Motivation

Data on manifolds:
- Geographic data
- Interaction graphs
- Correlation matrices
- Rigid transformations, 3D frames, etc.

Parameter Space as manifolds:
- Gaussian family: SPD(n)
- Beta distributions: \( \mathbb{S}^2 \)
- Russian manifolds

Learning that leverages the geometry of the data/parameter space

Cities coordinates

Karaté Club Network

Brain connectomes

Data on the sphere

Healty connect

Schizophrenic

Figure 1: Random samples of SPD matrices from three classes.

Figure 2: Geodesic grid on Poincaré disk.

Figure 3: Geodesic grid on 2D manifold of beta distributions.

Figure 4: Embedding of the histograms in the 2D manifold of beta distributions.

Figure 5: Schild's ladder: a numerical scheme to approach parallel transport (Guigui and Pennec, 2000).

Geometric Learning and Information Geometry

Support research in Geometric Learning and Information Geometry

Teach "hands-on" Geometry, Learning and Information Geometry

Democratize the use of Geometric Learning and Information Geometry in applications

Support research in Geometric Learning and Information Geometry

Objective

Generalized definition of the mean: \( x = \arg \min_{x \in M} \sum_{i=1}^{n} \text{dist}(x, x_i)^2 \)

\( x \in M \)


References


Stefan Heyder, Olivier Pietre, Niklas Koep, Yann Cabanes, Thomas Gerald, Paul Chauchat, Daniel Brooks, Bernhard Kainz, Claire Donnat, Susan Holmes, Xavier Pennec

1Université Côte d’Azur, Inria Sophia Antipolis, Epione Research Project, France.

Keywords: Riemannian Geometry, Python, Open-Source

Geometrics: A Python Package for Geometry in Machine Learning and Information Geometry


Holmes, Xavier Pennec

Geometrics is an open-source Python package for computations and statistics on nonlinear manifolds (Molane et al., 2020). Geometrics relies on three different back-ends, scikit-learn, pytorch and tensorflow, with a generic common API.

from geomstats.geometry.beta_distributions import BetaDistributions

beta = BetaDistributions()

embeddings = beta.maximum_likelihood_fit(samples)

beta = BetaDistributions()

embeddings = beta.maximum_likelihood_fit(samples)

from geomstats.geometry.beta_distributions import BetaDistributions

beta = BetaDistributions()

embeddings = beta.maximum_likelihood_fit(samples)

Beta distributions to classify histograms

We use maximum likelihood to fit beta distributions to histograms of cortical thickness maps (Brigit et al., 2020). The manifold of beta distributions endowed with the Fisher-Rao metric has negative curvature. Geodesic distances can be computed numerically to apply k-near neighbor or k-means algorithms.

References
