaces that have a low-dimensional intrinsic structure (e.g. transforming objects or sounds). Further the approach can be applied to any finite group since it reduces the analysis of them to permutation groups.[3]. The preliminary work outlined in this paper focused for simplicity on low-dimensional permutation symmetries. It also opens many additional, natural extensions to be explored.

One intriguing direction is that of sparsity. Interestingly group codes (first studied by Slepian in his seminal paper [45]) or dictionaries ([53]) tend to have low self-coherence. This is related to the exact recovery condition [13] which states that exact recovery of a signal x is guaranteed if $\|x\|_0 < 0.5(1 + 1/\mu(W))$, where μ is the self-coherence of the weights. In other words, reversing the inequality we have that, forcing a very sparse representation of the signals, if possible, induces the learned dictionaries to have a low self-coherence, which is a necessary condition for group dictionaries. A related research direction comes from the observation that group dictionaries are also good candidates to be equiangular tight frames [56]; in fact, interestingly, the smallest self-coherence is achieved for equiangular frames (Welsh bound [50]; see also [21]).

Another important direction is that of how and if the proposed algorithm can be extended to learn non-group transformations, which constitutes by far the majority of real signal transformations. A preliminary attempt to tackle this question is given in the Appendix, Sec. G. If we assume that the signal transformations define a smooth manifold, locally, a Lie group is defined by the generators on the tangent space. This allows the complex global transformation to be decomposed into a collection of local transformations that obey a group structure which can be learned by the proposed regularization scheme. This is also the concept of hierarchical-compositional networks (like CNNs) where complex global transformations are decomposed into a hierarchy of simple, local ones [34].

Further important topics that have not been discussed and are left for future work include the dependence of the method on noise in the orbits and uncertainty from missing elements (e.g. partial orbits), selecting the number of elements in the dictionary (assumed known here) and learning multiple orbits, not via initialization like in this work but through multi-orbit versions of the regularizer. The use of supervision, e.g. a discriminative loss for the learning objective, and the generalization of CNNs to convolutional groups learned from data and network topologies is the far-fetching goal of this work.

Acknowledgments

This material is based upon work supported by the Center for Brains, Minds and Machines (CBMM), funded by NSF STC award CCF-1231216. The authors gratefully acknowledge Alan Yuille, Maximilian Nickel, Silvia Villa and Carlo Cilliberto, for the insightful discussions and feedback on early versions of this work, and Saverio Salzo for pointing to the proof of Lemma 1 (Appendix, Sec. C).

References

[1] Y. S. Abu-Mostafa. Hints and the VC Dimension. *Neural Computation*, 5(2):278–288, Mar. 1993.

[3] For multilayer representations, e.g. deep networks, that use a map as in Eq. (8), the representation will be permutation-equivariant after the first layer. Additional layers will then process permutation-transformed signals.

[2] F. Anselmi, J. Z. Leibo, L. Rosasco, J. Mutch, A. Tacchetti, and T. Poggio. Unsupervised learning of invariant representations. *Theoretical Computer Science*, 633:112–121, Jun. 2016.

[3] F. Anselmi, L. Rosasco, and T. Poggio. On Invariance and Selectivity in Representation Learning. *Information and Inference*, 5(2):134–158, Jun. 2015.

[4] Y. Bengio. Learning Deep Architectures for AI. *Foundations and Trends® in Machine Learning*, 2(1):1–127, 2009.

[5] Y. Bengio, A. Courville, and P. Vincent. Representation Learning: A Review and New Perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1798–1828, Aug. 2013.

[6] J. Bruna and S. Mallat. Invariant Scattering Convolution Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1872–1886, Aug. 2013.

[7] J. Bruna, A. Szlam, and Y. LeCun. Learning Stable Group Invariant Representations with Convolutional Networks. In *International Conference on Learning Representations (ICLR)*, 2013.

[8] A. Cayley. On the Theory of Groups, As Depending on the Symbolic Equation $\theta^n = 1$. *Philosophical Magazine Series 4*, VII(42):40–47,408–409, 1854.

[9] T. Cohen and M. Welling. Learning the Irreducible Representations of Commutative Lie Groups. In *International Conference on Machine Learning (ICML)*, volume 32, pages 1755–1763, Feb. 2014.

[10] T. S. Cohen and M. Welling. Group Equivariant Convolutional Networks. In *International Conference on Machine Learning (ICML)*, pages 2990–2999, Feb. 2016.

[11] T. S. Cohen and M. Welling. Steerable CNNs. In *International Conference on Learning Representations (ICLR)*, Dec. 2017.

[12] S. Dieleman, J. De Fauw, and K. Kavukcuoglu. Exploiting Cyclic Symmetry in Convolutional Neural Networks. In *International Conference on Machine Learning (ICML)*, pages 1889–1898, 2016.

[13] D. L. Donoho and X. Huo. Uncertainty principles and ideal atomic decomposition. *IEEE Transactions on Information Theory*, 47(7):2845–2862, 2001.

[14] Y. Eldar and H. Bolcskei. Geometrically uniform frames. *IEEE Transactions on Information Theory*, 49(4):993–1006, Apr. 2003.

[15] P. Fackler. Notes on Matrix Calculus. `http://www4.ncsu.edu/~pfackler/MatCalc.pdf`, 2005.

[16] R. Gens and P. M. Domingos. Deep Symmetry Networks. In *Advances in Neural Information Processing System (NIPS)*, pages 2537–2545, 2014.

[17] R. Grosse, R. Raina, H. Kwong, and A. Y. Ng. Shift-Invariance Sparse Coding for Audio Classification. In *Twenty-Third Conference on Uncertainty in Artificial Intelligence (UAI2007)*, Jun. 2007.

[18] B. Haasdonk and H. Burkhardt. Invariant kernel functions for pattern analysis and machine learning. *Machine Learning*, 68(1):35–61, May 2007.

[19] G. E. Hinton, A. Krizhevsky, and S. D. Wang. Transforming Auto-encoders. In *Artificial Neural Networks and Machine Learning (ICANN 2011)*, volume 6791 of *Lecture Notes in Computer Science*, pages 44–51, 2011.

[20] M. Jaderberg, K. Simonyan, and A. Zisserman. Spatial Transformer Networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

[21] B. D. Johnson and K. A. Okoudjou. Frame potential and finite abelian groups. In *Contemp. Math*, volume 464 of *Contemp. Math.*, pages 137–148. 2008.

[22] P. Jost, P. Vandergheynst, S. Lesage, and R. Gribonval. MoTIF: An Efficient Algorithm for Learning Translation Invariant Dictionaries. In *IEEE International Conference on Acoustics Speed and Signal Processing (ICASSP)*, volume 5, pages 857–860, 2006.

[23] K. Kavukcuoglu, M. Ranzato, R. Fergus, and Y. LeCun. Learning invariant features through topographic filter maps. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR) Workshops*, Jun. 2009.

[24] J. J. Kivinen and C. K. I. Williams. Transformation Equivariant Boltzmann Machines. In *International Conference on Artificial Neural Networks (ICANN)*, 2011.

[25] A. Krizhevsky, I. Sutskever, and G. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012.

[26] A. Laub. *Matrix Analysis.* Cambridge University Press, 2005.

[27] Q. V. Le, J. Ngiam, Z. Chen, D. Chia, P. W. Koh, and A. Y. Ng. Tiled convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, 2010.

[28] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015.

[29] H. Lee, R. Grosse, R. Ranganath, A. Y. Ng, and R. Grosse. Unsupervised learning of hierarchical representations with convolutional deep belief networks. *Communications of the ACM*, 54(10), Oct. 2011.

[30] K. Lenc and A. Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[31] S. Mallat. Understanding Deep Convolutional Networks. *Philosophical Transactions of the Royal Society A*, 374(2065), Apr. 2016.

[32] J. Masci, U. Meier, D. Cireşan, and J. Schmidhuber. Stacked Convolutional Auto-Encoders for Hierarchical Feature Extraction. In *International Conference on Artificial Neural Networks (ICANN)*, pages 52–59, 2011.

[33] R. Memisevic. Learning to relate images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):1829–46, Aug. 2013.

[34] H. Mhaskar, Q. Liao, and T. Poggio. When and Why Are Deep Networks Better than Shallow Ones ? In *Proceedings of the 31th Conference on Artificial Intelligence (AAAI 2017)*, pages 2343–2349, 2017.

[35] X. Miao and R. P. N. Rao. Learning the Lie groups of visual invariance. *Neural computation*, 19(10):2665–2693, oct 2007.

[36] R. Negrinho and A. Martins. Orbit Regularization. In *Advances in Neural Information Processing Systems*, pages 3221–3229, 2014.

[37] T. Poggio and F. Anselmi. *Visual Cortex and Deep networks: Learning Invariant Representations.* MIT Press, 2016.

[38] R. P. N. Rao and D. L. Ruderman. Learning Lie groups for invariant visual perception. *Advances in Neural Information Processing Systems*, 11:810–816, 1999.

[39] S. Ravanbakhsh, J. Schneider, and B. Poczos. Equivariance Through Parameter-Sharing. *arXiv*, 1702.08389, feb 2017.

[40] J. J. Rodrigues, P. M. Q. Aguiar, and J. M. F. Xavier. ANSIG - An analytic signature for permutation-invariant two-dimensional shape representation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE, Jun. 2008.

[41] T. J. Sejnowski, P. K. Kienker, and G. E. Hinton. Learning symmetry groups with hidden units: Beyond the perceptron. *Physica D: Nonlinear Phenomena*, 22(1-3):260–275, Oct. 1986.

[42] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio. Robust object recognition with cortex-like mechanisms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3):411–426, Mar. 2007.

[43] L. Sifre and S. Mallat. Rotation, scaling and deformation invariant scattering for texture discrimination. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1233–1240. IEEE, Jun. 2013.

[44] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2014.

[45] D. Slepian. Group codes for the gaussian channel. *Bell System Technical Journal*, 47(4):575–602, Apr. 1968.

[46] S. Soatto. Steps Towards a Theory of Visual Information: Active Perception, Signal-to-Symbol Conversion and the Interplay Between Sensing and Control. *arXiv*, 1110.2053, oct 2011.

[47] S. Soatto and A. Chiuso. Visual Representations: Defining Properties and Deep Approximations. In *International Conference on Learning Representations (ICLR)*, 2016.

[48] J. Sohl-Dickstein, C. M. Wang, and B. A. Olshausen. An Unsupervised Algorithm For Learning Lie Group Transformations. *CoRR*, abs/1001.1, jan 2010.

[49] K. Sohn and H. Lee. Learning Invariant Representations with Local Transformations. In *International Conference on Machine Learning (ICML)*, 2012.

[50] T. Strohmer and R. W. Heath. Grassmannian frames with applications to coding and communication. *Applied and Computational Harmonic Analysis*, 14(3):257–275, May 2003.

[51] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015.

[52] A. Tacchetti, S. Voinea, and G. Evangelopoulos. Discriminate-and-Rectify Encoders: Learning from Image Transformation Sets. *CBMM Memo 062*, Mar. 2017.

[53] M. Thill and B. Hassibi. Group Frames with Few Distinct Inner Products and Low Coherence. *IEEE Transactions on Signal Processing*, 63(19):5222–5237, Oct 2015.

[54] I. Tosic and P. Frossard. Dictionary Learning. *IEEE Signal Processing Magazine*, 28(2):27–38, Mar. 2011.

[55] L. Toth. Combining time- and frequency-domain convolution in convolutional neural network-based phone recognition. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 190–194, May 2014.

[56] R. Vale and S. Waldron. Tight frames generated by finite nonabelian groups. *Numerical Algorithms*, 48(1-3):11–27, Mar. 2008.

[57] C. M. Wang, J. Shol-Dickstein, I. Tosic, and B. A. Olshausen. Lie group transformation models for predictive video coding. In *Data Compression Conference Proceedings*, pages 83–92. IEEE, Mar. 2011.

[58] E. Xing, A. Ng, M. Jordan, and S. Russell. Distance Metric Learning with Application to Clustering with Side-Information. In *Advances in Neural Information Processing Systems 15*, pages 505˜—-˜512, 2003.

[59] S. Zacks. *The Theory of Statistical Inference.* Wiley, 1971.

[60] C. Zhang, S. Voinea, G. Evangelopoulos, L. Rosasco, and T. Poggio. Discriminative template learning in group-convolutional networks for invariant speech representations. In *INTERSPEECH 2015, Annual Conference of the International Speech Communication Association*, pages 3229–3233, Dresden, Germany, 2015.

The Appendix is organized as follows:

- In Sec. A, we define the concept of multiplication tables of finite groups.
- In Sec. B we formulate an analytic regularization term for (a) in Sec. C and provide the analytic gradient for use in gradient descent algorithms.
- In Sec. C we give a proof of the conditions that enforce orbit structure in the representation weights, i.e. a) the covariance matrix of the weights should commute with the covariance of the data, and b) the Gramian matrix should be a permuted matrix.
- In Sec. E we prove a result showing how the permutation invariant metric can be used to test the validity of the learned solutions.
- In Sec. F we provide additional possible forms of the orbit regularizer, showing their link to maximal invariants. We also highlight a different approach than that of the main text, namely finding the *nearest Latin Square to the Gramian matrix.*
- In Sec. G we discuss the conditions under which the proposed approach could be extended to non-group transformations.
- In Sec. H we provide additional visual results on the solutions and the dependency on parameters.

Appendix A. Multiplication tables of finite groups

Definition A.1. *For any finite group $\mathcal{G}$ with composition $\circ$, a group (or Cayley) multiplication table is a table collecting the group composition of each element with all group elements.*

In the example of the cyclic group of order three (Example 1), each column is a permutation of the others. In particular the set of permutations form a group, a subgroup of the symmetric group of cardinality three. This is a general property of multiplication tables of finite groups:

Theorem A.1 (Cayley,[8])**.** *Every finite group $\mathcal{G}$ is isomorphic to a subgroup of the symmetric group of cardinality $|\mathcal{G}|$ acting on $\mathcal{G}$.*

Appendix B. Explicit form of orbit regularizer

B.1 Proof of Theorem 1

In the following, we give a proof of Theorem 1, restated here for the case of any matrix G. In addition, we provide the exact forms for the constant matrix C and vector t required for computing $r(W)$. We start with a definition:

Definition B.1 (Permuted matrix)**.** *A matrix is permuted iff all its columns are permutations of a single vector.*

Theorem 1 (Symmetry regularization)**.** *]Let matrix $W \in \mathbb{R}^{d\times|\mathcal{G}|}$ and $r : \mathbb{R}^{|\mathcal{G}|\times|\mathcal{G}|} \to \mathbb{R}_+$ with*

$$r(W^TW) = \tau^T\delta(Cvec(G)), \;\; G = W^TW$$

where G is the Gramian matrix, C is a constant matrix calculating all differences of the components of vector $vec(G) \in \mathbb{R}^{|\mathcal{G}|^2}$, the function δ acts elementwise ($\delta(a) = 1$ if $a = 0$, otherwise 0), and τ is a constant weight vector. If $r(W^TW) = 0$, then G is a permuted matrix and viceversa.

Proof. Let G be a permuted matrix. Since column i of is a permutation of column j then:

$$\int d\lambda \; (h_i(\lambda) - h_j(\lambda))^2 = 0 \tag{26}$$

where

$$h_i(\lambda;\sigma) = \sum_{k=1}^{|\mathcal{G}|} f(G_{ki}, \lambda; \sigma).$$

We use the Parzen window approximation of the delta function

$$f(G_{ki}, \lambda; \sigma) = e^{-\frac{(G_{ki}-\lambda)^2}{\sigma^2}}, \;\; \sigma \ll 1,$$

with $\delta(G_{ki} - \lambda) = \lim_{\sigma\to 0} f(G_{ki}, \lambda; \sigma) \approx f(G_{ki}, \lambda; \sigma)$. By extension, $\lim_{\sigma\to 0} h_i(\lambda;\sigma)$ counts the frequency of $G_{ki} = \lambda$.

We can therefore write the condition in Eq. (26) as

$$\lim_{\sigma\to 0} \int d\lambda \left(\sum_{k=1}^{|\mathcal{G}|} e^{-\frac{(G_{ki}-\lambda)^2}{\sigma^2}} - \sum_{k=1}^{|\mathcal{G}|} e^{-\frac{(G_{kj}-\lambda)^2}{\sigma^2}} \right)^2 = 0. \tag{27}$$

An explicit calculation for all the columns, in the limit for $\sigma \to \; 0$, leads to the condition

$$\sum_{i,j,p,q=1}^{|\mathcal{G}|} (|\mathcal{G}|\delta_{ij} - 1)\delta(G_{pi} - G_{qj}) = 0 \tag{28}$$

or its smooth approximation

$$\sum_{i,j,p,q=1}^{|\mathcal{G}|} (|\mathcal{G}|\delta_{ij} - 1)e^{-\frac{(G_{pi}-G_{qj})^2}{2\sigma^2}} = 0, \;\; \sigma \ll 1.$$

The condition can be rewritten in a compact way by vectorizing G:

$$\tau^T g_\sigma(C\mathrm{vec}(G)) = 0 \tag{29}$$

where $g_\sigma(x) = \exp(-x^2/\sigma^2)$, τ is the $|\mathcal{G}|^2(|\mathcal{G}|^2 - 1)/2 \times 1$ vector of weights $(|\mathcal{G}|\delta_{ij} - 1), i, j = 1, \ldots, |\mathcal{G}|$ and C is a $(|\mathcal{G}|^2 - 1)|\mathcal{G}|^2/2 \times |\mathcal{G}|^2$ sparse, constant matrix that encodes all pairwise differences in a right multiplied vector of size $|\mathcal{G}|^2 \times 1$, namely:

$$C = \left[\begin{array}{ccccccc} 1 & -1 & 0 & \dots & 0 & 0 & 0 \\ 1 & 0 & -1 & \dots & 0 & 0 & 0 \\ 1 & 0 & 0 & \dots & 0 & 0 & 0 \\ \multicolumn{7}{c}{\dotfill} \\ 0 & 0 & 0 & \dots & 1 & -1 & 0 \\ 0 & 0 & 0 & \dots & 1 & 0 & -1 \\ 0 & 0 & 0 & \dots & 0 & 1 & -1 \end{array}\right]$$

Since h is a maximal invariant, the converse is also true. □

Appendix C. Proof of Theorem 2 (Orbit condition)

Theorem 2 (Symmetry adapted regularization)**.** *Let $X \in \mathbb{R}^{d\times Q|\mathcal{G}|}$ a matrix whose columns are all elements from the union of orbits of Q vectors in $\mathbb{R}^d$ w.r.t. a finite group $\mathcal{G} = \{g_i,\ i = 1,\dots,|\mathcal{G}|\}$, where $g_i \in \mathbb{R}^{d\times d}$ is a matrix representation of the group action and $|\mathcal{G}|$ indicates the cardinality of the group. Let $W \in \mathbb{R}^{d\times|\mathcal{G}|}$ a matrix whose columns are the elements of an orbit of a vector $w \in \mathbb{R}^d$ w.r.t. a finite group $\tilde{\mathcal{G}}$ with $|\mathcal{G}| = |\tilde{\mathcal{G}}|$. Then W^TW is a permuted matrix. Further if $[XX^T, WW^T] = 0$ then $\tilde{\mathcal{G}} = \mathcal{G}$ (up to a constant unitary matrix conjugation).*

Let $X = (x_1,\dots,x_N) \in \mathbb{R}^{d\times N}$ the matrix storing the training set S_N, which follows Assumption 1, i.e. that the columns of X are a (randomly permuted) union of Q orbits generated by a finite group $\mathcal{G}$, with cardinality $|\mathcal{G}|$ and $N = Q|\mathcal{G}|$. In the following, we will denote by $g_i \in \mathcal{G} \subset \mathbb{R}^{d\times d}$, $i = 1,\dots,|\mathcal{G}|$ a matrix representation of the group action.

The point that W^TW is a permuted matrix for a finite group $\mathcal{G}$ follows from Cayley's theorem: The Gramian matrix $G = W^TW$ is a function of the multiplication table of $\mathcal{G}$, as each column $(G)_{:j} = W^Tw_j \in \mathbb{R}^{|\mathcal{G}|}$ entries form the set $\{\langle t, g_i^* g_j t\rangle\}$, $\forall i$ (or row $(G)_{i:}$ is the set $\left\{\left\langle g_j^* g_i t, t\right\rangle\right\}$, $\forall j$. By Theorem A.1, these sets are permutations corresponding to the subgroup of the symmetric group of $|\mathcal{G}|$ symbols. Then each column vector is a permutation of some vector, and G is a permuted matrix.

We now turn to proving the second part of Theorem 2, as stated in Lemma 1. We start with two preparatory lemmas: the first proves that if a vector is an eigenvector of XX^T then the elements of its $\mathcal{G}-$orbit are eigenvectors with the same eigenvalue; the second proves that the linear span of an orbit of an eigenvector with eigenvalue λ coincides with the eigenspace associated to λ.

Lemma C.1. *If X is a matrix whose column set is the union of orbits from $\mathcal{G}$ (as above), then $[XX^T, g_i] = 0$, $\forall i = 1,\dots,|\mathcal{G}|$. Moreover, if $v \in \mathbb{R}^d$ is an eigenvector of XX^T with eigenvalue λ, the set $\{g_iv, i = 1,\dots,|\mathcal{G}|\}$ is a set of eigenvectors with the same eigenvalue.*

Proof. Since the columns of X are the union of orbits of the same group $\mathcal{G}$ it is easy to prove (21), i.e.:

$$XX^T = \sum_{j=1}^{|\mathcal{G}|} g_j TT^T g_j^T$$

where T is the $d \times Q$ matrix of representatives for each orbit, composed by a single, arbitrary element from each orbit in X. Multiplying the right hand side by any element of $\mathcal{G}$ we have

$$XX^T g_i = \sum_{j=1}^{|\mathcal{G}|} g_j TT^T g_j^T g_i = g_i \sum_{s=1}^{|\mathcal{G}|} g_s TT^T g_s^T = g_i XX^T$$

where we used the change of variables $g_s = g_i^T g_j$ and the closure property of the group. Thus

$$[XX^T, g_i] = 0, \quad \forall i = 1, \dots, |\mathcal{G}|$$

This also implies that for any $v \in \mathbb{R}^d$ such that $XX^T v = \lambda v$

$$XX^T g_i v = g_i XX^T v = \lambda g_i v, \quad \forall i = 1, \dots, |\mathcal{G}|$$

i.e. all elements of the orbit of v, $O_v = \{g_i v, \forall i = 1, \dots, |\mathcal{G}|\}$, are eigenvectors of XX^T with eigenvalue λ. □

Lemma C.2. *Let E_λ the eigenspace of XX^T associated to the eigenvalue λ and v an arbitrary vector in E_λ. Then the linear span of the orbit of v coincides with the eigenspace, i.e.* $\mathrm{span}(O_v) = E_\lambda$.

Proof. Let $B = \{b_i, \;\; i = 1, \dots, M\}$ an orthogonal basis in E_λ with $M = \dim(E_\lambda)$. Thus any eigenvector $w \in E_\lambda$ can be expressed as a linear combination of the basis elements, i.e.

$$v = \sum_{i=1}^{M} \alpha_i b_i,$$

and an element of its $\mathcal{G}$-orbit O_v as:

$$g_j v = \sum_{i=1}^{M} \alpha_i g_j b_i, \quad \forall j = 1, \dots, |\mathcal{G}|.$$

Thus $g_j w$ is a linear combination of eigenvectors since each $g_j b_i$ is an eigenvector (by Lemma C.1). Clearly this holds also for any linear combination of the orbit elements i.e. we have $\mathrm{span}(O_v) \subseteq E_\lambda$. Note that for any choice of $u, v \in E_\lambda$:

$$\mathrm{span}(O_u) = \mathrm{span}(O_v)$$

since both $\mathrm{span}(O_u)$ and $\mathrm{span}(O_v)$ consist of all linear combinations of the set $\{g_j b_i\}$ and thus coincide. This implies:

$$\bigcup_{u \in E_\lambda} \mathrm{span}(O_u) = \mathrm{span}(O_v).$$

However:

$$E_\lambda \subseteq \bigcup_{u \in E_\lambda} \mathrm{span}(O_u) = \mathrm{span}(O_v) \subseteq E_\lambda$$

which implies that $\mathrm{span}(O_v) = E_\lambda$ for any choice of $v \in E_\lambda$. □

We can now prove Lemma 1 in the main text, which we re-state here:

Lemma 1. *Let W be an orbit w.r.t. a finite group $\tilde{\mathcal{G}}$ and $X = (x_1, \ldots, x_N) \in \mathbb{R}^{d\times N}, x_i \in S_N$ as in (10) with $|\mathcal{G}| = |\tilde{\mathcal{G}}|$. If $[XX^T, WW^T] = \mathbf{0}$ then $\tilde{\mathcal{G}} = \mathcal{G}$ (up to a constant unitary matrix conjugation).*

Proof. Note that XX^T, WW^T are Hermitian matrices. If two Hermitian matrices commute they have the same eigenspaces [26]. Let E_λ be one such eigenspace. From Lemma C.1 we have $E_\lambda = \tilde{\mathcal{G}}E_\lambda$ and $E_\lambda = \mathcal{G}E_\lambda$ for the two groups. Further, since E_λ is the minimal invariant subspace by Lemma C.2, it follows that the action representation of $\mathcal{G}$ and $\tilde{\mathcal{G}}$ restricted to E_λ is irreducible. From the fact that irreducible representations are equivalent up to a unitary transformation, we can deduce that $\mathcal{G} = \tilde{\mathcal{G}}$, up to a fixed unitary transformation. □

Appendix D. Analytic gradients for gradient-based learning

Regularizer gradient We calculate the derivative of $r(W) = \tau^T g_\sigma(C\text{vec}(W^TW))$ with respect to a vectorized form of W; using the chain rule we have:

$$\frac{\partial r(W)}{\partial \text{vec}(W)} = \left(\frac{\partial \langle \tau, p\rangle}{\partial p}\frac{\partial p}{\partial q}\frac{\partial q}{\partial s}\frac{\partial s}{\partial \text{vec}(W)}\right)^T \tag{30}$$

with $p = g_\sigma(q)$, $q = Cs$ and $s = \text{vec}(W^TW)$. The first three factors are easy to calculate and we are left with

$$\frac{\partial r(W)}{\partial \text{vec}(W)} = -\frac{2}{\sigma^2}\left(\tau^T \text{diag}\big((C\text{vec}(W^TW)) \odot g_\sigma(C\text{vec}(W^TW))\big)C\frac{\partial\ \text{vec}(W^TW)}{\partial \text{vec}(W)}\right)^T \tag{31}$$

where $\odot$ denotes the Hadamard product and diag$(\cdot)$ a $(|\mathcal{G}|^2(|\mathcal{G}|^2-1)/2) \times (|\mathcal{G}|^2(|\mathcal{G}|^2-1)/2)$ diagonal matrix. For the last partial derivative we have ([15]):

$$\frac{\partial \text{vec}(W^TW)}{\partial \text{vec}(W)} = (\mathbb{I}_{|\mathcal{G}|^2} + T)(\mathbb{I}_{|\mathcal{G}|} \otimes W^T) = R(\mathbb{I}_{|\mathcal{G}|} \otimes W^T) \tag{32}$$

where the matrix T is defined as $T\text{vec}(v) = \text{vec}(v^T)$, $\mathbb{I}_{|\mathcal{G}|^2}$, $\mathbb{I}_{|\mathcal{G}|}$ are the identity matrices of dimensions $|\mathcal{G}|^2 \times |\mathcal{G}|^2$, $|\mathcal{G}| \times |\mathcal{G}|$, and $R = \mathbb{I}_{|\mathcal{G}|^2} + T$. It can be proven that T can be written explicitly as:

$$\begin{cases} T_{ij} = 1 \; if \; j = 1 + |\mathcal{G}|(i-1) - (|\mathcal{G}|^2 - 1)\text{floor}\big(\frac{i-1}{|\mathcal{G}|}\big) \\ T_{ij} = 0 \text{ otherwise.} \end{cases} \tag{33}$$

Putting together all the pieces we finally have:

$$\begin{aligned} \frac{\partial r(W)}{\partial \text{vec}(W)} &= -\frac{2}{\sigma^2}(\mathbb{I}_{|\mathcal{G}|} \otimes W)P^T \text{diag}\left((C\text{vec}(W^TW)) \odot g_\sigma(C\text{vec}(W^TW))\right)^T \tau \\ &= -\frac{2}{\sigma^2}(\mathbb{I}_{|\mathcal{G}|} \otimes W)P^T (C\text{vec}(W^TW)) \odot g_\sigma(C\text{vec}(W^TW)) \odot \tau \end{aligned} \tag{34}$$

with $P = CR$ a constant matrix, which can be pre-calculated given the (fixed) group cardinality.

Commutator gradient: It is easy to derive the analytic gradient of the commutator term, i.e. the norm $\|c(X,W)\|_F^2 = \text{trace}(C(X,W)^T C(X,W))$ in (22), (25), where $C(X,W) = [XX^T, WW^T]$ is the commutator of the data and weight covariance matrices. The gradient with respect to W has the analytic form

$$\begin{aligned}
&\frac{\partial \|C(X,W)\|_F^2}{\partial W} = \frac{\partial}{\partial W}(-2Tr(XX^TWW^TXX^TWW^T) + 2Tr(XX^TWW^TWW^TXX^T))\\
&= -8XX^TWW^TXX^T + 4XX^TXX^TWW^TW + 4WW^TXX^TXX^TW\\
&= -4[XX^T, WW^T]XX^TW + 4XX^T[XX^T, WW^T]W\\
&= -4[[XX^T, WW^T], XX^T]W = -4[C(X,W), XX^T]W.
\end{aligned}$$

Appendix E. Pseudometric and solution testing

Lemma 2 (Pseudometric)**.** *If W is a solution of Eq. (22), i.e. it satisfies Theorem 2 conditions, then:*

$$r(W^T(x,x')) = 0 \iff x = gx' \quad \forall\, x, x' \in \mathbb{R}^d,\ g \in\ \mathcal{G}.$$

Proof. For the direct implication note that the Gramian matrix W^TW identifies W up to an arbitrary (but fixed) unitary transformation U; together with the regularization constrain $r(W^TW) = 0$ we thus have that W can be written as $W = UW_o$ where W_o is an orbit matrix such that $W_o^TW_o = W^TW$. Now if $r((\Phi(x), \Phi(x'))) = 0$ then

$$W_o^TU^Tx = PW_o^TU^Tx' = W_o^Tg^TU^Tx'$$

since the action of g on W is a permutation of its columns i.e. $gW = WP_g$. Thus calling $\tilde{x} = U^Tx$ and $\tilde{x}' = U^Tx'$ we have:

$$W_o^T\tilde{x} = W_o^Tg\tilde{x}'.$$

The last expression holds for any $x = U\tilde{x}, x' = U\tilde{x}'$ such that $r(W^T(x,x')) = 0$. This implies $\tilde{x} = g\tilde{x}'$ or equivalently $x = gx'$ since the rotation U of the signal space maps the space into itself $\tilde{\mathcal{X}} = U^T\mathcal{X} = \mathcal{X}$ (and is irrelevant from the learning point of view).

For the inverse implication we proceed by contradiction. Suppose $x = gx'$ but $r((\Phi(x), \Phi(x'))) \neq 0$. Thus for any permutation P

$$W^Tx \neq PW^Tx'.$$

Note now that the learned matrix W is an orbit of vector $\tilde{t}$ w.r.t. the conjugate representation of $\mathcal{G}$ induced by U, i.e. $W = \tilde{W}_o$. In fact we have

$$(W)_{:,i} = (UW_o)_{:,i} = Ug_it = Ug_iU^TUt = \tilde{g}_i\tilde{t}$$

and thus

$$\tilde{W}_o^Tx \neq P\tilde{W}_o^Tx' = \tilde{W}_o^T\tilde{g}x' \ \ \forall\, g \in \tilde{\mathcal{G}}.$$

Since $\mathcal{X} = \tilde{\mathcal{X}}$ we also have that (it simply corresponds to a fixed change of the orbit seeds)

$$\tilde{W}_o^T\tilde{x} \neq \tilde{W}_o^T\tilde{g}\tilde{x}' \ \ \forall\, \tilde{g} \in \tilde{\mathcal{G}}.$$

which implies $\tilde{x} \neq \tilde{g}\tilde{x}' \; \forall \; \tilde{g} \in \tilde{\mathcal{G}}$ i.e.

$$Ux \neq UgU^T Ux' \;\Rightarrow\; x \neq gx' \;\; \forall \; g \in \mathcal{G}$$

a contradiction. □

Appendix F. Alternative expressions for the regularizer

In Sec. B.1 we calculated the expression for Eq. (18) when the maximal permutation invariant is the distribution function, i.e. $f_\lambda = \delta(\cdots - \lambda)$. In the following we provide the expressions for:

1. $f_\lambda(\cdot) = (\cdot)H(\cdot - \lambda)$ or $H(\cdot - \lambda)$, H is the Heaviside step function: the corresponding maximal invariant corresponds to the calculation of the ordered statistics (see theorem below) or, in the second case, to the *cumulative distribution function* of the gramian rows (columns).
2. $f_\lambda(\cdot) = (\cdot)^\lambda, \; \lambda \in \mathbb{N}$: the corresponding maximal invariant corresponds to the calculation of the moment function of the gramian rows (columns).

Another maximal invariant, $f_\lambda(\cdot) = e^{\cdot\lambda}$, was explored in [40] and used for shape recognition. Note that any bijective function of a maximal invariant would also work, being itself a maximal invariant.

F.1 Maximal invariant: cumulative distribution function

We start with $f_\lambda(\cdot) = (\cdot)H(\cdot - \lambda)$, using:

Theorem F.1 (Corollary of Hardy and Littlewood (1929))**.** *Two vectors u, v are equal up to a permutation iff*

$$\sum_{i=1}^{|\mathcal{G}|} |u_i - a|_+ = \sum_{i=1}^{|\mathcal{G}|} |v_i - a|_+, \; \forall \; a \in \mathbb{R}$$

where $|x - a|_+ = xH(x - a)$.

Thus the condition that all matrix columns (or rows) are permutations of a single vector reads as:

$$\sum_{k=1}^{|\mathcal{G}|} |G_{ik} - a|_+ = \sum_{k=1}^{|\mathcal{G}|} |G_{jk} - a|_+, \; \forall \; a \in \mathbb{R}, \text{ fixed } i, \forall \; j \neq i \tag{35}$$

where G is the Gramian matrix associated to the frame.

Theorem F.2. *Let $\mathcal{G}$ a finite group, W a set of $|\mathcal{G}|$ vectors in $\mathbb{R}^d$ and let $G = W^T W$ the associated Gramian. If $-1 \leq G_{ij} \leq 1$ i.e. if W is a set of unit-norm vectors and W is an orbit of a single vector w.r.t. $\mathcal{G}$ then*

$$\sum_{ijkl=1}^{|\mathcal{G}|} (|\mathcal{G}|\delta_{i,j} - 1) G_{ik} G_{jl} (1 + \min(G_{ik}, G_{jl})) = 0. \tag{36}$$

Proof. Starting from eq. (35) we can equivalently restate the condition as

$$\int_{-1}^{1} da \sum_{ij=1}^{|\mathcal{G}|} \Big(\sum_{k=1}^{|\mathcal{G}|} |G_{ik} - a|_+ - \sum_{k=1}^{|\mathcal{G}|} |G_{jk} - a|_+ \Big)^2 = 0 \tag{37}$$

We can write

$$\begin{aligned} &\int_{-1}^{1} da \sum_{ij=1}^{|\mathcal{G}|} \Big(\sum_{k=1}^{|\mathcal{G}|} |G_{ik} - a|_+ - \sum_{k=1}^{|\mathcal{G}|} |G_{jk} - a|_+ \Big)^2 = \\ = &\int_{-1}^{1} da \sum_{ij=1}^{|\mathcal{G}|} \Big(\sum_{k=1}^{|\mathcal{G}|} G_{ik} H(G_{ik} - a) \Big)^2 + \sum_{ij=1}^{|\mathcal{G}|} \Big(\sum_{k=1}^{|\mathcal{G}|} G_{jk} H(G_{jk} - a) \Big)^2 - \\ &\sum_{ijkl=1, i\neq j}^{|\mathcal{G}|} G_{ik} G_{jl} H(G_{ik} - a) H(G_{jl} - a) \end{aligned}$$

The first and second term in the second line can both be rewritten as

$$\sum_{ijkl=1}^{|\mathcal{G}|} G_{ik} G_{il} \int_{-1}^{1} da\, H(G_{ik} - a) H(G_{il} - a) = \sum_{ijkl=1}^{|\mathcal{G}|} G_{ik} G_{il} (1 + \min(G_{ik}, G_{il}))$$

The third term similarly

$$\sum_{ijkl=1, j\neq i}^{|\mathcal{G}|} \int_{-1}^{1} da\, G_{ik} G_{jl} H(G_{ik} - a) H(G_{jl} - a) = \sum_{ijkl=1, j\neq i}^{|\mathcal{G}|} G_{ik} G_{jl} (1 + \min(G_{ik}, G_{jl})) \tag{38}$$

The condition in Eq.(38) can be therefore compactly written as:

$$\sum_{ijkl=1}^{|\mathcal{G}|} (|\mathcal{G}|\delta_{i,j} - 1) G_{ik} G_{jl} (1 + \min(G_{ik}, G_{jl})) = 0. \tag{39}$$

□

If we choose $f_\lambda(\cdot) = H(\cdot - \lambda)$ (cumulative function) the result of the calculations above will be simply:

$$\sum_{ijkl=1}^{|\mathcal{G}|} (|\mathcal{G}|\delta_{i,j} - 1)(1 + \min(G_{ik}, G_{jl})) = 0. \tag{40}$$

Using $min(x, y) = (1/2)(x + y - |x - y|)$ we have

$$\sum_{ijkl=1}^{|\mathcal{G}|} \frac{|\mathcal{G}|\delta_{i,j} - 1}{2} (2 + G_{ik} + G_{jl} - |G_{ik} - G_{jl}|) = \sum_{ijkl=1}^{|\mathcal{G}|} \frac{1 - |\mathcal{G}|\delta_{i,j}}{2} |G_{ik} - G_{jl}| = 0. \tag{41}$$

or, in a compact form as

$$r(W) = \tau^T |C \mathrm{vec}(W^T W)|.$$

F.2 Maximal invariant: moments

Another way to calculate a maximal invariant w.r.t. the permutation group is to consider all ℓ_p norms of the row vectors and impose their equality. For example if we have two vectors, v, u (two rows of the gramian) the condition gives:

$$\begin{cases} u_1 + \dots + u_d = v_1 + \dots + v_d \\ u_1^2 + \dots + u_d^2 = v_1^2 + \dots + v_d^2 \\ \dots \\ u_1^k + \dots + u_d^k = v_1^k + \dots + v_d^k \end{cases}$$

The set of equalities above implies that the polynomials $p(t) = (t - u_1)(t - u_2) \dots (t - u_d)$ and $r(t) = (t - v_1)(t - v_2) \dots (t - v_d)$ are identical (see [26]). In particular, $p(t)$ and $q(t)$ share the same system of roots (including their multiplicities) thus we conclude that the vectors u and v are equal, up to a permutation. Note that this strategy correspond to the choice $f_\lambda(\cdot) = (\cdot)^\lambda,\ \lambda \in \mathbb{R}_+$.

Therefore an alternative condition of that of eq (38) can be written as

$$\sum_{pij} \Big(\sum_{k=1}^{|\mathcal{G}|} G_{ik}^p - \sum_{k=1}^{|\mathcal{G}|} G_{jk}^p \Big)^2 = 0 \tag{42}$$

Following the same steps of the previous section we have

$$\sum_{pij=1}^{|\mathcal{G}|} \Big(\sum_{k=1}^{|\mathcal{G}|} G_{ik}^p - \sum_{k=1}^{|\mathcal{G}|} G_{jk}^p \Big)^2 = \sum_{pijkl=1}^{|\mathcal{G}|} (|\mathcal{G}|\delta_{ij} - 1) G_{ik}^p G_{lj}^p \tag{43}$$

If we consider normalized dictionaries the sum above is not convergent since all terms of the form $G_{ii}G_{jj} = 1$ will lead to infinities. The idea is, from an optimization point of view, to fix a priory the gramian elements that lead to infinities (that we impose to be equal to one) and optimize over the rest. In this case summing over the p powers gives:

$$\sum_{ijkl=1, i\neq k \wedge j \neq l}^{|\mathcal{G}|} \frac{|\mathcal{G}|\delta_{i,j} - 1}{1 - G_{ik}G_{jl}} \tag{44}$$

an expression that seems easier to minimize then that in eq. (36). It can be written in a compact for as

$$r(W^T W) = j^T (M \odot (J - \mathrm{vec}(G)\mathrm{vec}(G)^T)^{-1}) j \tag{45}$$

where J is the all ones matrix, j is the all ones vector and M is a constant matrix encoding the weights $1 - G_{ik}G_{lj}$.

F.3 Common eigenspace interpretation of the condition

In the following we demonstrate that the condition in eq. (35) is equivalent to $[|G|_+^a, J] = \mathbf{0},\ \forall\, a \in \mathbb{R}$ where J is the all ones matrix. In particular we prove

Theorem F.3. *Let W a set of vectors in $\mathbb{R}^d$. If W is an orbit of some vector $t \in \mathbb{R}^d$ w.r.t. a finite group $\mathcal{G}$ then*

$$[|G|_+^a, J] = \mathbf{0} \; \forall \; a \in \mathbb{R} \; \textit{ or equivalently } \; \int da \; \left\|[|G|_+^a, J]\right\|_F^2 = 0,$$

where $[\cdot,\cdot]$ indicates the commutator and $|G|_+^a = |G-a|_+$. The condition above can be also stated as $|G|_+^a j = \lambda_a j$, $\forall \; a \in \mathbb{R}$ where j is the all ones vector and $\lambda_a \in \mathbb{R}$.

Proof. Note that

$$\sum_{k=1}^{|\mathcal{G}|} |G_{ik} - a|_+ = \mathbf{1}^T |G|_+^a e_i$$

where e_i is the i^{th} canonical vector. We can rewrite the condition of eq. (35) as:

$$\int da \; \sum_{ij=1}^{|\mathcal{G}|} (j^T |G|_+^a (e_i - e_j))^2 = \int da \; \sum_{ij=1}^{|\mathcal{G}|} j^T |G|_+^a (e_i - e_j)(e_i - e_j)^T |G|_+^a j.$$

Being

$$\sum_{ij=1}^{|\mathcal{G}|} (e_i - e_j)(e_i - e_j)^T = 2|\mathcal{G}|\mathbb{I} - 2J$$

(where $\mathbb{I}$ is the identity matrix and $|\mathcal{G}|$ the group cardinality) we can rewrite

$$\int da \; j^T |G|_+^a (2|\mathcal{G}|\mathbb{I} - 2J)|G|_a^+ j = \int da \; tr(|G|_+^a (2|\mathcal{G}|\mathbb{I} - 2J)|G|_+^a J). \tag{46}$$

Using the identity $JJ = J^2 = |\mathcal{G}|J$ we can rewrite eq. (46) as:

$$tr(|G|_+^a (2|\mathcal{G}|\mathbb{I} - 2J)|G|_+^a J) = 2tr((|G|_+^a)^2 J^2) - 2tr((|G|_+^a J)^2).$$

Note now that for any two matrices A, B:

$$2tr[(AB)^2] - 2tr(A^2B^2) = tr(ABAB + BABA - ABBA - BAAB) = tr[(AB - BA)^2].$$

Thus we can write

$$2tr((|G|_+^a)^2 J^2) - 2tr((|G|_+^a J)^2) = -tr((|G|_+^a J - J|G|_+^a)^2).$$

Finally note that being $|G|_+^a J - J|G|_+^a = -(|G|_+^a J - J|G|_+^a)^T$, we have

$$tr(|G|_+^a (2|\mathcal{G}|\mathbb{I} - 2J)|G|_+^a J) = tr((|G|_+^a J - J|G|_+^a)(|G|_+^a J - J|G|_+^a)^T) = \left\|[|G|_+^a, J]\right\|_F^2.$$

The condition for the Gramian rows to be the permutations of a single vector is therefore:

$$\int da \; \left\|[|G|_+^a, J]\right\|_F^2 = 0. \tag{47}$$

□

Remark F.1. *A similar condition can be found if we use the Hadamard powers nonlinearity instead of the threshold function:*

$$\sum_{p=1}^{\infty} \; \|[G^p, J]\|_F^2 = 0. \tag{48}$$

F.4 Nearest Latin square

In this section we propose an alternative approach to that of the paper that make use of Latin Squares. We start with the definition of a Latin Square.

Definition F.1 (Latin square)**.** *Latin square of dimension k is an $k \times k$ array filled with k different symbols, each occurring exactly once in each row and exactly once in each column.*

In particular we consider Latin squares which are real matrices. It is well known that Latin Squares are multiplication tables of finite *quasigroups* and viceversa. Quasigroups, loosely speaking, are groups without the associativity property. However, in the paper, we only focus on groups and therefore we are interested on the subset of Latin Squares that are multiplication tables of finite groups. Those are sometimes called *Cayley tables.*

In here we formulate a minimization problem to find the nearest Latin square to a given matrix, which, in our setting, is a Gramian $G = W^T W$ of an unknown dictionary W.

Any Latin square L composed by $|\mathcal{G}|$ symbols $s_1, \ldots, s_{|\mathcal{G}|}$ can be written as

$$L = s_1 P_1 + \cdots + s_{|\mathcal{G}|} P_{|\mathcal{G}|} \tag{49}$$

with each P_k a permutation matrix, i.e., $\{P_k, k = 1, \ldots, |\mathcal{G}|\} \in \mathcal{P}$ is a set of permutation matrices. The fact that each symbol is occurring exactly once in each row and exactly once in each column is equivalent to imposing:

$$P_1 + \cdots + P_{|\mathcal{G}|} = J \tag{50}$$

where J is the $|\mathcal{G}| \times |\mathcal{G}|$ all-ones matrix. For example the simplest Latin square composed by two symbols $\{1, 2\}$ can be decomposed as:

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} = 1 \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + 2 \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

In our setting the list of symbols is given by, e.g., the first column (or any column for that matter) of the Gramian, $G_{k,1},\ k = 1, \ldots, |\mathcal{G}|$. Therefore finding the nearest Latin square to a given G can be equivalently posed as the solution to the minimization problem:

$$\begin{aligned} &\underset{\{P_k, k=1,\ldots,|\mathcal{G}|\} \in \mathcal{P}}{\arg\min} \quad \left\| \sum_{k=1}^{|\mathcal{G}|} G_{k,1} P_i - G \right\|_F^2 \\ &\text{subject to} \quad \sum_{k=1}^{|\mathcal{G}|} P_k = J \end{aligned} \tag{51}$$

where $\|\cdot\|_F$ is the Frobenius norm and $\mathcal{P} \in \mathbb{R}^{|\mathcal{G}| \times |\mathcal{G}|}$ is the set of $|\mathcal{G}| \times |\mathcal{G}|$ permutation matrices.

The combinatorial nature of minimizing over permutation matrices makes the problem (51) NP-complete. We can relax it by working on the set of convex combinations of permutations (called permutahedron or Birkhoff polytope). This convex set, whose vertices are the $|\mathcal{G}| \times |\mathcal{G}|$

permutation matrices, coincides (Birkhoff - von Neumann Theorem) with the set of doubly stochastic matrices defined as:

$$\mathcal{M}_k \triangleq \{M \in \mathbb{R}^{|\mathcal{G}|\times|\mathcal{G}|} \mid Mj = M^T j = j,\ M \geq 0\} \tag{52}$$

where j is a vector of all ones. A doubly stochastic matrix is a square matrix of nonnegative real numbers, each of whose rows and columns sums to one.

In general the solution of Eq. (51) within the set of matrices defined in Eq. (52) will be in $\mathcal{M}_k$. However notice that a doubly stochastic matrix with maximal sparsity, i.e. a single unit entry in each row and each column is a permutation. Thus, in order to find a strategy to force the solution to be in $\mathcal{P}$ as prescribed by Eq. (49), we can add a penalty on the sparsity of the individual permutation matrices to obtain:

$$\begin{aligned} &\underset{\{M_k, k=1,\dots,|\mathcal{G}|\}\in\mathcal{M}_k}{\arg\min} \quad \left\| \sum_{k=1}^{|\mathcal{G}|} G_{k,1} M_k - G \right\|_F^2 + \lambda \sum_{k=1}^{|\mathcal{G}|} \|M_k\|_0 \\ &\text{subject to} \quad \sum_{k=1}^{|\mathcal{G}|} M_k = J,\ M_k \mathbf{1} = M_k^T \mathbf{1} = \mathbf{1},\ M \geq 0 \end{aligned} \tag{53}$$

and use to the ℓ_0 pseudo-norm through the ℓ_1 norm. By letting λ grow we can drive solutions with maximal sparsity for the corresponding matrices M_k.

Remark F.2. *Alternatively we can exploit the fact that $\mathcal{P} = \mathcal{M}_k \cap \mathcal{O}$ (where $\mathcal{O}$ is the set of orthogonal matrices) and that a doubly stochastic matrix of Frobenius norm $\sqrt{|\mathcal{G}|}$ is necessarily orthogonal. Thus we can add the penalty term $\lambda \sum_{k=1}^{|\mathcal{G}|} (\|M_k\|_F - \sqrt{|\mathcal{G}|})^2$ and force the solution to be a set of permutations.*

Empirically adding both penalty terms, orthogonality and sparsity, improves the convergence.

Appendix G. Approximate invariance for non-group transformations

In this section we briefly discuss extensions of this work for getting an approximately invariant signature for transformations that do not have a group structure. In fact, most realistic signal transformations will not have a group structure. However assuming that the transformation defines a smooth manifold we have (by the theory of Lie manifolds) that locally a Lie group is defined by the generators on the tangent space. We illustrate this in a simple example.

Let $x \in \mathcal{X} \subseteq \mathbb{R}^d$ and $s : \mathbb{R}^d \times \mathbb{R}^P \to \mathbb{R}^d$ a C^∞ transformation depending on $\Theta = (\theta_1, \dots, \theta_P)$ parameters. For any fixed $x \in \mathcal{X}$ the set $M = (s(x, \Theta),\ \Theta \in \mathbb{R}^P)$ describe a differentiable manifold. If we expand the transformation around e.g. $\vec{0}$ we have:

$$s(x,\Theta) = s(x,\vec{0}) + \sum_{i=1}^{P} \frac{\partial s(x,\Theta)}{\partial \theta_i} \theta_i + o(\|\Theta\|^2) = x + \sum_{i=1}^{P} \theta_i L_{\theta_i}(x) + o(\|\Theta\|^2) \tag{54}$$

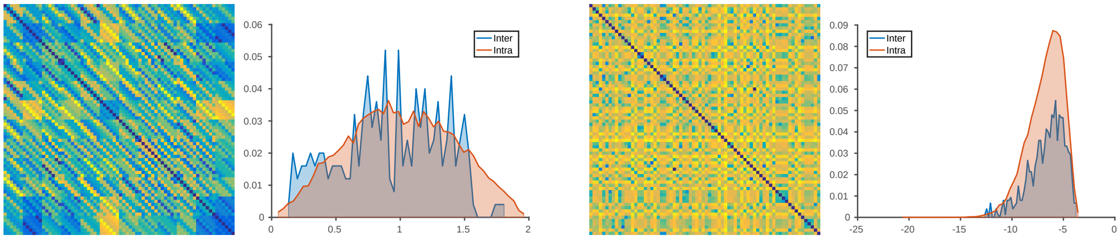
where L_{θ_i} are the infinitesimal generators of the transformation in the i^{th} direction.

Therefore locally (when the term $o(\|\Theta\|^2)$ can be neglected) the associated group transformation can be expressed by exponentiation as:

$$g(\Theta) = \exp(\theta_1 L_{\theta_1} + \theta_2 L_{\theta_2} + \dots + \theta_P L_{\theta_P}).$$

In other words instead of a global group structure of the transformation we will have a collection of local transformations that obey a group structure. Thus in this light the local learned weights will be orbits w.r.t. the local group approximating the non-group global transformation.

Appendix H. Additional experimental results



(a) $\langle x_i, x_j \rangle$

(b) random W

Figure 6: **Cyclic group C_6 (order 6)**: (a) cosine distance in the input space $\mathcal{X}$ and (b) permutation test $r(\Phi(x_i), \Phi(x_j))$, where $\Phi(x_i) = W^T x_i$ and W a 6×6 dictionary obtained from (b) uniformly at random sampled vectors in the unit ball. Data from Fig. 3.

H.1 Example solutions and Gramians

Minimization of Eq. (25) was initialized M times to produce M orbit solutions for each value of β, corresponding possibly to different local minima of the loss function. A possible ordering of the solutions can be achieved by pooling, in our case using L_2 on the representation coefficients and checking the variance of the inter-orbit distribution distance on an known orbit set. Small variance denotes a better approximation of the underlying symmetry in the data as the resulting representation gives inter-orbit distances that are very close (approximating invariance due to the group averaging operation). Figures 7, 9 and 11 show 20, 10, and 6 generated solutions for C_6, D_6 and T_h respectively, when using $M = 50$ and 8 values for β as above. Solutions are ordered based on their intra-orbit variance, using the validation set intra-orbit membership, assumed known just for evaluation purposes. For the cyclic group, the symmetries in many of the solutions might be easy to spot. Their corresponding Gramian matrices (Figs. 8, 10 and 12), can be seen by close inspection to have a permutation structure, again in some cases more obvious than in others.

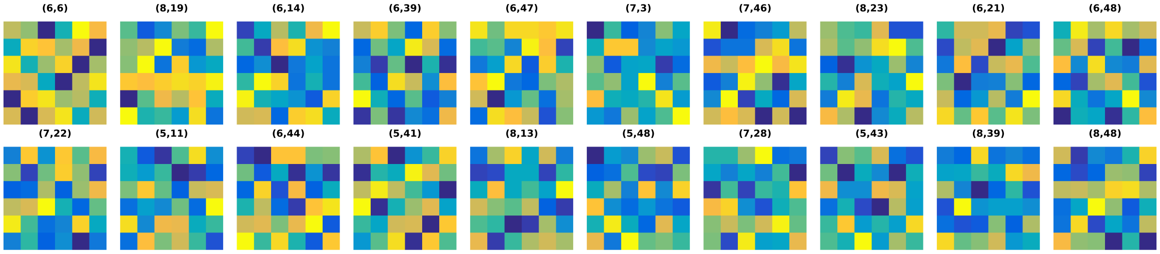


Figure 7: **Cyclic group (order 6)**: Learned orbits (20 shown), arranged with increasing variance of the intra-orbit signature distance. Each matrix W, corresponding to one value of β and one initialization is a 6×6 array of column-wise arranged dictionary elements. The entries $(\cdot, \cdot)$ denote β and solution index pairs, with $\log_{10}(\beta) = \{0, \dots, -7\}$ and $s = \{1, \dots, 50\}$; $s = 1$ corresponds to the solution with minimum objective function. Learning with $J = 100$ ReLU regularizers and $r(W^T W)$.

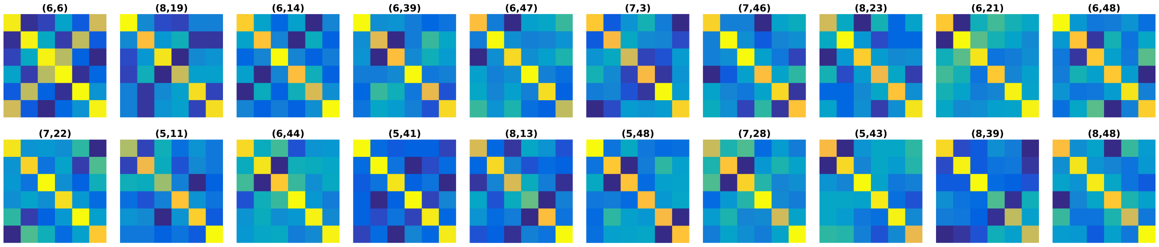


Figure 8: **Cyclic group (order 6)**: Gramian matrices $G = W^T W$ corresponding to the ordered solutions.

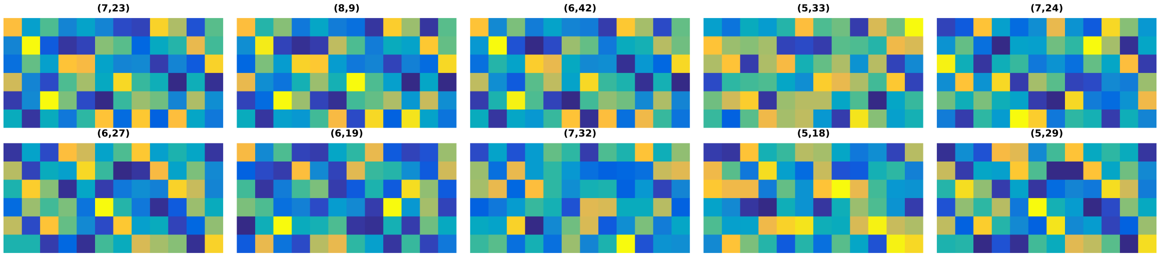


Figure 9: **Dihedral group (order 12)**: Solution orbits (10 shown), arranged with increasing variance of the intra-orbit signature distance. Each matrix W, corresponding to one solution and one value of β is a 6×12 array of column-wise arranged dictionary elements. The entries $(\cdot, \cdot)$ denote β and solution index pairs, with $\log_{10}(\beta) = \{0, \dots, -7\}$ and $s = \{1, \dots, 50\}$; $s = 1$ corresponds to the solution with minimum objective function. Learning with $J = 100$ ReLU regularizers and $r(W^T W)$.

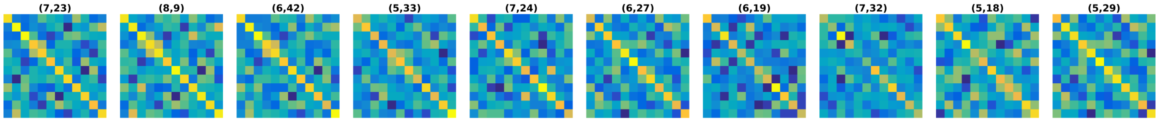


Figure 10: **Dihedral group (order 12)**: Gramian matrices $G = W^T W$ (size 12×12) of the solutions.

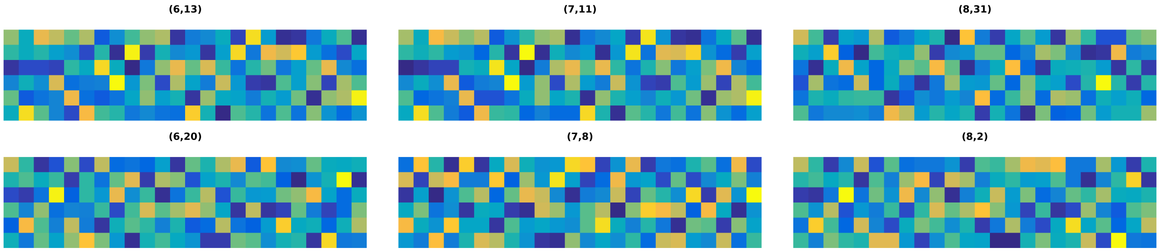


Figure 11: **Pyritohedral group (order 24)**: Solution orbits (6 shown), arranged with increasing variance of the intra-orbit signature distance. Each W, corresponding to one solution/initialization and one value of β is a 6×24 array of column-wise arranged dictionary elements. The entries $(\cdot, \cdot)$ denote β and solution index pairs, with $\log_{10}(\beta) = \{0, \dots, -7\}$ and $s = \{1, \dots, 50\}$; $s = 1$ corresponds to the solution with minimum objective function. Learning with $J = 100$ ReLU regularizers and $r(W^T W)$.

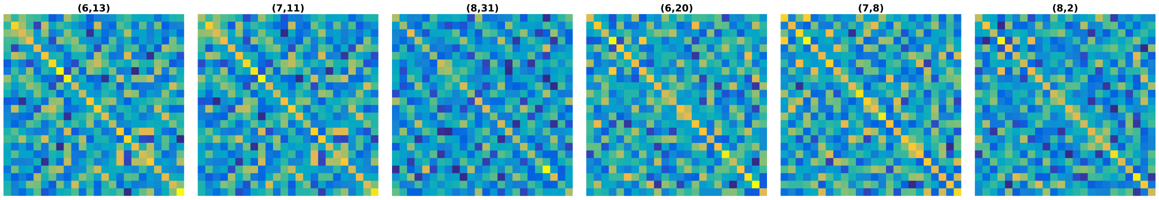
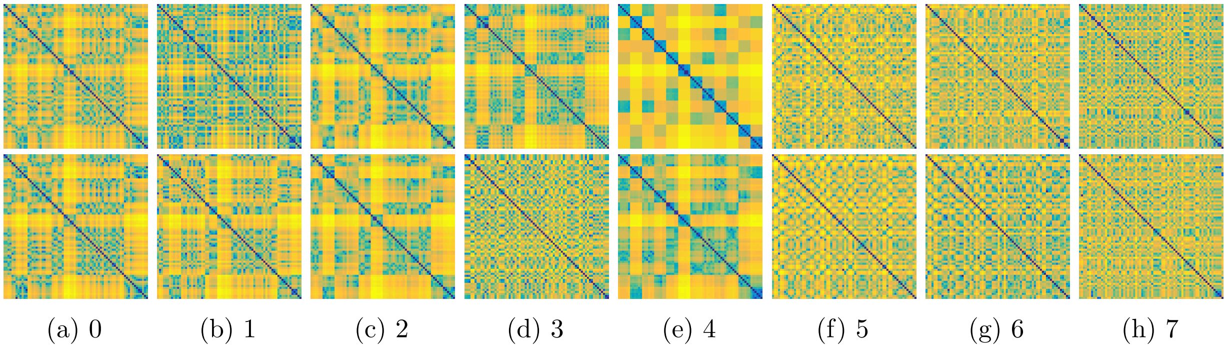


Figure 12: **Pyritohedral group (order 24)**: Gramian matrices $G = W^T W$ (size 24×24) of the solutions.

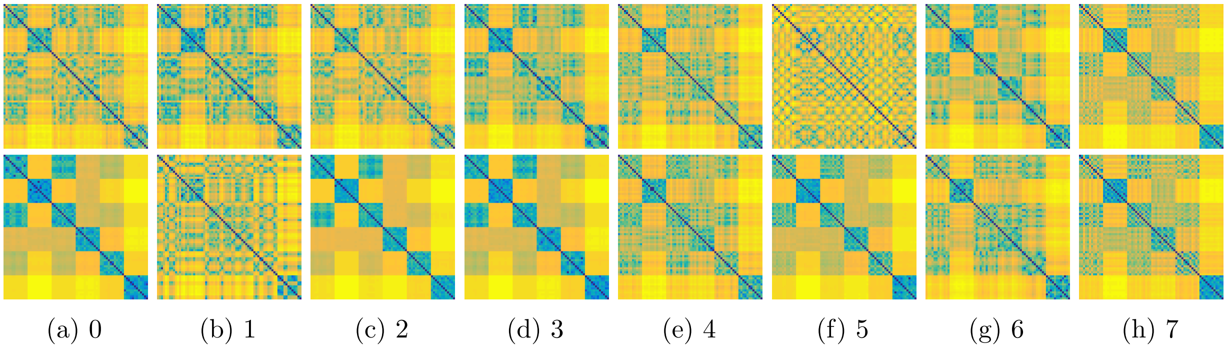
H.2 Dependency on parameters

Figures 13, 14, and 13 show the dependency of the minimization of Eq. (25) (minimum loss function out of 50 initializations) on the values of β, that controls the amount of regularization, and the hyper-parameter J controlling the number of nonlinearities. We tried $\log_{10} \beta = \{0, \dots, -7\}$ and $J = \{100, 30\}$. Top row shows $J = 100$, bottom row $J = 300$ with smaller values for β left-to-right. Each matrix is the same part of the distance matrices used in Figures 3, 4 and 5 for C_6, D_6 and T_h corresponding to 12, 6 and 3 orbits. A block diagonal structure of the matrices is an indication for equivariant maps (as expressed via small intra-orbit permutation distances). Even though there is no clear pattern emerging, one can note that the larger (no Abelian) groups improve by larger J and smaller β.



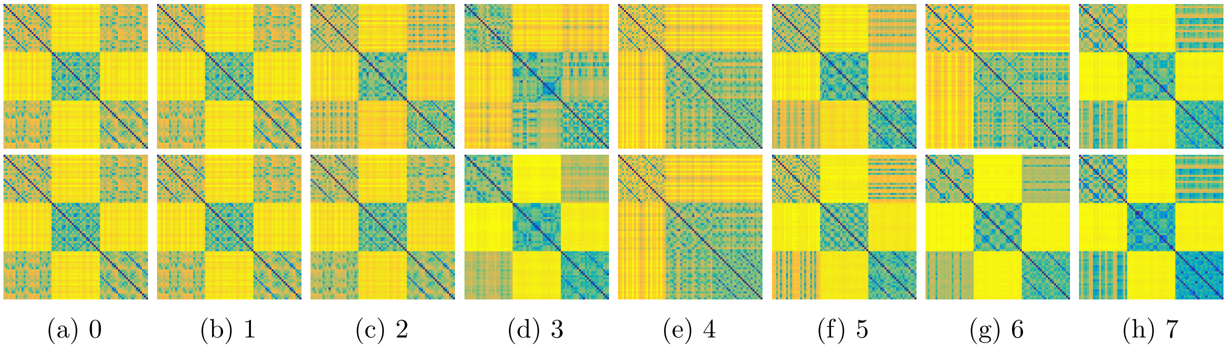
(a) 0 (b) 1 (c) 2 (d) 3 (e) 4 (f) 5 (g) 6 (h) 7

Figure 13: **Cyclic group C_6 (order 6)**: Dependency of the solution of Eq. (22) on β, i.e. the regularization constant and J, i.e. number of ReLU nonlinearities. Columns show $\log_{10}(1/\beta)$ in $0, \ldots, 7$ and rows correspond to $J = 100$ (top) and $J = 300$ (bottom) nonlinearities, excluding $r(W)$. Matrices are showing the permutation test r value for 12 orbits (arranged in 12 blocks of 6 orbit elements).



(a) 0 (b) 1 (c) 2 (d) 3 (e) 4 (f) 5 (g) 6 (h) 7

Figure 14: **Dihedral group D_6 (order 12)**: Dependency of the solution of Eq. (22) on β, i.e. the regularization constant and J, i.e. number of ReLU nonlinearities. Matrices are showing the permutation test r value for 6 orbits (arranged in 6 blocks of 12 orbit elements).



(a) 0 (b) 1 (c) 2 (d) 3 (e) 4 (f) 5 (g) 6 (h) 7

Figure 15: **Pyritohedral group T_h (order 24)**: Dependency of the solution of Eq. (22) on β, i.e. the regularization constant and J, i.e. number of ReLU nonlinearities. Matrices are showing the permutation test r value for 3 orbits (arranged in 3 blocks of 24 orbit elements).