AdS-GNN - a Conformally Equivariant Graph Neural Network

Maksim Zhdanov*
AMLab, University of Amsterdam
m.zhdanov@uva.nl

Nabil Iqbal*
Department of Mathematical Sciences, Durham University
AMLab, University of Amsterdam
nabil.iqbal@durham.ac.uk

Erik Bekkers
AMLab, University of Amsterdam
e.j.bekkers@uva.nl

Patrick Forré
AMLab, AI4Science Lab, University of Amsterdam
p.d.forre@uva.nl

Abstract

Conformal symmetries, i.e. coordinate transformations that preserve angles, play a key role in many fields, including physics, mathematics, computer vision and (geometric) machine learning. Here we build a neural network that is equivariant under general conformal transformations. To achieve this, we lift data from flat Euclidean space to Anti de Sitter (AdS) space. This allows us to exploit a known correspondence between conformal transformations of flat space and isometric transformations on the Anti de Sitter space. We then build upon the fact that such isometric transformations have been extensively studied on general geometries in the geometric deep learning literature. In particular, we then employ message-passing layers conditioned on the proper distance, yielding a computationally efficient framework. We validate our model on point cloud classification (SuperPixel MNIST) and semantic segmentation (PascalVOC-SP).

Code available at: https://github.com/maxxxzdn/adsgnn

1 Introduction

The notion of *symmetry* is a key tool both in our understanding of nature and for the construction of machine learning systems that perceive nature. The construction of *equivariant* neural network architectures that encode specific symmetries has powerful advantages both conceptual and computational. In particular, much work has been dedicated to building networks that are equivariant under symmetries such as rotations and translations.

In this work, we will study the symmetry group of *conformal transformations* i.e. the set of transformations on $\mathbb{R}^d$ that preserve angles. This includes translations, rotations, reflections, scalings and so called special conformal transformations. Importantly, this includes scale transformations – i.e. rigid rescalings of the whole system – as a subgroup. Conformal and scale invariance play a central role in many diverse fields. To give some examples: biological visual systems seem to exhibit insensitivity to scale (Logothetis et al. (1995); Han et al. (2020), systems undergoing a second-order phase transition generally exhibit conformal symmetries at the critical point (Cardy (1996))[1], and diverse applications exist in computational geometry and computer vision (see e.g. Sharon & Mumford (2006); Lei et al. (2023))

*Equal contribution.

[1]Indeed in physics most systems exhibiting scale invariance also exhibit conformal invariance "for free" (Polchinski (1988), Nakayama (2015)).

It is reasonable to believe that a neural network that is equivariant under conformal transformations will be naturally insensitive to scale, instead focusing on robust properties of shape and form, lending to many possible applications. In this work, we construct such a conformally equivariant network. Our approach acts on point clouds and lifts the data into an auxiliary higher dimensional space called Anti de Sitter (AdS) space. As outlined below, this approach is inspired by ideas in conformal field theory in theoretical physics.

Previous work: Construction of equivariant neural networks under general symmetry groups started with Cohen & Welling (2016) and is now a rich and well-studied field, see e.g. Weiler et al. (2023) for a review and the treatment of the general case of isometric transformations on general Riemannian manifolds. An extension to semi-Riemannian manifolds can be found in Zhdanov et al. (2024). See Vadgama et al. (2025) for a study of the benefits of equivariance on point clouds. These existing methods do not address the full conformal group which we study here. Some literature exists on neural networks with scale equivariance, see. e.g Bekkers (2020); Sosnovik et al. (2019). Our usage of AdS space may be thought of as a kind of scale space (see e.g. Witkin (1987); Worrall & Welling (2019)). A difference in our case is that AdS has a somewhat more rigid structure in that it enforces equivariance under the larger group of conformal transformations, and not only scale transformations. Recent work in the physics literature includes Halverson et al. (2024) which describes how to use neural networks to obtain a conformal field theory.

2 AdS-GNN

We briefly review conformal symmetry and the geometry of AdS space before describing our approach.

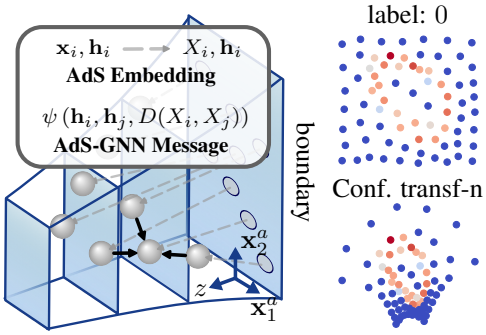


Figure 1: AdS-GNN lifts points from Euclidean space to Anti de Sitter space and computes message passing conditioned on the proper distance.

2.1 Brief description of conformal symmetry and AdS_{d+1} isometries

A *global conformal transformation* of the Euclidean space $\mathbb{R}^d$ is an injective smooth map $\varphi : \mathbb{R}^d \backslash \{x_\varphi\} \to \mathbb{R}^d, x \mapsto \varphi(x)$, defined on $\mathbb{R}^d$ except on a possible point x_φ [2] such that angles are always preserved, i.e. for all $x \in \mathbb{R}^d \setminus \{x_\varphi\}$ and $v_1, v_2 \in \mathbb{R}^d \setminus \{0\}$ we require:

$$\frac{\langle \varphi'(x)v_1, \varphi'(x)v_2 \rangle}{\|\varphi'(x)v_1\| \cdot \|\varphi'(x)v_2\|} =: \cos(\measuredangle(\varphi'(x)v_1, \varphi'(x)v_2)) \stackrel{!}{=} \cos(\measuredangle(v_1, v_2)) := \frac{\langle v_1, v_2 \rangle}{\|v_1\| \cdot \|v_2\|}, \qquad (1)$$

where $\langle ., . \rangle$ and $\|.\|$ denote the standard Euclidean scalar product and norm, resp., and $\varphi'(x)$ the Jacobian matrix of φ at x. Note, that, in contrast to *isometric transformations*, we do not require that the distances/norms are preserved. The *group of all global conformal transformations* of $\mathbb{R}^d$ is denoted by $\mathrm{Conf}_{\mathrm{g}}(\mathbb{R}^d)$. We define the *(restricted) group of conformal transformations* $\mathrm{Conf}(\mathbb{R}^d)$ here as the connected component of the identity of $\mathrm{Conf}_{\mathrm{g}}(\mathbb{R}^d)$.[3] The conformal group $\mathrm{Conf}(\mathbb{R}^d)$ then acts through the group action of $SO(d+1,1)$ (mod $\{\pm 1\}$, resp.), whose viewpoint, in terms of separated parameters, we will explain in the following.

[2] By allowing φ to not be defined on a certain point $x_\varphi \in \mathbb{R}^d$, we effectively allow φ to map x_φ to the "points at infinity" ∞ in the conformal compactification $\mathbb{S}^d$ of $\mathbb{R}^d$. In fact, every *global* conformal transformation of $\mathbb{R}^d$ uniquely extends to an angle-preserving diffeomorphism of $\mathbb{S}^d$ for $d \geq 2$, see Schottenloher (2008) Thm. 2.9 and 2.11. This is why we introduce the definition with H in this way.

[3] It was shown in Schottenloher (2008) Thm. 2.9 and 2.11 that for $d \geq 2$ the group of *global* conformal transformations $\mathrm{Conf}_{\mathrm{g}}(\mathbb{R}^d)$ is isomorphic to $O(d+1,1)/\{\pm 1\}$, which is isomorphic to $SO(d+1,1)$ if d is odd. Be aware of the unusual convention for $SO(p,q)$ in that specific reference, which we do not follow here. Also note that in Amir-Moéz (1967) it was shown that the group of all *linear* conformal transformations of $\mathbb{R}^d$ is the group $\mathbb{R}_{>0} \times O(d)$, of scaled orthogonal transformations.

A general element G of the group can be written in terms of parameters $(\lambda, t, b, M) \in \mathbb{R}_{>0} \times \mathbb{R}^d \times \mathbb{R}^d \times SO(d)$ and acts on a point $x \in \mathbb{R}^d$ through the composition of maps defined by:

$$\begin{array}{cccc} x' = x + t & x' = Mx & x' = \lambda x & \dfrac{x'}{\|x'\|^2} = \dfrac{x}{\|x\|^2} - b, \\ \text{translations} & \text{rotations} & \text{scalings} & \text{special conformal transformations} \end{array} \tag{2}$$

resulting in a transformed point $x' = Gx$. These $(d+1)(d+2)/2$ parameters assemble into an element of $SO(d+1, 1)$, see e.g. Di Francesco et al. (1997) for a review.

Given a typical presentation of input data, it is not straightforward to express it in a fashion that transforms simply under representations of the conformal group. To organize the data, we will instead lift it from $\mathbb{R}^d$ to an auxiliary space with *one higher dimension*, i.e. *Anti de Sitter space* AdS_{d+1}. The conformal group $\mathrm{Conf}(\mathbb{R}^d)$ of $\mathbb{R}^d$ acts naturally as the *isometry group* $\mathrm{Isom}(\mathrm{AdS}_{d+1})$ on AdS_{d+1}. The fact that local operations in the interior of AdS_{d+1} results in conformally invariant operations on $\mathbb{R}^d$ is very well known in the context of the AdS/CFT correspondence in quantum gravity Maldacena (1999). Here we will use some of the well-studied kinematics of that correspondence as a convenient tool to build convolutional kernels on AdS_{d+1}. Indeed, a general framework for constructing convolutional layers that are equivariant under isometry groups of any pseudo-Riemannian manifold was given in Weiler et al. (2023); Zhdanov et al. (2024), and our work can be viewed as building on a special case of that.

More precisely, denoting a point in AdS_{d+1} as $X = (X^1, \ldots, X^{d+1}) = (x^1, \ldots, x^d, z) = (x, z)$ with z the extra dimension, the Riemannian metric on AdS_{d+1} is given by:

$$ds^2 = \sum_{\mu,\nu=1}^{d+1} g_{\mu\nu}(X) dX^\mu dX^\nu =: \frac{1}{z^2}\left(\sum_{a=1}^{d}(dx^a)^2 + dz^2\right) \tag{3}$$

In the Supplementary Material, we review how this geometry can be understood as a hyperboloid embedded in $\mathbb{R}^{d+1,1}$ with a natural action of $SO(d+1, 1)$. Here we just note that the isometry group acts as $X' = GX$:

$$(x', z') = (x + t, z) \qquad (x', z') = (Mx, z) \qquad (x', z') = (\lambda x, \lambda z), \tag{4}$$

and

$$\left(\frac{x'}{\|x'\|^2 + z'^2}, \frac{z'}{\|x'\|^2 + z'^2}\right) = \left(\frac{x}{\|x\|^2 + z^2} + b, \frac{z}{\|x\|^2 + z^2}\right) \tag{5}$$

Note that the manifold (3) has a d-dimensional boundary at $z = 0$. This boundary is mapped to itself under the isometries. Furthermore, the isometry group acts on the boundary points $(x^a, z = 0)$ precisely as in (2). Thus one should imagine that conformal data on $\mathbb{R}^d$ "lives on the boundary of AdS_{d+1}". In what follows a key role will be played by the $SO(d+1, 1)$ invariant proper distance between $D(X, X')$ between two points in AdS, which is given by

$$\cosh D(X, X') = \frac{z^2 + z'^2 + \sum_{a=1}^{d}(x^a - x'^a)^2}{2zz'} \tag{6}$$

We will now describe a way to extend the data from the boundary into the bulk of AdS_{d+1}.

2.2 Embedding points in AdS

Consider a point cloud of N points in $\mathbb{R}^d$, $\{x_i\}$, $i \in \{1, \cdots, N\}$, We would like to lift this data into the bulk of AdS_{d+1} in a manner that preserves the symmetries.

A first attempt from the correspondence of symmetries shown in (4) is to simply embed each of the points directly into the boundary $z = 0$, i.e. with $X_i^\mu = (x_i^a, z = 0)$. However the metric (3) has a singularity at $z = 0$ – e.g. from (6) note that each point at $z = 0$ is at infinite proper AdS distance from any points at $z > 0$ – and thus such an attempt will require us to pick a regulating value of z. Using a fixed constant will explicitly break the symmetries. For each x_i, we instead pick a z_i obtained from the distance to its neighbours in a manner that preserves scale symmetry, as outlined in Algorithm 1 and explained further in the Supplementary Material.

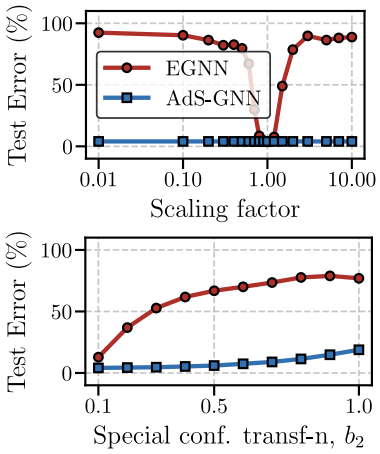
Table 1: Classification error on SuperPixel MNIST. We augment the test set with random rotations and scaling. ✗ indicates random-guess performance.

Model	**Error rate, %**		
	non-augmented	rotated	rotated+scaled
MONET	8.89	✗	✗
SplineCNN	4.78	✗	✗
GCGP	4.2	✗	✗
GAT	3.81	✗	✗
PNCNN	1.24 ± 0.12	✗	✗
PΘNITA	1.17 ± 0.11	1.17	✗
EGNN	4.17 ± 0.45	4.17	✗
AdS-GNN (**Ours**)	4.09 ± 0.27	4.09	4.09

Table 2: Classification error on PascalVOC-SP.

Model	EGNN	AdS-GNN
Test F1 ↑	27.80 ± 0.74	28.07 ± 0.57

Figure 2: Test error on augmented data, SuperPixel MNIST.



Intuitively, the z coordinate corresponds to the *length scale* of the degrees of freedom we are considering[4]. Our choice above amounts to saying that the appropriate length scale for a point x_i is related to its (appropriately averaged) distance from its neighbours. This exactly preserves scale invariance, but it gently breaks special conformal transformations. This is expected on physical grounds, as generally any choice of regulator necessarily breaks conformal invariance (Cardy (1996)). In our experiments, we thus check generalization under special conformal transformations empirically in Fig. 2 for a special conformal transformation as in (2) parametrized by $b = (0, b_2)$ and verify that the breaking is mild.

Algorithm 1 AdS Embedding

Require: $X = \{\mathbf{x}_i\}_{i=1}^N \subset \mathbb{R}^d$, $k_{\text{lift}} \in \mathbb{N}$, $z_0 \in \mathbb{R}$
1: **for** each point $i \in \{1, \ldots, N\}$ **do**
2: $\quad z_i \leftarrow z_0$
3: $\quad \text{neighbors}_i \leftarrow \text{KNN}(x_i, X, k_{\text{lift}})$
4: $\quad (\hat{x}_i, \hat{z}_i) \leftarrow \text{ComputeAdSCoM}(\text{neighbors}_i)$
5: **end for**
6: **return** $\{(x_i, \hat{z}_i)\}_{i=1}^N \subset AdS_{d+1}$

2.3 Message passing

Given this set of points $\{X_i\}$ in AdS, we now operate on it using a graph neural network. To orient ourselves, we recall first an earlier model, that of E(n) Equivariant Graph Neural Networks (EGNNs) Satorras et al. (2021); Liu et al. (2024). These are graph neural networks that are equivariant to rotations, translations, reflections and permutations. The input to the model is a graph $G = (\mathcal{V}, \mathcal{E})$ which is embedded into the Euclidean space $\mathbb{R}^D$. We denote the latent d-dimensional feature vector of v_i as $\mathbf{h}_i$. The l-th layer of EGNN is defined as

$$\mathbf{m}_{ij} = \psi_e(\mathbf{h}_i^l, \mathbf{h}_j^l, ||\mathbf{p}_i - \mathbf{p}_j||_2), \qquad \text{EGNN message} \tag{7}$$
$$\mathbf{h}_i^{l+1} = \psi_h(\mathbf{h}_i^l, \mathbf{m}_i), \quad \mathbf{m}_i = \sum_{j \in \mathcal{N}(i)} \mathbf{m}_{ij}, \qquad \text{aggregate} + \text{update}$$

where $\mathcal{N}(i)$ represents the set of neighbours of node v_i, ψ_e, ψ_h are message and update MLPs.

We thus define **AdS-GNN**: we adopt the model to operate on AdS, where a graph G is embedded. If edges $\mathcal{E}$ are not provided, we induce connectivity with k_{con} nearest neighbours using the proper distance (6). In the message function (7), we also use the proper distance instead of the Euclidean:

$$\mathbf{m}_{ij} = \psi_e(\mathbf{h}_i^l, \mathbf{h}_j^l, D(X_i, X_j)), \qquad \text{AdS-GNN message} \tag{8}$$

which yields an efficient conformal group equivariant GNN without substantial computational overhead compared to its Euclidean counterpart. Note that by conditioning on AdS proper distance, we introduce a notion of locality both in ordinary space and in scale (as represented by the z coordinate). Though the embedding of the point cloud mildly breaks special conformal transformations, the graph neural network itself is exactly invariant under all of $\text{Isom}(\text{AdS}_{d+1})$.

[4]This is familiar from the physics of the AdS/CFT correspondence, where it is well-understood that the infrared physics lives deeper in the bulk Susskind & Witten (1998).

3 EXPERIMENTAL RESULTS

SuperPixel MNIST We benchmark AdS-GNN on the super-pixel MNIST dataset Monti et al. (2017), which consists of 2D point clouds of MNIST digits segmented into 75 superpixels. Results are given in Table 1. Even though AdS-GNN performs on par with its roto-equivariant counterpart, it still falls behind PΘNITA as it is unable to handle orientation and relies on invariant descriptors. We also study the response of a model to various augmentations (see Fig. 2). As expected, AdS-GNN is precisely scale-invariant. For special conformal transformations, there is a small breaking of symmetry arising from the uplift, which we relate to the tiny size of the domain (75 points).

PascalVOC-SP We also compare AdS-GNN to EGNN on the LRGB data Dwivedi et al. (2022), see Table 2. The difference in performance is statistically insignificant, which indicates that conformal equivariance does not constrain the model significantly and still allows for high expressivity.

4 CONCLUSION

In this paper, we introduced AdS-GNN - a neural network that is equivariant with respect to conformal transformations. In future work, we aim to implement a more expressive message function inspired by PΘNITA Bekkers et al. (2024) and gauge-equivariant networks Basu et al. (2022). We also anticipate applications to critical phenomena in conformal field theory as well as tasks in computer vision that require robust characterization of shape and form.

ACKNOWLEDGMENTS

MZ was supported by Microsoft Research AI4Science. This work was supported by a grant from the Simons Foundation (PD-Pivot Fellow-00004147, NI and EB). NI is supported in part by the STFC under grant number ST/T000708/1.

REFERENCES

Ofer Aharony, Steven S. Gubser, Juan Martin Maldacena, Hirosi Ooguri, and Yaron Oz. Large N field theories, string theory and gravity. *Phys. Rept.*, 323:183–386, 2000. doi: 10.1016/S0370-1573(99)00083-6.

Ali R Amir-Moéz. Conformal Linear Transformations. *Mathematics Magazine*, 40(5):268–270, 1967.

Sourya Basu, Jose Gallego-Posada, Francesco Viganò, James Rowbottom, and Taco Cohen. Equivariant mesh attention networks. *TMLR*, 2022.

Erik J Bekkers. B-spline cnns on lie groups. In *International Conference on Learning Representations*, 2020. URL `https://openreview.net/forum?id=H1gBhkBFDH`.

Erik J. Bekkers, Sharvaree P. Vadgama, Rob Hesselink, Putri A. van der Linden, and David W. Romero. Fast, expressive se(n) equivariant networks through weight-sharing in position-orientation space. In *ICLR*, 2024.

John Cardy. *Scaling and renormalization in statistical physics*, volume 5. Cambridge university press, 1996.

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

P. Di Francesco, P. Mathieu, and D. Senechal. *Conformal Field Theory*. Graduate Texts in Contemporary Physics. Springer-Verlag, New York, 1997. ISBN 978-0-387-94785-3, 978-1-4612-7475-9. doi: 10.1007/978-1-4612-2256-9.

Vijay Prakash Dwivedi, Ladislav Rampásek, Michael Galkin, Ali Parviz, Guy Wolf, Anh Tuan Luu, and Dominique Beaini. Long range graph benchmark. In *NeurIPS*, 2022.

Matthias Fey, Jan Eric Lenssen, Frank Weichert, and Heinrich Müller. Splinecnn: Fast geometric deep learning with continuous b-spline kernels. In *CVPR*, 2018.

Marc Anton Finzi, Roberto Bondesan, and Max Welling. Probabilistic numeric convolutional neural networks. In *ICLR*, 2021.

GA Galperin. A concept of the mass center of a system of material points in the constant curvature spaces. *Communications in Mathematical Physics*, 154:63–84, 1993.

James Halverson, Joydeep Naskar, and Jiahua Tian. Conformal Fields from Neural Networks. 9 2024.

Yena Han, Gemma Roig, Gad Geiger, and Tomaso Poggio. Scale and translation-invariance for novel objects in human vision. *Scientific reports*, 10(1):1411, 2020.

Na Lei, Feng Luo, Shing-Tung Yau, and Xianfeng Gu. Computational conformal geometric methods for vision. In *Handbook of Mathematical Models and Algorithms in Computer Vision and Imaging: Mathematical Imaging and Vision*, pp. 1739–1790. Springer, 2023.

Cong Liu, David Ruhe, Floor Eijkelboom, and Patrick Forré. Clifford Group Equivariant Simplicial Message Passing Networks. *International Conference on Learning Representations (ICLR)*, 2024.

Nikos K Logothetis, Jon Pauls, and Tomaso Poggio. Shape representation in the inferior temporal cortex of monkeys. *Current biology*, 5(5):552–563, 1995.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Juan Martin Maldacena. The Large N limit of superconformal field theories and supergravity. *Int. J. Theor. Phys.*, 38:1113–1133, 1999. doi: 10.1023/A:1026654312961. [Adv. Theor. Math. Phys.2,231(1998)].

Federico Monti, Davide Boscaini, Jonathan Masci, Emanuele Rodolà, Jan Svoboda, and Michael M. Bronstein. Geometric deep learning on graphs and manifolds using mixture model cnns. In *CVPR*, 2017.

Yu Nakayama. Scale invariance vs conformal invariance. *Phys. Rept.*, 569:1–93, 2015. doi: 10.1016/j.physrep.2014.12.003.

Joseph Polchinski. Scale and Conformal Invariance in Quantum Field Theory. *Nucl. Phys. B*, 303: 226–236, 1988. doi: 10.1016/0550-3213(88)90179-4.

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In Marina Meila and Tong Zhang (eds.), *ICML*, 2021.

Martin Schottenloher. *A Mathematical Introduction to Conformal Field Theory*. Springer, 2008.

Eitan Sharon and David Mumford. 2d-shape analysis using conformal mapping. *International Journal of Computer Vision*, 70(1):55–75, 2006.

Ivan Sosnovik, Michał Szmaja, and Arnold Smeulders. Scale-equivariant steerable networks. *arXiv preprint arXiv:1910.11093*, 2019.

Leonard Susskind and Edward Witten. The Holographic bound in anti-de Sitter space. 5 1998.

Sharvaree Vadgama, Mohammad Mohaiminul Islam, Domas Buracus, Christian Shewmake, and Erik Bekkers. On the utility of equivariance and symmetry breaking in deep learning architectures on point clouds, 2025. URL `https://arxiv.org/abs/2501.01999`.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *ICLR*, 2018.

Ian Walker and Ben Glocker. Graph convolutional gaussian processes. In *ICML*, 2019.

Maurice Weiler, Patrick Forré, Erik Verlinde, and Max Welling. *Equivariant and Coordinate Independent Convolutional Networks*. 2023. URL `https://maurice-weiler.gitlab.io/cnn_book/EquivariantAndCoordinateIndependentCNNs.pdf`.

Andrew P Witkin. Scale-space filtering. In *Readings in computer vision*, pp. 329–332. Elsevier, 1987.

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

Maksim Zhdanov, David Ruhe, Maurice Weiler, Ana Lucic, Johannes Brandstetter, and Patrick Forré. Clifford-steerable convolutional neural networks. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=XTglHJjzQI.

A AdS_{d+1} PRIMER

Here for completeness we review some facts about the geometry of Euclidean AdS_{d+1}, i.e. hyperbolic space in $d+1$ dimensions.

A.1 EMBEDDING IN $\mathbb{R}^{d+1,1}$

We first review a construction of AdS_{d+1} as a hyperboloid embedded in the larger-dimensional flat space $\mathbb{R}^{d+1,1}$. This material is standard, see e.g. Aharony et al. (2000) for a review. Consider $\mathbb{R}^{d+1,1}$ with coordinates $Y^A = (Y^0, Y^a, Y^{d+1})$ where in our notation a, b indices run over $\mathbb{R}^d$, μ, ν run over AdS_{d+1}, and A, B run over $\mathbb{R}^{d+1,1}$. We write the flat metric on $\mathbb{R}^{d+1,1}$ as:

$$ds^2 = \sum_{A,B} \eta_{AB} dx^A dx^B = -(dY^0)^2 + \sum_{i=1}^{d+1} (dY^i)^2 \,. \tag{9}$$

Consider also the submanifold $H \subset \mathbb{R}^{d+1,1}$ defined as

$$(Y^0)^2 - \sum_{i=1}^{d+1} (Y^i)^2 = 1 \tag{10}$$

This submanifold has two connected components with positive and negative Y^0 respectively. Each of these components is an isometric version of the Euclidean AdS_{d+1}. We will stick to the one with $Y^0 > 0$. To put coordinates $X^\mu = (x^a, z)$ on it we can write:

$$Y^0 = +\frac{z}{2}\left(1 + \frac{1}{z^2}(1 + x^a x^a)\right) > 0 \tag{11}$$

$$Y^a = \frac{1}{z} x^a \tag{12}$$

$$Y^{d+1} = \frac{z}{2}\left(1 - \frac{1}{z^2}(1 - x^a x^a)\right) \tag{13}$$

The action of $[\Lambda] \in O(d+1,1)/\{\pm I\}$ then works as follows:

$$[\Lambda].Y := \mathrm{sign}((\Lambda Y)^0) \cdot (\Lambda Y),$$

where ΛY is just vector matrix multiplication and where the scalar multiplication with $\mathrm{sign}((\Lambda Y)^0$ corrects the sign of the 0-th entry (the component of (-1)-signature). Together we get:

$$(([\Lambda].Y)^0)^2 - \sum_{i=1}^{d+1} (([\Lambda].Y)^i)^2 = 1, \qquad ([\Lambda].Y)^0 > 0, \tag{14}$$

and thus: $[\Lambda].Y \in \mathrm{AdS}_{d+1}$. This gives us a well-defined group action of $O(d+1,1)/\{\pm I\}$ on AdS_{d+1}, and, in particular, a well-defined group action of $\mathrm{Conf}(\mathbb{R}^d)$ on AdS_{d+1}.

We now compute the induced metric on the hyperboloid

$$g_{MN}(X) = \sum_{A,B} \frac{\partial Y^A}{\partial X^M} \frac{\partial Y^B}{\partial X^N} \eta_{AB} \tag{15}$$

which turns out to be

$$ds^2 = \frac{1}{z^2}\left(dz^2 + \sum_{a=1}^{d}(dx^a)^2\right) \tag{16}$$

as expected from equation 3.

A.2 THE CENTER OF MASS OF A SET OF POINTS ON AdS

We will require an expression for the "center of mass" $C(\{X_i\})$ of a set of points on AdS. This problem was solved in Galperin (1993); the basic idea is to view the hyperboloid as a submanifold of $\mathbb{R}^{d+1,1}$ as above, use additivity properties there to find a vector, and then find the intersection of the ray in the direction of that vector with the hyperboloid.

In practice, this is quite simple to implement. Denote the center of mass by $\bar{Y}^A$, and the set of N points for which we want the centroid by (x_i^a, z_i). We would like to find the analogous coordinates for the centroid $(\bar{x}^a, \bar{z})$.

We have that

$$\bar{Y}^0 \equiv \frac{\bar{z}}{2}\left(1 + \frac{1}{\bar{z}^2}(1 + \sum_a \bar{x}^a\bar{x}^a)\right) = \frac{1}{\mathcal{N}N}\sum_i \frac{z_i}{2}\left(1 + \frac{1}{z_i^2}(1 + \sum_a x_i^a x_i^a)\right) \tag{17}$$

$$\bar{Y}^a \equiv \frac{\bar{x}^a}{\bar{z}} = \frac{1}{\mathcal{N}N}\sum_i \frac{x_i^a}{z_i} \tag{18}$$

$$\bar{Y}^{d+1} \equiv \frac{\bar{z}}{2}\left(1 - \frac{1}{\bar{z}^2}(1 - \sum_a \bar{x}^a\bar{x}^a)\right) = \frac{1}{\mathcal{N}N}\sum_i \frac{z_i}{2}\left(1 - \frac{1}{z_i^2}(1 - \sum_a x_i^a x_i^a)\right) \tag{19}$$

The first equality is the definition of the embedding, the second is the definition of the centroid from Galperin. Here $\mathcal{N}$ is a normalization constant which is picked to guarantee that

$$(\bar{Y}^0)^2 - \sum_a (\bar{Y}^a)^2 - (\bar{Y}^{d+1})^2 = 1 \tag{20}$$

So to find the centroid, the easiest thing to do is to compute the sums on the right hand side of the second equality, which thus determines the vector $\bar{Y}^A$ up to an overall scale $\mathcal{N}$; then we enforce the norm constraint above which lets us find $\mathcal{N}$ and thus fixes the vector $\bar{Y}^A$ completely. We then express the answer in useful coordinates by solving for $(\bar{z}, \bar{x}^a)$ through

$$\bar{z} = \frac{1}{\bar{Y}^0 - \bar{Y}^{d+1}}, \qquad \bar{x}^a = \frac{\bar{Y}^a}{\bar{Y}^0 - \bar{Y}^{d+1}}\ . \tag{21}$$

To get some intuition for the procedure, we study it in the case of two points $X_1 = (x_1^a, \epsilon)$ and $X_2 = (x_2^a, \epsilon)$ starting at the same value of the z coordinate. We find

$$C(X_1, X_2) = \left(\frac{1}{2}(x_1^a + x_2^a), \frac{1}{2}\sqrt{|x_1 - x_2|^2 + 4\epsilon^2}\right) \tag{22}$$

i.e. we simply take the average of the spatial coordinates and move inwards in z by an amount which depends on the separation between the two points in the spatial direction. In this case the center of mass is actually the midpoint of the geodesic that connects the two points.

A.3 AdS EMBEDDING EXPLANATION

The center of mass function is used for the AdS embedding of the point cloud shown in Algorithm 1. We briefly elaborate on the algorithm here.

As explained in the bulk text, to perform the embedding of the point cloud, we need a way to pick z_i for each point x_i in a manner that preserves the symmetries. We first embed each point into AdS using

$$X_i^\mu = (x_i^a, z = z_0) \tag{23}$$

with a small regulator z_0. For each point, we then compute the AdS center of mass $\hat{X} = (\hat{x}_i, \hat{z}_i)$ of its k_{lift} nearest neighbours using the approach above.

The geometry of AdS implies that the center of mass will generally be deeper inside than the original points; e.g. in this case it will have a finite z value which depends on the relative separation of the points, as shown explicitly in a 2-point example in (22). We then perform a final embedding of the point using this z value, i.e.

$$X_i^\mu = (x_i^a, \hat{z}_i) \tag{24}$$

B IMPLEMENTATION DETAILS

In every experiment, we use the AdamW optimize Loshchilov & Hutter (2019) with a learning rate 10^{-3}. Every model is trained on a single Nvidia RTX6000 GPU. Both EGNN and AdS-GNN are implemented in JAX. All experiments are run 5 times with different seeds. k_{con} is set to 16, k_{lift} to 5.

In the SuperPixel-MNIST experiment, the task is to predict a digit given a point cloud representation. We compare against MONET Monti et al. (2017), SplineCNN Fey et al. (2018), GCCP Walker & Glocker (2019), GAT Velickovic et al. (2018), PNCNN Finzi et al. (2021) and PΘNITA Bekkers et al. (2024). Every model is trained with batch size 128, baseline results are taken from Bekkers et al. (2024).

In the Pascal-VOC experiment, the task is to predict a semantic segmentation label for each superpixel node (total of 21 classes). Each graph is embedded in 2D Euclidean space, each node is associated with 12 scalar features. We used the batch size of 96.