Online estimation of MGGD : the Riemannian Averaged Natural Gradient method

Jialun Zhou, Salem Said, Yannick Berthoumieu

Online estimation of MGGD : the Riemannian Averaged Natural Gradient method

Jialun ZHOU
University of Bordeaux
Laboratoire IMS
Talence, France
Email: jialun.zhou@u-bordeaux.fr

Salem SAID
University of Bordeaux
Laboratoire IMS
Talence, France
Email: salem.said@u-bordeaux.fr

Yannick BERTHOUMIEU
University of Bordeaux
Laboratoire IMS
Talence, France
Email: yannick.berthoumieu@u-bordeaux.fr

***Abstract*—Multivariate Generalized Gaussian Distributions (MGGD) are a rich class of multivariate distributions, which have gained importance across many engineering applications (image processing, computer vision, radar and biomedical signal processsing). Unfortunately, estimating the parameters of MGGD leads to non-linear matrix equations, whose solution becomes unpractical in high-dimensional problems, or when dealing with very large datasets. To overcome this difficulty, the present paper proposes a new method for online estimation of MGGD parameters, called the Riemannian Averaged Natural Gradient (RANG) method. The RANG method is suitable for application with high-dimensional and large datasets, since it requires modest memory and computational resources. The present paper formulates this new method, and presents some computer simulations, to showcase its performance. It is seen that, while the RANG method makes less exhaustive use of available data, it still achieves identical performance, to classical maximum-likelihood estimation, for sufficiently large datasets.**

I. Introduction

Multivariate Generalized Gaussian distributions (MGGD) were originally introduced in [1], and further studied in [2], [3]. They have attracted much attention due to their numerous applications. They are used in image processing, to model the distribution of wavelet and curvelet coefficients [24], and for image denoising [25], context-based image retrieval [26], image thresholding [27] and texture classification [28]. They also find applications in radar signal processing [29], video coding and denoising [30], and biomedical signal processing [32].

Across these applications, the estimation of the parameters of MGGD is a recurrent and often important problem. In particular, MLE (Maximum Likelihood Estimation) was studied in [13]–[16], with several FP (Fixed-Point) algorithms proposed in [17]–[20]. While these algorithms are applicable to moderate-sized datasets, they do no scale well for high-dimensional or large datasets. To deal with this issue, the present paper introduces a new method for online estimation of the parameters of MGGD. This new method, called the Riemannian Averaged Natural Gradient (RANG) method, requires modest memory and computational resources, yet still achieves the same performance as maximum likelihood estimation, when dealing with sufficiently large datasets.

The RANG method is motivated by recent progress in the field of Riemannian stochastic optimization. This started in 2013, with the Riemannian stochastic gradient method [6], which sparked several developments [7], [8], [16]. Here, these more theoretical developments are used in an applied context, in order to guarantee the following properties.

The RANG method, being an online method, requires modest memory and computational resources, and is therefore applicable with high-dimensional and large datasets. This method still achieves the same performance as maximum likelihood estimation, for sufficiently large datasets. In practice, the RANG method does not require the user to spend additional time tuning its parameters, and the time that it takes to converge is only very weakly affected by the choice of initialization.

The following Section II states the problem of online estimation of MGGD parameters, and provides some background on the Riemannian geometry of MGGD. Section III describes the RANG method and explains its motivation, by first considering a simpler method of online estimation, here called the Riemannian natural gradient (RNG) method. Finally, Section IV presents computer simulations which showcase the performance and desirable properties of the RANG method.

II. Problem statement and background

A. The problem of online estimation

The present paper introduces an original method for online estimation of Multivariate Generalized Gaussian Distributions (MGGD). Here, an MGGD is given by its probability density function [4]

$$f(x|\mu,\Sigma,\beta)=\frac{C_p(\beta)}{|\Sigma|^{\frac{1}{2}}}\exp\left[-\frac{1}{2}\left[(x-\mu)^\top\Sigma^{-1}(x-\mu)\right]^\beta\right] \quad (1)$$

for $x\in\mathbb{R}^p$, where $\mu\in\mathbb{R}^p$ is the expectation, $\Sigma\in\mathcal{P}_p$ is the scatter matrix, and $\beta\in(0,+\infty)$ is the shape parameter ($\mathcal{P}_p$ denotes the space of $p\times p$ symmetric positive definite matrices). The normalizing constant $C_p(\beta)$ is given by

$$C_p(\beta)=\frac{\Gamma\left(\frac{p}{2}\right)\beta}{\pi^{\frac{p}{2}}\Gamma\left(\frac{p}{2\beta}\right)2^{\frac{p}{2\beta}}} \quad (2)$$

where $\Gamma(\cdot)$ is the gamma function. For a dataset $\mathcal{X}_N=(x_1,\cdots,x_N)$ generated from an MGGD density (1), the expectation μ is easily estimated, by taking the arithmetic mean of $\mathcal{X}_N$. Therefore, in the following, μ is assumed to be known and equal to zero, and the focus is on estimating the couple $\theta=(\Sigma,\beta)$, whose true value will be denoted $\theta^*=(\Sigma^*,\beta^*)$.

Maximum-likelihood estimation of the true value θ^* of θ amounts to finding the global maximum of the function

$$L(\theta) = \sum_{n=1}^{N} \ell(x_n|\theta) \tag{3}$$

where $\ell(x|\theta)$ is the log-likelihood function

$$\ell(x|\theta) = \log C_p(\beta) - \frac{1}{2}\log\det(\Sigma) - \frac{1}{2}\left[x^\top \Sigma^{-1} x\right]^\beta \tag{4}$$

for $\theta = (\Sigma, \beta)$. Often, this is reduced to a fixed point problem, and solved iteratively [21].

In general, there are two difficulties, which stand in the way of applying numerical optimization methods (gradient ascent, Newton method, *etc*), to the problem of maximizing (3). First, since it is necessary to respect the constraint that Σ is a symmetric positive definite matrix, this is a highly non-linear constrained maximization problem. Second, if N is quite large, then numerical maximization methods will require excessively large memory and computational resources.

The present paper proposes to deal with the first difficulty by employing the Riemannian geometry of the space $\mathcal{P}_p$ (of $p \times p$ symmetric positive definite matrices). The second difficulty will be resolved by resorting to online estimation. Roughly, online estimation involves an update formula $\mathcal{F}$ that requires only one sample x_{n+1} for each iteration and eventually converges to the true value θ^*, with a speed of convergence which is suitable for practical applications.

$$\begin{aligned} &\theta_{n+1} = \mathcal{F}(x_{n+1}, \theta_n, \gamma_{n+1}) \\ &\theta_n \to \theta^* \quad \text{when } n \to \infty \end{aligned} \tag{5}$$

where γ_{n+1} is the update step size and its choice conditions the speed of convergence. To define this formula $\mathcal{F}$ we need to recall some useful background about the Riemannian geometry of $\mathcal{P}_p$ in the following Paragraph II-B.

B. The Riemannian geometry of $\mathcal{P}_p$

Using the Riemannian geometry of $\mathcal{P}_p$ will guarantee that all computations are *intrinsic*, in the sense that they automatically produce matrices which are symmetric positive definite, and there is no need to perform any additional checks in order to verify this constraint is respected. When the dimension p is large, such checks can be especially cumbersome.

The space $\mathcal{P}_p$ is an open subset of the vector space $\mathcal{S}_p$ of $p \times p$ real symmetric matrices. Therefore [10], $\mathcal{P}_p$ is a differentiable manifold of dimension $p(p+1)/2$. Moreover, for each $\Sigma \in \mathcal{P}_p$, the tangent space to $\mathcal{P}_p$ at Σ, denoted $T_\Sigma \mathcal{P}_p$, is naturally identified with $\mathcal{S}_p$.

The starting point is to equip $\mathcal{P}_p$ with the information metric of the MGGD model [11] [12]. For each $\Sigma \in \mathcal{P}_p$, this defines a scalar product on the tangent space $T_\Sigma \mathcal{P}_p$, given by [12]

$$\langle u, v\rangle_\Sigma = I_1 \mathrm{tr}\left(\Sigma^{-1} u \Sigma^{-1} v\right) + I_2 \mathrm{tr}(\Sigma^{-1}u)\mathrm{tr}(\Sigma^{-1}v) \tag{6}$$

for $u, v \in T_\Sigma \mathcal{P}_p \simeq \mathcal{S}_p$, where $I_1 = \frac{1}{p}\left(\mathcal{A} - \frac{p^2}{4}\right)$ and $I_2 = \frac{\mathcal{A}}{p(p-1)/2-1}\left(1 - \frac{1}{p}\right)$ with $\mathcal{A} = \frac{p}{2}\left(\frac{p}{2} + \beta\right)$.

In turn, this metric defines a Riemannian exponential map [10]. For each $\Sigma \in \mathcal{P}_p$ and $u \in T_\Sigma \mathcal{P}_p$, this computes a new point $\mathrm{Exp}_\Sigma(u)$ which automatically belongs to $\mathcal{P}_p$. Roughly, this new point results by moving away from Σ in the direction of u, along a geodesic curve of the metric (6). Here, the following formula holds [22] [23]

$$\mathrm{Exp}_\Sigma(u) = \Sigma \exp(\Sigma^{-1} u) \tag{7}$$

where exp is the matrix exponential. Now, equation (7) is the main ingredient in the formula $\mathcal{F}$ discussed in (5).

As for any Riemannian metric [10], the Fisher information metric also defines a Riemannian distance on $\mathcal{P}_p$. For any Σ_0 and Σ_1 in $\mathcal{P}_p$, this is called Rao's information distance between Σ_0 and Σ_1, defined as

$$\begin{aligned} d^2(\Sigma_0, \Sigma_1) &= I_1 \mathrm{tr}\left\{\left[\log(\Sigma_0^{-1}\Sigma_1)\right]^2\right\} \\ &\quad + I_2\left\{\mathrm{tr}\left[\log(\Sigma_0^{-1}\Sigma_1)\right]\right\}^2 \end{aligned} \tag{8}$$

where log is the symmetric matrix logarithm function.

III. Online estimation of MGGD

Based on the above concepts, the present section proposes an online algorithm to estimate the parameters of MGGD, which requires modest memory and computational resources, and is therefore applicable with large or high-dimensional datasets. This feature is not available in previously proposed methods [13]–[20]. For preparation, the simpler case where β is known will be considered.

A. RNG estimation for known β

From definition (6) of the information metric, it is possible to compute the natural gradient (or information gradient) [34], of $\ell(x|\theta)$ in (4), with respect to Σ. This turns out to be

$$\begin{aligned} \nabla_\Sigma^{inf} \ell(x|\Sigma, \beta) &= \frac{1}{I_1}\left[\frac{h'(y)}{p} y - \frac{1}{2}\right]\Sigma \\ &\quad + \frac{1}{I_2}\left[h'(y)\left(S_x - \frac{y}{p}\Sigma\right)\right] \end{aligned} \tag{9}$$

where $S_x = xx^\top$, and $h(y) = \frac{1}{2}y^\beta$ with $y = x^\top \Sigma^{-1} x$. For lack of space, the derivation of (9) cannot be given here.

Combining the natural gradient (9) with equation (7) provides the update formula $\mathcal{F}$, required in (5). Precisely, this is defined as follows

$$\Sigma_{n+1} = \mathrm{Exp}_{\Sigma_n}\left(\gamma_{n+1} \nabla_{\Sigma_n}^{inf} \ell(x_{n+1}|\Sigma_n, \beta)\right) \tag{10}$$

with step sizes $\gamma_n = \frac{a}{n+1}$ where $a \in (0, \infty)$. This stochastic Riemannian Natural Gradient (RNG) method is stated as follows:

Algorithm 1 RNG for estimation of Σ with known β

Input: a dataset $X = \{x_1, \cdots, x_N\}$, an initial guess Σ_0;
Output: the estimator $\hat{\Sigma}$;

1: **for** $n = 1, \cdots, N-1$ **do**
2: Compute the natural gradient $\nabla_{\Sigma_n}^{inf} \ell(x_{n+1}|\Sigma_n, \beta)$ by equation (9);
3: Compute Σ_{n+1} by equation (10);
4: **end for**
5: $\hat{\Sigma} \leftarrow \Sigma_N$;

For any $a > 1/2$, this algorithm can achieve the optimal non-asymptotic rate of convergence

$$\mathbb{E}\left[d^2(\Sigma_n, \Sigma^*)|\Sigma^*\right] \le \frac{K}{n} \tag{11}$$

in terms of the information distance (8), where K is some positive constant. Moreover, it is asymptotically efficient, which means it achieves the same performance as maximum likelihood estimation, when N becomes large [8]. These fast rate and asymptotic efficiency hold for all values of β (mathematically, this means they hold even when the objective function (3) is non-concave with respect to the metric (6)).

In theory, the RNG method can be extended to the case where β is unknown, using the information metric derived for this case in [13]. However, this information metric is very complicated, and its geodesics are not known explicitly. The computations required for the RNG method then become excessively complex and numerically unstable.

To overcome this problem, a novel solution is considered in the following section III-B, under the name of RANG (Riemannian Averaged Natural Gradient) method.

B. RANG estimation for unknown β

As mentioned above, calculating the information metric of the complete model (that is, for both Σ and β) is a bad choice. The most straightforward idea is to find an online method which does not require this information metric. Guided by this idea, the present paragraph proposes an alternating online method.

At first, assuming β is known, this method updates Σ, using the natural gradient (9). Then, assuming Σ is known, it updates β, using the natural gradient with respect to β.

Here, the natural gradient with respect to β is given by [13]

$$\nabla_\beta^{inf}\ell(x|\Sigma,\beta) = \frac{1}{I_c}\nabla_\beta\ell(x|\Sigma,\beta) \tag{12}$$

where

$$\begin{aligned}\nabla_\beta\ell(x|\Sigma,\beta) &= \tfrac{\partial}{\partial\beta}\ell(x|\Sigma,\beta)\\ &= \tfrac{1}{\beta}+\tfrac{p}{2\beta^2}\left[\Psi_0(\tfrac{p}{2\beta})+\log 2\right]\\ &\quad -\tfrac{1}{2}y^\beta\log(y)\end{aligned} \tag{13}$$

$$\begin{aligned}I_c &= \mathbb{E}\left[\tfrac{\partial}{\partial\beta}\ell(x|\Sigma,\beta)\tfrac{\partial}{\partial\beta}\ell(x|\Sigma,\beta)\right]\\ &= \tfrac{1}{\beta^2}\Big\{1+\left(\tfrac{p}{2\beta}\right)^2\Psi_1\left(\tfrac{p}{2\beta}\right)+\tfrac{p}{\beta}\left[\ln(2)+\Psi_0\left(\tfrac{p}{2\beta}\right)\right]\\ &\quad +\tfrac{p}{2\beta}\left[(\ln(2))^2+\Psi_0(1+\tfrac{p}{2\beta})\left[\ln(4)+\Psi_0(1+\tfrac{p}{2\beta})\right]\right.\\ &\quad \left.+\Psi_1\left(1+\tfrac{p}{2\beta}\right)\right]\Big\}\end{aligned} \tag{14}$$

where y was defined in (9), and Ψ_0, Ψ_1 are the digamma and trigamma functions [33]. Now, since β is a real number, the update rule similar to (10) reduces to

$$\beta_{n+1} = \beta_n + \alpha_\beta\nabla_{\beta_n}^{inf}\ell(x_{n+1}|\Sigma_n,\beta_n) \tag{15}$$

where α_β is the step size, which will be maintained constant. Indeed, for the case of unknown β, the rate (11) no more holds for a decreasing step size. In this way, to have a sufficiently fast algorithm, constant step size is a better choice. For the same reason, the step size for Σ is also replaced by a constant, denoted α_Σ. Then,

$$\Sigma_{n+1} = \mathrm{Exp}_{\Sigma_n}\left(\alpha_\Sigma\nabla_{\Sigma_n}^{inf}\ell(x_{n+1}|\Sigma_n,\beta_n)\right) \tag{16}$$

Maintaining constant step sizes guarantees an agressive global search strategy. However, in this case, it becomes necessary to stabilize the new update rules (15) and (16). This is done using an online Riemannian averaging step, similar to [7].

The output of this online Riemannian averaging step is a new sequence $(\tilde\Sigma_n, \tilde\beta_n)$, obtained from the outputs (Σ_n, β_n) of (15) and (16), as follows

$$\tilde\beta_{n+1} = \frac{n}{n+1}\tilde\beta_n + \frac{1}{n+1}\beta_{n+1} \tag{17}$$

$$\tilde\Sigma_{n+1} = \mathrm{Exp}_{\tilde\Sigma_n}\left[\frac{1}{n+1}\mathrm{Exp}_{\tilde\Sigma}^{-1}(\Sigma_{n+1})\right] \tag{18}$$

In other words, $\tilde\beta_{n+1}$ is the weighted average of $\tilde\beta_n$ and β_{n+1}. On the other hand, $\tilde\Sigma_{n+1}$ is the geodesic weighted average of $\tilde\Sigma_n$ and Σ_{n+1} [7].

Accordingly, the following algorithm is proposed.

Algorithm 2 RANG for estimation of Σ and β

Input: a dataset $X = \{x_1, \cdots, x_N\}$, an initial guess (Σ_0, β_0) and $(\tilde\Sigma_0, \tilde\beta) = (\Sigma_0, \beta_0)$;
Output: the estimator $\hat\Sigma$, $\hat\beta$;
1: **for** $n = 1, \cdots, N-1$ **do**
2: Calculate the gradients by (9) and (12);
3: Update Σ_{n+1} and β_{n+1} by (15) and (16);
4: Take the average for $\tilde\Sigma_{n+1}$ and $\tilde\beta_{n+1}$ by (17) and (18);
5: **end for**
6: $\hat\Sigma \leftarrow \tilde\Sigma_N$ and $\hat\beta \leftarrow \tilde\beta_N$;

This algorithm displays a fast rate of convergence, to the true value $\theta^* = (\Sigma^*, \beta^*)$, with little to no effect from the choice of initial guess. Experimental results of this method are presented in the following section.

IV. Simulation results

A. Simulations with known β

Consider first the case where β is known, corresponding to III-A. Figure 1 showcases the asymptotic efficiency of the RNG method. This figure was generated from a dataset $\mathcal{X}_N$ where $N = 10^6$. The matrices in this dataset are of size $p \times p$ where $p = 7$. The step sizes γ_n in (10) were taken to be $\gamma_n = \frac{1}{n}$, so that $a = 1$. Asymptotic efficiency is clearly shown by the fact that the estimation error from the RNG method coincides with the estimation error of MLE, where the MLE is implemented by RDG [9] (Riemannian Deterministic Gradient) and FP [20] (Fixed point) method, around the point of 50000 data samples.

Figure 2 shows that the RNG method reaches the same rate of convergence (11), whether the objective function (3) is concave ($\beta = 4$) or non-concave ($\beta = 1/3$). In either case, the slope of the error curve, in log-log plot, is ultimately constant and equal to -1, corresponding to the upper bound $\frac{K}{n}$ in (11).

Figure 3 compares the time consumption of the RNG method, with that of MLE estimations implemented by RDG and FP. Under the condition of ensuring the same performance, as the sample size increases, the time consumption of MLE increases significantly, compared with the RNG method. Thus, the RNG method has equivalent performance to MLE, but with significantly less time consumption.

B. Simulation with unknown β

When the parameter β is unknown, the RANG method of III-B is used. Here, the dataset $\mathcal{X}_N$ has $N = 2 \times 10^6$ samples, with matrices of size $p \times p$ where $p = 7$. The chosen step size is $\alpha_\beta = \alpha_\Sigma = 0.001$, which performs robustly in this case. Figure 4, showcases the speed of convergence obtained using this algorithm.

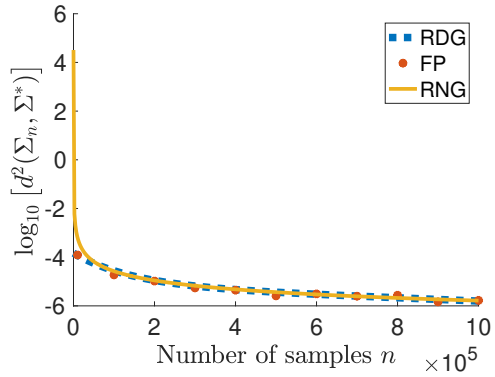


Fig. 1. Comparison of the efficiency between RDG, FP and RNG.

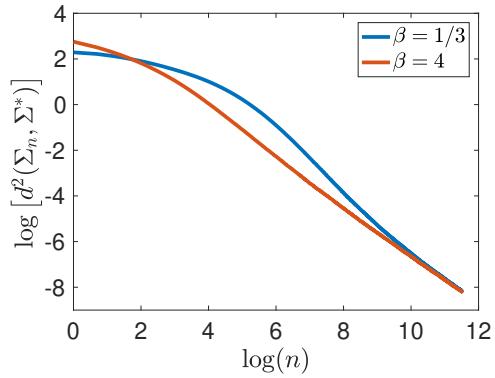


Fig. 2. Estimation error of the RNG method for different β values.

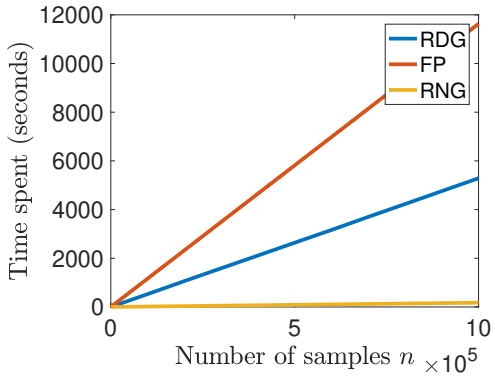
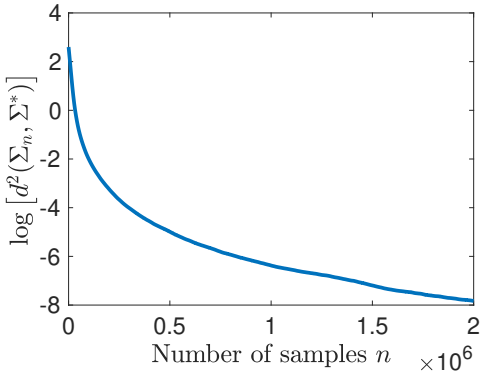


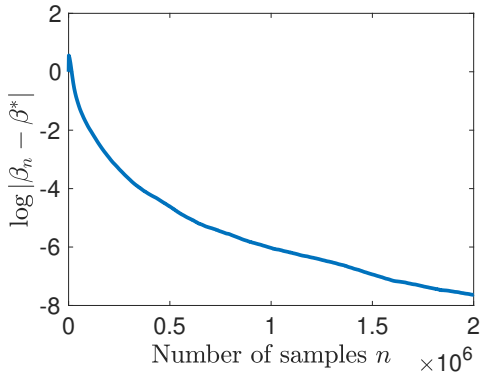
Fig. 3. Comparison of the time consumption between RDG, FP and RNG.

V. Conclusion

This paper has addressed the problem of online estimation of the parameters of Multivariate Generalized Gaussian Distributions. Although this problem has strong motivation, from applications to large or high-dimensional datasets, it has not received sufficient attention in the literature. To deal with this situation, two algorithms have been proposed, the RNG algorithm for estimation in the case of known shape parameter, and the RANG algorithm, for estimation in the case of unknown shape parameter. These algorithms rely on the information geometry of multivariate generalized Gaussian distributions, and the idea of Riemannian averaging of the outputs of a Riemannian natural gradient method with constant step size. It has been shown, through computer experiments, that these algorithms achieve the same asymptotic performance as maximum likelihood estimation, while requiring significantly less computation time. This reflects their reduced complexity, which makes them suitable for applications where there are significant restrictions on computational and memory resources.



(a) Error variation of Σ parameter



(b) Error variation of β parameter

Fig. 4. RANG algorithm for an unknown β

References

[1] Kelker D. *Distribution theory of spherical distributions and a location-scale parameter generalization[J].* Sankhyā: The Indian Journal of Statistics, Series A, 1970: 419-430.

[2] Rangaswamy M, Weiner D D, Ozturk A. *Non-Gaussian random vector identification using spherically invariant random processes[J].* IEEE Transactions on Aerospace and Electronic Systems, 1993, 29(1): 111-124.

[3] Rangaswamy M, Weiner D, Ozturk A. *Computer generation of correlated non-Gaussian radar clutter[J].* IEEE Transactions on Aerospace and Electronic Systems, 1995, 31(1): 106-116.

[4] S. Kotz, *Statistical Distributions in Scientific Work, I.* Dordrecht: Reidel, 1968, ch. Multivariate distributions at a cross road, pp. 247–270.

[5] Cover T M, Thomas J A. *Elements of information theory[M].* John Wiley and Sons, 2012.

[6] Silvere Bonnabel. *Stochastic gradient descent on riemannian manifolds.* IEEE Transactions on Automatic Control, 2013.

[7] Tripuraneni N, Flammarion N, Bach F, et al. *Averaging stochastic gradient descent on Riemannian manifolds[J].* arXiv preprint arXiv:1802.09128, 2018.

[8] Zhou J, Said S. *Recursive parameter estimation in a Riemannian manifold[J].* arXiv preprint arXiv:1805.06811, 2018.

[9] Absil P A, Mahony R, Sepulchre R. *Optimization algorithms on matrix manifolds[M].* Princeton University Press, 2009.

[10] Lee J M. *Introduction to smooth manifolds[M].* Springer, 2001.

[11] Amari S, Nagaoka H. *Methods of information geometry[M].* American Mathematical Soc., 2007.

[12] Berkane M, Oden K, Bentler P M. *Geodesic estimation in elliptical distributions[J].* Journal of Multivariate Analysis, 1997, 63(1): 35-46.

[13] Verdoolaege G, Scheunders P. *On the geometry of multivariate generalized Gaussian models[J].* Journal of mathematical imaging and vision, 2012, 43(3): 180-193.

[14] Verdoolaege G, Scheunders P. *Geodesics on the manifold of multivariate generalized Gaussian distributions with an application to multicomponent texture discrimination[J].* International Journal of Computer Vision, 2011, 95(3): 265.

[15] Zhang T, Wiesel A, Greco M S. *Multivariate generalized gaussian distribution: Convexity and graphical models[J].* IEEE Transactions on Signal Processing, 2013, 61(16): 4141-4148.

[16] Zhang H, Sra S. *First-order methods for geodesically convex optimization[C]* Conference on Learning Theory. 2016: 1617-1638.

[17] Pascal F, Chitour Y, Ovarlez J P, et al. *Covariance structure maximum-likelihood estimates in compound Gaussian noise: Existence and algorithm analysis[J].* IEEE Transactions on Signal Processing, 2007, 56(1): 34-48.

[18] Chitour Y, Pascal F. *Exact maximum likelihood estimates for SIRV covariance matrix: Existence and algorithm analysis[J].* IEEE Transactions on signal processing, 2008, 56(10): 4563-4573.

[19] Maronna R A. *Robust M-estimators of multivariate location and scatter[J].* The annals of statistics, 1976: 51-67.

[20] Pascal F, Bombrun L, Tourneret J Y, et al. Parameter estimation for multivariate generalized Gaussian distributions[J]. IEEE Transactions on Signal Processing, 2013, 61(23): 5960-5971.

[21] Boukouvalas Z, Said S, Bombrun L, et al. *A new riemannian averaged fixed-point algorithm for MGGD parameter estimation[J].* IEEE Signal Processing Letters, 2015, 22(12): 2314-2318.

[22] Sra S, Hosseini R. *Conic geometric optimization on the manifold of positive definite matrices[J].* SIAM Journal on Optimization, 2015, 25(1): 713-739.

[23] Pennec X, Fillard P, Ayache N. *A Riemannian framework for tensor computing[J].* International Journal of computer vision, 2006, 66(1): 41-66.

[24] Boubchir L, Fadili J. *Multivariate statistical modeling of images with the curvelet transform[C].*ISSPA. 2005: 747-750.

[25] Cho D, Bui T D, Chen G. *Image denoising based on wavelet shrinkage using neighbor and level dependency[J].* International journal of wavelets, multiresolution and information processing, 2009, 7(03): 299-311.

[26] Verdoolaege G, De Backer S, Scheunders P. *Multiscale colour texture retrieval using the geodesic distance between multivariate generalized Gaussian models[C]* 2008 15th IEEE International Conference on Image Processing. IEEE, 2008: 169-172.

[27] Bazi Y, Bruzzone L, Melgani F. *Image thresholding based on the EM algorithm and the generalized Gaussian distribution[J].* Pattern Recognition, 2007, 40(2): 619-634.

[28] Scharcanski J. *A wavelet-based approach for analyzing industrial stochastic textures with applications[J].* IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 2006, 37(1): 10-22.

[29] Desai M N, Mangoubi R S. *Robust Gaussian and non-Gaussian matched subspace detection[J].* IEEE Transactions on Signal Processing, 2003, 51(12): 3115-3127.

[30] Coban M Z, Mersereau R M. *Adaptive subband video coding using bivariate generalized Gaussian distribution model[C].* IEEE International Conference on Acoustics, Speech, and Signal Processing Conference Proceedings. 1996, 4: 1990-1993.

[31] Yang J, Wang Y, Xu W, et al. *Image and video denoising using adaptive dual-tree discrete wavelet packets[J].* IEEE transactions on circuits and systems for video technology, 2009, 19(5): 642-655.

[32] Elguebaly T, Bouguila N. *Bayesian learning of generalized Gaussian mixture models on biomedical images[C].* IAPR Workshop on Artificial Neural Networks in Pattern Recognition. Springer, Berlin, Heidelberg, 2010: 207-218.

[33] Abramowitz, M. and Stegun, I. A. (Eds.). *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* 9th printing. New York: Dover, pp. 258-259, 1972.

[34] Amari S I. *Natural gradient works efficiently in learning[J].* Neural computation, 1998, 10(2): 251-276.