How the Optimizer Shapes Learned Solutions in Equivariant Neural Networks

Teodor-Mihai Stupariu [1 2] **Andrei Manolache** [1 3 4]

Abstract

Equivariant neural networks encode geometric symmetries by construction, yet they are often difficult to optimize and can underperform less constrained architectures. A growing body of work addresses this through architectural modifications such as constraint relaxation or approximate equivariance, while the role of the optimizer remains comparatively underexplored. We study this direction by comparing Muon and Adam across several equivariant and geometric architectures under pointcloud and molecular learning settings. On ModelNet40, where the comparison is clearest, Muon consistently improves over Adam across all architectures considered. We then analyze the trained ModelNet40 checkpoints through Hessian estimates, loss surface visualizations, and spectral properties of learned weights and intermediate representations. The checkpoints reached by Muon have larger Hessian curvature summaries but more regular loss surfaces, and their learned weights and representations have higher stable and effective ranks. These observations suggest that the interaction between optimizer design and geometric inductive bias deserves further attention from the community.

1. Introduction and Related Work

Equivariant neural networks encode geometric symmetries directly into their architecture, providing a natural inductive bias for problems whose data carries known geometric symmetries (Thomas et al., 2018; Satorras et al., 2021; Bronstein et al., 2021). Hard equivariance constraints make optimization difficult, with the loss landscape of equivariant models shown to contain critical points and even spurious minima (Xie & Smidt, 2025), and can underperform less constrained alternatives at scale (Xie et al., 2025; Brehmer et al., 2025). A natural response, and the dominant one in recent literature, has been to soften the architectural constraints. Approximate (Wang et al., 2022; Petrache & Trivedi, 2023) and relaxed equivariance (Pertigkiozoglou et al., 2024; Manolache et al., 2025; Elhag et al., 2025) methods allow controlled departures from exact symmetry, improving optimization while retaining some symmetry bias. These methods place the architectural constraints at the center of the optimization problem, while the optimizer has received almost no attention in this literature.

Recent work argues that the optimizer is a meaningful source of inductive bias, and that different optimizers do not merely converge at different rates but arrive at qualitatively different solutions when training neural networks (Pascanu et al., 2025). If the difficulty of training equivariant models is partly an optimization problem (Pertigkiozoglou et al., 2024; Manolache et al., 2025), then changing the optimizer is a natural place to intervene, and one largely independent of the architectural relaxations above. We test this idea empirically by comparing Adam (Kingma & Ba, 2015) with Muon (Jordan et al., 2024), a recently proposed optimizer that orthogonalizes its momentum buffer through Newton-Schulz iterations. Our primary setting is ModelNet40 (Wu et al., 2014) and its corrupted variant ModelNet40-C (Sun et al., 2022), where we compare architectures with different geometric inductive biases: EGNN (Satorras et al., 2021), with explicit $E(n)$ equivariance; DGCNN (Wang et al., 2019), a permutation-equivariant graph convolutional network that builds local geometric structure through dynamic k-NN connectivity; and PointNet (Qi et al., 2016), which is permutation invariant via symmetric pooling. We complement this comparison with molecular-learning experiments using the $E(3)$-equivariant GotenNet (Aykent & Xia, 2025) and the permutation-equivariant GINE (Hu et al., 2020) on molecular datasets (Ramakrishnan et al., 2014; Dwivedi et al., 2022; Gómez-Bombarelli et al., 2018).

On ModelNet40, where the comparison is clearest, Muon and Adam differ beyond test accuracy. The trained solutions show distinct local loss geometry and distinct spectral structure in their learned weights and intermediate representations. Muon also reaches higher-curvature checkpoints but with surrounding loss structure that appears more regular.

[1]Bitdefender, Romania [2]Tudor Vianu High School of Computer Science, Romania [3]University of Stuttgart, Germany [4]International Max Planck Research School for Intelligent Systems, Germany. Correspondence to: Andrei Manolache <andrei.manolache@ki.uni-stuttgart.de>.

Workshop on Weight-Space Symmetries, held in conjunction with the 43rd International Conference on Machine Learning, Seoul, South Korea. 2026. Copyright 2026 by the author(s).

Table 1. Best-checkpoint accuracy on ModelNet40 ("Clean") and ModelNet40-C ("Corrupted"). Results are mean accuracy ± std over four seeds; Δ is Muon minus Adam. Higher is better. Muon improves over Adam for every model and evaluation setting.

Eval.	Model	Adam	Muon	Δ
Clean	EGNN	76.91±0.94	**82.08±0.36**	+5.17
Clean	PointNet	84.53±0.70	**87.21±0.39**	+2.67
Clean	DGCNN	87.10±0.69	**89.06±0.17**	+1.96
Corrupted	EGNN	65.76±0.95	**70.12±0.10**	+4.36
Corrupted	PointNet	72.85±1.05	**75.87±0.28**	+3.02
Corrupted	DGCNN	75.26±1.63	**77.84±0.27**	+2.58

Table 2. GotenNet performance on QM9. Results are mean MAE ± std over four seeds. Lower is better. Muon improves over Adam on most targets.

Property	Adam	Muon
Cv	**0.0230±0.0008**	0.0246±0.0015
G	0.0103±0.0032	**0.0087±0.0017**
H	0.0122±0.0020	**0.0085±0.0020**
U	**0.0101±0.0042**	0.0138±0.0079
U0	0.0103±0.0054	**0.0090±0.0022**
zpve	**0.0021±0.0003**	0.0022±0.0005
mu	0.0124±0.0014	**0.0121±0.0011**
alpha	0.0445±0.0058	**0.0408±0.0043**
homo	0.0225±0.0016	**0.0214±0.0007**
lumo	0.0177±0.0019	**0.0173±0.0012**
gap	0.0377±0.0017	**0.0363±0.0025**
r2	0.4320±0.1222	**0.2310±0.1076**

These observations motivate treating optimizer choice as a meaningful part of training equivariant models.

We summarize our contributions below:

1. We compare Adam and Muon across equivariant and geometric architectures under clean, corrupted, and molecular-learning setups.
2. We study the local loss geometry using Hessian estimates and loss-surface visualizations, finding higher curvature for Muon but smoother local loss structure.
3. We examine the spectral structure of learned weights and intermediate representations, finding higher stable and effective ranks under Muon.

2. Experiments and Analysis

We organize the experiments around three questions. First, *how does replacing Adam with Muon affect performance across architectures and evaluation settings?* Second, *how do the resulting solutions differ in local loss geometry?* Third, *does optimizer choice affect the spectral structure of the learned weights and representations?* We begin with the quantitative comparison on ModelNet40, its corrupted variant, and molecular-learning benchmarks.

2.1. Quantitative evaluation

For each dataset and optimizer, we perform a grid search over learning rate and weight decay. We then train four seeds with the selected configuration and report best-checkpoint performance. We first evaluate Adam and Muon on ModelNet40 using EGNN, PointNet, and DGCNN. These architectures impose different geometric inductive biases: EGNN enforces $E(3)$-equivariance, DGCNN builds local geometric structure through dynamic neighbourhoods, and PointNet relies primarily on permutation-invariant aggregation.

Table 1 shows the main classification results. Muon improves over Adam for all three architectures on ModelNet40, and the same ordering holds under ModelNet40-C corruptions, showing clearly that changing only the optimizer can improve performance across architectures with different geometric biases.

We next evaluate GotenNet on QM9 (Ramakrishnan et al., 2014), where the task again involves $E(3)$-equivariant modelling. Table 2 shows that Muon improves most QM9 targets. The additional (permutation-equivariant) message-passing experiments in Appendix A are less conclusive. With GINE on Peptides-func (Dwivedi et al., 2022) (Table 4) and ZINC (Gómez-Bombarelli et al., 2018) (Table 5), Muon reaches stronger best checkpoints on some seeds, but seed-averaged performance is comparable to Adam on Peptides-func and worse on ZINC. These results suggest that optimizer choice is most consequential for some model-dataset pairs, with the strongest evidence in our experiments coming from 3D geometric settings.

2.2. Local loss geometry

Recent work on equivariant loss landscapes shows that equivariance constraints can interact with hidden parameter symmetries to create critical points and even spurious minima (Xie & Smidt, 2025). Motivated by this view, we compare the local loss geometry around trained Adam and Muon models on ModelNet40. Following Li et al. (2018), we visualize loss slices around trained checkpoints. For each optimizer, we select the checkpoint whose accuracy is closest to the four-seed mean. Fig. 1 shows the PointNet loss slice, with corresponding 2D contours and 3D surfaces for the other architectures included in Appendix B.

The two optimizers lead to visibly different local geometry. In the PointNet example, the Adam slice has more irregular contour structure, while the Muon slice varies more smoothly across the plotted region. These plots are qual-

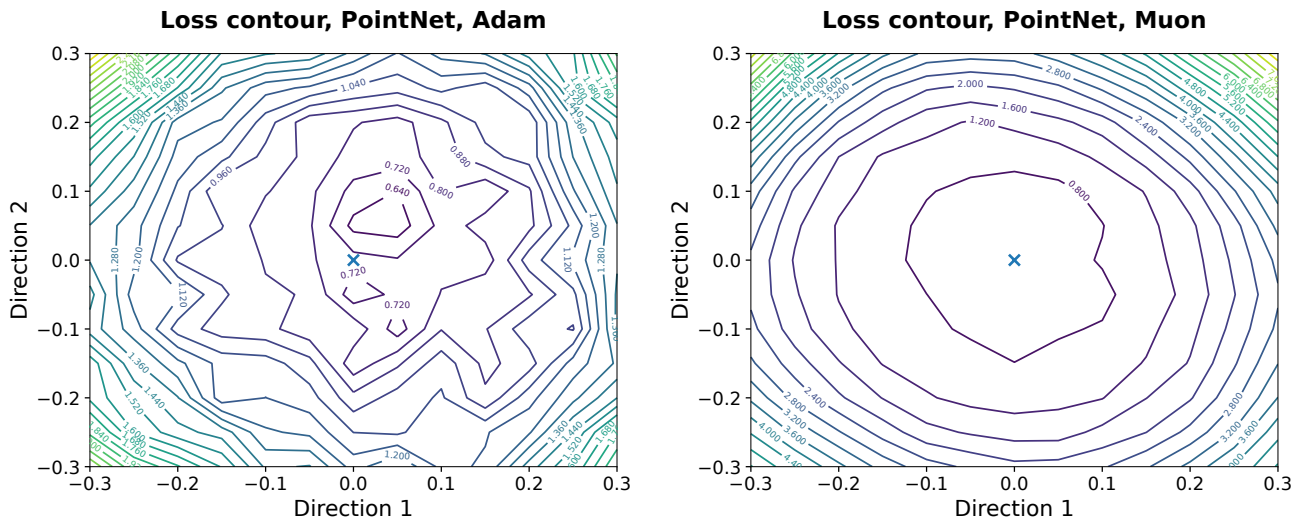


Figure 1. Local loss contours around representative PointNet checkpoints trained with Adam and Muon on ModelNet40. The plots show normalized two-dimensional loss slices around each trained solution; the Muon slice appears smoother, while the Adam slice shows more irregular contour structure.

Table 3. Representative Hessian estimates on clean ModelNet40 checkpoints.

Metric	Model	Adam	Muon	Ratio
Top eig.	EGNN	27.14	128.83	4.75×
Top eig.	PointNet	32.75	714.49	21.82×
Top eig.	DGCNN	12.14	136.23	11.22×
Trace	EGNN	402.37	1472.78	3.66×
Trace	PointNet	482.61	7362.05	15.25×
Trace	DGCNN	184.47	1218.74	6.61×

itative, since they only show a low-dimensional view of the full loss surface. This caution is especially relevant for equivariant models. Xie & Smidt (2025) show that hidden parameter symmetries can split an equivariant model's loss landscape into regions with different minima, so even if a different optimization method makes the landscape easier to traverse locally, it does not change the set of minima defined by the objective. Optimizer choice can nevertheless change the trajectory through the landscape, and therefore which solution is reached. The improvements in Table 1 suggest that Muon can reach better solutions while keeping the equivariant architecture fixed. This is complementary to constraint-relaxation approaches, which modify the training problem to help SGD reach better minima (Pertigkiozoglou et al., 2024; Manolache et al., 2025; Elhag et al., 2025). A plausible hypothesis is that Muon's orthogonalized momentum updates provide a trajectory-level bias toward more favorable regions of the landscape.

We next ask whether the smoother geometry corresponds to lower local curvature. For each trained checkpoint, we compute the loss on a fixed data subset and estimate Hessian summaries using autograd Hessian-vector products. We use power iteration for the top eigenvalue and a Hutchinson estimator with Rademacher probes for the trace (Pearlmutter, 1994; Hutchinson, 1990; Ghorbani et al., 2019).

Table 3 shows that the checkpoints reached by Muon have larger top-eigenvalue and trace estimates for every architecture. This rules out the most direct reading of the loss slices: the smoother contours are not evidence that Muon finds lower-curvature solutions. Around the Muon checkpoints, the sampled loss surface appears more regular, even though the curvature summaries at the checkpoint are larger. This distinction is cautioned by Dinh et al. (2017), who show that common notions of parameter-space sharpness can change under parameter symmetries and reparameterizations without changing the represented function. We therefore interpret the Hessian estimates and loss slices together: optimizer choice changes the local geometry of the reached checkpoint, but the change is not summarized by curvature alone.

2.3. Spectral structure of weights and representations

For each trained checkpoint, we compute two spectral statistics on the singular values $\sigma_1 \geq \sigma_2 \geq \dots$ of every weight matrix W, and analogously of the representations at each layer. The first is the *stable rank*, $\|W\|_F^2/\|W\|_2^2 = \sum_i \sigma_i^2/\sigma_1^2$; the second is the *effective rank*, $\exp(H(p))$, where $p_i = \sigma_i/\sum_j \sigma_j$ and H is the Shannon entropy (Roy & Vetterli, 2007). Both lie in $[1, \mathrm{rank}(W)]$ and are maximized by a uniform spectrum, with smaller values indicating concentration along a few dominant singular directions.

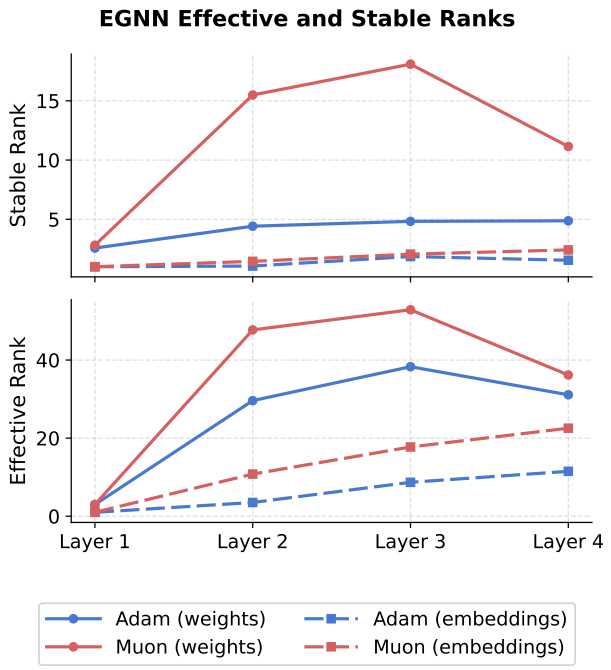


Figure 2. Per-layer stable rank (top) and effective rank (bottom) for EGNN trained on ModelNet40 with Adam (blue) and Muon (red). Solid lines are rank values for trainable weight matrices, with dashed lines from the corresponding intermediate embeddings. Muon produces less concentrated weight and activation spectra than Adam at every layer.

Throughout this section we analyze the trained Adam and Muon checkpoints from Section 2.2.

Weights. Across all three ModelNet40 architectures, the checkpoints reached by Muon have higher stable and effective ranks than Adam at the majority of trainable layers. Figure 2 shows the EGNN case, where this holds at every layer. This contrasts with the low-rank implicit bias of gradient methods observed in other settings (Arora et al., 2019). Muon's orthogonalized momentum step instead treats singular directions of the update more evenly. Jordan et al. (2024) motivate this design with the empirical observation that Adam and SGD-momentum updates for 2D parameters are often near-low-rank, with orthogonalization rescaling the "rare directions" that carry small but important update signal. The higher stable and effective ranks we observe in trained Muon weights are consistent with this picture, but should not be read as explaining Muon's performance on their own (Shumaylov et al., 2026).

Representations. Low representation rank is a known failure mode in deep transformers, where pure self-attention layers with no skip connections drive outputs to rank one (Dong et al., 2021), with associated gradient pathologies during training (Noci et al., 2022). The mechanism is attention-specific and our equivariant architectures differ in important ways, but it motivates examining representation rank in addition to weight rank. We extract activations at every layer of each trained model and compute the same spectral statistics on the resulting feature matrices, mean-pooling over points at intermediate layers and using each architecture's native pooling at the final layer. Figure 2 shows the per-layer comparison for EGNN, where the effect is most visible: Muon's representations stable and effective ranks are consistently higher than Adam's across the network, with roughly a $2\times$ effective-rank difference at the final layer. The same observations hold for DGCNN and PointNet (Appendix Section C), at different magnitudes.

Both analyses thus point in the same direction at both the weight and the representation level, across architectures with quite different geometric inductive biases. We do not establish a causal link from these spectral properties to the accuracy gains in Tables 1 and 2, but the consistency of the effect motivates further investigation of how an optimizer's spectral behavior and a model's geometric constraints interact, and whether equivariant models in particular benefit from optimizers that distribute capacity across singular directions rather than concentrating it.

3. Conclusions and further work

We examined whether the optimizer, independent of architectural relaxations, can change the solutions reached when training equivariant networks. Muon improves over Adam on 3D point-cloud and most molecular tasks, and the trained Muon solutions differ from Adam's in their local loss geometry and in the spectral structure of their weights and intermediate representations. We see these results as preliminary: they identify the optimizer as a meaningful lever for equivariant model training, but the underlying mechanisms remain to be explained.

Several directions follow naturally. First, extending our post-hoc landscape and spectral analyses to training dynamics would show how the differences emerge during training, not only at convergence. Second, a more systematic study of symmetry groups, operations, and datasets is needed: in our experiments the largest gains appear on 3D tasks with $SE(3)$-style equivariance, while permutation-equivariant message passing on graphs benefits less. Third, the spectral results suggest an optimizer-design principle: update rules that keep the singular spectra of trained weights and representations from concentrating in a few directions may improve training in equivariant models. Muon provides one example through orthogonalized momentum updates, but other mechanisms could target this property more directly. More broadly, an optimizer designed for equivariance would exploit the structure of the symmetry group and constrained parameter space, rather than treating equivariance as a black-box constraint at training time.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

Arora, S., Cohen, N., Hu, W., and Luo, Y. *Implicit regularization in deep matrix factorization*. Curran Associates Inc., Red Hook, NY, USA, 2019.

Aykent, S. and Xia, T. Gotennet: Rethinking efficient 3d equivariant graph neural networks. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=5wxCQDtbMo.

Brehmer, J., Behrends, S., de Haan, P., and Cohen, T. Does equivariance matter at scale?, 2025. URL https://arxiv.org/abs/2410.23179.

Bronstein, M. M., Bruna, J., Cohen, T., and Velickovic, P. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *CoRR*, abs/2104.13478, 2021. URL https://arxiv.org/abs/2104.13478.

Dinh, L., Pascanu, R., Bengio, S., and Bengio, Y. Sharp minima can generalize for deep nets. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, pp. 1019–1028. JMLR.org, 2017.

Dong, Y., Cordonnier, J.-B., and Loukas, A. Attention is not all you need: Pure attention loses rank doubly exponentially with depth. In *International Conference on Machine Learning*, 2021. URL https://api.semanticscholar.org/CorpusID:232134936.

Dwivedi, V. P., Rampášek, L., Galkin, M., Parviz, A., Wolf, G., Luu, A. T., and Beaini, D. Long range graph benchmark. In *Thirty-sixth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2022. URL https://openreview.net/forum?id=in7XC5RcjEn.

Elhag, A. A. A., Rusch, T. K., Giovanni, F. D., and Bronstein, M. M. Relaxed equivariance via multitask learning. In *ICLR 2025 Workshop on Machine Learning for Genomics Explorations*, 2025. URL https://openreview.net/forum?id=8kZSO4WbTh.

Ghorbani, B., Krishnan, S., and Xiao, Y. An investigation into neural net optimization via hessian eigenvalue density. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2232–2241. PMLR, 09–15 Jun 2019. URL https://proceedings.mlr.press/v97/ghorbani19b.html.

Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P., and Aspuru-Guzik, A. Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules. *ACS Cent. Sci.*, 4(2):268–276, February 2018. ISSN 2374-7943. doi: 10.1021/acscentsci.7b00572.

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HJlWWJSFDH.

Hutchinson, M. A stochastic estimator of the trace of the influence matrix for laplacian smoothing splines. *Communications in Statistics - Simulation and Computation*, 19(2):433–450, 1990. doi: 10.1080/03610919008812866. URL https://doi.org/10.1080/03610919008812866.

Jordan, K., Jin, Y., Boza, V., You, J., Cesista, F., Newhouse, L., and Bernstein, J. Muon: An optimizer for hidden layers in neural networks, 2024. URL https://kellerjordan.github.io/posts/muon/.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Li, H., Xu, Z., Taylor, G., Studer, C., and Goldstein, T. Visualizing the loss landscape of neural nets. In *Neural Information Processing Systems*, 2018.

Manolache, A., Chamon, L. F. O., and Niepert, M. Learning (approximately) equivariant networks via constrained optimization. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=NM4emKloy6.

Noci, L., Anagnostidis, S., Biggio, L., Orvieto, A., Singh, S. P., and Lucchi, A. Signal propagation in transformers: theoretical perspectives and the role of rank collapse. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA, 2022. Curran Associates Inc. ISBN 9781713871088.

Pascanu, R., Lyle, C., Modoranu, I.-V., Borras, N. E., Alistarh, D., Velickovic, P., Chandar, S., De, S., and Martens, J. Optimizers qualitatively alter solutions and we should leverage this, 2025. URL https://arxiv.org/abs/2507.12224.

Pearlmutter, B. A. Fast exact multiplication by the hessian. *Neural Computation*, 6(1):147–160, 01 1994. ISSN 0899-7667. doi: 10.1162/neco.1994.6.1.147. URL https://doi.org/10.1162/neco.1994.6.1.147.

Pertigkiozoglou, S., Chatzipantazis, E., Trivedi, S., and Daniilidis, K. Improving equivariant model training via constraint relaxation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=tWkL7k1u5v.

Petrache, M. and Trivedi, S. Approximation-generalization trade-offs under (approximate) group equivariance. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=DnO6LTQ77U.

Qi, C. R., Su, H., Mo, K., and Guibas, L. J. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016.

Ramakrishnan, R., Dral, P. O., Rupp, M., and von Lilienfeld, O. A. Quantum chemistry structures and properties of 134 kilo molecules. *Sci. Data*, 1(140022):140022, August 2014. ISSN 2052-4463. doi: 10.1038/sdata.2014.22.

Roy, O. and Vetterli, M. The effective rank: A measure of effective dimensionality. In *2007 15th European Signal Processing Conference*, pp. 606–610, 2007.

Satorras, V. G., Hoogeboom, E., and Welling, M. E(n) equivariant graph neural networks. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 9323–9332. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/satorras21a.html.

Shumaylov, Z., Costa, N. D., Zaika, P., Mucsányi, B., Massucco, A., Gelberg, Y., Schönlieb, C.-B., Gal, Y., and Hennig, P. Muon is not that special: Random or inverted spectra work just as well, 2026. URL https://arxiv.org/abs/2605.11181.

Sun, J., Zhang, Q., Kailkhura, B., Yu, Z., Xiao, C., and Mao, Z. M. Benchmarking robustness of 3d point cloud recognition against common corruptions. *arXiv preprint arXiv:2201.12296*, 2022.

Thomas, N., Smidt, T. E., Kearnes, S., Yang, L., Li, L., Kohlhoff, K., and Riley, P. Tensor field networks: Rotation- and translation-equivariant neural networks for 3d point clouds. *CoRR*, abs/1802.08219, 2018. URL http://dblp.uni-trier.de/db/journals/corr/corr1802.html#abs-1802-08219.

Wang, R., Walters, R., and Yu, R. Approximately equivariant networks for imperfectly symmetric dynamics. In *International Conference on Machine Learning*. PMLR, 2022.

Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., and Solomon, J. M. Dynamic graph cnn for learning on point clouds. *ACM Trans. Graph.*, 38(5), October 2019. ISSN 0730-0301. doi: 10.1145/3326362. URL https://doi.org/10.1145/3326362.

Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., and Xiao, J. 3d shapenets: A deep representation for volumetric shapes. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1912–1920, 2014. URL https://api.semanticscholar.org/CorpusID:206592833.

Xie, Y. and Smidt, T. A tale of two symmetries: Exploring the loss landscape of equivariant models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=rH4aGTL4jY.

Xie, Y., Daigavane, A., Kotak, M., and Smidt, T. The price of freedom: Exploring expressivity and runtime tradeoffs in equivariant tensor products. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=EvIwwGYTLc.

A. More quantitative results

This section collects the additional graph message-passing experiments referenced in the main text. These experiments are intended to test whether the optimizer effect observed on the 3D ModelNet40 and QM9 settings transfers to permutation-equivariant models with graph benchmarks. The best single checkpoints improve under Muon, but the seed-averaged behavior is comparable to Adam.

For both datasets we use the same reporting convention as in the main text. Hyperparameters are selected separately for each optimizer by validation performance, four seeds are then trained with the selected configuration. We report both the best checkpoint observed across seeds and the mean performance across seeds. Peptides-func is evaluated by average precision, where larger is better, while ZINC is evaluated by mean absolute error, where smaller is better.

Table 4. GINE performance on Peptides-func. Results are averaged over four seeds. Higher AP is better. Muon reaches the best checkpoint, while mean performance is comparable to Adam.

Optimizer	Best AP ↑	Mean AP ↑
Adam	0.7098	$0.7021_{\pm 0.0047}$
Muon	**0.7201**	$\mathbf{0.7036}_{\pm 0.0108}$

Table 5. GINE performance on ZINC. Results are averaged over four seeds. Lower MAE is better. Muon reaches the best checkpoint, while Adam has better mean performance.

Optimizer	Best MAE ↓	Mean MAE ↓
Adam	0.1538	$\mathbf{0.1692}_{\pm 0.0153}$
Muon	**0.1425**	$0.1808_{\pm 0.0428}$

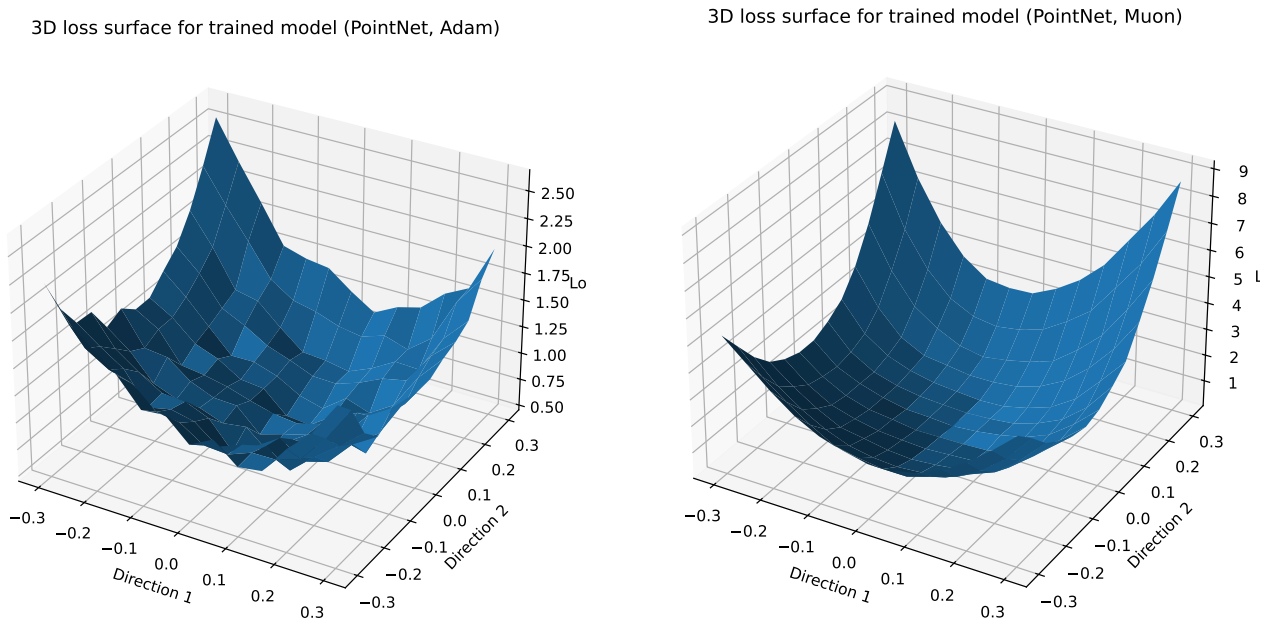
The Peptides-func results from Table 4 are close on average, but Muon obtains the strongest individual checkpoint, with the four-seed mean being only marginally higher than Adam, with larger variance. On ZINC the picture is less favorable to Muon. Although Muon again reaches the best single checkpoint, its average MAE is worse than Adam's. These results support the observations made in the main text: changing the optimizer is clearly useful in the 3D geometric settings we study, but does not offer uniformly reliable improvement across all equivariance setups.

B. Additional loss landscape visualizations

Figures 3 to 5 extend the loss-landscape observations from the main text. They are included as qualitative local views around representative checkpoints. As in the PointNet comparison, each slice is centered at the selected trained checkpoint and uses normalized directions in parameter space.

For DGCNN, the same pattern seen in the PointNet slice is visible at the level of these projections. The Adam slices contain more irregular contour geometry and sharper local variation, whereas the Muon slices are visually more regular over the plotted region.

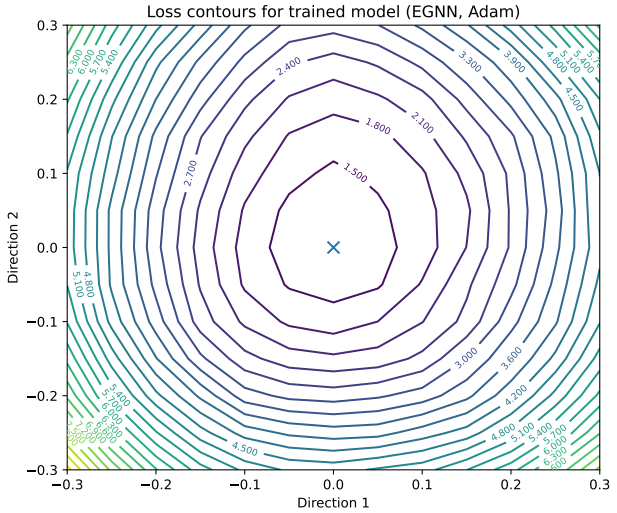
EGNN behaves differently. Both Adam and Muon produce visually regular slices over the plotted region, without the irregular contour structure visible in Adam's PointNet and DGCNN slices. The Hessian summaries in Table 3 still place the Muon EGNN checkpoint at substantially higher top eigenvalue and trace than the Adam one, so the visual similarity does not extend to curvature at the checkpoint. We read this as additional support for the main-text observation that loss-slice plots and curvature summaries probe related, but complementary, and not identical properties of the local geometry.



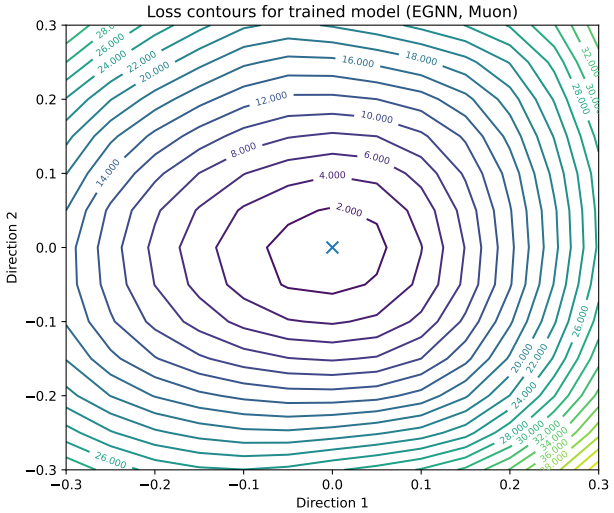
(a) PointNet, Adam 3D surface

(b) PointNet, Muon 3D surface

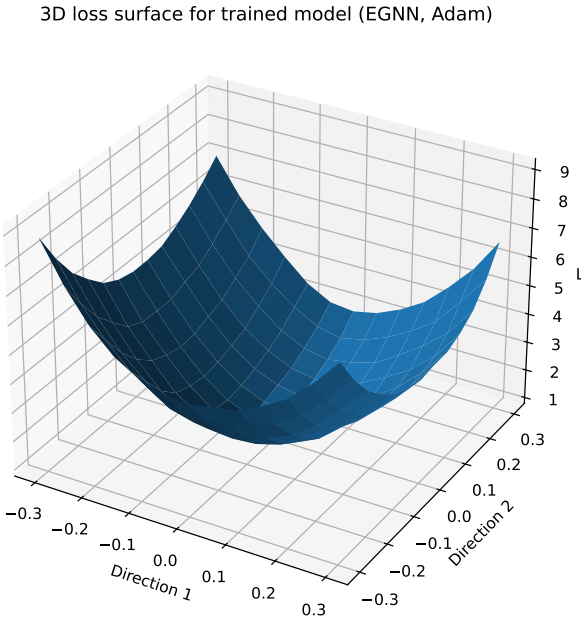
Figure 3. Additional PointNet 3D local loss surfaces around representative Adam and Muon checkpoints. These surfaces correspond to the same local slice construction used for the PointNet contours in the main text.



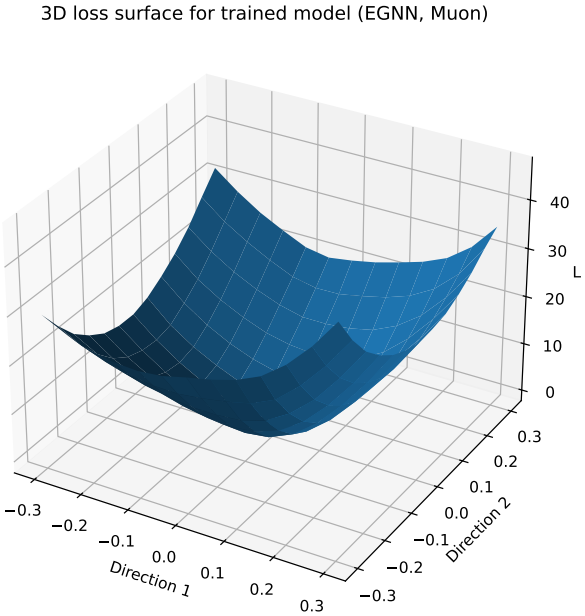
(a) EGNN, Adam contour



(b) EGNN, Muon contour

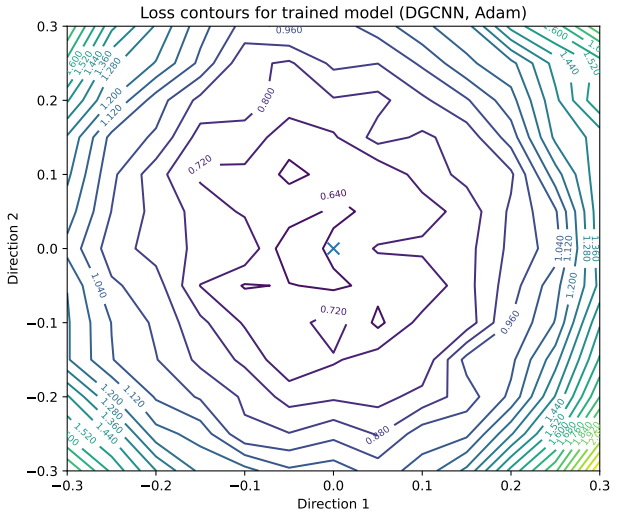


(c) EGNN, Adam surface

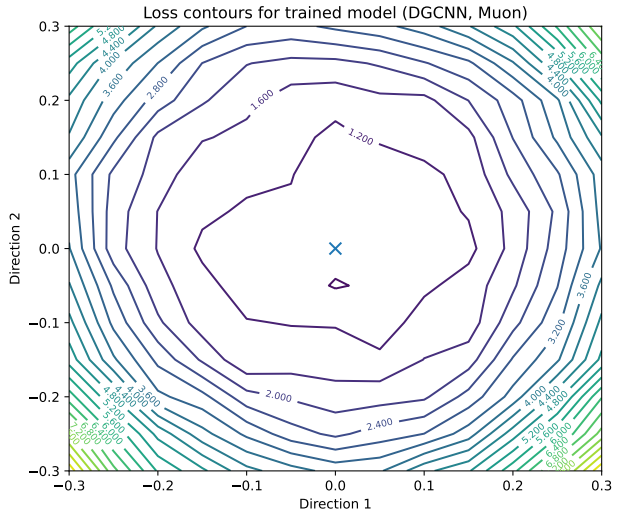


(d) EGNN, Muon surface

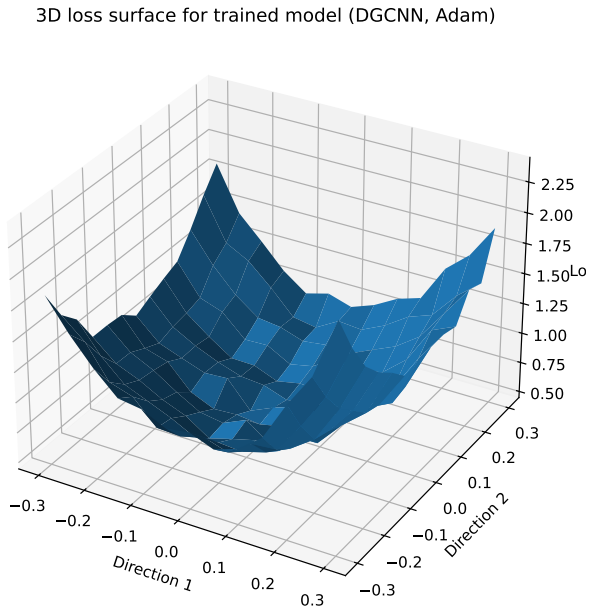
Figure 4. Additional EGNN local loss visualizations around representative Adam and Muon checkpoints. The top row shows 2D contour slices and the bottom row shows the corresponding 3D loss surfaces.



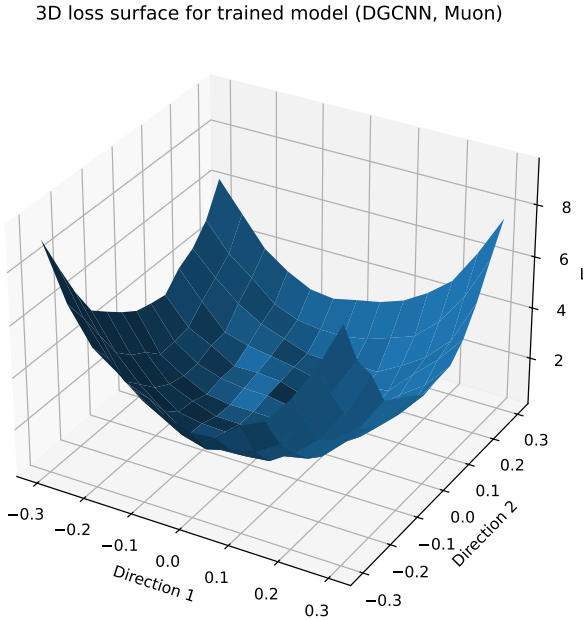
(a) DGCNN, Adam contour



(b) DGCNN, Muon contour



(c) DGCNN, Adam surface

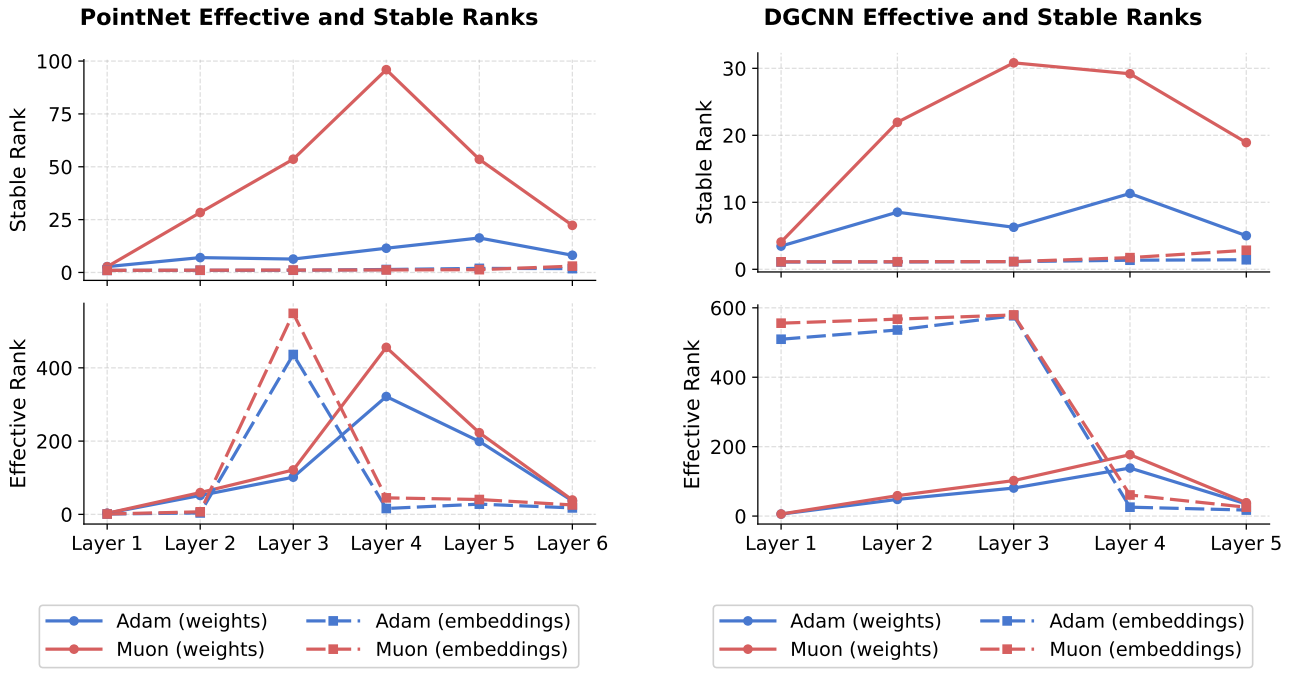


(d) DGCNN, Muon surface

Figure 5. Additional DGCNN local loss visualizations around representative Adam and Muon checkpoints. The top row shows 2D contour slices and the bottom row shows the corresponding 3D loss surfaces.

C. Additional effective ranks

Figure 6 shows the PointNet and DGCNN counterparts of the rank comparison reported for EGNN in the main text. The diagnostic is identical to the one used for EGNN: for trainable weight matrices and intermediate representations, we compute stable rank and entropy-based effective rank from the singular-value spectrum. Higher values indicate that the spectrum is less concentrated in a small number of dominant singular directions.



(a) PointNet *(b)* DGCNN

Figure 6. Additional per-layer stable and effective rank comparisons for weights and intermediate representations. Solid lines show trainable weight matrices and dashed lines show the corresponding representation matrices, computed on the fixed ModelNet40 test subset used in the main text.

For PointNet, Muon increases the rank statistics for both weights and representations across most layers, with the largest visible separation in the weights stable rank, and in the effective rank curves. For DGCNN, the separation is smaller but follows the same direction.