Generalizing Point Embeddings using the Wasserstein Space of Elliptical Distributions

Boris MUZELLEC (ENSAE ParisTech) *Marco CUTURI (ENSAE ParisTech & Google Brain)*

Goal: Extend point/vector embeddings to probability distributions to represent uncertainty and variance.

Approach: Use elliptical distributions in the Wasserstein geometry. Distances and gradients in closed form and numerically stable.

Results: Projection-free, GPU friendly optimization algorithms for the Wasserstein-Bures metric. Trained embeddings competitive with state of the art on various tasks.

I. Probabilistic Embeddings

? How to make the most of given dimensionality d ?

- New trend: embed data as probability measures
  - Represents uncertainty & variance
  - Point embeddings are a particular case

[Vilnis & McCallum'15]: (Diag) Gaussians in KL or ℓ_2 geometry

Problems:
- numerical stability
- handling degeneracy

II. The Wasserstein-Bures Metric: A Riemannian metric for elliptical distributions

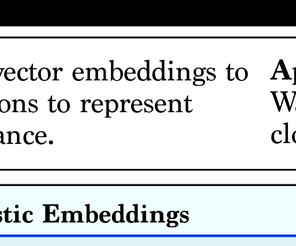
Elliptical Distributions

Def. Probability measures with elliptically contoured densities

$$f(\mathbf{x}) = \frac{1}{\sqrt{|\mathbf{C}|}} h((\mathbf{x}-\mathbf{m})^\top \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})),$$

$$\int_{\mathbb{R}^d} h(\|\mathbf{x}\|^2)\mathrm{d}\mathbf{x} = 1,\ \mathbf{C} \in \mathrm{PSD}_n$$

Ex: Multivariate distributions
Elliptical uniform
(Multivariate) t-Student...



Wasserstein Distance

$$W_2^2(\mu,\nu) \overset{\text{def}}{=} \inf_{X\sim\mu, Y\sim\nu} \mathbb{E}\|X-Y\|_2^2$$

Wasserstein Distance for Elliptical Distributions

- Closed-form for elliptical distributions:

$$W_2^2(\alpha,\beta) = \|\mathbf{m}_\alpha - \mathbf{m}_\beta\|_2^2 + \mathfrak{B}^2(\mathrm{var}\alpha, \mathrm{var}\beta)$$

Bures: a Riemannian metric for PSD matrices

$$\mathfrak{B}^2(\mathbf{A},\mathbf{B}) \overset{\text{def}}{=} \mathrm{Tr}(\mathbf{A}+\mathbf{B}-2(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{\frac12})$$

III. The Bures Geometry

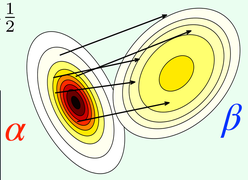
Monge Maps

- Transport map $T_{\alpha\beta}: x \to \mathbf{m}_\beta + \mathbf{T}^{\mathbf{AB}}(x-\mathbf{m}_\alpha)$

$$\mathbf{T}^{\mathbf{AB}} \overset{\text{def}}{=} \mathbf{A}^{-\frac12}(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{\frac12}\mathbf{A}^{-\frac12}$$

- Monge formulation:

$$W_2^2(\alpha,\beta) = \int_{\mathbb{R}^d} \|x - T_{\alpha\beta}(x)\|^2 \mathrm{d}\alpha(x)$$



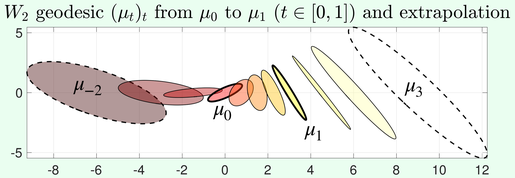
Bures Gradient & Geodesics

$$\nabla_{\mathbf{A}}\mathfrak{B}^2(\mathbf{A},\mathbf{B}) = \mathbf{I} - \mathbf{T}^{\mathbf{AB}}$$

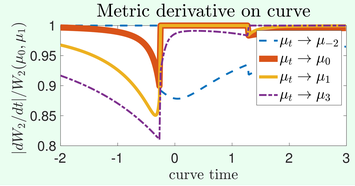
- Riemannian geodesics:

$$\mathbf{C}_{\mathbf{AB}}(t) = [(1-t)\mathbf{I} - t\mathbf{T}^{\mathbf{AB}}]\mathbf{A}[(1-t)\mathbf{I} - t\mathbf{T}^{\mathbf{AB}}]$$

W_2 geodesic $(\mu_t)_t$ from μ_0 to μ_1 ($t \in [0,1]$) and extrapolation



Metric derivative on curve



Euclidean Parametrization $\mathbf{A} = \mathbf{L}_{\mathbf{A}}\mathbf{L}_{\mathbf{A}}^\top$

- Avoid projecting on PSD_n:

$$\Phi(\mathbf{L}_{\mathbf{A}}) \overset{\text{def}}{=} \mathbf{L}_{\mathbf{A}}\mathbf{L}_{\mathbf{A}}^\top$$

Projection-free gradient descent

- Gradient descent updates:

$$\mathbf{L}_{\mathbf{A}} - t\nabla_{\mathbf{L}_{\mathbf{A}}}\mathfrak{B}^2(\mathbf{A},\mathbf{B}) \in \Phi^{-1}\{\mathbf{C}_{\mathbf{AB}}(t)\}$$

Gradient Descent = Walking on the geodesics

IV. Optimizing the Bures metric

Computing $\nabla_{\mathbf{A}}\mathfrak{B}^2(\mathbf{A},\mathbf{B})$ and $\nabla_{\mathbf{B}}\mathfrak{B}^2(\mathbf{A},\mathbf{B})$

Goal: minimize # of root computations

Naive way: 4 runs of Alg. 1

Use equivalent formula for $\mathbf{T}^{\mathbf{AB}}$

$$\left.\begin{matrix}\{\mathbf{A}^{\frac12}, \mathbf{A}^{-\frac12}\} \\ \{(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{\frac12}, (\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{-\frac12}\}\end{matrix}\right\} \text{Computed for } \mathfrak{B}^2(\mathbf{A},\mathbf{B})$$

$$\mathbf{T}^{\mathbf{AB}} = \mathbf{A}^{-\frac12}(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{\frac12}\mathbf{A}^{-\frac12} \qquad \mathbf{T}^{\mathbf{BA}} = \mathbf{A}^{\frac12}(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{-\frac12}\mathbf{A}^{\frac12}$$

$$\nabla_{\mathbf{A}}\mathfrak{B}^2(\mathbf{A},\mathbf{B}) \qquad \nabla_{\mathbf{B}}\mathfrak{B}^2(\mathbf{A},\mathbf{B})$$

Gradients without computing new roots

Computing matrix roots (and their inverses)

Problem: SVD not GPU friendly

- Batch approximation using only matrix-matrix products:

Algorithm 1 Newton-Schulz Square Root

Input: PSD matrix $\mathbf{A}$, $\epsilon > 0$
$\mathbf{Y} \leftarrow \frac{\mathbf{A}}{(1+\epsilon)\|\mathbf{A}\|}$, $\mathbf{Z} \leftarrow \mathbf{I}$
while not converged **do**
  $\mathbf{T} \leftarrow (3\mathbf{I} - \mathbf{Z}\mathbf{Y})/2$
  $\mathbf{Y} \leftarrow \mathbf{Y}\mathbf{T}$
  $\mathbf{Z} \leftarrow \mathbf{T}\mathbf{Z}$
end while
$\mathbf{Y} \leftarrow \sqrt{(1+\epsilon)\|\mathbf{A}\|}\mathbf{Y}$
$\mathbf{Z} \leftarrow \frac{\mathbf{Z}}{\sqrt{(1+\epsilon)\|\mathbf{A}\|}}$
Output: square root $\mathbf{Y}$, inverse square root $\mathbf{Z}$

GPU friendly

Root + Inverse: two for one!

Wasserstein-Bures Similarity Measures

- Polarization: $[\alpha : \beta] := \langle \mathbf{m}_\alpha, \mathbf{m}_\beta \rangle + \mathrm{Tr}(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{\frac12}$
- Sum of cosines: $\mathfrak{S}(\alpha,\beta) := \frac{\langle \mathbf{m}_\alpha, \mathbf{m}_\beta \rangle}{\|\mathbf{m}_\alpha\|\|\mathbf{m}_\beta\|} + \frac{\mathrm{Tr}(\mathbf{A}^{\frac12}\mathbf{B}\mathbf{A}^{\frac12})^{\frac12}}{\sqrt{\mathrm{Tr}\mathbf{A}\mathrm{Tr}\mathbf{B}}}$

V. Applications: Representing objects using elliptical distributions

Word Embeddings

- Extract (word, context) pairs:

ALL	MODELS	ARE	WRONG	BUT	SOME	ARE USEFUL
ALL	MODELS	ARE	WRONG	BUT	SOME	ARE USEFUL
ALL	MODELS	ARE	WRONG	BUT	SOME	ARE USEFUL

- Optimize hinge loss:

$$\min \sum_w \sum_{c\in\mathrm{Pos}(w)} [M - [\mu_w : \nu_c] + \frac{1}{n} \sum_{c'\in\mathrm{Neg}(w)} [\mu_w : \nu_{c'}]]_+$$

- Datasets: ukWaC + WaCkypedia

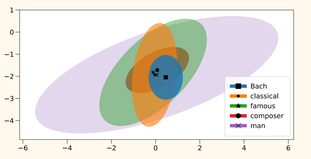


Table 1: Spearman rank correlation scores on similarity datasets, compared with [Vilnis & McCallum'15]

Dataset	W2G/45/C	Ell/12/CM
SimLex	**25.09**	24.09
WordSim	53.45	**66.02**
WordSim-R	61.70	**71.07**
WordSim-S	48.99	**60.58**
MEN	65.16	**65.58**
MC	59.48	**65.95**
RG	**69.77**	65.58
YP	**37.18**	25.14
MT-287	**61.72**	59.53
MT-771	**57.63**	56.78
RW	**40.14**	29.04

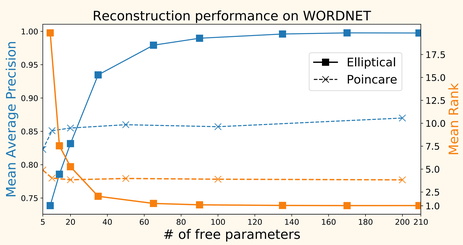
- Use $\mathfrak{S}(\alpha,\beta)$ for retrieval

Hypernymy

- DAG structure: 'A is a hypernym of B iff any B is a A'
  e.g. 'mammal' $\geq_{\mathrm{H}}$ 'dog'

$$\min \sum_u \sum_{v\in\mathrm{Hyp}^{-1}(u)} \log \frac{e^{[\mu_u : \mu_v]}}{e^{[\mu_u : \mu_v]} + \sum_{v'\notin\mathrm{Hyp}^{-1}(u)} e^{[\mu_u : \mu_{v'}]}}$$

- Use $[\alpha : \beta]$ for retrieval



https://github.com/BorisMuzellec/EllipticalEmbeddings