Riemannian Multinomial Logistics Regression for SPD Neural Networks

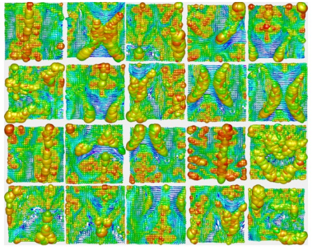
– CVPR 2024 –

Ziheng Chen[1], Yue Song[1], Gaowen Liu[2], Ramana Rao Kompella[2], Xiaojun Wu[3], Nicu Sebe[1]

1 University of Trento, Italy
2 Cisco Systems, USA
3 Jiangnan University, China

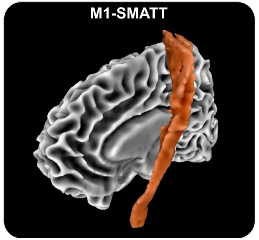
Applications of SPD Manifolds

Image generation



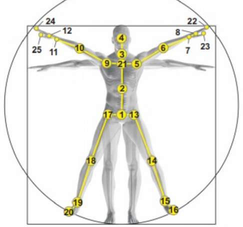
Huang et al., 2019

Medical Imaging



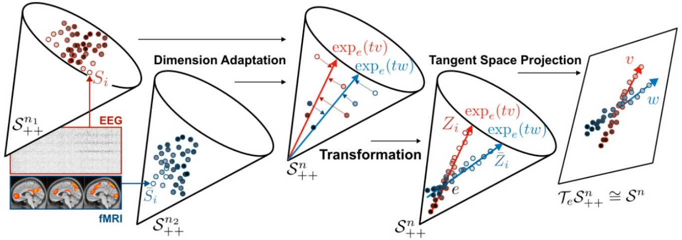
Chakraborty et al., 2020

Action Recognition



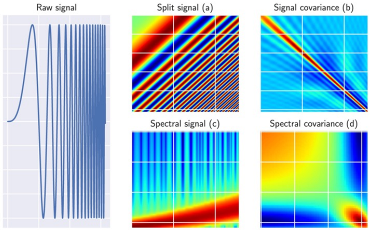
Nguyen, Xuan Son 2021

Brain-Computer Interfaces



Ju et al., 2024

Radar Classification

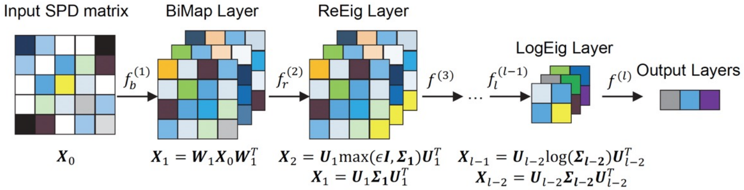


Brooks et al., 2020

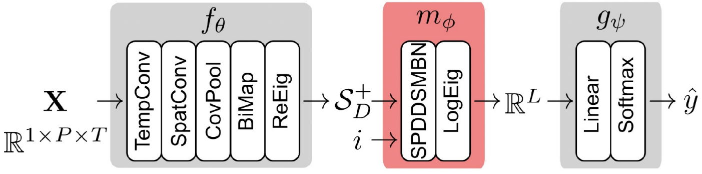
Huang, Zhiwu, Jiqing Wu, and Luc Van Gool. "Manifold-valued image generation with wasserstein generative adversarial nets." AAAI, 2019.
Chakraborty, Rudrasis, et al. "Manifoldnet: A deep neural network for manifold-valued data with applications." IEEE-TPAMI, 2020.
Ju, Ce, et al. "Deep geodesic canonical correlation analysis for covariance-based neuroimaging data." ICLR, 2024.
Nguyen, Xuan Son. "Geomnet: A neural network based on riemannian geometries of SPD matrix space and cholesky space for 3D skeleton-based interaction recognition." ICCV, 2021.
Brooks, Daniel, et al. "Deep learning and information geometry for drone micro-Doppler radar classification." RadarConf, 2020.

Classification on SPD Neural Networks

Tangent Space:



SPDNet (Huang et al., 2017)



TSMNet (Kobler et al., 2022)

Parameterization:

$$Y_t = \mathrm{FM}\left(\left\{M_{t-1}^{(\alpha)}\right\}, \left\{w^{(y,\alpha)}\right\}\right), \quad R_t = \mathrm{T}\left(Y_t, g^{(r)}\right)$$

$$T_t = \mathrm{FM}\left(\left\{R_t, X_t\right\}, w^{(t)}\right), \quad \Phi_t = \mathrm{T}\left(T_t, g^{(p)}\right)$$

$$\forall \alpha \in J, \quad M_t^{(\alpha)} = \mathrm{FM}\left(\left\{M_{t-1}^{(\alpha)}, \Phi_t\right\}, \alpha\right)$$

$$S_t = \mathrm{FM}\left(\left\{M_t^{(\alpha)}\right\}, \left\{w^{(s,\alpha)}\right\}\right), \quad O_t = \mathrm{Chol}\left(\mathrm{ReLU}\left(\mathrm{Chol}\left(\mathrm{T}\left(S_t, g^{(y)}\right)\right)\right)\right)$$

SPD-SRU (Chakraborty et al., 2018)

They rely on approximated spaces

How to build intrinsic classification layers on manifolds?

Huang, Zhiwu, and Luc Van Gool. "A riemannian network for spd matrix learning." AAAI, 2017.
Kobler, Reinmar, et al. "SPD domain-specific batch normalization to crack interpretable unsupervised domain adaptation in EEG." Neurips, 2022.
Chakraborty, Rudrasis, et al. "A statistical recurrent model on the manifold of symmetric positive definite matrices." Neurips, 2018.

Contributions

SPDMLR:

- A general framework for SPD Multinomial Logistics Regression (MLR) under PEMs
- Specific SPD MLRs under parameterized LCM and LEM
- An intrinsic theoretical explanation of the most popular LogEig classifier

MLR Revisiting

Euclidean MLR:

$$\forall k \in \{1, \ldots, C\}, p(y = k \mid x) \propto \exp\left(\left(\langle a_k, x \rangle - b_k\right)\right)$$

Reformulation into margin distance to hyperplane

$$p(y = k \mid x) \propto \exp(\operatorname{sign}(\langle a_k, x - p_k \rangle) \|a_k\| d(x, H_{a_k, p_k}))$$

$$H_{a_k, p_k} = \{x \in \mathbb{R}^n : \langle a_k, x - p_k \rangle = 0\}$$

Gyro SPD MLR:

- Requires gyro vector structures
- Relies on gyro distance, instead of geodesic distance
- Solves formulation case by case

Our SPD MLR:

- Focus on Pullback Euclidean Metrics (PEMs)
- Only needs Riemannian geometry
- Relies on geodesic distance
- Proposes a general formulation for PEMs

Nguyen, Xuan Son, and Shuo Yang. "Building neural networks on matrix manifolds: A Gyrovector space approach." ICML, 2023.

SPD MLR under PEMs

From Euclidean to SPD

$$p(y=k \mid x) \propto \exp(\operatorname{sign}(\langle a_k, x-p_k\rangle)\|a_k\| d(x, H_{a_k,p_k})) \qquad H_{a_k,p_k} = \{x \in \mathbb{R}^n : \langle a_k, x-p_k\rangle = 0\}$$

Riemannian reformulation

Definition 3.1 (SPD hyperplanes). Given $P \in \mathcal{S}^n_{++}$, $A \in T_P\mathcal{S}^n_{++} \backslash \{\mathbf{0}\}$, we define the SPD hyperplane as

$$\tilde{H}_{A,P} = \{S \in \mathcal{S}^n_{++} : g_P(\operatorname{Log}_P S, A) = \langle \operatorname{Log}_P S, A\rangle_P = 0\}, \tag{12}$$

where P and A are referred to as shift and normal matrices, respectively.

Definition 3.2 (SPD MLR). SPD MLR is defined as

$$p(y=k \mid S) \propto \exp(\operatorname{sign}(\langle A_k, \operatorname{Log}_{P_k}(S)\rangle_{P_k})\|A_k\|_{P_k} d(S, \tilde{H}_{A_k,P_k})), \tag{13}$$

where $P_k \in \mathcal{S}^n_{++}$, $A_k \in T_{P_k}\mathcal{S}^n_{++} \backslash \{\mathbf{0}\}$, $\langle \cdot, \cdot \rangle_{P_k} = g_{P_k}$, and $\|\cdot\|_{P_k}$ is the norm on $T_{P_k}\mathcal{S}^n_{++}$ induced by g at P_k, and $\tilde{H}_{A_k,P_k}$ is a margin hyperplane in $\mathcal{S}^n_{++}$ as defined in Eq. (12). $d(S, \tilde{H}_{A_k,P_k})$ denotes the margin distance between S and SPD hyperplane $\tilde{H}_{A_k,P_k}$, which is formulated as:

$$d(S, \tilde{H}_{A_k,P_k})) = \inf_{Q \in \tilde{H}_{A_k,P_k}} d(S, Q), \tag{14}$$

where $d(S, Q)$ is the geodesic distance induced by g.

Proposition 3.3 (Submanifolds). *The SPD hyperplane (as defined in Eq. (12)) under any geometrically complete Riemannian metric g is a regular submanifold of SPD manifolds.*

Submanifolds are natural generalizations of the Euclidean hyperplanes.

SPD MLR under PEMs

Margin distance and MLR

Margin distance:

Lemma 3.5. *Given a PEM g, the margin distance defined in Eq. (14) has a closed-form solution:*

$$d(S, \tilde{H}_{A_k,P_k})) = d(\phi(S), H_{\phi_{*,P_k}(A_k),\phi(P_k)}), \tag{15}$$

$$= \frac{|\langle \phi(S) - \phi(P_k), \phi_{*,P_k}(A_k) \rangle|}{\|A_k\|_{P_k}}, \tag{16}$$

where $|\cdot|$ is the absolute value.

Optmization:

Lemma 3.6. *Given a PEM, any parallel transportation is equivalent to the differential map of a left translation and vice versa.*

Lemma 3.7. *Given two fixed SPD matrices $Q_1, Q_2 \in \mathcal{S}^n_{++}$, we have the following equivalence for parallel transportations under a PEM,*

$$\begin{aligned} &\forall \tilde{A}_{1,k} \in T_{Q_1}\mathcal{S}^n_{++}, \exists! \tilde{A}_{2,k} \in T_{Q_2}\mathcal{S}^n_{++}, \\ &s.t. \Gamma_{Q_1 \to P_k}(\tilde{A}_{1,k}) = \Gamma_{Q_2 \to P_k}(\tilde{A}_{2,k}). \end{aligned} \tag{18}$$

General formulation:

Theorem 3.8 (SPD MLR under a PEM). *Under any PEM, SPD MLR and SPD hyperplane is*

$$p(y = k \mid S) \propto \exp(\langle \phi(S) - \phi(P_k), \phi_{*,I}(\tilde{A}_k) \rangle), \tag{19}$$

$$\tilde{H}_{\tilde{A}_k,P_k} = \{S \in \mathcal{S}^n_{++} : \langle \phi(S) - \phi(P_k), \phi_{*,I}(\tilde{A}_k) \rangle = 0\}, \tag{20}$$

where $\tilde{A}_k \in T_I\mathcal{S}^n_{++}/\{0\} \cong \mathcal{S}^n/\{0\}$ is a symmetric matrix, and $P_k \in \mathcal{S}^n_{++}$ is an SPD matrix.

MLR on the parameterized LEM and LCM

Corollary 4.1 (SPD MLRs under the deformed LEM and LCM). *The SPD MLRs under (α, β)-LEM is*

$$p(y = k \mid S) \propto \exp\left[\langle \operatorname{mlog}(S) - \operatorname{mlog}(P_k), \tilde{A}_k \rangle^{(\alpha,\beta)}\right], \tag{21}$$

where $\tilde{A}_k \in T_I \mathcal{S}^n_{++} \cong \mathcal{S}^n$ *and* $P_k \in \mathcal{S}^n_{++}$. *The SPD MLRs under (θ)-LCM is*

$$p(y = k \mid S) \propto \exp\left[\frac{1}{\theta}\langle \lfloor \tilde{K} \rfloor - \lfloor \tilde{L}_k \rfloor + \left[\operatorname{Dlog}(\mathbb{D}(\tilde{K})) - \operatorname{Dlog}(\mathbb{D}(\tilde{L}_k))\right], \lfloor \tilde{A}_k \rfloor + \frac{1}{2}\mathbb{D}(\tilde{A}_k) \rangle\right], \tag{22}$$

where $\tilde{K} = \operatorname{Chol}(S^\theta)$, $\tilde{L}_k = \operatorname{Chol}(P_k^\theta)$, *and* $\mathbb{D}(\tilde{A}_k)$ *denotes a diagonal matrix with diagonal elements of* $\tilde{A}_k$.

Visualization of SPD hyperplane

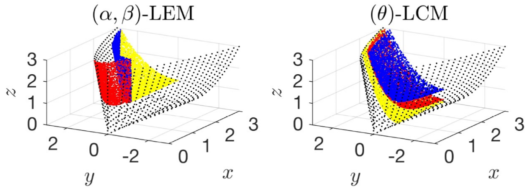
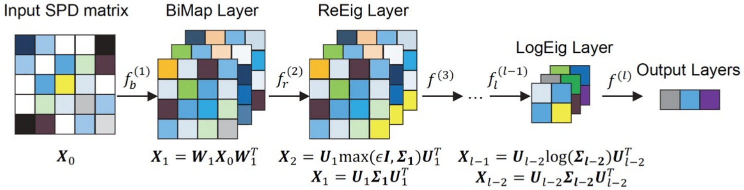


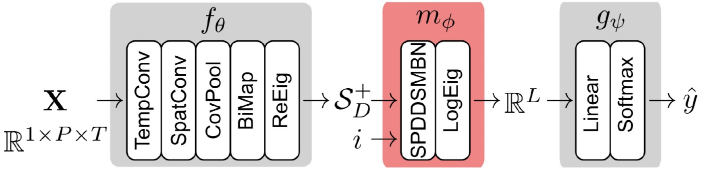
Figure 1. Conceptual illustration of SPD hyperplanes induced by (α, β)-LEM and (θ)-LCM. In each subfigure, the black dots are symmetric positive semi-definite (SPSD) matrices, denoting the boundary of $\mathcal{S}^2_{++}$, while the blue, red, and yellow dots denote three SPD hyperplanes.

An Intrinsic explanantion for LogEig MLR

Tangent Space:



SPDNet (Huang et al., 2017)



TSMNet (Kobler et al., 2022)

> **Proposition 5.1.** *Endowing SPD manifolds with the standard LEM, optimizing SPD parameter P_k in Eq. (21) by LEM-based RSGD and Euclidean parameter A_k by Euclidean SGD, the LEM-based SPD MLR is equivalent to a LogEig MLR with parameters in FC layer optimized by Euclidean SGD.*

LogEig MLR respects a Riemannian MLR

Experiments

Backbone	Classifier	[20,16,8]	[20,16,14,12,10,8]
SPDNet	LogEig MLR	92.88±1.05	93.47±0.45
	Gyro-AIM	94.53±0.95	94.32±0.94
	(1,0)-LEM	93.55±1.21	94.60±0.70
	(1,1)-LEM	**95.64±0.83**	**95.87±0.58**
	(1)-LCM	93.49±1.25	93.93±0.98
	(0.5)-LCM	**94.59±0.82**	**95.16±0.67**

Table 3. Results of SPDNet with different classifiers on the Radar dataset.

Backbone	Classifier	[93,30]	[93,70,30]	[93,70,50,30]
SPDNet	LogEig MLR	57.42±1.31	60.69±0.66	60.76±0.80
	Gyro-AIM	58.07±0.64	60.72±0.62	61.14±0.94
	(1,0)-LEM	57.02±0.75	61.34±0.62	60.78±0.86
	(1)-LCM	62.04±1.05	62.11±2.11	62.89±2.09
	(0.5)-LCM	**65.66±0.73**	**65.79±0.63**	**65.71±0.75**

Table 4. Results of SPDNet with different classifiers on the HDM05 dataset.

Backbone	Classifier	Inter-session	Inter-subject
SPDDSMBN	LogEig MLR	53.83±9.77	49.68±7.88
	Gyro-AIM	53.36±9.92	50.65±8.13
	(1,0)-LEM	53.16±9.73	51.41±7.98
	(1)-LCM	55.71±8.57	51.60±8.43
	(1.5)-LCM	**56.43±8.79**	**51.65±5.90**

Table 5. Results of SPDDSMBN with different classifiers on the Hinss2021 dataset under inter-subject and inter-session scenarios. The presented results are the ones of balanced accuracy under the leaving 5% out cross-validation scenario.

Methods	Radar	HDM05	Hinss2021	
			inter-session	inter-subject
Baseline	1.36	1.95	0.18	8.31
MLR-Gyro-AIM	1.75	31.64	0.38	13.3
MLR-LEM	1.5	4.7	0.24	10.13
MLR-LCM	**1.35**	**3.29**	**0.18**	**8.35**

Table 6. Comparison of training efficiency (s/epoch) of SPDNet (SPDDSMBN) under different classifiers. The most efficient MLR is highlighted in **bold**.

Huang, Zhiwu, and Luc Van Gool. "A riemannian network for spd matrix learning." AAAI, 2017.
Kobler, Reinmar, et al. "SPD domain-specific batch normalization to crack interpretable unsupervised domain adaptation in EEG." Neurips, 2022.

Q & A

Thanks you
Q & A

Code

Paper

Homepage