DETECTING SYMMETRY-BREAKING IN MOLECULAR DATA DISTRIBUTIONS

Hannah Lawrence*[1] & Elyssa Hofgard*[1] & Yuxuan Chen[2] & Tess Smidt[1] & Robin Walters[2]
Massachusetts Institute of Technology[1], Northeastern University[2]
{hanlaw,ehofgard,tsmidt}@mit.edu,
chen.yuxuan7,r.walters}@northeastern.edu

ABSTRACT

Equivariant models, which enforce physical symmetries (such as rotations and permutations), have proven very successful at materials science tasks. The usual justification for this success is that symmetry transformations relate data samples, which improves generalization and data efficiency. However, this explanation assumes that transformed versions of a given molecule are highly likely under the data distribution. In this work, we develop a method for testing this assumption by measuring the amount of symmetry in a data distribution. Specifically, we propose a two-sample classifier test which distinguishes between the original dataset and its randomly augmented symmetrization. Unlike existing tests of group invariance, our method does not require defining an appropriate parametric test or kernel. We find that in commonly used materials science datasets such as QM9 and MD17, the orientations of molecules are highly non-uniform. Our findings suggest the success of equivariant models on these datasets may depend on other inductive biases, such as local equivariance. Moreover, non-equivariant models may be strongly benefiting from canonicalization of the molecules' orientations, an oft-overlooked part of the data generation process. As machine learning becomes increasingly important for materials discovery, it is essential to have tools to critically evaluate the assumptions underlying our data.

1 INTRODUCTION

Equivariant neural networks have had considerable success in materials science, from modeling molecular dynamics (Batzner et al., 2022) to predicting quantum mechanically accurate properties of molecules and crystals (Rackers et al., 2023; Fang et al., 2024; Liao et al., 2023). By integrating physical symmetries into the model architecture as group invariances, equivariant neural networks can often achieve superior generalization and data efficiency, and enjoy state-of-the-art performance on materials benchmarking datasets such as OC20, QM9, and MD17 (Liao et al., 2023; Batzner et al., 2022; Frey et al., 2023; Rackers et al., 2023; Owen et al., 2023), demonstrating their potential for use in automated materials screening, design, and property modeling.

The success of equivariant methods has typically been explained in terms of improved sample efficiency and generalizability, resulting from their ability to relate data x and transformed data gx (Cohen & Welling, 2016). For $g \in G$, a symmetry group, equivariant neural networks NN are constrained such that $\mathrm{NN}(gx) = g\,\mathrm{NN}(x)$, thus tying the predictions for x and gx. It is thus an *explicit assumption* for equivariant models that the ground truth function satisfies $f(gx) = gf(x)$. However, there is also an *implicit assumption* that transformed samples gx occur relatively uniformly in distribution, i.e. the input density $p(x) \approx p(gx)$. Theoretical results in equivariance almost always assume that x and gx are equally likely under the data distribution (Elesedy & Zaidi, 2021).

In this paper, we study *distributional symmetry breaking* (Wang et al., 2024c)—when a datapoint x and its transform gx are *not* equally likely under the data distribution[1]. Our ultimate goal, which we hope to address in future work, is to understand how distributional symmetry breaking affects the

[1]This differs from *functional symmetry breaking* (Wang et al., 2024c), where the mapping between inputs and outputs is not fully equivariant (e.g. during a phase transition in a material).

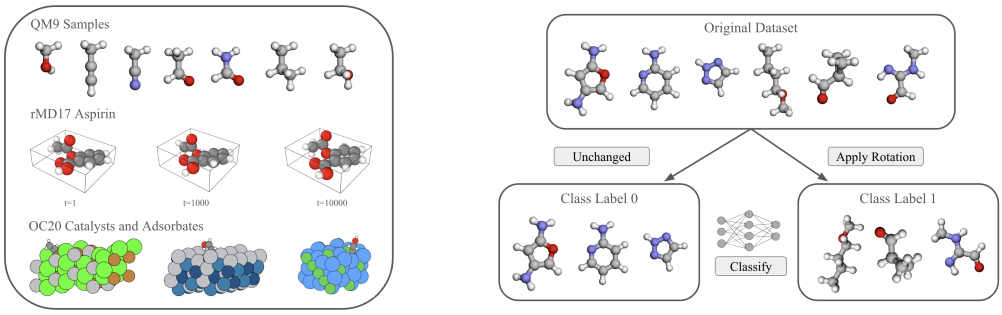

Figure 1: (left) Visualizations of unrotated samples from each dataset, with their canonicalization clearly visible. (right) A classifier test for symmetry: we generate a synthetic dataset to train a binary classifier, labeling non-rotated samples 0 and rotated samples 1.

performance of equivariant vs. non-equivariant models. Intuitively, equivariance can help performance by reducing the impact of sparsely sampled parts of the distribution, but it may also discard meaningful asymmetries across orbits $\{gx\}_{g\in G}$. This information may be inherent, such as the natural orientations of "6" and "9" (which are useful for distinguishing classes in MNIST), or user-defined, such as the conventions used to orient crystal structures along their highest symmetry axes. Indeed, Cohen et al. (2018) demonstrated that rotational equivariance only improves performance on MNIST when the dataset is artificially rotated. Complementing this finding, Shao et al. (2024) prove that no equivariant algorithm applied to distributionally asymmetric data can achieve optimal sample complexity.

We currently lack an effective numerical measure of the amount of symmetry breaking in a distribution, especially without domain knowledge of the distribution (Wang et al., 2024a; 2023; 2024c). Thus, we provide a measurement of the degree of symmetry breaking, which can place a function on the spectrum between a symmetrized distribution on one side, and fully canonicalized—where only a single sample x in each orbit $\{gx\}_{g\in G}$ is in distribution —on the other. We hope this metric will prove useful for (1) diagnosing why (and when) equivariant methods provide an advantage on existing tasks, and (2) aiding model selection on new datasets.

We propose the use of a two-sample classifier test (Lopez-Paz & Oquab, 2017), in which a binary classifier is trained to distinguish between samples from p_X and $\bar{p}_X$. The accuracy of this classifier on a held-out test set is a natural, *interpretable* measure of distance between p_X and $\bar{p}_X$, which (1) could allow for future interpretability methods (applied to the classifier itself) and (2) sidesteps the arbitrary kernel selection, offloading it to the less impactful choice of architecture.

We apply both (1) the classifier method and (2) adaptations of kernel MMD for point cloud kernels to benchmark datasets: QM9 (Wu et al., 2017), revised MD17 (Christensen & von Lilienfeld, 2020), and OC20 (Chanussot* et al., 2021). We find that all three datasets are highly non-uniform under 3D rotations. Since these benchmarks are a cornerstone of fundamental AI methods development for materials, understanding their biases is crucial for advancing automated materials discovery. As many ML practitioners may not be familiar with the data generation processes underlying materials science datasets, this serves as an interpretable tool to measure distributional symmetry-breaking.

2 Proposed Metric

Consider datapoints $x \in \mathcal{X}$ drawn from a distribution p_X, acted on by a compact group G. We assume that there is a ground truth labeling function $f : \mathcal{X} \to \mathcal{Y}$ that is equivariant, i.e. $f(gx) = gf(x)$. We do *not* assume that $p_X(x) = p_X(gx)$; instead, we wish to quantify the degree to which p_X breaks distributional symmetry by failing to satisfy this equality. To this end, define the symmetrized density $\bar{p}_X(x) = \int_{g\in G} p_X(gx)dg$. The density $\bar{p}_X$ is the "closest" invariant distribution to p_X, i.e. for any G-invariant measure on $\mathcal{X}$ it minimizes $\int_x (i(x) - p_X(x))^2 dx$ over all invariant densities i. We now wish to approximate some distance between p_X and $\bar{p}_X$ based on finite training samples (where sampling from $\bar{p}_X$ can be emulated by applying random G-augmentations).

Chiu & Bloem-Reddy (2023) set d to be the maximum mean discrepancy (MMD) with respect to some choice of kernel, corresponding to a non-parametric two sample statistical test. While it is already interesting to apply this metric directly to materials datasets, their experiments do not include kernels suitable for point clouds. Rectifying this requires choosing a kernel suitable for $\mathcal{X}$, which may be non-trivial (particularly for geometric data that includes chemical information), and as noted in Lopez-Paz & Oquab (2017), may not return values in units that are directly interpretable[2].

As an alternative, we propose applying a two sample classifier test, which is a common tool for detecting and quantifying distribution shift in machine learning (Lopez-Paz & Oquab, 2017). Define a distance[3] d between distributions as the test accuracy of a neural net NN trained to distinguish between the two distributions as a binary classification problem:

$$d(p_0, p_1) = \mathbb{E}_{c\sim\text{Bern}(\frac{1}{2})}\mathbb{E}_{x\sim p_c}\left[\mathbb{1}\big(\,\text{NN}(x) = c\big)\right]$$

If p_X is already group-invariant, then $p_X = \bar{p}_X$ and no network can reliably distinguish between samples from the two[4]. Concretely, we construct a binary classification dataset from an original dataset $\mathcal{X}$ as shown in Figure 1, with half of $\mathcal{X}$ transformed by random group elements (label 1), and the rest of the dataset left as is (label 0). We focus our attention on the rotation group $SO(3)$, as it is one of the most fundamental groups in materials science, but our methodology extends to other groups. The trained classifier's test accuracy is easily interpretable, reflecting how often the classifier can distinguish between the original and symmetrized distributions.

Following Chiu & Bloem-Reddy (2023), we can moreover formulate our setup as a two-sample test, with null hypothesis H_0 that $p_X(x) = \bar{p}_X(x)$ and test statistic given by the test accuracy of the classifier. Given a sample $\{x_1, \ldots, x_n\}$ drawn from $\mathcal{X}$, we estimate a p-value for a two-sample test of level α by Monte Carlo sampling, retraining the classifier for each "sample" (see Appendix B.1).

3 Experiments

Our experiments serve dual goals. First, we *quantify* the distributional symmetry-breaking in commonly-used benchmark materials datasets for property, energy, and force predictions (Figure 4). Second, we *validate* the classifier metric by synthetically modifying p_X (Figure 3). In particular, we randomly rotate a specified fraction f of p_X, thereby making it more similar to $\bar{p}_X$ (and more distributionally symmetric); when $f = 1.0$, we recover $\bar{p}_X$. We therefore expect the test accuracies to linearly interpolate between 50% (at 1.0) and their value at 0.0 (the original p_X), which they do.

Experimental setup: For our classifier, we use a simple transformer architecture with embeddings for atom types and 3D molecular positions. To compare to Chiu & Bloem-Reddy (2023), we also adapt MMD-based methods through implementing kernels based on point cloud distances (Chamfer, Hausdorff, and a simple distance based on the point cloud means and covariances, see Appendix C).[5] We report both the raw classifier test accuracy, and the p-values for comparisons to baselines. The baselines and classifier metrics generally agree on these datasets, which provides strong evidence for our conclusion that the datasets are canonicalized. Our conclusions for each dataset are supported by statistically significant p-values for a significance level $\alpha = 0.05$. Figure 2 highlights the importance of selecting an appropriate kernel, as the naive kernel does not perform as expected. For full experimental details and plots, see Appendix B.

QM9 The QM9 dataset consists of 130k stable organic molecules with $\leq$9 heavy atoms, together with 19 quantum mechanical properties. As shown in Figure 4 and Figure 12, QM9 is highly canonicalized by 3D positions. The original paper (Wu et al., 2017) states that molecular conformers were generated using the commercial software CORINA, which likely performs some canonicalization of the SMILES strings by default (see Figure 14). Additional plots, including classifier predictions under rotation, are shown in Appendix B.3.

MD17 The revised MD17 dataset contains 100k structures from molecular dynamics (MD) simulations for 10 small organic molecules, with energies and forces. We train a separate model for each molecule, thus illuminating the relative alignment of each conformer throughout its trajectory.

[2] unless used in a Monte Carlo p-value estimate

[3] If we considered *all* classifiers, this distance would be linearly related to the total variation distance.

[4] Note that the network is asked to distinguish between two distributions that differ only by group operations (e.g. rotations), and therefore should *not* be group-invariant.

[5] Note these distances do not account for atom types.

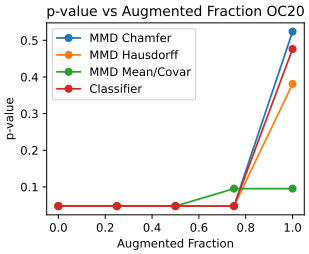

Figure 2: p-value vs Augmented Fraction OC20 (surface+adsorbate).

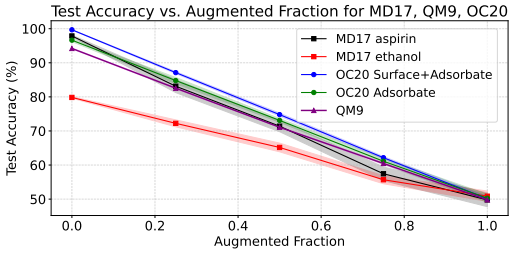

Figure 3: Test accuracy vs rotated fraction for aspirin and ethanol from rMD17, OC20 surface+adsorbate, OC20 adsorbate, and QM9. See Appendix B.2 for other rMD17 molecules.

Molecule	**Test Acc.**
rMD17 Aspirin	97.869
rMD17 Ethanol	79.834
OC20 S+A	99.280
OC20 A	96.529
QM9	94.204

Figure 4: Test accuracy on the original dataset (the leftmost values of Figure 3).

While we find that all trajectories have distributional symmetry breaking, interestingly, the degree varies widely between molecules (see Figure 3, or Figure 5 for all molecules). We hypothesize that this is both due to the initial conditions for the simulation, and the differing physical structures of each molecule.

Open Catalyst 2020 (OC20) The OC20 dataset consists of small molecules (adsorbates) placed on periodic crystalline catalysts (surfaces), represented by slabs in the xy plane (Chanussot* et al., 2021). We apply our metric to the surface+adsorbate system, and also solely to the adsorbate. We hypothesize that both systems will be highly canonicalized, due to the slab's alignment with the xy plane. We find (Figure 3) that the adsorbate is a bit less canonicalized than the slab, but the slab orientation likely still has a large impact on the adsorbate's preferred orientation.

4 Conclusion

In this work, we presented an interpretable metric for quantifying the degree of distributional symmetry-breaking present in a dataset, without any prior domain knowledge of the dataset. Applying our method to commonly used materials benchmarks (QM9, MD17, and OC20), we observed an extremely high level of canonicalization. Since these datasets are routinely used in the development of foundational AI methods for materials discovery, understanding aspects of their generation and biases is crucial. We view such quantifications as an important first step, with domain expertise eventually required to judge whether the detected asymmetry is inherent and useful (such as MNIST 6s and 9s), or incidental and user-defined (such as arbitrary canonicalization).

Future Work Statistical tests and metrics for invariance only indicate *if* a dataset was canonicalized, and not *how* it was canonicalized. Our trained classifier offers intriguing possibilities for obtaining a finer-grained picture via interpretability methods. Moreover, the implications of our findings on equivariant vs non-equivariant model selection remain unclear, since we found that commonly used datasets are very canonicalized, yet equivariant models are still SOTA. Future directions include applying our test to local neighborhoods of molecules, thereby investigating the hypothesis that locality is an important bias for successful equivariant methods, as well as modifying our test to incorporate task-dependence (i.e. asking if the dataset is distributionally asymmetric in a way that is *helpful* for a given task). Finally, distributional symmetry-breaking with respect to *permutations* is highly relevant for autoregressive foundation models. For example, Gruver et al. (2024) trained a LLM to generate stable materials as text, thereby breaking permutation symmetries Surprisingly, permutation augmentations hurt performance. We could apply our metric to probe for preferred orderings in their training dataset as one possible explanation for this phenomenon, providing valuable insight for future LLM approaches to materials generation.

ACKNOWLEDGMENTS

The authors are grateful to Vasco Portilheiro for helpful discussions. Hannah Lawrence is supported by the Fannie and John Hertz Foundation. Elyssa Hofgard is supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy Computational Science Graduate Fellowship under Award Number DE-SC0024386. Robin Walters is supported by NSF Grant 2134178.

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

REFERENCES

Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E(3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nat. Commun.*, 13(1):2453, May 2022.

Lowik Chanussot*, Abhishek Das*, Siddharth Goyal*, Thibaut Lavril*, Muhammed Shuaibi*, Morgane Riviere, Kevin Tran, Javier Heras-Domingo, Caleb Ho, Weihua Hu, Aini Palizhati, Anuroop Sriram, Brandon Wood, Junwoong Yoon, Devi Parikh, C. Lawrence Zitnick, and Zachary Ulissi. Open catalyst 2020 (oc20) dataset and community challenges. *ACS Catalysis*, 2021. doi: 10.1021/acscatal.0c04525.

Kenny Chiu and Benjamin Bloem-Reddy. Non-parametric hypothesis tests for distributional group symmetry. In *NeurIPS AI for Science Workshop*, 2023.

Stefan Chmiela, Alexandre Tkatchenko, Huziel E. Sauceda, Igor Poltavsky, Kristof T. Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. 2017. doi: 10.1126/sciadv.1603015. URL https://www.science.org/doi/10.1126/sciadv.1603015. Publisher: American Association for the Advancement of Science.

Anders S Christensen and O Anatole von Lilienfeld. On the role of gradients for machine learning of molecular energies and forces. *Machine Learning: Science and Technology*, 1(4): 045018, oct 2020. doi: 10.1088/2632-2153/abba6f. URL https://dx.doi.org/10.1088/2632-2153/abba6f.

Taco S. Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical CNNs. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=Hkbd5xZRb.

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

Krish Desai, Benjamin Nachman, and Jesse Thaler. Symmetry discovery with deep learning. *Physical Review D*, 105(9):096031, 2022.

Bryn Elesedy and Sheheryar Zaidi. Provably strict generalisation benefit for equivariant models. In *International conference on machine learning*, pp. 2959–2969. PMLR, 2021.

Shiang Fang, Mario Geiger, Joseph G. Checkelsky, and Tess Smidt. Phonon predictions with E(3)-equivariant graph neural networks. *arXiv preprint: arXiv:2403.11347*, 2024. URL https://arxiv.org/abs/2403.11347v1.

Marc Anton Finzi, Gregory Benton, and Andrew Gordon Wilson. Residual pathway priors for soft equivariance constraints. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman

Vaughan (eds.), *Advances in Neural Information Processing Systems*, 2021. URL https://openreview.net/forum?id=k505ekjMzww.

Nathan C Frey, Ryan Soklaski, Simon Axelrod, Siddharth Samsi, Rafael Gómez-Bombarelli, Connor W Coley, and Vijay Gadepally. Neural scaling of deep chemical models. *Nature Machine Intelligence*, 5(11):1297–1305, October 2023.

Nate Gruver, Anuroop Sriram, Andrea Madotto, Andrew Gordon Wilson, C. Lawrence Zitnick, and Zachary Ward Ulissi. Fine-tuned language models generate stable inorganic materials as text. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=vN9fpfqoP1.

Elyssa Hofgard, Rui Wang, Robin Walters, and Tess E. Smidt. Relaxed equivariant graph neural networks. *arXiv preprint arXiv:2407.20471*, 2024.

Yi-Lun Liao, Brandon Wood, Abhishek Das, and Tess Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. *arXiv preprint arXiv:2306.12059*, 2023.

David Lopez-Paz and Maxime Oquab. Revisiting classifier two-sample tests. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJkXfE5xx.

Daniel McNeela. Almost equivariance via lie algebra convolutions. *arXiv preprint arXiv:2310.13164*, 2023.

Cameron J Owen, Steven B Torrisi, Yu Xie, Simon Batzner, Jennifer Coulter, Albert Musaelian, Lixin Sun, and Boris Kozinsky. Complexity of Many-Body interactions in transition metals via Machine-Learned force fields from the TM23 data set. February 2023.

Joshua A Rackers, Lucas Tecot, Mario Geiger, and Tess E Smidt. A recipe for cracking the quantum scaling limit with machine learned electron densities. *Mach. Learn.: Sci. Technol.*, 4(1):015027, February 2023.

Han Shao, Omar Montasser, and Avrim Blum. A theory of pac learnability under transformation invariances. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2024. Curran Associates Inc. ISBN 9781713871088.

Tess E Smidt, Mario Geiger, and Benjamin Kurt Miller. Finding symmetry breaking order parameters with euclidean neural networks. *Physical Review Research*, 3(1):L012002, 2021.

Alonso Urbano and David W. Romero. Self-supervised detection of perfect and partial input-dependent symmetries. *arXiv preprint arXiv:2312.12223*, 2024.

Dian Wang, Jung Yeon Park, Neel Sortur, Lawson LS Wong, Robin Walters, and Robert Platt. The surprising effectiveness of equivariant models in domains with latent symmetry. In *The Eleventh International Conference on Learning Representations*, 2023.

Dian Wang, Xupeng Zhu, Jung Yeon Park, Mingxi Jia, Guanang Su, Robert Platt, and Robin Walters. A general theory of correct, incorrect, and extrinsic equivariance. *Advances in Neural Information Processing Systems*, 36, 2024a.

Dian Wang, Xupeng Zhu, Jung Yeon Park, Mingxi Jia, Guanang Su, Robert Platt, and Robin Walters. A general theory of correct, incorrect, and extrinsic equivariance. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2024b. Curran Associates Inc.

Rui Wang, Elyssa Hofgard, Robin Walters, and Tess Smidt. Discovering symmetry breaking in physical systems with relaxed group convolution. *arXiv preprint arXiv:2310.02299*, 2024c.

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. *Chemical Science*, 9(2):513, October 2017. doi: 10.1039/c7sc02664a. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC5868307/.

Jianke Yang, Robin Walters, Nima Dehmamy, and Rose Yu. Generative adversarial symmetry discovery. *arXiv preprint arXiv:2302.00236*, 2023.

APPENDICES

A Related Work **7**

B Experiments **7**

B.1 Computation of p-values 7

B.2 rMD17 7

B.3 QM9 9

B.3.1 Evaluation of classifier predictions on rotated inputs 10

B.4 Open Catalyst Project 2020 (OC20) 10

C Maximum Mean Discrepancy (MMD) for Point Clouds **12**

C.1 Maximum Mean Discrepancy (MMD) 12

C.2 Naive Kernel (Mean/Covar) 14

C.3 Chamfer Distance Kernel 14

C.4 Hausdorff Distance Kernel 14

A RELATED WORK

Various works have addressed discovering symmetry breaking in physical datasets including Wang et al. (2024c); Finzi et al. (2021); McNeela (2023); Hofgard et al. (2024); Smidt et al. (2021); Urbano & Romero (2024). In particular, Wang et al. (2024c) distinguishes between distributional and functional symmetry breaking. Distributional symmetry has also been termed extrinsic equivariance (Wang et al., 2023) and Wang et al. (2024b) showed that, in some cases, using an equivariant model for a problem with extrinsic equivariance can be harmful. Shao et al. (2024) established that any equivariant algorithm applied to extrinsically equivariant data, under certain assumptions on the hypothesis class, cannot obtain optimal sample complexity in terms of PAC learnability. This provides strong motivation for our method to detect extrinsic equivariance.

Desai et al. (2022) proposes a generative model framework for discovering distributional symmetry breaking (SymmetryGAN) with respect to some reference density. Indeed, Desai et al. (2022) and Yang et al. (2023) train discriminative networks for symmetry discovery in a similar way to our binary classifier, but do not produce a quantitative measure of distributional asymmetry. Chiu & Bloem-Reddy (2023) framed testing for distributional symmetry breaking as a non-parametric hypothesis test, following literature on two-sample tests, and uses the distance between the group-averaged and the original distributions as the test statistic. Lopez-Paz & Oquab (2017) posits that one can use a binary classifier for the test statistic for a more interpretable metric.

B EXPERIMENTS

B.1 COMPUTATION OF P-VALUES

AlgorithmB.1 outlines the process for computing p-values.

B.2 RMD17

We use the revised MD17 dataset Christensen & von Lilienfeld (2020), as the original MD17 dataset has a high level of numerical noise Chmiela et al. (2017). The revised MD17 dataset was calculated with a more accurate DFT functional/convergence criteria than the original MD17. We use the provided five train/test splits from `https://figshare.com/articles/dataset/Revised_MD17_dataset_rMD17_/12672038` and train a separate model for each molecule. Note it is not recommended to train a model on more than 1,000 samples from rMD17 Christensen

Algorithm 1 P-value Computation

Input: Training set $\mathcal{D}_{\text{train}}$, test set $\mathcal{D}_{\text{test}}$, calibration distances sample size n_1, actual distances sample size n_2, distance function Distance$(\cdot, \cdot)$.
Output: p-value
actual_dists $\leftarrow$ []
calibration_dists $\leftarrow$ []
▷ Compute calibration distances under null hypothesis
for $i = 1$ to n_1 **do**
    Sample training set $\tilde{\mathcal{D}}_{\text{train}}$ and test set $\tilde{\mathcal{D}}_{\text{test}}$ from $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$.
    Apply rotation transformation to all data
    $d_c \leftarrow$ Distance$(\tilde{\mathcal{D}}_{\text{train}}, \tilde{\mathcal{D}}_{\text{test}})$
    calibration_dists.append(d_c)
end for
▷ Compute actual distances
for $i = 1$ to n_2 **do**
    Sample training set $\tilde{\mathcal{D}'}_{\text{train}}$ and test set $\tilde{\mathcal{D}'}_{\text{test}}$ from $\mathcal{D}_{\text{train}}$ and $\mathcal{D}_{\text{test}}$.
    Apply rotation transformation to subset of data
    $d_a \leftarrow$ Distance$(\tilde{\mathcal{D}'}_{\text{train}}, \tilde{\mathcal{D}'}_{\text{test}})$
    actual_dists.append(d_a)
end for
$\bar{d}_a \leftarrow \frac{1}{n_2} \sum_{i=1}^{n_2}$ actual_dists$[i]$ ▷ Compute mean of actual distances
count $\leftarrow |\{d_c \in \text{calibration\_dists} : d_c > \bar{d}_a\}|$
p-value $\leftarrow \frac{1+\text{count}}{1+n_1}$
return p-value

& von Lilienfeld (2020), even though the dataset has 100,000 conformers for each trajectory. We train a generic transformer with 812k parameters for 50 epochs on the train/test splits provided with the Adam optimizer at learning rate `1e-5` and batch size `128`.

As seen in Figure 5, all molecules are canonicalized, yet ethanol and malohaldehyde have a noticeably lower degree of canonicalization. We plan to explore this finding in future work and determine whether it is related to local symmetric motifs within each molecule. As a physical sanity check for our distributional symmetry breaking metric, we plot the distributions for the principal moments of inertia for each molecule. Examples of more canonicalized and less canonicalized molecules as determined by our metric are shown in Figure 6.

For a discrete system of point masses, the inertia tensor $\mathbf{I}$ is given by:

$$\mathbf{I} = \sum_i m_i \left[\|\mathbf{r}_i\|^2 \mathbf{I} - \mathbf{r}_i \mathbf{r}_i^T \right]$$

The eigenvalues of the inertia tensor are the principal moments and represent the resistances to rotation around the body's principal axes (which are the eigenvectors). Intuitively, if a molecule is more canonicalized over the MD trajectory, we would expect it to stay in one orientatation and for the distributions of the principal axes over time to remain distinct. If it is less canonicalized, there may be more overlap between the distributions.

Figure 8 demonstrates the values used in our computation of the p-values for each method (on a row) and different levels of augmentation in the detection dataset (column) for one of the molecules in rMD17 (benzene). The p-value plots were computed using 20 samples (for each histogram) of size 1k corresponding to the given train/test splits, trained for 20 epochs (in the case of the classifier metric). As shown, all methods separate the calibration distances from the actual distances, resulting in identical, statistically significant p-values. As the tests are asked to distinguish between increasingly similar datasets (moving from left to right), the histograms gradually move closer together, until they overlap. For ease of visualization, Figure 7 plots the mean distance computed from each histogram for benzene (excluding the calibration distances). We also plot the p-value vs. the augmented fraction Figure 9. As for OC20 in Figure 2, the Chamfer and Hausdorff kernels exhibit similar trends to the classifier, and the naive mean/covar kernel exhibits less reasonable behavior.

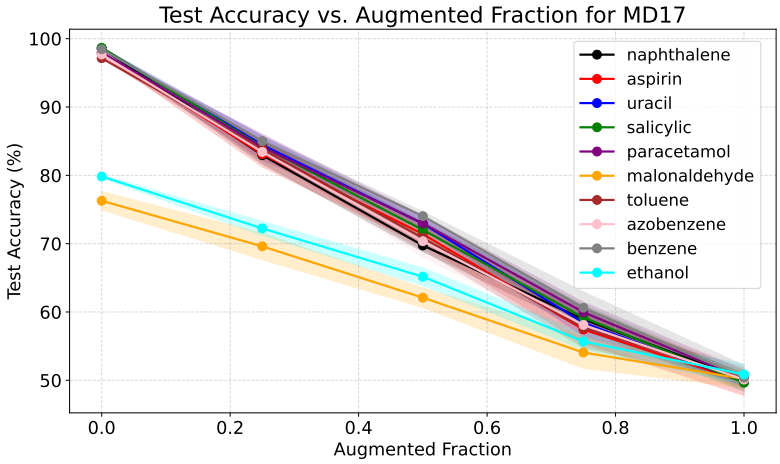

Figure 5: Test accuracy vs. augmented fraction for all molecules in rMD17. Note the difference between the 8 more canonicalized molecules and ethanol/malonaldehyde.

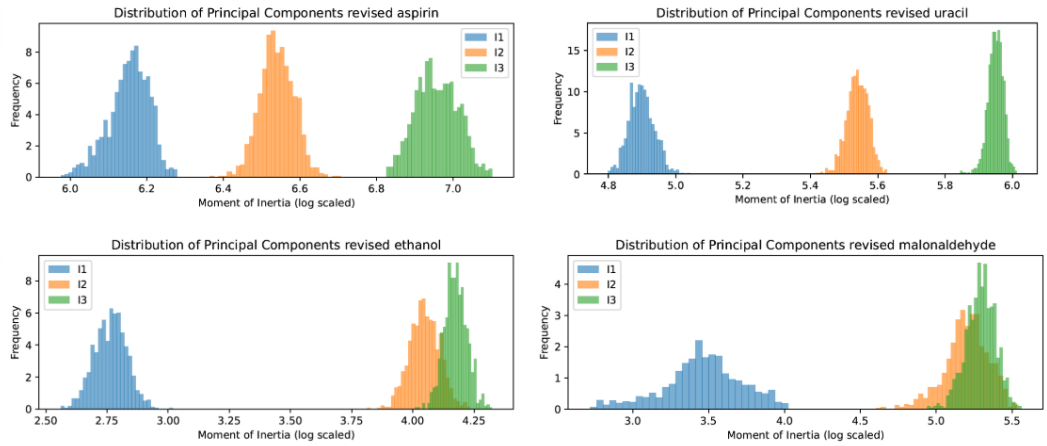

Figure 6: Comparisons of the principal components of the inertia tensor for more canonicalized (top row) and less canonicalized (bottom row) molecules.

This illustrates the importance of choosing a good kernel and provides a relative advantage of our method. All other molecules in rMD17 exhibted similar trends for the p-values.

B.3 QM9

We use `torch_geometric` for loading the dataset, and train a generic transformer architecture with 812k parameters for 20 epochs with a randomly selected 60%/20%/20% train/validation/test split and the Adam optimizer at learning rate `1e-5` and batch size `128`.

Figure 10 demonstrates the values used in our computation of the p-values for each method (on a row) and different levels of augmentation in the detection dataset (column). The p-value plots were computed using 20 samples (for each histogram) of size 1k, trained for 20 epochs (in the case of the classifier metric). All methods exhibit the expected behavior: as the augmented fraction increases —i.e. as the distribution becomes more similar to the reference, perfectly symmetrized distribution—the distance decreases. It is important to note that the classifier distance does not match Figure 3 due to the difference in batch size: the classifier was trained on a much smaller dataset, and as shown in the loss plot in Figure 13, training did not converge in this time. This time constraint was necessary to facilitate the number of runs necessary to compute a p-value. However, conversely, the baseline methods cannot scale to the entire datset, whereas the classifier method

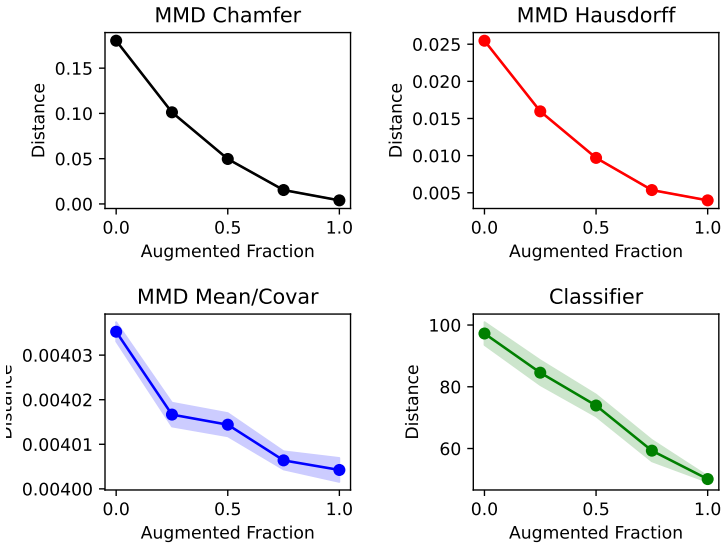

Figure 7: Different distance metrics from a perfectly symmetrized distribution, as a function of the degree of synthetic augmentation of the rMD17 dataset for benezene. (Higher augmented fraction indicates a greater similarity to the symmetrized distribution).

can. Moreover, even without the convergence, the histograms corresponding to the classifier metric in Figure 10 are still sufficiently well-separated to provide reasonable p-values on our synthetic experiment. See also Figure 12 for the p-values; note that all methods agree at the level of p-value on the original dataset.

As with the MD17 dataset, we also plot the distribution of the principal moments of inertia. As QM9 contains different molecules (with different masses), we normalize the inertia tensor for each molecule by its total mass. This is shown in Figure 14. We note that I_1 is more sharply peaked, while I_2 and I_3 are quite similar. This suggests there are two directions that are rotationally equivalent for many molecules (e.g. in-plane symmetry such as in a benzene ring) and that there may be one consistent direction that molecules are aligned with. We plan to conduct further exploration of how the QM9 dataset was generated using CORINA and the algorithms used within CORINA.

B.3.1 Evaluation of classifier predictions on rotated inputs

A primary motivator for using the classifier distance for distributional asymmetry detection, is for the opportunities for exploration and interpretation of the trained classifier. As a first step in this direction, we evaluate the sigmoid of the classifier's logits on a discrete grid of 3D rotations (where two Euler angles are varied and the third is held constant) of a few inputs, shown in Figure 15.

B.4 Open Catalyst Project 2020 (OC20)

For our study, we use the 200K subset from the structure to energy and forces (S2EF) task, available at `https://fair-chem.github.io/core/datasets/oc20.html#structure-to-energy-and-forces-s2ef-task`. It would be interesting in the future to explore other tasks (e.g. Initial Structure to Relaxed Structure) and larger dataset sizes, as the OC20 dataset training set alone has 20 million structures. We use the preprocessing pipeline provided at `https://fair-chem.github.io/core/datasets/oc20.html`. Positions for each catalyst+adsorbate are tagged with 0: catalyst surface, 1: catalyst sub-surface, and 2: adsorbate. The unit cell for the catalyst is repeated twice in the x direction, twice in the y direction, and once in the z direction, leading to the slab's alignment with the xy plane. This alignment most likely trivially causes our metric to report distributional symmetry breaking. It would thus be interesting in future work to consider how to treat periodic crystalline systems. The p-value plots were computed

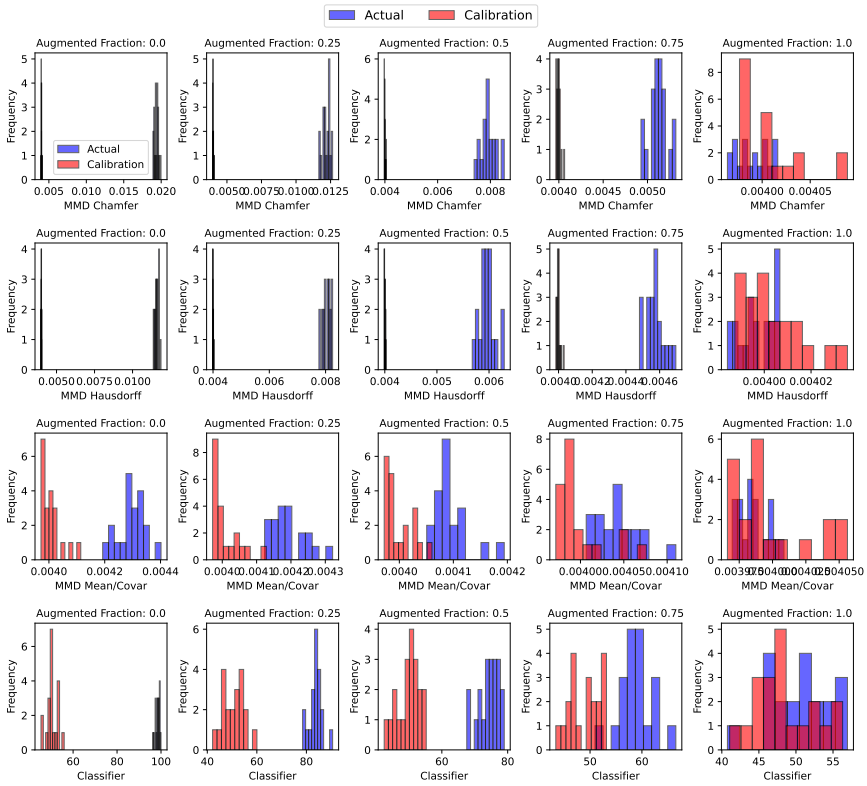

Figure 8: Distance metrics for different methods, and at different levels of augmentation for benzene (i.e. different levels of underlying distributional similarity).

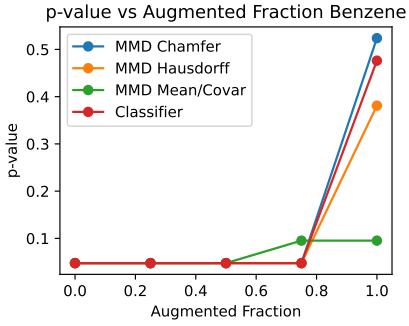

Figure 9: p-value vs augmented fraction for benzene rMD17.

using 20 samples (for each histogram) of size 50k, trained for 20 epochs (in the case of the classifier metric). The p-values follow the expected trends as was the case for the other datasets.

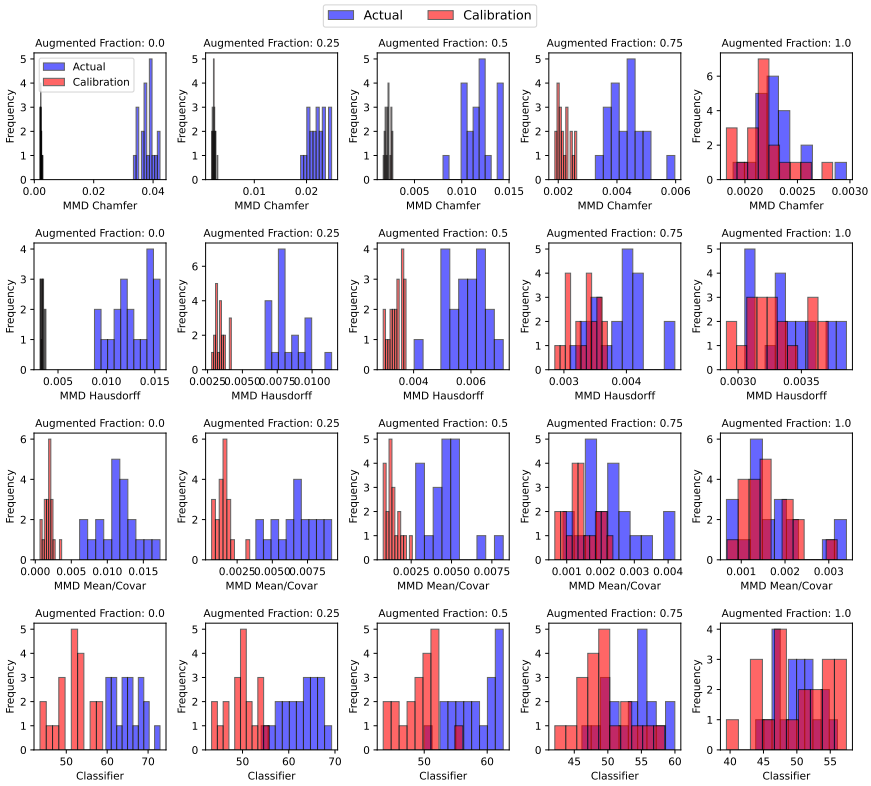

Figure 10: Distance metrics for different methods, and at different levels of augmentation (i.e. different levels of underlying distributional similarity).

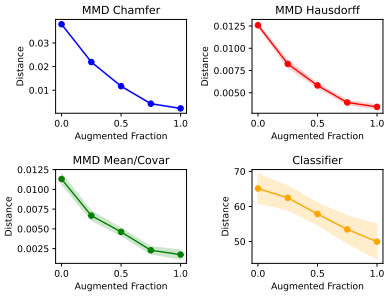

Figure 11: Different distance metrics from a perfectly symmetrized distribution, as a function of the degree of synthetic augmentation of the QM9 dataset. (Higher augmented fraction indicates a greater similarity to the symmetrized distribution.)

C Maximum Mean Discrepancy (MMD) for Point Clouds

C.1 Maximum Mean Discrepancy (MMD)

MMD is a statistical distance metric that measures the discrepancy between two probability distributions p_0, p_1. Unlike many other distance metrics, MMD does not require any assumptions about the distributions or explicit density estimation. Thus, MMD is useful for high-dimensional or complex

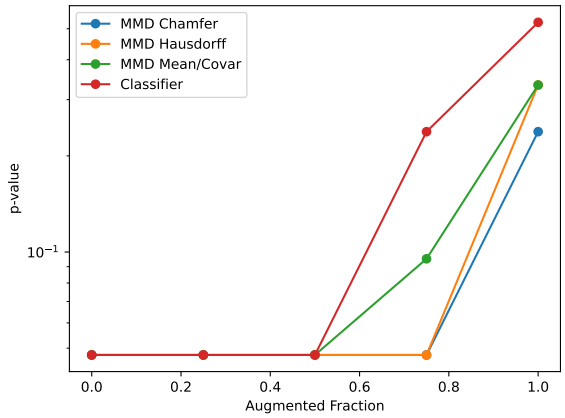

Figure 12: p-values for different methods, and at different levels of augmentation (i.e. different levels of underlying distributional similarity).

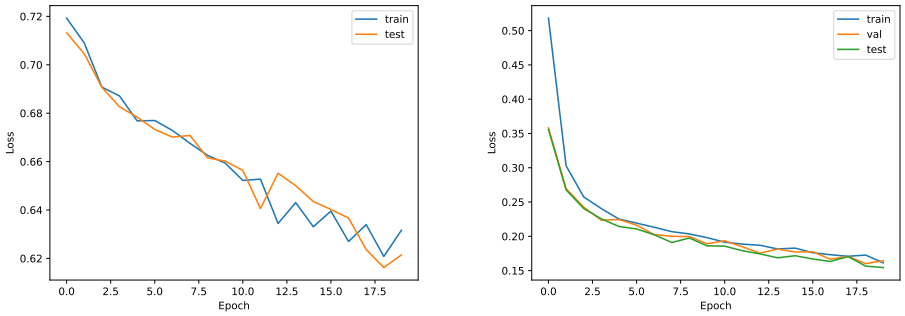

Figure 13: Left: the loss curve from one of the 20 training runs used to compute the classifier distance in the p-value computation, on 1k examples. Right: the loss curve from a training run used to compute the classifier distance over the full dataset. As shown, the loss converged much faster for the full dataset, whereas with only 1k examples (one one-hundredth of the size), convergence is much slower.

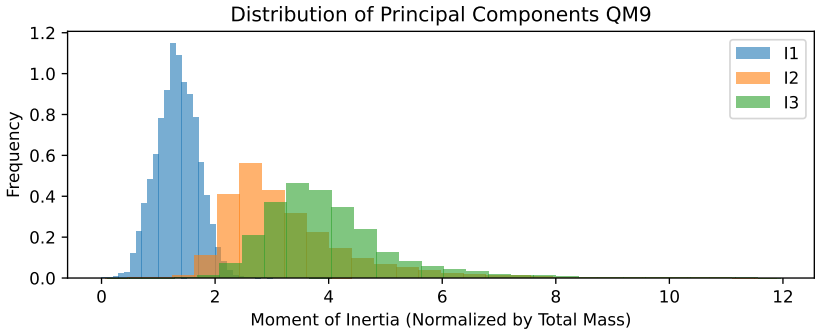

Figure 14: Principal moments of inertia distribution normalized by molecular mass for QM9.

distributions. The definition of MMD is:

$$\text{MMD}^2(p_0, p_1) = \mathbb{E}_{x_0, x_0' \sim p_0}\left[k(x_0, x_0')\right] + \mathbb{E}_{x_1, x_1' \sim p_1}\left[k(x_1, x_1')\right] - 2\mathbb{E}_{x_0 \sim p_0, x_1 \sim p_1}\left[k(x_0, x_1)\right],$$

Where $k(\cdot, \cdot)$ is the kernel function. To compute the MMD, we can use the empirical MMD, which is an unbiased estimator of the true MMD and only needs a set of samples from each distribution. Algorithm C.1 provides pseudocode for the implementation of empirical MMD.

To compute the MMD, we need to choose a kernel function that is positive definite and characteristic. The choice of kernel can have a significant impact on the MMD value. Based on natural distance measures between point clouds, we implement three different kernels for our experiments: the naive kernelMean/Covar, the Chamfer distance kernel, and the Hausdorff distance kernel.

C.2 Naive Kernel (Mean/Covar)

The most naive way to compute the distance between point clouds is to compute the distance between their respective means and covariances. We call this method "MMD Mean/Covar", as well as the naive kernel. Since the naive kernel only uses the means and covariances of the point clouds, it lacks the ability to capture the local information of the point clouds. AlgorithmC.2 gives an implementation of the Naive kernel:

C.3 Chamfer Distance Kernel

The Chamfer distance is a commonly used distance metric, measuring the similarity between two point clouds. It is defined as the sum of the average of squared Euclidean distances from each point in one set to its nearest neighbor in the other set. Formally, the Chamfer distance is defined as:

$$\text{CD}(X, Y) = \frac{1}{|X|} \sum_{x \in X} \min_{y \in Y} ||x - y||^2 + \frac{1}{|Y|} \sum_{y \in Y} \min_{x \in X} ||x - y||^2,$$

where X and Y are two point clouds, x and y are points in the point clouds, and $||x - y||$ is the Euclidean distance between points x and y.

Since the Chamfer distance kernel uses the minimum distance between points, it mainly captures local information, and always ignores global structure (such as the overall shape distribution and point cloud density). AlgorithmC.3 gives an implementation of Chamfer distance kernel.

C.4 Hausdorff Distance Kernel

The Hausdorff distance is also a distance metric that measures the distance between two sets of points. By replacing the average operation in Chamfer distance with the maximum operation, we obtain the Hausdorff distance as:

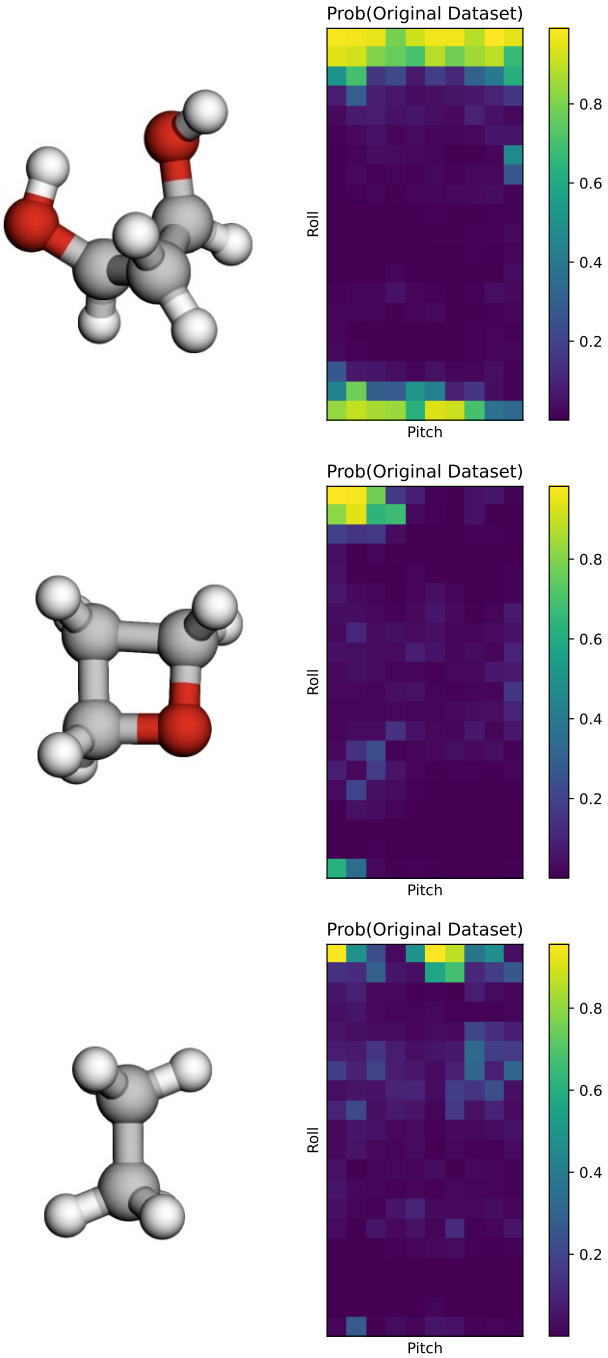

Figure 15: Left: example molecules from QM9. Right: the corresponding plot of the trained binary classifier's predicted probability that that rotation of the input came from the original, canonicalized dataset.

Algorithm 2 Compute Maximum Mean Discrepancy (MMD)

Require: x, y (input samples), $mask\_x, mask\_y$ (optional masks), $kernel\_func$ (kernel function)

$n_x \leftarrow$ length of x, $n_y \leftarrow$ length of y ▷ Compute XX pairwise similarities
$xx\_indices \leftarrow$ upper triangular indices of (n_x, n_x)
if $mask\_x$ is not None **then**
    $xx\_distances \leftarrow kernel\_func(x[xx\_indices_0], x[xx\_indices_1], mask\_x[xx\_indices_0], mask\_x[xx\_indices_1])$
    $xx\_diag \leftarrow kernel\_func(x, x, mask\_x, mask\_x)$
else
    $xx\_distances \leftarrow kernel\_func(x[xx\_indices_0], x[xx\_indices_1])$
    $xx\_diag \leftarrow kernel\_func(x, x)$
end if
$xx\_mean \leftarrow \frac{2\sum xx\_distances + \sum xx\_diag}{n_x \cdot n_x}$
▷ Compute YY pairwise similarities
$yy\_indices \leftarrow$ upper triangular indices of (n_y, n_y)
if $mask\_y$ is not None **then**
    $yy\_distances \leftarrow kernel\_func(y[yy\_indices_0], y[yy\_indices_1], mask\_y[yy\_indices_0], mask\_y[yy\_indices_1])$
    $yy\_diag \leftarrow kernel\_func(y, y, mask\_y, mask\_y)$
else
    $yy\_distances \leftarrow kernel\_func(y[yy\_indices_0], y[yy\_indices_1])$
    $yy\_diag \leftarrow kernel\_func(y, y)$
end if
$yy\_mean \leftarrow \frac{2\sum yy\_distances + \sum yy\_diag}{n_y \cdot n_y}$
▷ Compute XY cross similarities
if $mask\_x$ is not None and $mask\_y$ is not None **then**
    $xy\_distances \leftarrow kernel\_func(x[:, None], y[None, :], mask\_x[:, None], mask\_y[None, :])$
else
    $xy\_distances \leftarrow kernel\_func(x[:, None], y[None, :])$
end if
$xy\_mean \leftarrow$ mean of $xy\_distances$
▷ Compute final MMD value
$mmd \leftarrow xx\_mean + yy\_mean - 2 \cdot xy\_mean$
return mmd

Algorithm 3 Naive Kernel Computation

Require: x, y (input tensors), $mask\_x, mask\_y$ (optional masks, related to the variable numbers of nodes across input molecules), σ (scaling parameter)

▷ Compute mean and covariance with or without masks

if $mask\_x$ is not None and $mask\_y$ is not None **then**

$mean_x \leftarrow \frac{\sum x}{\sum mask\_x}$

$mean_y \leftarrow \frac{\sum y}{\sum mask\_y}$

$cov_x \leftarrow \frac{\sum xx^T}{\sum mask\_x}$

$cov_y \leftarrow \frac{\sum yy^T}{\sum mask\_y}$

else

$mean_x \leftarrow$ mean of x, $mean_y \leftarrow$ mean of y

$cov_x \leftarrow \frac{\sum xx^T}{|x|}$

$cov_y \leftarrow \frac{\sum yy^T}{|y|}$

end if

▷ Compute embeddings

$embedding_x \leftarrow$ concatenate($mean_x$, flatten(cov_x))

$embedding_y \leftarrow$ concatenate($mean_y$, flatten(cov_y))

▷ Compute pairwise distance and apply Gaussian kernel

$dist \leftarrow ||embedding_x - embedding_y||$

$kernel\_val \leftarrow \exp(-dist^2/\sigma)$

return $kernel\_val$

Algorithm 4 Chamfer Kernel Computation

Require: x, y (input tensors), $mask\_x, mask\_y$ (optional masks), σ (scaling parameter)

▷ Compute Chamfer distances

$dist_1 \leftarrow$ minimum pairwise Euclidean distance from x to y

$dist_2 \leftarrow$ minimum pairwise Euclidean distance from y to x

▷ Handle masks if provided

if $mask\_x$ is not None and $mask\_y$ is not None **then**

$masked\_min\_dist_1 \leftarrow dist_1 \cdot mask\_x$

$masked\_min\_dist_2 \leftarrow dist_2 \cdot mask\_y$

$chamfer\_dist \leftarrow \frac{1}{2}\left(\frac{\sum masked\_min\_dist_1}{\sum mask\_x} + \frac{\sum masked\_min\_dist_2}{\sum mask\_y}\right)$

else

$chamfer\_dist \leftarrow \frac{1}{2}\left(\text{mean}(dist_1) + \text{mean}(dist_2)\right)$

end if

▷ Apply Gaussian kernel transformation

$kernel\_val \leftarrow \exp\left(-\frac{chamfer\_dist}{2\sigma^2}\right)$

return $kernel\_val$

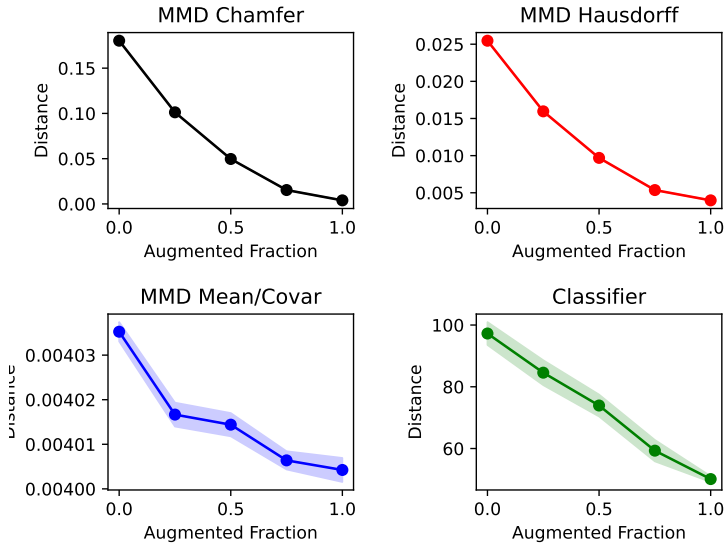

Figure 16: Different distance metrics from a perfectly symmetrized distribution, as a function of the degree of synthetic augmentation of the OC20 dataset. (Higher augmented fraction indicates a greater similarity to the symmetrized distribution).

$$\mathrm{HD}(X, Y) = \max\left(\max_{x\in X}\min_{y\in Y}||x-y||, \max_{y\in Y}\min_{x\in X}||x-y||\right).$$

Because the Hausdorff distance kernel uses the maximum distance between points, it is more sensitive to outliers than Chamfer distance. AlgorithmC.4 gives an implementation of Hausdorff distance kernel.

Algorithm 5 Hausdorff Kernel Computation

Require: x, y (input tensors), $mask\_x, mask\_y$ (optional masks), σ (scaling parameter)

▷ Compute pairwise minimum distances

$dist_1 \leftarrow$ minimum pairwise Euclidean distance from x to y

$dist_2 \leftarrow$ minimum pairwise Euclidean distance from y to x

▷ Handle masks if provided

if $mask\_x$ is not None and $mask\_y$ is not None **then**

  $masked\_dist_1 \leftarrow dist_1 \cdot mask\_x$

  $masked\_dist_2 \leftarrow dist_2 \cdot mask\_y$

  $hausdorff\_dist \leftarrow \max\left(\max(masked\_dist_1), \max(masked\_dist_2)\right)$

else

  $hausdorff\_dist \leftarrow \max\left(\max(dist_1), \max(dist_2)\right)$

end if

▷ Apply Gaussian kernel transformation

$kernel\_val \leftarrow \exp\left(-\frac{hausdorff\_dist}{2\sigma^2}\right)$

return $kernel\_val$

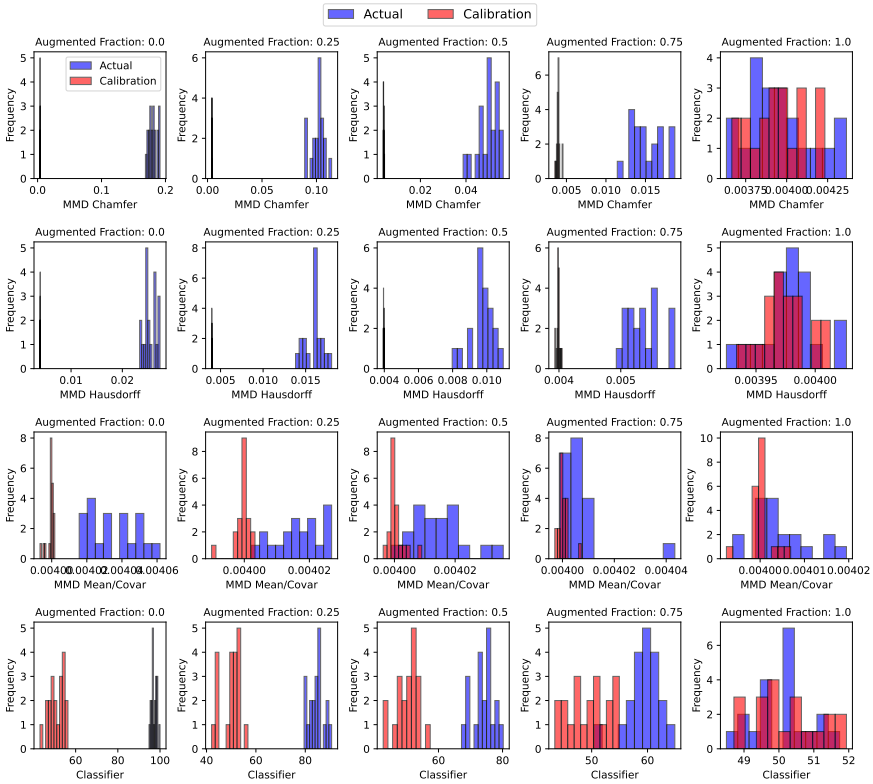

Figure 17: Distance metrics for different methods, and at different levels of augmentation for OC20 (i.e. different levels of underlying distributional similarity).