Data Visualization using Functional Data Analysis

Haozhe Chen*, Andres Duque Correa*, Guy Wolf†‡, Kevin R. Moon*
*Dept. of Mathematics & Statistics, Utah State University, Logan, UT, USA, {a02314155, andres.dc, kevin.moon}@usu.edu
†Mila - Quebec AI Institute, Montreal, Canada, wolfguy@mila.quebec
‡Dept. of Mathematics & Statistics, Université de Montréal, Montreal, Canada

***Abstract*—Data visualization via dimensionality reduction is an important tool in exploratory data analysis. However, when the data are noisy, many existing methods fail to capture the underlying structure of the data. Furthermore, existing methods that can theoretically eliminate all noise are difficult to implement in high dimensions. Here we propose a new data visualization method called Functional Information Geometry (FIG) for dynamical processes that denoises the data by leveraging time information and mitigates the curse of dimensionality using approaches from functional data analysis. We experimentally demonstrate that FIG outperforms other methods in terms of capturing the true structure, hyperparameter robustness, and computational speed. We then use our method to visualize EEG brain measurements of sleep activity.**

***Index Terms*—Functional Data Analysis, dynamical processes, data denoising, dimensionality reduction, data visualization**

I. Introduction and Problem Setting

High-dimensional datasets often contain redundant information, inflating their extrinsic dimension, while their true structure often lies on a low-dimensional manifold with added noise. Manifold learning methods aim to recover this structure while preserving essential information. These techniques have been applied across various fields, including image classification [1]–[3], image synthesis [4], [5], video analysis [6], [7], and single-cell RNA-sequencing [8]. Classical nonlinear dimensionality reduction methods include Isomap [9], MDS [10], LLE [11], and Laplacian Eigenmaps [12], while modern approaches such as t-SNE [13] and UMAP [14] have become widely used. However, these methods struggle in high-noise settings. Diffusion maps (DM) [15] address this by leveraging a Markov diffusion process to emphasize data connectivity and preserve intrinsic geometry, but they are not ideal for visualization due to encoding information in higher dimensions [16], [17]. PHATE [17] builds upon DM to improve visualization by learning manifold structure while denoising.

For dynamical systems and time series data, additional structure can be leveraged for more effective denoising. If the data-generating process satisfies certain assumptions, noise effects translate into linear transformations in probability space [18], [19]. The Mahalanobis distance, invariant to such transformations, can be leveraged to define a noise-resilient distance metric. The Empirical Intrinsic Geometry (EIG) framework achieves this by computing the Mahalanobis distance between local histograms [18], [19], but histogram estimation suffers from the curse of dimensionality. Dynamical Information Geometry (DIG) [20] extends EIG for visualization using a diffusion framework [15], [17], yet retains the limitations of histograms. To overcome this, we introduce Functional Information Geometry (FIG), which constructs Mahalanobis distances in the probability space without direct density estimation [21], [22]. FIG embeds these distances using diffusion and information geometry, improving robustness, efficiency, and visualization quality. We demonstrate its advantages over DIG and EIG on simulated time series and real-world EEG data, showing enhanced structural fidelity, stability, and computational efficiency.

This research was supported in part by Canada CIFAR AI Chair [G.W.], NSERC Discovery grant 03267 [G.W.], NIH grant R01GM135929 [G.W.], and NSF grant 2212325 [K.M.]. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

We use the same state-space formalism given in EIG [18], [19] and DIG [20] for time series data:

$$\mathbf{x}_t = \mathbf{y}_t(\boldsymbol{\theta}_t) + \boldsymbol{\xi}_t \tag{1}$$

$$d\theta_t^i = a^i(\theta_t^i) + dw_t^i, i = 1, ..., d. \tag{2}$$

Let $\mathbf{x}_t$ be a noisy observation of the clean multivariate time series $\mathbf{y}_t$, driven by hidden states $\boldsymbol{\theta}_t$, with noise $\boldsymbol{\xi}_t$ independent of $\mathbf{y}_t$. The conditional density is $p(\boldsymbol{y}|\boldsymbol{\theta})$, where $\boldsymbol{\theta}_t$ evolves under independent drift functions a^i, ensuring local independence between θ_t^i and θ_t^j for $j \neq i$. The driving noise w_t^i follows a Brownian motion.

To approximate pairwise distances between $\boldsymbol{\theta}_t$, we use the fact that $p(\mathbf{x}|\boldsymbol{\theta})$ is a linear transformation of $p(\mathbf{y}|\boldsymbol{\theta})$ [18], [19]. Since the densities are unknown, [18]–[20] approximate them using histograms $\mathbf{h}_t = (h_t^1, ..., h_t^{N_b})$ over a time window L_1. The expectation $\mathbb{E}[h_t^j]$ is also a linear transformation of $p(\mathbf{x}|\boldsymbol{\theta})$, allowing the noise-resilient Mahalanobis distance:

$$d^2(\mathbf{x}_t, \mathbf{x}_s) = \big(\mathbb{E}[\mathbf{h}_t] - \mathbb{E}[\mathbf{h}_s]\big)^T \big(\widehat{C}_t + \widehat{C}_s\big)^{-1} \big(\mathbb{E}[\mathbf{h}_t] - \mathbb{E}[\mathbf{h}_s]\big), \tag{3}$$

where $\widehat{C}_t$ and $\widehat{C}_s$ are local covariance matrices. Under certain conditions, this approximates the true state distances, i.e., $\|\boldsymbol{\theta}_t - \boldsymbol{\theta}_s\|^2 \simeq d^2(\mathbf{x}_t, \mathbf{x}_s)$ [19].

We propose an alternative Mahalanobis distance that eliminates histogram construction while preserving these properties. The resulting distances serve as input to PHATE [17], which converts distances into local affinities using an adaptive α-decay kernel, builds a diffusion process to capture global structure, and embeds potential distances via metric MDS [10], yielding robust visualizations. PHATE has been adapted previously for supervised learning [23], multi-scale analysis [24], and neural network geometry [25].

II. Functional Information Geometry

Constructing histograms and covariance matrices for Eq. 3 in high-dimensional spaces is challenging due to the curse of dimensionality. Since the noise in Eqs. 1 and 2 induces a linear transformation in probability space, we propose a noise-resilient distance that avoids density estimation using functional data analysis. To achieve this, we define the Mahalanobis distance between functions, leveraging concepts from [21], [22]. However, direct application of the FDA framework is unsuitable due to: 1) We treat probability densities centered at data points as functions, whereas standard FDA learns functions from data. 2) The densities may have multivariate inputs. 3) We need a Mahalanobis distance for functions (or densities) from different distributions.

A. Vector Mahalanobis Distance.

We will first define the vector Mahalanobis distance in terms of principal components (PCs) as the functional Mahalanobis distance will similarly use functional principal components.

Assume $\mathbf{u}$ and $\mathbf{v}$ have different distributions. Let the covariance matrices and means be C_u, C_v, $\mathbf{m}_u$, and $\mathbf{m}_v$, respectively. In [18], [20], the joint covariance between two observations was defined as $(C_u + C_v)$. Thus the squared Mahalanobis distance is given by:

$$d_M^2(\mathbf{u}, \mathbf{v}) = (\mathbf{u} - \mathbf{v})^T (C_u + C_v)^{-1} (\mathbf{u} - \mathbf{v}). \tag{4}$$

In [26], a Taylor expansion around the observable variables $\mathbf{u}$ and $\mathbf{v}$ are given, which yields the second-order approximation of the Euclidean distance between unobservable hidden processes $\boldsymbol{\theta}_u$ and $\boldsymbol{\theta}_v$:

$$\begin{aligned}\|\boldsymbol{\theta}_u - \boldsymbol{\theta}_v\|^2 = \frac{1}{2}(\mathbf{u} - \mathbf{v})^T (C_u^{-1} + C_v^{-1})(\mathbf{u} - \mathbf{v}) \\ + \mathcal{O}(\|\mathbf{u} - \mathbf{v}\|^4).\end{aligned} \tag{5}$$

We can thus instead define the squared squared vector Mahalanobis distance as:

$$d_M^2(\mathbf{u}, \mathbf{v}) = (\mathbf{u} - \mathbf{v})^T (C_u^{-1} + C_v^{-1})(\mathbf{u} - \mathbf{v}). \tag{6}$$

In [22], a method for computing the Mahalanobis distance using principal components was proposed. Let $C_u = V_u \Lambda_u V_u^T$ and $C_v = V_v \Lambda_v V_v^T$. The PC scores are $\mathbf{s}_{uu} = V_u^T(\mathbf{u} - \mathbf{m}_u)$, $\mathbf{s}_{vv} = V_v^T(\mathbf{v} - \mathbf{m}_v)$, $\mathbf{s}_{uv} = V_u^T(\mathbf{v} - \mathbf{m}_u)$, $\mathbf{s}_{vu} = V_v^T(\mathbf{u} - \mathbf{m}_v)$. Then the squared Mahalanobis distance is:

$$d_M^2(\mathbf{u}, \mathbf{v}) = ||\Lambda_u^{-1/2}(\mathbf{s}_{uu} - \mathbf{s}_{uv})||^2 + ||\Lambda_v^{-1/2}(\mathbf{s}_{vu} - \mathbf{s}_{vv})||^2. \tag{7}$$

B. Functional Mahalanobis Distance Between Densities.

We extend the previous distances to the functional setting, where functions are probability densities. Let f be a functional random variable that is also a probability density in $L^2(T)$, where $T \subseteq \mathbb{R}^d$. Define the density mean $\mu_f(t) = \mathbb{E}[f(t)]$ and the covariance operator Γ_f as

$$\Gamma_f(\eta) = \mathbb{E}[(f - \mu_f) \otimes (f - \mu_f)(\eta)], \tag{8}$$

where for any $\eta \in L^2(T)$,

$$(f - \mu_f) \otimes (f - \mu_f)(\eta) = \langle f - \mu_f, \eta \rangle (f - \mu_f), \tag{9}$$

with the L^2 inner product:

$$\langle f - \mu_f, \eta \rangle = \int_T (f(x) - \mu_f(x))\eta(x)dx.$$

In [21], an expression of f is given when Γ_f exists

$$f = \mu_f + \sum_{k=1}^{\infty} s_k \psi_k, \tag{10}$$

where $s_k = \langle f - \mu_f, \psi_k \rangle$ are the functional PC scores of f, ψ_k is the eigenfunction associated with eigenvalue λ_k with $\sum_{k=1}^{\infty} \lambda_k < \infty$ and $\Gamma_f(\psi_k) = \lambda_k \psi_k$ for all k.

In [27], a regularized inverse operator that approximates Γ_f^{-1} while preserving key properties was introduced. If Γ_f^{-1} exists, it is given by:

$$\Gamma_f^{-1}(\xi) = \sum_{k=1}^{\infty} \frac{1}{\lambda_k} (\psi_k \otimes \psi_k)(\xi).$$

The regularized inverse operator, denoted Γ_K^{-1}, is defined as:

$$\Gamma_K^{-1}(\xi) = \sum_{k=1}^{K} \frac{1}{\lambda_k} (\psi_k \otimes \psi_k)(\xi), \tag{11}$$

where K is a threshold, and ξ lies in the range of Γ [21], [22]. In practice, K should be tuned with the window size for convergence, while considering computational constraints. A relatively small K yields good results in our experiments.

We now adopt these quantities to densities and derive the corresponding Mahalanobis distance. Consider data points $\mathbf{x}_1, \ldots, \mathbf{x}_n$ with corresponding probability densities $f_1, \ldots, f_n$ with identical covariance operators. The functional Mahalanobis distance between functions f_i and f_j is denoted by $d_{FM}(f_i, f_j)$, and the squared distance is given by:

$$d_{FM}^2(f_i, f_j) = \left\langle \Gamma_K^{-1/2}(f_i - f_j), \Gamma_K^{-1/2}(f_i - f_j) \right\rangle. \tag{12}$$

Now let f_i and f_j be different distributions with mean functions μ_i and μ_j, and covariance operators Γ_i and Γ_j. Their regularized inverses are defined as before with eigenvalues λ_{ik} and λ_{jk}. Following Eqs. 6 and 12, we define our squared functional Mahalanobis distance between functions f_i and f_j:

$$\begin{aligned}d_{FM}^2(f_i, f_j) = \left\langle \Gamma_i^{-1/2}(f_i - f_j), \Gamma_i^{-1/2}(f_i - f_j) \right\rangle \\ + \left\langle \Gamma_j^{-1/2}(f_i - f_j), \Gamma_j^{-1/2}(f_i - f_j) \right\rangle.\end{aligned} \tag{13}$$

In [21], a derivation of (12) in terms of PCs was provided. Following a similar approach, we derive the PC version of (13):

$$d_{FM}^2(f_i, f_j) = \sum_{k=1}^{K} (\omega_{iik} - \omega_{ijk})^2 + \sum_{k=1}^{K} (\omega_{jik} - \omega_{jjk})^2, \tag{14}$$

where $\omega_{ijk} = s_{ijk}/\lambda_{ik}^{1/2}$, and s_{ijk} are functional PCs with $s_{ijk} = \langle f_j - \mu_i, \psi_{ik} \rangle$. Under certain assumptions, we have:

$$\|\theta_i - \theta_j\| = d_{FM}^2(f_i, f_j) + O(\|(f_i - f_j)\|^4).$$

C. Learning the Principal Component Scores

The final step is to derive the functional PC scores s_{ijk}. Unlike standard FPCA, which fits basis functions to data, we leverage probability density properties to compute empirical averages, avoiding direct density estimation.

We define "neighbors" for the densities $f_1, \ldots, f_n$, where each data point $\mathbf{x}_i$ is drawn from f_i, as the densities with corresponding points within a fixed time window centered at $\mathbf{x}_i$. Alternatively, neighbors can be determined by Euclidean distance or other criteria, allowing our proposed distance to be computed as long as a neighborhood structure is defined.

Assuming the densities $f_1, \ldots, f_n$ are known (a condition we will later relax), the mean function of f_i is estimated as $\mu_{f_i}(\mathbf{x}) \approx \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i} f_j(\mathbf{x})$, where $\mathcal{N}_i$ is the set of indices such that f_j is a neighbor of f_i with window size $\mathcal{L}_1$. In practice, different window sizes may be used for tasks like averaging basis functions or computing covariance matrices, as shown in Fig. 1. For simplicity, we use the same set of neighbors (and window size) for all tasks.

Now let's assume two data points $\mathbf{x}$ and $\mathbf{z}$ are drawn from f_i and f_j respectively. An estimate of the the covariance function associated with f_i is then

$$v_i(\mathbf{x},\mathbf{z}) = \frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}(f_l(\mathbf{x})f_l(\mathbf{z}) - \mu_{f_i}(\mathbf{x})\mu_{f_i}(\mathbf{z})). \tag{15}$$

The eigenequation of v_i is

$$\int v_i(\mathbf{x},\mathbf{z})\psi_i(\mathbf{z})d\mathbf{z} = \lambda_i\psi_i(\mathbf{x}). \tag{16}$$

We assume a basis expansion for the eigenfunction $\psi_i(\mathbf{x}) = \phi(\mathbf{x})^T\mathbf{b}_i$ where $\phi : \mathbb{R}^d \to \mathbb{R}^M$ is a set of basis functions such as the Fourier basis or splines. Using 15 and 16 and the definition of $\psi_i(\mathbf{x})$, we have the transformation:

$$\begin{aligned}\lambda_i\phi(\mathbf{x})^T\mathbf{b}_i &= \int v_i(\mathbf{x},\mathbf{z})\psi(\mathbf{z})d\mathbf{z} \\ &= \left[\frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i} f_l(\mathbf{x})\mathbf{a}_l^T - \mu_{f_i}(\mathbf{x})\frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\mathbf{a}_l^T\right]\mathbf{b}_i,\end{aligned} \tag{17}$$

where $\mathbf{a}_i = \int f_i(\mathbf{z})\phi(\mathbf{z})d\mathbf{z}$. The vector $\mathbf{a}_i$ is the expected value of the basis function ϕ with respect to the density f_i and can be approximated using a sample mean $\hat{\mathbf{a}}_i = \frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\phi(\mathbf{x}_l)$.

Define $W = \int \phi(\mathbf{x})\phi(\mathbf{x})^T d\mathbf{x}$. If ϕ is an orthonormal basis (such as the Fourier basis), then W is the identity matrix. Now multiply both sides of Eq. 17 by $\phi(\mathbf{x})$ and then integrate to obtain:

$$\begin{aligned}\lambda_i\int \phi(\mathbf{x})\phi(\mathbf{x})^T d\mathbf{x}\mathbf{b}_i &= \lambda_i W\mathbf{b}_i \\ &= \int \phi(\mathbf{x})\frac{1}{|\mathcal{N}_i|}\left[\sum_{l\in\mathcal{N}_i} f_l(\mathbf{x})\mathbf{a}_l^T - \mu_{f_i}(\mathbf{x})\sum_{l\in\mathcal{N}_i}\mathbf{a}_l^T\right]\mathbf{b}_i d\mathbf{x} \\ &= \left[\frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\mathbf{a}_l\mathbf{a}_l^T - \mu_{a_i}\mu_{a_i}^T\right]\mathbf{b}_i = \tilde{A}_i\mathbf{b}_i,\end{aligned} \tag{18}$$

where $\mu_{a_i} = \frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\mathbf{a}_l$ and $\tilde{A}_i = \frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\mathbf{a}_l\mathbf{a}_l^T - \mu_{a_i}\mu_{a_i}^T$. The matrix $\tilde{A}_i$ can be viewed as the sample covariance matrix of the vector $\mathbf{a}_i$. By letting $\mathbf{u_i} = W^{1/2}\mathbf{b}_i$, we get the following eigenequation:

$$W^{-1/2}\tilde{A}_i W^{-1/2}\mathbf{u}_i = \lambda_i\mathbf{u}_i. \tag{19}$$

Now let $\hat{\mu}_i = \frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\hat{\mathbf{a}}_l$ and approximate $\tilde{A}_i \approx \frac{1}{|\mathcal{N}_i|}\sum_{l\in\mathcal{N}_i}\hat{\mathbf{a}}_l\hat{\mathbf{a}}_l^T - \hat{\mu}_l\hat{\mu}_l^T$. If we find the first K eigenvalues and eigenvectors of Eq. 19, which we denote individually as λ_{ik} and $\mathbf{u}_{ik}$ for $k = 1, \ldots, K$, we then obtain the projections

$$s_{ijk} = (\hat{\mathbf{a}}_j - \hat{\mu}_i)^T W^{-1/2}\mathbf{u}_{ik}. \tag{20}$$

We then get $\omega_{ijk} = s_{ijk}/\lambda_{ik}^{1/2}$, which is used in Eq. 14 to obtain the final distance. This distance will then accurately approximate the distance between the parameters $\boldsymbol{\theta_t}$ as long as the approximations of $\mathbf{a}_i$ and $\tilde{A}_i$ are accurate. To embed the final distances into low dimensions, we input the distances into the PHATE algorithm to obtain the final FIG embedding.

FIG has the advantage of avoiding density estimation and basis fitting (unlike EIG and DIG). Instead, it requires estimating $\mathbf{a}_i$ vectors via empirical averaging of basis functions, which extends easily to higher dimensions.

Our final consideration is the numerical stability of the eigenvalues λ_{ik}. Some informative eigenvectors $\mathbf{u}_{ik}$ have low eigenvalues, and division by $\sqrt{\lambda_{ik}}$ amplifies numerical errors. Exponentiation offers a similar function shape but is more stable, so we use the following normalized PC scores:

$$\omega_{ijk} = s_{ijk}/e^{\lambda_{ik}}. \tag{21}$$

See Figure. 1 for a summary of all of the steps in FIG.

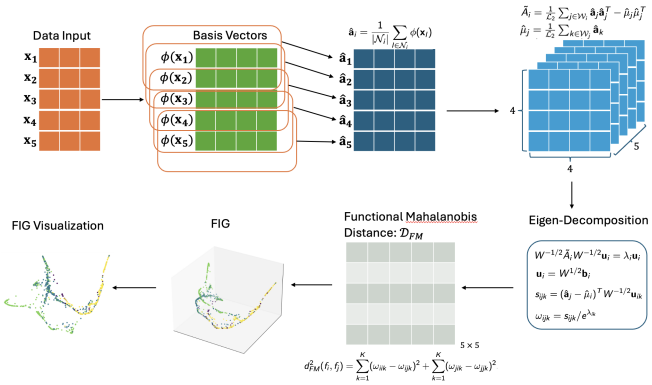

Fig. 1. Overview of the FIG algorithm. $\mathcal{N}_i$ has window size $\mathcal{L}_1$. $\mathcal{W}_i, \mathcal{W}_j$ are windows of points centered at $\mathbf{x}_i$ and $\mathbf{x}_j$ respectively with size $\mathcal{L}_2$.

III. Experiments

We conducted simulations inspired by [18] to model the movement of a radiating object across a 3D sphere, governed by horizontal (azimuth) angle θ_t^1 and vertical (elevation) angle θ_t^2. We represent the movement with the vector $\boldsymbol{\theta} = [\theta_t^1, \theta_t^2]$. In our experiments, we simulated 1000 random steps with added noise $\boldsymbol{\xi}_t$ as in Equation 1. Using the algorithm in Fig. 1 and 7 Fourier basis functions per dimension, we derived the distance matrix and obtained a 2D FIG embedding.

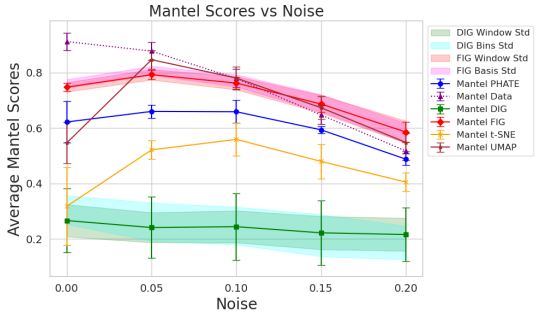

Fig. 2. Mantel coefficient between different embedding distances and the ground truth parameters $\boldsymbol{\theta}$ of the simulated random walk. FIG outperforms all methods in the high noise setting and is competitive in the low noise setting.

We added random Gaussian noise with mean zero and varying standard deviation σ. To assess noise robustness, we compared FIG with baseline methods: DIG, PHATE, UMAP, and t-SNE, increasing the noise until the Mantel coefficient between the noisy data and $\boldsymbol{\theta}$ dropped below 0.5. All embeddings were 2D. We used the Python APIs for PHATE, UMAP, and t-SNE: phate, umap, t-sne. For DIG and FIG, we tested window sizes from 10 to 30 for both $\mathcal{L}_1$ and $\mathcal{L}_2$, varying bins (DIG) from 10 to 30 and basis (FIG) from 5 to 11.

To assess global disparities among embeddings, we used the Mantel [28] test, which computes correlation coefficients (0 to 1) between two distance sets, accounting for the interdependence of distances. We first calculate pairwise Euclidean distances for the embeddings and the dynamical process $\boldsymbol{\theta}$, then compute their Mantel correlations. For reproducibility, we used five random seeds per noise level. Figure 2 shows the mean Mantel correlations (solid lines) with standard deviations (error bars). A dotted purple line represents the correlations computed with noisy data. Shaded regions indicate sensitivity to window sizes, histogram bins (DIG), and basis functions (FIG). We observe that as noise increases, the Mantel coefficient between the data and $\boldsymbol{\theta}$ drops significantly. In the noiseless setting, no method outperforms the original data, though FIG performs best among embeddings. As noise rises, all methods degrade, but FIG remains superior, is less sensitive to hyperparameters, and is more noise-resilient. At noise level 0.15, FIG surpasses the original data, with a notably large gap over DIG, highlighting its superior structure capture.

We apply FIG to EEG data from [29], [30], with results in Figures 3 and 4. The dataset is an 18-dimensional multivariate time series, sampled at 512Hz, classified into six sleep stages per R&K rules (REM, Awake, S-1, S-2, S-3, S-4), each spanning 30 seconds. To address data sparsity, we merge S-1 with S-2 and S-3 with S-4. The data is band-filtered (8-40 Hz) and down-sampled to 128Hz.

Given the dataset's large size (3M samples), we preprocess it using Fourier basis functions, averaging $\phi(\mathbf{x}_t)$ over segments. Following Fig. 1, we use seven Fourier basis functions per dimension. Using the same DIG [20] settings, we set $\mathcal{L}_1 = 3840$, matching the number of observations in a 30-second interval. The final embeddings are 2-D.

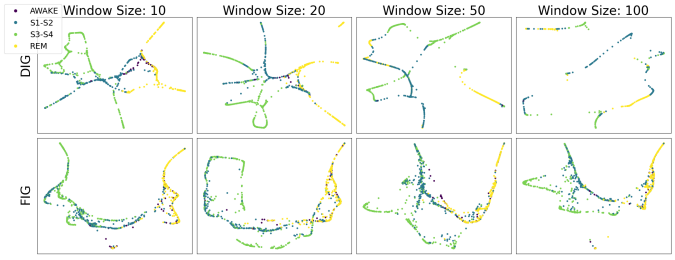

Fig. 3. A visual comparison of FIG and DIG on EEG brain measurements during different sleep stages. FIG is more robust to different window sizes than DIG.

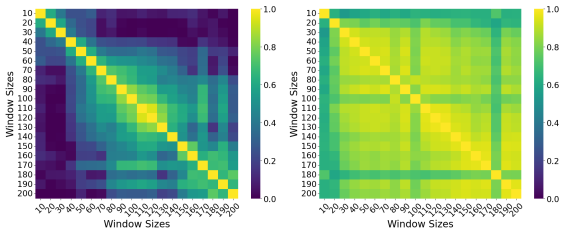

Fig. 4. A comparison of DIG (left) and FIG (right) Mantel scores on EEG brain measurements. FIG is more robust to different window sizes than DIG.

For histogram estimation, DIG uses $N_b = 20$ bins per dimension, while FIG and DIG embeddings are computed across window sizes $\mathcal{L}_2 = 10, 20, \ldots, 200$. Figure 3 compares 2-D embeddings of FIG and DIG across $\mathcal{L}_2$, while Figure 4 shows mean Mantel correlations across 20 window sizes, computed over five random seeds.

From Figure 3, we observe qualitatively that as the window size increases, the 2-dimensional embeddings of DIG tend to lose important structural information such as the connections between different sleep stages. In contrast, the embeddings of FIG remain stable while providing similar branching and trajectory structures as DIG with smaller window sizes. This suggests that FIG is more resilient to the choice of window size than DIG. Figure 4 corroborates this numerically. Here we observe that the FIG embeddings with different window sizes have higher Mantel correlation coefficients with each other than DIG. On the other hand, we observe a high Mantel correlation for FIG among different window sizes, which corresponds to the visualization results.

IV. Conclusion

We developed FIG, a novel visualization method using functional data analysis to compute a noise-resilient distance in probability space. Unlike EIG and DIG, FIG avoids density estimation, making it ideal for high-dimensional data. Experiments show FIG's robustness, with stable visualizations across different window lengths for Mahalanobis distances. A limitation is that we have only applied FIG to time series data due to the need for a "time window." However, as discussed in Section II, FIG can be extended to non-time series data by defining appropriate neighborhoods, which we plan to explore in future work.

REFERENCES

[1] M. Lezcano Casado, "Trivializations for gradient-based optimization on manifolds," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[2] V. Verma, A. Lamb, C. Beckham, A. Najafi, I. Mitliagkas, D. Lopez-Paz, and Y. Bengio, "Manifold mixup: Better representations by interpolating hidden states," in *International conference on machine learning*. PMLR, 2019, pp. 6438–6447.

[3] P. Rodríguez, I. Laradji, A. Drouin, and A. Lacoste, "Embedding propagation: Smoother manifold for few-shot classification," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVI 16*. Springer, 2020, pp. 121–138.

[4] M. Dai, H. Hang, and X. Guo, "Adaptive feature interpolation for low-shot image generation," in *European Conference on Computer Vision*. Springer, 2022, pp. 254–270.

[5] X. Luo, Z. Han, and L. Yang, "Progressive attentional manifold alignment for arbitrary style transfer," in *Proceedings of the Asian Conference on Computer Vision*, 2022, pp. 3206–3222.

[6] X. Zhen, R. Chakraborty, N. Vogt, B. B. Bendlin, and V. Singh, "Dilated convolutional neural networks for sequential manifold-valued data," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 10 621–10 631.

[7] R. Wang, X.-J. Wu, Z. Chen, T. Xu, and J. Kittler, "Dreamnet: A deep riemannian manifold network for spd matrix learning," in *Proceedings of the Asian Conference on Computer Vision*, 2022, pp. 3241–3257.

[8] K. R. Moon, J. S. Stanley III, D. Burkhardt, D. van Dijk, G. Wolf, and S. Krishnaswamy, "Manifold learning-based methods for analyzing single-cell rna-sequencing data," *Current Opinion in Systems Biology*, vol. 7, pp. 36–46, 2018.

[9] M. Balasubramanian and E. L. Schwartz, "The isomap algorithm and topological stability," *Science*, vol. 295, no. 5552, pp. 7–7, 2002.

[10] T. F. Cox and M. A. Cox, *Multidimensional scaling*. CRC press, 2000.

[11] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[12] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[13] L. Van der Maaten and G. Hinton, "Visualizing data using t-sne." *Journal of machine learning research*, vol. 9, no. 11, 2008.

[14] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *arXiv preprint arXiv:1802.03426*, 2018.

[15] R. R. Coifman and S. Lafon, "Diffusion maps," *Appl. Comput. Harmon. Anal.*, vol. 21, no. 1, pp. 5–30, 2006.

[16] L. Haghverdi, M. Büttner, F. A. Wolf, F. Buettner, and F. J. Theis, "Diffusion pseudotime robustly reconstructs lineage branching," *Nature methods*, vol. 13, no. 10, pp. 845–848, 2016.

[17] K. R. Moon, D. Van Dijk, Z. Wang, S. Gigante, D. B. Burkhardt, W. S. Chen, K. Yim, A. v. d. Elzen, M. J. Hirn, R. R. Coifman *et al.*, "Visualizing structure and transitions in high-dimensional biological data," *Nature biotechnology*, vol. 37, no. 12, pp. 1482–1492, 2019.

[18] R. Talmon and R. Coifman, "Empirical intrinsic geometry for nonlinear modeling and time series filtering," *Proceedings of the National Academy of Sciences*, vol. 110, no. 31, p. 12535– 12540, 2013.

[19] ——, "Intrinsic modeling of stochastic dynamical systems using empirical geometry," *Applied and Computational Harmonic Analysis*, vol. 39, no. 1, p. 138–160, 2015.

[20] A. F. Duque, G. Wolf, and K. R. Moon, "Visualizing high dimensional dynamical processes," *IEEE International Workshop on Machine Learning for Signal Processing*, pp. 1–6, 2019.

[21] P. Galeano, E. Joseph, and R. E. Lillo, "The mahalanobis distance for functional data with applications to classification," *Technometrics*, vol. 57, no. 2, pp. 281–291, 2015.

[22] J. Ramsay and B. Silverman, "Principal components analysis for functional data," *Functional data analysis*, pp. 147–172, 2005.

[23] J. S. Rhodes, A. Cutler, G. Wolf, and K. R. Moon, "Random forest-based diffusion information geometry for supervised visualization and data exploration," in *2021 IEEE Statistical Signal Processing Workshop (SSP)*. IEEE, 2021, pp. 331–335.

[24] M. Kuchroo, J. Huang, P. Wong, J.-C. Grenier, D. Shung, A. Tong, C. Lucas, J. Klein, D. B. Burkhardt, S. Gigante *et al.*, "Multiscale phate identifies multimodal signatures of covid-19," *Nature biotechnology*, vol. 40, no. 5, pp. 681–691, 2022.

[25] S. Gigante, A. S. Charles, S. Krishnaswamy, and G. Mishne, "Visualizing the phate of neural networks," *Advances in neural information processing systems*, vol. 32, 2019.

[26] A. Singer and R. Coifman, "Non linear independent component analysis with diffusion maps," *Applied and Computational Harmonic Analysis*, vol. 25, no. 2, p. 226–239, 2008.

[27] A. Mas, "Weak convergence in the functional autoregressive model," *Journal of Multivariate Analysis*, vol. 98, no. 6, pp. 1231–1261, 2007.

[28] N. Mantel, "The detection of disease clustering and a generalized regression approach," *Cancer research*, vol. 27, no. 2 Part 1, pp. 209–220, 1967.

[29] M. Terzano, L. Parrino, A. Smerieri, R. Chervin, S. Chokroverty, C. Guilleminault, M. Hirshkowitz, M. Mahowald, H. Moldofsky, A. Rosa, R. Thomas, and A. Walters, "Atlas, rules, and recording techniques for the scoring of cyclic alternating pattern (cap) in human sleep," *Sleep medicine, vol. 3, no. 2*, p. 187–199, 2002.

[30] A. Goldberger, L. Amaral, J. H. L. Glass, P. Ivanov, R. Mark, J. Mietus, G. Moody, C.-K. Peng, and H. Stanley, "Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, p. e215–e220, 2000.