On the Generalization of Equivariance and Convolution in Neural Networks to the Action of Compact Groups

Risi Kondor [1] **Shubhendu Trivedi** [2]

Abstract

Convolutional neural networks have been extremely successful in the image recognition domain because they ensure equivariance to translations. There have been many recent attempts to generalize this framework to other domains, including graphs and data lying on manifolds. In this paper we give a rigorous, theoretical treatment of convolution and equivariance in neural networks with respect to not just translations, but the action of any compact group. Our main result is to prove that (given some natural constraints) convolutional structure is not just a sufficient, but also a necessary condition for equivariance to the action of a compact group. Our exposition makes use of concepts from representation theory and noncommutative harmonic analysis and derives new generalized convolution formulae.

1. Introduction

One of the most successful neural network architectures is convolutional neural networks (CNNs) (LeCun et al., 1989). In the image recognition domain, where CNNs were originally conceived, convolution plays two crucial roles. First, it ensures that in any given layer, exactly the same filters are applied to each part of the image. Consequently, if the input image is translated, the activations of the network in each layer will translate the same way. This property is called *equivariance* (Cohen & Welling, 2016). Second, in conjunction with pooling, convolution ensures that each neuron's effective receptive field is a spatially contiguous domain. As we move higher in the network, these domains generally get larger, allowing the CNN to capture structure in images at *multiple different scales*.

[1]Departments of Statistics and Computer Science, The University of Chicago [2]Toyota Technological Institute at Chicago. Correspondence to: Risi Kondor <risi@cs.uchicago.edu>, Shubhendu Trivedi <shubhendu@ttic.edu>.

Proceedings of the 35^{th} International Conference on Machine Learning, Stockholm, Sweden, PMLR 80, 2018.

Recently, there has been considerable interest in extending neural networks to more exotic types of data, such as graphs or functions on manifolds (Niepert et al., 2016; Defferrard et al., 2016; Duvenaud et al., 2015; Li et al., 2016; Cohen et al., 2018; Monti et al., 2017; Masci et al., 2015). In these domains, equivariance and multiscale structure are just as important as for images, but finding the right notion of convolution is not obvious.

On the other hand, mathematics does offer a sweeping generalization of convolution tied in deeply with some fundamental ideas of abstract algebra: if G is a compact group and f and g are two functions $G \to \mathbb{C}$, then the convolution of f with g is defined

$$(f * g)(u) = \int_G f(uv^{-1})\, g(v)\, d\mu(v). \qquad (1)$$

Note the striking similarity of this formula to the ordinary notion of convolution, except that in the argument of f, $u - v$ has been replaced by the group operation uv^{-1}, and integration is with respect to the Haar measure, μ.

The goal of this paper is to relate (1) to the various looser notions of convolution used in the neural networks literature, and show that several practical neural networks implicitly already take advantange of the above group theoretic concept of convolution. In particular, we prove the following theorem (paraphrased here for simplicity).

Theorem 1. *A feed forward neural network $\mathcal{N}$ is equivariant to the action of a compact group G on its inputs if and only if each layer of $\mathcal{N}$ implements a generalized form of convolution derived from (1).*

To the best of our knowledge, this is the first time that the connection between equivariance and convolution in neural networks has been stated at this level of generality. The main technical challenge in our paper is that the activations in each layer of a neural net correspond to functions on a sequence of space *acted on* by G (called *homogeneous spaces* or *quotient spaces*) rather than functions on G itself. This necessitates a discussion of group convolution that is rather more thoroughgoing than is customary in pure algebra.

This paper does not present any new algorithms or neural

network architectures. Rather, its goal is to provide the language for thinking about generalized notions of equivariance and convolution in neural networks, and thereby facilitate the development of future architectures for data with non-trivial symmetries. To avoid interruptions in the flow of our exposition, we chose to first present the theory in its abstract form, and then illustrate it with examples in Section 6. For better understanding, the reader might choose to skip back and forth between these sections. One work that is close in spirit to the present paper but only considers discrete groups is (Ravanbakhsh et al., 2017).

2. Notation

In the following $[a]$ will denote the set $\{1, 2, \dots, a\}$. Given a set $\mathcal{X}$ and a vector space V, $L_V(\mathcal{X})$ will denote the space of functions $\{f\colon \mathcal{X} \to V\}$.

3. Equivariance in neural networks

A feed-forward neural network consists of some number of "neurons" arranged in $L+1$ distinct layers. Layer $\ell = 0$ is the input layer, where data is presented to the network, while layer $\ell = L$ is where the output is read out. Each neuron n_x^ℓ (denoting neuron number x in layer ℓ) has an *activation* f_x^ℓ. For the input layer, the activations come directly from the data, whereas in higher layers they are computed via a simple function of the activations of the previous layer, such as

$$f_x^\ell = \sigma\big(b_x^\ell + \textstyle\sum_y w_{x,y}^\ell\, f_y^{\ell-1}\big). \tag{2}$$

Here, the $\{b_x^\ell\}$ bias terms and the $\{w_{x,y}^\ell\}$ weights are the network's learnable parameters, while σ is a fixed nonlinear function, such as the ReLU function $\sigma(z) = \max(0, z)$. In the simplest case, each f_x^ℓ is a scalar, but, in the second half of the paper we consider neural networks with more general, vector or tensor valued activations.

For the purposes of the following discussion it is actually helpful to take a slightly more abstract view, and, instead of focusing on the individual activations, consider the activations in any given layer collectively as a function $f^\ell\colon \mathcal{X}_\ell \to V_\ell$, where $\mathcal{X}_\ell$ is a set indexing the neurons and V_ℓ is a vector space. Omitting the bias terms in (2) for simplicity, each layer $\ell = 1, 2, \dots, L$ can then just be thought of as implementing a linear transformation $\phi_\ell\colon L_{V_{\ell-1}}(\mathcal{X}_{\ell-1}) \to L_{V_\ell}(\mathcal{X}_\ell)$ followed by the pointwise nonlinearity ξ. Our operational definition of neural networks for the rest of this paper will be as follows.

Definition 1. *Let $\mathcal{X}_0, \dots, \mathcal{X}_L$ be a sequence of index sets, $V_0, \dots, V_L$ vector spaces, $\phi_1, \dots, \phi_L$ linear maps*

$$\phi_\ell\colon L_{V_{\ell-1}}(\mathcal{X}_{\ell-1}) \longrightarrow L_{V_\ell}(\mathcal{X}_\ell),$$

*and $\sigma_\ell\colon V_\ell \to V_\ell$ appropriate pointwise nonlinearities, such as the ReLU operator. The corresponding **multi-layer feed-forward neural network (MFF-NN)** is then a sequence of maps $f_0 \mapsto f_1 \mapsto f_2 \mapsto \dots \mapsto f_L$, where $f_\ell(x) = \sigma_\ell(\phi_\ell(f_{\ell-1})(x))$.*

If we are interested in constructing a neural net for recognizing $m \times m$ pixel images, it is tempting to take $\mathcal{X}_0 = [m] \times [m]$ and define $\mathcal{X}_1, \dots, \mathcal{X}_L$ similarly. However, again for notational simplicity, we extend each of these index sets to the entire integer plane $\mathbb{Z}^2$, and simply assume that outside of the square region $[m] \times [m]$, $f^0(x_1, x_2) = 0$. A traditional *convolutional neural network (CNN)* is a network of this type where the ϕ_ℓ functions are constrained to have the special form

$$\phi_\ell(f_{\ell-1})(x_1, x_2) = \sum_{u_1=1}^{w} \sum_{u_2=1}^{w} f_{\ell-1}(x_1 - u_1, x_2 - u_2)\; \chi_\ell(u_1, u_2). \tag{3}$$

The above function is known as the *discrete convolution* of $f^{\ell-1}$ with the *filter* χ, and is usually denoted $f_{\ell-1} * \chi_\ell$. In most CNNs the width w of the filters is quite small, on the order of $3 \sim 10$, while the number of layers can be as small as 3 or as large as a few dozen.

Some of the key features of CNNs are immediately apparent just from the convolution formula (3):

1. The number of parameters in CNNs is much smaller than in general (fully connected) feed-forward networks, since we only have to learn the w^2 numbers defining the χ_ℓ filters rather than $O((m^2)^2)$ weights.
2. (3) applies the same filter to every part of the image. Therefore, if the networks learns to recognize a certain feature, e.g., eyes, in one part of the image, then it will be able to do so in any other part as well.
3. Equivalently to the above, if the input image is translated by any vector (t_1, t_2) (i.e., $f^{0\prime}(x_1, x_2) = f^0(x_1 - t_1, x_2 - t_2)$), then all higher layers will translate in exactly the same way. This property is called **equivariance** (sometimes *covariance*) to translations.

The goal of the present paper is to understand the mathematical generalization of the above properties to other domains, such as graphs, manifolds, and so on.

3.1. Group actions

The jumping off point to our analysis is the observation that the above is a special case of the following scenario.

1. We have a set $\mathcal{X}$ and a function $f\colon \mathcal{X} \to \mathbb{C}$.
2. We have a group G acting on $\mathcal{X}$. This means that each $g \in G$ has a corresponding transformation $T_g\colon \mathcal{X} \to \mathcal{X}$, and for any $g_1, g_2 \in G$, $T_{g_2 g_1} = T_{G_2} \circ T_{G_1}$.
3. The action of G on $\mathcal{X}$ extends to functions on $\mathcal{X}$ by
$$\mathbb{T}_g\colon f \mapsto f' \qquad f'(T_g(x)) = f(x).$$

In the case of translation invariant image recognition, $\mathcal{X} = \mathbb{Z}^2$, G is the group of integer translations, which is isomorphic to $\mathbb{Z}^2$ (note that this is a very special case, in general

$\mathcal{X}$ and G are different objects), the action is

$$T_{(t_1,t_2)}(x_1, x_2)) = (x_1 + t_1, x_2 + t_2) \qquad (t_1, t_2) \in \mathbb{Z}^2,$$

and the corresponding (induced) action on functions is

$$\mathbb{T}\colon f \mapsto f' \qquad f'(x_1, x_2) = f(x_1 - t_1, x_2 - t_2).$$

We give several other (more interesting) examples of group actions in Section 6, but for now continue with our abstract development. Also note that to simplify notation, in the following, where this does not cause confusion, we will simply write group actions as $x \mapsto g(x)$ rather than the more cumbersome $x \mapsto T_g(x)$.

Most of the actions considered in this paper have the property that taking any $x_0 \in \mathcal{X}$, for any other $x \in \mathcal{X}$ can be reached by the action of some $g \in G$, i.e., $x = g(x_0)$. This property is called **transitivity**, and if the action of G on $\mathcal{X}$ is transitive, we say that $\mathcal{X}$ is a **homogeneous space** of G.

3.2. Equivariance

Equivariance is a concept that applies very broadly, whenever we have a group acting on a pair of spaces and there is a map from functions on one to functions on the other.

Definition 2. *Let G be a group and $\mathcal{X}_1, \mathcal{X}_2$ be two sets with corresponding G-actions*

$$T_g\colon \mathcal{X}_1 \to \mathcal{X}_1, \qquad T'_g\colon \mathcal{X}_2 \to \mathcal{X}_2.$$

*Let V_1 and V_2 be vector spaces, and $\mathbb{T}$ and $\mathbb{T}'$ be the induced actions of G on $L_{V_1}(\mathcal{X}_1)$ and $L_{V_2}(\mathcal{X}_2)$. We say that a (linear or non-linear) map $\phi\colon L_{V_1}(\mathcal{X}_1) \to L_{V_2}(\mathcal{X}_2)$ is **equivariant** with the action of G (or G**–equivariant** for short) if*

$$\phi(\mathbb{T}_g(f)) = \mathbb{T}'_g(\phi(f)) \qquad \forall f \in L_{V_1}(\mathcal{X}_1)$$

for any group element $g \in G$.

Equivariance is represented graphically by a so-called commutative diagram, in our case

$$\begin{array}{ccc} L_{V_1}(\mathcal{X}_1) & \xrightarrow{\mathbb{T}_g} & L_{V_1}(\mathcal{X}_1) \\ \downarrow{\scriptstyle \phi} & & \downarrow{\scriptstyle \phi} \\ L_{V_2}(\mathcal{X}_2) & \xrightarrow{\mathbb{T}'_g} & L_{V_2}(\mathcal{X}_2) \end{array}$$

We are finally in a position to define the objects that we study in this paper, namely generalized equivariant neural networks.

Definition 3. *Let $\mathcal{N}$ be a feed-forward neural network as defined in Definition 1, and G be a group that acts on each index space $\mathcal{X}_0, \dots, \mathcal{X}_L$. Let $\mathbb{T}^0, \mathbb{T}^1, \dots, \mathbb{T}^L$ be the corresponding actions on $L_{V_0}(\mathcal{X}_0), \dots, L_{V_L}(\mathcal{X}_L)$. We say that $\mathcal{N}$ is a G–**equivariant feed-forward network** if, when the inputs are transformed $f_0 \mapsto \mathbb{T}^0_g(f_0)$ (for any $g \in G$), the activations of the other layers correspondingly transform as $f_\ell \mapsto \mathbb{T}^\ell_g(f_\ell)$.*

It is important to note how general the above framework is. In particular, we have not said whether G and $\mathcal{X}_0, \dots, \mathcal{X}_L$ are discrete or continuous. In any actual implementation of a neural network, the index sets would of course be finite. However, it has been observed before that in certain cases, specifically when $\mathcal{X}_0$ is an object such as the sphere or other manifold which does not have a discretization that fully takes into account its symmetries, it is easier to describe the situation in terms of abstract "continuous" neural networks than seemingly simpler discrete ones (Cohen et al., 2018).

Note also that invariance is a special case of equivariance, where $T_g = \mathrm{id}$ for all g. In fact, this is another major reason why equivariant architectures are so prevalent in the literature: any equivariant network can be turned into a G–invariant network simply by tacking on an extra layer that is equivariant in this degenerate sense (in practice, this often means either averaging or creating a histogram of the activations of the last layer). Nowhere is this more important than in graph learning, where it is a hard constraint that whatever representation is learnt by a neural network, it must be invariant to reordering the vertices. Today's state of the art solution to this problem are message passing networks (Gilmer et al., 2017), whose invariance behavior we discuss in section 6. Another architecture that achieves invariance by stacking equivariant layers followed by a final invariant one is that of scattering networks (Mallat, 2012).

4. Convolution on groups and quotient spaces

According to its usual definition in signal processing, the **convolution** of two functions $f, g\colon \mathbb{R} \to \mathbb{R}$ is

$$(f * g)(x) = \int f(x - y)\, g(y)\, dy. \tag{4}$$

Intuitively, we can think of f as a template and g as a modulating function (or the other way round, since convolution on $\mathbb{R}$ is commutative): we get $f * g$ by a placing a "copy" of f at each point on the x axis, but scaled by the value of g at that point, and superimposing the results. The discrete variant of (4) for $f, g\colon \mathbb{Z} \to \mathbb{R}$ is of course

$$(f * g)(x) = \sum_{y \in \mathbb{Z}} f(x - y)\, g(y), \tag{5}$$

and both the above formulae have natural generalizations to higher dimensions. In particular, (3) is just the two dimensional version of (5) with a limited width filter.

What we are interested in for this paper, however, is the much broader generalization of convolution to the case when f and g are functions on a compact group G. As already mentioned in the introduction, this takes the form

$$(f * g)(u) = \int_G f(uv^{-1})\, g(v)\, d\mu(v). \tag{6}$$

Note that (6) only differs from (4) in that $x-y$ is replaced by the group operation uv^{-1}, which is not surprising, since the group operation on $\mathbb{R}$ in fact is exactly $(x,y)\mapsto x+y$, and the "inverse" of y in the group sense is $-y$. Furthermore, the Haar measure μ makes an appearance. At this point, the main reason that we restrict ourselves to compact groups is because this guarantees that μ is essentially unique[1]. The discrete counterpart of (6) for countable (including finite) groups is

$$(f*g)(u)=\sum_{v\in G} f(uv^{-1})\,g(v). \tag{7}$$

All these definitions are standard and have deep connections to the algebraic properties of groups. In contrast, the various extensions of convolution to homogeneous spaces that we derive below are not often discussed in pure algebra.

4.1. Convolution on quotient spaces

The major complication in neural networks is that $\mathcal{X}_0,\dots,\mathcal{X}_L$ (which are the spaces that the $f_0,\dots,f_L$ activations are defined on) are homogeneous spaces of G, rather than being G itself. Fortunately, the strong connection between the structure of groups and their homogeneous spaces (see boxed text) allows generalizing convolution to this case as well. Note that from now on, to keep the exposition as simple as possible, we present our results assuming that G is countable (or finite). The generalization to continuous groups is straightforward. We also allow all our functions to be complex valued, because representation theory itself, which is the workhorse behind our results, is easiest to formulate over $\mathbb{C}$.

Definition 4. *Let G be a finite or countable group, $\mathcal{X}$ and $\mathcal{Y}$ be (left or right) quotient spaces of G, $f\colon \mathcal{X}\to\mathbb{C}$, and $g\colon \mathcal{Y}\to\mathbb{C}$. We then define the **convolution** of f with g as*

$$(f*g)(u)=\sum_{v\in G} f{\uparrow}^G(uv^{-1})\,g{\uparrow}^G(v),\qquad u\in G. \tag{8}$$

This definition includes $\mathcal{X}=G$ or $\mathcal{Y}=G$ as special cases, since any group is a quotient space of itself with respect to the trivial subgroup $H=\{e\}$.

Definition 4 hides the facts that depending on the choice of $\mathcal{X}$ and $\mathcal{Y}$: (a) the summation might only have to extend over a quotient space of G rather than the entire group, (b) the result $f*g$ might have symmetries that effectively make it a function on a quotient space rather than G itself (this is exactly what the case will be in generalized convolutional networks). Therefore we now discuss three special cases.

[1]Non-compact groups would also cause trouble because their representation theory is much more involved. $\mathbb{R}^2$, which is the group behind traditional CCNs, is of course not compact. The reason that it is still amenable to our analysis (with small modifications) is that it belongs to a handful of families of exceptional non-compact groups that are "easy".

ESSENTIAL DEFINITIONS FOR QUOTIENT SPACES

Certain connections between the structure of a group G and its homogeneous space $\mathcal{X}$ are crucial for our exposition. First, by definition, fixing an "origin" $x_0\in\mathcal{X}$, any $x\in\mathcal{X}$ can be reached as $x=g(x_0)$ for some $g\in G$. This allows us to "index" elements of $\mathcal{X}$ by elements of G. Since we use this mechanism so often, we introduce the shorthand $[g]_{\mathcal{X}}=g(x_0)$, which hides the dependence on the (arbitrary) choice of x_0.

Second, elementary group theory tells us that the set of group elements that fix x_0 actually form a subgroup H. By further elementary results (see Appendix), the set of group elements that map $x_0\mapsto x$ is a so-called **left coset** $gH:=\{gh\mid h\in H\}$. The set of all such cosets forms the **(left) quotient space** G/H. Therefore, $\mathcal{X}$ can be identified with G/H.

Now for each gH coset we may pick a **coset representative** $g'\in gH$, and let $\overline{x}$ denote the representative of the coset of group elements that map x_0 to x. Note that while the map $g\mapsto [g]_{G/H}$ is well defined, the map $x\mapsto\overline{x}$ going in the opposite direction is more arbitary, since it depends on the choice of coset representatives.

The **right quotient space** $H\backslash G$ is similarly defined as the space of **right cosets** $Hg:=\{hg\mid h\in H\}$. Furthermore, if K is another subgroup of G, we can talk about **double cosets** $HgK=\{hgk\mid h\in H,k\in K\}$ and the corresponding space $H\backslash G/K$.

Given $f\colon G\to\mathbb{C}$, we define its **projection** to $\mathcal{X}=G/H$

$$f{\downarrow_{\mathcal{X}}}\colon \mathcal{X}\to\mathbb{C}\qquad f{\downarrow_{\mathcal{X}}}(x)=\frac{1}{|H|}\sum_{g\in\overline{x}H} f(g).$$

Conversely, given $f\colon\mathcal{X}\to\mathbb{C}$, we define the **lifting** of f to G

$$f{\uparrow}^G\colon G\to\mathbb{C}\qquad f{\uparrow}^G(g)=f([g]_{\mathcal{X}}).$$

Projection and lifting to/from right quotient spaces and double quotient spaces is defined analogously.

CASE I: $\mathcal{X}=G$ AND $\mathcal{Y}=G/H$

When $f\colon G\to\mathbb{C}$ but $g\colon G/H\to\mathbb{C}$ for some subgroup H of G, (8) reduces to

$$(f*g)(u)=\sum_{v\in G} f(uv^{-1})\,g{\uparrow}^G(v).$$

Plugging $u'=uh$ into this formula (for any $h\in H$) and changing the variable of summation to $w:=vh^{-1}$ gives

$$\begin{aligned}(f*g)(u')&=\sum_{v\in G} f(uhv^{-1})\,g{\uparrow}^G(v)\\&=\sum_{w\in G} f(uw^{-1})\,g{\uparrow}^G(wh).\end{aligned}$$

However, since w and wh are in the same left H–coset, $g{\uparrow}^G(wh)=g{\uparrow}^G(w)$, so $(f*g)(u')=(f*g)(u)$, i.e.,

$f * g$ is constant on left H–cosets. This makes it natural to interpret $f * g$ as a function on G/H rather than the full group. Thus, we have the following definition.

If $f\colon G \to \mathbb{C}$, and $g\colon G/H \to \mathbb{C}$ then $f * g\colon G/H \to \mathbb{C}$ with

$$(f * g)(x) = \sum_{v \in G} f(\overline{x}v^{-1})\, g([v]_{G/H}). \tag{9}$$

CASE II: $\mathcal{X} = G/H$ AND $\mathcal{Y} = H\backslash G$

When $f\colon G/H \to \mathbb{C}$, but $g\colon G \to \mathbb{C}$, (8) reduces to

$$(f * g)(u) = \sum_{v \in G} f{\uparrow}^G(uv^{-1})\, g(v). \tag{10}$$

This time it is not $f{*}g$, but g that shows a spurious symmetry. Letting $v' = hv$ (for any $h \in H$), by the right H–invariance of $f{\uparrow}^G$, $f{\uparrow}^G(uv'^{-1}) = f{\uparrow}^G(uv^{-1}h^{-1}) = f{\uparrow}^G(uv)$. Considering that any v can be uniquely written as $v = h\overline{y}$, where $\overline{y}$ is the representative of one of its cosets, while $h \in H$, we get that (10) factorizes in the form

$$\begin{aligned}(f * g)(u) &= \sum_{y \in H\backslash G} f{\uparrow}^G(u\overline{y}^{-1}) \sum_{h \in H} g(h\overline{y}) \\ &= \sum_{y \in H\backslash G} f{\uparrow}^G(u\overline{y}^{-1})\, \tilde{g}(y),\end{aligned}$$

where $\tilde{g}(y) := \sum_{h \in H} g(h\overline{y})$. In other words, without loss of generality we can take g to be a function on $H\backslash G$ rather than the full group.

If $f\colon G/H \to \mathbb{C}$, and $g\colon H\backslash G \to \mathbb{C}$, then $f * g\colon G \to \mathbb{C}$ with

$$(f * g)(u) = |H| \sum_{y \in H\backslash G} f([u\overline{y}^{-1}]_{G/H})\, g(y). \tag{11}$$

CASE III: $\mathcal{X} = G/H$ AND $\mathcal{Y} = H\backslash G/K$

Finally, we consider the case when $f\colon G/H \to \mathbb{C}$ and $g\colon G/K \to \mathbb{C}$ for two subgroups H, K of G, which might or might not be the same. This combines features of the above two cases in the sense that, similarly to Case I, setting $u' = uk$ for any $k \in K$ and letting $w = vk^{-1}$,

$$\begin{aligned}(f * g)(u') &= \sum_{v \in G} f{\uparrow}^G(u'v^{-1})\, g{\uparrow}^G(v) = \\ &= \sum_{v \in G} f{\uparrow}^G(ukv^{-1})\, g{\uparrow}^G(v) = \sum_{w \in G} f{\uparrow}^G(uw^{-1})\, g{\uparrow}^G(wk) \\ &= \sum_{w \in G} f{\uparrow}^G(uw^{-1})\, g{\uparrow}^G(w) = (f * g)(u),\end{aligned}$$

showing that $f * g$ is right K–invariant, and therefore can be regarded as a function $G/K \to \mathbb{C}$. At the same time, similarly to (10), letting $v = h\overline{y}$,

$$\begin{aligned}(f * g)(u) &= \sum_{y \in H\backslash G} f{\uparrow}^G(u\overline{y}^{-1}) \sum_{h \in H} g{\uparrow}^G(h\overline{y}) \\ &= \sum_{y \in H\backslash G} f{\uparrow}^G(u\overline{y}^{-1})\, \tilde{g}(y),\end{aligned}$$

where $\tilde{g}(y) := \sum_{h \in H} g(h\overline{y})$, which is left H–invariant. Therefore, without loss of generality, we can take g to be a function $H\backslash G/K \to \mathbb{C}$.

If $f\colon G/H \to \mathbb{C}$, and $g\colon H\backslash G/K \to \mathbb{C}$ then we define the **convolution** of f with g as $f * g\colon G/K \to \mathbb{C}$ with

$$(f * g)(x) = |H| \sum_{y \in H\backslash G} f([\overline{x}\overline{y}^{-1}]_{\mathcal{X}})\, g([\overline{y}]_{H\backslash G/K}). \tag{12}$$

Since $f \mapsto f * g$ is a map from one homogeneous space, $\mathcal{X} = G/H$, to another homogeneous space, $\mathcal{Y} = H/K$, it is this last defintion that will be of most relevance to us in constructing neural networks.

4.2. Relationship to Fourier analysis

The nature of convolution on homogeneous spaces is further explicated by considering its form in Fourier space (see (Terras, 1999)). Recall that the **Fourier transform** of a function f on a countable group is defined as

$$\widehat{f}(\rho_i) = \sum_{u \in G} f(u)\, \rho_i(u), \qquad i = 0, 1, 2, \ldots, \tag{13}$$

where $\rho_0, \rho_1, \ldots$ are matrix valued functions called **irreducible representations** or **irreps** of G (see Appendix for details). As expected, the generalization of this to the case when f is a function on G/H, $H\backslash G$ or $H\backslash G/K$ is

$$\widehat{f}(\rho_i) = \sum_{u \in G} \rho_i(u)\, f{\uparrow}^G(u), \qquad i = 1, 2, \ldots.$$

Analogous formulae hold for continuous groups, involving integration with respect to the Haar measure.

At first sight it might be surprising that the Fourier transform of a function on a quotient space consists of the same number of matrices of the same sizes as the Fourier transform of a function on G itself, since G/H, $H\backslash G$ or $H\backslash G/K$ are smaller objects than G. This puzzle is resolved by the following proposition, which tells us that in the latter cases, the Fourier matrices have characteristic sparsity patterns.

Proposition 1. *Let ρ be an irrep of G, and assume that on restriction to H it decomposes into irreps of H in the*

form $\rho|_H = \mu_1 \oplus \mu_2 \oplus \ldots \oplus \mu_k$. Let $\widehat{f}$ be the Fourier transform of a function $f\colon G/H \to \mathbb{C}$. Then $[\widehat{f}(\rho)]_{*,j} = 0$ unless the block at column j in the decomposition of $\rho|_H$ is the trivial representation. Similarly, if $f\colon H\backslash G \to \mathbb{C}$, then $[\widehat{f}(\rho)]_{i,*} = 0$ unless the block of $\rho|_H$ at row i is the trivial representation. Finally, if $f\colon H\backslash G/K \to \mathbb{C}$, then $[\widehat{f}(\rho)]_{i,j} = 0$ unless the block of $\rho|_H$ at row i is the trivial representation of H and *the block at column j in the decomposition of $\rho|_K$ is the trivial representation of K.*

Schematically, this proposition implies that in the three different cases, the Fourier matrices have three different forms of sparsity:

G/K $\quad$ $H\backslash G$ $\quad$ $H\backslash G/K$

Fortuitously, just like in the classical, Euclidean case, convolution also takes on a very nice form in the Fourier domain, even when f or g (or both) are defined on homogeneous spaces.

Proposition 2 (Convolution theorem on groups)**.** *Let G be a compact group, H and K subgroups of G, and f, g be complex valued functions on G, G/H, $H\backslash G$ or $H\backslash G/K$. In any combination of these cases,*

$$\widehat{f * g}(\rho_i) = \widehat{f}(\rho_i)\, \widehat{g}(\rho_i) \tag{14}$$

for any given system of irreps $\mathcal{R}_G = \{\rho_0, \rho_1, \ldots\}$.

Plugging in matrices with the appropriate sparsity patterns into (14) now gives us an intuitive way of thinking about Case I–III above.

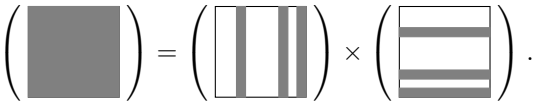
CASE I: $\mathcal{X} = G$ **AND** $\mathcal{Y} = G/H$

Mutiplying a column sparse matrix with a dense matrix from the left gives a column sparse matrix with the same pattern, therefore $f * g$ is a function on G/H:

$\widehat{f * g}(\rho)$ $\quad$ $\widehat{f}(\rho)$ $\quad$ $\widehat{g^{\uparrow G}}(\rho)$

CASE II: $\mathcal{X} = G/H$ **AND** $\mathcal{Y} = H\backslash G$

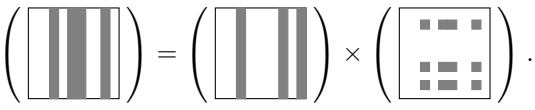
Multiplying a column sparse matrix from the right by another matrix picks out the corresponding rows of the second matrix. Therefore, if f is a function on G/H, then w.l.o.g. we can take g to be a function on $H\backslash G$.



$\widehat{f * g}(\rho)$ $\quad$ $\widehat{f^{\uparrow G}}(\rho)$ $\quad$ $\widehat{g^{\uparrow G}}(\rho)$

CASE III: $f\colon G/H \to \mathbb{C}$ **AND** $g\colon H\backslash G/K \to \mathbb{C}$

Finally, if f is a function on G/H, and we want to make $f * g$ to be a function on G/K, then we should take $g\colon H\backslash G/K$:



$\widehat{f * g}(\rho)$ $\quad$ $\widehat{f^{\uparrow G}}(\rho)$ $\quad$ $\widehat{g^{\uparrow G}}(\rho)$

5. Main result: the connection between convolution and equivariance

We are finally in a position to define the notion of generalized convolutional networks, and state our main result connecting convolutions and equivariance.

Definition 5. *Let G be a compact group and $\mathcal{N}$ an $L+1$ layer feed-forward network in which the i'th index set is G/H_i for some subgroup H_i of G. We say that $\mathcal{N}$ is a G-**convolutional neural network** (or* **G-CNN** *for short) if each of the linear maps $\phi_1, \ldots, \phi_L$ in $\mathcal{N}$ is a generalized convolution (see Definition 4) of the form*

$$\phi_\ell(f_{\ell-1}) = f_{\ell-1} * \chi_\ell$$

with some filter $\chi_\ell \in L_{V_{\ell-1} \times V_\ell}(H_{\ell-1}\backslash G/H_\ell)$.

Theorem 1. *Let G be a compact group and $\mathcal{N}$ be an $L+1$ layer feed-forward neural network in which the ℓ'th index set if of the form $\mathcal{X}_\ell = G/H_\ell$, where H_ℓ is some subgroup of G. Then $\mathcal{N}$ is equivariant to the action of G in the sense of Definition 3 if and only if it is a G-CNN.*

Proving this theorem in the forward direction is relatively easy and only requires some elementary facts about cosets and group actions.

Proof of Theorem 1 (forward direction). Assume that we translate $f_{\ell-1}$ by some group element $g \in G$ and get $f'_{\ell-1}$, i.e., $f'_{\ell-1} = \mathbb{T}_g^{\ell-1}(f_{\ell-1})$, where $f'_{\ell-1}(x) = f_{\ell-1}(g^{-1}x)$. Then

$$\begin{aligned}\phi_\ell(f'_{\ell-1})(u) &= (f'_{\ell-1} * \chi_\ell)(u) \\ &= \sum_{v\in G} f'_{\ell-1}([uv^{-1}]_\mathcal{X})\,\chi_\ell(v) \\ &= \sum_{v\in G} f_{\ell-1}(g^{-1}([uv^{-1}]_\mathcal{X}))\,\chi_\ell(v).\end{aligned}$$

By $g^{-1}([uv^{-1}]_\mathcal{X}) = [g^{-1}uv^{-1}]_\mathcal{X}$ this is further equal to

$$\begin{aligned}&\sum_{v\in G} f_{\ell-1}([g^{-1}uv^{-1}]_\mathcal{X})\,\chi_\ell(v) \\ &\qquad = (f_{\ell-1} * \chi_\ell)(g^{-1}u) = \phi_\ell(f_{\ell-1})(g^{-1}u).\end{aligned}$$

Therefore, $\phi_\ell(f_{\ell-1})$ is equivariant with $f_{\ell-1}$. Since σ_ℓ is a pointwise operator, so is $f_\ell = \sigma_\ell(\phi_\ell(f_{\ell-1}))$. By induction on ℓ, using the transitivity of equivariance, this implies that every layer of $\mathcal{N}$ is equivariant with layer 0. Note that this proof holds not only in the base case, when each f_ℓ is a function $\mathcal{X} \to \mathbb{C}$, but also in the more general case when $f_\ell \colon \mathcal{X}_\ell \to V_\ell$ and the filters are $\chi_\ell \colon \mathcal{X}_\ell \to V_{\ell-1} \times V_\ell$. ■

Proving the "only if" part of Theorem 1 is highly technical, therefore we leave it to the Appendix.

6. Examples of algebraic convolution in neural networks

We are not aware of any prior papers that have exposed the above algebraic theory of equivariance and convolution in its full generality. However, there are a few recent publications that implicitly exploit these ideas in specific contexts.

6.1. Rotation equivariant networks

In image recognition applications it is a natural goal to achieve equivariance to both translation and rotation. The most common approach is to use CNNs, but with filters that are replicated at a certain number of rotational angles (typically multiples of 90 degrees), connected in such as a way as to achieve a generalization of equivariance called *steerability*. Steerability also has a group theoretic interpretation, which is most lucidly explained in (Cohen & Welling, 2017).

The recent papers (Marcos et al., 2017) and (Worrall et al., 2017) extend these architectures by considering continuous rotations at each point of the visual field. Thus, putting aside the steerability aspect for now and only considering the behavior of the network at a single point, both these papers deal with the case where $G = \mathrm{SO}(2)$ (the two dimensional rotation group) and $\mathcal{X}$ is the circle S^1. The group $\mathrm{SO}(2)$ is commutative, therefore its irreducible representations are one dimensional, and are, in fact, $\rho_j(\theta) = e^{2\pi\iota j\theta}$, where $\iota = \sqrt{-1}$. While not calling it a group Fourier transform, Worrall et al. (2017) explicitly expand the local activations in this basis and scale them with weights, which, by virtue of Proposition 2, amounts to convolution on the group, as prescribed by our main theorem.

The form of the nonlinearity in (Worrall et al., 2017) is different from that prescribed in Definition 3, which leads to a coupling between the indices of the Fourier components in any path from the input layer to the output layer. This is compensated by what they call their "equivariance condition", asserting that only Fourier components for which $M = \sum_\ell j_\ell$ is the same may mix. This restores equivariance in the last layer, but analyzing it group theoretically is beyond the scope of the present paper.

6.2. Spherical CNNs

Closest in spirit to the present work is the recent paper (Cohen et al., 2018), which proposes a convolutional architecture for recognizing images painted on the sphere, satisfying equivariance with respect to rotations of the sphere. Thus, in this case, $G = \mathrm{SO}(3)$, the group of three dimensional rotations, and $\mathcal{X}_\ell$ is the sphere, S^2.

The case of rotations acting on the sphere is one of the textbook examples of continuous group actions. In particular, letting x_0 be the North pole, we see that two-dimensional rotations in the x–z plane fix x_0, therefore, S^2 is identified with the quotient space $\mathrm{SO}(3)/\mathrm{SO}(2)$. The irreducible representations of $\mathrm{SO}(3)$ are given by the so-called Wigner matrices. The ℓ'th irreducible representation is $2\ell+1$ dimensional and of the form

$$[\rho_\ell(\theta,\phi,\psi)]_{m,m'} = e^{-\iota m'\phi}\, d^\ell_{m',m}(\theta)\, e^{-\iota m\psi},$$

where $m, m' \in \{-\ell, \dots, \ell\}$, (θ,ϕ,ψ) are the Euler angles of the rotation and the $d^\ell_{m',m}(\theta)$ funcion is related to the spherical harmonics. It is immediately clear that on restriction to $\mathrm{SO}(2)$ (corresponding to $\theta, \phi = 0$) only the middle column in each of these matrices reduces to the trivial representation of $\mathrm{SO}(2)$, therefore, by Proposition 1, in the case $f \colon \mathrm{SO}(3)/\mathrm{SO}(2) \to \mathbb{C}$, only the middle column of each $\widehat{f}(\rho_\ell)$ matrix will be nonzero, and that middle column will be given by the customary spherical harmonic expansion coefficients.

Cohen et al. (2018) explicitly make this connection between spherical harmonics and $SO(3)$ Fourier transforms, and store the activations in terms of this representation. Moreover, just like in the present paper, they define convolution in terms of the noncommutative convolution theorem (Proposition 2), use pointwise nonlinearities, and prove that the resulting neural network is $\mathrm{SO}(3)$–equivariant. However, they do not prove the converse, i.e., that equivariance implies that the network *must* be convolutional. To apply the nonlinearity, the algorithm presented in (Cohen et al., 2018) requires repeated forward and backward $\mathrm{SO}(3)$ fast Fourier transforms. While this leads to a non-conventional archi-

tecture, the discussion echoes our observation that when dealing with continuous symmetries such as rotations, one must generalize to more abstract "continuous" neural networks, as afforded by Definition 3.

6.3. Message passing neural networks

There has been considerable interest in extending the convolutional network formalism to learning from graphs (Niepert et al., 2016; Defferrard et al., 2016; Duvenaud et al., 2015), and the current consensus for approaching this problem is to use neural networks based on the message passing idea (Gilmer et al., 2017). Let $\mathcal{G}$ be a graph with n vertices. Message passing neural networks (MPNNs) are usually presented in terms of an iterative process, where in each round ℓ, each vertex v collects the labels of its neighbors $w_1, \dots, w_k$, and updates its own label $\tilde{f}_v$ according to a simple formula such as

$$\tilde{f}_v^\ell = \Phi\big(\tilde{f}_{w_1}^{\ell-1} + \dots + \tilde{f}_{w_k}^{\ell-1}\big).$$

An equivalent way of seeing this process, however, is in terms of the "receptive fields" $\mathcal{S}_v^\ell$ of each vertex at round ℓ, i.e., the set of all vertices that v has received information from by round ℓ.

Remarkably, this allows us to view MPNNs as group convolutional networks. In particular, a receptive field of size k is just a k element subset $\{s_1, \dots, s_k\} \subset \{1, 2, \dots, n\}$, and the symmetric group $\mathbb{S}_n$ (the group of permutations of $\{1, 2, \dots, n\}$) acts on the set of such subsets transitively by

$$\{s_1, \dots, s_k\} \overset{\sigma}{\mapsto} \{\sigma(s_1), \dots, \sigma(s_k)\} \qquad \sigma \in \mathbb{S}_n.$$

Since permuting the $n-k$ vertices *not* in $\mathcal{S}$ amongst themselves, as well as permuting the k vertices that are in $\mathcal{S}$ both leave $\mathcal{S}$ invariant, the stablizier of this action is $\mathbb{S}_{n-k} \times \mathbb{S}_k$. Thus, the set of all k-subsets of vertices is identified with the quotient space $\mathcal{X} = \mathbb{S}_n/(\mathbb{S}_k \times \mathbb{S}_{n-k})$, and the labeling function for k-element receptive fields is identified with a function $f^k \colon \mathcal{X} \to \mathbb{C}$. Effectively, this turns the MPNN into a generalized feed-forward network in the sense of Definition 3. Note that f^k is a redundant representation of the labeling function because $\mathbb{S}_n/(\mathbb{S}_k \times \mathbb{S}_{n-k})$ includes subsets that do not correspond to contiguous neighborhoods. However this is not a problem because for such $\mathcal{S}$ we simply set $f^k(\mathcal{S}) = 0$.

One of the advantages of the message passing formalism is that by construction it ensures that $\tilde{f}_v^\ell$ labels only depend on the graph topology and are invariant to simply renumbering the vertices of $\mathcal{G}$. In terms of our "k–subset network" this means that each f^k must be $\mathbb{S}_n$–equivariant.

The "k–subset network" is interesting because, in contrast to the previous two examples, it is a case where each index set $\mathcal{X}_\ell = \mathbb{S}_n/(\mathbb{S}_{n-\ell} \times \mathbb{S}_\ell)$ is different. The form of the corresponding convolutions $L_{V_{\ell-1}}(\mathcal{X}_{\ell-1}) \to L_{V_\ell}(\mathcal{X}_\ell)$ are best described in the Fourier domain. Unfortunately, this requires some background in the representation theory of symmetric groups, which is beyond the scope of the present paper (Sagan, 2001). We content ourselves by stating that the irreps of $\mathbb{S}_n$ are indexed by so-called integer partitions, $(\lambda_1, \dots, \lambda_k)$, where $\lambda_1 \geq \dots \geq \lambda_k$ and $\sum_i \lambda_i = n$. Moreover the structure of the Fourier transform of a function $f \colon \mathbb{S}_n/(\mathbb{S}_{n-\ell} \times \mathbb{S}_\ell)$ dictated by Proposition 1 in this case is that each of the Fourier matrices are zero except for a single column in each of the $\widehat{f}((n-p, p))$ components, where $0 \leq p \leq \ell$. The main theorem of our paper dictates that the linear map ϕ_ℓ in each layer must be a convolution. In the case of Fourier matrices with such extreme sparsity structure, this effectively means that each of the ℓ Fourier matrices can be multiplied by a scalar, χ_p^ℓ. These are the learnable parameters of the network. A real MPNN of course has multiple channels and various corresponding parameters, which could also be introduced in the k–subset network. The above observation about the form of χ is nonetheless interesting, because it at once implies that permutation equivariance is a severe constraint the significantly limits the form of the convolutional filters, yet the framework is still richer than traditional MPNNs where the labels of the neighbors are simply summed.

7. Conclusions

Convolution has emerged as one of the key organizing principles of deep neural network architectures. Nonetheless, depending on their background, the word "convolution" means different things to different researchers. The goal of this paper was to show that in the common setting when there is a group acting on the data that the architecture must be equivariant to, convolution has a specific mathematical meaning that has far reaching consequences: we proved that a feed forward network is equivariant to the group action if and only if it respects this notion of convolution.

This theory gives a clear prescription to practitioners on how to design neural networks for data with non-trivial symmetries, such as data on the sphere, etc.. In particular, we argue for the benefit of Fourier space representations, similar to those that have appeared in (Worrall et al., 2017; Cohen et al., 2018)),

Acknowledgements

This work was supported in part by DARPA Young Faculty Award D16AP00112.

References

Cohen, T. S. and Welling, M. Group equivariant convolutional networks. *Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

Cohen, T. S. and Welling, M. Steerable CNNs. In *Interna-*

tional Conference on Learning Representations (ICLR), 2017.

Cohen, T. S., Geiger, M., Köhler, J., and Welling, M. Spherical CNNs. *International Conference on Learning Representations*, 2018.

Defferrard, M., Bresson, X., and Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gomez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., and Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. In *Advances in Neural Information Processing Systems (NIPS)*, 2015.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. Neural message passing for quantum chemistry. *Proceedings of the International Conference on Machine Learning (ICML)*, 2017.

LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1:541–551, 1989.

Li, Y., Tarlow, D., Brockschmidt, M., and Zemel, R. Gated graph sequence neural networks. In *International Conference on Learning Representations (ICLR)*, 2016.

Mallat, S. Group invariant scattering. *Communications on Pure and Applied Mathematics*, 65(10), 2012.

Marcos, D., Volpi, M., Komodakis, N., and Tuia, D. Rotation equivariant vector field networks. In *Proceedings of International Conference on Computer Vision*, 2017.

Masci, J., Boscaini, D., Bronstein, M. M., and Vandergheynst, P. Geodesic convolutional neural networks on Riemannian manifolds. *2015 IEEE International Conference on Computer Vision Workshop (ICCVW)*, 2015.

Monti, F., Boscaini, D., Masci, J., Rodolà, E., Svoboda, J., and Bronstein, M. M. Geometric deep learning on graphs and manifolds using mixture model cnns. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Niepert, M., Ahmed, M., and Kutzkov, K. Learning convolutional neural networks for graphs. In *Proceedings of the International Conference on Machine Learning (ICML)*, 2016.

Ravanbakhsh, S., Schneider, J., and Poczos, B. Equivariance through parameter-sharing. In *Proceedings of International Conference on Machine Learning*, 2017.

Sagan, B. E. *The Symmetric Group*. Graduate Texts in Mathematics. Springer, 2001.

Terras, A. *Fourier analysis on finite groups and applications*, volume 43 of *London Mathematical Society Student Texts*. Cambridge Univ. Press, 1999.

Worrall, D. E., Garbin, S. J., Turmukhambetov, D., and Brostow, G. J. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of International Conference on Computer Vision and Pattern Recognition*, 2017.