SIGN AND BASIS INVARIANT NETWORKS FOR SPECTRAL GRAPH REPRESENTATION LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Many machine learning tasks involve processing eigenvectors derived from data. Especially valuable are Laplacian eigenvectors, which capture useful structural information about graphs and other geometric objects. However, ambiguities arise when computing eigenvectors: for each eigenvector v, the sign flipped $-v$ is also an eigenvector. More generally, higher dimensional eigenspaces contain infinitely many choices of eigenvector bases. In this work we introduce SignNet and BasisNet — new neural architectures that are invariant to all requisite symmetries and hence process collections of eigenspaces in a principled manner. Our networks are universal, i.e., they can approximate any continuous function of eigenvectors with the proper invariances. They are also theoretically strong for graph representation learning — they can provably approximate any spectral graph convolution, spectral invariants that go beyond message passing neural networks, and other graph positional encodings. Experiments show the strength of our networks for learning spectral graph filters and learning graph positional encodings.

1 INTRODUCTION

Numerous machine learning models process eigenvectors, which arise in various scenarios including principal component analysis, matrix factorizations, and operators associated to graphs or manifolds. An important example is the use of graph Laplacian eigenvectors to encode information about the structure of a graph (Belkin & Niyogi, 2003). Positional encodings that involve Laplacian eigenvectors have recently been used to generalize Transformers to graphs (Kreuzer et al., 2021; Dwivedi & Bresson, 2021), and to improve the expressive power and empirical performance of graph neural networks (GNNs) (Dwivedi et al., 2022). Furthermore, these eigenvectors are crucial for defining spectral operations on graphs for graph signal processing and spectral GNNs (Ortega et al., 2018).

However, there are nontrivial symmetries that should be accounted for when processing eigenvectors. If v is a unit-norm eigenvector, then so is $-v$, with the same eigenvalue. More generally, if an eigenvalue has higher multiplicity, then there are infinitely many unit-norm eigenvectors that can be chosen. Indeed, a full set of orthonormal eigenvectors is only defined up to a change of basis in each eigenspace. In the case of sign invariance, for any k eigenvectors there are 2^k possible choices of sign. Accordingly, prior works randomly flip eigenvector signs during training in order to approximately learn sign invariance (Kreuzer et al., 2021; Dwivedi et al., 2020). However, learning all 2^k invariances is challenging and limits the effectiveness of Laplacian eigenvectors for encoding positional information. Sign invariance is a special case of basis invariance when all eigenvalues are distinct, but higher dimensional basis invariance is even more difficult to deal with, and we show that these higher dimensional eigenspaces are abundant in real datasets (see Appendix E.2).

This work solves the sign and basis ambiguity problems by developing new neural networks, SignNet and BasisNet, with sign invariance and basis invariance built-in. Our networks are universal and can approximate any continuous function of eigenvectors with the proper invariances. Moreover, our networks are theoretically powerful for graph representation learning — they can approximate spectral graph convolutions and compute graph properties like subgraph counts that message passing neural networks cannot. Finally, Laplacian eigenvectors with SignNet and BasisNet can approximate previously proposed graph positional encodings, (Li et al., 2020; Dwivedi et al., 2022; Mialon et al., 2021; Feldman et al., 2022). We experimentally demonstrate the strength of SignNet and BasisNet for learning spectral graph convolutions and graph positional encodings.

2 SIGN AND BASIS INVARIANT NETWORKS

For an $n \times n$ symmetric matrix, let $\lambda_1 \leq \ldots \leq \lambda_n$ be eigenvalues and $v_1, \ldots, v_n$ the corresponding eigenvectors, which we may assume form an orthonormal basis. For instance, we could consider the normalized graph Laplacian $L = I - D^{-1/2}AD^{-1/2}$, where $A \in \mathbb{R}^{n \times n}$ is the adjacency matrix and D is the diagonal degree matrix of some undirected graph. Our goal is to parameterize a class of models $f(v_1, \ldots, v_k)$ taking k eigenvectors as input that respects the eigenvector symmetries. For instance, $-v_i$ is also an eigenvector for any v_i, so a well-defined function f should be *sign invariant*:

$$f(v_1, \ldots, v_k) = f(s_1 v_1, \ldots, s_k v_k) \tag{1}$$

for all sign choices $s_i \in \{-1, 1\}$. That is, we want f to be invariant to the product group $\{-1, 1\}^k$. This captures all eigenvector symmetries if the λ_i are distinct. However, if the eigenvalues have higher multiplicity, then there are further symmetries to consider. Let $V_1, \ldots, V_l$ be bases of eigenspaces — i.e. $V_i = \begin{bmatrix} v_{i_1} | & \ldots & | v_{i_{d_i}} \end{bmatrix} \in \mathbb{R}^{n \times d_i}$ has orthonormal columns and spans the eigenspace associated with the shared eigenvalue $\mu_i = \lambda_{i_1} = \ldots = \lambda_{i_{d_i}}$. Any other orthonormal basis that spans the eigenspace is of the form $V_i Q$ for some orthogonal $Q \in O(d_i) \subseteq \mathbb{R}^{d_i \times d_i}$ (see Appendix B.2). Thus, a function f that is invariant to changes of basis in each eigenspace satisfies

$$f(V_1, \ldots, V_l) = f(V_1 Q_1, \ldots, V_l Q_l), \qquad Q_i \in O(d_i). \tag{2}$$

In other words, f is invariant to the product group $O(d_1) \times \ldots \times O(d_l)$. The l and d_i may vary between matrices, so we show below that our architectures can handle this. Note that $O(1) = \{-1, 1\}$, so sign invariance is a special case of this basis invariance when all eigenvalues are distinct.

We often further desire such a function f to be invariant or equivariant to permutation along the entries (or rows) of each vector, as is natural for graph data. Thus, we often require $f(PV_1, \ldots, PV_l) = Pf(V_1, \ldots, V_l)$ for any permutation matrix $P \in \mathbb{R}^{n \times n}$. Figure 1 illustrates the full setup.

2.1 WARMUP: NEURAL NETWORKS ON ONE EIGENSPACE

We begin by designing sign or basis invariant neural networks on a single eigenvector or eigenspace. For one subspace, a function $h : \mathbb{R}^n \to \mathbb{R}^s$ is sign invariant if and only if $h(v) = \phi(v) + \phi(-v)$ for some continuous $\phi : \mathbb{R}^n \to \mathbb{R}^s$, and h is permutation equivariant if and only if ϕ is (proof in Appendix C). We parameterize ϕ by a neural network — any architecture choice ensures sign invariance, whilst permutation equivariance can be achieved using DeepSets or most GNNs.

Switching focus to basis invariance for a single d dimensional subspace, our aim is to design maps $f : \mathbb{R}^{n \times d} \to \mathbb{R}^n$ that are invariant to right multiplication by $Q \in O(d)$, and equivariant to permutations along the row axis. We make use of the mapping $V \mapsto VV^\top$, which is $O(d)$ invariant. Mapping $V \mapsto VV^\top$ does not lose any information if we treat V as equivalent to VQ for any $Q \in O(d)$. This is justified by the classical first fundamental theorem of $O(d)$ (Kraft & Procesi, 1996), which has recently been applied in machine learning by Villar et al. (2021). The resulting $VV^\top \in \mathbb{R}^{n \times n}$ is a matrix for which permutations act on the rows and columns: we want a function $\varphi : \mathbb{R}^{n \times n} \to \mathbb{R}^n$ on $VV^\top$ to be permutation equivariant, so $\varphi(PVV^\top P^\top) = P\varphi(VV^\top)$. To achieve this permutation equivariance from matrices to vectors, we use an invariant graph network (IGN) (Maron et al., 2018) — a neural network mapping to and from tensors of arbitrary order $\mathbb{R}^{n^{d_1}} \to \mathbb{R}^{n^{d_2}}$ that has the desired permutation equivariance. So we parameterize a family with the requisite invariance and equivariance as follows:

$$f(V) = \mathrm{IGN}(VV^\top). \tag{3}$$

Proposition 4 shows that this architecture is universal, but this requires high order tensors to be used for IGN. The next section discusses a more efficient architecture with restricted tensor dimensions.

2.2 NEURAL NETWORKS ON MULTIPLE EIGENSPACES

For our fully general sign invariant and basis invariant networks, eigenspaces are independently processed using the models introduced in Section 2.1, then the processed eigenspaces are aggregated into a single embedding. This approach is motivated by the general decomposition theorem for product spaces in Section 3, which says that we may decompose along these single subspaces.

SignNet. The *unconstrained SignNet* $f : \mathbb{R}^{n\times k} \to \mathbb{R}^s$ on eigenvectors $v_1, \ldots, v_k$ takes the form:

$$f(v_1, \ldots, v_k) = \rho\left([\phi(v_i) + \phi(-v_i)]_{i=1}^k\right), \tag{4}$$

where ϕ and ρ are unrestricted neural networks, and $[\cdot]_i$ denotes concatenation. If we desire f to be a permutation equivariant function that outputs vectors in $\mathbb{R}^{n\times s}$, then we restrict ϕ and ρ to be permutation equivariant networks that map vectors to vectors such as DeepSets (Zaheer et al., 2017), Transformers (Vaswani et al., 2017), or GNNs. We call the permutation equivariant version *SignNet*.

BasisNet. We take a similar approach for basis invariance. Let $V_i \in \mathbb{R}^{n\times d_i}$ be an orthonormal basis of a d_i dimensional eigenspace. Then we parameterize our *unconstrained BasisNet* f by

$$f(V_1, \ldots, V_l) = \rho\left([\phi_{d_i}(V_i V_i^\top)]_{i=1}^l\right), \tag{5}$$

where each ϕ_{d_i} is shared amongst all subspaces of the same dimension d_i, and l is the number of eigenspaces. As l differs between different graphs, we may use zero-padding or a sequence model like a Transformer to parameterize ρ. Again, ϕ_{d_i} and ρ are generally unrestricted neural networks. If we want permutation equivariance, then we let ρ be permutation equivariant and $\phi_{d_i} = \mathrm{IGN}_{d_i} : \mathbb{R}^{n^2} \to \mathbb{R}^n$ be IGNs from matrices to vectors. For efficiency, we will only use matrices and vectors in the IGNs (that is, no tensors in $\mathbb{R}^{n^p}$ for $p > 2$), i.e., we use 2-IGN. Our resulting *BasisNet* is

$$f(V_1, \ldots, V_l) = \rho\left([\mathrm{IGN}_{d_i}(V_i V_i^\top)]_{i=1}^l\right). \tag{6}$$

3 Universality and Theoretical Expressive Power

Without permutation equivariance constraints, the universality of both architectures introduced in Section 2.2 follow as corollaries of the following general decomposition result.

Theorem 1 (Decomposition Theorem)**.** *Let $\mathcal{X}_1, \ldots, \mathcal{X}_k$ be topological spaces, and let G_i be a group acting on $\mathcal{X}_i$ for each i. We assume mild topological conditions on $\mathcal{X}_i$ and G_i hold. For any continuous $f : \mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to \mathbb{R}^s$ that is invariant to the action of $G = G_1 \times \ldots \times G_k$, there exists continuous ϕ_i and a continuous ρ such that*

$$f(v_1, \ldots, v_k) = \rho(\phi_1(v_1), \ldots, \phi_k(v_k)). \tag{7}$$

Furthermore: (1) each ϕ_i can be taken to be invariant to G_i, (2) the domain $\mathcal{Z}$ of ρ is compact if each $\mathcal{X}_i$ is compact, (3) if $\mathcal{X}_i = \mathcal{X}_j$ and $G_i = G_j$, then ϕ_i can be taken to be equal to ϕ_j.

This result says that when a product of groups G acts on a product of spaces $\mathcal{X}$, for invariance to the product group G it suffices to individually process each smaller group G_i on $\mathcal{X}_i$ and then aggregate the results. Along with the proof of Theorem 1, the mild topological assumptions are explained in Appendix C.1. The assumptions hold for sign invariance and basis invariance.

Corollary 1. *Unconstrained SignNet can represent any sign invariant function and unconstrained BasisNet can represent any basis invariant function.*

Table 1: Test MAE (↓) results on the ZINC dataset with 500k parameter budget. All models use edge features. Numbers are the mean and standard deviation over 4 runs, each with different seeds.

Positional encoding	GatedGCN	Sparse Transformer	PNA	GINE
No PE	$0.252_{\pm 0.007}$	$0.283_{\pm 0.030}$	$0.128_{\pm 0.000}$	$0.170_{\pm 0.002}$
LapPE (flip)	$0.198_{\pm 0.011}$	$0.223_{\pm 0.007}$	$0.139_{\pm 0.000}$	$0.178_{\pm 0.004}$
SignNet	$\mathbf{0.121}_{\pm \mathbf{0.005}}$	$\mathbf{0.115}_{\pm \mathbf{0.008}}$	$\mathbf{0.105}_{\pm \mathbf{0.007}}$	$\mathbf{0.147}_{\pm \mathbf{0.005}}$

Now, we prove that our SignNets and BasisNets can compute useful basis invariant and permutation equivariant functions on Laplacian eigenvectors for graph representation learning.

SignNets and BasisNets Generalize Spectral Graph Convolution For node features $X \in \mathbb{R}^{n\times q}$ and an eigendecomposition $V\Lambda V^\top$, a spectral graph convolution takes the form $f(V, \Lambda, X) = \sum_{i=1}^n \theta_i v_i v_i^\top X$, for parameters θ_i, that may optionally be continuous functions $h(\lambda_i) = \theta_i$ of the eigenvalues (Bruna et al., 2014; Defferrard et al., 2016). This includes important functions like heat kernels and generalized PageRanks on graphs (Li et al., 2019). Spectral graph convolutions are permutation equivariant and sign invariant, and if $\theta_i = h(\lambda_i)$, they are invariant to change of bases in each eigenspace. Our SignNet and BasisNet universally approximate spectral graph convolutions.

Table 2: Sum of squared errors for spectral graph convolution regression (with no test set). Lower is better. Numbers are mean and standard deviation over 50 images from He et al. (2021).

	Low-pass	High-pass	Band-pass	Band-rejection	Comb
ARMA	$.053_{\pm.029}$	$.042_{\pm.024}$	$.107_{\pm.039}$	$.148_{\pm.089}$	$.202_{\pm.116}$
ChebNet	$.003_{\pm.002}$	$\mathbf{.001}_{\pm.001}$	$.005_{\pm.003}$	$.009_{\pm.006}$	$.022_{\pm.016}$
BernNet	$\mathbf{.001}_{\pm.002}$	$\mathbf{.001}_{\pm.001}$	$\mathbf{.000}_{\pm.000}$	$.048_{\pm.042}$	$.027_{\pm.019}$
Transformer	$3.662_{\pm1.97}$	$3.715_{\pm1.98}$	$1.531_{\pm1.30}$	$1.506_{\pm1.29}$	$3.178_{\pm1.93}$
Transformer Eig Flip	$4.454_{\pm2.32}$	$4.425_{\pm2.38}$	$1.651_{\pm1.53}$	$2.567_{\pm1.73}$	$3.720_{\pm1.94}$
Transformer Eig Abs	$2.727_{\pm1.40}$	$3.172_{\pm1.61}$	$1.264_{\pm.788}$	$1.445_{\pm.943}$	$2.607_{\pm1.32}$
DeepSets SignNet	$.004_{\pm.013}$	$.086_{\pm.405}$	$.021_{\pm.115}$	$.008_{\pm.037}$	$\mathbf{.003}_{\pm.016}$
Transformer SignNet	$.003_{\pm.016}$	$.004_{\pm.025}$	$.001_{\pm.004}$	$.006_{\pm.023}$	$.093_{\pm.641}$
DeepSets BasisNet	$.009_{\pm.018}$	$.003_{\pm.015}$	$.008_{\pm.030}$	$\mathbf{.004}_{\pm.011}$	$.015_{\pm.060}$
Transformer BasisNet	$.079_{\pm.471}$	$.014_{\pm.038}$	$.005_{\pm.018}$	$.006_{\pm.016}$	$.014_{\pm.051}$

Proposition 1. *SignNet universally approximates all spectral graph convolutions. BasisNet universally approximates all parametric spectral graph convolutions.*

BasisNets can Compute Spectral Invariants Many works compare the expressive power of GNNs to combinatorial invariants on graphs, especially the k-Weisfeiler-Lehman (k-WL) tests of graph isomorphism. We argue that it is more natural to analyze our methods in terms of *spectral invariants*, which are computed from the eigenvalues and eigenvectors of graphs. A widely studied spectral invariant is the collection of graph angles, which are defined as the values $\alpha_{ij} = \|V_i V_i^\top e_j\|_2$, where $V_i \in \mathbb{R}^{n \times d_i}$ is an orthonormal basis for the ith adjacency matrix eigenspace, and e_j is the jth standard basis vector. These are easily computed by our networks, and thus our networks inherit the strength of these invariants for learning functions on graphs.

Proposition 2. *BasisNet universally approximates the graph angles α_{ij}. The eigenvalues and graph angles (and thus BasisNets) can determine the number of length 3, 4, or 5 cycles, whether a graph is connected, and the number of length k closed walks from any vertex to itself.*

In contrast, message passing GNNs are not able express any of these properties (Arvind et al., 2020; Chen et al., 2020). Although spectral invariants are strong, Fürer (2010) shows that the eigenvalues and graph angles are not stronger than the 3-WL test (i.e. the 2-FWL test).

4 EXPERIMENTS

Graph Regression. We study the effectiveness of SignNet in learning positional encodings (PEs) from Laplacian eigenvectors on the ZINC dataset. We consider three settings: 1) No positional encoding, 2) Laplacian PE (LapPE) – the k eigenvectors of the graph Laplacian with smallest eigenvalue are concatenated with existing node features, 3) SignNet positional features – passing the eigenvectors through a SignNet and concatenating the output with node features. See Appendix G.2 for details. Results comparing different positional encoding methods are displayed in Table 1. For all 4 base model, the PE learned with SignNet yields the best test MAE (lower is better). Notably, this includes the cases of PNA and GINE, for which Laplacian PE with simple random sign flipping was unable to improve performance over using no PE at all.

Learning Spectral Graph Convolutions. We test the ability of our basis invariant networks for learning spectral graph convolutions. We compare against spectral GNNs (Bianchi et al., 2021; Defferrard et al., 2016; He et al., 2021). Also, we consider standard Transformers with only node features, with eigenvectors and sign flip augmentation, and with absolute values of eigenvectors. These models all either use eigenvectors in a sign invariant way or do not use eigenvectors. We use DeepSets in SignNet and 2-IGN (Maron et al., 2018) in BasisNet for ϕ, use a DeepSets for ρ in both cases, and then feed the features into another DeepSets or a standard Transformer (Vaswani et al., 2017) to make the final predictions. That is, we are only given graph information through the eigenvectors and eigenvalues, and we do not use message passing. Table 2 displays the results. SignNet and BasisNet allow DeepSets and Transformers to outperform all other methods on the band-rejection and comb filters, and they are close to the best model on the other filters.

REFERENCES

Vikraman Arvind, Frank Fuhlbrück, Johannes Köbler, and Oleg Verbitsky. On weisfeiler-leman invariance: Subgraph counts and related graph properties. In *Journal of Computer and System Sciences*, volume 113, pp. 42–59. Elsevier, 2020.

László Babai, Dmitry Y Grigoryev, and David M Mount. Isomorphism of graphs with bounded eigenvalue multiplicity. In *Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pp. 310–324, 1982.

Muhammet Balcilar, Guillaume Renton, Pierre Héroux, Benoit Gaüzère, Sébastien Adam, and Paul Honeine. Analyzing the expressive power of graph neural networks in a spectral perspective. In *Int. Conference on Learning Representations (ICLR)*, volume 8, 2020.

Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural computation*, 15(6):1373–1396, 2003.

Beatrice Bevilacqua, Fabrizio Frasca, Derek Lim, Balasubramaniam Srinivasan, Chen Cai, Gopinath Balamurugan, Michael M Bronstein, and Haggai Maron. Equivariant subgraph aggregation networks. In *Int. Conference on Learning Representations (ICLR)*, volume 10, 2022.

Filippo Maria Bianchi, Daniele Grattarola, Lorenzo Livi, and Cesare Alippi. Graph neural networks with convolutional arma filters. In *IEEE transactions on pattern analysis and machine intelligence*. IEEE, 2021.

Xavier Bresson and Thomas Laurent. Residual gated graph ConvNets. In *preprint arXiv:1711.07553*, 2017.

Rasmus Bro, Evrim Acar, and Tamara G Kolda. Resolving the sign ambiguity in the singular value decomposition. In *Journal of Chemometrics: A Journal of the Chemometrics Society*, volume 22, pp. 135–140. Wiley Online Library, 2008.

Michael M Bronstein, Joan Bruna, Yann LeCun, Arthur Szlam, and Pierre Vandergheynst. Geometric deep learning: going beyond euclidean data. In *IEEE Signal Processing Magazine*, volume 34, pp. 18–42. IEEE, 2017.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and deep locally connected networks on graphs. In *Int. Conference on Learning Representations (ICLR)*, volume 2, 2014.

Chen Cai and Yusu Wang. Convergence of invariant graph networks. In *preprint arXiv:2201.10129*, 2022.

Ines Chami, Sami Abu-El-Haija, Bryan Perozzi, Christopher Ré, and Kevin Murphy. Machine learning on graphs: A model and comprehensive taxonomy. In *preprint arXiv:2005.03675*, 2020.

Zhengdao Chen, Soledad Villar, Lei Chen, and Joan Bruna. On the equivalence between graph isomorphism testing and function approximation with GNNs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, pp. 1589–15902, 2019.

Zhengdao Chen, Lei Chen, Soledad Villar, and Joan Bruna. Can graph neural networks count substructures? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 10383–10395, 2020.

Eli Chien, Jianhao Peng, Pan Li, and Olgica Milenkovic. Adaptive universal generalized pagerank graph neural network. In *Int. Conference on Learning Representations (ICLR)*, volume 9, 2021.

Fan Chung. *Spectral graph theory*. American Mathematical Soc., 1997.

Gabriele Corso, Luca Cavalleri, Dominique Beaini, Pietro Liò, and Petar Veličković. Principal neighbourhood aggregation for graph nets. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 13260–13271, 2020.

Leonardo Cotta, Christopher Morris, and Bruno Ribeiro. Reconstruction for powerful graph representations. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Dragoš Cvetković, Peter Rowlinson, and Slobodan Simic. *Eigenspaces of graphs*. Number 66 in Encyclopedia of Mathematics and its Applications. Cambridge University Press, 1997.

Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 29, pp. 3844–3852, 2016.

Vijay Prakash Dwivedi and Xavier Bresson. A generalization of transformer networks to graphs. In *AAAI Workshop on Deep Learning on Graphs: Methods and Applications*, 2021.

Vijay Prakash Dwivedi, Chaitanya K Joshi, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Benchmarking graph neural networks. In *preprint arXiv:2003.00982*, 2020.

Vijay Prakash Dwivedi, Anh Tuan Luu, Thomas Laurent, Yoshua Bengio, and Xavier Bresson. Graph neural networks with learnable structural and positional representations. In *Int. Conference on Learning Representations (ICLR)*, volume 10, 2022.

Or Feldman, Amit Boyarski, Shai Feldman, Dani Kogan, Avi Mendelson, and Chaim Baskin. Weisfeiler and leman go infinite: Spectral and combinatorial pre-colorings. In *preprint arXiv:2201.13410*, 2022.

Matthias Fey and Jan Eric Lenssen. Fast graph representation learning with pytorch geometric. In *ICLR (Workshop on Representation Learning on Graphs and Manifolds)*, volume 7, 2019.

Martin Fürer. On the power of combinatorial and spectral invariants. In *Linear algebra and its applications*, volume 432, pp. 2373–2380. Elsevier, 2010.

Jean Gallier and Jocelyn Quaintance. *Differential geometry and Lie groups: a computational perspective*, volume 12. Springer Nature, 2020.

Juan A Navarro González and Juan B Sancho de Salas. *C^∞-differentiable spaces*, volume 1824. Springer, 2003.

Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864, 2016.

Will Hamilton. Graph representation learning. In *Synthesis Lectures on Artifical Intelligence and Machine Learning*, volume 14, pp. 1–159. Morgan & Claypool Publishers, 2020.

Mingguo He, Zhewei Wei, Hongteng Xu, et al. Bernnet: Learning arbitrary graph spectral filters via bernstein approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: Datasets for machine learning on graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 22118–22133, 2020a.

Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *Int. Conference on Learning Representations (ICLR)*, volume 8, 2020b.

Leo Huang, Andrew J Graven, and David Bindel. Density of states graph kernels. In *Proceedings of the 2021 SIAM International Conference on Data Mining (SDM)*, pp. 289–297. SIAM, 2021.

John J Irwin, Teague Sterling, Michael M Mysinger, Erin S Bolstad, and Ryan G Coleman. Zinc: a free tool to discover chemistry for biology. In *Journal of chemical information and modeling*, volume 52, pp. 1757–1768. ACS Publications, 2012.

Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Int. Conference on Learning Representations (ICLR)*, volume 5, 2017.

Hanspeter Kraft and Claudio Procesi. Classical invariant theory, a primer. *Lecture Notes.*, 1996.

Devin Kreuzer, Dominique Beaini, Will Hamilton, Vincent Létourneau, and Prudencio Tossou. Rethinking graph transformers with spectral attention. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

John M Lee. Smooth manifolds. In *Introduction to Smooth Manifolds*. Springer, 2013.

Ron Levie, Federico Monti, Xavier Bresson, and Michael M Bronstein. Cayleynets: Graph convolutional neural networks with complex rational spectral filters. In *IEEE Transactions on Signal Processing*, volume 67, pp. 97–109. IEEE, 2018.

Pan Li, Eli Chien, and Olgica Milenkovic. Optimizing generalized pagerank methods for seed-expansion community detection. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 32, 2019.

Pan Li, Yanbang Wang, Hongwei Wang, and Jure Leskovec. Distance encoding: Design provably more powerful neural networks for graph representation learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 4465–4478, 2020.

Haggai Maron, Heli Ben-Hamu, Nadav Shamir, and Yaron Lipman. Invariant and equivariant graph networks. In *Int. Conference on Learning Representations (ICLR)*, volume 6, 2018.

Haggai Maron, Ethan Fetaya, Nimrod Segol, and Yaron Lipman. On the universality of invariant networks. In *Int. Conference on Machine Learning (ICML)*, pp. 4363–4371. PMLR, 2019.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pp. 43–52, 2015.

Grégoire Mialon, Dexiong Chen, Margot Selosse, and Julien Mairal. GraphiT: Encoding graph structure in transformers. In *preprint arXiv:2106.05667*, 2021.

Christopher Morris, Nils M Kriege, Franka Bause, Kristian Kersting, Petra Mutzel, and Marion Neumann. Tudataset: A collection of benchmark datasets for learning with graphs. In *ICML workshop on Graph Representation Learning and Beyond*, 2020.

Antonio Ortega, Pascal Frossard, Jelena Kovačević, José MF Moura, and Pierre Vandergheynst. Graph signal processing: Overview, challenges, and applications. In *Proceedings of the IEEE*, volume 106, pp. 808–828. IEEE, 2018.

Pál András Papp, Karolis Martinkus, Lukas Faber, and Roger Wattenhofer. Dropgnn: random dropouts increase the expressiveness of graph neural networks. *Advances in Neural Information Processing Systems*, 34, 2021.

Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710, 2014.

Omri Puny, Matan Atzmon, Heli Ben-Hamu, Edward J Smith, Ishan Misra, Aditya Grover, and Yaron Lipman. Frame averaging for invariant and equivariant network design. In *Int. Conference on Learning Representations (ICLR)*, volume 10, 2022.

Kaspar Riesen and Horst Bunke. Iam graph database repository for graph based pattern recognition and machine learning. In *Joint IAPR International Workshops on Statistical Techniques in Pattern Recognition (SPR) and Structural and Syntactic Pattern Recognition (SSPR)*, pp. 287–297. Springer, 2008.

Nimrod Segol and Yaron Lipman. On universal equivariant set networks. In *Int. Conference on Learning Representations (ICLR)*, volume 7, 2019.

Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. Collective classification in network data. In *AI magazine*, volume 29, pp. 93–93, 2008.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. 2018.

Behrooz Tahmasebi, Derek Lim, and Stefanie Jegelka. Counting substructures with higher-order graph neural networks: Possibility and impossibility results. In *preprint arXiv:2012.03174*, 2020.

Edric Tam and David Dunson. Multiscale graph comparison via the embedded laplacian distance. In *preprint arXiv:2201.12064*, 2022.

Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. SIAM, 1997.

Anton Tsitsulin, Davide Mottin, Panagiotis Karras, Alexander Bronstein, and Emmanuel Müller. Netlsd: hearing the shape of a graph. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2347–2356, 2018.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 5998–6008, 2017.

Saurabh Verma and Zhi-Li Zhang. Hunt for the unique, stable, sparse and fast feature learning on graphs. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 88–98, 2017.

Soledad Villar, David Hogg, Kate Storey-Fisher, Weichi Yao, and Ben Blum-Smith. Scalars are universal: Equivariant machine learning, structured like classical physics. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Ulrike Von Luxburg. A tutorial on spectral clustering. *Statistics and computing*, 17(4):395–416, 2007.

Haorui Wang, Haoteng Yin, Muhan Zhang, and Pan Li. Equivariant and stable positional encoding for more powerful graph neural networks. In *Int. Conference on Learning Representations (ICLR)*, volume 10, 2022.

Hassler Whitney. The self-intersections of a smooth n-manifold in $2n$-space. In *Annals of Mathematics*, pp. 220–246, 1944.

Zhenqin Wu, Bharath Ramsundar, Evan N Feinberg, Joseph Gomes, Caleb Geniesse, Aneesh S Pappu, Karl Leswing, and Vijay Pande. Moleculenet: a benchmark for molecular machine learning. In *Chemical science*, volume 9, pp. 513–530. Royal Society of Chemistry, 2018.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. A comprehensive survey on graph neural networks. In *IEEE transactions on neural networks and learning systems*, volume 32, pp. 4–24. IEEE, 2020.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *Int. Conference on Learning Representations (ICLR)*, volume 7, 2019.

Keyulu Xu, Jingling Li, Mozhi Zhang, Simon S Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka. What can neural networks reason about? In *Int. Conference on Learning Representations (ICLR)*, volume 8, 2020.

Pinar Yanardag and SVN Vishwanathan. Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1365–1374, 2015.

Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 34, 2021.

Jiaxuan You, Jonathan M Gomes-Selman, Rex Ying, and Jure Leskovec. Identity-aware graph neural networks. In *Association for the Advancement of Artificial Intelligence (AAAI)*, volume 35, pp. 10737–10745, 2021.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 3391–3401, 2017.

Jiawei Zhang, Haopeng Zhang, Congying Xia, and Li Sun. Graph-BERT: Only attention is needed for learning graph representations. In *preprint arXiv:2001.05140*, 2020.

Zhen Zhang, Mianzhi Wang, Yijian Xiang, Yan Huang, and Arye Nehorai. RetGK: Graph kernels based on return probabilities of random walks. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 31, pp. 3964–3974, 2018.

Lingxiao Zhao, Wei Jin, Leman Akoglu, and Neil Shah. From stars to subgraphs: Uplifting any GNN with local structure awareness. In *Int. Conference on Learning Representations (ICLR)*, volume 10, 2022.

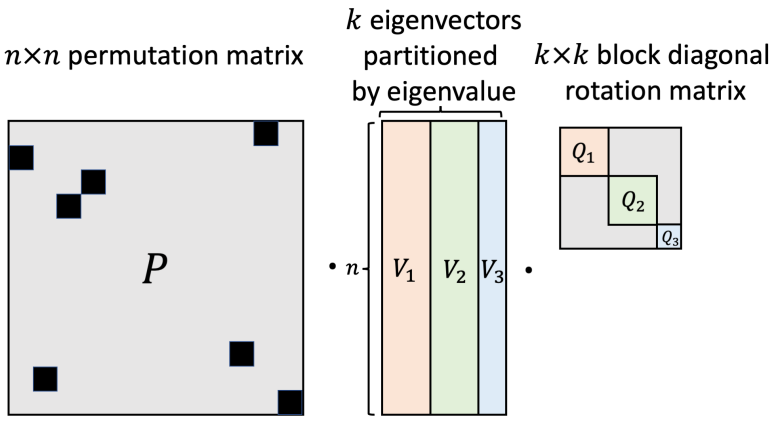

Figure 1: Symmetries of eigenvectors of a symmetric matrix with permutation symmetries (e.g. a graph Laplacian). A neural network applied to the eigenvector matrix (middle) should be invariant or equivariant to permutation of the rows (left product with a permutation matrix P) and invariant to the choice of eigenvectors in each eigenbasis (right product with a block diagonal orthogonal matrix $\mathrm{Diag}(Q_1, Q_2, Q_3)$).

A Related Work

A.1 Graph Positional Encodings

Various graph positional encodings have been proposed, which have been motivated for increasing expressive power or practical performance of graph neural networks, and for generalizing Transformers to graphs. Positional encodings are related to so-called position-aware network embeddings (Chami et al., 2020), which capture distances between nodes in graphs. These include network embedding methods like Deepwalk (Perozzi et al., 2014) and node2vec (Grover & Leskovec, 2016), which have been recently integrated into GNNs that respect their invariances by Wang et al. (2022). Further, Li et al. (2020) studies the theoretical and practical benefits of incorporating distance features into graph neural networks. Dwivedi et al. (2022) proposes a method to inject learnable positional encodings into each layer of a graph neural network, and uses a simple random walk based node positional encoding. You et al. (2021) proposes a node positional encoding $\mathrm{diag}(A^k)$, which captures the number of closed walks from a node to itself. Dwivedi et al. (2020) propose to use Laplacian eigenvectors as positional encodings in graph neural networks, with sign ambiguities alleviated by sign flipping data augmentation.

While positional encodings in sequences as used for Transformers (Vaswani et al., 2017) are able to leverage the canonical order in sequences, there is no such useful canonical order for nodes in a graph, due in part to permutation symmetries. Thus, different permutation equivariant positional encodings have been proposed to help generalize Transformers to graphs. Dwivedi & Bresson (2021) directly adds in linearly projected Laplacian eigenvectors to node features before processing these features with a Transformer. Kreuzer et al. (2021) propose an architecture that uses attention over Laplacian eigenvectors and eigenvalues to learn node or edge positional encodings. Mialon et al. (2021) uses spectral kernels such as the diffusion kernel to define relative positional encodings that modulate the attention matrix. Ying et al. (2021) achieve state-of-the-art empirical performance with simple Transformers that incorporate shortest-path based relative positional encodings. Zhang et al. (2020) also utilizes shortest-path distances for positional encodings in their graph Transformer.

A.2 Eigenvector Symmetries in Graph Representation Learning

Many works that attempt to respect the invariances of eigenvectors solely focus on sign invariance (Dwivedi et al., 2020; Dwivedi & Bresson, 2021; Dwivedi et al., 2022; Kreuzer et al., 2021). This may be reasonable for continuous data, where eigenvalues of associated matrices may be usually distinct and separated (e.g. Puny et al. (2022) finds that this empirically holds for covariance

matrices of n-body problems). However, discrete graph Laplacians are known to have higher multiplicity eigenvalues in many cases, and in Appendix E.2 we find this to be true in various types of real-world graph data. Graphs without higher multiplicity eigenspaces are easier to deal with; in fact, graph isomorphism can be tested in polynomial time on graphs of bounded multiplicity for adjacency matrix eigenvalues (Babai et al., 1982), with a time complexity that is lower for graphs with lower maximum multiplicities.

A recent work of Wang et al. (2022) proposes full orthogonal group invariance for functions that process positional encodings. In particular, for positional encodings $Z \in \mathbb{R}^{n \times k}$, they parameterize functions $f(Z)$ such that $f(Z) = f(ZQ)$ for all $Q \in O(k)$. This indeed makes sense for network embeddings like node2vec (Grover & Leskovec, 2016), as their objective functions are based on inner products and are thus orthogonally invariant. While they prove stability results when enforcing full orthogonal invariance for eigenvectors, this is too strict of a constraint compared to our basis invariance. For instance, when $k = n$ and all eigenvectors are used in V, the condition $f(V) = f(VQ)$ implies that f is a constant function on orthogonal matrices, since any orthogonal matrix W can be obtained as $W = VQ$ for $Q = V^\top W \in O(n)$. In other words, for bases of eigenspaces $V_1, \ldots, V_l$ and $V = [V_1 \quad \ldots \quad V_l]$, Wang et al. (2022) enforces $VQ \cong V$, while we enforce $V\mathrm{Diag}(Q_1, \ldots, Q_l) \cong V$. While the columns of $V\mathrm{Diag}(Q_1, \ldots, Q_l)$ are still eigenvectors, the columns of VQ generally are not.

A.3 Graph Spectra and Learning on Graphs

More generally, graph spectra is widely used in analyzing graphs, and spectral graph theory (Chung, 1997) studies the connection between graph properties and graph spectra. Different graph kernels have been defined based on graph spectra, which use robust and discriminative notions of generalized spectral distance (Verma & Zhang, 2017), the spectral density of states (Huang et al., 2021), random walk return probabilities (Zhang et al., 2018), or the trace of the heat kernel (Tsitsulin et al., 2018). Graph signal processing relies on spectral operations to define Fourier transforms, frequencies, convolutions, and other useful concepts for processing data on graphs (Ortega et al., 2018). The closely related spectral graph neural networks (Wu et al., 2020; Balcilar et al., 2020) parameterize neural architectures that are based on similar spectral operations.

B Definitions, Notations, and Background

B.1 Basic Topology and Algebra Definitions

We will use some basic topology and algebra for our theoretical results. A topological space $(\mathcal{X}, \tau)$ is a set $\mathcal{X}$ along with a family of subsets $\tau \subseteq 2^{\mathcal{X}}$ satisfying certain properties, which gives useful notions like continuity and compactness. From now on, we will omit mention of τ, and refer to a topological space as the set $\mathcal{X}$ itself. For topological spaces $\mathcal{X}$ and $\mathcal{Y}$, we write $\mathcal{X} \cong \mathcal{Y}$ and say that $\mathcal{X}$ is homeomorphic to $\mathcal{Y}$ if there exists a continuous bijection with continuous inverse from $\mathcal{X}$ to $\mathcal{Y}$. We will say $\mathcal{X} = \mathcal{Y}$ if the underlying sets and topologies are equal as sets (we will often use this notion of equality for simplicity, even though it can generally be substituted with homeomorphism). For a function $f : \mathcal{X} \to \mathcal{Y}$ between topological spaces $\mathcal{X}$ and $\mathcal{Y}$, the image $\mathrm{im} f$ is the set of values that f takes, $\mathrm{im} f = \{f(x) : x \in \mathcal{X}\}$. This is also denoted $f(\mathcal{X})$. A function $f : \mathcal{X} \to \mathcal{Y}$ is called a topological embedding if it is a homeomorphism from $\mathcal{X}$ to its image.

A group G is a set along with a multiplication operation $G \times G \to G$, such that multiplication is associative, there is a multiplicative identity $e \in G$, and each $g \in G$ has a multiplicative inverse g^{-1}. A topological group is a group that is also a topological space such that the multiplication and inverse operations are continuous.

A group G may act on a set $\mathcal{X}$ by a function $\cdot : G \times \mathcal{X} \to \mathcal{X}$. We usually denote $g \cdot x$ as gx. A topological group is said to act continuously on a topological space $\mathcal{X}$ if $\cdot$ is continuous. For any group G and topological space $\mathcal{X}$, we define the coset $Gx = \{gx : x \in \mathcal{X}\}$, which can be viewed as an equivalance class of elements that can be transformed from one to another by a group element. The quotient space $\mathcal{X}/G = \{Gx : x \in \mathcal{X}\}$ is the set of all such equivalence classes, with a topology induced by $\mathcal{X}$.

For $x \in \mathbb{R}^s$, $\|x\|_2$ denotes the standard Euclidean norm. By the ∞ norm of functions $f : \mathcal{Z} \to \mathbb{R}^s$ from a compact $\mathcal{Z}$ to a Euclidean space $\mathbb{R}^s$, we mean $\|f\|_\infty = \sup_{z \in \mathcal{Z}} \|f(z)\|_2$.

B.2 Background on Eigenspace Invariances

Let $V = [v_1 \quad \dots \quad v_d]$ and $W = [w_1 \quad \dots \quad w_d] \in \mathbb{R}^{n \times d}$ be two orthonormal bases for the same d dimensional subspace of $\mathbb{R}^n$. Define a linear map $T : \mathbb{R}^d \to \mathbb{R}^d$ by mapping $T(v_i) = w_i$. This map is orthogonal, because for any $v = \sum_i a_i v_i$ and $v' = \sum_i a'_i v_i$ in $\mathbb{R}^d$, we have

$$T(v)^\top T(v) = \sum_i a_i a'_i w_i^\top w_i = \sum_i a_i a'_i = v^\top v. \tag{8}$$

Thus, T can be written as an orthogonal matrix Q such that $VQ = W$. This means that any two orthonormal bases of the same subspace can be mapped to each other by right multiplication by an orthogonal matrix.

For another perspective on this, define the Grassmannian $\mathrm{Gr}(d, n)$ as the smooth manifold consisting of all d dimensional subspaces of $\mathbb{R}^n$. Further define the Stiefel manifold $\mathrm{St}(d, n)$ as the set of all orthonormal tuples $[v_1 \quad \dots \quad v_d] \in \mathbb{R}^{n \times d}$ of d vectors in $\mathbb{R}^n$. Letting $O(d)$ act by right multiplication, it holds that $\mathrm{St}(d, n)/O(d) \cong \mathrm{Gr}(d, n)$. This implies that any $O(d)$ invariant function on $\mathrm{St}(d, n)$ can be viewed as a function on subspaces. See e.g. Gallier & Quaintance (2020) Chapter 5 for more information on this. We will use this relationship in our proofs of universal representation.

When we consider permutation invariance or equivariance, the permutation acts on dimensions of size n. Then a tensor $X \in \mathbb{R}^{n^k \times d}$ is called an order k tensor with respect to this permutation symmetry, where order 0 are called scalars, order 1 tensors are called vectors, and order 2 tensors are called matrices. Note that this does not depend on d; in this work, we only ever consider vectors and scalars with respect to the $O(d)$ action.

C Proofs of Universality

We begin by proving the two propositions for the single subspace case from Section 2.1.

Proposition 3. *A continuous function $h : \mathbb{R}^n \to \mathbb{R}^s$ is sign invariant if and only if*

$$h(v) = \phi(v) + \phi(-v) \tag{9}$$

for some continuous $\phi : \mathbb{R}^n \to \mathbb{R}^s$. A continuous $h : \mathbb{R}^n \to \mathbb{R}^n$ is sign invariant and permutation equivariant if and only if equation 9 holds for a continuous permutation equivariant $\phi : \mathbb{R}^n \to \mathbb{R}^n$.

Proof. If $h(v) = \phi(v) + \phi(-v)$, then h is obviously sign invariant. On the other hand, if h is sign invariant, then letting $\phi(v) = h(v)/2$ gives that $h(v) = \phi(v) + \phi(-v)$, and ϕ is of course continuous.

If $h(v) = \phi(v) + \phi(-v)$ for a permutation equivariant ϕ, then $h(-Pv) = \phi(-Pv) + \phi(Pv) = P\phi(-v) + P\phi(v) = P(\phi(v) + \phi(-v)) = Ph(v)$, so h is permutation equivariant and sign invariant. If h is permutation equivariant and sign invariant, then define $\phi(v) = h(v)/2$ again; it is clear that ϕ is continuous and permutation equivariant. □

Proposition 4. *Any continuous, $O(d)$ invariant $h : \mathbb{R}^{n \times d} \to \mathbb{R}^s$ is of the form $h(V) = \varphi(VV^\top)$ for a continuous φ. For a compact domain $\mathcal{Z} \subseteq \mathbb{R}^{n \times d}$, maps of the form $V \mapsto \mathrm{IGN}(VV^\top)$ universally approximate continuous functions $h : \mathcal{Z} \subseteq \mathbb{R}^{n \times d} \to \mathbb{R}^n$ that are $O(d)$ invariant and permutation equivariant.*

Proof. The case without permutation equivariance holds by the First Fundamental Theorem of $O(d)$ (Lemma 2).

For the permutation equivariant case, let $\mathcal{Z}' = \{VV^\top : V \in \mathcal{Z}\}$ and let $\epsilon > 0$. Note that $\mathcal{Z}'$ is compact, as it is the continuous image of a compact set. Since h is $O(d)$ invariant, the first fundamental theorem of $O(d)$ shows that there exists a continuous function $\varphi : \mathcal{Z}' \subseteq \mathbb{R}^{n \times n} \to \mathbb{R}^n$

such that $h(V) = \varphi(VV^\top)$. Since h is permutation equivariant, for any permutation matrix P we have that

$$h(PV) = P \cdot h(V) \tag{10}$$

$$\varphi(PVV^\top P^\top) = P \cdot \varphi(VV^\top), \tag{11}$$

so φ is a continuous permutation equivariant function from matrices to vectors. Then note that Keriven & Peyré (2019) show that invariant graph networks (of generally high tensor order in hidden layers) universally approximate continuous permutation equivariant functions from matrices to vectors on compact sets of matrices. Thus, an IGN can ϵ-approximate φ, and hence $V \mapsto \mathrm{IGN}(VV^\top)$ can ϵ-approximate h. □

C.1 Proof of Decomposition Theorem

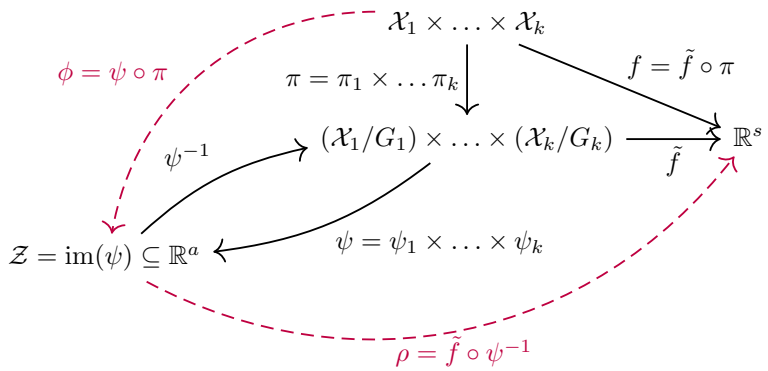

Figure 2: Commutative diagram for our proof of Theorem 1. Black arrows denote functions from topological constructions, and red dashed lines denote functions that we parameterize by neural networks ($\phi = \phi_1 \times \ldots \times \phi_k$ and ρ).

Here, we give the formal statement of Theorem 1, which provides the necessary topological assumptions for the theorem to hold. In particular, we only require the G_i be a topological group that acts continuously on $\mathcal{X}_i$ for each i, and that there exists a topological embedding of each quotient space into some Euclidean space. The first assumption is very mild, and it holds for any finite or compact matrix group, which all of the invariances we consider in this paper can be represented as. Also, many different conditions can guarantee existence of a topological embedding of the quotient space in a Euclidean space. For instance, if the quotient space is a smooth manifold, then the Whitney Embedding Theorem (Lemma 5) guarantees such an embedding. Also, if the base space $\mathcal{X}_i$ is a Euclidean space and G_i is a finite or compact matrix Lie group, then a map built from G-invariant polynomials gives such an embedding (González & de Salas (2003) Lemma 11.13).

Figure 2 provides a commutative diagram representing the constructions in our proof.

Theorem 1 (Decomposition Theorem)**.** *Let $\mathcal{X}_1, \ldots, \mathcal{X}_k$ be topological spaces, and let G_i be a topological group acting continuously on $\mathcal{X}_i$ for each i. Assume that there is a topological embedding $\psi_i : \mathcal{X}_i/G_i \to \mathbb{R}^{a_i}$ of each quotient space into a Euclidean space $\mathbb{R}^{a_i}$ for some dimension a_i. Then, for any continuous function $f : \mathcal{X} = \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to \mathbb{R}^s$ that is invariant to the action of $G = G_1 \times \ldots \times G_k$, there exists continuous functions $\phi_i : \mathcal{X}_i \to \mathbb{R}^{a_i}$ and a continuous function $\rho : \mathcal{Z} \subseteq \mathbb{R}^a \to \mathbb{R}^s$, where $a = \sum_i a_i$ such that*

$$f(v_1, \ldots, v_k) = \rho(\phi_1(v_1), \ldots, \phi_k(v_k)). \tag{12}$$

Furthermore: (1) each ϕ_i can be taken to be invariant to G_i, (2) the domain $\mathcal{Z}$ is compact if each $\mathcal{X}_i$ is compact, (3) if $\mathcal{X}_i = \mathcal{X}_j$ and $G_i = G_j$, then ϕ_i can be taken to be equal to ϕ_j.

Proof. Let $\pi_i : \mathcal{X}_i \to \mathcal{X}_i/G_i$ denote the quotient map for $\mathcal{X}_i/G_i$. Since each $\mathcal{G}_i$ acts continuously, Lemma 3 gives that the quotient of the product space is the product of the quotient spaces, i.e. that

$$(\mathcal{X}_1 \times \ldots \times \mathcal{X}_k)/(G_1 \times \ldots G_k) \cong (\mathcal{X}_1/G_1) \times \ldots \times (\mathcal{X}_k/G_k), \tag{13}$$

and the corresponding quotient map $\pi : \mathcal{X}/G$ is given by

$$\pi = \pi_1 \times \ldots \times \pi_k, \qquad \pi(x_1, \ldots, x_k) = (\pi_1(x_1), \ldots, \pi_k(x_k)). \tag{14}$$

By passing to the quotient (Lemma 1), there exists a continuous $\tilde{f} : \mathcal{X}/\mathcal{G} \to \mathbb{R}^s$ on the quotient space such that $f = \tilde{f} \circ \pi$. By Lemma 4, each $\mathcal{X}_i/G_i$ is compact if $\mathcal{X}_i$ is compact. Defining the image $\mathcal{Z}_i = \psi_i(\mathcal{X}_i/G_i) \subseteq \mathbb{R}^{a_i}$, we thus know that $\mathcal{Z}_i$ is compact if $\mathcal{X}_i$ is compact.

Moreover, as ψ_i is a topological embedding, it has a continuous inverse ψ_i^{-1} on its image $\mathcal{Z}_i$. Further, we have a topological embedding $\psi : \mathcal{X}/G \to \mathcal{Z} = \mathcal{Z}_1 \times \ldots \times \mathcal{Z}_k$ given by $\psi = \psi_1 \times \ldots \times \psi_k$, with continuous inverse $\psi^{-1} = \psi_1^{-1} \times \ldots \times \psi_k^{-1}$.

Note that

$$f = \tilde{f} \circ \pi = (\tilde{f} \circ \psi^{-1}) \circ (\psi \circ \pi). \tag{15}$$

So we define

$$\rho = \tilde{f} \circ \psi^{-1} \qquad \rho : \mathcal{Z} \to \mathbb{R}^s \tag{16}$$

$$\phi_i = \psi_i \circ \pi_i \qquad \phi_i : \mathcal{X}_i \to \mathcal{Z}_i \tag{17}$$

$$\phi = \psi \circ \pi = \phi_1 \times \ldots \times \phi_k \qquad \phi : \mathcal{X} \to \mathcal{Z} \tag{18}$$

Thus, $f = \rho \circ \phi = \rho \circ (\phi_1 \times \ldots \times \phi_k)$, so equation equation 7 holds. Moreover, the ρ and ϕ_i are continuous, as they are compositions of continuous functions. Furthermore, (1) holds as each ϕ_i is invariant to G_i because each π_i is invariant to G_i. Since each $\mathcal{Z}_i$ is compact if $\mathcal{X}_i$ is compact, the product $\mathcal{Z} = \mathcal{Z}_1 \times \ldots \times \mathcal{Z}_k$ is compact if each $\mathcal{X}_i$ is compact, thus proving (2).

To show the last statement (3), note simply that if $\mathcal{X}_i = \mathcal{X}_j$ and $G_i = G_j$, then the quotient maps are equal, i.e. $\pi_i = \pi_j$. Moreover, we can choose the embeddings to be equal, so say $\psi_i = \psi_j$. Then, $\phi_i = \psi_i \circ \pi_i = \psi_j \circ \pi_j = \phi_j$, so we are done. □

C.2 Universality of SignNet and BasisNet

Here, we prove Corollary 1 on the universal representation and approximation capabilities of our unconstrained SignNets and unconstrained BasisNets. We proceed in several steps.

Expressive-BasisNet. While we restrict SignNet to only process vectors and BasisNet to only process vectors and matrices, higher order tensors are generally required for universality in expressing permutation equivariant or invariant networks (Keriven & Peyré, 2019; Maron et al., 2019). Thus, we will consider a theoretically powerful but computationally impractical variant of our model, in which we replace ρ and IGN_{d_i} in BasisNet with IGNs of arbitrary tensor order. We call this variant *Expressive-BasisNet*. In this Appendix, we will also prove that Expressive-BasisNet can universally approximate basis invariant and permutation equivariant continuous functions.

C.2.1 Sign Invariant Universal Representation

Recall that $\mathbb{S}^{n-1}$ denotes the unit sphere in $\mathbb{R}^n$. As we normalize eigenvectors to unit norm, the domain of our functions on k eigenvectors are on the compact space $(\mathbb{S}^{n-1})^k$.

Corollary 2 (Universal Representation for SignNet)**.** *A continuous function $f : (\mathbb{S}^{n-1})^k \to \mathbb{R}^s$ is sign invariant, i.e. $f(s_1 v_1, \ldots, s_k v_k) = f(v_1, \ldots, v_k)$ for any $s_i \in \{-1, 1\}$, if and only if there exists a continuous $\phi : \mathbb{R}^n \to \mathbb{R}^{2n-2}$ and a continuous $\rho : \mathbb{R}^{(2n-2)k} \to \mathbb{R}^s$ such that*

$$f(v_1, \ldots, v_k) = \rho\left([\phi(v_i) + \phi(-v_i)]_{i=1}^k\right). \tag{19}$$

Proof. It can be directly seen that any f of the above form is sign invariant.

Thus, we show that any sign invariant f can be expressed in the above form. First, we show that we can apply the general Theorem 1. The group $G_i = \{1, -1\}$ acts continuously and satisfies that $\mathbb{S}^{n-1}/\{1, -1\} = \mathbb{RP}^{n-1}$, where $\mathbb{RP}^{n-1}$ is the real projective space of dimension $n-1$. Since $\mathbb{RP}^{n-1}$ is a smooth manifold of dimension $n-1$, Whitney's embedding theorem states that there exists a (smooth) topological embedding $\psi_i : \mathbb{RP}^{n-1} \to \mathbb{R}^{2n-2}$ (Lemma 5).

Thus, we can apply the general theorem to see that $f = \rho \circ \tilde{\phi}^k$ for some continuous ρ and $\tilde{\phi}^k$. Note that each $\tilde{\phi}_i = \tilde{\phi}$ is the same, as each $\mathcal{X}_i = \mathbb{S}^{n-1}$ and $G_i = \{1, -1\}$ is the same. Also, Theorem 1 says that we may assume that $\tilde{\phi}$ is sign invariant, so $\tilde{\phi}(x) = \tilde{\phi}(-x)$. Letting $\phi(x) = \tilde{\phi}(x)/2$, we are done with the proof. □

C.2.2 SIGN INVARIANT UNIVERSAL REPRESENTATION WITH EXTRA FEATURES

Recall that we may want our sign invariant functions to process other data besides eigenvectors, such as eigenvalues or node features associated to a graph. Here, we show universal representation for when we have this other data that does not possess sign symmetry. The proof is a simple extension of Corollary 2, but we provide the technical details for completeness.

Corollary 3 (Universal Representation for SignNet with features)**.** *For a compact space of features $\Omega \subseteq \mathbb{R}^d$, let $f(v_1, \ldots, v_k, x_1, \ldots, x_k)$ be a continuous function $f : (\mathbb{S}^{n-1} \times \Omega)^k \to \mathbb{R}^s$.*

Then f is sign invariant for the inputs on the sphere, i.e.

$$f(s_1 v_1, \ldots, s_k v_k, x_1, \ldots, x_k) = f(v_1, \ldots, v_k, x_1, \ldots, x_k) \qquad s_i \in \{1, -1\}, \tag{20}$$

if and only if there exists a continuous $\psi : \mathbb{R}^{n+d} \to \mathbb{R}^{2n-2+d}$ and a continuous $\rho : \mathbb{R}^{(2n-2+d)k} \to \mathbb{R}^s$ such that

$$f(v_1, \ldots, v_k) = \rho\left(\varphi(v_1, x_1) + \varphi(-v_1, x_1),\ \ldots,\ \varphi(v_k, x_k) + \varphi(-v_k, x_k)\right). \tag{21}$$

Proof. Once again, the sign invariance of any f in the above form is clear.

We follow very similar steps to the proof of Corollary 2 to show that we may apply Theorem 1. We can view Ω as a quotient space, after quotienting by the trivial group that does nothing, $\Omega \cong \Omega/\{1\}$. The corresponding quotient map is id_Ω, the identity map. Also, Ω trivially topologically embeds in $\mathbb{R}^d$ by the inclusion map.

As $G_i = \{-1, 1\} \times \{1\}$ acts continuously, by Lemma 3 we have that

$$(\mathbb{S}^{n-1} \times \Omega)/(\{1, -1\} \times \{1\}) \cong (\mathbb{S}^{n-1}/\{1, -1\}) \times (\Omega/\{1\}) \cong \mathbb{RP}^{n-1} \times \Omega, \tag{22}$$

with corresponding quotient map $\pi \times \mathrm{id}_\Omega$, where π is the quotient map to $\mathbb{RP}^{n-1}$.

Letting $\tilde{\psi}$ be the embedding of $\mathbb{RP}^{n-1} \to \mathbb{R}^{2n-2}$ guaranteed by Whitney's embedding theorem (Lemma 5), we have that $\psi = \tilde{\psi} \times \mathrm{id}_\Omega$ is an embedding of $\mathbb{RP}^{n-1} \times \Omega \to \mathbb{R}^{2n-2+d}$. Thus, we can apply Theorem 1 to write $f = \rho \circ \tilde{\phi}^k$ for $\tilde{\phi} = (\tilde{\psi} \times \mathrm{id}_\Omega) \circ (\pi \times \mathrm{id}_\Omega)$, so

$$\tilde{\phi}(v_i, x_i) = (\tilde{\psi}(v_i), x_i), \tag{23}$$

where $\tilde{\phi}(v_i, x_i) = \tilde{\phi}(-v_i, x_i)$. Letting $\phi(v_i, x_i) = \tilde{\phi}(v_i, x_i)/2$, we are done. □

C.2.3 BASIS INVARIANT UNIVERSAL REPRESENTATION

Recall that $\mathrm{St}(d, n)$ is the Stiefel manifold of k-tuples of vectors $(v_1, \ldots, v_k)$ where $v_i \in \mathbb{R}^n$ and $v_1, \ldots, v_k$ are orthonormal. This is where our inputs lie, as our eigenvectors are unit norm and orthogonal. We will also make use of the Grassmannian $\mathrm{Gr}(d, n)$, which consists of all d-dimensional subspaces in $\mathbb{R}^n$. This is because the Grassmannian is the quotient space for the group action we want, $\mathrm{Gr}(d, n) \cong \mathrm{St}(d, n)/O(d)$, where $Q \in O(d)$ acts on $V \in \mathrm{St}(d, n) \subseteq \mathbb{R}^{n \times d}$ by mapping V to VQ (Gallier & Quaintance, 2020).

Corollary 4 (Universal Representation for BasisNet)**.** *For dimensions $d_1, \ldots, d_l \leq n$ let f be a continuous function on $\mathrm{St}(d_1, n) \times \ldots \times \mathrm{St}(d_l, n)$. Further assume that f is invariant to $O(d_1) \times \ldots \times O(d_l)$, where $O(d_i)$ acts on $\mathrm{St}(d_i, n)$ by multiplication on the right.*

Then there exist continuous $\rho : \mathbb{R}^{\sum_{i=1}^{l} 2d_i(n-d_i)} \to \mathbb{R}^s$ and continuous $\phi_i : \mathrm{St}(d_i, n) \to \mathbb{R}^{2d_i(n-d_i)}$ such that

$$f(V_1, \ldots, V_l) = \rho\left(\phi_1(V_1), \ldots, \phi_l(V_l)\right), \tag{24}$$

where the ϕ_i are $O(d_i)$ invariant functions, and we can take $\phi_i = \phi_j$ if $d_i = d_j$.

Proof. Letting $\mathcal{X}_i = \mathrm{St}(d_i, n)$ and $G_i = O(d_i)$, it can be seen that G_i acts continuously on $\mathcal{X}_i$. Also, we have that the quotient space $\mathrm{St}(d_i, n)/O(d_i) = \mathrm{Gr}(d_i, n)$ is the Grassmannian of d_i dimensional subspaces in $\mathbb{R}^n$, which is a smooth manifold of dimension $d_i(n-d_i)$. Thus, the Whitney embedding theorem (Lemma 5) gives a topological embedding $\psi_i : \mathrm{Gr}(d_i, n) \to \mathbb{R}^{2d_i(n-d_i)}$.

Hence, we may apply Theorem 1 to obtain continuous $O(d_i)$ invariant $\phi_i : \mathrm{St}(d_i, n) \to \mathbb{R}^{2d_i(n-d_i)}$ and continuous $\rho : \mathbb{R}^{\sum_{i=1}^{l} 2d_i(n-d_i)} \to \mathbb{R}^s$, such that $f = \rho \circ (\phi_1 \times \ldots \times \phi_l)$. Also, if $d_i = d_j$, then $\mathcal{X}_i = \mathcal{X}_j$ and $G_i = G_j$, so we can take $\phi_i = \phi_j$.

□

C.2.4 BASIS INVARIANT AND PERMUTATION EQUIVARIANT UNIVERSAL APPROXIMATION

With the restriction that $f(V_1, \ldots, V_l) : \mathbb{R}^{n \times \sum_i d_i} \to \mathbb{R}^n$ be permutation equivariant and basis invariant, we need to use the impractically expensive Expressive-BasisNet to approximate f. Universality of permutation invariant or equivariant functions from matrices to scalars or matrices to vectors is difficult to achieve in a computationally tractable manner (Maron et al., 2019; Keriven & Peyré, 2019). One intuitive reason to expect this is that universally approximating such functions allows solution of the graph isomorphism problem (Chen et al., 2019), which is a computationally difficult problem. While we have exact representation of basis invariant functions by continuous ρ and ϕ_i when there is no permutation equivariance constraint, we can only achieve approximation up to an arbitrary $\epsilon > 0$ when we require permutation equivariance.

Corollary 5 (Universal Approximation for Expressive-BasisNets)**.** *Let $f(V_1, \ldots, V_l) : \mathrm{St}(d_1, n) \times \ldots \times \mathrm{St}(d_l, n) \to \mathbb{R}^n$ be continuous, $O(d_1) \times \ldots \times O(d_l)$ invariant, and permutation equivariant. Then f can be ϵ-approximated by an Expressive-BasisNet.*

Proof. By invariance, Corollary 4 of the decomposition theorem shows that f can be written as

$$f(V_1, \ldots, V_l) = \rho\left(\phi_{d_1}(V_1), \ldots, \phi_{d_l}(V_l)\right) \tag{25}$$

for some continuous $O(d_i)$ invariant ϕ_{d_i} and continuous ρ. By the first fundamental theorem of $O(d)$ (Lemma 2), each ϕ_{d_i} can be written as $\phi_{d_i}(V_i) = \varphi_{d_i}(V_i, V_i^\top)$ for some continuous φ_{d_i}. Let

$$\mathcal{Z} = \{(V_1 V_1^\top, \ldots, V_l V_l^\top) : V_i \in \mathrm{St}(d_i, n)\} \subseteq \mathbb{R}^{n^2 \times l}, \tag{26}$$

which is compact as it is the image of the compact space $\mathrm{St}(d_1, n) \times \ldots \times \mathrm{St}(d_l, n)$ under a continuous function. Define $h : \mathcal{Z} \subseteq \mathbb{R}^{n^2 \times l} \to \mathbb{R}^n$ by

$$h(V_1 V_1^\top, \ldots, V_l V_l^\top) = \rho\left(\varphi_{d_1}(V_1 V_1^\top), \ldots, \varphi_{d_l}(V_l V_l^\top)\right). \tag{27}$$

Then note that h is continuous and permutation equivariant from matrices to vectors, so it can be ϵ-approximated by an invariant graph network (Keriven & Peyré, 2019), call it $\widetilde{\mathrm{IGN}}$. If we define $\tilde{\rho} = \widetilde{\mathrm{IGN}}$ and $\mathrm{IGN}_{d_i}(V_i V_i^\top) = V_i V_i^\top$ (this identity operation is linear and permutation equivariant, so it can be exactly expressed by an IGN), then we have ϵ-approximation of f by

$$\widetilde{\mathrm{IGN}}(V_1 V_1^\top, \ldots, V_l V_l^\top) = \tilde{\rho}\left(\mathrm{IGN}_{d_1}(V_1 V_1^\top), \ldots, \mathrm{IGN}_{d_l}(V_l V_l^\top)\right). \tag{28}$$

□

C.3 PROOF OF UNIVERSAL APPROXIMATION FOR GENERAL DECOMPOSITIONS

Theorem 2 (Universal Approximation)**.** *Consider the same setup as Theorem 1, where $\mathcal{X}_i$ are also compact. Let Φ_i be a family of G_i-invariant functions that universally approximate G_i-invariant continuous functions $\mathcal{X}_i \to \mathbb{R}^{a_i}$, and let $\mathcal{R}$ be a set of continuous function that universally approximate continuous functions $\mathcal{Z} \subseteq \mathbb{R}^a \to \mathbb{R}^s$ where $a = \sum_i a_i$ for every compact $\mathcal{Z}$. Then for any $\varepsilon > 0$ and any G-invariant continuous function $f : \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to \mathbb{R}^s$ there exists $\phi \in \Phi$ and $\rho \in \mathcal{R}$ such that $\|f - \rho(\phi_1, \ldots, \phi_k)\|_\infty < \varepsilon$.*

Proof. Consider a particular G-invariant continuous function $f : \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to \mathbb{R}^s$. By Theorem 1 there exists G_i-invariant continuous functions $\phi_i' : \mathcal{X}_i \to \mathbb{R}^{a_i}$ and a continuous function $\rho' : \mathcal{Z} \subseteq \mathbb{R}^a \to \mathbb{R}^s$ (where $a = \sum_i a_i$) such that

$$f(v_1, \ldots, v_k) = \rho'(\phi_1'(v_1), \ldots, \phi_k'(v_k)).$$

Now fix an $\varepsilon > 0$. For any $\rho \in \mathcal{R}$ and any $\phi_i \in \Phi_i$ $(i = 1, \ldots k)$ we may bound the difference from f as follows (suppressing the v_i's for brevity),

$$\begin{aligned}
&\|f - \rho(\phi_1, \ldots, \phi_k)\|_\infty \\
&= \|\rho'(\phi_1', \ldots, \phi_k') - \rho(\phi_1, \ldots, \phi_k)\|_\infty \\
&= \|\rho'(\phi_1', \ldots, \phi_k') - \rho(\phi_1', \ldots, \phi_k') + \rho(\phi_1', \ldots, \phi_k') - \rho(\phi_1, \ldots, \phi_k)\|_\infty \\
&\leq \|\rho'(\phi_1', \ldots, \phi_k') - \rho(\phi_1', \ldots, \phi_k')\|_\infty + \|\rho(\phi_1', \ldots, \phi_k') - \rho(\phi_1, \ldots, \phi_k)\|_\infty \\
&= \mathrm{I} + \mathrm{II}
\end{aligned}$$

Now let $K' = \prod_{i=1}^k \mathrm{im}\phi_i'$. Since each ϕ_i' is continuous and defined on a compact set $\mathcal{X}_i$ we know that $\mathrm{im}\phi_i'$ is compact, and so the product K is also compact. Since K' is compact, it is contained in a closed ball $B(r)$ of radius $r > 0$ centered at the origin. Let K be the closed ball $B(r+1)$ of radius $r+1$ centered at the origin, so K contains K' and a ball of radius 1 around each point of K'. We may extend ρ' continuously to K as needed, so assume $\rho' : K \to \mathbb{R}^s$. By universality of $\mathcal{R}$ we may pick a particular $\rho : K \to \mathbb{R}^s$, $\rho \in \mathcal{R}$ such that

$$\mathrm{I} = \sup_{\{v_i \in \mathcal{X}_i\}_{i=1}^k} \|\rho'(\phi_1', \ldots, \phi_k') - \rho(\phi_1', \ldots, \phi_k')\|_\infty \leq \sup_{z \in K} \|\rho'(z) - \rho(z)\|_2 < \varepsilon/2.$$

Keeping this choice of ρ, it remains only to bound II. As ρ is continuous on a compact domain, it is in fact uniformly continuous. Thus, we can choose a $\delta' > 0$ such that if $\|y - z\|_2 \leq \delta'$, then $\|\rho(y) - \rho(z)\|_\infty < \epsilon/2$, and then we define $\delta = \min(\delta', 1)$.

Since Φ_i universally approximates ϕ_i' we may pick $\phi_i \in \Phi_i$ such that $\|\phi_i - \phi_i'\|_\infty < \delta/\sqrt{k}$, and thus $\|(\phi_1, \ldots, \phi_k) - (\phi_1', \ldots \phi_k')\|_\infty \leq \delta$. With this choice of ϕ_i, we know that $\prod_{i=1}^k \mathrm{im}\phi_i \subseteq K$ (because each $\phi_i(x_i)$ is within distance 1 of $\phi_i'(x_i)$). Thus, $\rho(\phi_1(x_1), \ldots, \phi_k(x_k))$ is well-defined, and we have

$$\begin{aligned}
\mathrm{II} &= \|\rho(\phi_1', \ldots, \phi_k') - \rho(\phi_1, \ldots, \phi_k)\|_\infty \\
&= \sup_{\{x_i \in \mathcal{X}_i\}_{i=1}^k} \|\rho(\phi_1'(x_1), \ldots, \phi_k'(x_k)) - \rho(\phi_1(x_1), \ldots, \phi_k(x_k))\|_2 \\
&< \varepsilon/2
\end{aligned}$$

due to our choice of δ, which completes the proof. □

D Basis Invariance for Graph Representation Learning

D.1 Spectral Graph Convolution

In this section, we consider spectral graph convolutions, which for node features $X \in \mathbb{R}^{n \times q}$ take the form $f(V, \Lambda, X) = \sum_{i=1}^n \theta_i v_i v_i^\top X$ for some parameters θ_i. We can optionally take $\theta_i = h(\lambda_i)$ for some continuous function $h : \mathbb{R} \to \mathbb{R}$ of the eigenvalues. This form captures most popular spectral graph convolutions in the literature (Bruna et al., 2014; Hamilton, 2020; Bronstein et al., 2017); often, such convolutions are parameterized by taking h to be some analytic function such as a simple affine function (Kipf & Welling, 2017), a linear combination in a polynomial basis (Defferrard et al., 2016; Chien et al., 2021), or a parameterization of rational functions (Levie et al., 2018; Bianchi et al., 2021).

First, it is well known and easy to see that spectral graph convolutions are permutation equivariant, as for a permutation matrix P we have

$$f(PV, \Lambda, PX) = \sum_i \theta_i P v_i v_i^\top P^\top P X = \sum_i \theta_i P v_i v_i^\top X = P f(V, \Lambda, X). \tag{29}$$

Also, it is easy to see that they are sign invariant, as $(-v_i)(-v_i)^\top = v_i v_i^\top$. However, if the θ_i do not depend on the eigenvalues, then the spectral graph convolution is not necessarily basis invariant. For instance, if v_1 and v_2 are in the same eigenspace, and we change basis by permuting $v_1' = v_2$ and $v_2' = v_1$, then if $\theta_1 \neq \theta_2$ the spectral graph convolution will generally change as well.

On the other hand, if $\theta_i = h(\lambda_i)$ for some function $h : \mathbb{R} \to \mathbb{R}$, then the spectral graph convolution is basis invariant. This is because if v_i and v_j belong to the same eigenspace, then $\lambda_i = \lambda_j$ so $h(\lambda_i) = h(\lambda_j)$. Thus, if $v_{i_1}, \ldots, v_{i_d}$ are eigenvectors of the same eigenspace with eigenvalue λ, we have that $\sum_{l=1}^d h(\lambda_{i_l}) v_{i_l} v_{i_l}^\top = h(\lambda) \sum_{l=1}^d v_{i_l} v_{i_l}^\top$. Now, note that $\sum_{l=1}^d v_{i_l} v_{i_l}^\top$ is the orthogonal projector onto the eigenspace (Trefethen & Bau III, 1997). A change of basis does not change this orthogonal projector, so such spectral graph convolutions are basis invariant.

Another way to see this basis invariance is through basic algebra. Let $V_1, \ldots, V_l$ be the eigenspaces of dimension $d_1, \ldots, d_l$, where $V_i \in \mathbb{R}^{n \times d_i}$. Let the corresponding eigenvalues be $\mu_1, \ldots, \mu_l$. Then

for any orthogonal matrices $Q_i \in O(d_i)$, we have

$$\sum_{i=1}^{n} h(\lambda_i) v_i v_i^\top = \sum_{j=1}^{l} V_j h(\mu_j) I_{d_j} V_j^\top \tag{30}$$

$$= \sum_{j=1}^{l} V_j h(\mu_j) I_{d_j} Q_j Q_j^\top V_j^\top \tag{31}$$

$$= \sum_{j=1}^{l} (V_j Q_j) h(\mu_j) I_{d_j} (V_j Q_j)^\top, \tag{32}$$

so the spectral graph convolution is invariant to substituting $V_j Q_j$ for V_j.

Now, we give the proof that shows SignNet and BasisNet can universally approximate spectral graph convolutions.

Proposition 1 (Learning Spectral Graph Convolutions)**.** *Suppose the node features $X \in \mathbb{R}^{n \times q}$ take values in compact sets. Then SignNet can universally approximate any spectral graph convolution, and both BasisNet and Expressive-BasisNet can universally approximate any parametric spectral graph convolution.*

Proof. Note that eigenvectors and eigenvalues of normalized Laplacian matrices take values in compact sets, since the eigenvalues are in $[0, 2]$ and we take eigenvectors to have unit-norm. Thus, the whole domain of the spectral graph convolution is compact.

Let $\varepsilon > 0$. First, consider a spectral graph convolution $f(V, \Lambda, X) = \sum_{i=1}^{n} \theta_i v_i v_i^\top X$. For SignNet, let $\phi(v_i, \lambda_i, X)$ approximate the function $\tilde{\phi}(v_i, \lambda_i, X) = \theta_i v_i v_i^\top X$ to within ε/n error, which DeepSets can do since this is a continuous permutation equivariant function from vectors to vectors (Segol & Lipman, 2019) (note that we can pass λ_i as a vector in $\mathbb{R}^n$ by instead passing $\lambda_i \mathbf{1}$, where $\mathbf{1}$ is the all ones vector). Then $\rho = \sum_{i=1}^{n}$ is a linear permutation equivariant operation that can be exactly expressed by DeepSets, so the total error is within ε. The same argument applies when $\theta_i = h(\lambda_i)$ for some continuous function h.

For the basis invariant case, consider a parametric spectral graph convolution $f(V, \Lambda, X) = \sum_{i=1}^{n} h(\lambda_i) v_i v_i^\top X$. Note that if the eigenspace bases are $V_1, \ldots, V_l$ with eigenvalues $\mu_1, \ldots, \mu_l$, we can write the $f(V, \Lambda, X) = \sum_{i=1}^{l} h(\mu_j) V_j V_j^\top X$. Again, we will let $\rho = \sum_{i=1}^{l}$ be a sum function, which can be expressed exactly by DeepSets. Thus, it suffices to show that $h(\mu_j) V_j V_j^\top X$ can be ϵ/n approximated by a 2-IGN (i.e. an IGN that only uses vectors and matrices).

Note that since h is continuous, we can use an elementwise MLP (which IGNs can learn) to approximate $f_1(\mu \mathbf{1}\mathbf{1}^\top, VV^\top, X) = (h(\mu)\mathbf{1}\mathbf{1}^\top, VV^\top, X)$ to arbitrary precision (note that we represent the eigenvalue μ as a constant matrix $\mu \mathbf{1}\mathbf{1}^\top$). Also, since a 2-IGN can learn matrix vector multiplication (Cai & Wang (2022) Lemma 10), we can approximate $f_2(h(\mu)\mathbf{1}\mathbf{1}^\top, VV^\top, X) = (h(\mu)\mathbf{1}\mathbf{1}^\top, VV^\top X)$, as $V_i V_i^\top \in \mathbb{R}^{n^2}$ is a matrix and $X \in \mathbb{R}^{n \times q}$ is a vector with respect to permutation symmetries. Finally, we use an elementwise MLP to approximate the scalar-vector multiplication $f_3(h(\mu)\mathbf{1}\mathbf{1}^\top, VV^\top, X) = h(\mu) VV^\top X$. Since $f_3 \circ f_2 \circ f_1(\mu \mathbf{1}\mathbf{1}^\top, VV^\top, X) = h(\mu) VV^\top X$, and since 2-IGNs universally approximate each f_i, applying Lemma 6 shows that a 2-IGN can approximate $h(\mu) VV^\top X$ to ϵ/n accuracy, so we are done. Since Expressive-BasisNet is stronger than BasisNet, it can also universally approximate these functions. □

From the proof, we can see that SignNet and BasisNet need only learn simple functions for the ρ and ϕ when h is simple, or when the filter is non-parametric and we need only learn θ_i. Xu et al. (2020) propose the principle of algorithmic alignment, and show that if separate modules of a neural network each need only learn simple functions (that is, functions that are well-approximated by low-order polynomials with small coefficients), then the network may be more sample efficient. If we do not require permutation equivariance, and parameterize SignNet and BasisNet with simple MLPs, then algorithmic alignment may suggest that our models are sample efficient. Indeed, $\rho = \sum$ is a simple linear function with coefficients 1, and $\phi(V, \lambda, X) = h(\lambda) VV^\top X$ is quadratic in V and linear in X, so it is simple if h is simple.

D.2 Existing Positional Encodings

Here, we show that our SignNets and BasisNets universally approximate various types of existing graph positional encodings. The key is to show that these positional encodings are related to spectral graph convolution matrices and diagonals, and to show that our networks can approximate these matrices and diagonals.

Proposition 5. *SignNets and BasisNets universally approximate the diagonal of any spectral graph convolution matrix* $f(V,\Lambda) = \operatorname{diag}\left(\sum_{i=1}^n h(\lambda_i)v_iv_i^\top\right)$. *BasisNets can additionally universally approximate any spectral graph convolution matrix* $f(V,\Lambda) = \sum_{i=1}^n h(\lambda_i)v_iv_i^\top$.

Proof. Note that v_i and λ_i come from compact sets, as $\lambda_i \in [0,2]$ for the normalize Laplacian and v_i is of unit norm. Also, as diag is linear, the spectral graph convolution diagonal can be written $\sum_{i=1}^n h(\lambda_i)\operatorname{diag}(v_iv_i^\top)$.

Let $\epsilon > 0$. For SignNet, let $\rho = \sum_{i=1}^n$, which can be exactly expressed as it is a permutation equivariant linear operation from vectors to vectors. Then $\phi(v_i,\lambda_i)$ can approximate the function $\lambda_i\operatorname{diag}(v_iv_i^\top)$ to arbitrary precision, as it is a permutation equivariant function from vectors to vectors (Segol & Lipman, 2019). Thus, letting ϕ approximate the function to ϵ/n accuracy, SignNet can approximate f to ϵ accuracy.

Let l be the number of eigenspaces $V_1,\ldots,V_l$, so $f(V,\Lambda) = \sum_{i=1}^l h(\mu_i)V_iV_i^\top$. For BasisNet, we need only show that it can approximate the spectral graph convolution matrix to ϵ/l accuracy, as a 2-IGN can exactly express the diag function in ρ, since it is a linear permutation equivariant function from matrices to vectors. A 2-IGN can universally approximate the function $f_1(\mu_i, V_iV_i^\top) = (h(\mu_i), V_iV_i^\top)$, as it can express any elementwise MLP. Also, a 2-IGN can universally approximate the scalar-matrix multiplication $f_2(h(\mu_i), V_iV_i^\top) = h(\mu_i)V_iV_i^\top$ by another elementwise MLP. Since $h(\mu_i)V_iV_i^\top = f_2 \circ f_1(\mu_i, V_iV_i^\top)$, Lemma 6 shows that a single 2-IGN can approximate this composition to ϵ/l accuracy, so we are done.

□

Proposition 6. *SignNets and BasisNets universally approximate heat kernel positional encodings (Feldman et al., 2022) and random walk node positional encodings (RWPE) (Dwivedi et al., 2022). BasisNets universally approximate diffusion and* p*-step random walk relative positional encodings (Mialon et al., 2021), as well as generalized PageRank and landing probability distance encodings (Li et al., 2020).*

Proof. We will show that we can apply the above Proposition 5, by showing that all of these positional encodings are spectral graph convolutions. The heat kernel embeddings are of the form $\operatorname{diag}\left(\sum_{i=1}^n \exp(-t\lambda_i)v_iv_i^\top\right)$ for some choices of the parameter t, so they can be approximated by SignNets or BasisNets. Also, the diffusion kernel (Mialon et al., 2021) is just the matrix of this heat kernel, and the p-step random walk kernel is $\sum_{i=1}^n(1-\gamma\lambda_i)^p v_iv_i^\top$ for some parameter γ, so BasisNets can universally approximate both of these.

For the other positional encodings, we let v_i be the eigenvectors of the random walk Laplacian $I - D^{-1}A$ instead of the normalized Laplacian $I - D^{-1/2}AD^{-1/2}$. The eigenvalues of these two Laplacians are the same, and if $\tilde{v}_i$ is an eigenvector of the normalized Laplacian then $D^{-1/2}\tilde{v}_i$ is an eigenvector of the random walk Laplacian with the same eigenvalue (Von Luxburg, 2007).

Then with v_i as the eigenvectors of the random walk Laplacian, the random walk positional encodings (RWPE) in Dwivedi et al. (2022) take the form

$$\operatorname{diag}\left((D^{-1}A)^k\right) = \operatorname{diag}\left(\sum_{i=1}^n (1-\lambda_i)^k v_iv_i^\top\right), \tag{33}$$

for any choices of integer k.

The distance encodings proposed in Li et al. (2020) take the form

$$f_3(AD^{-1}, (AD^{-1})^2, (AD^{-1})^3, \ldots), \tag{34}$$

for some function f_3. We restrict to continuous f_3 here; shortest path distances can be obtained by a discontinuous f_3 that we discuss below. Their generalized PageRank based distance encodings can be obtained by

$$\sum_{i=1}^{n} \left(\sum_{k \geq 1} \gamma_k (1 - \lambda_i)^k \right) v_i v_i^\top \tag{35}$$

for some $\gamma_k \in \mathbb{R}$, so this is a spectral graph convolution. They also define so-called landing probability based positional encodings, which take the form

$$\sum_{i=1}^{n} (1 - \lambda_i)^k v_i v_i^\top, \tag{36}$$

for some choices of integer k. Thus, BasisNets can approximate these distance encoding matrices. □

Another powerful class of positional encodings is based on shortest path distances between nodes in the graph (Ying et al., 2021; Li et al., 2020). Shortest path distances can be expressed in a form similar to the spectral graph convolution, but require a highly discontinuous function. If we define $\varphi(x_1, \ldots, x_n) = \min_{i: x_i \neq 0} i$ to be the lowest index such that x_i is nonzero, then we can write the shortest path distance matrix as $\varphi(D^{-1}A, (D^{-1}A)^2, \ldots, (D^{-1}A)^n)$, where φ is applied elementwise to return an $n \times n$ matrix. As $(D^{-1}A)^k = \sum_{i=1}^{n} (1 - \lambda_i)^k v_i v_i^\top$, BasisNets can learn the inside arguments, but cannot learn the discontinuous function φ.

D.3 Spectral Invariants

Here, we consider the graph angles $\alpha_{ij} = \|V_i V_i^\top e_j\|_2$, for $i = 1, \ldots, l$ where l is the number of eigenspaces, and $j = 1, \ldots, n$. It is clear that graph angles are permutation invariant and basis invariant. These graph angles have been extensively studied, so we simply cite a number of interesting properties of them. That graph angles determine the number of length 3, 4 and 5 cycles, the connectivity of a graph, and the number of length k closed walks is all shown in Chapter 4 of Cvetković et al. (1997). Other properties may be of use for graph representation learning as well. For instance, the eigenvalues of node-deleted subgraphs of a graph $\mathcal{G}$ are determined by the eigenvalues and graph angles of $\mathcal{G}$; this may be useful in extending recent graph neural networks that are motivated by node deletion and the reconstruction conjecture (Cotta et al., 2021; Bevilacqua et al., 2022; Papp et al., 2021; Tahmasebi et al., 2020).

Now, we prove that BasisNet can universally approximate the graph angles. The graph properties we consider in the proposition are all integer valued (e.g. the number of cycles of length 3 in a graph is an integer). Thus, any two graphs that differ in these properties will differ by at least 1, so as long as we have universal approximation to $\varepsilon < 1/2$, we can distinguish any two graphs that differ in these properties. Recall the statement of Proposition 2.

Proposition 2. *BasisNet can universally approximate the graph angles α_{ij}. The eigenvalues and graph angles (and thus BasisNets) can determine the number of length 3, 4, and 5 cycles, whether a graph is connected, and the number of length k closed walks from any vertex to itself.*

Proof. Note that the graph angles satisfy

$$\alpha_{ij} = \|V_i V_i^\top e_j\|_2 = \sqrt{e_j^\top V_i V_i^\top V_i V_i^\top e_j} = \sqrt{e_j^\top V_i V_i^\top e_j}, \tag{37}$$

where V_i is a basis for the ith adjacency matrix eigenspace, and $e_j^\top V_i V_i^\top e_j$ is the (j, j)-entry of $V_i V_i^\top$. These graph angles are just the elementwise square roots of the diagonals of the matrices $V_i V_i^\top$. As $f_1(V_i V_i^\top) = \mathrm{diag}(V_i V_i^\top)$ is a permutation equivariant linear function from matrices to vectors, 2-IGN on $V_i V_i^\top$ can exactly compute this with 0 error. Then a 2-IGN can learn an elementwise MLP to approximate the elementwise square root $f_2(\mathrm{diag}(V_i V_i^\top)) = \sqrt{\mathrm{diag}(V_i V_i^\top)}$ to arbitrary precision. Finally, there may be remaining operations f_3 that are permutation invariant or permutation equivariant from vectors to vectors; for instance, the α_{ij} are typically gathered into a matrix of size $l \times n$ where the columns are lexicographically sorted (l is the number of

eigenspaces) (Cvetković et al., 1997), or we may have a permutation invariant readout to compute a subgraph count. A DeepSets can approximate f_3 without any higher order tensors besides vectors (Zaheer et al., 2017; Segol & Lipman, 2019).

As 2-IGNs can approximate each f_i individually, a single 2-IGN can approximate $f_3 \circ f_2 \circ f_1$ by Lemma 6. Also, since the graph properties considered in the proposition are integer-valued, BasisNet can distinguish any two graphs that differ in one of these properties. □

E EIGENVECTORS AND EIGENVALUES IN PRACTICE

In this section, we study the properties of eigenvalues and eigenvectors computed by numerical algorithms on real-world data.

E.1 SIGN AND BASIS AMBIGUITIES IN NUMERICAL EIGENSOLVERS

When learning from real-world data, we use eigenvectors that are computed by numerical algorithms. These algorithms return specific eigenvectors for each eigenspace, so there is some theoretically arbitrary choice of sign or basis of each eigenspace. The general symmetric matrix eigensolvers `numpy.linalg.eigh` and `scipy.linalg.eigh` both call LAPACK routines. They both proceed as follows: for a symmetric matrix A, they first decompose it as $A = QTQ^\top$ for orthogonal Q and tridiagonal T, then they compute the eigendecomposition of $T = W\Lambda W^\top$, so the eigendecomposition of A is $A = (QW)\Lambda(W^\top Q^\top)$. There are multiple ambiguities here: for diagonal sign matrices $S = \mathrm{Diag}(s_1, \ldots, s_n)$ and $S' = \mathrm{Diag}(s'_1, \ldots, s'_n)$, where $s_i, s'_i \in \{-1, 1\}$, we have that $A = QS(STS)SQ^\top$ is also a valid tridiagonalization, as QS is still orthogonal, $SS = I$, and STS is still tridiagonal. Also, $T = (WS')\Lambda(S'W^\top)$ is a valid eigendecomposition of T, as WS' is still orthogonal.

In practice, the general symmetric matrix eigensolvers `numpy.linalg.eigh` and `scipy.linalg.eigh` differ between frameworks but are consistent with the same framework. More specifically, for a symmetric matrix A, we find that the eigenvectors computed with the default settings in numpy tend to differ by a choice of sign or basis from those that are compute with the default settings in scipy. On the other hand, the called LAPACK routines are deterministic, so the eigenvectors returned by numpy are the same in each call, and the eigenvectors returned by scipy are likewise the same in each call.

Eigensolvers for sparse symmetric matrices like `scipy.linalg.eigsh` are required for large scale problems. This function calls ARPACK, which uses an iterative method that starts with a randomly sampled initial vector. Due to this stochasticity, the sign and basis of eigenvectors returned differs between each call.

Bro et al. (2008) develops a data-dependent method to choose signs for each singular vector of a singular value decomposition. Still, in the worst case the signs chosen will be arbitrary, and they do not handle rotational ambiguities in higher dimensional eigenspaces. Other works have made choices of sign, such as by picking the sign so that the eigenvector's entries are in the largest lexicographic order (Tam & Dunson, 2022). This choice of sign may work poorly for learning on graphs, as it is sensitive to permutations on nodes. For some graph regression experiments in Section 4, we try a choice of sign that is permutation invariant, but we find it to work poorly.

E.2 HIGHER DIMENSIONAL EIGENSPACES IN REAL GRAPHS

Here, we investigate the normalized Laplacian eigenspace statistics of real world graph data. For any graph that has distinct Laplacian eigenvalues, only sign invariance is required in processing eigenvectors. However, we find that graph data tends to have higher multiplicity eigenvalues, so basis invariance would be required for learning symmetry-respecting functions on eigenvectors.

Indeed, we show statistics for multi-graph datasets in Table 3 and for single-graph datasets with more nodes per graph in Table 4. For multi-graph datasets, we consider :

- Molecule graphs: ZINC (Irwin et al., 2012; Dwivedi et al., 2020), ogbg-molhiv (Wu et al., 2018; Hu et al., 2020a)

Table 3: Eigenspace statistics for datasets of multiple graphs. From left to right, the columns are: dataset name, number of graphs, range of number of nodes per graph, largest multiplicity, and percent of graphs with an eigenspace of dimension > 1.

Dataset	Graphs	# Nodes	Max. Mult	% Graphs mult. > 1
ZINC	12,000	9-37	9	64.1
ogbg-molhiv	41,127	2 - 222	42	68.0
IMDB-M	1,500	7 - 89	37	99.9
COLLAB	5,000	32 - 492	238	99.1
PROTEINS	1,113	4 - 620	20	77.3
COIL-DEL	3,900	3 - 77	4	4.00

Table 4: Eigenspace statistics for single graphs. From left to right, the columns are: dataset name, number of nodes, distinct eigenvalues (i.e. distinct eigenspaces), number of unique multiplicities, largest multiplicity, and percent of eigenvectors belonging to an eigenspace of dimension > 1.

Dataset	Nodes	Distinct λ	# Mult.	Max Mult.	% Vecs mult. > 1
32×32 image	1,024	513	3	32	96.9
Cora	2,708	2,187	11	300	19.7
Citeseer	3,327	1,861	12	491	44.8
Amazon Photo	7,650	7,416	8	136	3.71

- Social networks: IMDB-M, COLLAB (Yanardag & Vishwanathan, 2015; Morris et al., 2020),
- Bioinformatics graphs: PROTEINS (Morris et al., 2020)
- Computer vision graphs: COIL-DEL (Riesen & Bunke, 2008; Morris et al., 2020).

For single-graph datasets, we consider:

- The 32×32 image grid as in Section 4
- Citation networks: Cora, Citeseer (Sen et al., 2008)
- Co-purchasing graphs with Amazon Photo (McAuley et al., 2015; Shchur et al., 2018).

We see that these datasets all contain higher multiplicity eigenspaces, so sign invariance is insufficient for fully respecting symmetries. The majority of graphs in each multi-graph dataset besides COIL-DEL contain higher multiplicity eigenspaces. Also, the dimension of these eigenspaces can be quite large compared to the size of the graphs in the dataset. The single-graph datasets have a large proportion of their eigenvectors belonging to higher dimensional eigenspaces. Thus, basis invariance may play a large role in processing spectral information from these graph datasets.

F Counting Substructures and Regressing Graph Properties

Table 5: Counting substructures and regressing graph properties. With Laplacian PEs, SignNet improves performance, while sign flip data augmentation is less consistent. Mean and standard deviations are reported on 3 runs. The same 4-layer GIN is used across all settings for fair comparison.

Base Model	PE	Counting Substructures (MAE)				Graph Properties ($\log_{10}$(MAE))		
		Triangle	Tailed Tri.	Star	4-Cycle	IsConnected	Diameter	Radius
GIN	No PE	$0.357_{\pm 0.002}$	$0.248_{\pm 0.007}$	$0.020_{\pm 0.001}$	$0.225_{\pm 0.002}$	$-1.873_{\pm 0.014}$	$-3.487_{\pm 0.031}$	$-4.832_{\pm 0.027}$
	LapPE (flip)	$0.147_{\pm 0.009}$	$0.120_{\pm 0.007}$	$0.014_{\pm 0.001}$	$0.136_{\pm 0.011}$	$-1.884_{\pm 0.020}$	$-3.456_{\pm 0.018}$	$-4.800_{\pm 0.027}$
	SignNet	$\mathbf{0.037}_{\pm \mathbf{0.002}}$	$\mathbf{0.036}_{\pm \mathbf{0.011}}$	$\mathbf{0.012}_{\pm \mathbf{0.001}}$	$\mathbf{0.028}_{\pm \mathbf{0.004}}$	$\mathbf{-2.426}_{\pm \mathbf{0.128}}$	$\mathbf{-3.658}_{\pm \mathbf{0.031}}$	$\mathbf{-4.931}_{\pm \mathbf{0.037}}$

Substructure counts (e.g. of cycles) and global graph properties (e.g. connectedness, diameter, radius) are important graph features that are known to be informative for problems in bio- and

chemo-informatics (Chen et al., 2020; Corso et al., 2020). Following the setting in Zhao et al. (2022), we show that SignNet with Laplacian positional encodings boosts the ability of simple GNNs to count substructures and regress graph properties. We take a 4-layer GIN as the base model for all settings, and for SignNet we use GIN as ϕ and a Transformer as ρ to handle variable numbers of eigenvectors (see Appendix. G.3 for details). As shown in Table 5, Laplacian PEs with flip-based data augmentation improves performance in counting substructures but not in regressing graph properties, while Laplacian PEs processed by SignNet significantly boost performance on all tasks.

G FURTHER EXPERIMENTAL DETAILS

G.1 SPECTRAL GRAPH CONVOLUTION DETAILS

In Section 4, we conduct node regression experiments for learning spectral graph convolutions. The experimental setup is mostly taken from He et al. (2021). However, we take the dataset of 50 images in He et al. (2021) (originally from the Image Processing Toolbox of MATLAB), and resize them from 100×100 to 32×32. Thus, each image is viewed as a 1024-node graph. The node features $X \in \mathbb{R}^n$ are the grayscale pixel intensities of each node. Then we apply the same spectral graph convolutions on them as in He et al. (2021), and train neural networks to learn these as regression targets. As in prior work, we report sum of squared errors on the training set to test expressivity.

Just as in He et al. (2021), we only train and evaluate on nodes that are not connected to the boundary of the grid (that is, we only evaluate on the 28×28 middle section). For all experiments we limit each model to 50,000 parameters. For each of the GNN baselines (GPR-GNN, ARMA, ChebNet, BernNet), we select the best performing out of 4 hyperparameter settings: either 2 or 4 convolution layers, and a hidden dimension of size 32 or D, where D is just large enough to stay with 50,000 parameters (for instance, $D = 128$ for BernNet).

We use DeepSets or standard Transformers as our prediction network. This takes in the output of SignNet or BasisNet and concatenates it with the node features, then outputs a scalar prediction. We use a 3 layer output network for DeepSets SignNet, and 2 layer output networks for all other configurations. All networks use ReLU activations.

For SignNet, we use DeepSets for both ϕ and ρ. Our ϕ takes in eigenvectors only, then our ρ takes the outputs of ϕ and the eigenvalues. We use three layers for ϕ and ρ.

For BasisNet, we use the same DeepSets for ρ as in SignNet, and 2-IGNs for the ϕ_{d_i}. There are three distinct multiplicities for the grid graph (1, 2, and 32), so we only need 3 separate IGNs. Each IGN consists of an $\mathbb{R}^{n^2 \times 1} \to \mathbb{R}^{n \times d'}$ layer and two $\mathbb{R}^{n \times d''} \to \mathbb{R}^{n \times d'''}$ layers, where the d' are hidden dimensions. There are no matrix to matrix operations used, as the memory requirements are intensive for these ≥ 1000 node graphs. The ϕ_{d_i} only take in $V_i V_i^\top$ from the eigenspaces, and the ρ takes the output of the ϕ_{d_i} as well as the eigenvalues.

G.2 GRAPH REGRESSION DETAILS

We parameterize the SignNet by taking ϕ to be 8 layer GIN with ReLU activation (Xu et al., 2019) and ρ to be an 8 layer MLP with batch normalization and ReLU activation. We consider four different base models: GatedGCN (Bresson & Laurent, 2017), a Transformer with sparse attention only over neighbours as introduced by Kreuzer et al. (2021), PNA (Corso et al., 2020), and GIN (Xu et al., 2019) with edge features (i.e. GINE) (Hu et al., 2020b). The combined total number of parameters of the SignNet and the base model is kept within a 500k budget by adjusting network width.

The input eigenvector $v_i \in \mathbb{R}^n$, where n is the number of nodes in the graph, is treated as a single scalar feature for each node. The aggregator ρ is applied separately to the concatenatation of the k different embeddings for each node in a graph, resulting in one single embedding per node. This embedding is concatenated to the node features for that node, and the result passed as input to the base (predictor) model.

Our training pipeline largely follows that of Dwivedi et al. (2022), and we use the GatedGCN and PNA base models from the accompanying implementation (see `https://github.com/vijaydwivedi75/gnn-lspe`). The Sparse Transformer base model architecture we use, which

Table 6: Results on the ZINC dataset with 500k parameter budget and no edge features. Numbers are the mean and standard deviation over 4 runs each with different seeds.

Base model	Positional encoding	k	#params	Test MAE (↓)
GIN	No PE	16	497k	$0.348_{\pm 0.014}$
	LapPE (flip)	16	498k	$0.341_{\pm 0.011}$
	SignNet	16	500k	$\mathbf{0.238}_{\pm \mathbf{0.012}}$
GAT	No PE	16	501k	$0.464_{\pm 0.011}$
	LapPE (flip)	16	502k	$0.462_{\pm 0.013}$
	SignNet	16	499k	$\mathbf{0.243}_{\pm \mathbf{0.008}}$

like GAT computes attention only across neighbouring nodes, is introduced by Kreuzer et al. (2021). Finally, the GINE implementation is based on the PyTorch Geometric implementation (Fey & Lenssen, 2019).

All models in Table 1 use edge features for learning and inference. To show that SignNet is also useful when no edge features are available, we ran ZINC experiments without edge features as well. The results are displayed in Table 6.

G.3 Substructures and Graph Properties Regression Details

We use the random graph dataset from Chen et al. (2020) for counting substructures and the synthetic dataset from Corso et al. (2020) for regressing graph properties. To keep fair comparison we fix the base model as a 4-layer GIN model with hidden size 128. As graphs can have different number of nodes, the number of Laplacian eigenvectors is the same as the number of nodes and hence is different across graphs. One can truncate it to keep the top k most important eigenvectors but this loses information. We may instead use all eigenvectors by designing the model to handle positional encodings of varying sizes. SignNet can accept varying numbers of Laplacian eigenvectors as input by choosing ρ and ϕ to be neural networks allowing variable-size input. Specifically, we choose ϕ as 4-layer GIN (independently applied to every eigenvector) and ρ as 1-layer Transformer (independently applied to every node). Combined with proper batching and masking, we have a SignNet that takes Laplacian eigenvectors $V \in \mathbb{R}^{n \times n}$ and outputs fixed size sign-invariant encoding node features $f(V, \Lambda, X) \in \mathbb{R}^{n \times d}$, where n varies between graphs but d is fixed. We use this SignNet in our experiments and compare with other methods of handling PEs.

H Useful Lemmas

In this section, we collect useful lemmas for our proofs. These lemmas generally only require basic tools to prove. Our first lemma is a crucial property of quotient spaces.

Lemma 1 (Passing to the quotient)**.** *Let $\mathcal{X}$ and $\mathcal{Y}$ be topological spaces, and let $\mathcal{X}/G$ be a quotient space, with corresponding quotient map π. Then for every continuous G-invariant function $f : \mathcal{X} \to \mathcal{Y}$, there is a unique continuous $\tilde{f} : \mathcal{X}/G \to \mathcal{Y}$ such that $f = \tilde{f} \circ \pi$.*

Proof. For $z \in \mathcal{X}/G$, by surjectivity of π we can choose an $x_z \in \mathcal{X}$ such that $\pi(x_z) = z$. Define $\tilde{f} : \mathcal{X}/G \to \mathcal{Y}$ by $\tilde{f}(z) = f(x_z)$. This is well-defined, since if $\pi(x_z) = \pi(x)$ for any other $x \in \mathcal{X}$, then $gx_z = x$ for some $g \in G$, so

$$f(x) = f(gx_z) = f(x_z) = \tilde{f}(z), \tag{38}$$

where the second equality uses the G-invariance of f. Note that $\tilde{f}$ is continuous by the universal property of quotient spaces. Also, $\tilde{f}$ is the unique function such that $f = \tilde{f} \circ \pi$; if there were another function $h : \mathcal{X}/G \to \mathcal{Y}$ with $h(z) \neq \tilde{f}(z)$, then $h(z) \neq f(x_z)$, so $h(\pi(x_z)) = h(z) \neq f(x_z)$. □

Next, we give the First Fundamental Theorem of $O(d)$, a classical result that has been recently used for machine learning by Villar et al. (2021). This result shows that an orthogonally invariant $f(V)$

can be expressed as a function $h(VV^\top)$. We give a proof that if f is continuous, then h is also continuous.

Lemma 2 (First Fundamental Theorem of $O(d)$)**.** *A continuous function $f : \mathbb{R}^{n\times d} \to \mathbb{R}^s$ is orthgonally invariant, i.e. $f(VQ) = f(V)$ for all $Q \in O(d)$, if and only if $f(V) = h(VV^\top)$ for some continuous h.*

Proof. If $f(V) = h(VV^\top)$, then we have $f(VQ) = h(VQQ^\top V^\top) = h(VV^\top)$ so f is orthogonally invariant.

For the other direction, invariant theory shows that the $O(d)$ invariant polynomials are generated by the inner products $v_i^\top v_j$, where $v_i \in \mathbb{R}^d$ are the rows of V (Kraft & Procesi, 1996). Let $p : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times n}$ be the map $p(V) = VV^\top$. Then González & de Salas (2003) Lemma 11.13 shows that the quotient space $\mathbb{R}^{n\times d}/O(d)$ is homeomorphic to a closed subset $p(\mathbb{R}^{n\times d}) = \mathcal{Z} \subseteq \mathbb{R}^{n\times n}$, let $\tilde{p}$ refer to this homeomorphism, and note that $\tilde{p} \circ \pi = p$ by passing to the quotient (Lemma 1). Then any continuous $O(d)$ invariant f passes to a unique continuous $\tilde{f} : \mathbb{R}^{n\times d}/O(d) \to \mathbb{R}^s$ (Lemma 1), so $f = \tilde{f} \circ \pi$ where π is the quotient map. Define $h : \mathcal{Z} \to \mathbb{R}^s$ by $h = \tilde{f} \circ \tilde{p}^{-1}$, and note that h is a composition of continuous functions and hence continuous. Finally, we have that $h(VV^\top) = h(\tilde{p} \circ \pi(V)) = \tilde{f} \circ \pi(V) = f(V)$, so we are done. □

The next lemma allows us to decompose a quotient of a product space into a product of smaller quotient spaces.

Lemma 3. *Let $\mathcal{X}_1, \ldots, \mathcal{X}_k$ be topological spaces and $G_1, \ldots, G_k$ be topological groups such that each G_i acts continuously on $\mathcal{X}_i$. Denote the quotient maps by $\pi_i : \mathcal{X}_i \to \mathcal{X}_i/G_i$. Then the quotient of the product is the product of the quotient, i.e.*

$$(\mathcal{X}_1 \times \ldots \times \mathcal{X}_k)/(G_1 \times \ldots \times G_k) \cong (\mathcal{X}_1/G_1) \times \ldots \times (\mathcal{X}_k/G_k), \tag{39}$$

and $\pi_1 \times \ldots \times \pi_k : \mathcal{X}_1 \times \ldots \mathcal{X}_k \to (\mathcal{X}_1/G_1) \times \ldots \times (\mathcal{X}_k/G_k)$ is quotient map.

Proof. First, we show that $\pi_1 \times \ldots \times \pi_k$ is a quotient map. This is because 1. the quotient map of any continuous group action is an open map, so each π_i is an open map, 2. the product of open maps is an open map, so $\pi_1 \times \ldots \times \pi_k$ is an open map and 3. a continuous surjective open map is a quotient map, so $\pi_1 \times \ldots \times \pi_k$, which is continuous and surjective, is a quotient map.

Now, we need only apply the theorem of uniqueness of quotient spaces to show 39 (see e.g. Lee (2013), Theorem A.31). Letting $q : \mathcal{X}_1 \times \ldots \times \mathcal{X}_k \to (\mathcal{X}_1 \times \ldots \times \mathcal{X}_k)/(G_1 \times \ldots \times G_k)$ denote the quotient map for this space, it is easily seen that $q(x_1, \ldots, x_k) = q(y_1 \ldots, y_k)$ if and only if $\pi_1 \times \ldots \times \pi_k(x_1, \ldots, x_k) = \pi_1 \times \ldots \times \pi_k(y_1, \ldots, y_k)$, since either of these is true if and only if there exist $g_i \in G_i$ such that $x_i = g_i y_i$ for each i. Thus, we have an isomorphism of these quotient spaces. □

The following lemma shows that quotients of compact spaces are also compact, which is useful for universal approximation on quotient spaces.

Lemma 4 (Compactness of quotients of compact spaces)**.** *Let $\mathcal{X}$ be a compact space. Then the quotient space $\mathcal{X}/G$ is compact.*

Proof. Denoting the quotient map by $\pi : \mathcal{X} \to \mathcal{X}/G$ and letting $\{U_\alpha\}_\alpha$ be an open cover of $\mathcal{X}/G$, we have that $\{\pi^{-1}(U_\alpha)\}_\alpha$ is an open cover of $\mathcal{X}$. By compactness of $\mathcal{X}$, we can choose a finite subcover $\{\pi^{-1}(U_{\alpha_i})\}_{i=1,\ldots,n}$. Then $\{\pi(\pi^{-1}(U_{\alpha_i}))\}_{i=1,\ldots,n} = \{U_{\alpha_i}\}_{i=1,\ldots,n}$ by surjectivity, and $\{U_{\alpha_i}\}_{i=1,\ldots,n}$ is thus an open cover of $\mathcal{X}/G$. □

The Whitney embedding theorem gives a nice condition that we apply to show that the quotient spaces $\mathcal{X}/G$ that we deal with embed into Euclidean space. It says that when $\mathcal{X}/G$ is a smooth manifold, then it can be embedded into a Euclidean space of double the dimension of the manifold. The proof is outside the scope of this paper.

Lemma 5 (Whitney Embedding Theorem (Whitney, 1944))**.** *Every smooth manifold $\mathcal{M}$ of dimension $n > 0$ can be smoothly embedded in $\mathbb{R}^{2n}$.*

Finally, we give a lemma that helps prove universal approximation results. It says that if functions f that we want to approximate can be written as compositions $f = f_L \circ \ldots \circ f_1$, then it suffices to universally approximate each f_i and compose the results to universally approximate the f. This is especially useful for proving universality of neural networks, as we may use some layers to approximate each f_i, then compose these layers to approximate the target function f.

Lemma 6 (Layer-wise universality implies universality)**.** *Let $\mathcal{Z} \subseteq \mathbb{R}^{d_0}$ be a compact domain, let $\mathcal{F}_1, \ldots, \mathcal{F}_L$ be families of continuous functions where $\mathcal{F}_i$ consists of functions from $\mathbb{R}^{d_{i-1}} \to \mathbb{R}^{d_i}$ for some $d_1, \ldots, d_L$. Let $\mathcal{F}$ be the family of functions $\{f_L \circ \ldots f_1 : \mathcal{Z} \to \mathbb{R}^{d_L}, f_i \in \mathcal{F}_i\}$ that are compositions of functions $f_i \in \mathcal{F}_i$.*

For each i, let Φ_i be a family of continuous functions that universally approximates $\mathcal{F}_i$. Then the family of compositions $\Phi = \{\phi_L \circ \ldots \circ \phi_1 : \phi_i \in \Phi_i\}$ universally approximates $\mathcal{F}$.

Proof. Let $f = f_L \circ \ldots \circ f_1 \in \mathcal{F}$. Let $\tilde{\mathcal{Z}}_1 = \mathcal{Z}$, and then for $i \geq 2$ let $\tilde{\mathcal{Z}}_i = f_{i-1}(\tilde{\mathcal{Z}}_{i-1})$. Then each $\tilde{\mathcal{Z}}_i$ is compact by continuity of the f_i. For $1 \leq i < L$, let $\mathcal{Z}_i = \tilde{\mathcal{Z}}_i$, and for $i = L$ let $\mathcal{Z}_L$ be a compact set containing $\tilde{\mathcal{Z}}_L$ such that every ball of radius one centered at a point in $\tilde{\mathcal{Z}}_L$ is still contained in $\mathcal{Z}_L$.

Let $\epsilon > 0$. We will show that there is a $\phi \in \Phi$ such that $\|f - \phi\|_\infty < \epsilon$ by induction on L. This holds trivially for $L = 1$, as then $\Phi = \Phi_1$.

Now, let $L \geq 2$, and suppose it holds for $L - 1$. By universality of Φ_L, we can choose a $\phi_L : \mathcal{Z}_L \to \mathbb{R}^{d_L} \in \Phi_L$ such that $\|\phi_L - f_L\|_\infty < \epsilon/2$. As ϕ_L is continuous on a compact domain, it is also uniformly continuous, so we can choose a $\tilde{\delta} > 0$ such that $\|y - z\|_2 < \tilde{\delta} \implies \|\phi_L(y) - \phi_L(z)\|_2 < \epsilon/2$.

Let $\delta = \min(\tilde{\delta}, 1)$. By induction, we can choose $\phi_{L-1} \circ \ldots \circ \phi_1, \phi_i \in \Phi_i$ such that

$$\|\phi_{L-1} \circ \ldots \circ \phi_1 - f_{L-1} \circ \ldots \circ f_1\|_\infty < \delta. \tag{40}$$

Note that $\phi_{L-1} \circ \ldots \circ \phi_1(\mathcal{Z}) \subseteq \mathcal{Z}_L$, because for each $x \in \mathcal{Z}$, $\phi_{L-1} \circ \ldots \circ \phi_1(x)$ is within $\delta \leq 1$ Euclidean distance to $f_{L-1} \circ \ldots \circ f_1(x) \in \tilde{\mathcal{Z}}_L$, so it is contained in $\mathcal{Z}_L$ by construction. Thus, we may define $\phi = \phi_L \circ \ldots \circ \phi_1 : \mathcal{Z} \to \mathbb{R}^{d_L}$, and compute that

$$\|\phi - f\|_\infty \leq \|\phi - \phi_L \circ f_{L-1} \circ \ldots \circ f_1\|_\infty + \|\phi_L \circ f_{L-1} \circ \ldots \circ f_1 - f\|_\infty \tag{41}$$

$$< \|\phi - \phi_L \circ f_{L-1} \circ \ldots \circ f_1\|_\infty + \epsilon/2, \tag{42}$$

since $\|\phi_L - f_L\|_\infty < \epsilon/2$. To bound this other term, let $x \in \mathcal{Z}$, and for $y = \phi_{L-1} \circ \ldots \circ \phi_1(x)$ and $z = f_{L-1} \circ \ldots \circ f_1(x)$, we know that $\|y - z\|_2 < \delta$, so $\|\phi_L(y) - \phi_L(z)\|_2 < \epsilon/2$ by uniform continuity. As this holds for all x, we have $\|\phi - \phi_L \circ f_{L-1} \circ \ldots \circ f_1\|_\infty \leq \epsilon/2$, so $\|\phi - f\|_\infty < \epsilon$ and we are done. □