Learning Lie Group Symmetry Transformations with Neural Networks

Alex Gabel [*1] **Victoria Klein** [*2] **Riccardo Valperga** [*1] **Jeroen S. W. Lamb** [2] **Kevin Webster** [2] **Rick Quax** [3]
Efstratios Gavves [1]

Abstract

The problem of detecting and quantifying the presence of symmetries in datasets is useful for model selection, generative modeling, and data analysis, amongst others. While existing methods for hard-coding transformations in neural networks require prior knowledge of the symmetries of the task at hand, this work focuses on discovering and characterizing unknown symmetries present in the dataset, namely, Lie group symmetry transformations beyond the traditional ones usually considered in the field (rotation, scaling, and translation). Specifically, we consider a scenario in which a dataset has been transformed by a one-parameter subgroup of transformations with different parameter values for each data point. Our goal is to characterize the transformation group and the distribution of the parameter values. The results showcase the effectiveness of the approach in both these settings.

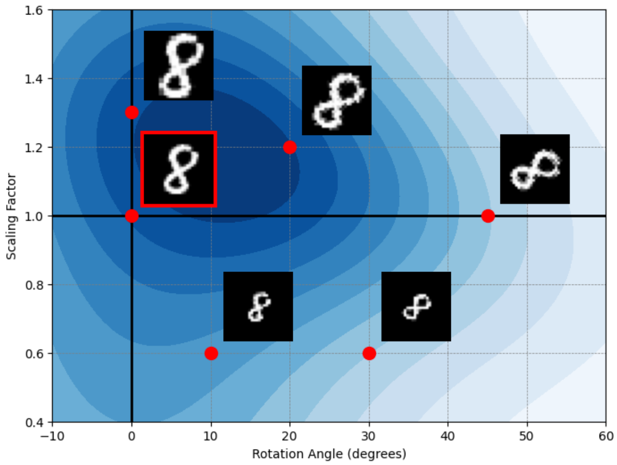


Figure 1. The distribution of transformations in a toy dataset that correspond to the Lie groups of rotation and (isotropic) scaling, given in terms of the parameters degree and scaling factor respectively; crucially, these groups are differentiable and can be (locally) decomposed into one-parameter subgroups.

1. Introduction

It has been shown that restricting the hypothesis space of functions that neural networks are able to approximate using known properties of data improves performance in a variety of tasks (Worrall & Welling, 2019; Cohen et al., 2018; Weiler et al., 2018; Zaheer et al., 2017; Cohen & Welling, 2016). The field of Deep Learning has produced a prolific amount of work in this direction, providing practical parameterizations of function spaces with the desired properties that are also universal approximators of the target functions (Yarotsky, 2022). In physics and, more specifically, time-series forecasting of dynamical systems, symmetries are ubiquitous and laws of motion are often symmetric with respect to various transformations such as rotations and translations, while transformations that preserve solutions of equations of motions are in one way or another associated with conserved quantities (Noether, 1918). In computer vision, successful neural network architectures are often invariant with respect to transformations that preserve the perceived object identity as well as all pattern information, such as translation, rotation and scaling. Many of these transformations are smooth and differentiable, and thus belong to the family of Lie groups, which is the class of symmetries we deal with in this work.

Although methods that hard-code transformations are capable of state-of-the-art performance in various tasks, they all require prior knowledge about symmetries in order to restrict the function space of a neural network. A broad class of, a priori unknown, transformations come into play in the context of modelling dynamical systems and in applications to physics. On the other hand, in vision tasks,

[*] Equal contribution [1] VIS Lab (Institute of Informatics), University of Amsterdam [2] Department of Mathematics, Imperial College London [3] CSL (Institute of Informatics), University of Amsterdam. Correspondence to: Victoria Klein <victoria.klein18@imperial.ac.uk>, Alex Gabel <a.gabel@uva.nl>, Riccardo Valperga <r.valperga@uva.nl>.

Proceedings of the 2^{nd} Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML) at the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. 2023. Copyright 2023 by the author(s).

identity-preserving transformations are often known beforehand. Despite this, these transformations are expressed differently by different datasets. As a result, algorithms for not only *discovering* unknown symmetries but also *quantifying* the presence of specific transformations in a given dataset, may play a crucial role in informing model selection for scientific discovery or computer vision, by identifying and describing physical systems through their symmetries and selecting models that are invariant or equivariant with respect to only those symmetries that are *actually* present in the dataset under consideration.

In this work, we address the problem of qualitatively and quantitatively detecting the presence of symmetries with respect to one-parameter subgroups within a given dataset (see Figure 1). In particular, let $\phi(t)$ be a one parameter subgroup of transformations. We consider the scenario in which a dataset $\{x_i\}_{i=1}^N$ has been acted on by $\phi(t)$, with a *different* value of the parameter t for every point x_i. Our goal is to characterise the group of transformations $\phi(t)$, as well as the *distribution* from which the parameters t have been sampled. We propose two models: a naive approach that successfully manages to identify the underlying one-parameter subgroup, and an autoencoder model that learns transformations of a one-parameter subgroup in the latent space and is capable of extracting the overall shape of the t-distributions. The cost of the latter is that the one-parameter subgroup in the latent space is not necessarily identical to that in pixel space. The work is structured as follows: Section 2 introduces some basic tools from Lie group theory; Section 3 outlines the method; Section 5 provides an overview of the existing methods that are related to our own; and lastly, results are shown in Section 4.

2. Background

The theoretical underpinnings of symmetries or invariance can be described using group theory (Fulton & Harris, 1991). In particular, we present the necessary theory of one-parameter subgroups (Olver, 1993) on which our method is based, following the logic of Oliveri (2010).

2.1. One-parameter subgroups

We focus on learning invariances with respect to one-parameter subgroups of a Lie group G, which offer a natural way to describe continuous symmetries or invariances of functions on vector spaces.

Definition 2.1. A **one-parameter subgroup** of G is a differentiable homomorphism $\phi : \mathbb{R} \to G$, more precisely, such that $\phi(t + s) = \phi(t)\phi(s)$ for all $t, s \in \mathbb{R}$.

Let the action of ϕ on the vector space $X \subset \mathbb{R}^n$ be a transformation $T : X \times \mathbb{R} \to X$ that is continuous in $x \in X$ and $t \in \mathbb{R}$. Because of continuity, for sufficiently small t and some fixed $x \in X$, the action is given by

$$T(x,t) \approx x + tA(x) \text{ where } A(x) := \left.\frac{\partial T(x,t)}{\partial t}\right|_{t=0} . \quad (1)$$

Note that this is equivalent to taking a first-order Taylor expansion in t around $t = 0$.

2.2. Generators

In general, we can use $A(x)$ in (1) to construct what is known as the **generator** of a one-parameter subgroup ϕ of a Lie group G, that in turn will characterise an ordinary differential equation, the solution to which coincides with the action T on X.

Let $C^\infty(X)$ be the space of smooth functions from X to X. The generator of ϕ is defined as a linear differential operator $L : C^\infty(X) \to C^\infty(X)$ such that

$$L = \sum_{i=0}^{n} (A(x))_i \frac{\partial}{\partial x_i} \quad (2)$$

describing the vector field of the infinitesimal increment $A(x)t$ in (1), where $\partial/\partial x_i$ are the unit vectors of X in the coordinate directions for $i = 1, \dots, n$. It can be shown (Olver, 1993) that, for a fixed $x \in X$, that $T(x,t)$ is the solution to the ordinary differential equation

$$\frac{dT(x,t)}{dt} = LT(x,t) \quad \text{where} \quad T(x,0) = x. \quad (3)$$

The solution to (3) is the exponential $T(x,t) = e^{tL}x$ where

$$e^{tL} := \sum_{k=0}^{\infty} \frac{(tL)^k}{k!}, \quad (4)$$

where L^k is the operator L applied k times iteratively.

For a one-parameter subgroup ϕ of a matrix Lie group $G \subset GL(n, \mathbb{R})$ and a fixed $x \in X$, it can be shown (Olver, 1993) that there exists a unique matrix $A \in \mathbb{R}^{n \times n}$ such that $A(x) = Ax$. This is a more restrictive approach as groups such as translations cannot be written as a matrix multiplication.

3. Method

As in Rao & Ruderman (1998); Sanborn et al. (2022); Dehmamy et al. (2021) the semi-supervised symmetry detection setting that we consider consists of learning the generator L of a one-parameter subgroup ϕ from pairs of observations of the form $\{(x_i, \bar{x} = T(x_i, t_i))\}_{i=1}^N$, where N is the number of observations and each $t_i \in \mathbb{R}$ is drawn from some unknown distribution $p(t)$. Not only do we attempt to learn the generator L, but also the unknown distribution $p(t)$ of the parameters $\{t_i\}_{i=1}^N$.

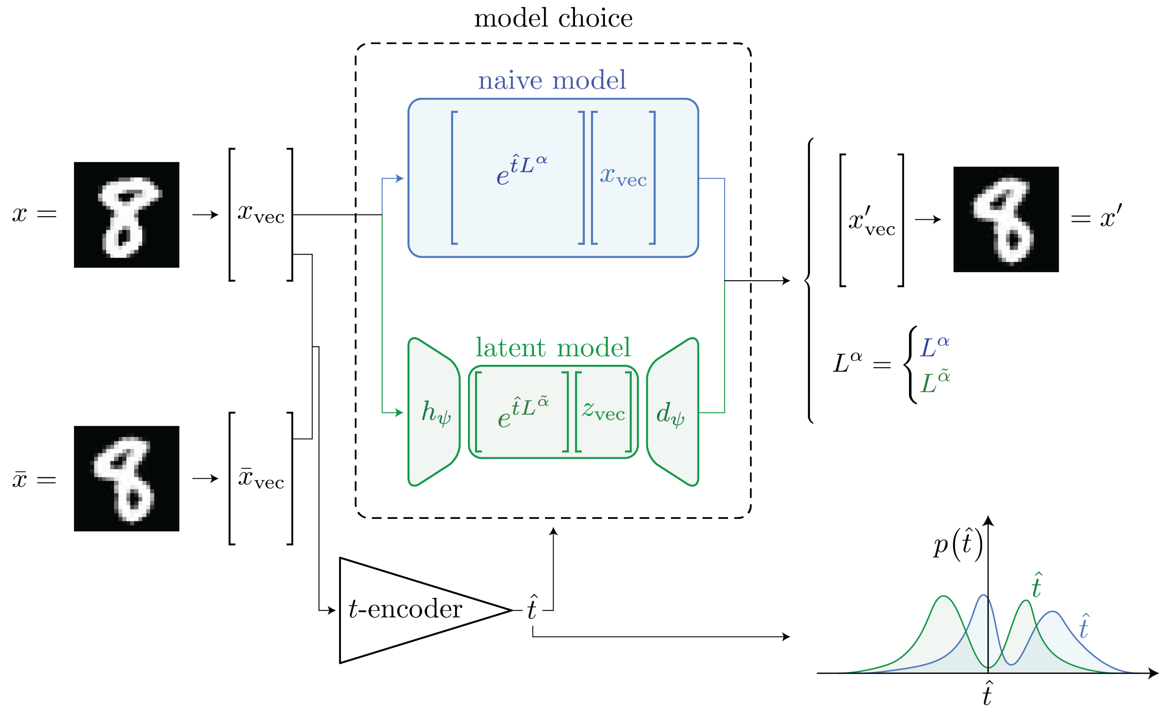


Figure 2. Model architecture.

3.1. Parametrisation of the generator

Deciding how to parametrise L has an effect on the structure of the model and ultimately on what one-parameter subgroups we are able to learn. For simplicity, consider one-parameter subgroups acting on $X \subset \mathbb{R}^2$, although this operator can be defined for higher-dimensional vector spaces. The generator L of ϕ is given as in Eq. (2) and we parametrise $A(x, y)$ as a linear operator in the basis $\{1, x, y\}$ with a coefficient matrix $A = \alpha \in \mathbb{R}^{2\times 3}$, giving

$$\begin{aligned} L^\alpha := (\alpha_{11} + \alpha_{12}x + \alpha_{23}y)\frac{\partial}{\partial x} \\ + (\alpha_{12} + \alpha_{22}x + \alpha_{23}y)\frac{\partial}{\partial y}. \end{aligned} \tag{5}$$

In this particular basis, for different values of α, the generator L^α is able to express one-parameter sub-groups of the affine group. This includes the "traditional" symmetries that are usually considered (translation, rotation, and isotropic scaling) and all other affine transformations[1]. This can be generalized to any functional form of the generator by augmenting the basis accordingly.

3.2. Discretisation and interpolation

The generator L^α is constructed as an operator that acts on a function $f : \mathbb{R}^2 \to \mathbb{R}$, given, in practice, by $I \in \mathbb{R}^{n\times n}$ such that $I_{ij} = f(i, j)$ are evaluations of f on a regularly-sampled $n \times n$ grid M of points $M_{ij} = (i, j) \in \mathbb{R}^2$. We then vectorise I, obtaining a point in a vector space $\tilde{I} \in \mathbb{R}^{n^2}$ such that $\tilde{I}_{i+j} := I_{ij}$ and construct the matrix operator $L^\alpha \in \mathbb{R}^{n^2\times n^2}$ as

$$\begin{aligned} L^\alpha := (\alpha_{11} + \alpha_{12}X_x + \alpha_{13}X_y)\frac{\partial}{\partial X_x} \\ + (\alpha_{21} + \alpha_{22}X_x + \alpha_{23}X_y)\frac{\partial}{\partial X_y}, \end{aligned} \tag{6}$$

acting on $\tilde{I}$, where $X_x \in \mathbb{R}^{n^2\times n^2}$ and $X_y \in \mathbb{R}^{n^2\times n^2}$ are such that $(X_x)_{ij} := i$ and $(X_y)_{ij} := j$, while $\partial/\partial X_x$ and $\partial/\partial X_y$ are also matrix operators in $\mathbb{R}^{n^2\times n^2}$. The exponential in (4) and the action T coincides with the matrix exponential.

In order to define $\partial/\partial X_x$ and $\partial/\partial X_y$ as operators that transform by infinitesimal amounts at discrete locations, we require an interpolation function. The Shannon-Whittaker theorem (Marks, 2012) states that any square-integrable, piecewise continuous function that is band-limited in the frequency domain can be reconstructed from its discrete samples if they are sufficiently close and equally spaced. For sake of interpolations, we will also assume that the

[1] Alternatively, the constant terms can be thought of as the drift terms (i.e. translation) and the four others can be arranged into a diffusion matrix.

function is periodic.

Interpolation: 1D In the case where M is a discrete set of n points in 1D, we have that $I(i+n) = I(i)$ for all $i = 1, \ldots, n$ samples. Shannon-Whittaker interpolation reconstructs the signal for all $x \in \mathbb{R}$ as

$$\begin{aligned} I(x) &= \sum_{i=0}^{n-1} I(i) Q(x-i), \quad \text{where} \\ Q(x) &= \frac{1}{n}\left[1 + 2\sum_{p=1}^{n/2-1} \cos\left(\frac{2\pi p x}{n}\right)\right] \end{aligned} \tag{7}$$

Differentiating Q with respect to x and evaluating it at every $x_i \in M$ gives an analytic expression for a vector field in $\mathbb{R}^n$, describing continuous changes in x at all n points (Rao & Ruderman, 1998). This is precisely what $\partial/\partial x$ or $\partial/\partial y$ in (5) are.

Interpolation: 2D In the case where M is a grid of $n \times n$ points in 2D, we construct the $n \times n$ matrices of the partial derivatives of Q with respect to x and y, analogously to the 1D case, stacking them to construct the $n^2 \times n^2$ block diagonal matrices $\partial/\partial X_x$ and $\partial/\partial X_y$. It is worth noting that alternative interpolation techniques can be used to obtain the operators and the method does not depend on any specific one.

Two different architectures, the main model and the latent model, are proposed to learn L^α and, in doing so, the action T.

3.2.1. Naive model

The coefficients α of L^α are approximated by fixed coefficients that are shared across the dataset, while the parameter t_i is approximated by $\hat{t}_i$ that depends on the input pair $(x_i, \bar{x}_i)$. We learn

1. the coefficients $\alpha \in \mathbb{R}^{2\times 3}$ of the generator L^α and
2. the parameters θ of an MLP f_θ that returns $f_\theta(x_i, \bar{x}_i) =: \hat{t}_i$ as a function of every input pair,

such that the solution to (3) for L^α is approximated by

$$\hat{T}(x_i, \bar{x}_i) := e^{f_\theta(x_i, \bar{x}_i) L^\alpha} x_i. \tag{8}$$

The model objective is then given by the reconstruction loss

$$\mathcal{L}_T(x_i, \bar{x}_i) = ||\hat{T}_\phi(x_i, \bar{x}_i) - \bar{x}_i||^2. \tag{9}$$

3.2.2. Latent model

While the model described above will prove to work sufficiently well for learning the coefficients α of L^α, the matrix exponential function in $\hat{T}$ in (8) can be costly to compute and difficult to optimise in high dimensions; consider that the cost of the matrix exponential in a single forward pass is roughly $O(n^3)$ using the algorithm of Al-Mohy & Higham (2010).

As a result, a different version of the model is proposed that incorporates an autoencoder for reducing dimension. The concept remains the same, but x_i is now mapped to some latent space $Z \subset \mathbb{R}^{n_Z}$ for $n_Z \ll n$, such that the exponential is taken in a significantly lower dimension. This is done by an encoder $h_\psi : X \to Z$ and a decoder $d_\psi : Z \to X$ such that $z_i = h_\psi(x_i)$ and $x_i \approx d_\psi(z_i)$.

We learn

1. the parameters ψ of an MLP autoencoder,
2. the coefficients $\tilde{\alpha} \in \mathbb{R}^{2\times 3}$ of the generator $L^{\tilde{\alpha}}$ for a one-parameter subgroup ϕ_Z acting on the latent space Z,
3. the parameters θ of an MLP f_θ that returns $f_\theta(x_i, \bar{x}_i) =: \hat{t}_i$ as a function of every original input pair $(x_i, \bar{x}_i)$,

such that the solution to (3) for L^α, the generator in the original space, is approximated by

$$\hat{T}^Z(x_i, \bar{x}_i) = d_\psi(e^{f_\theta(x_i, \bar{x}_i) L^{\tilde{\alpha}}} h_\psi(x_i)). \tag{10}$$

It is important to note that enforcing good reconstruction of the autoencoder alone does not enforce the commutativity of the diagram in Figure 3. To make it commutative, we use an objective that is a weighted sum of multiple terms. A simple reconstruction term for the autoencoder on each input example

$$\mathcal{L}_R(x_i) := ||d_\psi(h_\psi(x_i)) - x_i||^2, \tag{11}$$

a transformation-reconstruction term in the original space

$$\mathcal{L}_T^X(x_i, \bar{x}_i) := ||\hat{T}_\phi^Z(x_i, \bar{x}_i) - \bar{x}_i||^2, \tag{12}$$

a transformation-reconstruction term in the latent space

$$\mathcal{L}_T^Z(x_i, \bar{x}_i) := ||e^{f_\theta(x_i, \bar{x}_i) L^{\tilde{\alpha}}} h_\psi(x_i) - h_\psi(\bar{x}_i)||^2, \tag{13}$$

and a Lasso term on the generator coefficients $\tilde{\alpha}$. The overall loss of the latent model is

$$\begin{aligned} \mathcal{L}(x_i, \bar{x}_i) = \lambda_R(&\mathcal{L}_R(x_i) + \mathcal{L}_R(\bar{x}_i)) \\ &+ \lambda_X \mathcal{L}_T^X(x_i, \bar{x}_i) + \lambda_Z \mathcal{L}_T^Z(x_i, \bar{x}_i) \\ &+ \lambda_L ||\tilde{\alpha}||^2, \end{aligned} \tag{14}$$

where $\lambda_R, \lambda_X, \lambda_Z, \lambda_L \in \mathbb{R}$ are treated as hyperparameters.

$$\begin{tikzcd}
X \arrow[r, "T(\cdot,t)"] \arrow[d, "h_\psi"'] & X \\
Z \arrow[r, "e^{f_\theta(\cdot,\cdot)L^{\tilde{\alpha}}}"'] & Z \arrow[u, "d_\psi"']
\end{tikzcd}$$

Figure 3. The commuting diagram enforced by the objective function in the latent model: $T(x,t) \approx d_\psi(e^{f_\theta(x_i,\bar{x}_i)L^{\tilde{\alpha}}} h_\psi(x))$.

Recovering the group It is important to note that the one-parameter subgroup corresponding to the generator $L^{\tilde{\alpha}}$ and the generator L^α are *not* necessarily the same; $L^{\tilde{\alpha}}$ is the generator corresponding to some action on X of a one-parameter subgroup ϕ, while L^α is a different generator corresponding to some action on Z of a different one-parameter subgroup ϕ_Z.

3.3. Uniqueness

For both the naive model in Section 3.2.1 and the latent model in 3.2.2, the approximations $\hat{t}_i$ for the values of the parameters t_i require interpretation. Both models parameterise $\hat{T}$ or $\hat{T}^Z$ with the products $\hat{t}_i L^\alpha$ or $\hat{t}_i L^{\tilde{\alpha}}$ respectively, where $\hat{t}_i = f_\theta(x_i, \bar{x}_i)$. While both the values of $\hat{t}_i L^\alpha$ and $\hat{t}_i L^{\tilde{\alpha}}$ are unique for a given action on X and Z respectively, their decomposition is only unique up to a constant. Therefore, L^α or $L^{\tilde{\alpha}}$ and $\hat{t}$ approximate the generators and the parameter respectively up to a constant. Consequently, the one-parameter subgroup ϕ can only be deduced by the values of the individual coefficients in α *relative to one another*, as opposed to in absolute, likewise for ϕ_Z and $\tilde{\alpha}$. We therefore recover a scaled approximation for the distribution of $\hat{t}_i$.

3.4. The most general setting

Suppose we are given a labelled dataset $\mathcal{D} = \{(x_i, c_i)\}_{i=1}^N$ and a one-parameter subgroup ϕ. Then we call $\mathcal{D}$ *symmetric* or *invariant with respect to* ϕ if the action of ϕ preserves the object identity of the data points, where by object identity we mean any property of the data that we might be interested in. For example, in the case of MNIST handwritten digits, rigid transformations preserve their labels [2] and therefore, can be considered symmetries of the dataset. Now suppose that every x_i in $\mathcal{D}$ is acted on with a one-parameter subgroup ϕ_t to get $T\mathcal{D} = \{(T(x_i, t_i), c_i)\}_{i=1}^N$. The most general, fully unsupervised symmetry detection setting consists of learning ϕ, and characterize the distribution of the parameter t from just $\bar{\mathcal{D}}$. The idea is that, under the assumption that points with the same label are sufficiently similar for the subgroup transformation to account for the important difference[3], we can use labels to group data points, and compare those data points using methods such as the one presented in this paper. We leave the fully unsupervised symmetry detection setting for future work although we will emphasize that the proposed method can, in principle, be used in such setting without substantial changes to the architecture.

4. Experiments

4.1. Experiment setting

In practice, we experiment with a dataset of MNIST digits transformed with either 2D rotations or translations in one direction. To test the method's ability to learn distributions of these transformations, for each one-parameter subgroup $\phi \in \{SO(2), T(2)\}$ we construct a dataset $\{x_i, T(x_i, t_i)\}_{i=1}^N$ by sampling the parameters $t_i \in \mathbb{R}$ from various *multimodal* distributions.

As in (Rao & Ruderman, 1998), the dataset is composed of signals $I : M \longrightarrow \mathbb{R}$ regularly-sampled from a discrete grid of n^2 points $(x, y) \in \mathbb{R}^2$ for $n = 28$. The signals I are vectorised into points in $\mathbb{R}^{784}$ as described in Section 3.2. The implementation of the naive model is available here.

4.2. Main model experiments

The naive model architecture outlined in 3.2.1 consists of a fully-connected, 3-layer MLP for f_θ that was trained jointly with the coefficients α_{ij} using Adam (Kingma & Ba, 2014) with a learning rate of 0.001. Given the disproportionate number of trainable parameters in f_θ and the 6 coefficients in α, updating α_{ij} roughly 10 times for every update of θ in f_θ was found to be beneficial during training.

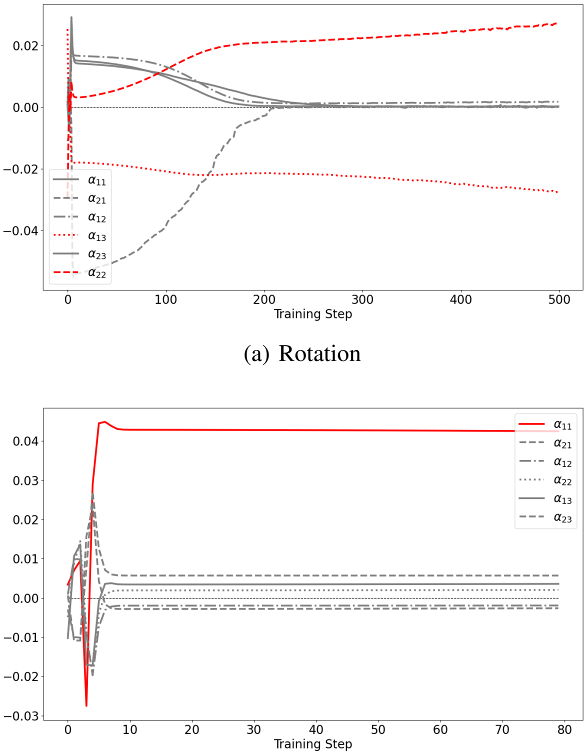
Coefficients Figure 4 shows the evolution of α_{ij} during training. It can be seen that after a few hundred steps, the coefficients α_{ij} that do not correspond to the infinitesimal generator of the symmetry expressed by the dataset drop to zero, while those that do, settle to values compatible with those of the ground truth generator L.

4.3. Latent model experiments

The latent model outlined in 3.2.2 consists of a fully-connected, 3-layer MLP f_θ, as in (8), to approximate

[2] With the exception of the number '9' that, if rotated 180 degrees becomes a '6'.

[3] Keeping MNIST hand-written digits as our paradigmatic example, digits with the same label differ by small transformations that account for handwriting style differences.



(a) Rotation

(b) Translation in x

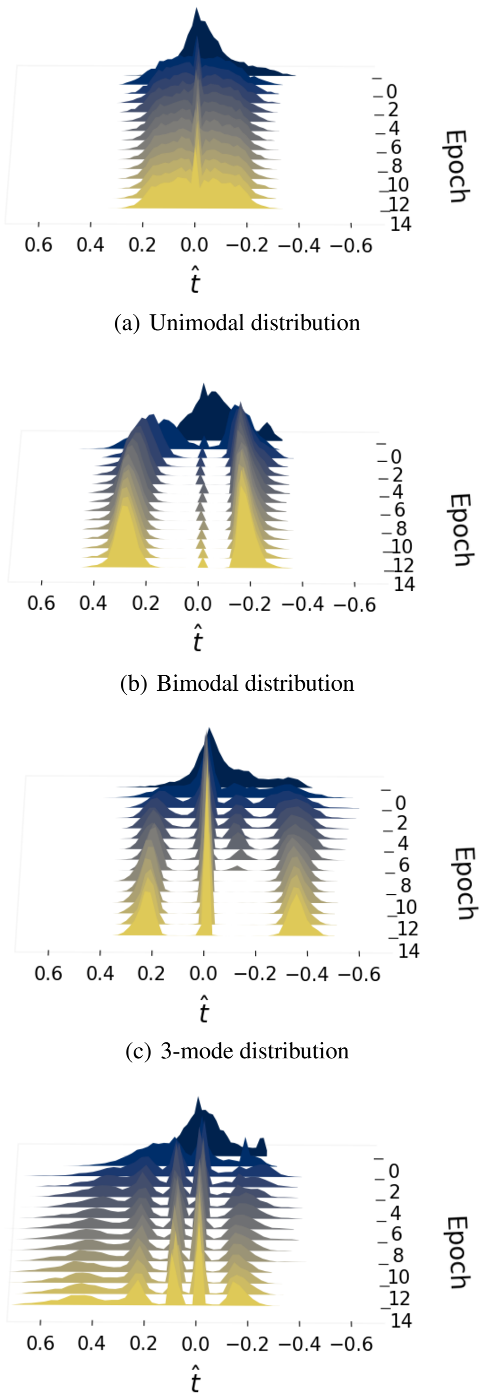
Figure 4. Training evolution of the coefficients α defining the generator L^{α} of the one-parameter subgroup, that are shown to converge to the ground-truth non-zero coefficients α for rotated ($-\alpha_{22} = \alpha_{13} = 1$ and 0 otherwise) and translated ($\alpha_{11} = 1$ and 0 otherwise) MNIST.

$\hat{t}$, and two fully-connected, 3-layer MLPs with decreasing/increasing hidden dimensions for the encoder h_{ψ} and d_{ψ}. We set the latent space to $n_Z = 25$. Similar to the naive model experiment above, f_{θ} was trained jointly with the coefficients α_{ij} using Adam (Kingma & Ba, 2014) with learning rate 0.001.

Parameters After every epoch (roughly 500 steps), the outputs of $\hat{t} = f_{\theta}$ were collected in a histogram to show $p(\hat{t})$. Figure 5 shows how the distribution of $\hat{t}$ changes during training and how multimodal distributions are clearly recovered, showing the same number of modes as the ground truth distribution from which the transformations were sampled.

5. Related Work

Symmetries in Neural Networks Numerous studies have tackled the challenges associated with designing neural network layers and/or models that are equivariant with respect to specific transformations (Finzi et al., 2021).



(a) Unimodal distribution

(b) Bimodal distribution

(c) 3-mode distribution

(d) 5-mode distribution

Figure 5. Training evolution of the distributions $p(\hat{t})$ of the learned parameters $\hat{t}$ computed by f_{θ} for the validation set. The figure shows that $p(\hat{t})$ resembles the original multi-modal distributions $p(t)$ of the transformations expressed by the dataset.

These transformations include continuous symmetries such as scaling (Worrall & Welling, 2019), rotation on spheres (Cohen et al., 2018), local gauge transformations (Cohen et al., 2019) and general E(2) transformations on the Euclidean plane (Weiler & Cesa, 2019), as well as discrete transformations like permutations of sets (Zaheer et al., 2017) and reversing symmetries (Valperga et al., 2022). Another line of research focuses on establishing theoretical principles and practical techniques for constructing general group-equivariant neural networks. Research in such areas show improved performances on tasks related to symmetries, but nonetheless require prior knowledge about the symmetries themselves.

Symmetry Detection Symmetry detection aims to discover symmetries from observations, a learning task that is of great importance in of itself. Detecting symmetries in data not only lends itself to more efficient and effective machine learning models but also in discovering fundamental laws that govern data, a long-standing area of interest in the physical sciences. Learned symmetries can then be incorporated after training in equivariant models or used for data augmentation for downstream tasks. In physics and dynamical systems, the task of understanding and discovering symmetries is a crucial one; in classical mechanics and more generally Hamiltonian dynamics, continuous symmetries of the Hamiltonian are of great significance since they are associated, through Noether's theorem (Noether, 1918), to conservation laws such as conservation of angular momentum or conservation of charge.

The first work on learning symmetries of one-parameter subgroups from observations were Rao & Ruderman (1998) and Miao & Rao (2007), which outline MAP-inference methods for learning infinitesimally small transformations. Sohl-Dickstein et al. (2010) propose a transformation-specific smoothing operation of the transformation space to overcome the issue of a highly non-convex reconstruction objective that includes an exponential map. These methods are close to ours in that we also make use of the exponential map to obtain group elements from their Lie algebra. Despite this, Sohl-Dickstein et al. (2010) do not consider the task of characterizing the distribution of the parameter of the subgroup nor do they consider the whole of pixel-space, using small patches instead. Cohen & Welling (2014) focus on disentangling and learning the distributions of multiple compact "toroidal" one-parameter subgroups in the data.

Neural Symmetry Detection A completely different approach to symmetry discovery is that of Sanborn et al. (2022), who's model uses a group invariant function known as the bispectrum to learn group-equivariant and group-invariant maps from observations. Benton et al. (2020) consider a task similar to ours, attempting to learn groups with respect-to-which the data is invariant, however, the objective places constraints directly on the network parameters as well as the distribution of transformation parameters with which the data is augmented. Alternatively, Dehmamy et al. (2021) require knowledge of the specific transformation parameter for each input pair (differing by that transformation), unlike our model, where no knowledge of the one-parameter group is used in order to find the distribution of the transformation parameter.

Latent Transformations Learning transformations of a one-parameter subgroup in latent space (whether that subgroup be identical to the one in pixel space or not) has been accomplished by Keurti et al. (2023) and Zhu et al. (2021). Nevertheless, other works either presuppose local structure in the data by using CNNs instead of fullly-connected networks or focus on disentangling interpretable features instead of directly learning generators that can be used as an inductive bias for a new model.

In contrary to the other works mentioned above, we propose a promising framework in which we can simultaneously

- perform symmetry detection in pixel-space, without assuming any inductive biases are present in the data *a priori*,
- parametrize the generator such that non-compact groups (e.g. translation) can be naturally incorporated,
- and learn both the generator and the parameter distributions.

6. Discussion

In this work we proposed a framework for learning one-parameter subgroups of Lie group symmetries from observations. Our method uses a neural network to predict the one-parameter of every transformation that has been applied to datapoints, and the coefficients of a linear combination of pre-specified generators. We show that our method can learn the correct generators for a variety of transformations as well as characterize the distribution of the parameter that has been used for transforming the dataset.

While the goal of learning both the coefficients of the generator and the distribution of the transformation parameter has not been accomplished by only one model in this work, modifying our existing framework to do so is a priority for future work. In addition, the proposed method lends itself well to being composed to form multiple layers, which can then be applied to datasets that express multiple symmetries. By doing so, ideally, each layer would learn one individual symmetry. We leave this study, and the more general, fully unsupervised setting described in 3.4, for future work.

Acknowledgements

This publication is based on work partially supported by the EPSRC Centre for Doctoral Training in Mathematics of Random Systems: Analysis, Modelling and Simulation (EP/S023925/1) and the Dorris Chen Award granted by the Department of Mathematics, Imperial College London.

References

Al-Mohy, A. H. and Higham, N. J. A new scaling and squaring algorithm for the matrix exponential. *SIAM Journal on Matrix Analysis and Applications*, 31(3):970–989, 2010.

Benton, G., Finzi, M., Izmailov, P., and Wilson, A. G. Learning invariances in neural networks from training data. *Advances in neural information processing systems*, 33: 17605–17616, 2020.

Cohen, T. and Welling, M. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, pp. 1755–1763. PMLR, 2014.

Cohen, T. and Welling, M. Group equivariant convolutional networks. In *International conference on machine learning*, pp. 2990–2999. PMLR, 2016.

Cohen, T., Weiler, M., Kicanaoglu, B., and Welling, M. Gauge equivariant convolutional networks and the icosahedral cnn. In *International conference on Machine learning*, pp. 1321–1330. PMLR, 2019.

Cohen, T. S., Geiger, M., Köhler, J., and Welling, M. Spherical cnns. *arXiv preprint arXiv:1801.10130*, 2018.

Dehmamy, N., Walters, R., Liu, Y., Wang, D., and Yu, R. Automatic symmetry discovery with lie algebra convolutional network. *Advances in Neural Information Processing Systems*, 34:2503–2515, 2021.

Finzi, M., Welling, M., and Wilson, A. G. A practical method for constructing equivariant multilayer perceptrons for arbitrary matrix groups, 2021.

Fulton, W. and Harris, J. *Representation Theory: A First Course*. Graduate Texts in Mathematics. Springer New York, 1991. ISBN 9780387974958. URL https://books.google.nl/books?id=6GUH8ARxhp8C.

Keurti, H., Pan, H.-R., Besserve, M., Grewe, B. F., and Schölkopf, B. Homomorphism autoencoder – learning group structured representations from observed transitions, 2023.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Langley, P. Crafting papers on machine learning. In Langley, P. (ed.), *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)*, pp. 1207–1216, Stanford, CA, 2000. Morgan Kaufmann.

Marks, R. J. I. *Introduction to Shannon sampling and interpolation theory*. Springer Science & Business Media, 2012.

Miao, X. and Rao, R. P. Learning the lie groups of visual invariance. *Neural computation*, 19(10):2665–2693, 2007.

Noether, E. Invariante variationsprobleme. *Nachrichten von der Gesellschaft der Wissenschaften zu Göttingen, Mathematisch-Physikalische Klasse*, 1918:235–257, 1918. URL http://eudml.org/doc/59024.

Oliveri, F. Lie symmetries of differential equations: Classical results and recent contributions. *Symmetry*, 2, 06 2010. doi: 10.3390/sym2020658.

Olver, P. *Applications of Lie Groups to Differential Equations*. Graduate Texts in Mathematics. Springer New York, 1993. ISBN 9780387950006. URL https://books.google.nl/books?id=sI2bAxgLMXYC.

Rao, R. and Ruderman, D. Learning lie groups for invariant visual perception. *Advances in neural information processing systems*, 11, 1998.

Sanborn, S., Shewmake, C., Olshausen, B., and Hillar, C. Bispectral neural networks. *arXiv preprint arXiv:2209.03416*, 2022.

Sohl-Dickstein, J., Wang, C. M., and Olshausen, B. A. An unsupervised algorithm for learning lie group transformations. *arXiv preprint arXiv:1001.1027*, 2010.

Valperga, R., Webster, K., Turaev, D., Klein, V., and Lamb, J. Learning reversible symplectic dynamics. In *Learning for Dynamics and Control Conference*, pp. 906–916. PMLR, 2022.

Weiler, M. and Cesa, G. General e (2)-equivariant steerable cnns. *Advances in Neural Information Processing Systems*, 32, 2019.

Weiler, M., Geiger, M., Welling, M., Boomsma, W., and Cohen, T. S. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *Advances in Neural Information Processing Systems*, 31, 2018.

Worrall, D. and Welling, M. Deep scale-spaces: Equivariance over scale. *Advances in Neural Information Processing Systems*, 32, 2019.

Yarotsky, D. Universal approximations of invariant maps by neural networks. *Constructive Approximation*, 55(1): 407–474, 2022.

Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., and Smola, A. J. Deep sets. *Advances in neural information processing systems*, 30, 2017.

Zhu, X., Xu, C., and Tao, D. Commutative lie group vae for disentanglement learning, 2021.