
Implicit Bias of Linear Equivariant Networks

Hannah Lawrence¹ Kristian Georgiev¹ Andrew Dienes¹ Bobak T. Kiani¹

Abstract

Group equivariant convolutional neural networks (G-CNNs) are generalizations of convolutional neural networks (CNNs) which excel in a wide range of technical applications by explicitly encoding symmetries, such as rotations and permutations, in their architectures. Although the success of G-CNNs is driven by their *explicit* symmetry bias, a recent line of work has proposed that the *implicit* bias of training algorithms on particular architectures is key to understanding generalization for overparameterized neural nets. In this context, we show that L -layer full-width linear G-CNNs trained via gradient descent for binary classification converge to solutions with low-rank Fourier matrix coefficients, regularized by the $2/L$ -Schatten matrix norm. Our work strictly generalizes previous analysis on the implicit bias of linear CNNs to linear G-CNNs over all finite groups, including the challenging setting of non-commutative groups (such as permutations), as well as band-limited G-CNNs over infinite groups. We validate our theorems via experiments on a variety of groups, and empirically explore more realistic nonlinear networks, which locally capture similar regularization patterns. Finally, we provide intuitive interpretations of our Fourier space implicit regularization results in real space via uncertainty principles.

1. Introduction

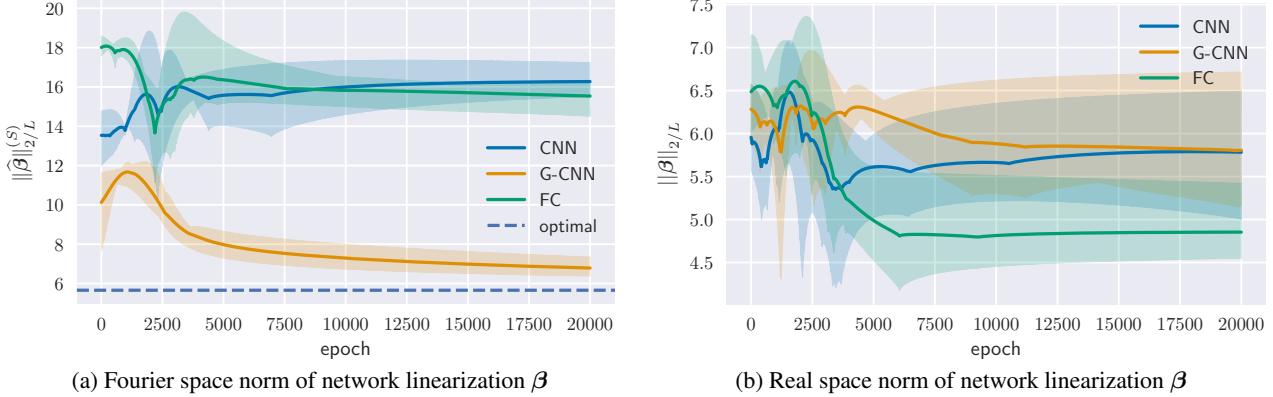
Modern deep learning algorithms typically have many more parameters than data points, and their ability to achieve good generalization in this overparameterized setting is largely unexplained by current theory. Classic generalization bounds, which bound the generalization error when models are not overly “complex,” are vacuous for neural networks that can

¹Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, USA. Correspondence to: Bobak T. Kiani <bkiani@mit.edu>.

perfectly fit random training labels (Zhang et al., 2017). More recent work analyzes the complexity of deep learning algorithms by instead characterizing the properties of the functions they output. Notably, prior work has shown that training via gradient descent implicitly regularizes towards certain hypothesis classes with low complexity, which may generalize better as a result. For example, in underdetermined least squares regression, gradient descent converges to the ℓ_2 -norm minimizer, while a pointwise-square reparametrization converges to the ℓ_1 -norm minimizer (Gunasekar et al., 2018a). For separable linear regression, Soudry et al. (2018) proved that the learned predictor under gradient descent converges in direction to the max-margin solution. Such phenomena are consistent with certain linear neural networks, e.g., Lyu and Li (2020) extended this max-margin result to gradient descent on any homogeneous neural network, and Gunasekar et al. (2018b) showed that learned linear fully-connected and convolutional networks implicitly regularize the ℓ_2 norm and a depth-dependent norm in Fourier space, respectively.

From a more applied perspective, a large body of work imposes structured inductive biases on deep learning algorithms to exploit symmetry patterns (Kondor, 2007; Reisert, 2008; Cohen and Welling, 2016b). One prominent method parameterizes models over functions that are *equivariant* with respect to a symmetry group (*i.e.*, outputs transform predictably in response to input transformation). In fact, Kondor and Trivedi (2018) showed that any group equivariant network can be expressed as a series of group convolutional layers interwoven with pointwise nonlinearities, demonstrating that group convolutional neural networks (G-CNNs) are the most general family of equivariant networks.

Our Contributions: The explicit inductive bias of G-CNNs is the main reason for their usage. Yet, to the best of our knowledge, the implicit bias imposed by equivariant architectures has not been explored. Here, we greatly generalize the results of Yun et al. (2020) and Gunasekar et al. (2018b) to linear G-CNNs whose hidden layers perform group equivariant transformations. We show, surprisingly, that L -layer G-CNNs are implicitly regularized by the $2/L$ -Schatten norm, which is the $2/L$ norm of a matrix’s singular values, over the irreducible representations in the Fourier basis. As a result, convergence is biased towards sparse solutions in the Fourier regime, as summarized in Theorem 1 and il-


 (a) Fourier space norm of network linearization β

 (b) Real space norm of network linearization β

Figure 1: Training via gradient descent of linear G-CNNs (with linearization β) implicitly biases towards sparse singular values for Fourier matrix coefficients of β (see [Theorem 1](#)). [Figure 1a](#) traces the D_8 Fourier space sparsity over the course of training three architectures in an overparameterized classification task, beginning from epoch 1. The G-CNN converges to a Fourier-sparse linearization, in contrast with fully-connected and convolutional networks. The horizontal line shows the minimum Fourier Schatten norm among all interpolating predictors, obtained as the solution of a convex relaxation; since the G-CNN norm converges toward this value, this verifies our main statement of convergence to a local minimum. [Figure 1b](#) illustrates the uncertainty principles, discussed in [Section 6](#) and illustrated more explicitly in [Figure 5](#) that sparseness in (group) Fourier space necessitates being “dense” in real space, and vice versa.

lustrated in [Figure 1a](#), as well as in further experiments in [Section 6](#).

New Technical Ingredients: Our primary technical contribution is to generalize the proof technique of [Gunasekar et al. \(2018b\)](#) by realizing it in a more general setting, translating it to non-abelian groups using the language of representation theory. Since convolutions in real space correspond to (non-commutative) matrix multiplications, rather than scalar multiplications, in the appropriate group Fourier space, this substantially complicates analysis of KKT conditions. Nonetheless, we use the framework of non-commutative Fourier and convex analysis to prove that a particular function on the singular values of the Fourier-space linearization is being implicitly optimized.

Theorem 1 (main result; informal). *Let $\text{NN}_L(\cdot)$ denote an L layer linear group convolutional neural network, in which each hidden layer performs group cross-correlation over the group G with a full-width kernel, and the final layer is a fully connected layer. When learning linearly separable data $\{\mathbf{x}_i, y_i\}_{i=1}^n$ using the exponential loss function, the network converges in direction to a linear function $\text{NN}_L(\mathbf{x}) \propto \beta^T \mathbf{x}$, where β is given by a stationary point of the following minimization problem:*

$$\min_{\beta} \|\hat{\beta}\|_{2/L}^{(S)} \quad \text{s.t. } y_i \mathbf{x}_i^T \beta \geq 1 \quad \forall i \in [n] \quad (1)$$

Here, $\|\hat{\beta}\|_{2/L}^{(S)}$ denotes the $2/L$ -Schatten norm of the matrix

Fourier transform of β (see [Definition 4.1](#)) equivalent to

$$\|\hat{\beta}\|_{2/L}^{(S)} = \left[\sum_{\rho \in \widehat{G}} d_\rho \left(\|\hat{\beta}(\rho)\|_{2/L}^{(S)} \right)^{2/L} \right]^{L/2}, \quad (2)$$

where \widehat{G} is a complete set of unitary irreducible representations of G and d_ρ is the dimension of irreducible representation ρ .

We note that both sparsity (for vectors) and low-rankness (for matrices) are desirable properties for many applications, not least because such predictors are efficient to store and manipulate, thus potentially expanding the scope of application of G-CNNs to areas where sparsity and low-rankness are explicit desiderata. Connecting our findings to research on uncertainty theorems ([Wigderson and Wigderson, 2021](#)), we also show that the implicit regularization towards sparseness (or low rank irreducible representations) in the Fourier regime necessarily implies that solutions in the real regime are “dense,” as illustrated in [Figure 1b](#). These results provide a more intuitive and practical perspective into the inductive bias of G-CNNs and the types of functions that they learn.

We proceed as follows. [Section 2](#) discusses related works and their relation to our contributions. In [Section 3](#), we define notation. [Section 4](#) provides a basic background in the group theory and Fourier analysis necessary to understand our results, with a more complete exposition in [Appendix B](#). Our main results are stated in [Section 5](#), with the main proof ideas for the abelian (or commutative) and non-abelian (or non-commutative) cases given in [Section 5.1](#)

and Section 5.2, respectively (complete proofs can be found in Appendix A). Section 6 validates our theoretical results with synthetic experiments on a variety of groups, and exploratory experiments validating our theory on non-linear networks. Finally, we discuss these results and future questions in Section 7.

2. Related Work

Enforcing equivariance and symmetries via parameter sharing schemes was introduced in the group theoretic setting in Cohen and Welling (2016a) and Gens and Domingos (2014). Despite considerable interest in equivariant learning, no works to our knowledge have explored the implicit regularization of gradient descent on equivariant convolutional neural networks. We show that the tensor formulation of neural networks in Yun et al. (2020) and the proofs in Gunasekar et al. (2018b) encompass G-CNNs for which the underlying group is cyclic, and we naturally extend their results to G-CNNs over any commutative group (see Section 5.1). However, these works do not cover the case of convolutions with respect to non-commutative groups, such as three-dimensional rotations and permutations, which incidentally include some of the most compelling applications of group equivariance in practice (Zaheer et al., 2017; Anderson et al., 2019; Esteves et al., 2018). As such, articulating the implicit bias in the more general non-abelian case is important for understanding many of the current group equivariant architectures (Zaheer et al., 2017; Kondor et al., 2018; Esteves et al., 2018; Weiler and Cesa, 2019). Non-abelian convolutions require more structure to theoretically analyze compared to abelian convolutions: the former are merely pointwise multiplications in Fourier space, whereas the latter are *matrix* multiplications between irreducible representations, and therefore cannot be expressed in the tensor language of Yun et al. (2020). Instead, we build on the optimization tools and comparable convergence assumptions of Gunasekar et al. (2018b) to explicitly characterize the stationary points of convergence for non-abelian G-CNNs.

We also note that our results are consistent with those of Razin and Cohen (2020) showing that implicit generalization is often captured by measures of complexity which are quasi-norms, such as tensor rank. Our results prove that linear G-CNNs are biased towards low rank solutions in the Fourier regime, via regularization of the $2/L$ -Schatten norms over Fourier matrix coefficients (also a quasi-norm). Lastly, there is a line of work focusing on understanding the expressivity (Kondor and Trivedi, 2018; Cohen et al., 2019; Yarotsky, 2021) and generalization (Sannai and Imaizumi, 2019; Lyle et al., 2020; Bulusu et al., 2021; Elesedy and Zaidi, 2021) of equivariant networks, but not specifically their implicit regularization.

To analyze bounded-width filters, which are more com-

monly used in practice, a recent work by Jagadeesan et al. (2021) shows that the implicit regularization for an arbitrary filter width K is unlikely to admit a closed-form solution. Separate from calculating the exact form of implicit regularization, there is a rich line of work that details the trade-offs between restricting a function in its real versus Fourier regimes via uncertainty principles (Meshulam, 1992; Wigderson and Wigderson, 2021). While the connection between uncertainty theorems and bounded-width convolutional neural networks has not been thoroughly explored, Caro et al. (2021) and Nicola and Trapasso (2021) highlight the importance of uncertainty principles for understanding the behaviour of modern CNNs.

3. Notation

Throughout this text, we denote scalars in lowercase script (a), vectors in bold lowercase script (\mathbf{a}), matrices in either bold uppercase script (\mathbf{A}) or lowercase script hat ($\hat{\mathbf{a}}$) when vectors are transformed into the Fourier regime (see Definition 4.1), and tensors in bold non-italic uppercase script (\mathbf{A}). For f a function with range in \mathbb{C} , we overload notation slightly and let \bar{f} denote the function with an element-wise complex conjugate applied, i.e. $\bar{f}(x) = \overline{f(x)}$. If f is defined on a group G , let $f^-(g) = f(g^{-1})$. For a vector $\mathbf{a} \in \mathbb{C}^n$ or a matrix $\mathbf{A} \in \mathbb{C}^{m \times n}$, we denote its conjugate transpose as \mathbf{a}^\dagger and \mathbf{A}^\dagger respectively. We use $\langle \cdot, \cdot \rangle$ to denote the standard vector inner product between two vectors and $\langle \cdot, \cdot \rangle_M$ to denote the inner product between matrices defined as $\langle \mathbf{A}, \mathbf{B} \rangle_M = \text{tr}(\mathbf{AB}^\dagger)$. $\|\cdot\|_p$ denotes the vector p -norm ($p = 2$ when subscript is hidden) and $\|\cdot\|_p^{(S)}$ denotes the p -Schatten norm¹ of a matrix (equivalent to the p -vector norm of the singular values of the matrix).

We denote groups by uppercase letters G , an irreducible representation (irrep) of a group G by ρ or ρ_i , and a complete set of irreps by \widehat{G} , so every unitary irrep ρ is equivalent (up to isomorphism) to exactly one element of \widehat{G} . The dimension of a given irrep ρ is d_ρ .

4. Background in group theory and Group-Equivariant CNNs

In this study, we analyze linear G-CNNs in a binary classification setting, where hidden layers perform equivariant operations over a finite group G , and networks have no nonlinear activation function after their linear group operations. Inputs \mathbf{x}_i are vectors of dimension $|G|$ (i.e., vectorized group functions $\mathbf{x} : G \rightarrow \mathbb{R}$), and targets y_i are scalars taking the value of either +1 or -1. Hidden layers in our G-CNNs

¹Despite our terminology, p -vector and p -Schatten norms are technically quasi-norms for $p < 1$.

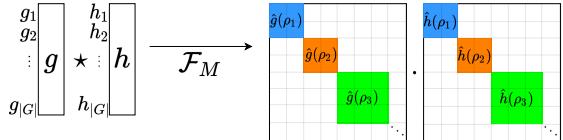


Figure 2: Cross-correlation of two functions over a group is equivalent to matrix multiplication over irreps (shown as blocks of a larger matrix here) in Fourier space.

perform cross-correlation over a group G , defined as

$$(g \star h)(u) = \sum_{v \in G} g(uv)h(v) \quad (3)$$

where $g, h : G \rightarrow \mathbb{R}$. Note that the above is equivariant to the left action of the group, *i.e.*, if $w \in G$ and $g'_w(u) = g(wu)$, then $(g'_w \star h)(u) = (g \star h)(wu)$. The final layer of our G-CNN is a fully connected layer mapping vectors of length $|G|$ to scalars. We note that this final layer in general will not construct functions that are symmetric to the group operations, as strictly enforcing group invariance in this linear setting will result in trivial outputs (only scalings of the average value of the input). Nonetheless, this model still captures the composed convolutions of G-CNNs, and is similar in construction to many practical G-CNN models, whose earlier G-convolutions still capture useful high-level equivariant features. For instance, the spherical CNN of Cohen et al. (2018) also has a final fully connected layer.

Analogous to the discrete Fourier transform, there exists a *group* Fourier transform mapping a function into the Fourier basis over the irreps of G .

Definition 4.1 (Group Fourier transform). Let $f : G \rightarrow \mathbb{C}$. Given a fixed ordering of G , let e_u be the standard basis vector in $\mathbb{R}^{|G|}$ that is 1 at the location of u and 0 elsewhere. Then, $\hat{f} = \sum_{u \in G} f(u)e_u$ is the vectorized function f . Given \widehat{G} a complete set of unitary irreps of G , let $\rho \in \widehat{G}$ be a given irrep of dimension d_ρ , $\rho : G \longrightarrow \text{GL}(d_\rho, \mathbb{C})$ ². The group Fourier transform of f , $\widehat{f} : \widehat{G} \rightarrow \mathbb{C}$ at a representation ρ is defined as (Terras, 1999)

$$\widehat{f}(\rho) = \sum_{u \in G} f(u)\rho(u). \quad (4)$$

By choosing a fixed ordering of \widehat{G} , one can similarly construct \widehat{f} as a block-diagonal matrix version of \widehat{f} (as in Figure 2). We define \mathcal{F}_M to be the *matrix* Fourier transform that takes f to \widehat{f} :

$$\widehat{f} = \mathcal{F}_M f = \bigoplus_{\rho \in \widehat{G}} \widehat{f}(\rho)^{\oplus d_\rho} \in \text{GL}(|G|, \mathbb{C}). \quad (5)$$

²Note that for an abelian group, $d_\rho = 1 \forall \rho$. For standard Fourier analysis over the cyclic group, each ρ is a complex sinusoid at some frequency.

\widehat{f} or $\mathcal{F}_M f$ are shortened notation for the complete Fourier transform. Furthermore, by vectorizing the matrix \widehat{f} , there is a *unitary* matrix \mathcal{F} taking f to \widehat{f} , analogous to the standard discrete Fourier matrix. We use the following explicit construction of \mathcal{F} : denoting $e_{[\rho, i, j]}$ as the column-major vectorized basis for element ρ_{ij} in the group Fourier transform, then we can form the matrix

$$\mathcal{F} = \sum_{u \in G} \sum_{\rho \in \widehat{G}} \frac{\sqrt{d_\rho}}{\sqrt{|G|}} \sum_{i, j=1}^{d_\rho} \rho(u)_{ij} e_{[\rho, i, j]} e_u^T. \quad (6)$$

Intuitively, for each group element g , the matrix \mathcal{F} contains all the irrep images $\rho(g)$ ‘flattened’ into a single column. See Appendix B for further exposition.

Convolution and cross-correlation are equivalent, up to scaling, to matrix operations after Fourier transformation. For example, for cross-correlation (Equation 3), $(\widehat{g} \star \widehat{h})(\rho) = \widehat{g}(\rho)\widehat{h}(\rho)^\dagger$. This simple fact, illustrated in Figure 2, is behind the proofs of our implicit bias results.

5. Main Results

We consider linear group-convolutional networks for classification analogous to those of Yun et al. (2020) and Gunnasekar et al. (2018b). A linear G-CNN is composed of several group cross-correlation layers followed by a fully connected layer. The input is formalized as a function on the group (according to some pre-defined ordering of elements), $x : G \rightarrow \mathbb{R}$, and the output is a scalar. Explicitly, let G be a finite group with x, w_1, \dots, w_L real-valued functions on G . The network output is $\text{NN}(x) = \langle x \star w_1 \star \dots \star w_{L-1}, w_L \rangle \triangleq \langle x, \beta \rangle$. As an example, in Figure 3, we illustrate both a “practical” G-CNN architecture and this linearized version that we will study, for the group D_8 . Let $\mathbf{W} = [w_1 \dots w_L]$ be the concatenation of all network parameters, and $\beta = \mathcal{P}(\mathbf{W}) = w_L \star w_{L-1} \star \dots \star w_1$ be the “end-to-end” linear predictor consisting of composed cross-correlation operations. One can check (as we do in Lemma A.1) that $\widehat{\beta} \triangleq \mathcal{F}_M \beta = \widehat{w_L} \dots \widehat{w_1}$. Networks are trained via gradient descent over the exponential loss function on linearly separable data $\{\mathbf{x}_i, y_i\}_{i=1}^N$. Iterates take the form

$$\mathbf{W}_\ell^{(t+1)} = \mathbf{W}_\ell^{(t)} - \eta_t \nabla_{\mathbf{W}_\ell} \mathcal{L}(\mathcal{P}(\mathbf{W})) \quad (7)$$

where $\ell(\langle \mathbf{x}, \beta \rangle, y) = \exp(-\langle \mathbf{x}, \beta \rangle \cdot y)$ and $\mathcal{L}(\beta) = \sum_{i=1}^N \ell(\langle \mathbf{x}_i, \beta \rangle, y_i)$.

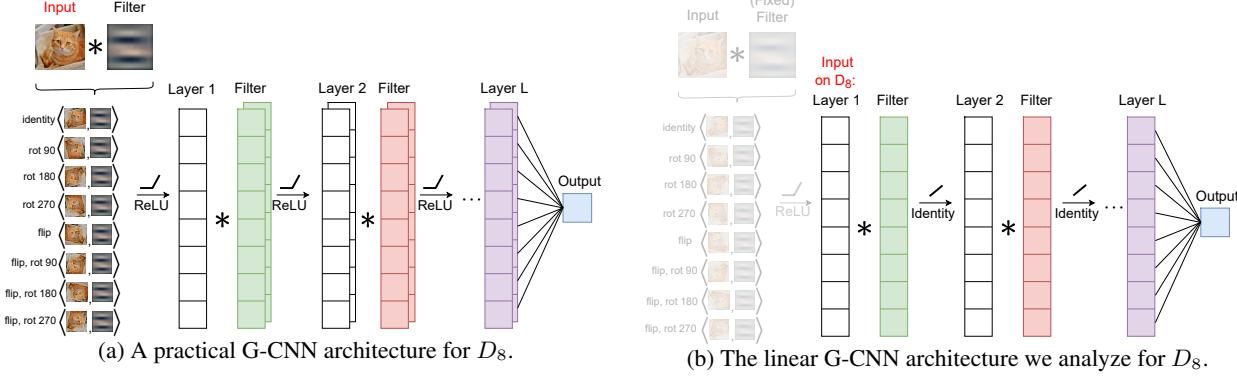


Figure 3: A comparison between a practical G-CNN architecture for the group D_8 , and its corresponding linear idealization that we will theoretically analyze. D_8 is a group of size 8, consisting of all rotations by 90 degrees and reflections. In a practical architecture, as shown in the first panel, the input may be an image, or anything upon which D_8 acts; it can be convolved over D_8 with respect to a learnable filter, yielding a function on D_8 (i.e. a function that takes 8 values). The following layers intersperse D_8 -convolutions, possibly with several channels, and nonlinearities, before a final fully connected layer yields a scalar output. In contrast, the second panel shows the simplified architecture we analyze here. In particular, there are no non-linearities and only a single channel is used. Furthermore, we assume the input is already a function on D_8 , i.e. the input is an 8-dimensional vector. (One can think of this input as a fixed featurization of some image convolved over D_8 with a *fixed* filter.)

5.1. Abelian

Similar to ordinary (cyclic) convolution³, the commutative property of abelian groups implies that convolutions in real space are equivalently pointwise multiplication of irreps in Fourier space, since all irreps are one-dimensional for commutative groups ($d_\rho = 1 \forall \rho$). To start, recall the key definition of Yun et al. (2020), determining which network architectures fall within the purview of their results:

Proposition 5.1 (paraphrased from (Yun et al., 2020)). *Let $\mathbf{M}(\mathbf{x})$ be a map from data $\mathbf{x} \in \mathbb{R}^d$ to a data tensor $\mathbf{M}(\mathbf{x}) \in \mathbb{R}^{k_1 \times k_2 \times \dots \times k_L}$. The input into an L -layer tensorized neural network can be written as an orthogonally decomposable data tensor if there exists a full column rank matrix $\mathbf{S} \in \mathbb{C}^{m \times d}$ and semi-unitary matrices $\mathbf{U}_1, \dots, \mathbf{U}_L \in \mathbb{C}^{k_\ell \times m}$ where $d \leq m \leq \min_\ell k_\ell$ such that:*

$$\mathbf{M}(\mathbf{x}) = \sum_{j=1}^m [\mathbf{S}\mathbf{x}]_j ([\mathbf{U}_1]_{\cdot, j} \otimes [\mathbf{U}_2]_{\cdot, j} \otimes \dots \otimes [\mathbf{U}_L]_{\cdot, j})$$

and moreover the network output is the tensor multiplication between $\mathbf{M}(\mathbf{x})$ and each layer's parameters:

$$\begin{aligned} \text{NN}(\mathbf{x}; \Theta) &= \mathbf{M}(\mathbf{x}) \cdot (\mathbf{w}_1, \dots, \mathbf{w}_L) \\ &= \sum_{i_1=1}^d \dots \sum_{i_L=1}^d \mathbf{M}(\mathbf{x})_{i_1 \dots i_L} (\mathbf{w}_1)_{i_1} \dots (\mathbf{w}_L)_{i_L} \end{aligned}$$

³We include a canonical result in the appendix, Theorem A.4, demonstrating that all finite abelian groups are direct products of cyclic groups, i.e. multidimensional translational symmetries.

Indeed, a linear G-CNN over an abelian group can be expressed in a way that satisfies Proposition A.3 for an appropriate choice of \mathbf{S} and $\mathbf{U}_1, \dots, \mathbf{U}_L$, as stated in the following proposition.

Proposition 5.2. *Let $\mathbf{M}(\mathbf{x})$ be an orthogonally decomposable data tensor with associated matrices $\mathbf{S}, \mathbf{U}_1, \dots, \mathbf{U}_L$ as in Proposition 5.1. Given a finite abelian group G , let $d = m = k_\ell = |G|$ and $\mathcal{F} \in \mathbb{C}^{d \times d}$ be the group Fourier transform of G (see Definition 4.1). With $\mathbf{S} = d^{\frac{L-1}{2}} \mathcal{F}$, unitary matrices $\mathbf{U}_\ell = \mathcal{F}^{-1} = \mathcal{F}^\dagger$, and the data tensor $\mathbf{M}(\mathbf{x})$ defined correspondingly, the output of a G-CNN with real-valued filters $\mathbf{w}_1, \dots, \mathbf{w}_L$ is a tensor operation:*

$$\mathbf{M}(\mathbf{x}) \cdot (\mathbf{w}_1, \dots, \mathbf{w}_L) = \langle \mathbf{x} * \mathbf{w}_1 * \dots * \mathbf{w}_{L-1}, \mathbf{w}_L \rangle$$

The proof is deferred to Appendix A. Fundamentally, the result requires not only that \mathcal{F} is unitary, which holds for all finite groups, but also that cross-correlation is pointwise multiplication (up to a conjugate transpose) in Fourier space, i.e. $\mathcal{F}(\mathbf{x} * \mathbf{w}) = (\mathcal{F}\mathbf{x}) \odot (\mathcal{F}\mathbf{w})^\dagger$. This property only holds for commutative groups, as matrix multiplication is pointwise multiplication only for matrices of dimension $d_\rho = 1$. Given Proposition 5.2, we apply the implicit bias statement of Yun et al. (2020).

Theorem 5.3 (Implicit regularization of linear G-CNNs for G an abelian group). *Suppose there exists $\lambda > 0$ such that the initial directions $\bar{\mathbf{w}}_1, \dots, \bar{\mathbf{w}}_L$ of the network parameters satisfy $|[\mathcal{F}\bar{\mathbf{w}}_\ell]_j|^2 - |[\mathcal{F}\bar{\mathbf{w}}_L]_j|^2 \geq \lambda$ for all $\ell \in [L-1]$ and $j \in [m]$, i.e. if the Fourier transform magnitudes of the initial directions look sufficiently different pointwise (which*

occurs with high probability for e.g. a Gaussian random initialization). Then, $\beta = \mathcal{P}([\mathbf{w}_1, \dots, \mathbf{w}_L])$ converges in a direction that aligns with a stationary point \mathbf{z}_∞ of the following optimization program:

$$\min_{\mathbf{z} \in \mathbb{C}^m} \|\mathcal{F}\mathbf{z}\|_{2/L} \quad \text{s.t.} \quad y_i \langle \mathbf{x}_i, \mathbf{z} \rangle \geq 1, \forall i \in [n] \quad (8)$$

As noted in [Theorem A.4](#) of the Appendix, all finite abelian groups can be expressed as a direct product of cyclic groups. In contrast, many groups (rotations, subgroups of permutations, etc.) with much richer structure are non-commutative, and we now turn our attention to the non-abelian case.

5.2. Non-Abelian

In Fourier space, non-abelian convolution consists of matrix multiplication over irreps, and does *not* fit the pointwise multiplication structure of [Proposition 5.2](#). We instead build upon the results of [Gunasekar et al. \(2018b\)](#), and directly analyze the stationary points of the proposed optimization program to prove the following:

Theorem 5.4 (Non-abelian; see also [Theorem A.5](#)). *Consider a classification task with ground-truth linear predictor β , trained via a linear G-CNN architecture with $L > 2$ layers under the exponential loss. For almost all β -separable datasets $\{\mathbf{x}_i, y_i\}_{i=1}^n$, any bounded sequence of step sizes η_t , and almost all initializations: if the loss converges to 0, the gradients converge in direction, and the iterates themselves all converge in direction to a classifier with positive margin, then the resultant predictor is a scaling of a first order stationary point of the optimization problem:*

$$\min_{\beta} \|\widehat{\beta}\|_{2/L}^{(S)} \quad \text{s.t.} \quad \forall n, y_n \langle \mathbf{x}_n, \beta \rangle \geq 1 \quad (9)$$

To prove the above statement, we show that linear G-CNNs converge to stationary points of [Equation 9](#) via KKT conditions, which is also the high-level method of [Gunasekar et al. \(2018b\)](#). However, our proof diverges in several key ways. First, we carefully redefine operations of the G-CNN as a series of inner products and cross-correlations with respect to the matrix Fourier transform of [Definition 4.1](#). Second, in this Fourier space, we analyze the subdifferential of the Schatten norms, to ultimately show that the KKT conditions of [Equation 9](#) are satisfied. In contrast, [Gunasekar et al. \(2018b\)](#) analyze the subdifferential of a different objective, the ordinary $2/L$ -vector norm. The fact that the irreps of a group are only unique up to isomorphism (e.g., conjugation by a unitary matrix) hints at the Schatten norm as the correct regularizer, since the Schatten norm is among the norms invariant to unitary matrix conjugation, but this must be confirmed by rigorous analysis. These features are specific to the non-abelian case. More specifically, the proof of this result follows the outline below:

1. First, by applying a general result of [Gunasekar et al. \(2018b\)](#), [Theorem A.6](#), we can immediately characterize the implicit regularization in the full space of parameters, \mathbf{W} (in contrast to the end-to-end linear predictor β), as a (scaled) stationary point \mathbf{W}^∞ of the following optimization problem in \mathbf{W} :

$$\min_{\mathbf{W} \in \mathbb{R}^P} \|\mathbf{W}\|_2^2 \quad \text{s.t.} \quad \forall n, y_n \langle \mathbf{x}_n, \mathcal{P}(\mathbf{W}) \rangle \geq 1 \quad (10)$$

2. Separately, we define a *distinct* optimization problem, [Equation 9](#), in β , with the aim of showing that stationary points of [Equation 10](#) are a subset of those of [Equation 9](#), up to scaling.
3. The *necessary* KKT conditions for [Equation 10](#) characterize its stationary points:

$$\begin{aligned} \exists \{\alpha_n \geq 0\} : \alpha_n = 0 \text{ if } y_n \langle \mathbf{x}_n, \mathcal{P}(\mathbf{W}^\infty) \rangle > 1 \\ \mathbf{w}_i^\infty = \nabla_{\mathbf{w}_i} \left\langle \mathcal{P}(\mathbf{W}^\infty), \sum_n \alpha_n y_n \mathbf{x}_n \right\rangle \end{aligned} \quad (11)$$

From here, we show that the *sufficient* KKT conditions for [Equation 9](#) are also satisfied by the corresponding end-to-end predictor. In particular, we calculate the set of subgradients⁴ $\partial^o \|\widehat{\beta}\|_p^{(S)}$ for $p = \frac{2}{L} < 1$, and then use [Equation 11](#) to derive recurrences demonstrating that a positive scaling of $\sum_n \alpha_n y_n \widehat{\mathbf{x}}_n$ is a member of this set.

Remark 5.5. For abelian groups where all irreps are one-dimensional, $\widehat{\beta}$ in [Theorem 5.4](#) is a diagonal matrix. Thus, the p -Schatten norm coincides with the p -vector norm of the diagonal entries, recovering results in [Section 5.2](#). However, [Theorem 5.4](#) requires stronger convergence assumptions.

Infinite dimensional groups: [Theorem 5.4](#) applies to all *finite* groups, but G-CNNs have extensive applications for *infinite* groups, where outputs of convolutions are infinite-dimensional. Here, it is common to assume sparsity in the Fourier coefficients and “band-limit” filters over a set of low-frequency irreps (under some natural group-specific ordering) that form a finite dimensional linear subspace (we denote the representation of a function in this band-limited Fourier space as $\widehat{\mathbf{w}}$). G-CNNs with band-limited filters take precisely the form of the finite G-CNNs from [Theorem 5.4](#). Thus, slight modifications yield the following for infinite groups (see [Appendix A.2.1](#) for details).

Corollary 5.6 (Infinite-dimensional groups with band-limited functions; see also [Theorem A.11](#)). *Let G be a compact Lie group with irreps \widehat{G} , and let $B \subset \widehat{G}$ with $|B| < \infty$.⁵*

⁴ ∂^o is the local subgradient of ([Clarke, 1975](#)): $\partial^o f(\beta) = \text{conv}\{\lim_{i \rightarrow \infty} \nabla f(\beta + \mathbf{h}_i) : \mathbf{h}_i \rightarrow 0\}$

⁵For example, $G = SO(3)$ and B indexes all Wigner d-matrices with $|\ell| \leq L$ ([Kondor et al., 2018](#)).

Proceed fully in Fourier space, in the subspace corresponding to B : consider a linearly separable classification task with ground-truth linear predictor $\widehat{\beta}$ and inputs \underline{x} , both real-valued and supported only on irreps in B , and proceed by gradient descent on the band-limited Fourier-space filters. Under near-identical conditions as Theorem 5.4, the resultant predictor is a scaling of a first order stationary point of:

$$\min_{\widehat{\beta}} \left\| \widehat{\beta} \right\|_{2/L}^{(S)} \quad \text{s.t.} \quad \forall n, y_n \langle \underline{x}_n, \widehat{\beta} \rangle \geq 1 \quad (12)$$

6. Experiments

We first experimentally confirm our theory in a simple setting illustrating the effects of implicit regularization.⁶ Then, we relax the crucial assumption of linearity in our setup, to empirically show that the results may hold locally even in nonlinear settings (including the practical case of spherical CNNs (Cohen et al., 2018)). Note that the results for nonlinear networks in Section 6.2 are *only* empirical in nature, and Theorems 5.3 and A.5 do not necessarily hold in the more general nonlinear setting. For all binary classification tasks, we use three-layer networks with inputs and convolution weights in $\mathbb{R}^{|G|}$, and all plots begin at epoch 1. Since we are interested in the resulting implicit bias alone, we only analyze loss on data in the training set. A complete description of our experimental setup can be found in Appendix E.

Throughout this section, we plot norms in the Fourier and real regimes side-by-side to highlight unavoidable trade-offs in implicit regularization between the two conjugate regimes. These trade-offs, which have a rich history of study in physics and group theory, are commonly termed uncertainty principles (Wigderson and Wigderson, 2021). Since non-abelian groups have matrix-valued irreducible representations, uncertainty theorems must account for norms and notions of support in the context of matrices. One especially relevant uncertainty theorem states that sparseness in the real or Fourier regime necessarily implies dense support in the conjugate regime:

Theorem 6.1 (Meshulam uncertainty theorem (Meshulam, 1992)). *Given a finite group G and $f : G \rightarrow \mathbb{C}$, let \widehat{G} be the set of irreps of G and \widehat{f} be the vectorized function (see Definition 4.1). Then*

$$|\text{supp}(f)| \left[\sum_{\rho \in \widehat{G}} d_\rho \text{rank}(\widehat{f}(\rho)) \right] \geq |G| \quad (13)$$

The theorem above shows that the rank of a function's Fourier matrix coefficients is the proper notion of support in

⁶Our code is available here: <https://github.com/kristian-georgiev/implicit-bias-of-linear-equivariant-networks>

the uncertainty theorem for a non-abelian group. In the context of our result, this implies that the learned linear function is likely to have large support in real space (at least locally, with respect to the network parameters). Other uncertainty principles are detailed in Appendix C.

Remark 6.2 (Generalization). The implication of our results on generalization are highly task and data-dependent. Indeed, one could create *contrived* experiments where the inductive bias leads to *worse* or *better* results, which would depend entirely on whether the ground-truth linear predictor has small Fourier Schatten norm (i.e. based on the uncertainty principle above, whether the ground-truth function is sparse in real or Fourier space). Noting that band-limited functions have small Schatten norm, and that practical spherical CNNs band-limit (as natural spherical images can often be well-represented by their first few spherical harmonics), there is reason to believe this implicit bias could aid in generalization on natural data distributions, but the primary intention of our result is purely to highlight that such an implicit bias exists and characterize it mathematically. As such, our experiments demonstrate only effects on training error, rather than generalization.

6.1. Empirical Confirmation of Theory

We first trace the regularization through (training) epochs for networks trained to classify data with ± 1 labels. We consider three groups here, with more in Appendix E. Figure 1 shows the implicit bias for a G-CNN over the dihedral group D_8 , a simple non-abelian group that captures the geometry of a square⁷. Inputs are vectors with elements drawn i.i.d. from the standard normal distribution. Figure 4 shows the implicit bias for a G-CNN over the non-abelian group $(C_{28} \times C_{28}) \times D_8$ which acts on images (the digits 1 and 5) from the MNIST dataset.

In the three settings above, we compare the behaviors of G-CNN, traditional CNN⁸, and fully-connected (FC) network architectures with similar instantiations. We plot the real space vector norm and Fourier space Schatten norm of the network linearization over training epochs. All models perfectly fit the data in this overparameterized setting, and convergence to a given regularized solution corresponds to convergence in the loss to zero.

Consistent with theory, G-CNN architectures shown in Figures 1 and 4 have the smallest Fourier space Schatten norms among the architectures. Note that since our theory only implies a G-CNN will reach a stationary point of the Schatten norm minimization *subject to fitting the training data*, Schatten norms greater than 0 are expected. The form of the implicit bias is visualized in the example shown in Fig-

⁷ D_n denotes the dihedral group of order n .

⁸“CNN” generically refers to a G-CNN over the cyclic group of size equal to the size of the input.

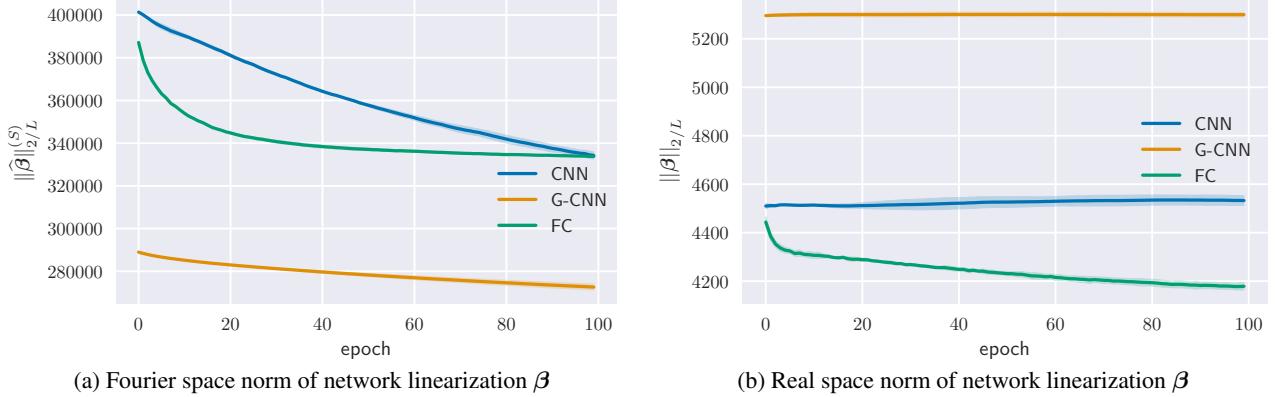


Figure 4: Norms of the linearizations of three different linear architectures for the non-abelian group $G = (C_{28} \times C_{28}) \times D_8$ trained using the digits 1 and 5 from the MNIST dataset.

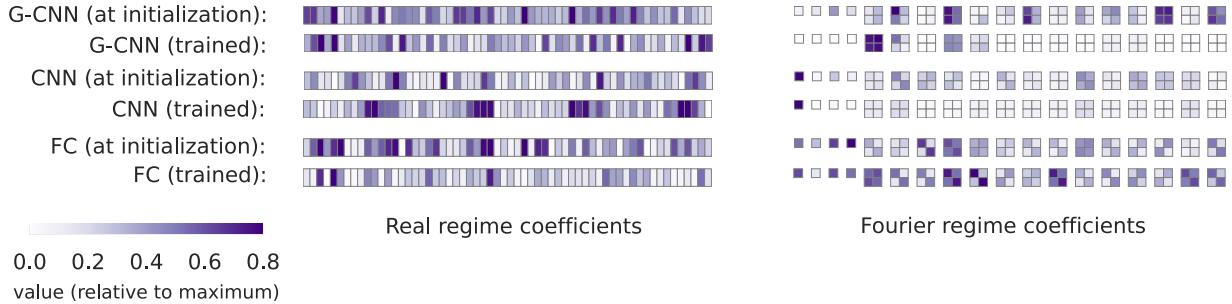


Figure 5: Linearized functions of the G-CNN (over the dihedral group D_{60}) are sparse in the Fourier regime of the group. Furthermore, the sparsity pattern shows up in blocks of 4, which are the blocks containing coefficients of individual irreps of the dihedral group D_{60} . Linearized functions in the real regime of the G-CNN, in contrast, are rather dense, highlighting the uncertainty principles inherent in the implicit bias of G-CNNs. Sparsity in the group Fourier regime is not evident in CNNs or fully connected (FC) networks. This is the same setting as that in Figure 1, except with D_{60} instead of D_8 (see Appendix D for more details).

ure 5, which shows the values of the linearization over irreps in Fourier space (see Appendix D for further details). As expected, the G-CNN outputs a linearization that is sparse over low-rank irreps in the Fourier regime. FC networks exhibit no group Fourier regime regularization, while standard CNNs exhibit some regularization since their irreducible representations are similar to those of the D_{60} -CNN (see e.g. B.8 in the Appendix). The differing behaviors of CNNs and FC networks show that implicit regularization is a consequence of the choice of architecture, and not inherent to the task itself.

6.2. Assessment of Theory on Nonlinear Networks

Here we introduce rectified linear unit (ReLU) nonlinearities between hidden layers to analyze implicit bias in the presence of nonlinearity. Our theoretical results do not necessarily hold in this case, so we are *exploring, rather than confirming*, their validity for nonlinear networks. Given a G-CNN with ReLU activations, we wish to calculate the

Schatten norm of the Fourier matrix coefficients of the network linearization β . However, networks can no longer be collapsed into linear functions due to the nonlinearities. Instead, we construct local linear approximations at each data point via a first-order Taylor expansion, calculate the norms of interest according to this linearization, and average the results across the dataset to get a single aggregate value. We evaluate the implicit bias of a nonlinear G-CNN (with linear final layer) on the dihedral group D_{60} with synthetic data. We also evaluate a nonlinear spherical-CNN, where $G = SO(3)$ is an infinite group, with spherical MNIST data; see Cohen et al. (2018) for details. Linear approximations are used only to analyze implicit regularization, and not to further interpret the outputs of the locally linear neural network, as such analysis can give rise to misleading or fragile feature importance maps (Ghorbani et al., 2019).

Remarkably, as shown in Figures 6a and 6b, our results remain valid in this nonlinear setting. While this does not guarantee that our implicit bias characterization will hold in

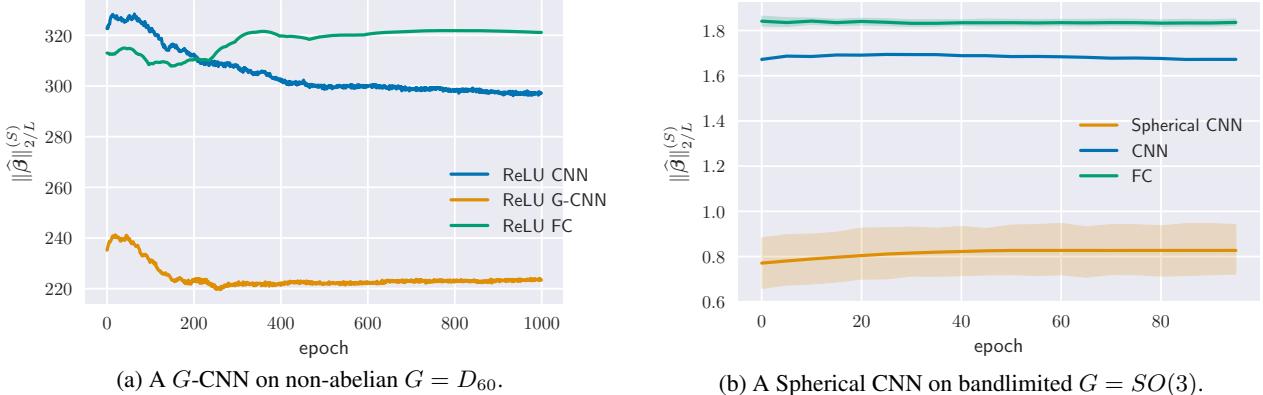


Figure 6: Group Fourier norms for *nonlinear* architectures with ReLU activations show that nonlinear G-CNNs implicitly regularize locally. Both figures track the mean norm of the per-sample local linearizations of each network. Figure 6a is on a network with a final linear layer, and performs binary classification on 10 isotropic Gaussian data points. Figure 6b is on a spherical CNN, trained on rotated spherical MNIST (Cohen et al., 2018). See Appendix E.3 for details.

more general settings, it is encouraging that our theoretical predictions seem to numerically hold, despite the violation of linearity and, in the case of Figure 6b, the cross-entropy loss function. Additional figures detailing the real-space behaviour are provided in Appendix E.

7. Discussion

In this work, we have shown that L -layer linear G-CNNs with full width kernels are biased towards sparse solutions in the Fourier regime regularized by the $2/L$ -Schatten norm over Fourier matrix coefficients. Our analysis applies to linear G-CNNs, over either finite groups or infinite groups with band-limited inputs, which are trained to perform binary classification. In advancing our results on implicit regularization, we highlight some limitations of this work and important future directions:

- **Nonlinearities:** Adding nonlinearities to the G-CNNs studied here expands the space of functions which the G-CNNs can express, but implicit regularization in this nonlinear setting may be challenging to characterize as G-CNNs are no longer linear predictors. Local analysis may be possible in special cases, *e.g.*, in the infinite width limit (Lee et al., 2017).
- **Bounded width kernels:** Our results apply to full-width kernels, supported on the entire group. Expanding results to bounded-width kernels, *i.e.*, those with sparse support, is an obvious future direction, though prior work indicates that closed form solutions may not exist for these special cases (Jagadeesan et al., 2021).
- **Different loss function and learning settings:** We study the exponential loss function on binary classification. It is an open question how the implicit bias

changes for classification over more than two classes, even for CNNs and fully-connected networks, as well as G-CNNs (Gunasekar et al., 2018b).

Although concise implicit regularization measures are challenging to analyze for realistic, nonlinear architectures, linear networks provide an instructive case study with precise analytic results. In this work, by proving that linear group equivariant CNNs trained with gradient descent are regularized towards low-rank matrices in Fourier space, we hope to advance the broader agenda of understanding generalization, and in particular how networks with diverse architectures — particularly those with built-in symmetries — learn.

ACKNOWLEDGMENTS

We thank Ankur Moitra and Chulhee Yun for helpful discussions early in the development of this project. HL is supported by the Fannie and John Hertz Foundation and the National Science Foundation Graduate Research Fellowship under Grant No. 1745302.

References

- B. Anderson, T.-S. Hy, and R. Kondor. *Cormorant: Covariant Molecular Neural Networks*. Curran Associates Inc., Red Hook, NY, USA, 2019.
- S. Bulusu, M. Favoni, A. Ipp, D. I. Müller, and D. Schuh. Generalization capabilities of translationally equivariant neural networks. *arXiv preprint arXiv:2103.14686*, 2021.
- J. O. Caro, Y. Ju, R. Pyle, S. Dey, W. Brendel, F. Anselmi, and A. Patel. Local convolutions cause an implicit bias towards high frequency adversarial examples, 2021.

- F. H. Clarke. Generalized gradients and applications. *Transactions of the American Mathematical Society*, 205:247–262, 1975.
- T. S. Cohen and M. Welling. Group equivariant convolutional networks. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning - Volume 48*, ICML’16, page 2990–2999. JMLR.org, 2016a.
- T. S. Cohen and M. Welling. Group Equivariant Convolutional Networks. *arXiv*, 2016b.
- T. S. Cohen, M. Geiger, J. Köhler, and M. Welling. Spherical cnns. *arXiv preprint arXiv:1801.10130*, 2018.
- T. S. Cohen, M. Geiger, and M. Weiler. A general theory of equivariant cnns on homogeneous spaces. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d’Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL <https://proceedings.neurips.cc/paper/2019/file/b9cfe8b6042cf759dc4c0cccb27a6737-Paper.pdf>.
- D. L. Donoho and P. B. Stark. Uncertainty principles and signal recovery. *SIAM Journal on Applied Mathematics*, 49(3):906–931, 1989.
- D. S. Dummit and R. M. Foote. *Abstract Algebra*, volume 3. Wiley Hoboken, 2004.
- B. Elesedy and S. Zaidi. Provably strict generalisation benefit for equivariant models. *arXiv preprint arXiv:2102.10333*, 2021.
- C. Esteves, C. Allen-Blanchette, A. Makadia, and K. Daniilidis. Learning $\text{so}(3)$ equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.
- R. Gens and P. M. Domingos. Deep symmetry networks. *Advances in neural information processing systems*, 27: 2537–2545, 2014.
- A. Ghorbani, A. Abid, and J. Zou. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3681–3688, 2019.
- S. Gunasekar, J. Lee, D. Soudry, and N. Srebro. Characterizing implicit bias in terms of optimization geometry. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1832–1841. PMLR, 10–15 Jul 2018a. URL <http://proceedings.mlr.press/v80/gunasekar18a.html>.
- S. Gunasekar, J. D. Lee, D. Soudry, and N. Srebro. Implicit bias of gradient descent on linear convolutional networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018b. URL <https://proceedings.neurips.cc/paper/2018/file/0e98aeeb54acf612b9eb4e48a269814c-Paper.pdf>.
- M. Jagadeesan, I. P. Razenshteyn, and S. Gunasekar. Inductive bias of multi-channel linear convolutional networks with bounded weight norm. *CoRR*, abs/2102.12238, 2021. URL <https://arxiv.org/abs/2102.12238>.
- R. Kondor. A novel set of rotationally and translationally invariant features for images based on the non-commutative bispectrum. *arXiv preprint cs/0701127*, 2007.
- R. Kondor and S. Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups, 2018.
- R. Kondor, Z. Lin, and S. Trivedi. Clebsch–gordan nets: a fully fourier space spherical convolutional neural network. *Advances in Neural Information Processing Systems*, 31: 10117–10126, 2018.
- J. Lee, Y. Bahri, R. Novak, S. S. Schoenholz, J. Pennington, and J. Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.
- A. S. Lewis. The convex analysis of unitarily invariant matrix functions. *J. Convex Anal*, pages 173–183, 1995.
- C. Lyle, M. van der Wilk, M. Kwiatkowska, Y. Gal, and B. Bloem-Reddy. On the benefits of invariance in neural networks. *arXiv preprint arXiv:2005.00178*, 2020.
- K. Lyu and J. Li. Gradient descent maximizes the margin of homogeneous neural networks. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=SJeL1gBKPS>.
- E. Matusiak, M. Özaydin, and T. Przebinda. The donoho–stark uncertainty principle for a finite abelian group. *Acta Math. Univ. Comenianae*, 73(2):155–160, 2004.
- R. Meshulam. An uncertainty inequality for groups of order pq . *European journal of combinatorics*, 13(5):401–407, 1992.
- F. Nicola and S. I. Trapasso. On the stability of deep convolutional neural networks under irregular or random deformations, 2021.
- N. Razin and N. Cohen. Implicit regularization in deep learning may not be explainable by norms. In H. Larochelle,

M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 21174–21187. Curran Associates, Inc., 2020. URL <https://proceedings.neurips.cc/paper/2020/file/f21e255f89e0f258accbe4e984eef486-Paper.pdf>.

M. Reisert. Group integration techniques in pattern analysis—a kernel view. 2008.

A. Sannai and M. Imaizumi. Improved generalization bound of group invariant/equivariant deep networks via quotient feature space. *arXiv preprint arXiv:1910.06552*, 2019.

D. Soudry, E. Hoffer, M. S. Nacson, S. Gunasekar, and N. Srebro. The implicit bias of gradient descent on separable data. *Journal of Machine Learning Research*, 19(70):1–57, 2018. URL <http://jmlr.org/papers/v19/18-188.html>.

A. Terras. *Fourier analysis on finite groups and applications*. Number 43. Cambridge University Press, 1999.

G. A. Watson. Characterization of the subdifferential of some matrix norms. *Linear algebra and its applications*, 170(0):33–45, 1992.

M. Weiler and G. Cesa. General E(2)-Equivariant Steerable CNNs. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

A. Wigderson and Y. Wigderson. The uncertainty principle: variations on a theme. *Bulletin of the American Mathematical Society*, 58(2):225–261, 2021.

D. Yarotsky. Universal approximations of invariant maps by neural networks. *Constructive Approximation*, pages 1–68, 2021.

C. Yun, S. Krishnan, and H. Mobahi. A unifying view on implicit bias in training linear neural networks. *CoRR*, abs/2010.02501, 2020. URL <https://arxiv.org/abs/2010.02501>.

M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Poczos, R. R. Salakhutdinov, and A. J. Smola. Deep sets. *Advances in Neural Information Processing Systems*, 30, 2017.

C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals. Understanding deep learning requires rethinking generalization. 2017. URL <https://arxiv.org/abs/1611.03530>.

A. Proofs

We begin with a simple lemma detailing the Fourier space end-to-end predictor for a G-CNN.

Lemma A.1 (G-CNN in Fourier space). *A G-CNN given by $\langle \mathbf{x} * \mathbf{w}_1 * \cdots * \mathbf{w}_{L-1}, \mathbf{w}_L \rangle$ is equivalent to $\langle \mathbf{x}, \mathbf{w}_L * \mathbf{w}_{L-1}^- * \cdots * \mathbf{w}_1^- \rangle$, or in Fourier space to $\frac{1}{|G|} \langle \widehat{\mathbf{x}}, \widehat{\mathbf{w}}_L \cdots \widehat{\mathbf{w}}_1 \rangle_M$.*

Proof.

$$\begin{aligned}
 N(\mathbf{x}) &= \langle \mathbf{x} * \mathbf{w}_1 * \cdots * \mathbf{w}_{L-1}, \mathbf{w}_L \rangle \\
 &= \frac{1}{|G|} \langle \mathcal{F}_M(\mathbf{x} * \mathbf{w}_1 * \cdots * \mathbf{w}_{L-1}), \mathcal{F}_M \mathbf{w}_L \rangle_M \\
 &= \frac{1}{|G|} \text{tr}[\mathcal{F}_M(\mathbf{x} * \mathbf{w}_1 * \cdots * \mathbf{w}_{L-1}) \widehat{\mathbf{w}}_L^\dagger] \\
 &= \frac{1}{|G|} \text{tr}[\mathcal{F}_M(\mathbf{x} * \mathbf{w}_1 * \cdots * \mathbf{w}_{L-2}) \widehat{\mathbf{w}}_{L-1}^\dagger \widehat{\mathbf{w}}_L^\dagger] \\
 &= \frac{1}{|G|} \text{tr}[\widehat{\mathbf{x}} \widehat{\mathbf{w}}_1^\dagger \cdots \widehat{\mathbf{w}}_{L-1}^\dagger \widehat{\mathbf{w}}_L^\dagger] \\
 &= \frac{1}{|G|} \langle \widehat{\mathbf{x}}, \widehat{\mathbf{w}}_L \cdots \widehat{\mathbf{w}}_1 \rangle_M
 \end{aligned} \tag{A.1}$$

To see the stated equality in real space, observe that $\widehat{\mathbf{w}_1^-} = \widehat{\mathbf{w}_1}^\dagger$ by definition. Thus,

$$\mathcal{F}_M(\mathbf{w}_L * \mathbf{w}_{L-1}^- * \cdots * \mathbf{w}_1^-) = \widehat{\mathbf{w}}_L \cdots \widehat{\mathbf{w}}_1$$

□

A.1. Abelian

The proof of [Proposition 5.2](#) is included below.

First, recall the primary theorem of [Yun et al. \(2020\)](#):

Theorem A.2 (paraphrased from [\(Yun et al., 2020\)](#)). *If there exists $\lambda > 0$ such that the initial directions $\bar{\mathbf{v}}_1, \dots, \bar{\mathbf{v}}_L$ of the network parameters satisfy $|[\mathcal{F}\bar{\mathbf{v}}_\ell]_j|^2 - |[\mathcal{F}\bar{\mathbf{v}}_L]_j|^2 \geq \lambda$ for all $\ell \in [L-1]$ and $j \in [m]$, i.e. of the Fourier transform magnitudes of the initial directions look sufficiently different pointwise (which is likely for e.g. a random initialization), then $\beta(\Theta(t))$ converges in a direction that aligns with $\mathbf{S}^T \boldsymbol{\rho}_\infty$ where $\boldsymbol{\rho}_\infty \in \mathbb{C}^m$ denotes a stationary point of the following optimization program:*

$$\min_{\boldsymbol{\rho} \in \mathbb{C}^m} \|\boldsymbol{\rho}\|_{2/L} \quad \text{s.t.} \quad y_i \mathbf{x}_i^T \mathcal{F}^T v \boldsymbol{\rho} \geq 1, \forall i \in [n] \tag{A.2}$$

Since $\mathbf{S} = \mathcal{F}$ is invertible, then in fact $\beta(\Theta(t))$ converges in a direction that aligns with a stationary point \mathbf{z}_∞ of the following optimization program:

$$\min_{\mathbf{z} \in \mathbb{C}^m} \|\mathcal{F}\mathbf{z}\|_{2/L} \quad \text{s.t.} \quad y_i \mathbf{x}_i^T \mathbf{z} \geq 1, \forall i \in [n] \tag{A.3}$$

We proceed by showing that abelian G-CNNs can be written as a vector of parameters contracted (according to tensor operations) with an orthogonally decomposable data tensor, which is the primary condition for [Theorem A.2](#) to hold.

Proposition A.3 (paraphrased from [\(Yun et al., 2020\)](#)). *Let $\mathbf{M}(\mathbf{x})$ be a function that maps data $\mathbf{x} \in \mathbb{R}^d$ to a data tensor $\mathbf{M}(\mathbf{x}) \in \mathbb{R}^{k_1 \times k_2 \times \cdots \times k_L}$. The data input into an L -layer tensorized neural network can be written in the form of an **orthogonally decomposable data tensor** if there exists a full column rank matrix $\mathbf{S} \in \mathbb{C}^{m \times d}$ and semi-unitary matrices $\mathbf{U}_1, \dots, \mathbf{U}_L \in \mathbb{C}^{k_\ell \times m}$ where $d \leq m \leq \min_\ell k_\ell$ such that $\mathbf{M}(\mathbf{x})$ can be written as:*

$$\mathbf{M}(\mathbf{x}) = \sum_{j=1}^m [\mathbf{Sx}]_j ([\mathbf{U}_1]_{:,j} \otimes [\mathbf{U}_2]_{:,j} \otimes \cdots \otimes [\mathbf{U}_L]_{:,j}) \tag{A.4}$$

and such that the network output is the tensor multiplication between $\mathbf{M}(x)$ and each layer's parameters:

$$\begin{aligned}\text{NN}(\mathbf{x}; \Theta) &= \mathbf{M}(\mathbf{x}) \cdot (\mathbf{W}_1, \dots, \mathbf{W}_L) \\ &= \sum_{i_1=1}^d \cdots \sum_{i_L=1}^d \mathbf{M}(\mathbf{x})_{i_1 \dots i_L} (\mathbf{W}_1)_{i_1} \cdots (\mathbf{W}_L)_{i_L}\end{aligned}$$

For [Proposition A.3](#). By direct manipulation:

$$\begin{aligned}f(\mathbf{x}; \Theta) &= \mathbf{M}(\mathbf{x}) \cdot (\mathbf{v}_1, \dots, \mathbf{v}_L) \\ &= \sum_{i_1=1}^d \cdots \sum_{i_L=1}^d \mathbf{M}(\mathbf{x})_{i_1 \dots i_L} (\mathbf{v}_1)_{i_1} \cdots (\mathbf{v}_L)_{i_L} \\ &= \sum_{i_1=1}^d \cdots \sum_{i_L=1}^d \left(\sum_{j=1}^d [\mathbf{Sx}]_j ([\mathbf{U}_1]_{\cdot, j} \otimes [\mathbf{U}_2]_{\cdot, j} \otimes \cdots \otimes [\mathbf{U}_L]_{\cdot, j}) \right)_{i_1 \dots i_L} (\mathbf{v}_1)_{i_1} \cdots (\mathbf{v}_L)_{i_L} \\ &= \sum_{i_1=1}^d \cdots \sum_{i_L=1}^d \left(\sum_{j=1}^d [\mathbf{Sx}]_j ([\mathbf{U}_1]_{i_1, j} [\mathbf{U}_2]_{i_2, j} \cdots [\mathbf{U}_L]_{i_L, j}) \right) (\mathbf{v}_1)_{i_1} \cdots (\mathbf{v}_L)_{i_L} \\ &= \sum_{j=1}^d [\mathbf{Sx}]_j ([\mathbf{U}_1^T \mathbf{v}_1]_{i_1} [\mathbf{U}_2^T \mathbf{v}_2]_{i_2} \cdots [\mathbf{U}_L^T \mathbf{v}_L]_{i_L}) \\ &= d^{\frac{L-1}{2}} \sum_{j=1}^d [\mathcal{F}\mathbf{x}]_j ([\overline{\mathcal{F}\mathbf{v}_1}]_{i_1} [\overline{\mathcal{F}\mathbf{v}_2}]_{i_2} \cdots [\overline{\mathcal{F}\mathbf{v}_L}]_{i_L}) \\ &= d^{\frac{L-1}{2}} \sum_{j=1}^d [\mathcal{F}\mathbf{x}]_j ([\overline{\mathcal{F}\mathbf{v}_1}]_{i_1} [\overline{\mathcal{F}\mathbf{v}_2}]_{i_2} \cdots [\overline{\mathcal{F}\mathbf{v}_L}]_{i_L}) \\ &= d^{\frac{L-1}{2}} \sum_{j=1}^d [\mathcal{F}\mathbf{x}]_j ([\overline{\mathcal{F}\mathbf{v}_1}]_{i_1} [\overline{\mathcal{F}\mathbf{v}_2}]_{i_2} \cdots [\overline{\mathcal{F}\mathbf{v}_L}]_{i_L}) \\ &= \langle \mathcal{F}\mathbf{x}, \mathcal{F}\mathbf{v}_1 \odot \cdots \odot \mathcal{F}\mathbf{v}_L \rangle \\ &= \langle \mathcal{F}\mathbf{x} \odot \overline{\mathcal{F}\mathbf{v}_1}, \mathcal{F}\mathbf{v}_2 \odot \cdots \odot \mathcal{F}\mathbf{v}_L \rangle \\ &= \langle \mathcal{F}(\mathbf{x} \star \mathbf{v}_1) \odot \overline{\mathcal{F}\mathbf{v}_2}, \mathcal{F}\mathbf{v}_3 \odot \cdots \odot \mathcal{F}\mathbf{v}_L \rangle \\ &= \langle \mathcal{F}(\mathbf{x} \star \mathbf{v}_1 \star \mathbf{v}_2) \odot \overline{\mathcal{F}\mathbf{v}_3}, \mathcal{F}\mathbf{v}_4 \odot \cdots \odot \mathcal{F}\mathbf{v}_L \rangle \\ &= \langle \mathcal{F}(\mathbf{x} \star \mathbf{v}_1 \star \mathbf{v}_2 \star \cdots \star \mathbf{v}_{L-1}), \mathcal{F}\mathbf{v}_L \rangle \\ &= \langle \mathbf{x} \star \mathbf{v}_1 \star \cdots \star \mathbf{v}_{L-1}, \mathbf{v}_L \rangle\end{aligned}$$

Here, we have used that the filters are real-valued. □

Note that [Theorem 5.3](#) then merely requires that $\|\mathcal{F}^{-T} \mathbf{z}\| = \|\overline{\mathcal{F}\mathbf{z}}\| = \|\mathcal{F}\mathbf{z}\|$, for real-valued \mathbf{z} .

A.1.1. FUNDAMENTAL THEOREM OF FINITE ABELIAN GROUPS

While the proof in the previous section is complete and correct, intuition (and/or alternate analysis) for abelian groups is aided by the important fact that all finite abelian groups are a direct product of cyclic groups.

Theorem A.4 (Fundamental theorem of finite abelian groups ([Dummit and Foote, 2004](#))). *Any finite abelian group is a direct product of a finite number of cyclic groups whose orders are prime powers uniquely determined by the group.*

Given a decomposition of an abelian group G into k cyclic groups $C_{d_1} \times \cdots \times C_{d_k}$, one can easily construct the group Fourier transform as a Kronecker product of discrete Fourier transform matrices which are the group Fourier transforms of

the respective cyclic groups.

$$G = C_{d_1} \times \cdots \times C_{d_k} \implies \mathcal{F} = \bigotimes_{i=1}^k \mathcal{F}_{d_i}, \quad (\text{A.5})$$

where \bigotimes denotes the Kronecker product over matrices and \mathcal{F}_d is the standard (unitary) discrete Fourier transform matrix of dimension d defined as

$$\mathcal{F}_d = \frac{1}{\sqrt{d}} \begin{bmatrix} \omega_d^{0 \cdot 0} & \omega_d^{0 \cdot 1} & \dots & \omega_d^{0 \cdot (d-1)} \\ \omega_d^{1 \cdot 0} & \omega_d^{1 \cdot 1} & \dots & \omega_d^{1 \cdot (d-1)} \\ \vdots & \vdots & \ddots & \vdots \\ \omega_d^{(d-1) \cdot 0} & \omega_d^{(d-1) \cdot 1} & \dots & \omega_d^{(d-1) \cdot (d-1)} \end{bmatrix}, \quad \omega_d = e^{-\frac{2\pi i}{d}}. \quad (\text{A.6})$$

From this result, it is clear that the desired properties of the Fourier transform and convolution hold.

A.2. Non-abelian

Theorem A.5. Consider a classification task with ground-truth linear predictor β , trained via a linear G -CNN architecture $\text{NN}(\mathbf{x}) = \langle \mathbf{x} \star \mathbf{w}_1 \star \cdots \star \mathbf{w}_{L-1}, \mathbf{w}_L \rangle$ (see Section 5 for architecture details) with $L \geq 2$ layers under the exponential loss. Then for almost any datasets $\{\mathbf{x}_i, y_i\}$ separable by β , any bounded sequence of step sizes η_t , and almost all initializations, suppose that:

- The loss $\mathcal{L}(\mathbf{W})$ converges to 0
- The gradients with respect to the end-to-end linear predictor, $\nabla_\beta \mathcal{L}(\mathcal{P}(\mathbf{W}^t))$, converge in direction as $t \rightarrow \infty$
- The iterates \mathbf{W}^t themselves converge in direction as $t \rightarrow \infty$ to a separator $\mathcal{P}(\mathbf{W}^t)$ with positive margin

When $L = 2$, we need an additional technical assumption, Assumption A.7. Then, the resultant linear predictor $\widehat{\beta}^\infty$ is a positive scaling of a first order stationary point of the optimization problem:

$$\min_{\beta} \|\widehat{\beta}\|_{2/L}^{(S)} \quad \text{s.t.} \quad \forall n, y_n \langle \beta_n, \mathbf{x}_n \rangle_M \geq 1 \quad (\text{A.7})$$

In this section, we prove the non-abelian case, Theorem A.5. The proof of our result proceeds according to the following outline:

1. By applying a general result of Gunasekar et al. (2018b), Theorem A.6, we characterize the implicit regularization in the full space of parameters, \mathbf{W} (in contrast to the end-to-end linear predictor β), as the stationary point of an optimization problem Equation A.9 in \mathbf{W} .
2. Separately, we define a *distinct* optimization problem, Equation A.7 in β . The goal is to demonstrate that stationary points of Equation A.9 are a subset of the stationary points of Equation A.7.
3. The *necessary* KKT conditions for Equation A.9 characterize its stationary points. Using this characterization, we show that the *sufficient* KKT optimality conditions for Equation A.7 are in fact also satisfied for the corresponding end-to-end predictor. Thus, we show that for any stationary point \mathbf{W}^\dagger of Equation A.9, the linear predictor $\mathcal{P}(\mathbf{W}^\dagger)$ is a stationary point of Equation A.7.

First, recall that Gunasekar et al. (2018b) prove the following general result about the implicit regularization of any homogeneous polynomial parametrization. (Lyu and Li (2020) later strengthened this result to the case of arbitrary homogeneous mappings, and showed furthermore that the margin is monotonically increasing under gradient flow, but the result of Gunasekar et al. (2018b) is suitable for our needs.)

Theorem A.6 (Homogeneous polynomial parametrization, Theorem 4 of (Gunasekar et al., 2018b)). *Let W be the concatenation of all (real-valued) parameters \mathbf{W}_i . For any homogeneous polynomial map $\mathcal{P} : \mathbb{R}^P \rightarrow \mathbb{R}^{|G|}$ from parameters $\mathbf{W}_i \in \mathbb{R}^P$ to linear predictors, almost all datasets $\{\mathbf{x}_n, y_n\}_{n=1}^N$ separable by the ground truth predictor $\beta := \{\mathcal{P}(\mathbf{W}) : \mathbf{W} \in \mathbb{R}^P\}$, almost all initializations \mathbf{W}^0 , and any bounded sequence of step sizes $\{\eta_t\}_t$, consider the gradient descent updates:*

$$\mathbf{W}^{t+1} = \mathbf{W}^t - \eta_t \nabla_{\mathbf{W}} \mathcal{L}(\mathbf{W}^t) = \mathbf{W}^t - \eta_t \nabla_{\mathbf{W}} \mathcal{P}(\mathbf{W}^t) \nabla_{\beta} \mathcal{L}(\mathcal{P}(\mathbf{W}^t)) \quad (\text{A.8})$$

Suppose furthermore that the exponential loss converges to zero, that the gradients $\nabla_{\beta} \mathcal{L}(\beta^t)$ converge in direction, and that the iterates \mathbf{W}^t themselves converge in direction to yield a separator with positive margin. Then, the limit direction of the parameters $\bar{\mathbf{W}}^{\infty} = \lim_{t \rightarrow \infty} \frac{\mathbf{W}^{(t)}}{\|\mathbf{W}^{(t)}\|_2}$ is a positive scaling of a first order stationary point of the following optimization problem:

$$\min_{\mathbf{W} \in \mathbb{R}^P} \|\mathbf{W}\|_2^2 \quad \text{s.t.} \quad \forall n, y_n \langle \mathbf{x}_n, \mathcal{P}(\mathbf{W}) \rangle \geq 1. \quad (\text{A.9})$$

To keep track of constant factors, let $\mathbf{W}^{\infty} = \tau \bar{\mathbf{W}}^{\infty}$ denote the first order stationary point itself. Furthermore, let \mathbf{W}_i^{∞} denote the individual layers (or parameter blocks) comprising \mathbf{W}^{∞} , and similarly let $\bar{\mathbf{W}}_i^{\infty}$ denote the individual layers comprising $\bar{\mathbf{W}}^{\infty}$. We then have via the KKT conditions that:

$$\begin{aligned} & \exists \{\alpha_n : \alpha_n \geq 0\}_{n=1}^N \text{ s.t. } \alpha_n = 0 \text{ if } y_n \langle \mathbf{x}_n, \mathcal{P}(\mathbf{W}^{\infty}) \rangle > 1 \\ & \mathbf{W}_i^{\infty} = \nabla_{\mathbf{W}_i} \mathcal{P}(\mathbf{W}^{\infty}) \left[\sum_n \alpha_n y_n \mathbf{x}_n \right] = \nabla_{\mathbf{W}_i} \left\langle \mathcal{P}(\mathbf{W}^{\infty}), \sum_n \alpha_n y_n \mathbf{x}_n \right\rangle \end{aligned} \quad (\text{A.10})$$

While this is an interesting result alone, the goal of implicit regularization is to characterize the final linear predictor (which is some function \mathcal{P} of the complete parametrization W). To that end, consider the following optimization problem in β :

$$\min_{\beta} \|\hat{\beta}\|_{2/L}^{(S)} \quad \text{s.t.} \quad \forall n, y_n \langle \beta_n, \mathbf{x}_n \rangle \geq 1 \quad (\text{A.11})$$

We will leverage the *necessary* KKT conditions from Equation A.10 to show that first-order stationary points of Equation A.9 are (up to a scaling) also first-order stationary points of Equation A.7, using the *sufficient* KKT conditions for Equation A.7.

Using standard KKT sufficiency conditions, the first-order stationary points of Equation A.7 are those vectors β such that there exist $\tilde{\alpha}_1, \dots, \tilde{\alpha}_n$ satisfying:

1. **Feasibility:** $\forall n, y_n \langle \beta, \mathbf{x}_n \rangle \geq 1$ and $\tilde{\alpha}_i \geq 0 \ \forall i$
2. **Complementary slackness:** $\forall i, \tilde{\alpha}_i = 0$ if $y_n \langle \beta, \mathbf{x}_n \rangle > 1$
3. **Membership in subdifferential:** $\sum_n \tilde{\alpha}_n y_n \hat{\mathbf{x}}_n \in \partial^o \|\hat{\beta}\|_{2/L}^{(S)}$

In the third condition above, ∂^o is the local sub-differential of (Clarke, 1975): $\partial^o f(\beta) = \text{conv}\{\lim_{i \rightarrow \infty} \nabla f(\beta + \mathbf{h}_i) : \mathbf{h}_i \rightarrow 0\}$ ⁹.

We will need the following assumption in the special case $L = 2$:

Assumption A.7 ($L = 2$ bounded subgradient). Let $\hat{\mathbf{z}} = \sum_n \tilde{\alpha}_n y_n \hat{\mathbf{x}}_n$ result from the KKT conditions of the optimization problem in \mathbf{W} , Equation A.10, as described previously. Then, we assume that $\|\hat{\mathbf{z}}\|_{\infty}^{(S)} \leq 1$.

Let $\beta^{\infty} = \mathcal{P}(\mathbf{W}^{\infty})$ and let $\tilde{\alpha}_i = \frac{1}{\gamma} \alpha_i$ for all i , where γ is equal to $(\|\hat{\beta}^{\infty}\|_{2/L}^{(S)})$ for $L > 2$ and to 1 otherwise. (Note that by homogeneity of \mathcal{P} , $\mathcal{P}(\mathbf{W}^{\infty}) = \mathcal{P}(\tau \bar{\mathbf{W}}^{\infty}) = \tau^L \mathcal{P}(\bar{\mathbf{W}}^{\infty})$.) We will check these conditions one by one for β^{∞} and $\tilde{\alpha}_i$, with the first two following immediately from Theorem A.6 and the last one requiring the most manipulation.

Feasibility Trivially, $\tilde{\alpha}_i \geq 0 \ \forall i$ by definition of α in Equation A.10. Similarly, $y_n \langle \mathbf{x}_n, \beta^{\infty} \rangle = y_n \langle \mathbf{x}_n, \mathcal{P}(\mathbf{W}^{\infty}) \rangle \geq 1$.

⁹ \mathbf{h}_i is a sequence of vectors in some linear space, and we take $\mathbf{h}_i \rightarrow 0$ as an entry-wise statement. This is because the vectors are finite-dimensional, so all norms are equivalent.

Complementary slackness If $y_n \langle \beta^\infty, \mathbf{x}_n \rangle > 1 = y_n \langle \mathcal{P}(\mathbf{W}^\infty), \mathbf{x}_n \rangle > 1$, then $\tilde{\alpha}_n \propto \alpha_n = 0$.

Membership in subdifferential We first characterize the set $\partial^o \|\widehat{\beta}\|_{2/L}^{(S)}$ for a generic matrix β , and then show that $\widehat{\mathbf{z}} \triangleq \sum_n \tilde{\alpha}_n y_n \widehat{\mathbf{x}}_n \in \partial^o \|\widehat{\beta}^\infty\|_{2/L}^{(S)}$. When $L = 2$ and thus $p = 1$, the Schatten norm $\|\widehat{\beta}^\infty\|_1$ is indeed a norm and its subgradient is known; see e.g. Watson (1992). We restate this result below:

Lemma A.8 (Subdifferential of p -Schatten norm, $p = 1$). Suppose $L = 2$, such that $\frac{2}{L} = p = 1$. Let \mathbf{A} be an $n \times n$ complex-valued matrix. Then we have

$$\partial^o \|\mathbf{A}\|_1^{(S)} = \left\{ \mathbf{G} : \|\mathbf{A}\|_1^{(S)} = \text{tr}[\mathbf{G}^\dagger \mathbf{A}], \|\mathbf{G}\|_\infty^{(S)} \leq 1 \right\} \quad (\text{A.12})$$

When $L > 2$, previous works have characterized the subdifferential of the $2/L$ -Schatten norm. For example, Lewis (1995) characterizes the subdifferential of any unitary matrix norm $\partial \|\mathbf{X}\|$ as all those matrices with the same left and right singular vectors as \mathbf{X} , but whose singular values lie in the subdifferential of the corresponding norm of the singular values of \mathbf{X} (Corollary 2.5). For our purposes, a different formulation of the subdifferential will be more useful later in the overall proof. To keep the paper relatively self-contained, we prove the following result from scratch.

Lemma A.9 (Subdifferential of p -Schatten norm, $p < 1$). Suppose $L > 2$ and let $p = \frac{2}{L}$, such that $0 < p < 1$. Let \mathbf{A} be an $n \times n$ complex-valued matrix with singular value decomposition $\mathbf{A} = \mathbf{U} \mathbf{D} \mathbf{V}^\dagger$. Let Π_V project onto the row space of \mathbf{A} , i.e. $\Pi_V(\mathbf{M}) = \mathbf{M} \mathbf{V} \mathbf{V}^\dagger$. Then we have

$$\partial^o \|\mathbf{A}\|_p^{(S)} = \left\{ \mathbf{G} : \mathbf{A}^\dagger \mathbf{G} = \frac{1}{\|\mathbf{A}\|_p^{(S)}} \sqrt{\mathbf{A}^\dagger \mathbf{A}}^p \text{ and } \mathbf{G} \mathbf{A}^\dagger = \frac{1}{\|\mathbf{A}\|_p^{(S)}} \sqrt{\mathbf{A} \mathbf{A}^\dagger}^p \right\} \quad (\text{A.13})$$

Proof. Suppose \mathbf{A} is rank r and has singular value decomposition $\mathbf{A} = \sum_{i=1}^r d_i \mathbf{u}_i \mathbf{v}_i^\dagger$, where $d_i > 0$ for all i . Consider unit vectors $\mathbf{W}_1, \dots, \mathbf{W}_{n-r}$ which are a basis for the orthogonal subspace to $\text{Span}(\mathbf{u}_1, \dots, \mathbf{u}_r)$, and unit vectors $\mathbf{s}_1, \dots, \mathbf{s}_{n-r}$ which are a basis for the orthogonal subspace to $\text{Span}(\mathbf{v}_1, \dots, \mathbf{v}_r)$. Treating the space of $n \times n$ complex matrices as a n^2 -dimensional linear space, we see that $\{\mathbf{u}_i \mathbf{v}_i^\dagger\}_{i=1}^r$ form a basis for an r -dimensional subspace. Let Π_A denote the projector onto this space, $\Pi_A = \sum_{i=1}^r \mathbf{u}_i \mathbf{v}_i^\dagger$. Note that $\Pi_A \mathbf{A} = \mathbf{A}$. For any set of $n - r$ sequences $\{\{\epsilon_{i,m}\}_{m=1}^\infty\}_{i=1}^{n-r}$ such that $\lim_{m \rightarrow \infty} \epsilon_{m,i} = 0$ for all $i = 1, \dots, n - r$, consider the particular sequence of matrices $\{\mathbf{H}_m\}_{m=1}^\infty$ defined by

$$\mathbf{H}_m = \sum_{i=1}^{n-r} \epsilon_{m,i} \mathbf{W}_i \mathbf{s}_i^\dagger$$

By definition, $\lim_{m \rightarrow \infty} \|\mathbf{H}_m\|_{\text{Fro}} = 0$, where $\|\mathbf{H}_m\|_{\text{Fro}} \triangleq \|\mathbf{H}_m\|_2^{(S)}$. Also, if \mathbf{M} is a full rank matrix with singular value decomposition $\mathbf{A} \mathbf{D} \mathbf{B}^\dagger$, $\|\mathbf{M}\|_p^{(S)}$ is differentiable at \mathbf{M} and $\nabla \|\mathbf{M}\|_p^{(S)} = \frac{1}{\|\mathbf{M}\|_p^{(S)}} \mathbf{A} \mathbf{D}^{p-1} \mathbf{B}^\dagger$. For convenience of notation, let \mathbf{U} be the matrix with i^{th} column \mathbf{u}_i , \mathbf{W} the matrix with i^{th} column \mathbf{W}_i , and similarly for \mathbf{V} and \mathbf{S} with respect to vectors \mathbf{v}_i and \mathbf{s}_i respectively. Also, let \mathbf{D} be the diagonal matrix with d_i on the i^{th} diagonal.

Combining this fact with the construction of \mathbf{H}_m , we have that

$$\nabla \|\mathbf{A} + \mathbf{H}_m\|_p^{(S)} = \nabla \left\| \sum_{i=1}^r d_i \mathbf{u}_i \mathbf{v}_i^\dagger + \sum_{i=1}^{n-r} \epsilon_{m,i} \mathbf{W}_i \mathbf{s}_i^\dagger \right\|_p^{(S)} \quad (\text{A.14})$$

$$= \nabla \left\| [\mathbf{U} \quad \mathbf{W}] \begin{bmatrix} \mathbf{D} & 0 & \dots & 0 \\ 0 & \epsilon_{m,1} & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & \epsilon_{m,n-r} \end{bmatrix} [\mathbf{V}^\dagger \quad \mathbf{S}^\dagger] \right\|_p^{(S)} \quad (\text{A.15})$$

$$= \frac{1}{(\sum_{i=1}^r d_i^p + \sum_{i=1}^{n-r} \epsilon_{m,i}^p)^{\frac{1}{p}}} \left(\mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger + \mathbf{W} \begin{bmatrix} \epsilon_{m,1}^{p-1} & \dots & 0 \\ \vdots & \ddots & \ddots \\ 0 & \dots & \epsilon_{m,n-r}^{p-1} \end{bmatrix} \mathbf{S}^\dagger \right) \quad (\text{A.16})$$

In the limit as m goes to infinity, $\sum_{i=1}^{n-r} \epsilon_{m,i}^p$ approaches 0. However, $p - 1 < 0$ implies that $\lim_{m \rightarrow \infty} \epsilon_{m,i}^{p-1} = \pm\infty$. By taking convex combinations, one can create any matrix with left and right singular vectors \mathbf{W} and \mathbf{S}^\dagger . Formally, we have:

$$\partial^o \|\mathbf{A}\|_p^{(S)} = \text{conv} \left\{ \lim_{m \rightarrow \infty} \nabla \|\mathbf{A} + \mathbf{H}_m\|_p : \mathbf{H}_m \rightarrow 0 \right\} \quad (\text{A.17})$$

$$= \frac{1}{\|\mathbf{A}\|_p^{(S)}} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger + \frac{1}{\|\mathbf{A}\|_p^{(S)}} \text{conv} \left\{ \lim_{m \rightarrow \infty} \mathbf{W} \begin{bmatrix} \epsilon_{m,1}^{p-1} & \dots & 0 \\ \vdots & \ddots & \ddots \\ 0 & \dots & \epsilon_{m,n-r}^{p-1} \end{bmatrix} \mathbf{S}^\dagger : \epsilon_{m,i} \rightarrow_m 0 \right\} \quad (\text{A.18})$$

$$= \frac{1}{\|\mathbf{A}\|_p^{(S)}} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger + \{\mathbf{W} \Sigma \mathbf{S}^\dagger : \Sigma \text{ is real and diagonal}\} \quad (\text{A.19})$$

Let Π_U project onto the column space of A , i.e. $\Pi_U(\mathbf{M}) = \mathbf{U} \mathbf{U}^\dagger \mathbf{M}$. Note that for any rank-one matrix $\mathbf{M} = \mathbf{a} \mathbf{b}^\dagger$, if \mathbf{a} is not orthogonal to each column of \mathbf{U} , then $\mathbf{U} \mathbf{U}^\dagger \mathbf{M} \neq 0$. Similarly, if \mathbf{b} is not orthogonal to each column of \mathbf{V} , then $\mathbf{M} \mathbf{V} \mathbf{V}^\dagger \neq 0$. Thus for an arbitrary matrix \mathbf{M} , by decomposing it into a sum of rank-one matrices via its SVD, we see that $\Pi_U \mathbf{M} = \Pi_V \mathbf{M} = 0$ implies that both the row and column spaces of \mathbf{M} are orthogonal to those of \mathbf{A} , respectively. Thus, we can project via Π_U and Π_V to disregard the second term of [Equation A.19](#), and obtain the following expression for the subgradient:

$$\partial^o \|\mathbf{A}\|_p^{(S)} = \left\{ \mathbf{G} : \Pi_U \mathbf{G} = \Pi_V \mathbf{G} = \left(\|\mathbf{A}\|_p^{(S)} \right)^{-p} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger \right\} \quad (\text{A.20})$$

Consider first the equality

$$\Pi_U \mathbf{G} = \left(\|\mathbf{A}\|_p^{(S)} \right)^{-p} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger \quad (\text{A.21})$$

We can left-multiply by \mathbf{A}^\dagger without changing the set of matrices \mathbf{G} satisfying this relation. To see this, one can check that if $\mathbf{A}^\dagger \Pi_U \mathbf{G} = \left(\|\mathbf{A}\|_p^{(S)} \right)^{-p} \mathbf{A}^\dagger \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger$, left-multiplying on both sides by $\mathbf{U} \mathbf{D}^{-1} \mathbf{V}^\dagger$ recovers [Equation A.21](#).

Similarly, consider the second equality:

$$\Pi_V \mathbf{G} = \left(\|\mathbf{A}\|_p^{(S)} \right)^{-p} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger \quad (\text{A.22})$$

We can right-multiply by \mathbf{A}^\dagger without changing the set of matrices \mathbf{G} satisfying this relation. To see this, one can check that if

$$(\Pi_V \mathbf{G}) \mathbf{A}^\dagger = \left(\|\mathbf{A}\|_p^{(S)} \right)^{-p} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger \mathbf{A}^\dagger$$

Then right-multiplying on both sides by $\mathbf{U} \mathbf{D}^{-1} \mathbf{V}^\dagger$ recovers [Equation A.22](#). Finally, observe that $\mathbf{A}^\dagger \Pi_U \mathbf{G} = \mathbf{V} \mathbf{D} \mathbf{U}^\dagger \mathbf{U} \mathbf{U}^\dagger \mathbf{G} = \mathbf{A}^\dagger \mathbf{G}$ and $(\Pi_V \mathbf{G}) \mathbf{A}^\dagger = \mathbf{G} \mathbf{V} \mathbf{V}^\dagger \mathbf{V} \mathbf{D} \mathbf{U}^\dagger = \mathbf{G} \mathbf{V} \mathbf{D} \mathbf{U}^\dagger = \mathbf{G} \mathbf{A}^\dagger$. Furthermore, $\mathbf{A}^\dagger \frac{1}{\|\mathbf{A}\|_p^{(S)}} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger = \mathbf{V} \mathbf{D}^p \mathbf{V}^\dagger = \frac{1}{\|\mathbf{A}\|_p^{(S)}} \sqrt{\mathbf{A}^\dagger \mathbf{A}}^p$ and $\frac{1}{\|\mathbf{A}\|_p^{(S)}} \mathbf{U} \mathbf{D}^{p-1} \mathbf{V}^\dagger \mathbf{A}^\dagger = \frac{1}{\|\mathbf{A}\|_p^{(S)}} \sqrt{\mathbf{A} \mathbf{A}^\dagger}^p$. This completes the proof of the lemma. \square

Lemmas [A.8](#) and [A.9](#) characterize the subdifferential of the Schatten norm. Now, we show that $\sum_n \tilde{\alpha}_n y_n \hat{\mathbf{x}}_n$ satisfies [Equation A.13](#).

Lemma A.10. *Recall that our G-CNN is given by $\text{NN}(x) = \langle \mathbf{x} \star \mathbf{w}_1 \star \dots \star \mathbf{w}_{L-1}, \mathbf{w}_L \rangle$, with the vector of parameters $\mathbf{W} = [\mathbf{w}_1 \dots \mathbf{w}_L]$ and end-to-end linear predictor given by $\mathcal{P}(\mathbf{W}) = \mathbf{w}_1 \star \dots \star \mathbf{w}_L$. Consider an arbitrary such vector of real-valued parameters. Also, we have assumed that the filters in real-space are real-valued, i.e. $\mathbf{w}_i \in \mathbb{R}^{|G|}$. Then the following relation holds:*

$$\mathcal{F}_M \nabla_{\mathbf{w}_\ell} \langle \mathcal{P}(\mathbf{W}), \mathbf{e}_i \rangle = \hat{\mathbf{w}}_{\ell+1}^\dagger \cdots \hat{\mathbf{w}}_L^\dagger \hat{\mathbf{e}}_i \hat{\mathbf{w}}_1^\dagger \cdots \hat{\mathbf{w}}_{\ell-1}^\dagger \quad (\text{A.23})$$

where e_i is the i^{th} standard basis vector.

Proof. Since $\langle \mathcal{P}(\mathbf{W}), e_i \rangle$ is real, we have $\langle \mathcal{P}(\mathbf{W}), e_i \rangle = \langle e_i, \mathcal{P}(\mathbf{W}) \rangle$. Plugging this in,

$$\begin{aligned}
 \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \mathcal{P}(\mathbf{W})[e_i] &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \langle \mathcal{P}(\mathbf{W}), e_i \rangle \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \langle e_i, \mathcal{P}(\mathbf{W}) \rangle \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \frac{1}{|G|} \langle \mathcal{F}_M e_i, \mathcal{F}_M \mathcal{P}(\mathbf{W}) \rangle_M \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \frac{1}{|G|} \langle \hat{e}_i, \hat{\mathbf{w}}_L \cdots \hat{\mathbf{w}}_1 \rangle_M \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \frac{1}{|G|} \text{tr}[\hat{e}_i (\hat{\mathbf{w}}_L \cdots \hat{\mathbf{w}}_1)^\dagger] \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \frac{1}{|G|} \text{tr}[\hat{\mathbf{w}}_{\ell+1}^\dagger \cdots \hat{\mathbf{w}}_L^\dagger \hat{e}_i \hat{\mathbf{w}}_1^\dagger \cdots \hat{\mathbf{w}}_\ell^\dagger] \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \frac{1}{|G|} \langle \hat{\mathbf{w}}_{\ell+1}^\dagger \cdots \hat{\mathbf{w}}_L^\dagger \hat{e}_i \hat{\mathbf{w}}_1^\dagger \cdots \hat{\mathbf{w}}_{\ell-1}^\dagger, \hat{\mathbf{w}}_\ell \rangle_M \\
 &= \mathcal{F}_M \nabla_{\mathbf{w}_\ell} \left\langle \mathcal{F}_M^{-1}(\hat{\mathbf{w}}_{\ell+1}^\dagger \cdots \hat{\mathbf{w}}_L^\dagger \hat{e}_i \hat{\mathbf{w}}_1^\dagger \cdots \hat{\mathbf{w}}_{\ell-1}^\dagger), \mathbf{w}_\ell \right\rangle \\
 &= \hat{\mathbf{w}}_{\ell+1}^\dagger \cdots \hat{\mathbf{w}}_L^\dagger \hat{e}_i \hat{\mathbf{w}}_1^\dagger \cdots \hat{\mathbf{w}}_{\ell-1}^\dagger
 \end{aligned} \tag{A.24}$$

□

Letting $\mathbf{z} = \sum_n \tilde{\alpha}_n y_n \mathbf{x}_n$ and $\mathbf{W} = \mathbf{W}^\infty$, Lemma A.10 implies that

$$\mathcal{F}_M \nabla_{\mathbf{w}_\ell^\infty} \langle \mathcal{P}(\mathbf{W}^\infty), \mathbf{z} \rangle = \hat{\mathbf{w}}_{\ell+1}^\infty \cdots \hat{\mathbf{w}}_L^\infty \hat{\mathbf{z}} \hat{\mathbf{w}}_1^\infty \cdots \hat{\mathbf{w}}_{\ell-1}^\infty \tag{A.25}$$

By combining Equation A.10 with Lemma A.10, we have

$$\frac{1}{\gamma} \hat{\mathbf{w}}_\ell^\infty = \frac{1}{\gamma} \nabla_{\mathbf{w}_\ell^\infty} \left\langle \mathcal{P}(\mathbf{W}^\infty), \sum_n \alpha_n y_n \mathbf{x}_n \right\rangle \tag{A.26}$$

$$= \nabla_{\mathbf{w}_\ell^\infty} \left\langle \mathcal{P}(\mathbf{W}^\infty), \sum_n \tilde{\alpha}_n y_n \mathbf{x}_n \right\rangle \tag{A.27}$$

$$= \hat{\mathbf{w}}_{\ell+1}^\infty \cdots \hat{\mathbf{w}}_L^\infty \hat{\mathbf{z}} \hat{\mathbf{w}}_1^\infty \cdots \hat{\mathbf{w}}_{\ell-1}^\infty \tag{A.28}$$

As a result:

$$\begin{aligned}
 \hat{\mathbf{w}}_\ell^\infty &= \gamma \hat{\mathbf{w}}_{\ell+1}^\infty \cdots \hat{\mathbf{w}}_L^\infty \hat{\mathbf{z}} \hat{\mathbf{w}}_1^\infty \cdots \hat{\mathbf{w}}_{\ell-1}^\infty \\
 \hat{\mathbf{w}}_\ell^\infty \hat{\mathbf{w}}_\ell^\infty \dagger &= \gamma \hat{\mathbf{w}}_{\ell+1}^\infty \cdots \hat{\mathbf{w}}_L^\infty \hat{\mathbf{z}} \hat{\mathbf{w}}_1^\infty \cdots \hat{\mathbf{w}}_\ell^\infty \dagger
 \end{aligned} \tag{A.29}$$

Applying this relation with $\ell = L$, we have that

$$\hat{\mathbf{w}}_L^\infty \hat{\mathbf{w}}_L^\infty \dagger = \gamma \hat{\mathbf{z}} \hat{\mathbf{w}}_1^\infty \cdots \hat{\mathbf{w}}_L^\infty \dagger \tag{A.30}$$

$$= \gamma \hat{\mathbf{z}} \hat{\beta}^\infty \dagger \tag{A.31}$$

Taking adjoints of both sides implies that $\hat{\mathbf{z}} \hat{\beta}^\infty \dagger$ is Hermitian, which will be useful later.

Let $\widehat{\beta}^\infty = \mathcal{F}_M \mathcal{P}(\mathbf{W}^\infty)$, from which we can derive the following recursion:

$$\begin{aligned}
 \widehat{\beta}^\infty \widehat{\beta}^{\infty\dagger} &= \widehat{\mathbf{w}}_L^\infty \cdots \widehat{\mathbf{w}}_1^\infty \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \\
 &= \gamma^1 \widehat{\mathbf{w}}_L^\infty \cdots \widehat{\mathbf{w}}_2^\infty \widehat{\mathbf{w}}_2^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \text{ by Equation A.29} \\
 &= \gamma^2 \widehat{\mathbf{w}}_L^\infty \cdots \widehat{\mathbf{w}}_3^\infty \widehat{\mathbf{w}}_3^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\mathbf{w}}_1^{\infty\dagger} \widehat{\mathbf{w}}_2^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \text{ again, by Equation A.29} \\
 &= \gamma^2 \widehat{\mathbf{w}}_L^\infty \cdots \widehat{\mathbf{w}}_3^\infty \widehat{\mathbf{w}}_3^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\beta}^{\infty\dagger} \text{ by definition of } \beta^\infty \\
 &= \gamma^2 \widehat{\mathbf{w}}_L^\infty \cdots \widehat{\mathbf{w}}_3^\infty \widehat{\mathbf{w}}_3^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} (\widehat{\mathbf{z}} \widehat{\beta}^{\infty\dagger})^2 \text{ by repeated application of Equation A.29} \\
 &= \gamma^L (\widehat{\mathbf{z}} \widehat{\beta}^{\infty\dagger})^L
 \end{aligned} \tag{A.32}$$

Similarly to Equation A.29, we have:

$$\widehat{\mathbf{w}}_\ell^{\infty\dagger} \widehat{\mathbf{w}}_\ell^\infty = \gamma \widehat{\mathbf{w}}_\ell^{\infty\dagger} \widehat{\mathbf{w}}_{\ell+1}^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{\ell-1}^{\infty\dagger} \tag{A.33}$$

By considering $\ell = 1$, we have $\widehat{\mathbf{w}}_1^{\infty\dagger} \widehat{\mathbf{w}}_1^\infty = \widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}}$, which shows that $\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}}$ is Hermitian as well.

Using Equation A.33, we can similarly reason about $\widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty$:

$$\begin{aligned}
 \widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty &= \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{w}}_L^\infty \cdots \widehat{\mathbf{w}}_1^\infty \\
 &= \gamma \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{L-1}^{\infty\dagger} (\widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{L-1}^{\infty\dagger}) \widehat{\mathbf{w}}_{L-1}^\infty \cdots \widehat{\mathbf{w}}_1^\infty \\
 &= \gamma (\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}}) \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{L-1}^{\infty\dagger} \widehat{\mathbf{w}}_{L-1}^\infty \cdots \widehat{\mathbf{w}}_1^\infty \\
 &= \gamma (\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}}) \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{L-1}^{\infty\dagger} \widehat{\mathbf{w}}_L^{\infty\dagger} \widehat{\mathbf{z}} \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{L-2}^{\infty\dagger} \widehat{\mathbf{w}}_{L-2}^\infty \cdots \widehat{\mathbf{w}}_1^\infty \\
 &= \gamma^2 (\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}})^2 \widehat{\mathbf{w}}_1^{\infty\dagger} \cdots \widehat{\mathbf{w}}_{L-2}^{\infty\dagger} \widehat{\mathbf{w}}_{L-2}^\infty \cdots \widehat{\mathbf{w}}_1^\infty \\
 &= \gamma^L (\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}})^L
 \end{aligned} \tag{A.34}$$

For $L > 2$, we have shown in Lemma A.9 that

$$\partial^o \|\widehat{\beta}^\infty\|_p^{(S)} = \left\{ \mathbf{G} : \widehat{\beta}^{\infty\dagger} \mathbf{G} = \frac{1}{\|\widehat{\beta}^\infty\|_p^{(S)}} \sqrt{\widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty}^p \text{ and } \mathbf{G} \widehat{\beta}^{\infty\dagger} = \frac{1}{\|\widehat{\beta}^\infty\|_p^{(S)}} \sqrt{\widehat{\beta}^\infty \widehat{\beta}^{\infty\dagger}}^p \right\} \tag{A.35}$$

Let's check that setting $G = \widehat{\mathbf{z}}$ satisfies this relation. Since $p = \frac{2}{L}$, $\frac{p}{2} = \frac{1}{L}$ and, by Equation A.32 and that $\widehat{\mathbf{z}} \widehat{\beta}^{\infty\dagger}$ is Hermitian:

$$(\widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty)^{p/2} = \gamma \widehat{\mathbf{z}} \widehat{\beta}^{\infty\dagger} \tag{A.36}$$

Similarly, by Equation A.34 and that $\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}}$ is Hermitian:

$$(\widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty)^{\frac{p}{2}} = \gamma \widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}} \tag{A.37}$$

By choice of γ , $\gamma = \|\widehat{\beta}^\infty\|_{\frac{2}{L}}^{(S)}$. Thus, $\widehat{\mathbf{z}} \in \partial^o \|\widehat{\beta}^\infty\|_p^{(S)}$ as desired for $L > 2$.

For $L = 2$, $p = 1$ and by Lemma A.8 we had the following expression for the subgradient:

$$\partial^o \|\mathbf{A}\|_1^{(S)} = \left\{ \mathbf{G} : \|\mathbf{A}\|_1^{(S)} = \text{tr}[\mathbf{G}^\dagger \mathbf{A}], \|\mathbf{G}\|_\infty^{(S)} \leq 1 \right\} \tag{A.38}$$

As a technical condition, we had to assume in Assumption A.7 that $\|\widehat{\mathbf{z}}\|_\infty^{(S)} \leq 1$. (We believe that with a refined analysis in future work, this assumption can be shown to be true given only our existing assumptions.) By the previous reasoning for $L = 2$, we had that $\widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty = \gamma^2 (\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}})^2$. Since $\widehat{\beta}^{\infty\dagger} \widehat{\beta}^\infty = \mathbf{U} \mathbf{D}^2 \mathbf{U}^\dagger$ is positive semi-definite and symmetric, and since $\widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}}$ is Hermitian, we can take the square root of both sides and obtain that $\mathbf{U} \mathbf{D} \mathbf{U}^\dagger = \gamma \widehat{\beta}^{\infty\dagger} \widehat{\mathbf{z}} = \gamma \widehat{\mathbf{z}}^\dagger \widehat{\beta}^\infty$. Thus, $\text{tr}[\widehat{\mathbf{z}}^\dagger \widehat{\beta}^\infty] = \text{tr}[\mathbf{D} \mathbf{U}^\dagger \mathbf{U}] = \text{tr}[\mathbf{D}] = \|\widehat{\beta}^\infty\|_1^{(S)}$, which is what was needed (since $\gamma = 1$ for the $L = 2$ case).

A.2.1. INFINITE GROUPS WITH BAND-LIMITED INPUTS

Here we consider the case of more general, infinite-dimensional compact Lie groups. Such groups admit a Fourier transform which is an operator between infinite-dimensional spaces, rather than a finite matrix as before, but which has the same key properties: a convolution theorem, and preservation of inner products. To be concrete, the Fourier transform is now defined as $\widehat{f}(\rho) = \int_{g \in G} \rho(g) f(g) \mu(g)$, where $\mu(g)$ denotes the Haar measure for the group.

Since it is impossible to store a general function $x : G \rightarrow \mathbb{R}$, one must make simplifying assumptions on the input x . A common assumption is that x is band-limited in Fourier space, i.e. \widehat{x} is supported only on a small (and finite) subset of Fourier coefficients contained within the irreps $\rho \in S$. For many such groups, there is a natural hierarchy of irreps (analogous to low frequencies and high frequencies in classical Fourier analysis), and so practical architectures typically assume only those corresponding to low frequencies are non-zero.

For ease of analysis, we will assume that the input functions and all convolutional filters are real-valued in Fourier space. The architecture of our G-CNN is the same as in the finite-dimensional group setting except that we will apply functions entirely in the finite-dimensional Fourier space over the irreps in S . Given a function \widehat{x} supported only on S in Fourier space, we run gradient descent over only the Fourier coefficients on S of filters \widehat{w}_i , and assume they are zero elsewhere. Before we proceed, we define \mathcal{F}_M in the natural way after restricting the irreps to the band-limiting space S :

$$\widehat{\mathbf{f}} = \mathcal{F}_M \mathbf{f} = \bigoplus_{\rho \in S} \widehat{f}(\rho)^{\oplus d_\rho} \in \mathrm{GL} \left(\sum_{\rho \in S} d_\rho^2, \mathbb{C} \right). \quad (\text{A.39})$$

In this band-limited space, for each filter, there are $\sum_{\rho \in S} d_\rho^2$ trainable parameters corresponding to the entries of the irreps in S . Note that these entries are also orthogonal to each other with respect to the inner product $\langle \cdot, \cdot \rangle_M$ and thus form a vector space of dimension $\sum_{\rho \in S} d_\rho^2$.

Recall that we previously applied [Theorem A.6](#) to the homogeneous polynomial parametrization from $(\mathbf{w}_1, \dots, \mathbf{w}_L) \rightarrow \langle \mathbf{x} * \mathbf{w}_1 * \dots * \mathbf{w}_{L-1}, \mathbf{w}_L \rangle$. Since we will operate in Fourier space, instead consider the homogeneous polynomial parametrization $\widehat{\mathbf{W}} \in \mathbb{R}^p$ containing the parameters stored in the matrices $\{\widehat{\mathbf{w}}_1, \dots, \widehat{\mathbf{w}}_L\}$. In other words, there are $p = L(\sum_{\rho \in S} d_\rho^2)$ parameters stored in the matrices contained in the set $\widehat{\mathbf{W}}$.

$$\widehat{\mathbf{W}}^{t+1} = \widehat{\mathbf{W}}^t - \eta_t \nabla_{\widehat{\mathbf{W}}} \mathcal{L}(\widehat{\mathbf{W}}^t) = \widehat{\mathbf{W}}^t - \eta_t \nabla_{\widehat{\mathbf{W}}} \mathcal{P}(\widehat{\mathbf{W}}^t) \nabla_{\widehat{\beta}} \mathcal{L}(\mathcal{P}(\widehat{\mathbf{W}}^t)) \quad (\text{A.40})$$

Note that here, iterates are only allowed to vary over the finite subset S of Fourier coefficients and are assumed to be equal to 0 elsewhere. If we assume further that the exponential loss converges to zero, that the gradients $\nabla_{\widehat{\beta}} \mathcal{L}(\widehat{\beta}^t)$ converge in direction, and that the iterates $\widehat{\mathbf{W}}^t$ themselves converge in direction to yield a separator with positive margin, then the limit direction of the parameters $\widehat{\mathbf{W}}^\infty = \lim_{t \rightarrow \infty} \frac{\widehat{\mathbf{W}}^t}{\|\widehat{\mathbf{W}}^t\|_2}$ is a positive scaling of a first order stationary point of the following optimization problem:

$$\min_{\widehat{\mathbf{W}} \in \mathbb{R}^p} \|\widehat{\mathbf{W}}\|_2^2 \quad \text{s.t. } \forall n, y_n \langle \widehat{\mathbf{x}}_n, \mathcal{P}(\widehat{\mathbf{W}}) \rangle_M \geq 1. \quad (\text{A.41})$$

Again letting $\widehat{\mathbf{W}}^\infty = \tau \overline{\widehat{\mathbf{W}}^\infty}$ denote the first order stationary point itself, $\widehat{\mathbf{W}}^\infty_i$ the individual layers (or parameter blocks) comprising $\widehat{\mathbf{W}}^\infty$, and $\widehat{\mathbf{W}}^\infty_i$ the individual layers comprising $\widehat{\mathbf{W}}^\infty$, we again have via the KKT conditions that:

$$\begin{aligned} & \exists \{\alpha_n : \alpha_n \geq 0\}_{n=1}^N \text{ s.t. } \alpha_n = 0 \text{ if } y_n \langle \widehat{\mathbf{x}}_n, \mathcal{P}(\widehat{\mathbf{W}}^\infty) \rangle_M > 1 \\ & \widehat{\mathbf{W}}_i^\infty = \nabla_{\widehat{\mathbf{W}}_i} \mathcal{P}(\widehat{\mathbf{W}}^\infty) \left[\sum_n \alpha_n y_n \widehat{\mathbf{x}}_n \right] = \nabla_{\widehat{\mathbf{W}}_i} \left\langle \mathcal{P}(\widehat{\mathbf{W}}^\infty), \sum_n \alpha_n y_n \widehat{\mathbf{x}}_n \right\rangle_M \end{aligned} \quad (\text{A.42})$$

Equivalently, writing the above in terms of the matrices $\widehat{\mathbf{w}}_\ell^\infty$, and defining $\widehat{\mathbf{z}} = \sum_n \alpha_n y_n \widehat{\mathbf{x}}_n$, we have an equivalence to [Lemma A.10](#).

$$\begin{aligned}
 \nabla_{\underline{w}_\ell} \mathcal{P}(\widehat{\underline{W}^\infty})[\widehat{\underline{z}}] &= \nabla_{\widehat{\underline{w}}_\ell^\infty} \langle \mathcal{P}(\widehat{\underline{W}^\infty}), \widehat{\underline{z}} \rangle_M \\
 &= \nabla_{\widehat{\underline{w}}_\ell^\infty} \langle \widehat{\underline{z}}, \mathcal{P}(\widehat{\underline{W}^\infty}) \rangle_M \\
 &= \nabla_{\widehat{\underline{w}}_\ell^\infty} \langle \widehat{\underline{z}}, \widehat{\underline{w}}_L^\infty \cdots \widehat{\underline{w}}_1^\infty \rangle_M \\
 &= \nabla_{\widehat{\underline{w}}_\ell^\infty} \text{tr}[\widehat{\underline{z}} (\widehat{\underline{w}}_L^\infty \cdots \widehat{\underline{w}}_1^\infty)^\dagger] \\
 &= \nabla_{\widehat{\underline{w}}_\ell^\infty} \text{tr}[\widehat{\underline{w}}_{\ell+1}^{\infty\dagger} \cdots \widehat{\underline{w}}_L^{\infty\dagger} \widehat{\underline{z}} \widehat{\underline{w}}_1^{\infty\dagger} \cdots \widehat{\underline{w}}_\ell^{\infty\dagger}] \\
 &= \nabla_{\widehat{\underline{w}}_\ell^\infty} \langle \widehat{\underline{w}}_{\ell+1}^{\infty\dagger} \cdots \widehat{\underline{w}}_L^{\infty\dagger} \widehat{\underline{z}} \widehat{\underline{w}}_1^{\infty\dagger} \cdots \widehat{\underline{w}}_{\ell-1}^{\infty\dagger}, \widehat{\underline{w}}_\ell^\infty \rangle_M \\
 &= \widehat{\underline{w}}_{\ell+1}^{\infty\dagger} \cdots \widehat{\underline{w}}_L^{\infty\dagger} \widehat{\underline{z}} \widehat{\underline{w}}_1^{\infty\dagger} \cdots \widehat{\underline{w}}_{\ell-1}^{\infty\dagger}
 \end{aligned} \tag{A.43}$$

Using the KKT conditions above and this fact, all of the manipulations demonstrating that a positive scaling of $\sum_n \alpha_n y_n \widehat{\underline{x}_n}$ is a first-order stationary point of the optimization problem below carry over exactly as they do in the previous part. This yields the following formal result:

Theorem A.11. *Consider a classification task with ground-truth linear predictor $\widehat{\beta}$, trained via a real-valued, Fourier-space, band-limited linear G-CNN architecture $\text{NN}(\widehat{\underline{x}}) = \langle \widehat{\underline{x}}, \widehat{\underline{w}}_L \dots \widehat{\underline{w}}_1 \rangle$ with $L \geq 2$ layers under the exponential loss. Then for almost any datasets $\{\underline{x}_i, y_i\}$ separable by β , any bounded sequence of step sizes η_t , and almost all initializations, suppose that:*

- The loss converges to 0
- The gradients with respect to the end-to-end linear predictor $\widehat{\beta}$ converge in direction
- The iterates $\widehat{\underline{w}}_i$ themselves converge in direction to a separator with positive margin

When $L = 2$, we need an additional technical assumption, *Assumption A.7*. Then, the resultant linear predictor $\widehat{\beta}^\infty$ is a positive scaling of a first order stationary point of the optimization problem:

$$\min_{\widehat{\beta}} \left\| \widehat{\beta} \right\|_{2/L}^{(S)} \quad \text{s.t.} \quad \forall n, y_n \langle \widehat{\underline{x}_n}, \widehat{\beta} \rangle_M \geq 1. \tag{A.44}$$

B. Group Fourier Transforms

To aid the reader in understanding the notation and structure behind the group Fourier transform, the following exposition is given for reference and convenience. Here, we introduce important concepts from representation theory and from there, provide explicit constructions the group Fourier transform.

A representation of a group G is a vector space V together with a G -linear map $\rho : G \rightarrow \text{GL}(V)$. Of particular interest is the *regular representation* which we construct as follows. Let G be a group of order n and choose $V = \mathbb{C}^n$. Consider an element $u \in \mathbb{C}[G]$, so $u = a_1 g_1 + \cdots + a_n g_n$ where $a_i \in \mathbb{C}$ and $g_n \in G$, and the associated vector $\mathbf{u} \in \mathbb{C}^n$ such that $\mathbf{u} = [a_1 \ \cdots \ a_n]$.

The action of left-multiplication of u for any $h \in G$ yields $hu = a_1(hg_1) + \cdots + a_n(hg_n)$, which is equivalent to a permutation of the coefficients, so there is a unique matrix $H \in \text{GL}(\mathbb{C}^n)$ such that $H\mathbf{u}$ is equivalent to the associated vector for hu . The G -linear map $L_h : h \mapsto H$ is the (left) *regular representation*.

The direct sum of two G -representations (ρ_1, V_1) and (ρ_2, V_2) can be constructed by

$$(\rho_1 \oplus \rho_2)(g) = \begin{bmatrix} \rho_1(g) & \\ & \rho_2(g) \end{bmatrix} \tag{B.1}$$

The dimension d_ρ of a representation (ρ, V) is defined to be $\dim(V)$. A finite-dimensional representation is *irreducible* if it cannot be written as the direct sum of two nontrivial representations. Denote \widehat{G} to be the set of isomorphism classes of

irreducible subrepresentations of L_u , and let d_{ρ_i} denote the dimension of $\rho_i \in \widehat{G}$. As it turns out, there is an isomorphism

$$L_u \cong \bigoplus_{\rho \in \widehat{G}} \rho^{\oplus d_\rho}$$

Where ρ ranges over one representative from each of the isomorphism classes of \widehat{G} , repeated according to its multiplicity. In general, this decomposition is not uniquely determined as it depends on the choice of representatives. Throughout this paper, we choose representatives such that each ρ is *unitary*, meaning that every $\rho(g)$ is a unitary matrix.

Every function $f : G \rightarrow \mathbb{C}$ can be considered as equivalent to an element $u_f \in \mathbb{C}[G]$ by setting $u_f = f(g_1)g_1 + \dots + f(g_n)g_n$. And as we have already seen, every $u \in \mathbb{C}[G]$ can naturally be considered a subrepresentation of L_u . Then the *Fourier transform* of f at a representation ρ , denoted $\widehat{f}(\rho) \in \mathrm{GL}(d_\rho, \mathbb{C})$ where $\widehat{f}(\rho) = \sum_{u \in G} f(u)\rho(u)$, can be considered as a projection of $L_u(u_f)$ onto the orthogonal subspace described by ρ . Throughout the paper we use slightly different notations and characterizations of the Fourier transform depending on the context, but all share this projection as the fundamental operation.

Recalling [Equation B.1](#), there is a representation isomorphic to L_u , which we will suggestively call \mathcal{F}_M , that block-diagonalizes the image of L_u into orthogonal subspaces of unitary irreducible representations (each $\rho \in \widehat{G}$ repeated with multiplicity d_ρ):

$$\mathcal{F}_M(u) = \begin{bmatrix} \rho_1(u) & & \\ & \ddots & \\ & & \rho_j(g) \end{bmatrix} \in \mathrm{GL}(\mathbb{C}^n) \quad (\text{B.2})$$

For the last piece of the puzzle, extend the domain of \mathcal{F}_M to all functions $f : G \rightarrow \mathbb{C}$ (by considering f as an element of $\mathbb{C}[G]$). Then we obtain

$$\mathcal{F}_M f = \begin{bmatrix} \widehat{f}(\rho_1) & & \\ & \ddots & \\ & & \widehat{f}(\rho_j) \end{bmatrix} \in \mathrm{GL}(\mathbb{C}^n) \quad (\text{B.3})$$

Which we call the *matrix Fourier transform* of f .

We also make use of the *Fourier basis matrix* \mathcal{F} , which depends only on the group G and not the function f . To construct it, we first need the operation *Flatten* which vertically stacks the columns of a matrix. Then define the transform

$$\phi(f) = \begin{bmatrix} \text{Flatten}(\widehat{f}(\rho_1)) \\ \vdots \\ \text{Flatten}(\widehat{f}(\rho_j)) \end{bmatrix} \quad (\text{B.4})$$

Letting $e_i : G \rightarrow \mathbb{C}$ be the indicator function $e_i(g_j) = \mathbb{1}_{i=j}$ we can describe the unitary Fourier basis matrix \mathcal{F} for a group G as a row-scaling of

$$\mathcal{F} \propto [\phi(e_1) \quad \phi(e_2) \quad \cdots \quad \phi(e_n)] \quad (\text{B.5})$$

In other words, given a column vectorization \mathbf{f} of a function f such that $\mathbf{f}_i = f(g_i)$, then \mathcal{F} is the matrix such that the ‘unflattening’ of $\mathcal{F}\mathbf{f}$ is equal to $\mathcal{F}_M f$ up to the row-scaling constants. Thus we can treat the group Fourier transform either as an abstract isomorphism or as a concrete matrix-vector multiplication, depending on the application.

The matrix \mathcal{F} can be explicitly constructed as described in [Definition 4.1](#). Denoting $\mathbf{e}_{[\rho, i, j]}$ as the column-major vectorized basis for element ρ_{ij} in the group Fourier transform, then we can form the matrix

$$\mathcal{F} = \sum_{u \in G} \sum_{\rho \in \widehat{G}} \frac{\sqrt{d_\rho}}{\sqrt{|G|}} \sum_{i, j=1}^{d_\rho} \rho(u)_{ij} \mathbf{e}_{[\rho, i, j]} \mathbf{e}_u^T. \quad (\text{B.6})$$

For visualization, consider the dihedral group $D_6 = \{1, r, r^2, a, ar, ar^2\}$, which has three irreducible representations ρ_1, ρ_2, ρ_3 (up to isomorphism) of dimensions 1, 1, and 2 respectively, and let $f : D_6 \rightarrow \mathbb{C}$. Using colors instead of values at first to avoid numerical clutter:

$$\widehat{f}(\rho_1) = \begin{bmatrix} \text{Blue} \end{bmatrix} \quad \widehat{f}(\rho_2) = \begin{bmatrix} \text{Orange} \end{bmatrix} \quad \widehat{f}(\rho_3) = \begin{bmatrix} \text{Green} \\ \text{Green} \end{bmatrix}$$

Since $L_{D_6} \cong \rho_1 \oplus \rho_2 \oplus \rho_3^2$, we can get something like

$$\mathcal{F}_M f = \begin{bmatrix} \text{Blue} & & & & & \\ & \text{Orange} & & & & \\ & & \text{Green} & & & \\ & & & \text{Green} & & \end{bmatrix} \in \mathrm{GL}(\mathbb{C}^n)$$

Whereas for the unitary Fourier basis matrix we have the form

$$\mathcal{F} \propto \begin{bmatrix} 1 & r & r^2 & a & ar & ar^2 \\ \rho_1 & \text{Blue} & \text{Blue} & \text{Blue} & \text{Blue} & \text{Blue} \\ \rho_2 & \text{Orange} & \text{Orange} & \text{Orange} & \text{Orange} & \text{Orange} \\ [\rho_3]_{11} & \text{Green} & \text{Green} & \text{Green} & \text{Green} & \text{Green} \\ [\rho_3]_{21} & \text{Green} & \text{Green} & \text{Green} & \text{Green} & \text{Green} \\ [\rho_3]_{12} & \text{Green} & \text{Green} & \text{Green} & \text{Green} & \text{Green} \\ [\rho_3]_{22} & \text{Green} & \text{Green} & \text{Green} & \text{Green} & \text{Green} \end{bmatrix}$$

Note that we do not yet include the row-scaling constants. Now explicitly, choosing ρ_1 the trivial irrep, ρ_2 the sign irrep, and ρ_3 the representation sending

$$\rho_3(a) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \quad \rho_3(r) = \begin{bmatrix} \omega & 0 \\ 0 & \omega^2 \end{bmatrix} \quad (\text{B.7})$$

Where $\omega = e^{2\pi i/3}$, then the matrix \mathcal{F} is exactly

$$\mathcal{F} = \frac{1}{\sqrt{6}} \begin{bmatrix} 1 & r & r^2 & a & ar & ar^2 \\ \rho_1 & \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} & & & & \\ \rho_2 & \begin{bmatrix} 1 & 1 & 1 & -1 & -1 & -1 \end{bmatrix} & & & & \\ [\rho_3]_{11} & \begin{bmatrix} \sqrt{2} & \sqrt{2}\omega & \sqrt{2}\omega^2 & 0 & 0 & 0 \end{bmatrix} & & & & \\ [\rho_3]_{21} & \begin{bmatrix} 0 & 0 & 0 & \sqrt{2} & \sqrt{2}\omega & \sqrt{2}\omega^2 \end{bmatrix} & & & & \\ [\rho_3]_{12} & \begin{bmatrix} 0 & 0 & 0 & \sqrt{2}\omega^2 & \sqrt{2}\omega & \sqrt{2} \end{bmatrix} & & & & \\ [\rho_3]_{22} & \begin{bmatrix} \sqrt{2} & \sqrt{2}\omega^2 & \sqrt{2}\omega & 0 & 0 & 0 \end{bmatrix} & & & & \end{bmatrix} \quad (\text{B.8})$$

Note that the above is only one possible way of writing \mathcal{F} since the 2-dimensional irreducible representation of D_6 is unique only up to conjugation by a unitary matrix.

C. Uncertainty principles for groups

In mathematics, uncertainty principles categorize trade-offs of the “amount of information” stored in a function between canonically conjugate regimes, *e.g.*, position (real regime) and momentum (Fourier regime) of a physical particle. More

generically, uncertainty principles show that a function and its Fourier transform cannot both be very localized or concentrated. In the context of group theory, uncertainty principles specifically show that sparse support in either the real or Fourier regime of a group necessarily implies dense support in the conjugate regime (Wigderson and Wigderson, 2021). Such results are directly relevant when interpreting implicit regularization of linear G-CNNs which bias gradient descent towards sparse solution in the Fourier basis of the group. In this section, we formally state and summarize these group theoretic uncertainty principles to provide intuition into the properties of functions which linear group convolutional networks are likely to learn.

Abelian groups For abelian groups, the fundamental uncertainty principle details a trade-off between the norms of a function in its real and Fourier bases.

Theorem C.1 (Generalization of Donoho-Stark Theorem (Wigderson and Wigderson, 2021)). *Given a finite abelian group G , let \widehat{G} be the set of irreducible representations of G and $f : G \rightarrow \mathbb{C}$ be a function mapping group elements to complex numbers. Let \mathbf{f} be the vectorized function and \mathcal{F} be the unitary group Fourier transform (see Definition 4.1), then*

$$\frac{\|\mathbf{f}\|_1}{\|\mathbf{f}\|_\infty} \frac{\|\mathcal{F}\mathbf{f}\|_1}{\|\mathcal{F}\mathbf{f}\|_\infty} \geq |G| \quad (\text{C.1})$$

Remark C.2. Since the size of the support of a vector is bounded by $|\text{supp}(\mathbf{v})| \geq \frac{\|\mathbf{v}\|_1}{\|\mathbf{v}\|_\infty}$, this directly implies that $|\text{supp}(\mathbf{f})| |\text{supp}(\mathcal{F}\mathbf{f})| \geq |G|$ recovering the Donoho-Stark theorem (Donoho and Stark, 1989; Matusiak et al., 2004).

Non-abelian groups Since non-abelian groups have matrix valued irreducible representations, uncertainty theorems must account for norms and notions of support in the context of matrices. Here, we will provide two different uncertainty theorems for the non-abelian setting – one via the rank of irreducible representations and another via the Schatten norm of irreducible representations. Uncertainty relations for non-abelian groups make use of the matrix group Fourier transform detailed in Definition 4.1.

Theorem C.3 (Meshulam uncertainty theorem (Meshulam, 1992)). *Given a finite non-abelian group G , let \widehat{G} be the set of irreducible representations of G and $f : G \rightarrow \mathbb{C}$ be a function mapping group elements to complex numbers. Let \mathbf{f} be the vectorized function and \mathcal{F}_M be the matrix group Fourier transform (see Definition 4.1), then*

$$|\text{supp}(\mathbf{f})| \text{rank}(\mathcal{F}_M \mathbf{f}) = |\text{supp}(\mathbf{f})| \left[\sum_{\rho \in \widehat{G}} d_\rho \text{rank}(\widehat{f}(\rho)) \right] \geq |G| \quad (\text{C.2})$$

The above theorem shows that the rank of the matrix Fourier transformed function is the proper notion of support in the uncertainty theorem for a non-abelian group. A stronger uncertainty principle which is a more direct corollary to Theorem C.1 can be obtained via the Schatten norms of the irreducible representations as shown below.

Theorem C.4 (Kuperberg uncertainty theorem (Wigderson and Wigderson, 2021)). *Given a finite non-abelian group G , let \widehat{G} be the set of irreducible representations of G and $f : G \rightarrow \mathbb{C}$ be a function mapping group elements to complex numbers. Let \mathbf{f} be the vectorized function and \mathcal{F}_M be the matrix group Fourier transform (see Definition 4.1), then*

$$\frac{\|\mathbf{f}\|_1}{\|\mathbf{f}\|_\infty} \frac{\|\mathcal{F}_M \mathbf{f}\|_1^{(S)}}{\|\mathcal{F}_M \mathbf{f}\|_\infty^{(S)}} = \frac{\|\mathbf{f}\|_1}{\|\mathbf{f}\|_\infty} \frac{\sum_{\rho \in \widehat{G}} d_\rho \|\widehat{f}(\rho)\|_1^{(S)}}{\max_{\rho \in \widehat{G}} \|\widehat{f}(\rho)\|_\infty^{(S)}} \geq |G|. \quad (\text{C.3})$$

D. Visualizing the implicit bias

Implicit biases induced by the G-CNN architectures studied here can be readily observed by analyzing coefficients of the linearized transformation in the Real or Fourier regimes. Here, we visualize the linearized outputs a 3-layer linear G-CNN over the Dihedral group D_{60} which has 4 scalar irreps and 14 irreps of dimension 2 (hence 2×2 matrices). Figure 5 shows these linearized coefficients of the G-CNN, CNN, and FC at initialization and training.

As evident in Figure 5, the learned coefficients of the G-CNN are sparse in the Fourier regime of the group. This sparsity pattern appears over blocks of irreps of length four, corresponding to coefficients of the 2×2 irreps of D_{60} . Furthermore, the values of the coefficients within a block are roughly constant, highlighting the bias towards low rank irreps. The relative denseness of coefficients of the trained G-CNN in the real regime is inherent due to the uncertainty principles of group

functions. Unlike the G-CNN, the fully connected network (FC) appears to have no bias towards sparseness in its coefficients in either the real or Fourier regime. On a related note, the cyclic group of CNNs share some of the same irreps as those of the G-CNN studied here. This may be one explanation for the partial sparsity patterns observed in the coefficients of the CNN in the Fourier regime.

E. Computational details

As described in Section 6, for all models we use three-layer networks over $\mathbb{R}^{|G|}$ unless otherwise stated and binary classification tasks trained via standard gradient descent with exponential loss. We often train networks on isotropic Gaussian data points which are random vectors whose entries are drawn i.i.d. from a standard Normal distribution. For the linear networks on simulated data (with a fully connected output layer) we use the groups D_8 and $(C_5 \times C_5) \rtimes Q_8$. For the linear and nonlinear networks on MNIST, we use $(C_{28} \times C_{28}) \times D_8$. For the networks with ReLU activations and a linear layer (instead of pooling) we use the groups D_8 and D_{60} . For the experiments on MNIST with non-linear networks, we have used the e2cnn package (Weiler and Cesa, 2019). The weights are initialized with the standard uniform initialization. We choose an appropriate learning rate for each task depending on the dimension and magnitude of the values. Due to the choice of the exponential loss as our loss function, we sometimes periodically increased the learning rate since gradients decay over time to speed up convergence. Since each problem is overparameterized the loss will almost surely converge to zero, so we choose enough training epochs to achieve satisfactory convergence. For each plot, we report 95% confidence intervals over 3 to 50 runs, depending on the classification task. The computational resources used are modest - commodity hardware should suffice to fully reproduce our results. To replicate experiments, please visit our code repository.¹⁰

E.1. Additional experiments - linear

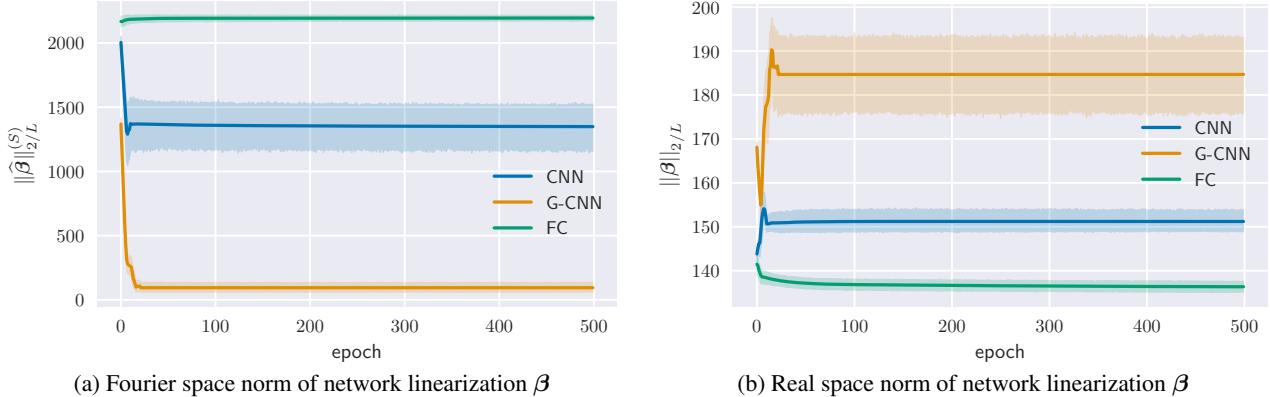


Figure 7: Norms of the linearizations of three different linear architectures for the non-abelian group $G = (C_5 \times C_5) \rtimes Q_8$ trained on a binary classification task with 10 isotropic Gaussian data points.

We include linear architecture experiments on two additional groups. First, we include a G-CNN with the group $(C_5 \times C_5) \rtimes Q_8$ (Figure 7), which is a non-abelian group that has irreducible representations of up to dimension 8 and displays implicit regularization over a more elaborate group structure. Inputs are vectors with elements drawn i.i.d. from the standard Normal distribution.

E.2. Additional experiments - nonlinear

We evaluate the implicit bias of an invariant ReLU G-CNN (with final pooling layer) with respect to translations, rotations, and flips on MNIST digits in Figure 8.

¹⁰Our code is available here: <https://github.com/kristian-georgiev/implicit-bias-of-linear-equivariant-networks>

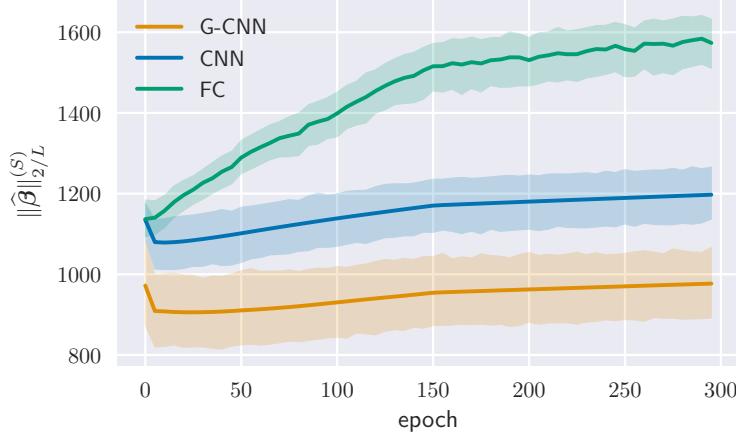


Figure 8: Group Fourier norm for the non-abelian group $(C_{28} \times C_{28}) \times D_8$ for an additional *nonlinear* architecture with ReLU activations shows that a nonlinear G-CNN seems to implicitly regularize locally. The figure tracks the mean norm of the per-sample local linearizations of each network. The network uses a pooling layer after the convolutional layers to maintain invariance. We evaluate a binary classification task using the MNIST digits 0 and 5.

E.3. Spherical CNN Experimental Details

For the spherical CNN experiments, our architecture is drawn directly from Cohen et al. (2018). Given an input image defined on a sphere, we apply to it a fixed S_2 -convolution, followed by two learnable $SO(3)$ -convolutions and a learnable fully-connected final layer, with ReLUs in-between. As in Cohen et al. (2018), convolutions are done in Fourier space. We bandlimit the signals up to $\ell = 5$, and then $\ell = 3$. The first $SO(3)$ -convolution has 20 channels, and the second one has 100. The use of varying bandlimits is adapted directly from Cohen et al. (2018), but departs from our theory in the linear case of infinite groups. We compute all Schatten norms with respect to the bandwidth $\ell = 5$. Since we use multiple (20) channels for the first convolution, we compute the linearization by computing the linearization for each channel separately, and then reporting the average Schatten norms over all channels. We note that this also departs from our theory towards a more realistic scenario, and we only provide these empirical observations for multi-channel networks.

The architecture is trained via stochastic gradient descent on the *cross-entropy loss*, as the exponential loss did not result in learning the training data. Although this further departs from the theory, other implicit regularization works have studied the cross-entropy and exponential losses simultaneously, suggesting that the same results may carry over. Indeed, it is promising that the Schatten norm regularization occurs with a different loss function in the nonlinear case. For comparison architectures, we use a fully connected architecture and a 2D *CNN* (applied to each discretization index of $SO(3)$). All architectures have as input a fixed, shared S_2 convolution and ReLU layer, which we conceptualize as a fixed, non-linear feature embedding. The reason for this is as follows: to apply the representation theory of $SO(3)$, we need inputs and linearization which are defined on $SO(3)$, not the homogeneous space S_2 (the sphere). As the output of an S_2 convolution is defined on $SO(3)$, the fixed feature embedding serves this purpose.

E.4. Omitted real-space plots

E.5. Loss plots

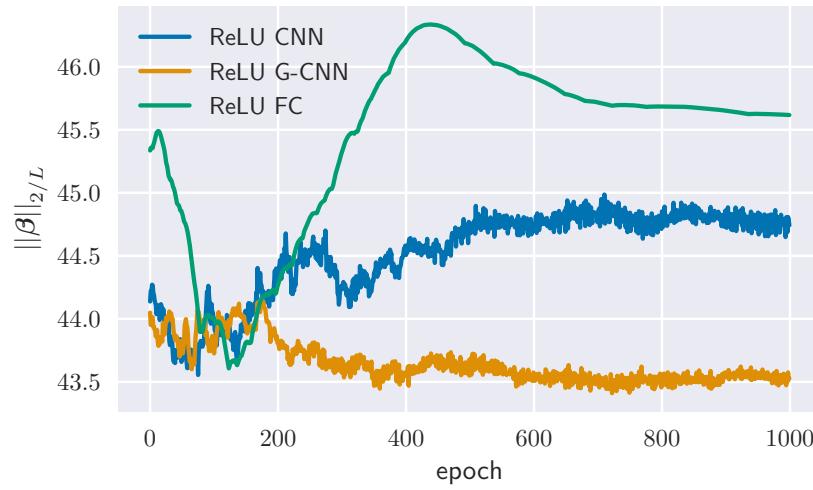


Figure 9: Real-space norm for D_{60} with ReLU networks, 10 Gaussian training points. See [Figure 6a](#) for comparable plot of norms in Fourier space.

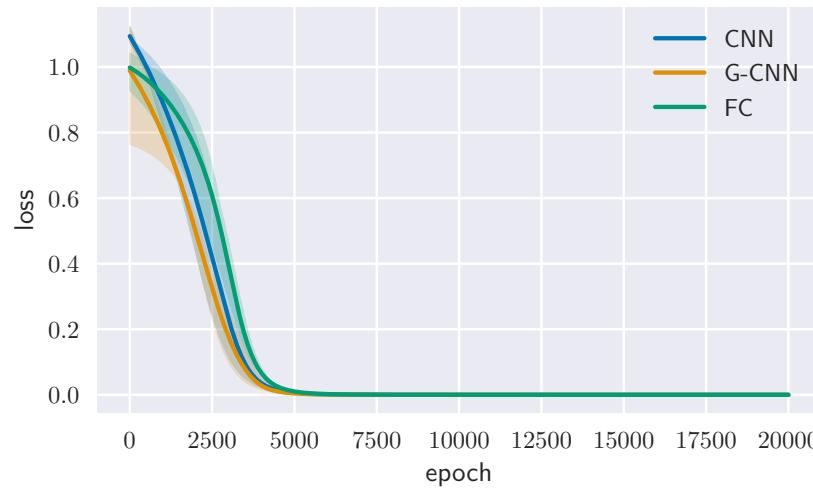


Figure 10: Loss trajectory for the setting of D_8 (see [Figure 1](#)). Networks are trained on 2 Fourier i.i.d. data points.

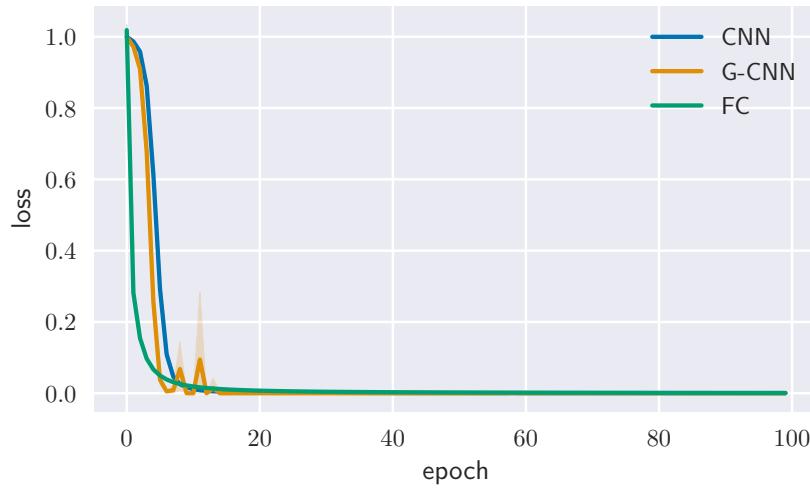


Figure 11: Loss trajectory for the setting of $C_{10} \times C_{10} \times C_2$ (see ??). Networks are trained on 6 Gaussian i.i.d. data points.

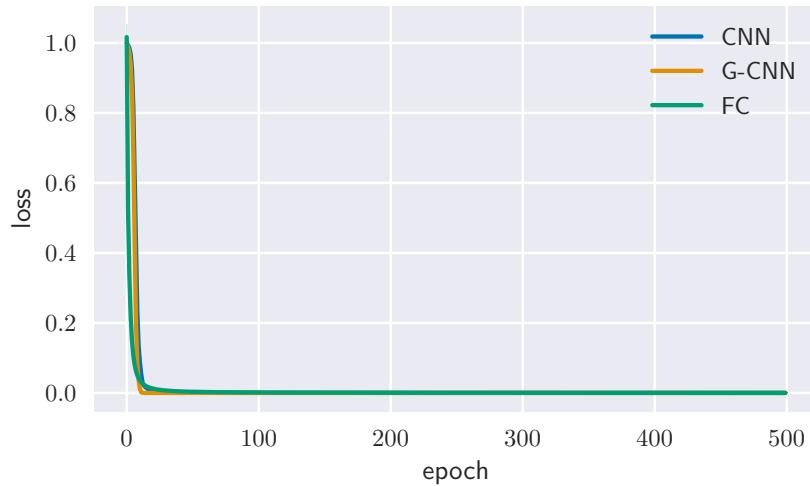


Figure 12: Loss trajectory for the setting of $(C_5 \times C_5) \rtimes Q_8$ (see Figure 7). Networks are trained on 10 Gaussian i.i.d. data points.

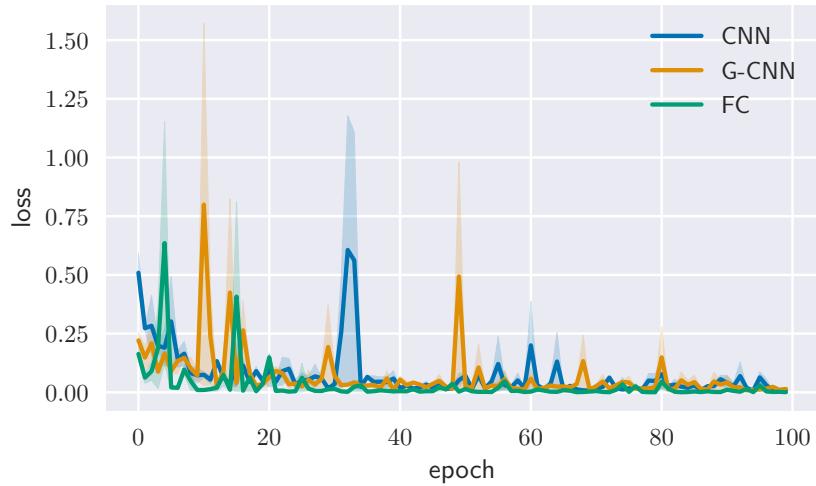


Figure 13: Loss trajectory for the setting of $(C_{28} \times C_{28}) \times D_8$ (see Figure 4). Networks are trained on the digits 1 and 5 from MNIST.

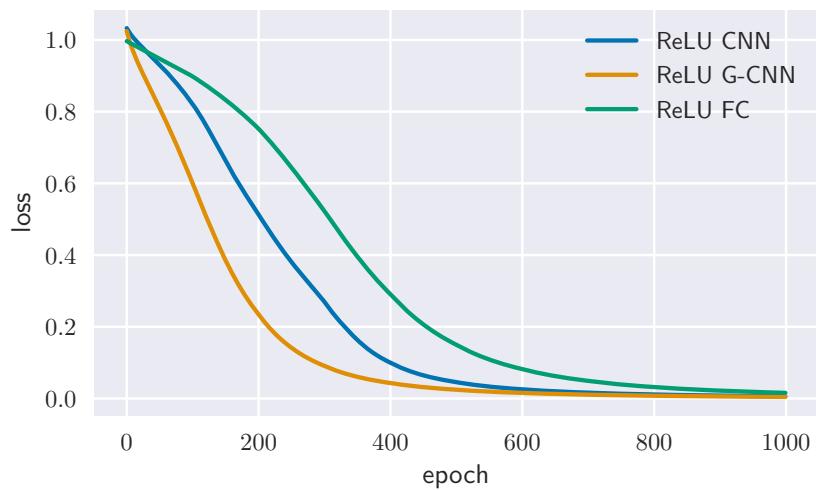


Figure 14: Loss trajectory for the setting of D_{60} (see Figure 6a). Networks are nonlinear with ReLU activations and trained on 10 distinct frequencies as data points.

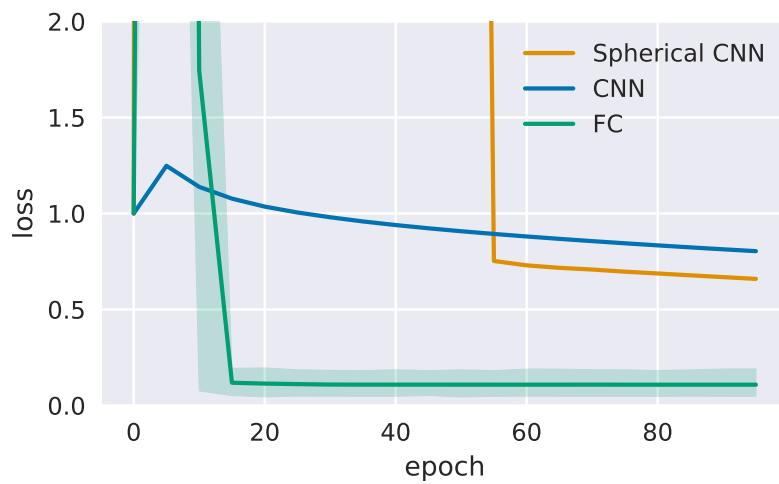


Figure 15: Loss trajectory for the setting of bandlimited $SO(3)$. Networks are nonlinear with ReLU activations and trained on a subset of spherical MNIST (digits 0 and 5).