IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 5 April 2023, accepted 27 April 2023, date of publication 10 May 2023, date of current version 18 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3274938

RESEARCH ARTICLE

Data Symmetries and Learning in Fully Connected Neural Networks

FABIO ANSELMI[1,2], LUCA MANZONI[1], ALBERTO D'ONOFRIO[1], ALEX RODRIGUEZ[1], GIULIO CARAVAGNA[1], LUCA BORTOLUSSI[1], AND FRANCESCA CAIROLI[1]

[1]Department of Mathematics and Geosciences, University of Trieste, 34127 Trieste, Italy
[2]Center for Brains, Minds and Machines, McGovern Institute, Massachusetts Institute of Technology, Cambridge, MA 02139, USA

Corresponding author: Fabio Anselmi (fabio.anselmi@units.it)

The work of Giulio Caravagna was supported by the Associazione Italiana Ricerca Cancro (AIRC) under Grant MFAG 2020 ID. 24913.
The work of Luca Bortolussi was supported by the PRIN project "SEDUCE" under Grant 2017TWRCNB.

ABSTRACT Symmetries in the data and how they constrain the learned weights of modern deep networks is still an open problem. In this work we study the simple case of fully connected shallow non-linear neural networks and consider two types of symmetries: full dataset symmetries where the dataset X is mapped into itself by any transformation g, i.e. $gX = X$ or single data point symmetries where $gx = x$, $x \in X$. We prove and experimentally confirm that symmetries in the data are directly inherited at the level of the network's learned weights and relate these findings with the common practice of data augmentation in modern machine learning. Finally, we show how symmetry constraints have a profound impact on the spectrum of the learned weights, an aspect of the so-called network implicit bias.

INDEX TERMS Artificial neural networks, symmetry invariance, equivariance.

I. INTRODUCTION AND PREVIOUS WORK

Symmetries are ubiquitous in nature from subatomic particles to man-made designs, art, and mathematics. It is natural therefore to suppose that an efficient signal representation would take advantage of such properties. In this spirit, the work we propose has the purpose to study, both theoretically and experimentally, how data symmetries are reflected in the learned weights of a shallow non-linear fully connected network, one of the simplest prototype architectures of modern artificial neural networks. In our setting, symmetries are understood as identity preserving transformations. Invariance to symmetries in pattern recognition is an old and challenging problem [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12]. More recently, in the context of machine learning, data symmetries have been used to derive and learn data representations with the properties of equivariance and invariance (see e.g. [13], [14], [15], [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26], [27], [28]). Interestingly, data symmetries can be explicitly used for reducing the sample complexity of downstream supervised learning, e.g., by constructing representations that are invariant to transformations irrelevant to the learning task (see e.g. [20], [29], [30]). For example, when aiming to solve an image classification task, information such as object position, scale or rotation should not affect the decision of a neural network. In fact, representations that reflect symmetries inherent in the data distribution define a quotient space where points are equivalent up to transformations. In this space, the sample complexity of learning (the size of the labeled training set) can be reduced by some aggregation strategy on the representation coefficients (e.g. [31], sec. II of [32], [14], [17], [33], [34]). On the contrary, representations that are not invariant to label-preserving data transformations have poor performance, as one can easily see training a network on a dataset of images and testing on, e.g., random rotations thereof. Convolutional neural networks (CNNs) for example have an explicit parametrization for equivariance and robustness to shifts in the input (translations) through convolutions and pooling, respectively (see also [35]). Other architectures, e.g. [36], embed specific parametrizations for rotation and scale invariance. The underlying hypothesis for the ability of such architectures to efficiently describe the dataset is the invariance of the single data label to specific transformations. Let us give a simple example in the case of translations and natural images. Due to the statistics of the natural images, the

The associate editor coordinating the review of this manuscript and approving it for publication was Sotirios Goudos.

probability of detection of an edge in an image is (approximately) independent of the part of the image considered. This observation is pointing to the fact that an image representation implemented by a translation equivariant or invariant network is preferable. This is indeed what happens with CNNs.

However, we can pose the same question from another point of view: what if instead of enforcing such symmetries (e.g. translation equivariance by convolutional operations) in the architecture, we enforce them in the dataset and see how the learned weights reflect them? Indeed, when such symmetries are not imposed by construction, it is not clear how the network's learned weights reflect data symmetries. *To the best of our knowledge, a rigorous formulation characterizing invariance properties of neural network weights in presence of data symmetries is lacking in the literature*.

In this work we partially fill this gap by studying how simple transformations impact learning in a ReLu shallow fully-connected network on the MNIST classification task when learning is achieved through gradient descent. Given the complexity of real-world image transformations, we begin by studying smaller, simpler classes of analytically tractable transformations i.e., those that belong to a group. In particular, we focus on common real world transformations such as translations, scales, rotations, and reflections (with respect to one or multiple axes). However, it should be noted that groups do not exhaust all possible object-identity preserving transformations: for example, object deformations or a change in an object's style or texture. Nevertheless, the major advantage of working with groups is that their mathematical structure is well understood, with many concepts and tools available for analysis. Specifically, we consider two types of symmetries (see section II): w.r.t. each single image in the dataset and w.r.t. the whole dataset. As an example of the first type of symmetry, we consider images that are symmetric w.r.t. reflections (vertical or horizontal or both). In this case each image x is a fixed point for any of the group elements, e.g. $Rx = x$, with R denoting the reflection transformation. An example of the second symmetry is when the dataset X is composed by images and all their transformations. In this case the whole dataset is left invariant by the action of any transformation i.e. $RX = X$. As we will see later this happens when the dataset is composed by orbits of data w.r.t. the transformations in the group (transformation-augmented dataset).

Here we provide formal results relating the invariance properties of the learned weights with the two types of symmetries described above and test our predictions with experiments, visualizing the learned weights (for a mathematical statement of the problem see the beginning of section III).

We start giving some background in section II introducing the definition of group transformations, orbits and invariance/equivariance properties of the data representation. Section III contains the theoretical results of our work. We first prove equivariance of the gradient of the augmented loss (Theorem 1). Then

- Corollary 1 characterizes the symmetry properties of the possible solutions of the data augmented problem.
- Corollary 2 shows that if each single datum is symmetric then the learned weights must be symmetric.
- Subsection III-B analyzes the case when data are only symmetric w.r.t. a subset of transformations and when multiple symmetries are present in the dataset.
- Subsection III-C analyzes how different data augmentations pose constraints for the learned weights in Fourier domain. This can be seen as an aspect of the so called *network implicit bias* (see [37], [38], [39], [40], [41], [42], [43], [44]).

In section IV we test our theoretical results in a simple task of classification on MNIST. The last section summarizes our results, discusses some problems and future lines of research.

II. THEORETICAL BACKGROUND: GROUPS, INVARIANCE, AND EQUIVARIANCE

We focus on transformations that have a group structure. We recall the formal definition of a group (see [45], [46]):

Definition 1: A group $(\mathcal{G}, \cdot)$ is a set of elements $\mathcal{G}$ with a binary composition rule $\cdot$ such that the following properties hold:

- *Closure: composing two group elements results in another group element.*

$$\forall a, b \in \mathcal{G}, \quad \exists c \in \mathcal{G} \ \ s.t. \ \ a \cdot b = c.$$

- *Identity: the identity element belongs to the group.*

$$\exists e \in \mathcal{G} \quad such\ that \quad \forall\, a \in \mathcal{G}, \quad e \cdot a = a \cdot e = a.$$

- *Inverse: each group element has an inverse.*

$$\forall\, a \in \mathcal{G}, \quad \exists a^{-1} \quad such\ that \quad a \cdot a^{-1} = e.$$

- *Associativity:*

$$(a \cdot b) \cdot c = a \cdot (b \cdot c), \quad \forall\, a, b, c \in \mathcal{G}.$$

We consider the input space X to be a subset of the d dimensional vector space $\mathbb{R}^d$. We denote the transformation of a point $x \in X$ by the group element $g \in (\mathbb{R}^{d\times d}, \cdot)$ as the action of the matrix $g \in \mathcal{G}$ on the vector $x \in X$ i.e. $gx := g \cdot x$. Moreover, we consider unitary groups [46] i.e. groups for which each element $g \in \mathcal{G}$ is such that $gg^T = g^T g = e$. For simplicity, we also consider discrete groups (or finite subgroups of continuous groups). The results below can be generalized to continuous groups by substituting the sums with integrals. One of the simplest examples of a unitary finite group is $\mathcal{R}_N$, the group of N rotations in the plane $\mathbb{R}^2$, whose elements are 2D rotation matrices of the form

$$R_{\theta_i} := \begin{bmatrix} \cos(\theta_i) & \sin(\theta_i) \\ -\sin(\theta_i) & \cos(\theta_i) \end{bmatrix} \in \mathbb{R}^{2\times 2}, \quad \theta_i = i\frac{2\pi}{N}$$

with $i \in [N]$, $[N] := \{1, 2, \ldots, N\}$. It is straightforward to verify that the set of matrices $\mathcal{R}_N := \{R_{\theta_i} : i \in [N]\}$ together with the operation of 2×2 matrix multiplication form a group. A key mathematical object in this context is that of an orbit. Let $Orb_{\mathcal{G}}(x)$ denote the *orbit* of $x \in X$ under the group $\mathcal{G}$,

defined as the set of transformations of x over all elements of the group:

$$Orb_{\mathcal{G}}(x) := \{gx : g \in \mathcal{G}\}. \tag{1}$$

For the group of plane rotations $\mathcal{R}_N$, the orbit of a vector $v \in \mathbb{R}^2$ is simply $Orb_{\mathcal{R}_N}(v) := \{R_{\theta_i} v : i \in [N]\}$, the set of all rotations of v. The notion of orbit is tightly linked with the modern machine learning technique of *data augmentation* (see e.g. [22]). Strictly speaking, augmenting the dataset w.r.t. a group transformation consists of generating a new dataset containing the old dataset and all its element orbits. Due to the closure property, an augmented dataset is mapped into itself by any group element. Let us consider, for simplicity, a dataset consisting of one signal and its orbit augmentation under the reflection group $\mathcal{G} = \{e, R\}$, where R is the reflection operator flipping right to left and left to right. The action of any $g \in \mathcal{G}$ on X corresponds to a permutation of the orbit elements and we have:

$$RX = R(x, Rx) = (Rx, RRx) = (Rx, x) = X.$$

In this case the *symmetry occurs at the level of the whole dataset*. In practice, in order to speed up the computations (see also section III-D), one only uses a few of the element orbits to perform data augmentation. The second important case we consider is that of single image symmetry i.e. for all $x \in X$ we have that for all $g \in \mathcal{G}$, $gx = x$. As an example, consider images of centered butterflies under reflection about the y-axis. Besides invariance, an important notion in this context is that of equivariance. For our specific setting we have the following definition:

Definition 2: A function $\phi : \mathbb{R}^k \to \mathbb{R}^k$ is called equivariant w.r.t. the group $\mathcal{G}$ if and only if

$$\phi(gx) = g\phi(x) \tag{2}$$

for all $g \in \mathcal{G}$ and $x \in X$.

Invariance can be considered a particular case of equivariance when $\phi(gx) = \phi(x)$ for all $g \in \mathcal{G}$ and $x \in X$. As we will see, the equivariance property of the gradient descent associated with the neural network loss will play a crucial role in deriving symmetry properties of the learned weights.

III. THEORETICAL RESULTS

In the following we present our main theoretical results as theorems and corollaries. For all the proofs we adopt the following setting. We consider training data $(x_i, y_i)_{i=1}^R$, $x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$ and class predictors $\hat{y} : \mathbb{R}^d \to \mathbb{R}$ of the form

$$\hat{y}(x) = v^T \sigma \langle W, x \rangle \tag{3}$$

where σ is a function acting component-wise on the scalar product of $W \in \mathbb{R}^{d \times k}$ with $x \in \mathbb{R}^d$ and $v \in \mathbb{R}^k$ (in the experiments we used the rectifier linear unit non linear function (ReLu)).

We consider the loss function

$$\mathcal{L}(W; X, y) = \frac{1}{R} \sum_{i=1}^{R} \ell(v^T \sigma \langle W, x_i \rangle, y_i), \tag{4}$$

where $\ell : \mathbb{R} \to \mathbb{R}_+$ is the individual example loss. In our experiments we used the cross entropy loss defined, in the simple case of binary classification as

$$\ell(y_i, \hat{y}_i) = y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)$$

where $\hat{y}_i$ is the predicted label and y_i the ground truth.

In the case of data augmentation under $\mathcal{G}$, the dataset is defined as:

$$(x_i, y_i)_{i=1}^R \to (g_j x_i, y_{ij})_{ij=1}^{R,|\mathcal{G}|} \tag{5}$$

where y_{ij} is the label corresponding to $g_j x_i$. We note that in the classification setting we assume the transformations do not change the object identity i.e. $y_{ij} = y_i, \ \forall j$.
The corresponding group-augmented loss is therefore given by

$$\mathcal{L}_a(W; X, y) = \frac{1}{Q} \sum_{i=1}^{R} \sum_{j=1}^{|\mathcal{G}|} \ell(v^T \sigma \langle W, g_j x_i \rangle, y_i), \tag{6}$$

where $Q := R|\mathcal{G}|$.

A. WEIGHT SYMMETRIES

Having provided the necessary definitions we can now give a formal statement of the problem we are analyzing:

Problem: Given a shallow neural network parameterized as in eq.3 and a dataset as in eq. 5 we want to characterize the invariance of the trained weights W w.r.t. a finite group $\mathcal{G}$ when the data are singularly or globally symmetric under $\mathcal{G}$.

In the next sections we will derive properties of the gradient of the losses in eqs. (4),(6) which, by gradient descent, determine the updates of the network weights to study how different data symmetries impact the learned weights. Let us start considering the case of data augmentation i.e. when the data are globally symmetric. We recall an important result proving that, in the case the data are augmented with their full orbits, the gradient descent of the augmented loss is an equivariant function w.r.t. the group of transformations:

Theorem 1 (See Also, e.g., [18], [47]): Consider the Loss in eq. (6). The gradient of the Loss is an equivariant function under $\mathcal{G}$:

$$\nabla_W \mathcal{L}_a(gW; X, y) = g \nabla_W \mathcal{L}_a(W; X, y), \quad \forall \, g \in \mathcal{G}.$$

Proof: Taking the gradient of eq. (6) we have:

$$\nabla_W \mathcal{L}_a(W; X, y) = \frac{1}{Q} \sum_{i=1}^{R} \sum_{j=1}^{|\mathcal{G}|} \ell'(v^T \sigma \langle W, g_j x_i \rangle, y_i) v^T \sigma' \langle W, g_j x_i \rangle g_j x_i$$

Calculating $\nabla_W \mathcal{L}_a(gW; X, y)$ we have:

$$\begin{aligned}
&\nabla_W \mathcal{L}_a(gW; X, y) \\
&= \frac{1}{Q}\sum_{i=1}^{R}\sum_{j=1}^{|\mathcal{G}|} \ell'(v^T \sigma\left\langle gW, g_j x_i\right\rangle, y_i) v^T \sigma'\left\langle gW, g_j x_i\right\rangle g_j x_i \\
&= \frac{1}{Q}\sum_{i=1}^{R}\sum_{j=1}^{|\mathcal{G}|} \ell'(v^T \sigma\left\langle W, g^T g_j x_i\right\rangle, y_i) v^T \sigma'\left\langle W, g^T g_j x_i\right\rangle g_j x_i \\
&= \frac{1}{Q}\sum_{i=1}^{R}\sum_{j=1}^{|\mathcal{G}|} \ell'(v^T \sigma\left\langle W, \hat{g}_j x_i\right\rangle, y_i) v^T \sigma'\left\langle W, \hat{g}_j x_i\right\rangle g\hat{g}_j x_i \\
&= g\nabla_W \mathcal{L}_a(W; X, y)
\end{aligned}$$

where in the fourth line we redefined $g^T g_j = \hat{g}_j$ to get $g_j = g\hat{g}_j$. The key property here is the closure of the group of transformations. ■

Theorem 1 allows to characterize the symmetry properties of the set of possible solutions. We have:

Corollary 1: Consider the Loss in eq. 6 and suppose we initialize the network with symmetric weights i.e. $gW_0 = W_0$ for all $g \in \mathcal{G}$. Then the learned weights are symmetric $W_T = gW_T$, where W_T are the weights at the final learning time T.

Proof: We first notice that due to the property $gX = X$, $\nabla_W \mathcal{L}_a(W; X, y) = \nabla_W \mathcal{L}_a(W; gX, y)$. Then by the dot product structure of the Loss we can see the transformation as applied to the weights i.e. $\langle W, gx\rangle = \left\langle g^T W, x\right\rangle$. Thus, using the equivariance property in theorem 1, we have that for all $g \in \mathcal{G}$:

$$\begin{aligned}
\nabla_W \mathcal{L}_a(W; X, y) &= \nabla_W \mathcal{L}_a(W; gX, y) \\
&= \nabla_W \mathcal{L}_a(g^T W; X, y) \\
&= g^T \nabla_W \mathcal{L}_a(W; gX, y).
\end{aligned}$$

Therefore each update of the weights in the gradient descent is group invariant and as a consequence, the learned weights at time T

$$W_T = W_0 - \gamma \sum_{k=1}^{T} \nabla_W \mathcal{L}_a(W_k; X, y).$$

are invariant. ■

We consider now the case of symmetric data i.e. $gx = x$ for all $x \in X, g \in \mathcal{G}$.

Corollary 2: Consider the Loss in eq. (4), with symmetric data i.e $g_j x_i = x_i$, $\forall\, j, i$. Then, if $gW_0 = W_0$ for all $g \in \mathcal{G}$ we have $gW_T = W_T$.

Proof: To prove the symmetry of the learned weights we look at the symmetry of each update of the weight. In order to do so, as in the previous proof, it is sufficient to prove that $g\nabla_W \mathcal{L}(W_k; X, y) = \nabla_W \mathcal{L}(W_k; X, y)$ for all $g \in \mathcal{G}$. To see that this is true consider the weight updates from time k to time $k+1$:

$$\begin{aligned}
&W_{k+1} - W_k \propto \nabla_W \mathcal{L}(W_k; X, y) \\
&\quad = \frac{1}{R}\sum_{i=1}^{R} \ell'(v^T \sigma\left\langle W_k, x_i\right\rangle), y_i) v^T \sigma'\left\langle W_k, x_i\right\rangle x_i.
\end{aligned}$$

The action of g is only applied to the vector x_i at the right of the expression above. By assumption, $gx_i = x_i$ so we have that the gradient is symmetric. Therefore, if W_0 is invariant, and given that each single update invariant, W_T is invariant. ■

B. DYNAMICS UNDER PARTIAL OR DIFFERENT SYMMETRIES

The symmetries studied in the previous sections are extreme cases in the sense that they involve either the whole dataset or single data instances. Intermediate situations can occur. In the following we analyze two of them.

1) PARTIAL SYMMETRIES

In the case of single instance symmetry, it can happen that the data instances are symmetric only w.r.t. to a subgroup of the transformations. More precisely, for each x_i in the dataset, we define a stabilizer group as:

$$\mathcal{G}_{x_i} = \{g \in \mathcal{G}, \quad s.t., \quad gx_i = x_i\} \tag{7}$$

i.e. the set of transformations of the group $\mathcal{G}$ that leave x_i unchanged. This set forms a subgroup of the original group. Thus the gradient update of the weights can be written as

$$\nabla_W \mathcal{L}(W, X, y) = \sum_i \frac{1}{|\mathcal{G}_{x_i}|} \sum_{g \in \mathcal{G}_{x_i}} \nabla_W \mathcal{L}(W, gx_i, y_i)$$

i.e. a sum of terms whose symmetry depends on each stabilizer size. Thus the learned weights symmetry will be driven by data that have bigger stabilizer sizes, i.e. more symmetric data.

2) DIFFERENT SYMMETRIES

A different point of view considers that the dataset can be partitioned into subsets, each one obeying a different symmetry i.e.

$$X = \bigcup_p X_p, \quad gx_i = x_i, \quad \forall\, g \in \mathcal{G}_p,\ x_i \in X_p$$

i.e. all signals in X_p are symmetric w.r.t. $\mathcal{G}_p$. In this case the gradient will be composed by addends symmetric w.r.t. different groups each weighted proportionally to the size of each X_p. In other words the most symmetric subset of the signals will dictate the weights symmetry.

C. IMPLICIT REGULARIZATION AND DATA AUGMENTATION

The idea behind the implicit regularization is that the loss landscape of a network has many minima, and which minimum one converges to after training depends on many factors, including the choice of model architecture and parametrization [37], [38], the initialization scheme [39] and the optimization algorithm [40], [41], [42]. The implicit regularization of state-of-the-art models has been shown to play a critical role in the generalization of deep neural networks [43], [44]. Recent theoretical work [37] on L-layer *deep linear*

networks proved that (i) fully connected layers induce a depth-independent ridge (ℓ_2) regularizer in the spatial domain of the networks weights whereas, surprisingly, *full-width convolutional layers* induce a depth-dependent *sparsity* ($\ell_{2/L}$) regularizer in the *frequency* domain. Thus implicit regularization emerges from a complex interaction between the ANN architecture, weight initialization scheme, learning rule and *data statistics*. In particular, the latter is deeply influenced by the symmetries in the dataset. In this section we are going to focus on a particular aspect of the implicit regularization: the effect of data augmentation or single instance symmetries on the Fourier spectrum of the learned weights. We begin providing a simple example illustrating the impact of translation augmentation of a function on its spectrum. The idea beyond translation augmentation is to create an invariant function w.r.t. the translation transformation. Suppose now we have a translation invariant function i.e. $\underline{f} : \mathbb{R} \to \mathbb{R}$ s.t. $\underline{f}(w+q) = \underline{f}(w), \quad \forall q \in R$. Clearly $\underline{f}$ is constant and $\hat{\underline{f}}(0)$ is the only non zero Fourier component, the DC component (where $\hat{f}$ indicates the Fourier transform). A way to prove this intuitive statement is as follows. Consider a translation invariant function given by

$$\underline{f}(w) = \int_{-\infty}^{\infty} dt\, f(w-t), \quad f : \mathbb{R} \to \mathbb{R}.$$

This is one of the possible ways to construct such a function, simply by averaging all translated versions of the original function f. Note that the function $\underline{f}$, by the change of variables $t' = w - t$, is simply the integral of the function and therefore a constant function. Its Fourier transform, using the identity $\widehat{f(\cdot - t)} = e^{ikt}\hat{f}(k)$ is:

$$\hat{\underline{f}}(k) = \int_{-\infty}^{+\infty} dt\; e^{ikt}\, \hat{f}(k) = \delta(k)\hat{f}(k)$$

i.e. the function, as expected, is constant and the only non-zero Fourier component is the DC. Suppose now we relax these invariance properties asking for robustness/approximate invariance to translations. One way is to integrate in the interval $[-a, a]$, which is also a more realistic scenario in translation augmentation where we shift the data in a finite interval. We have:

$$\underline{f}(w) = \int_{-a}^{a} dt\, f(w-t) = \int_{-\infty}^{+\infty} dt\; Ind_{[-a,a]}(t) f(w-t).$$

Taking the Fourier and using $\widehat{f(\cdot - t)} = e^{ikt}\hat{f}(k)$

$$\begin{aligned}\hat{\underline{f}}(k) &= \left(\int_{-\infty}^{\infty} dt\; e^{ikt} Ind_{[-a,a]}(t)\right)\hat{f}(k) \\ &= 2a\; sinc(2ka)\hat{f}(k)\end{aligned}$$

where we used the fact that the Fourier transform of an indicator function is the sinc function.
Thus the effect of averaging over an interval of translations is to dampen frequencies with a sinc function profile.

The next step is to consider a matrix function and a generic group $\mathcal{G}$. As we have seen in the previous sections a key property is that, due to the dot product structure $\langle W, x\rangle$, the transformation can be 'moved' from the input x to the weights W i.e. $\langle W, g_j x_i\rangle = \langle g_j^T W, x_i\rangle$. Thanks to this algebraic property we can then consider, in the case of data augmentation, g as acting on the network weights. Moreover, taking the Fourier transform, and using the identity [48]

$$\hat{\mathcal{L}}(gW)(K) = |\det g|^{-1}\hat{\mathcal{L}}((g^T)^{-1}K)$$

(where K indicated $2D$ frequencies), being that g is a unitary matrix the expression simplifies to

$$\hat{\mathcal{L}}(gW)(k) = \hat{\mathcal{L}}(gK)$$

since $det(g_i) = \pm 1$ and $(g^T)^{-1} = (g^T)^T = g$. Thus, the group $\mathcal{G}$ is equivalently acting on the frequency and space domain. We can therefore extend the results in the previous sections by considering instead of the Loss its Fourier transform. As a consequence, when analyzing the average spectrum of the learned weights, we expect: 1. in the case of translation augmentation, to see a shrinking in the spectrum of the learned weights, as argued with a simple example before; 2. in the case of rotation augmentation a rotationally symmetric spectrum; 3. in the case of scale augmentation a typical scale invariant $1/k$ spectrum and 4. in the case of mirror symmetry a mirror symmetric spectrum.

D. DATA AUGMENTATION AND STOCHASTIC ORBIT SAMPLING

In a real scenario not all orbits of the signals are available and therefore the loss in (6) only contains a few elements of the signal orbits. A common practice in ML is an "on the fly data" augmentation where the data are augmented using a randomly sampled transformation before taking the stochastic gradient descent step. To understand how the above theory adapts in such a scenario we show that sampling a few random orbit elements is sufficient for the results of the theorems and corollaries to hold. In particular we use a concentration inequality adapted to the sampling on the group. Let us define a stochastic sampled loss as:

$$\mathcal{L}_{st}(W; X, y) = \frac{1}{|B_J||B_{\mathcal{G}}|}\sum_{i\in B_J}\sum_{j\in B_{\mathcal{G}}} \ell(\sigma\langle W, g_j x_i\rangle, y_i), \quad (8)$$

where $B_J, B_{\mathcal{G}}$ are respectively the subset of randomly uniformly sampled signals and transformations and $|B_J|, |B_{\mathcal{G}}|$ their cardinality. Then, applying Hoeffding's inequality [49] gives :

$$Pr(|\mathcal{L}_{st}(W; X, y) - \mathcal{L}_a(W; X, y))| > \epsilon) < 2e^{-\frac{B_J B_{\mathcal{G}} \epsilon^2}{C}} \quad (9)$$

where C is a fixed constant assuming that the Loss is bounded. The equation points to the fact that if a few sampled data and transformations are available, the stochastic augmented loss is expected to be a good approximator of the augmented loss.

IV. EXPERIMENTAL VALIDATION

A. OPTIMIZATION

We employed a simple shallow non-linear network with ReLu non linearity trained on MNIST. We used a resized version of the MNIST images (64×64 instead of 28×28) for better visualization of the weights. The size of the hidden layer was fixed to 10 for simplicity. Empirically, a higher number of hidden units simply provided a sparser representation. We used an Adam optimizer [50] with maximum learning rate of $lr = 0.0001$ and batch size $bs = 1000$. We found that this set of hyperparameters choices allowed us to achieve stable training for all our experimental settings. We trained each model for a maximum of 100 epochs until convergence. The accuracy was varying highly from 60% to 96% due to the varying task difficulty and fixed number of hidden units. However this is not a problem within the scope of our study which does not aim to improve the accuracy nor propose a new algorithm but to study the impact of data symmetries on the learned weights. We considered, as explained in the introduction, the impact of two data manipulations: data augmentation and data symmetrization.

B. DATA AUGMENTATION

We employed four common forms of data augmentations: translation, rotation, scale and flip. We used standard Pytorch routines for augmentations where each image in the batch of data used in the gradient is transformed w.r.t. a randomly sampled transformation. Figure 1 shows how, as predicted in Corollary 1, the learned weights are invariant to the specific transformation used for data augmentation. Specifically, although approximately, the learned weights in the case of translation augmentation are planar waves of different directions and frequencies (b). Indeed, planar waves are invariant to translations of the size of the wave period. (c) reports the case of rotation augmentation: the weights have a clear rotation symmetry. Note that rotations, in the case of MNIST pose the problem of label confusion between the digit 6 and 9 which can be thought as one the rotated version of the other. However performing the training on a subset of MNIST that does not contain such digits did not show any significant difference in the symmetry of the learned weights. In the case of scale the visual interpretation is more difficult. To test for scale invariance we employed the well known fact that the spectrum of scale invariant images follows approximately a $1/k$ law. This is because for a given function h the Fourier transform of $h(ax)$, with a a scaling positive factor, is $(1/a)\hat{h}(k/a)$. Thus if $\hat{h}(k) = 1/k$ then the Fourier transform of the function is invariant to rescaling. To test for this we calculated the radial energy of the averaged spectrum of the filters and plotted it on a log scale: the profile should look like a straight line with slope -1. This is approximately true as reported in Figure 2. In the (e) panel are reported the learned weights when a flip augmentation is applied. As we can see, the weights are reflection symmetric.

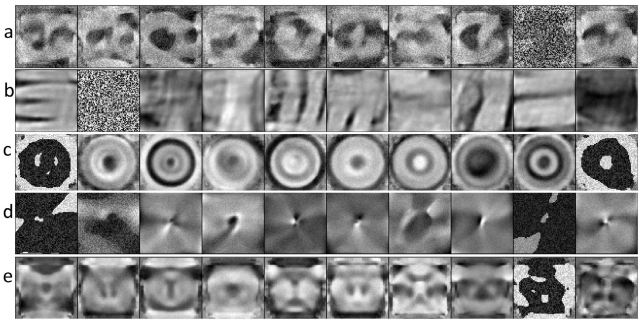


FIGURE 1. Learned weights in absence of data augmentation (a) and for, respectively, translation (b), rotation (c), scale (d) and horizontal flip (e) data augmentations.

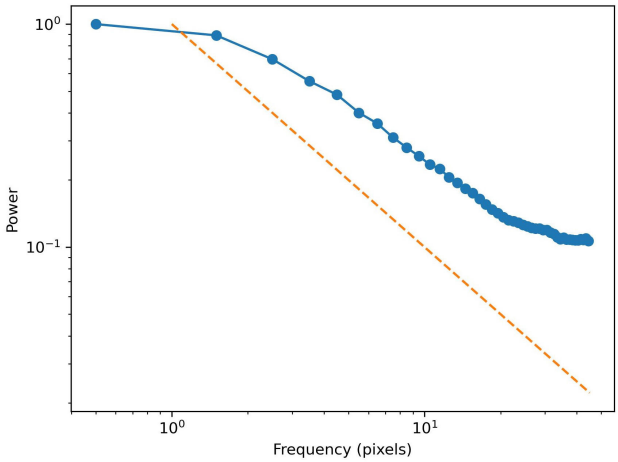


FIGURE 2. (Solid): Radial energy of the average spectrum of the filters learned with scale augmentation. (Dashed): Pure $1/f$ spectrum for comparison.

C. DATA SYMMETRIZATION

For data symmetrization we considered four different preparations of the dataset where each single image in MNIST is transformed to be symmetric (or approximately symmetric) w.r.t. the a specific transformation. In particular, for translation symmetry we simply replicated the central part of the image three times to implement a specific translation invariance (Figure 3, (a)). As shown in Figure 4 (a) the learned weights exactly reflect the symmetry as predicted by corollary 2. In the case of rotations we substituted each image of a digit with a superposition of the same digit rotated at many angles ([0, 10, $\cdots$, 360], Figure 3, (b)) creating in this way an approximately rotationally invariant image. As we see in Figure 4, (b) the filters are rotationally invariant although the phenomenon is less evident than in the case of data augmentation. In the case instead of scale (Figure 4 and 3, (c)) the spectral test (performed successfully in the case of scale augmentation) was inconclusive. We think this may be due to our incorrect strategy to generate a scale invariant image where few scaled versions of the same image are superposed. Increasing the number of superposed scaled images on the other hand drastically degraded the network performance.

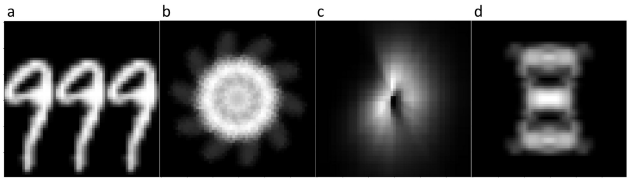


FIGURE 3. Single image (a 9) symmetry for translation (a), rotation (b), scale (c) and vertical-horizontal flip.

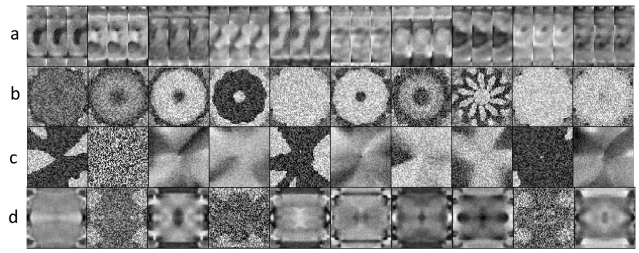


FIGURE 4. Learned weights for, respectively, translation (a), rotation (b), scale (c) and horizontal-vertical flip (d) single data symmetry.

For the flip symmetry instead (Figure 4 (d)) the filters are perfectly symmetric. Here we symmetrized the data both w.r.t. the vertical and horizontal axis performing the following steps:

$$x \to x + Hflip(x) \to x + Vflip(x)$$

to transform the original image into a vertical and horizontal reflection invariant image (3 (d)).

D. BIAS IN FOURIER

We tested how different data symmetrizations or augmentations have an impact on the average of the absolute value of the Fourier spectrum of the learned weights to confirm the theory predictions. Figure 5 shows that, in agreement with the theoretical results reported in IIIC, the averaged spectrum for translation augmentation (b) is significantly biased towards lower frequencies w.r.t. the not augmented case (a). Furthermore, as predicted, in the rotation augmentation case the spectrum has a rotational symmetry (c); In the case of scale symmetry (d), as illustrated in Figure 2 it shows an approximate scale invariance. Finally, in the case of flip transformations (e), the spectrum shows a vertical and horizontal axis of symmetry. For single data symmetrization the effect is less evident as shown in the second raw spectra of Figure 5. In particular for translations (b), the border effects, due to the symmetrization strategy are evident. For rotations (c) the invariance is less evident than in the case of data augmentation. For scale, differently from the augmentation case, the Fourier spectrum decay test failed to prove scale invariance (d). Instead, for the flip transformation, the Fourier transform is symmetric as predicted by the theory. We think that the origin of such partially positive results, as mentioned above, has to be found in the data symmetrization process which, as implemented at the moment, not only generates images that are not completely symmetric but are also very different from the original ones and significantly degrade performance in the trained network.

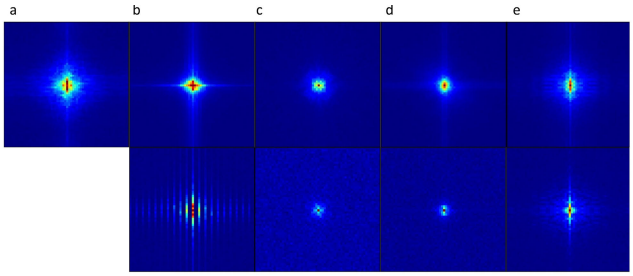


FIGURE 5. First raw:Averaged spectrum of the learned weights in absence of data augmentation (a) and for, respectively, translation (b), rotation (c), scale (d) and horizontal flip (e). Second raw: same but for single image symmetry.

V. CONCLUSION

This work studies how symmetries in the dataset, both at a global and at a single image level constrain the learned weights of a fully connected shallow non linear neural network. We derived results showing how symmetries are directly inherited in the learned weights and experimentally confirm our findings. We finally show how these symmetry constraints have a profound impact on the spectrum of the learned weights, an aspect of the so called network implicit bias. In the case of single-image symmetrization the results only partially confirm the theoretical predictions. We believe further work will be necessary to generate better symmetric data.

Although derived for a very simple architecture and dataset our work poses the basis to tackle the more complex question of understanding how the statistics of the training set are related to the nature of the learning weights. However, in the case of modern (oftentimes extremely complex) neural network architectures, a formal rigorous analysis would be very hard if not impossible. Moreover, many other types of data symmetries that do not have a group structure should be taken into consideration. In general it is difficult to characterize which type of symmetries the network is effectively learning when trained on a dataset where no a priori symmetries are imposed. This is not only difficult because those symmetries are unknown but also because what a network is learning depends on its implicit bias, which is also unknown. Learning and discovering such learned symmetries will be of great importance when interpret the information processing operated by the network. Furthermore, as mentioned in the introduction, weights symmetries that match data symmetries allow for a substantial reduction in the sample complexity of the learning. Those will be topics for future research.

ACKNOWLEDGMENT

Conflict of Interests The authors declare no conflict of interests.

REFERENCES

[1] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bull. Math. Biophys.*, vol. 5, no. 4, pp. 115–133, Dec. 1943.

[2] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biol. Cybern.*, vol. 36, no. 4, pp. 193–202, Apr. 1980.

[3] T. M. Caelli and Z.-Q. Liu, "On the minimum number of templates required for shift, rotation and size invariant pattern recognition," *Pattern Recognit.*, vol. 21, no. 3, pp. 205–216, Jan. 1988.

[4] R. Lenz, "Group invariant pattern recognition," *Pattern Recognit.*, vol. 23, nos. 1–2, pp. 199–217, Jan. 1990.

[5] P. Földiák, "Learning invariance from transformation sequences," *Neural Comput.*, vol. 3, no. 2, pp. 194–200, Jun. 1991.

[6] A. E. Grace and M. Spann, "A comparison between Fourier–Mellin descriptors and moment based features for invariant object recognition using neural networks," *Pattern Recognit. Lett.*, vol. 12, no. 10, pp. 635–643, Oct. 1991.

[7] J. Flusser and T. Suk, "Pattern recognition by affine moment invariants," *Pattern Recognit.*, vol. 26, no. 1, pp. 167–174, Jan. 1993.

[8] B. A. Olshausen, C. H. Anderson, and D. C. Van Essen, "A multiscale dynamic routing circuit for forming size- and position-invariant object representations," *J. Comput. Neurosci.*, vol. 2, no. 1, pp. 45–62, Mar. 1995.

[9] L. Van Gool, T. Moons, E. Pauwels, and A. Oosterlinck, "Vision and Lie's approach to invariance," *Image Vis. Comput.*, vol. 13, no. 4, pp. 259–277, May 1995.

[10] M. Michaelis and G. Sommer, "A lie group approach to steerable filters," *Pattern Recognit. Lett.*, vol. 16, no. 11, pp. 1165–1174, Nov. 1995.

[11] J. Wood, "Invariant pattern recognition: A review," *Pattern Recognit.*, vol. 29, no. 1, pp. 1–17, Jan. 1996.

[12] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex," *Nature Neurosci.*, vol. 2, no. 11, pp. 1019–1025, Nov. 1999.

[13] M. Lessmann and R. P. Würtz, "Learning invariant object recognition from temporal correlation in a hierarchical network," *Neural Netw.*, vol. 54, pp. 70–84, Jun. 2014.

[14] R. Gens and P. M. Domingos, "Deep symmetry networks," in *Proc. Adv. Neural Inf. Process. Syst. (NIPS)*, 2014, pp. 2537–2545.

[15] K. Lenc and A. Vedaldi, "Understanding image representations by measuring their equivariance and equivalence," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 991–999.

[16] Z. Shao and Y. Li, "Integral invariants for space motion trajectory matching and recognition," *Pattern Recognit.*, vol. 48, no. 8, pp. 2418–2432, Aug. 2015.

[17] F. Anselmi, L. Rosasco, and T. Poggio, "On invariance and selectivity in representation learning," *Inf. Inference, J. IMA*, vol. 5, no. 2, pp. 134–158, Jun. 2016.

[18] T. S. Cohen and M. Welling, "Group equivariant convolutional networks," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 48, 2016, pp. 1230–1242.

[19] S. Taco Cohen and M. Welling, "Steerable CNNs," in *Proc. Int. Conf. Learn. Represent.*, 2017.

[20] A. Achille and S. Soatto, "Emergence of invariance and disentanglement in deep representations," in *Proc. Inf. Theory Appl. Workshop (ITA)*, Feb. 2018, pp. 1–9.

[21] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 1597–1607.

[22] S. Chen, E. Dobriban, and J. Lee, "A group-theoretic framework for data augmentation," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds. Red Hook, NY, USA: Curran Associates, 2020, pp. 21321–21333.

[23] M. M. Bronstein, J. Bruna, T. Cohen, and P. Velickovic, "Geometric deep learning: Grids, groups, graphs, geodesics, and gauges," 2021, *arXiv:2104.13478*.

[24] N. Dehmamy, R. Walters, Y. Liu, D. Wang, and R. Yu, "Automatic symmetry discovery with lie algebra convolutional network," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds., 2021, pp. 2503–2515.

[25] C. Godfrey, D. Brown, T. Emerson, and H. Kvinge, "On the symmetries of deep learning models and their internal representations," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 11893–11905.

[26] S. Ravanbakhsh, J. Schneider, and B. Poczos, "Equivariance through parameter-sharing," in *Proc. 34th Int. Conf. Mach. Learn.*, vol. 70, 2017, pp. 2892–2901.

[27] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola, "Deep sets," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2017, pp. 3394–3404.

[28] M. Weiler and G. Cesa, "General E(2)-equivariant steerable CNNs," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 14357–14368.

[29] S. Soatto, "Steps towards a theory of visual information: Active perception, signal-to-symbol conversion and the interplay between sensing and control," 2011, *arXiv:1110.2053*.

[30] B. Haasdonk and H. Burkhardt, "Invariant kernel functions for pattern analysis and machine learning," *Mach. Learn.*, vol. 68, no. 1, pp. 35–61, May 2007.

[31] Y. S. Abu-Mostafa, "Learning from hints in neural networks," *J. Complex.*, vol. 6, no. 2, pp. 192–198, Jun. 1990.

[32] F. Anselmi, J. Z. Leibo, L. Rosasco, J. Mutch, A. Tacchetti, and T. Poggio, "Unsupervised learning of invariant representations," *Theor. Comput. Sci.*, vol. 633, pp. 112–121, Jun. 2016.

[33] A. Bietti, L. Venturi, and J. Bruna, "On the sample complexity of learning under geometric stability," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, M. Ranzato, A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, Eds. Red Hook, NY, USA: Curran Associates, 2021, pp. 18673–18684.

[34] J. Sokolic, R. Giryes, G. Sapiro, and M. R. D. Rodrigues, "Generalization error of invariant classifiers," in *Proc. AISTATS*, vol. 54, 2017, pp. 253–265.

[35] S. Zhang, J. Wang, X. Tao, Y. Gong, and N. Zheng, "Constructing deep sparse coding network for image classification," *Pattern Recognit.*, vol. 64, pp. 130–140, Apr. 2017.

[36] T. S. Cohen, M. Geiger, J. Köhler, and M. Welling, "Spherical CNNs," in *Proc. Int. Conf. Learn. Represent.*, vol. 48, 2018, pp. 2990–2999.

[37] S. Gunasekar, J. D. Lee, D. Soudry, and N. Srebro, "Implicit bias of gradient descent on linear convolutional networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2018, pp. 9461–9471.

[38] C. Yun, S. Krishnan, and H. Mobahi, "A unifying view on implicit bias in training linear neural networks," 2020, *arXiv:2010.02501*.

[39] J. Sahs, A. Damaraju, R. Pyle, O. Tavaslioglu, J. O. Caro, H. Y. Lu, and A. Patel, "A functional characterization of randomly initialized gradient descent in deep ReLU networks," Tech. Rep., 2020. [Online]. Available: https://openreview.net/forum?id=BJl9PRVKDS

[40] F. Williams, M. Trager, D. Panozzo, C. Silva, D. Zorin, and J. Bruna, "Gradient dynamics of shallow univariate ReLU networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8376–8385.

[41] B. Woodworth, S. Gunasekar, J. D. Lee, E. Moroshko, P. Savarese, I. Golan, D. Soudry, and N. Srebro, "Kernel and rich regimes in overparametrized models," 2020, *arXiv:2002.09277*.

[42] J. Sahs, R. Pyle, A. Damaraju, J. O. Caro, O. Tavaslioglu, A. Lu, and A. Patel, "Shallow univariate ReLU networks as splines: Initialization, loss surface, hessian, & gradient flow dynamics," 2020, *arXiv:2008.01772*.

[43] Z. Li, R. Wang, D. Yu, S. S. Du, W. Hu, R. Salakhutdinov, and S. Arora, "Enhanced convolutional neural tangent kernels," 2019, *arXiv:1911.00809*.

[44] S. Arora, S. S. Du, W. Hu, Z. Li, R. R. Salakhutdinov, and R. Wang, "On exact computation with an infinitely wide neural net," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8139–8148.

[45] S. Lang, *Algebra*. Cham, Switzerland: Springer, 2012.

[46] G. Landi and A. Zampini, *Linear Algebra and Analytic Geometry for Physical Sciences*. Cham, Switzerland: Springer, 2018.

[47] F. Anselmi, A. Patel, and L. Rosasco, "Neurally plausible mechanisms for learning selective and invariant representations," *J. Math. Neurosci.*, vol. 10, no. 1, pp. 1–15, Dec. 2020.

[48] N. Rahaman, A. Baratin, D. Arpit, F. Draxler, M. Lin, F. Hamprecht, Y. Bengio, and A. Courville, "On the spectral bias of neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 5301–5310.

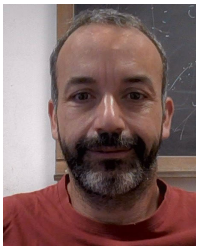
[49] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *J. Amer. Stat. Assoc.*, vol. 58, no. 301, pp. 13–30, Mar. 1963.

[50] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

FABIO ANSELMI was born in Tregnago, Verona, Italy, in 1976. He received the B.S. and M.S. degrees in theoretical physics from the University of Padova, in 2000, and the Ph.D. degree in quantum physics from Hertfordshire University, U.K., in 2004. From 2020 to 2022, he was an Assistant Professor with the Baylor College of Medicine, Houston, TX, USA. He is currently a Senior Researcher with the University of Trieste, Italy. In 2016, together with Tomaso Poggio, he published the book *Visual Cortex and Deep Networks: Learning Invariant Representations*. His research interests include edge between machine learning, in particular deep learning, computational neuroscience, and physics.

LUCA MANZONI received the Ph.D. degree in computer science from the University of Milano-Bicocca, in 2013. He is currently an Associate Professor with the University of Trieste, Italy. He has published more than 80 papers in international journals, conferences, and workshops. His research interests include natural computing models, such as P systems, reactions systems, and cellular automata and in the area of evolutionary computation, and genetic programming in particular. In 2012, he received a JSPS Postdoctoral Fellowship, and in 2017, he received an award as the Best Young Postdoctoral Researcher in computer science and mathematics with the University of Milano-Bicocca.

ALBERTO D'ONOFRIO received the M.Sc. degree in electrical engineering from the University of Pisa, in 1993, and the Ph.D. degree in medical computer science from Rome La Sapienza University, in 2000. He was the Group Leader in systems biomedicine with the European Institute of Oncology, Milan, Italy, from 2009 to 2014, where he was a Postdoctoral Reseracher (2000–2002) and then a Researcher (2000–2008), the Directeur de Recherche with the International Prevention Research Institute, Lyon, France, from 2014 to 2020, and a Visiting Professor with the Department of Mathematics and Statistics, University of Strathclyde, from 2017 to 2020. Since March 2022, he has been a Senior Researcher with the Department of Mathematics and Geosciences, University of Trieste, Italy. A pioneer of behavioral epidemiology of infectious diseases, his research interests include border between computer science and statistical physics, with applications to biomedicine.



ALEX RODRIGUEZ received the Ph.D. degree from the University of Barcelona, in 2012. He is currently a Senior Researcher in computer science with the University of Trieste. He was the ''Ludwig Boltzmann'' Senior Postdoctoral Fellow with the Condensed Matter and Statistical Physics Section, International Center for Theoretical Physics (ICTP), a Postdoctoral Fellow with the International School for Advanced Studies (SISSA), and a Professional Researcher with Universitat Politécnica de Catalunya (UPC). He is teaching in the Artificial Intelligence degree with the University of Trieste, in the SISSA Theoretical and Scientific Data Science Ph.D. Program, and the Master of High-Performance Computing (SISSA/ICTP) Program. His research interests include the interface between physics and data science: he wants to understand how the properties of data generated by physical simulations are related to the actual physical properties of the systems under study. He develops new machine learning methods (mostly unsupervised) that have been useful in many fields beyond physical sciences: clustering algorithms, density estimation, and manifold learning algorithms.

GIULIO CARAVAGNA received the Ph.D. degree in computer science from the University of Pisa, in 2011. He is currently an Associate Professor in computer science with the University of Trieste, where he leads the Cancer Data Science Laboratory. His laboratory works with the intersection of computational and experimental research, approaching real-world biological questions with state-of-the-art machine learning. He carried out postdoctoral training in systems biology with the University of Milan-Bicocca, machine learning for biology with The University of Edinburgh, and cancer genomics with the Institute of Cancer Research.

LUCA BORTOLUSSI received the Graduate degree in mathematics in Trieste, in 2003, and the Ph.D. degree in computer science from the University of Udine, in 2007.

In 2012, he was a Visiting Researcher with the School of Informatics, The University of Edinburgh. From 2013 to 2017, he was an Associate Researcher with CNR-ISTI, Pisa, within the QUANTICOL FP7 EU Project. From 2014 to 2015, he was a Professor in modeling and simulation with Saarland University. He was an Associate Professor (2015–2021) and an Assistant Professor (2006–2015) in computer science with the University of Trieste. He was a Guest Professor with Saarland University, in 2016, and (2018–2021). He is currently a Full Professor in computer science with the University of Trieste and leads the AI Laboratory. His research interests include the large realm of artificial intelligence, and lie at the boundary between symbolic and formal methods in computer science, statistical machine learning, modeling, simulation and control, cyber-physical systems, collective adaptive systems, explainable artificial intelligence, and a broad spectrum of applications in medicine, insurance, industry, sustainability, and climate change.

FRANCESCA CAIROLI received the B.Sc. degree in mathematics from the University of Milano-Bicocca, in 2013, and the M.Sc. degree in mathematics and the Ph.D. degree in computer science from the University of Trieste, in 2017 and 2022, respectively. She is currently an Assistant Professor with the Department of Mathematics and Geoscience, University of Trieste. Her research interests include to leverage the computational power of deep learning to tackle the scalability issues of formal methods and the simulation of complex systems.

Open Access funding provided by 'Università degli Studi di Trieste' within the CRUI CARE Agreement