ICLR 2022 Challenge for Computational Geometry & Topology: Design and Results

Adele Myers*
UC Santa Barbara, USA
adele@ucsb.edu

Saiteja Utpala*
UC Santa Barbara, USA

Shubham Talbar*
UC Santa Barbara, USA

Sophia Sanborn*
UC Berkeley, USA

Christian Shewmake*
UC Berkeley, USA

Claire Donnat*
The University of Chicago, USA

Johan Mathe*
Atmo, USA

Umberto Lupo*
EPFL, Switzerland

Rishi Sonthalia**
UCLA, USA

Xinyue Cui**
UCLA, USA

Tom Szwagier**
ENS Paris-Saclay, France

Arthur Pignet**
ENS Paris-Saclay, France

Andri Bergsson**
PayAnalytics Iceland

Søren Hauberg**
Technical University of Denmark, Denmark

Dmitriy Nielsen**
University of Copenhagen, Denmark

Stefan Sommer**
University of Copenhagen, Denmark

David Klindt**
NTNU, Norway

Erik Hermansen**
NTNU, Norway

Melvin Vaupel**
NTNU, Norway

Benjamin Dunn**
NTNU, Norway

Jeffrey Xiong **
Columbia University, USA

Noga Aharony **
Columbia University, USA

Itsik Pe'er **
Columbia University, USA

Felix Ambellan**
Freie Universität Berlin, Germany

Martin Hanik**
Freie Universität Berlin, Germany

Esfandiar Nava-Yazdani**
Zuse Institute Berlin, Germany

Christoph von Tycowicz**
Freie Universität Berlin, Germany

Nina Miolane*
UC Santa Barbara, USA
ninamiolane@ucsb.edu

* : Organizers and external jury; ** : Participants

Abstract

This paper presents the computational challenge on differential geometry and topology that was hosted within the ICLR 2022 workshop "Geometric and Topological Representation Learning". The competition asked par-

ticipants to provide implementations of machine learning algorithms on manifolds that would respect the API of the open-source software Geomstats (manifold part) and Scikit-Learn (machine learning part) or PyTorch. The challenge attracted seven teams in its two month duration. This paper describes the design of the challenge and summarizes its main findings. Code: https://github.com/geomstats/challenge-iclr-2022. DOI: 10.5281/zenodo.6554616.

1 Introduction

Traditional statistics and machine learning have been developed for data that lie on Euclidean vector spaces. Geometric statistics and geometric machine learning extend traditional methods to data that lie on nonlinear spaces such as manifolds. Geometric methods use the geometry of the data space as an inductive bias to guide modeling and analysis. This approach has shown great promise in many application domains, ranging from biomedical imaging to palaeontology. However, to date, the adoption of geometric methods has been limited by the scarce availability of open-source and unit-tested implementations of published algorithms. The challenge described in this white paper aimed to fill that gap.

The purpose of this challenge was to foster reproducible research in geometric machine learning, by crowd-sourcing the open-source implementation of learning algorithms on manifolds, as motivated in the ICLR 2021 challenge's white paper (Miolane et al., 2021). Participants were asked to contribute code for a published or unpublished algorithm, following Scikit-Learn/Geomstats' or PyTorch's APIs and computational primitives, benchmark it, and demonstrate its use in real-world scenarios.

This white paper is organized as follows. Section 2 describes the setup of the challenge, including its guidelines and evaluation criteria. Section 3 summarizes the submissions to the challenge, and Section 4 describes the principle take-aways of the challenge for the development of geometric learning via the software Geomstats. Section 5 provides the final ranking of the submissions.

2 Setup of the challenge

The challenge was held in conjunction with the workshop "Geometric and Topological Representation Learning" of the International Conference on Learning Representations (ICLR) 2022 [1]. Participants were asked to contribute code for a geometric machine learning algorithm, following Scikit-Learn/Geomstats' or PyTorch's APIs and computational primitives, benchmark it, and demonstrate its use in real-world scenarios.

Guidelines Each submission was required to take the form of a Jupyter Notebook. The participants were asked to submit their Jupyter Notebook via Pull Requests (PR) to the GitHub repository of the challenge[2]. This challenge requested Jupyter Notebook submissions because they offer authors a natural way to communicate results that are inherently reproducible, as anyone with access to the notebook may run the notebook to attain the same results.

Teams were accepted and there was no restriction on the number of team members. The principal developers of Geomstats (i.e. the co-authors of Geomstats published papers) were not allowed to participate.

The participants were asked to include the following sections in their submission:

- Introduction: Explain and motivate the choice of learning algorithm.
- Related work and implementations: Contrast the chosen learning algorithms with other algorithms, and describe existing implementations (if any).

[1] Workshop "Geometric and Topological Representation Learning": https://gt-rl.github.io/
[2] Challenge's repository: https://github.com/geomstats/challenge-iclr-2022

- Implementation of the learning algorithm.
- Tests on synthetic datasets and benchmark.
- Application to real-world datasets.

Evaluation criteria: fostering creativity The evaluation criteria were:

1. How "interesting"/"important"/"useful" is the learning algorithm? Note that this is a subjective evaluation criterion, where the reviewers evaluated what the implementation of this algorithm brings to the community (regardless of the quality of the code).
2. How readable/clean is the implementation? How well does the submission respect Scikit-Learn/Geomstats/PyTorch's APIs? If applicable: does it run across backends?
3. Is the submission well-written? Do the docstrings help understand the methods?
4. How informative are the tests on synthetic data sets, the benchmarks, and the real-world application?

Note that these criteria were not aimed to reward new learning algorithms, nor learning algorithms that outperform the state-of-the-art. Instead, these criteria were designed to reward clean code and exhaustive tests that will foster reproducible research in geometric learning.

Software engineering practices The participants were also encouraged to use software engineering best practices. Their code should be compatible with Python 3.8 and make an effort to respect the Python style guide PEP8. The Jupyter notebooks were automatically tested when a Pull Request was submitted and the tests were required to pass. If a dataset was used, the dataset had to be public and referenced. Participants could raise GitHub issues and/or request help or guidance at any time through the Geomstats slack workspace. Help and guidance was be provided modulo availability of the maintainers.

Comparison with the ICLR 2021 challenge The ICLR 2022 challenge described here has key differences with the challenge organized in 2021. The ICLR 2021 challenge was purely open-ended; by contrast, this ICLR 2022 challenge was more guided, as to foster computational developments into a restricted area of computational geometry: machine learning on manifolds.

3 Submissions to the Challenge

Seven teams participated in the challenge by submitting code before the deadline:

- NeuroSEED in Small Open Reading Frame Proteins (NeuroSEED),
- Wrapped Gaussian Process Regression on Riemannian Manifolds (WGPR),
- Riemannian Stochastic Neighbor Embedding (Rie-SNE),
- Topological Ensemble Detection with Differentiable Yoking (TEDDY),
- Sampler for Brownian Motion on Manifolds (Brownian Motion Sampler),
- Hyperbolic Embedding via Tree Learning (Tree Embedding),
- Sasaki Metric and Applications in Geodesic Analysis (Sasaki).

This section provides a summary of the submissions. In these summaries, we group the submissions into three categories:

- General Methods: these submissions implement tools that can be applied to any Riemannian Manifold (WGPR, Rie-SNE, Brownian Motion Sampler).
- Metric-Specific: these submissions focus on implementing a specific metric (Tree Embedding, Sasaki Metric).
- Application Driven: for these submissions, the geometry is present in the data, and a geometric machine learning method has been designed for a particular application (TEDDY, NeuroSEED).

3.1 General Methods

These submissions implement general tools which could theoretically be applied to any manifold.

Riemannian Stochastic Neighbor Embedding (Rie-SNE) Stochastic neighbor embedding (SNE) (Hinton & Roweis, 2003; van der Maaten & Hinton, 2008) aims to represent high-dimensional data clusters in lower dimensional space to make clustering visualization easier. The vast majority of dimensionality reduction techniques assume that data resides on a Euclidean domain, which presents problems when data lie in more complex spaces. This submission developed an extension of SNE that generalizes SNE to Riemannian manifolds (Bergsson & Hauberg, 2022). Rie-SNE reduces the dimensionality of a set of data by "projecting" the high dimensional data onto a lower dimensional manifold, where data is easier to visualize. Rie-SNE does this by using a Riemannian probability distribution rather than a Gaussian one when computing high-dimensional similarities between data points. The team tested their submission on synthetic data (MNIST data points mapped to a 784 dimensionsional sphere) and compared their manifold clustering visualization result to results from a traditional data visualization algorithm: tangent PCA (Fletcher et al., 2004). They found that the clustering structure in every set of data is significantly more evident in Rie-SNE than it is in tangent PCA. Tangent PCA distorted structure while Rie-SNE did not.

Wrapped Gaussian Process Regression on Riemannian Manifolds (WGPR) This submission implements a method for nonlinear regression on Riemannian manifolds. Although regression for datasets lying on Euclidean spaces is a well-established field, not many methods exist for data on Riemannian manifolds; they are either too simple like Geodesic Regression (Thomas Fletcher, 2013), or they lack interpretability. Gaussian Process Regression (GPR) (Goertler et al., 2019) is a well-known nonlinear regression method which incorporates prior knowledge about the data distribution in Euclidean spaces. (Mallasto & Feragen, 2018) generalize it to data lying on Riemannian manifolds using manifold-preserving tools, such as the ones implemented in Geomstats. This submission proposes an implementation of the latter's "Wrapped Gaussian Process Regression" in Geomstats and shows that it has many advantages over related Euclidean and geodesic models on synthetic examples (toy data on a sphere) as well as real life ones (diffusion Tensor Imaging in the corpus callosum).

Sampler for Brownian Motion on Manifolds Brownian motion, also known as the "random walk", is common in physical systems. The scope of this submission is to implement a sampler for Brownian motion with non-trivial covariance on manifolds in Geomstats. In other words, the submission designs a random walk such that the possible (random) positions at an arbitrary time t are distributed according to a distribution of choice (for example, Normal distribution). They achieve this via stochastic development of Euclidean Brownian motion using the bundle of linear frames (the frame bundle). The project demonstrates implementation on a two-dimensional sphere, but their method should work on general manifolds.

Geomstats implementation WGPR has been implemented into Geomstats as a tool for manifold regression. Rie-SNE provides a suggestion for visualizing high-dimensional data clustering. Currently, Geomstats does not utilize stochastic development on any manifolds. The Brownian motion submission proposes one way to do this.

3.2 Metric-Specific

Both the Tree Embedding and the Sasaki Metric submissions proposed implementations of a specific metric. Both submissions can be applied to any Riemannian manifold.

The Tree Embedding submission aims to provide a tool for visualizing tree-structured data in two dimensions. The Tree Embedding team utilized TreeRep (Sonthalia & Gilbert, 2020) (a recent tree learning algorithm) and Sakar's algorithm (Sarkar, 2012) to embed data in

tree structures and then reduce the dimensionality of the data set for ease of visualization. This submission will work best on data sets that lie on manifolds equipped with a metric that is close to a tree metric, but this submission is equipped to handle data with any structure. TreeRep is an algorithm which can either (i) take a tree metric and construct a tree structure or (ii) take a non-tree metric and construct a tree structure that best approximates a tree metric for that data. Their program learns a tree from a distance matrix, which represents data in a metric space. Then, their program embeds (weighted) trees in a 2D hyperbolic space (PoincareDisk from Geomstats) using Sakar's algorithm. The final result helps visualize data as a two-dimensional tree structure. They tested their pipeline on synthetic data generated by taking random points in a high-dimensional hyperbolic manifold. They also tested their pipeline on the Karate Dataset (Girvan & Newman, 2002) and the CS Ph.D. dataset (De Nooy et al., 2018). They showed that the popular "optimization-based methods" take much longer and do not necessarily produce better results.

The Sasaki submission implemented and tested a new SasakiMetric class, which will be implemented as a subclass of the RiemannianMetric class in Geomstats. The SasakiMetric subclass will be a valuable tool in Geomstats because the Sasaki metric is suitable for comparing geodesics on manifolds. It is also a natural choice of Riemannian structure for operating on tangent bundles of manifolds. Because the Sasaki metric is well-suited for comparing distances between geodesics on a manifold, it is extremely useful for comparing longitudinal data. As proof, the group tested their submission by calculating the mean trend for longitudinal rat scull data, from Vilmann's rat data set (Bookstein, 1992). In the study, eight landmark measurements were made on 18 rat sculls at various stages in the rat's life. Thus one "trajectory" in this case corresponds to a sequence of landmark measurements performed on one rat throughout its lifetime. Removing scaling and rotations, this team represented the data as points in the Kendall's shape space. They then used geodesic regression to fit a geodesic to each trajectory, and geodesics were canonically identified with points in the tangent bundle. They used computational primitives of Geomstats to calculate the mean trajectory (with respect to the Sasaki metric on the tangent bundle of Kendall's shape space) across all rats for each landmark, and they showed that this mean was reasonable.

Geomstats implementation The Tree Embedding submission presents a 2D visualization method, which is akin to manifold embedding or Riemannian manifold learning within a metric space. In addition to its tree representation capabilities, this submission could be valuable for future Geomstats implementation by providing tools for generating a 2D hyperbolic space from data in a (tree-like) metric space. The SasakiMetric is currently being implemented in Geomstats as a subclass of RiemannianMetric. It will provide a new geometry that existing Geomstats methods can be run on, which will open additional doors to new machine learning opportunities.

3.3 Application-Driven

Application-driven submissions aim to implement a geometric learning algorithm for a particular type of data with a specific geometric structure. Both the TEDDY and NeuroSEED submissions fit in this category.

TEDDY provides a first proof-of-concept for the problem of clustering based on topological signatures in a dataset. They built an unsupervised clustering algorithm that can separate neural ensembles tuned to distinct latent variable manifolds, including: $\mathbb{T}^2, \mathbb{S}^1, SO(3)$, and $\mathbb{R}^2$. In other words, TEDDY uses topological data analysis (TDA) to identify clusters in the dataset through their topological signatures. Note that this algorithm does not learn the manifold itself, but it does learn the topology of the space. Without this geometric prior, these clusters can be hard to identify, which highlights the importance of geometry in data science. Moreover, the proposed method rests on a continuous (over-parametrized) relaxation to a discrete optimization problem, i.e., clustering. The team applies their method to the clustering of neurons, but the application could extend into many fields. Experiments are performed on simulated data, which they generated using Geomstats.

NeuroSeed targets the field of genetics – more specifically, smORFs. smORFs are small proteins that are used in cells across an enormous range of functions. The "distance between small-proteins" is an important quantity because it will (i) bring us closer to classifying small proteins, and (ii) help us identify which genetic sequence each small protein comes from. Existing methods only analyze "smORFs distances" in Euclidean space. The NeuroSEED submission analyzes smORF distances in hyperbolic space and proves that this approach is more efficient and yields more accurate results. They use Geomstats' PoincareBall (a hyperbolic manifold) to calculate hyperbolic distance with a novel unsupervised manifold embedding method. Their model uses a recurrent neural network to produce an encoder that has learned how to accurately estimate the Levenshtein Distance between two given sequences and embed them onto a given topology. Experiments are performed on the SmORFinder dataset (Durrant & Bhatt, 2021). Their results illustrate that hyperbolic spaces are powerful geometries for representing hierarchies in data.

Geomstats implementation Both submissions are rather tailored to their applications, which makes them extremely powerful for their respective fields. It would be interesting to abstract them and generalize them for integration into Geomstats. For example, TEDDY could be structured to identify a wider range of topologies in datasets. NeuroSEED could help us create a general embedding encoder, whose goal is to respect a distance matrix given as input.

4 Take-Aways for Geomstats

Sampling In contrast to other existing libraries coupling probability and geometry, such as Geoopt (Kochurov et al., 2019), Geomstats does not have a sampling module. The contribution "Brownian motion on manifolds" opens the door to creating such a module. In tandem with a sampling module, we could consider extending Geomstats' probability module that provides probability densities on manifolds.

Data sets While Geomstats provides open-source data sets as resources for program testing, we saw that many participants used data sets from outside sources. This indicates that Geomstats could be improved by providing more sample data sets on manifolds. The submissions of the challenge pulled new open-source manifold-valued data sets that we can use to enrich the library's available data sets:

- Synthetic data sets of neural signals,
- SmORF data sets (which can be embedded in hyperbolic manifolds),
- CS PhD data sets (which can be embedded in hyperbolic manifolds).

Data visualization with manifold embedding Many of the submissions provided excellent examples for manifold embedding. For example, the Tree Embedding submission demonstrated how to embed a data set into a tree metric, and Rie-SNE demonstrated how to visualize high-dimensional data clustering by embedding the clusters into a two-dimensional image. Both serve as excellent resources for visualizing high-dimensional manifold data.

Manifold regression WPGR provides a method that can perform regression on Riemannian manifolds for trajectories that are not geodesics. This is an extremely valuable tool, and the team has implemented this into Geomstats.

New metric implementation The Sasaki team is currently working on implementing their SasakiMetric class in Geomstats as a subclass of the RiemannianMetric class.

5 Final ranking

The Condorcet method was used to rank the submissions and decide on the winners. Each team whose submission respected the guidelines was given one vote in the decision process. The other judges were selected Geomstats maintainers and collaborators.

Each team and each judge was asked to vote based on the evaluation criteria listed earlier in this document. Each team and each judge put in a vote for the three best submissions. Each of the three preferences had to be different: e.g. one could not select the same Jupyter Notebook for both first and second place. The votes remained secret. Only the four highest ranking submissions are published here:

1. Hyperbolic Embedding via Tree Learning,
2. Wrapped Gaussian Process Regression on Riemannian Manifolds,
3. Riemannian Stochastic Neighbor Embedding,
4. Sasaki Metric and Applications in Geodesic Analysis.

Regardless of this final ranking, we would like to stress that all the submissions were of very high quality. We warmly congratulate all the participants.

Author Contributions

Nina Miolane led the organization of the challenge. Nina Miolane, Saiteja Utpala, and Shubham Talbar were responsible for the GitHub repository. Adele Myers analyzed and summarized the results of the challenge. Adele Myers, Saiteja Utpala, Shubham Talbar, Sophia Sanborn, Christian Shewmake, Claire Donnat, Johan Mathe, and Umberto Lupo were the external reviewers in the evaluation process. The remaining authors of this white paper were the participants of the challenge.

Acknowledgments

The authors would like the thank the organizers of the ICLR 2022 workshop "Geometrical and Topological Representation Learning" for their valuable support in the organization of the challenge and specifically Bastian Rieck for his availability and help.

6 Conclusion

This white paper presented the motivations behind the organization of the "Computational Geometric and Topological Challenge" at the ICLR 2022 workshop "Geometric and Topological Representation Learning" and summarized the findings from the participants' submissions.

The submissions implemented methods for: data regression on manifolds (WGPR), manifold visualization through manifold embedding (Rie-SNE and Tree Embedding), clustering through manifold embedding (NeuroSEED), clustering through an unsupervised topology learning algorithm (TEDDY), and sampling from Brownian motion. Another submission implemented a new SasakiMetric class in Geomstats (Sasaki) and showcased learning algorithms on it.

References

Andri Bergsson and Søren Hauberg. Visualizing Riemannian data with Rie-SNE. In arXiv preprint, 2022.

Fred L. Bookstein. Morphometric Tools for Landmark Data. Cambridge University Press, January 1992. doi: 10.1017/cbo9780511573064.

Wouter De Nooy, Andrej Mrvar, and Vladimir Batagelj. Exploratory Social Network Analysis with Pajek: Revised and Expanded Edition for Updated Software. Structural Analysis in the Social Sciences. Cambridge University Press, 3 edition, 2018. doi: 10.1017/9781108565691.

Matthew G. Durrant and Ami S. Bhatt. Automated prediction and annotation of small open reading frames in microbial genomes. Cell Host &Microbe, 29(1):121–131.e4, January 2021. doi: 10.1016/j.chom.2020.11.002. URL https://doi.org/10.1016/j.chom.2020.11.002.

P.T. Fletcher, C. Lu, S.M. Pizer, and S. Joshi. Principal geodesic analysis for the study of nonlinear statistics of shape. IEEE Transactions on Medical Imaging, 23(8):995–1005, August 2004. doi: 10.1109/tmi.2004.831793.

M. Girvan and M. E. J. Newman. Community Structure in Social and Biological Networks. Proceedings of the National Academy of Sciences, 99(12):7821–7826, 2002. doi: 10.1073/pnas.122653799.

Jochen Goertler, Rebecca Kehlbeck, and Oliver Deussen. A visual exploration of gaussian processes. Distill, 2019. doi: 10.23915/distill.00017. https://distill.pub/2019/visual-exploration-gaussian-processes.

Geoffrey E Hinton and Sam Roweis. Stochastic neighbor embedding. In S. Becker, S. Thrun, and K. Obermayer (eds.), Advances in Neural Information Processing Systems, volume 15, pp. 857–864. MIT Press, 2003.

Maxim Kochurov, Rasul Karimov, and Serge Kozlukov. Geoopt: Riemannian Adaptive Optimization Methods with pytorch optim, 2019. URL https://github.com/geoopt/geoopt.

Anton Mallasto and Aasa Feragen. Wrapped Gaussian Process Regression on Riemannian Manifolds. In 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5580–5588, June 2018. doi: 10.1109/CVPR.2018.00585. ISSN: 2575-7075.

Nina Miolane, Matteo Caorsi, Umberto Lupo, Marius Guerard, Nicolas Guigui, Johan Mathe, Yann Cabanes, Wojciech Reise, Thomas Davies, António Leitão, Somesh Mohapatra, Saiteja Utpala, Shailja Shailja, Gabriele Corso, Guoxi Liu, Federico Iuricich, Andrei Manolache, Mihaela Nistor, Matei Bejan, Armand Mihai Nicolicioiu, Bogdan-Alexandru Luchian, Mihai-Sorin Stupariu, Florent Michel, Khanh Dao Duc, Bilal Abdulrahman, Maxim Beketov, Elodie Maignant, Zhiyuan Liu, Marek Černý, Martin Bauw, Santiago Velasco-Forero, Jesus Angulo, and Yanan Long. Iclr 2021 challenge for computational geometry & topology: Design and results, 2021.

Rik Sarkar. Low distortion Delaunay embedding of trees in hyperbolic plane. In Graph drawing, volume 7034 of Lecture Notes in Comput. Sci., pp. 355–366. Springer, Heidelberg, 2012. doi: 10.1007/978-3-642-25878-7_34.

Rishi Sonthalia and Anna Gilbert. Tree! I am no Tree! I am a Low Dimensional Hyperbolic Embedding. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), Advances in Neural Information Processing Systems, volume 33, pp. 845–856. Curran Associates, Inc., 2020.

P. Thomas Fletcher. Geodesic Regression and the Theory of Least Squares on Riemannian Manifolds. International Journal of Computer Vision, 105(2):171–185, November 2013. ISSN 1573-1405. doi: 10.1007/s11263-012-0591-y.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. Journal of Machine Learning Research, 9:2579–2605, 2008.