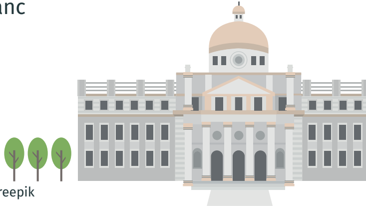
On-line Kronecker Product Structured Covariance Estimation with Riemannian geometry for t-distributed data

Florent Bouchard[1], Arnaud Breloy[2], **Ammar Mian**[3], Guillaume Ginolhac[3]

[1]: L2S, CentraleSupélec, CNRS, Univ. Paris Saclay
[2]: LEME, Univ. Paris Nanterre
[3]: LISTIC, Univ. Savoie Mont-Blanc



@Freepik

Outline

Outline

Kronecker structure and hetergoeneous clutter

Kronecker structure of data arises in numerous applications:

- MIMO : [YBO+04]
- MEG/EEG data : [dMHWH02]
- Space Time Adaptive Processing: [GZH16]
- Synthetic Aperture Radar : [MOAG19]

Moreover, when resolution of data is high (in radar), the data is **heterogeneous** and modeled by heavy-tailed distributions.

The model

The data set $\{\boldsymbol{x}_i\}_{i=1}^n \in (\mathbb{R}^p)^n$ is assumed to contain independent and identically distributed vectors drawn from the multivariate real Student t-distribution with **unknown** scatter matrix $\boldsymbol{\Sigma}$ and **known** $d \in \mathbb{N}^*$ degrees of freedom. The model, denoted $\boldsymbol{x} \sim \mathbb{R}t_d(\boldsymbol{0}, \boldsymbol{\Sigma})$, implies that the probability density function of $\boldsymbol{x}$ is of the form

$$f(\boldsymbol{x}) \propto |\boldsymbol{\Sigma}|^{-1/2} \left(1 + \frac{\boldsymbol{x}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{x}}{d}\right)^{-(d+p)/2} . \quad (1)$$

Kronecker structure

The scatter matrix $\boldsymbol{\Sigma}$ is assumed to admit a Kronecker product structure, *i.e.*, $\boldsymbol{\Sigma} = \boldsymbol{A} \otimes \boldsymbol{B}$, where $\boldsymbol{A} \in s\mathcal{S}_a^{++} = \{\boldsymbol{M} \in \mathcal{S}_a^{++} : |\boldsymbol{M}| = 1\}$ and $\boldsymbol{B} \in \mathcal{S}_b^{++}$.

Estimation problem

Parameter of interest: $\theta = (\boldsymbol{A}, \boldsymbol{B})$

Maximum likelihood estimation

$$\hat{\theta} = \underset{\theta \in (s\mathcal{S}_a^{++} \times \mathcal{S}_b^{++})}{\operatorname{argmin}} - \sum_{i=1}^{n} \log f(\boldsymbol{x}_i; \theta) \tag{2}$$

- Problem is non-convex in Euclidean sense but iterative algorithms exists [SBP16, MRB+21].
- On the other hand, considering the parameter space as the product manifold $\mathcal{M}_{a,b} = s\mathcal{S}_a^{++} \times \mathcal{S}_b^{++}$, it is geodesic-convex.

Problems to tackle

- High-dimensionality of Kronecker products is costly
- Online estimation

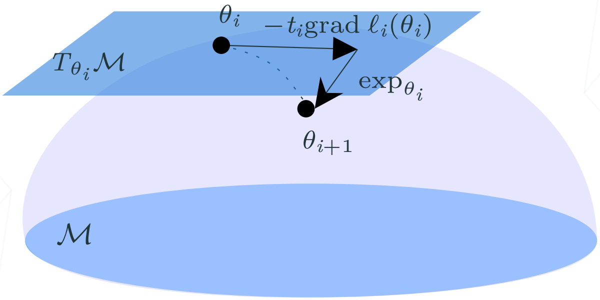
Solution: Use a Riemannian recursive framework [ZS19]

Stochastic gradient on manifolds allow to obtain fast and efficient estimation of the parameters. The scheme is as follows:

$$\theta_{i+1} = \exp_{\theta_i}\left(-t_i \mathrm{grad} \ell_i(\theta_i)\right), \tag{3}$$

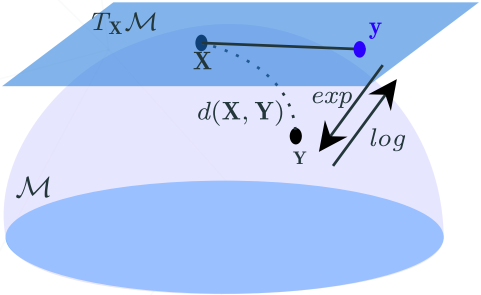
where $\ell_i(\theta) = -\log f(\mathbf{x}_i, \theta)$.

Recursive estimation illustration



Outline

What we need



Information geometry

We can use geometry of product manifold without taking into account the statistical model but not as efficient[Ama99].
$\longrightarrow$ Derivation of Fisher Information metric.

Gradient

We also need the **Riemannian gradient** of the likelihood $\ell_i(\theta)$.

Derivation of metric

Proposition

Given $\theta \in \mathcal{M}_{a,b}$, ξ and $\eta \in T_\theta \mathcal{M}_{a,b}$, the Fisher information metric on $\mathcal{M}_{a,b}$ induced by the likelihood is

$$\langle \xi, \eta \rangle_\theta = \alpha b \operatorname{tr}(\boldsymbol{A}^{-1} \boldsymbol{\xi}_{\boldsymbol{A}} \boldsymbol{A}^{-1} \boldsymbol{\eta}_{\boldsymbol{A}}) + \alpha a \operatorname{tr}(\boldsymbol{B}^{-1} \boldsymbol{\xi}_{\boldsymbol{B}} \boldsymbol{B}^{-1} \boldsymbol{\eta}_{\boldsymbol{B}}) + (\alpha - 1) a^2 \operatorname{tr}(\boldsymbol{B}^{-1} \boldsymbol{\xi}_{\boldsymbol{B}}) \operatorname{tr}(\boldsymbol{B}^{-1} \boldsymbol{\eta}_{\boldsymbol{B}}), \quad (4)$$

where $\alpha = {}^{(d+p)}/_{(d+p+1)}$.

Proof: Derived from results of [BBG+21, Proposition 7] on mappings and Fisher information metric of elliptical distributions in [BGRB19].

Exponential mapping and retraction

the Riemannian exponential mapping at $\theta \in \mathcal{M}_{a,b}$ is defined for $\xi \in T_\theta \mathcal{M}_{a,b}$ as

$$\exp_\theta^{\mathcal{M}_{a,b}}(\xi) = \left(\boldsymbol{A} \exp(\boldsymbol{A}^{-1}\boldsymbol{\xi}_{\boldsymbol{A}}), \boldsymbol{B} \exp(\boldsymbol{B}^{-1}\boldsymbol{\xi}_{\boldsymbol{B}})\right). \quad (5)$$

For better numerical cost and stability, it might be advantageous to prefer a second order approximation:

Retraction

$$R_\theta(\xi) = \left(\boldsymbol{A} + \boldsymbol{\xi}_{\boldsymbol{A}} + \frac{1}{2}\boldsymbol{\xi}_{\boldsymbol{A}}\boldsymbol{A}^{-1}\boldsymbol{\xi}_{\boldsymbol{A}}, \boldsymbol{B} + \boldsymbol{\xi}_{\boldsymbol{B}} + \frac{1}{2}\boldsymbol{\xi}_{\boldsymbol{B}}\boldsymbol{B}^{-1}\boldsymbol{\xi}_{\boldsymbol{B}}\right). \quad (6)$$

Riemannian gradient

Proposition

The Riemannian gradient of ℓ_i at $\theta \in \mathcal{M}_{a,b}$ according to the Fisher information metric is:

$$\operatorname{grad} \ell_i(\theta) = \left(\frac{1}{\alpha b} P_{\boldsymbol{A}}(\boldsymbol{A} \operatorname{sym}(\nabla_{\boldsymbol{A}} \ell_i(\theta)) \boldsymbol{A}), \frac{1}{\alpha a} \boldsymbol{B} \operatorname{sym}(\nabla_{\boldsymbol{B}} \ell_i(\theta)) \boldsymbol{B} - \frac{(\alpha - 1) \operatorname{tr}(\boldsymbol{B} \nabla_{\boldsymbol{B}} \ell_i(\theta))}{\alpha(\alpha + (\alpha - 1)p)} \boldsymbol{B} \right), \tag{7}$$

where $\operatorname{sym}(\cdot)$ returns the symmetrical part of its argument; $P_{\boldsymbol{A}} : \mathcal{S}_a \to T_{\boldsymbol{A}} s\mathcal{S}_a^{++}$ is the orthogonal projection map such that $P_{\boldsymbol{A}}(\boldsymbol{\xi}_{\boldsymbol{A}}) = \boldsymbol{\xi}_{\boldsymbol{A}} - \frac{\operatorname{tr}(\boldsymbol{A}^{-1}\boldsymbol{\xi}_{\boldsymbol{A}})}{a} \boldsymbol{A}$; and $\nabla \ell_i(\theta) = (\nabla_{\boldsymbol{A}} \ell_i(\theta), \nabla_{\boldsymbol{B}} \ell_i(\theta))$ is the Euclidean gradient of ℓ_i at θ, defined as

$$\nabla_{\boldsymbol{A}} \ell_i(\theta) = \frac{1}{2} \boldsymbol{A}^{-1} \left(b\boldsymbol{A} - \frac{d+p}{d + Q_i(\theta)} \boldsymbol{M}_i^T \boldsymbol{B}^{-1} \boldsymbol{M}_i \right) \boldsymbol{A}^{-1}, \nabla_{\boldsymbol{B}} \ell_i(\theta) = \frac{1}{2} \boldsymbol{B}^{-1} \left(a\boldsymbol{B} - \frac{d+p}{d + Q_i(\theta)} \boldsymbol{M}_i \boldsymbol{A}^{-1} \boldsymbol{M}_i^T \right) \boldsymbol{B}^{-1},$$

with $\boldsymbol{M}_i$, the $b \times a$ matrix such that $\operatorname{vec}(\boldsymbol{M}_i) = \boldsymbol{x}_i$ and $Q_i(\theta) = \operatorname{tr}(\boldsymbol{A}^{-1} \boldsymbol{M}_i^T \boldsymbol{B}^{-1} \boldsymbol{M}_i)$.

Recursive algorithm

Algorithm 1: Online estimation of θ

Result: Estimate $\theta = (\boldsymbol{A}, \boldsymbol{B})$
initialization with $\theta = (\boldsymbol{A}_0, \boldsymbol{B}_0)$;
for *i=1,...,n* **do**
  $\theta_i = R_{\theta_i}\left(-\frac{1}{i}\mathrm{grad}\ell_i(\theta_i)\right)$
end

Outline

Setup of Montecarlo i

100 sets $\{\boldsymbol{x}_i\}_{i=1}^n$ are drawn from the multivariate Student t-distribution with covariance $\boldsymbol{\Sigma}$ and $d \in \{3, 100\}$ degrees of freedom, where $n \in [\![1, 500]\!]$. To generate a $\boldsymbol{\Sigma}$:

$$\begin{gathered} \boldsymbol{\Sigma} = \boldsymbol{A} \otimes \boldsymbol{B}, \\ \boldsymbol{A} = \boldsymbol{U_A} \boldsymbol{\Lambda_A} \boldsymbol{U_A}^T, \qquad \boldsymbol{B} = \boldsymbol{U_B} \boldsymbol{\Lambda_B} \boldsymbol{U_B}^T, \end{gathered} \tag{8}$$

where $a = b = 4$,

- $\boldsymbol{U_A}$ and $\boldsymbol{U_B}$ are random orthogonal matrices,
- $\boldsymbol{\Lambda_A}$ and $\boldsymbol{\Lambda_B}$ are diagonal matrices whose minimal and maximal elements are $1/\sqrt{c}$ and $\sqrt{c}$ ($c = 10$ is the condition number with respect to inversion); their other elements are randomly drawn from the uniform distribution between $1/\sqrt{c}$ and $\sqrt{c}$; the determinant of $\boldsymbol{\Lambda_A}$ is then normalized.

Setup of Montecarlo ii

For this experiment, we consider the following estimators:

- the classical **maximum-likelihood estimator** obtained with Riemannian gradient descent (GD). Optimization for this estimator is performed with manopt toolbox [BMAS14].
- the **online version** obtained through stochastic gradient descent (SGD) presented here.

Setup of Montecarlo iii

Both algorithms are initialized with $\theta_0 = (\boldsymbol{I}_a, \boldsymbol{I}_b)$.

In order to measure the performance, we consider an error measure for each component $\boldsymbol{A}$ and $\boldsymbol{B}$, which are given by the usual Riemannian distances on $s\mathcal{S}_a^{++}$ and $\mathcal{S}_b^{++}$

$$\begin{aligned} \mathrm{err}(\widehat{\boldsymbol{A}}) &= \|\log(\boldsymbol{A}^{-1/2}\widehat{\boldsymbol{A}}\boldsymbol{A}^{-1/2})\|_2^2, \\ \mathrm{err}(\widehat{\boldsymbol{B}}) &= \|\log(\boldsymbol{B}^{-1/2}\widehat{\boldsymbol{B}}\boldsymbol{B}^{-1/2})\|_2^2. \end{aligned} \tag{9}$$

Comparison between gradient descent and recursive estimation ($d = 3$)

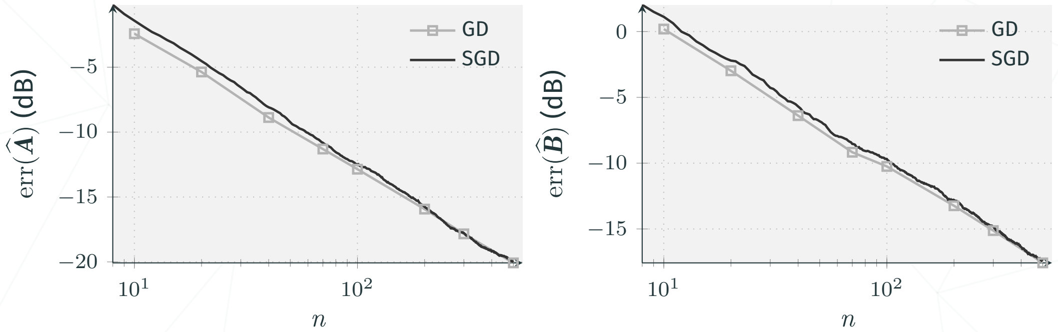


Figure 1: Mean of error measures on A (left) and B (right) of the classical gradient descent method (GD) and its on-line counterpart (SGD) as functions of the number of samples n. $d = 3$.

Comparison between gradient descent and recursive estimation ($d = 100$)

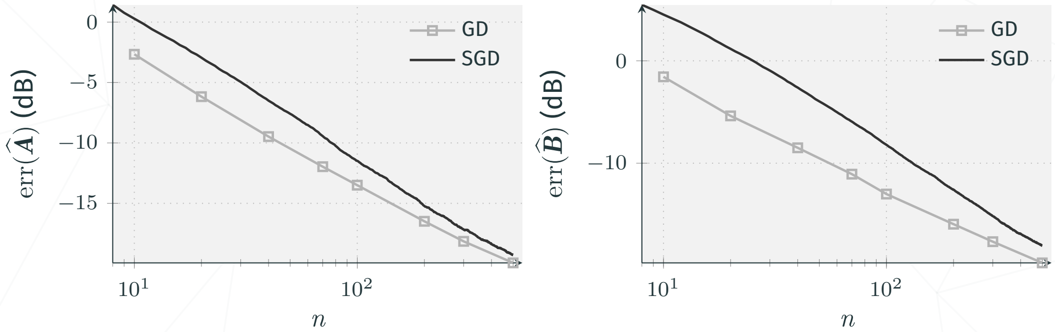


Figure 2: Mean of error measures on A (left) and B (right) of the classical gradient descent method (GD) and its on-line counterpart (SGD) as functions of the number of samples n. $d = 100$.

Outline

Conclusion

We have achieved:

- Information geometry on Kronecker products based on Fisher information metric of a Student-t distribution.
- Efficient online scheme for estimation.

Next:

- Extension to all elliptical distributions and deterministic compound-Gaussian distribution.
- Applications in STAP and SAR problems.

Outline

References i

- Shun-ichi Amari, *Natural gradient learning for over-and under-complete bases in ica*, Neural computation **11** (1999), no. 8, 1875–1883.
- F. Bouchard, A. Breloy, G. Ginolhac, A. Renaux, and F. Pascal, *A Riemannian framework for low-rank structured elliptical models*, IEEE Transactions on Signal Processing **69** (2021), 1185–1199.
- A. Breloy, G. Ginolhac, A. Renaux, and F. Bouchard, *Intrinsic Cramèr – Rao bounds for scatter and shape matrices estimation in CES distributions*, IEEE Signal Processing Letters **26** (2019), no. 2, 262–266.

References ii

- N. Boumal, B. Mishra, P.-A. Absil, and R. Sepulchre, *Manopt, a Matlab toolbox for optimization on manifolds*, Journal of Machine Learning Research **15** (2014), 1455–1459.
- J.C. de Munck, H.M. Huizenga, L.J. Waldorp, and R.A. Heethaar, *Estimating stationary dipoles from meg/eeg data contaminated with spatially and temporally correlated background noise*, IEEE Transactions on Signal Processing **50** (2002), no. 7, 1565–1572.
- Kristjan Greenewald, Edmund Zelnio, and Alfred Hero, *Robust sar stap via kronecker decomposition*, 2016.
- A. Mian, J. Ovarlez, A. M. Atto, and G. Ginolhac, IEEE Transactions on Geoscience and Remote Sensing **57** (2019), no. 6, 3919–3932.

References iii

- Bruno Meriaux, Chengfang Ren, Arnaud Breloy, Mohammed Nabil El Korso, and Philippe Forster, *Efficient estimation of kronecker product of linear structured scatter matrices under t-distribution*, 2020 28th European Signal Processing Conference (EUSIPCO), IEEE, 2021, pp. 2418–2422.
- Ying Sun, Prabhu Babu, and Daniel P Palomar, *Robust estimation of structured covariance matrix for heavy-tailed elliptical distributions*, IEEE Transactions on Signal Processing **64** (2016), no. 14, 3576–3590.
- Kai Yu, M. Bengtsson, B. Ottersten, D. McNamara, P. Karlsson, and M. Beach, *Modeling of wide-band mimo radio channels based on nlos indoor measurements*, IEEE Transactions on Vehicular Technology **53** (2004), no. 3, 655–665.

References iv

Jialun Zhou and Salem Said, *Fast, asymptotically efficient, recursive estimation in a Riemannian manifold*, Entropy **21** (2019), no. 10, 1021.