Dynamical Wasserstein Barycenters for Time-series Modeling

Anonymous Author(s)
Affiliation
Address
email

Abstract

Many time series can be modeled as a sequence of segments representing high-level discrete states, such as running and walking in a human activity application. Flexible models should describe the system state and observations in stationary "pure-state" periods as well as transition periods between adjacent segments, such as a gradual slowdown between running and walking. However, most prior work assumes instantaneous transitions between pure discrete states. We propose a dynamical model that can estimate the system state over time as well as the data-generating distributions of pure states in an unsupervised manner. Our model assumes each pure state generates data from a multivariate normal distribution, and characterizes transitions between states via the displacement-interpolation specified by the Wasserstein barycenter. The system state is represented by a simplex-valued barycentric weight vector which evolves over time via a random walk on the simplex. Parameter learning leverages the natural Riemannian geometry of Gaussian distributions under the Wasserstein distance, which leads to improved convergence speeds. Experiments on several human activity datasets show that our proposed Wasserstein-barycentric dynamical model accurately learns the generating distribution of pure states while improving state estimation for transition periods compared to the commonly used linear interpolation mixture models.

1 Introduction

We consider the problem of estimating the dynamically evolving state of a system from time-series data. The notion of "state" in such contexts typically is modeled in one of two ways. For many problems, the system state is a vector of continuously-valued quantities (Kalman, 1960), perhaps constrained in some manner. Estimation for this class of problems often employs some form of a state-space model describing the transition dynamics of the state evolution and an observation model relating the system state to the observed time series. Alternatively, discrete-state models take on one of a countable number of options at each point in time. In this domain, Hidden Markov models (HMMs) (Rabiner, 1989), applied extensively in areas such as speech processing (Gales and Young, 2007), are comprised of transition probabilities describing the state transition dynamics as well as emission distributions specifying the probabilistic model of the data observed in each discrete state.

Many time-series characterization problems of current interest warrant a hybrid of the aforementioned approaches where the system transitions in a continuous manner among a finite collection of "pure-states." For example, in human activity recognition (Bi et al., 2021) with accelerometer sensor data, there do indeed contain segments which correspond to distinct activities (run, sit, walk, etc.). Additionally, the sampling rates of the data are such that these pure-state segments are separated by distinct periods where the individual is proceeding from one such state to another (e.g. acceleration from standing to running over 5 seconds). Other problems that may exhibit similar behaviors include

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

tracking the evolution of economic (Chang et al., 2016) or climate (Chang et al., 2020) metrics. Characterizing these systems requires a model with a continuous state space to capture the gradual evolution of the system among the discrete set of pure-states.

Motivated by this class of applications, we consider models for time series in which the system's dynamical state is specified by a vector of convex combination weights for mixing a set of data-generating distributions that define each "pure" state. In many models, such a simplex-constrained state vector (Rudin, 1976) is interpreted as a set of assignment probabilities; that is, the system is taken to be in a pure state with uncertainty as to which. As a result, the probability model for intermediate states is a convex, *linear* combination of the pure-state emission distributions. While useful in many applications, this model fails to capture the transient behavior of many time series. Consider human activity in the simulated framework shown in Fig. 1 where the emission distribution of three actions (e.g. walk, run, jump) are given by ρ^1, ρ^2, ρ^3. In a scenario where the individual smoothly transitions from walk to run and then to jump, linear mixing yields the multi-modal distributions shown in Fig. 1(b). This approach models the walk-to-run transition as sometimes walk and sometimes run, which does not intuitively capture the nature of state transitions in this system.

A more suitable means of mixing pure-state distributions for our purposes is provided by displacement-interpolation (McCann, 1997), which is related to the Wasserstein distance (Peyré and Cuturi, 2019), a metric over the space of probability distributions (Sriperumbudur et al., 2010). Referring now to Fig. 1(c), this approach to combining probability models produces distributions whose mass is distributed in a manner that more intuitively captures the notion of the system being *between* two pure states. While the work of (McCann, 1997) is limited to the combination of two distributions, it is extended to multiple distributions using the notion of a Wasserstein barycenter (Agueh and Carlier, 2011). With respect to implementation, the Wasserstein distance applied to the space of Gaussian distributions has convenient geometric properties (Takatsu, 2011) allowing for closed form computation of the distance, barycenter, and application of Riemannian manifold optimization which we leverage extensively in this paper.

Inspired by this displacement-interpolation framework, we model a time series whose underlying probability distribution is the Wasserstein barycentric combination of a collection of Gaussian pure states with a time-varying barycentric weight that evolves on the probability simplex. To estimate the underlying distribution of the data at a given time point, we build on ideas from the statistical literature and characterize the structure of data in terms of the empirical distribution derived from a sliding window of samples (Aghabozorgi et al., 2015). Given these empirical distributions, our proposed estimation problem learns both the pure-state probability models and the time history of the simplex-valued barycentric mixing weights.

Motivated by state-space models with smooth state evolutions given by white noise driven Markov models in continuous state settings (Welch, 1997), our model uses the barycentric weight vector to represent the system state, which adds the extra constraint that it must lie in the simplex at every point in time. Dynamics driven by white Gaussian noise will generally not respect this constraint. Building on the approach taken in (Nguyen and Volkov, 2020), we employ a random walk where the driving noise is solely provided by a sequence of transition vectors consisting of independent, identically distributed (IID) random variables drawn from a mixture of two Beta distributions, representing stationary and transitional dynamics respectively. Combining this with the state vector in a convex manner, results in a simplex-value process.

Contributions: We introduce a *displacement-interpolation model for time series* where the data-generating distribution is given by the weighted Wasserstein barycenter of a set of pure-state emission distributions and a time-varying state vector. We propose a *simplex-valued random walk* with flexible dynamical structure to model the system state. We exploit the *Riemannian structure of Gaussian distributions* under the Wasserstein distance for parameter estimation for faster convergence speed. We evaluate on *human activity data* the ability of our method to capture *stationary and transition dynamics*, comparing with the linear interpolation mixture model.

Outline: Sec. 2 provides an overview of the Wasserstein distance, barycenter, and the associated geometry for Gaussian distributions. We then formalize our problem statement and estimation problem in Sec. 3. Sec. 4 discusses the model parameters, covering the dynamical simplex state-space model in Sec. 4.1 and the pure-state parameters in Sec. 4.2. Sec. 5 discusses the optimization of our model parameters leveraging geometric properties of the Wasserstein distance for Gaussians. Finally, Sec. 6 evaluates and discusses the advantages of our model in the context of human activity data.

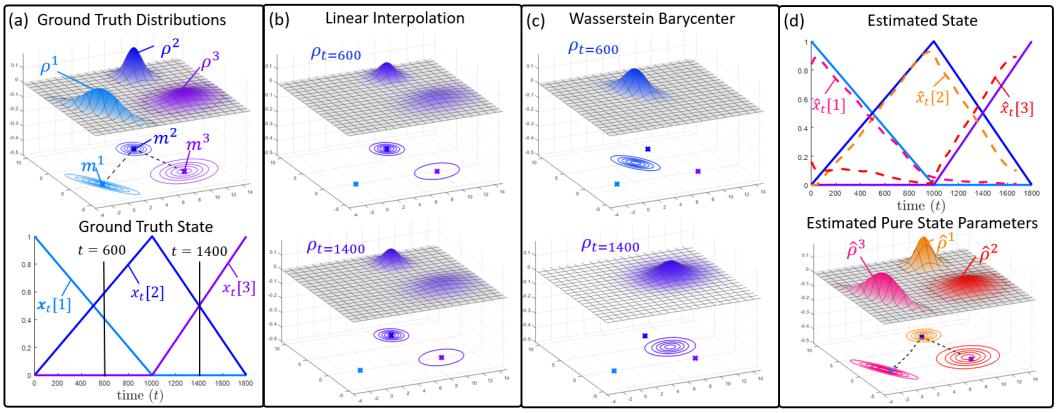

Figure 1: (a) Three Gaussian distributions ρ^1, ρ^2, ρ^3 each representing distinct activities with corresponding means $\boldsymbol{m}^1, \boldsymbol{m}^2, \boldsymbol{m}^3$ that are marked ('x') in all plots as reference points. The time series is drawn from a time-varying distribution according to the ground truth state vector as the system transitions linearly from ρ^1 to $\rho^2, t = 1, ..., 1000$, then continues to $\rho^3, t = 1000, ..., 1800$. (b) Under the linear interpolation state-transition model, the PDF at select times of $t = 600, 1400$ are linear combinations of ρ^1, ρ^2, ρ^3. (c) Alternatively, the proposed displacement-interpolation transition model between pure-states given by the Wasserstein barycenter translates the mass between pure states. (d) Following the Wasserstein barycentric model for time series, our proposed method accurately recovers both the pure state distributions and state vector from the observed time series.

2 Technical Background

A core component to our approach is to model the intermediate transition states of a time series using the Wasserstein barycenter (Agueh and Carlier, 2011) of probability distributions, which generalizes the displacement-interpolation framework (McCann, 1997) beyond two distributions. We refer the works of (Peyré and Cuturi, 2019), (Villani, 2009) for a detailed discussion on these concepts.

Consider the space of all Borel probability measures over $\mathbb{R}^d$ with finite second moment. The squared Wasserstein-2 distance for two distributions ρ_1, ρ_2 with squared Euclidean ground cost is defined as,

$$\mathcal{W}_2^2(\rho_1, \rho_2) = \inf_{T \in \Pi(\rho_1, \rho_2)} \int_{\mathbb{R}^d \times \mathbb{R}^d} \|\boldsymbol{\alpha} - \boldsymbol{\beta}\|_2^2 \, T(d\boldsymbol{\alpha}, d\boldsymbol{\beta}), \tag{1}$$

where $\Pi(\rho_1, \rho_2)$ is the set of all joint distributions with marginals ρ_1, ρ_2, and T is the optimal transport plan, the element that minimizes the total transportation cost. When these measures are Gaussian distributions, parameterized by their mean vectors $\boldsymbol{m}_i \in \mathbb{R}^d$, and their symmetric positive-definite covariance matrices $\boldsymbol{S}_i \in Sym_+^d$, the square of the Wasserstein-2 distance has a closed form solution (Takatsu, 2011),

$$\mathcal{W}_2^2\left(\rho_1(\boldsymbol{m}_1, \boldsymbol{S}_1), \rho_2(\boldsymbol{m}_2, \boldsymbol{S}_2)\right) = \underbrace{\|\boldsymbol{m}_1 - \boldsymbol{m}_2\|_2^2}_{\mathcal{E}^2(\boldsymbol{m}_1, \boldsymbol{m}_2)} + \underbrace{\operatorname{tr}\left(\boldsymbol{S}_1 + \boldsymbol{S}_2 - 2\left(\boldsymbol{S}_1^{\frac{1}{2}} \boldsymbol{S}_2 \boldsymbol{S}_1^{\frac{1}{2}}\right)\right)}_{\mathcal{B}^2(\boldsymbol{S}_1, \boldsymbol{S}_2)}. \tag{2}$$

This distance decomposes into sum of the squared Euclidean distance between mean vectors, $\mathcal{E}^2(\boldsymbol{m}_1, \boldsymbol{m}_2)$, and the squared Bures distance (Bhatia et al., 2017) between covariance matrices, $\mathcal{B}^2(\boldsymbol{S}_1, \boldsymbol{S}_2)$. Thus, the contributions of the mean and covariance to the Wasserstein distance between Gaussians are decoupled, a property that is uncommon for Gaussian distribution distances (Nagino and Shozakai, 2006) and has important implications for optimization and barycenter computation.

The Wasserstein barycenter extends the notion of the weighted average of points in $\mathbb{R}^d$ with the Euclidean distance to the space of probability distributions with the Wasserstein distance. Given a set of measures and the barycentric coordinate vector on the K-simplex, $\boldsymbol{x} \in \Delta^K$, the Wasserstein barycenter is the measure that minimizes this weighted Wasserstein distance to the set of measures.

$$\rho_B\left(\boldsymbol{x}, \{\rho_k\}_{k=1}^K\right) = \operatorname*{argmin}_{\rho \in \mathcal{P}_2(\mathbb{R}^d)} \sum_{k=1}^K \boldsymbol{x}[k] \mathcal{W}_2^2(\rho_k, \rho). \tag{3}$$

When ρ_k are Gaussian distributions, ρ_B defined in (3) is itself Gaussian (Agueh and Carlier, 2011) with parameters $\boldsymbol{m}_B, \boldsymbol{S}_B$. Again, because of the decomposition of the Wasserstein distance in (2), the Wasserstein barycentric problem in (3) can be solved separately for its components,

$$\boldsymbol{m}_B = \underset{\boldsymbol{m}\in\mathbb{R}^d}{\operatorname{argmin}} \sum_{k=1}^{K} \boldsymbol{x}[k]\mathcal{E}^2(\boldsymbol{m}_k, \boldsymbol{m}), \qquad \boldsymbol{S}_B = \underset{\boldsymbol{S}\in Sym_+^d}{\operatorname{argmin}} \sum_{k=1}^{K} \boldsymbol{x}[k]\mathcal{B}^2(\boldsymbol{S}_k, \boldsymbol{S}). \tag{4}$$

While $\boldsymbol{m}_B = \sum_k \boldsymbol{x}[k]\boldsymbol{m}_k$, the covariance component can be solved via the fixed-point iteration proposed in Álvarez Esteban et al. (2016).

3 Problem Formulation

Data Model. To apply this barycentric model to a discrete-time *time series*, we imagine a set of emission distributions ρ^k, each associated with a pure state. Given these, at each time t a simplex-valued state vector $\boldsymbol{x}_t$ is used to define the current time's distribution ρ_{B_t} according to the Wasserstein barycentric model given by (3). By assuming ρ^k are Gaussian, the resulting Gaussian barycenters are computed in closed form according to (4).

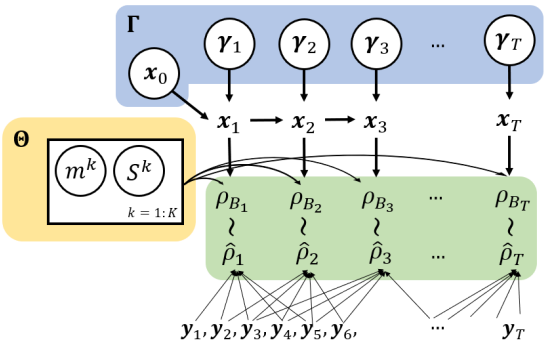

Figure 2: Time-series data model predicts a distribution ρ_{B_t} that is the Wasserstein barycenter given pure-state distributions with parameters $\boldsymbol{m}^k$, $\boldsymbol{S}^k$ and time-varying barycentric weights $\boldsymbol{x}_t$, a Markov chain driven by the sequence γ_t and initial state $\boldsymbol{x}_0$. Random variables are denoted by circles. Colors correspond to the contribution to the estimation problem in (9)

To fit such a model to observed vector-valued data $\boldsymbol{y}_1, ..., \boldsymbol{y}_T$, we compute an *empirical* Gaussian distribution $\hat{\rho}_t = \mathcal{N}(\boldsymbol{m}_t, \boldsymbol{S}_t)$ at each time t using a sliding window of $(2n+1)$ samples,

$$\boldsymbol{m}_t = \frac{1}{(2n+1)} \sum_{i=-n}^{n} \boldsymbol{y}_{t+i}, \qquad \boldsymbol{S}_t = \frac{1}{2n} \sum_{i=-n}^{n} (\boldsymbol{y}_{t+i} - \boldsymbol{m}_t)(\boldsymbol{y}_{t+i} - \boldsymbol{m}_t)^T, \tag{5}$$

where we are careful to use the unbiased estimators of these quantities.

Minimizing the Wasserstein distance between this sequence of empirical distributions $\hat{\rho}_t$ and the model-predicted distributions ρ_{B_t} drives our model's parameter learning.

Problem Formulation. We are given a vector-valued time series $\boldsymbol{y}_t$, $t = 1, ..., T$ that is modeled as being drawn from a time-varying Gaussian distribution ρ_t that is the Wasserstein barycenter of a set K of Gaussian distributions $\rho^k \equiv \mathcal{N}(\boldsymbol{m}^k, \boldsymbol{S}^k)$ with time-varying barycentric weights $\boldsymbol{x}_t$ (see Fig. 2). Given $\boldsymbol{y}_t$ and K, we wish to estimate the model whose parameters consist of a set of pure-state Gaussian distributions $\boldsymbol{\Theta} \equiv \left\{(\boldsymbol{m}^k, \boldsymbol{S}^k)\right\}_{k=1}^{K}$, and an initial state with a sequence of simplex-valued state vectors $\boldsymbol{X} \equiv \left\{\boldsymbol{x}_0, \{\boldsymbol{x}_t\}_{t=1}^{T}\right\}$.

The model predicted barycentric distribution $\rho_{B_t}(\boldsymbol{x}_t, \boldsymbol{\Theta}) \equiv N(\boldsymbol{m}_{B_t}, \boldsymbol{S}_{B_t})$ is a function of $\boldsymbol{x}_t$ and $\boldsymbol{\Theta}$ via (4). We pose the estimation of $\boldsymbol{X}$ and $\boldsymbol{\Theta}$ as the solution the following optimization problem seeking to balance fidelity to a prior model on the parameters of interest with the desire to minimize the time integrated Wasserstein distance between the model predicted and empirically observed distributions:

$$\hat{\boldsymbol{X}}, \hat{\boldsymbol{\Theta}} = \underset{\boldsymbol{X},\boldsymbol{\Theta}}{\operatorname{argmin}} - \log\left(p(\boldsymbol{X})p(\boldsymbol{\Theta})\right) + \lambda \sum_{t=1}^{T} \mathcal{W}_2^2(\hat{\rho}_t, \rho_{B_t}(\boldsymbol{x}_t, \boldsymbol{\Theta})). \tag{6}$$

The scalar weight $\lambda > 0$ trades off the the fit of the model barycenters to the empirical distributions with the probability of the state $\boldsymbol{X}$ and parameters $\boldsymbol{\Theta}$ under assumed *priors*. Our assumed probabilistic priors for the transition dynamics of the state vector on the probability simplex and pure-state parameters are covered in the following section.

4 Model Parameter Priors

4.1 Dynamical Simplex State-Vector

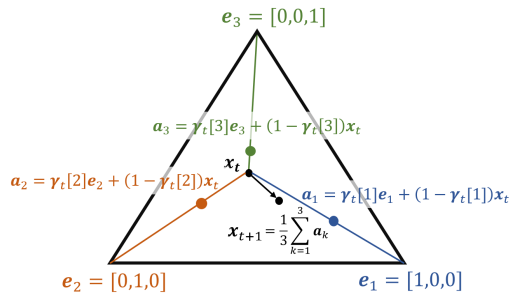

Figure 3: The update of the $\boldsymbol{x}_t$ is given by the transition step vector $\boldsymbol{\gamma}_t$. For each $k = 1, ..., K$, $\boldsymbol{\gamma}_t[k] \in [0, 1]$ represents a proportional step from $\boldsymbol{x}_t$ to the corresponding simplex vertex $\boldsymbol{e}_k$. The update to the state $\boldsymbol{x}_{t+1}$ is given by the average of each of the K update component.

Here we develop the dynamical model that generates the state vector $\boldsymbol{x}_t$. The geometry of the state space in the case of Δ^3 is shown in Fig. 3 where $\boldsymbol{x}_t$ lies in the convex hull of the three simplex vertices, the unit coordinate vectors $\boldsymbol{e}_k,\ k = 1, 2, 3$ each of which is associated with a pure-state in our problem. This is generalized to Δ^K for a K-state problem in an obvious manner. To ensure that $\boldsymbol{x}_{t+1}$ remains in this region, we define its update by a K-dimensional "innovations" vector, $\boldsymbol{\gamma}_t$. As seen in Fig. 3(a), $\boldsymbol{\gamma}_t[k] \in [0, 1]$ represents a proportional step from the current state $\boldsymbol{x}_t$ to $\boldsymbol{e}_k$ where a step of $\boldsymbol{\gamma}_t[k] = 0$ leaves the k-th component update at $\boldsymbol{x}_t$ while $\boldsymbol{\gamma}_t[k] = 1$ jumps to the vertex $\boldsymbol{e}_k$. Unlike prior methods such as Nguyen and Volkov (2020), we repeat this process for each of the k components and average their respective results to achieve the final update to $\boldsymbol{x}_t$, allowing the simplex-process to be driven solely by $\boldsymbol{\gamma}_t$. By construction, the resulting state update to $\boldsymbol{x}_t$ is constrained to K-simplex so long as $\boldsymbol{\gamma}_t[k] \in [0, 1]$ and is given by,

$$\boldsymbol{x}_t = \left(1 - \frac{1}{K}\sum_{k=1}^{K}\boldsymbol{\gamma}_t[k]\right)\boldsymbol{x}_{t-1} + \frac{1}{K}\boldsymbol{\gamma}_t. \tag{7}$$

Inspired by ideas from dynamical Bayesian nonparametric models (Ren et al., 2008), the prior over innovations γ_t is based on the Beta distribution (Yates and Goodman, 2005). Specifically, we use a two-component Beta mixture to capture *stationary* and *transition* regions using distinct parameters for each pure-state component k. We model these innovations as IID across time.

$$p\left(\{\boldsymbol{\gamma}_t\}_{t=1}^{T}\right) = \prod_{t=1}^{T}\prod_{k=1}^{K}\boldsymbol{w}[k]\text{Beta}\left(\boldsymbol{\gamma}_t[k]; \boldsymbol{a_0}[k], \boldsymbol{b_0}[k]\right) + (1 - \boldsymbol{w}[k])\,\text{Beta}\left(\boldsymbol{\gamma}_t[k]; \boldsymbol{a}_1[k], \boldsymbol{b}_1[k]\right). \tag{8}$$

This choice of this Beta-based prior for $\boldsymbol{\gamma}_t$ provides for significant flexibility in the manner in which $\boldsymbol{x}_t$ evolves on the simplex. Ensuring the Beta parameters $\boldsymbol{a}, \boldsymbol{b} > 1$ gives a uni-modal distribution on $[0, 1]$ with mean $\frac{\boldsymbol{a}}{\boldsymbol{a}+\boldsymbol{b}}$. The dynamics in the stationary regions expect innovations that are close to zero. This can be captured with a Beta distribution with $\boldsymbol{b}[k] >> \boldsymbol{a}[k]$. However, the transition dynamics can vary from between systems and even within pure states within a system. Therefore, in our implementation we learn the parameters that characterize transitions, while fixing stationary component at $\boldsymbol{a}_0[k] = 20, \boldsymbol{b}_0[k] = 1.1$. Our learnable parameters are denoted $\boldsymbol{H} \equiv \{\boldsymbol{w}, \boldsymbol{a}_1, \boldsymbol{b}_1\}$. To prevent mode collapse we constrain $\boldsymbol{a}_1 > 1.1$, $\frac{\boldsymbol{a}_1}{\boldsymbol{a}_1+\boldsymbol{b}_1} > 0.15$, and $w \in [0.01, 0.99]$.

As mentioned in Sec. 3, the simplex-state $\boldsymbol{x}_t$ represents the barycentric mixing weights used to compute the model-predicted distribution of the data. This sequence of states, however, is deterministic given the initial state and the subsequent sequence of innovation vectors according to (7). Since these innovations are the random variables of interest, it proves convenient to replace $\boldsymbol{X}$ in (6) with $\boldsymbol{\Gamma} \equiv \left\{\boldsymbol{x}_0, \{\boldsymbol{\gamma}_t\}_{t=1}^{T}\right\}$ resulting in the estimation problem,

$$\hat{\boldsymbol{\Gamma}}, \hat{\boldsymbol{\Theta}}, \hat{\boldsymbol{H}} = \underset{\boldsymbol{\Gamma},\boldsymbol{\Theta},\boldsymbol{H}}{\text{argmin}} \underbrace{-\log\left(p_{\boldsymbol{H}}(\boldsymbol{\Gamma})\ p(\boldsymbol{\Theta})\right) + \lambda \sum_{t=1}^{T}\mathcal{W}_2^2(\hat{\rho}_t, \rho_{B_t}(\boldsymbol{\Gamma}, \boldsymbol{\Theta}))}_{F\left(\boldsymbol{\Gamma},\boldsymbol{\Theta},\boldsymbol{H},\{\hat{\rho}_t\}_{t=1}^{T}\right)}. \tag{9}$$

In our implementation, we are indifferent to the initial state and therefore set $p(\boldsymbol{x}_0)$ as uniform over the simplex. The coloration in (9) is linked to the associated parameters in Fig. 2.

4.2 Pure-State Distributions

The final component to (9) is the prior on the pure-state parameters. By setting a reference normal distribution $\mathcal{N}(\boldsymbol{m}_0, \sigma_0\boldsymbol{I})$ we can define a probability density function over the space of all Gaussian

distributions derived from the Wasserstein distance to this reference distribution according to,

$$
\begin{aligned}
p(\boldsymbol{m}, \boldsymbol{S}) =& \kappa(s, \sigma_0) \exp\left\{-\frac{1}{2s^2}\mathcal{W}_2^2\left((\boldsymbol{m}, \boldsymbol{S}), (\boldsymbol{m}_0, \sigma_0^2\boldsymbol{I})\right)\right\} \\
=& \kappa(s, \sigma_0) \exp\left(-\frac{1}{2s^2}\left\|\boldsymbol{q} - \boldsymbol{q}_0\right\|_2^2\right),
\end{aligned} \tag{10}
$$

where s is a scalar hyperparameter that controls the variance of this prior. It can be shown that (10) is equivalent to a multivariate Gaussian distribution over $\boldsymbol{q} \equiv [\boldsymbol{m}, \boldsymbol{\omega}] \in \mathbb{R}^{2d}$, the joint space of means $\boldsymbol{m}$ and the eigenvalues $\boldsymbol{\omega}$ of the covariance matrices $\boldsymbol{S} \in Sym_+^d$. This Gaussian has mean $\boldsymbol{q}_0 = [\boldsymbol{m}_0, \sigma_0, ..., \sigma_0]$ and covariance equal to $s\boldsymbol{I}$. It follows that the normalizing constant needed for (10) to be a valid distribution given the normal CDF function Φ is $\kappa(s, \sigma_0) = \left(\left(2\pi s^2\right)\Phi\left(\frac{\sigma_0}{s}\right)\right)^{-d}$. We assume that the pure-state distributions parameters are mutually independent. To ensure that this prior scales similarly to the other terms in (9), all of which scale with the length of the time series T, we set $\log\left(p(\boldsymbol{\Theta})\right) = T\sum_{k=1}^{K}\log\left(p(\boldsymbol{m}^k, \boldsymbol{S}^k)\right)$.

5 Model Estimation

Algorithm 1: Dynamical Wasserstein Time-Series Estimation

Input:
$\boldsymbol{y}_t, t = 1 \dots T$: Time series observations
K: Number of pure states

Hyper Parameters:
n: Window size
λ: regularization parameter
s: variance on prior for $\boldsymbol{\Theta}$
(μ_0, σ_0): $p(\boldsymbol{\Theta})$ reference distribution
η: Convergence threshold

Output:
$\boldsymbol{\Theta} = \left\{\{\boldsymbol{m}^k, \boldsymbol{S}^k\}_{k=1}^K\right\}$: pure-state distributions
$\boldsymbol{\Gamma} = \left\{\boldsymbol{x}_0, \{\boldsymbol{\gamma}_t\}_{t=1}^T\right\}$: Initial state and innovations
$\boldsymbol{X} = \{\boldsymbol{x}_t\}_{t=1}^T$: Wasserstein barycentric state vector (Computed from $\boldsymbol{\Gamma}$ via (7))
$\boldsymbol{H} = \{\boldsymbol{w}, \boldsymbol{a}_1, \boldsymbol{b}_1\}$: Beta mixture parameters for transition dynamics

1 **for** $t = (n+1), ..., (T-n)$ **do**
2 $\quad \boldsymbol{m}_t = \frac{1}{(2n+1)}\sum_{i=-n}^{n}\boldsymbol{y}_{t+i}$; `// Preprocessing of windowed`
3 $\quad \boldsymbol{S}_t = \frac{1}{2n}\sum_{i=-n}^{n}(\boldsymbol{y}_{t+i} - \boldsymbol{m}_t)(\boldsymbol{y}_{t+i} - \boldsymbol{m}_t)^T$; `// empirical distributions`
4 $\quad \hat{\rho}_t = \mathcal{N}(\boldsymbol{m}_t, \boldsymbol{S}_t)$
5 **end**
6 $c^{(0)} = F\left(\boldsymbol{\Gamma}^{(0)}, \boldsymbol{\Theta}^{(0)}, \boldsymbol{H}^{(0)}, \{\hat{\rho}_t\}_{t=1}^T\right)$; `//` $F\left(\boldsymbol{\Gamma}, \boldsymbol{\Theta}, \boldsymbol{H}\,\{\hat{\rho}_t\}_{t=1}^T\right)$ `from (9)`
7 **do**
8 $\quad \boldsymbol{\Gamma}^{(n+1)}, \boldsymbol{H}^{(n+1)} = \text{argmin}_{\boldsymbol{\Gamma}, \boldsymbol{H}}\, F\left(\boldsymbol{\Gamma}^{(n)}, \boldsymbol{\Theta}^{(n)}, \boldsymbol{H}^{(n)}, \{\hat{\rho}_t\}_{t=1}^T\right)$; `// ADAM`
9 $\quad \boldsymbol{\Theta}^{(n+1)} = \text{argmin}_{\boldsymbol{\Theta}}\, F\left(\boldsymbol{\Gamma}^{(n+1)}, \boldsymbol{\Theta}^{(n)}, \boldsymbol{H}^{(n+1)}, \{\hat{\rho}_t\}_{t=1}^T\right)$; `// Riemannian line search`
10 $\quad c^{(n+1)} = F\left(\boldsymbol{\Gamma}^{(n+1)}, \boldsymbol{\Theta}^{(n+1)}, \{\hat{\rho}_t\}_{t=1}^T\right)$
11 **while** $(c^{(n)} - c^{(n+1)}) > \eta$;

Alg. 1 details the steps needed to learn all parameters: $\boldsymbol{\Gamma}$, the initial state and innovations sequence that drive the dynamical state model; $\boldsymbol{\Theta}$, the pure-state distribution means and covariance matrices; and $\boldsymbol{H}$, the transition dynamics. The algorithm performs coordinate descent (updating some variables while fixing others) to optimize the objective function in (9). We chose this structure because the update to $\boldsymbol{\Theta}$ is able to exploit specialized optimization structure. Gradient descent methods are used to implement each minimization step in Alg. 1 taking advantage of auto-differentiation in PyTorch (Paszke et al., 2017). Optimization for each set of parameters is $\mathcal{O}(TKd^3)$.

The Adam optimizer (Kingma and Ba, 2017) is used to solve the $\boldsymbol{\Gamma}, \boldsymbol{H}$ problem on line 8 of Alg. 1. To ensure that $\boldsymbol{\gamma}_t \in [0, 1]$ for $t = 1, \dots, T$. we clamp these parameters to $[\epsilon, 1-\epsilon]$ for $\epsilon = 1e^{-6}$. The initial state vector is clamped and normalized to stay on the simplex in a similar manner and the parameters of $\boldsymbol{H}$ are clamped as mentioned in Sec. 4.1.

Determination of $\mathbf{\Theta}$ requires estimation of the mean and covariance parameters for K Gaussian distributions. While a variety of methods each based on different geometries have been proposed for for this task (Lin, 2019; Hosseini and Sra, 2015; Arsigny et al., 2007), in this work we choose to leverage the geometry of Gaussian distributions under the Wasserstein distance (Malagò et al., 2018).

Natural Riemannian geometry of $\mathbf{\Theta}$. It is evident from the decomposition in (2), that optimization for $\mathbf{\Theta} \equiv \{(\boldsymbol{m}^k, \boldsymbol{S}^k)\}_{k=1}^K$ under Wasserstein geometry can be carried out over a Riemannian product manifold $\left(\mathbb{R}^d \times Sym_+^d\right)$ (Hu et al., 2020) with standard Euclidean geometry on $\mathbb{R}^d$, and Wasserstein-Bures geometry on Sym_+^d (Malagò et al., 2018; Takatsu, 2011; Bhatia et al., 2017). Therefore, we estimate $\mathbf{\Theta}$ over this Riemannian product manifold using a gradient descent line search algorithm (Absil et al., 2008). The supplement provides these optimization parameters and experimental results demonstrating improved optimization speeds compared to standard Euclidean geometry.

6 Real World Results

6.1 Data Sets and Evaluation

Our work is motivated by applications in human activity accelerometry for learning the probability distributions associated with specific actions in a context where the smooth transitions between states are observable. We evaluate our algorithm considering the following datasets:

Beep Test (BT): (*Proprietary*) 46 subjects run between two points to a metronome with increasing frequency. In this setting the subject alternates between running and standing thus we estimate a two state model. Data is captured from a three-axis accelerometer sampled at 100 Hz.

Microsoft Research Human Activity (MSR): (Morris et al., 2014) 126 subjects perform exercises in a gym setting. Exercises vary among subjects covering strength, cardio, cross-fit, and static exercises. Each time series is truncated to five minutes. We set $K = 2, ..., 7$ to the number of labeled discrete states in the truncated time series. The three-axis accelerometer is sampled at 50 Hz.

Performance metric. Although these data sets do exhibit transitory behaviors between actions, labels only exist in the form of discrete states. This assumes instantaneous transitions between activities. Since the ground truth transition states are not available, we evaluate the performance based on how well a given model's predicted output fits the observed data over the whole time series. This *model fit error* as defined in (11) is taken from the model fit term in (9).

$$e_B = \frac{1}{T}\sum_{t=1}^{T} \mathcal{W}_2^2(\hat{\rho}_t, \rho_{B_t}). \quad (11)$$

Baseline method. Since this problem of modeling transitions among distributions is largely unexplored, we primarily compare our proposed Wasserstein barycentric model with linear interpolation model, where the state vector is viewed as a probability of assignment. Under this common model, each timestep's emission distribution is a Gaussian mixture, $\rho_{G_t} = \sum_k \boldsymbol{x}_t[k]\mathcal{N}(\boldsymbol{m}^k, \boldsymbol{S}^k)$. This is implemented in our framework simply by replacing ρ_{B_t} with ρ_{G_t} in Eqs. (9), (11). To compute the Wasserstein distance between a Gaussian and Gaussian mixture, we use the upper bound in (Chen et al., 2018) for parameter estimation, but use Monte-Carlo estimation (Sriperumbudur et al., 2010) for a more precise computation for the model fit error e_G in (11).

	BT	MSR
n	100	250
λ	100	100
s	1.0	1.0
η	1e-4	1e-4

Table 1: Model hyperparameters

Data preprocessing. For data with high sampling rate, modeling the system at each time point $t = 1, 2, ..., T$ is often unnecessary and results in large overlaps between sliding windows. Therefore, to reduce redundancy and simplify the computational burden, we update the model on a set interval, providing estimates at $t = 1, \delta, 2\delta, ..., T$ where $\delta = (n/4)$ for a window of length n.

Hyperparameters and initialization. Hyperparameter choices are documented in Tab. 1 and motivated by the included parameter selection study. We set $\boldsymbol{m}_0$ to the mean of the observed data, and σ_0 to the average eigenvalue of the set of covariance matrices obtained from a K-component GMM fit to the data using expectation-maximization via code from (Pedregosa et al., 2011). Our $\mathbf{\Theta}$ estimation problem is non-convex, so the solution is dependent on initialization. To ensure fair comparison, we use the same initialization for each method: a time-series clustering method (Cheng et al., 2020a) that applies matched-filtered change point detection (Cheng et al., 2020b) for the Wasserstein distance. Details regarding optimization settings and parameter initialization are in the supplement.

6.2 Experimental Results and Analysis

Qualitative evaluation. For a sample MSR time series, Fig. 4 shows the ability of our model to estimate pure-state distributions for the five true activities performed, and to capture both the stationary and transition periods well. Of interest are the segments of "Non-Exercise" that have significant contributions from "Pushup" (e.g. dashed lines at $t = 2.5k, 7.5k, 10.5k$). The data shows these regions are distinct from the pure-state "Non-Exercise" regions (e.g. solid lines at $t = 6.5k, 11.5k$). In these periods, the data mean and variance appear to be intermediate values between the "Pushup"and "Non-Exericise" pure states, suggesting the utility of our model. Detailed results from all other MSR time series are included in the supplemental material.

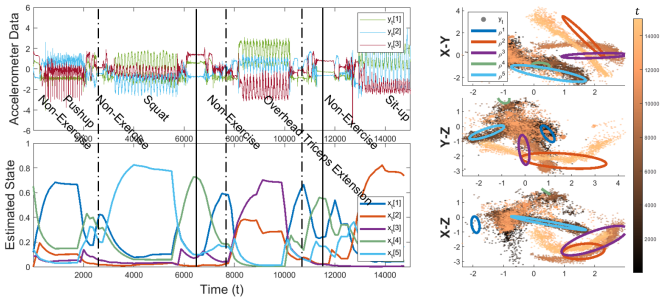

Figure 4: (top left) Three-axis accelerometer of a sample MSR time series consisting of 5 actions. (right) Estimated pure-state Gaussian distributions projected onto each pairwise axes. (bot left) The Wasserstein barycentric weights are given by the estimated system state.

We further visualize the learned state vectors for both our model and the baseline over time for the BT data in Fig. 5, revealing the benefits of our approach for transition modeling. This data has two pure states (stand, run), and transition periods can be clearly seen where the subject accelerates and decelerates between each running segment (e.g. Fig. 5(a) $t = 750, 1300$). The GMM in Fig. 5(b) identifies the alternating discrete states, however each of the transition regions appear identical, switching between states almost instantly. Only in the Wasserstein barycentric model in Fig. 5(d) can we differentiate the intermediate states that occur in the transition regions. This discrepancy between the models is highlighted in Fig. 5(c), where the difference between e_B and e_G is accentuated in transition regions.

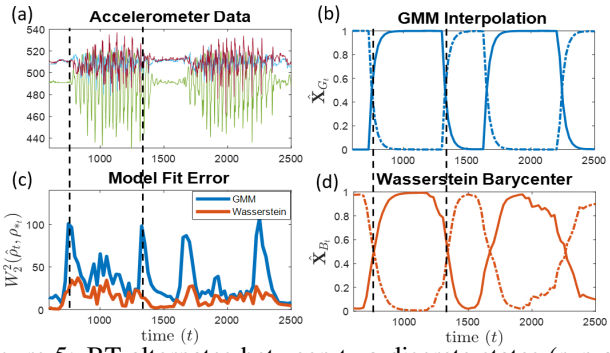

Figure 5: BT alternates between two discrete states (run and stand). (a) The transition between states are observable in the three-axis accelerometer. (b) The state model learned by the GMM linear model is akin to a binary switch in states whereas (c) the Wasserstein barycentric model closely tracks both gradual and quick state changes. (d) This difference is highlighted in the discrepancy in $\mathcal{W}_2^2(\hat{\rho}_t, \rho_{B_t})$ in the transition regions.

Quantitative error comparison. Fig. 6 shows how our Wasserstein barycentric interpolation model delivers improved fits to the data compared to the Gaussian mixture interpolation model. We see a drop in error ($e_B < e_G$) for each of the 126 MSR time series. Since the distributions ρ_{B_t} and ρ_{G_t} are equivalent when the system exists in a pure-state, the benefits of the barycentric model come from the closer fit during the transition periods given by the displacement-interpolation model.

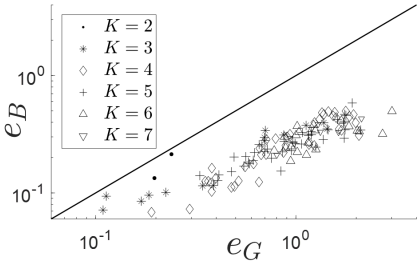

Figure 6: Comparison for each MSR time series between the GMM and Wasserstein barycentric interpolation model. Points below $y = x$ indicate a better data fit for the Wasserstein model.

Ablation study of the dynamics prior. We also consider a variation of our model using a (non-mixture) single Beta distribution prior for γ_t with parameters $\boldsymbol{a} = 1.1, \boldsymbol{b} = 3.0, \lambda = 10, s = 2.0$. Because this uses a single Beta for learning both stationary and transition dynamics, the model is more sluggish in adapting to fast changing states as seen in the supplement. Under this configuration, the MSR dataset has an average $e_B = 0.50$ compared to the average $e_B = 0.27$ shown in Fig. 6 for the Beta-mixture learnable prior.

Riemannian Optimization. The supplement provides experimental results demonstrating improved optimization speed for estimating $\mathbf{\Theta}$ using the Riemannian product manifold discussed in Sec. 5 compared to using the standard Euclidean geometry.

Parameter Selection. We explore the sensitivity of the results to variations in two of the key hyperparameters: the regularization parameter λ and the pure state variance, s whose inverse also plays the role of a regularization parameter in (9). From Tab. 2, increasing λ generally improves the model fit as is expected from (9). For $\lambda \geq 100$, there is an optimal value $s = 1.0$ which corresponds to "reasonable" pure state distribution seen in Fig. 7. For s too small, the distributions are constrained to the centroid of the data. For s too large the effects of outliers in the data yield pure state distributions that are disjoint from the data themselves, a result that is allowed from the simplicial structure of the model. In this regime the the barycentric state vector moves away from the vertices towards the interior of the simplex causing more perceived uncertainty between states.

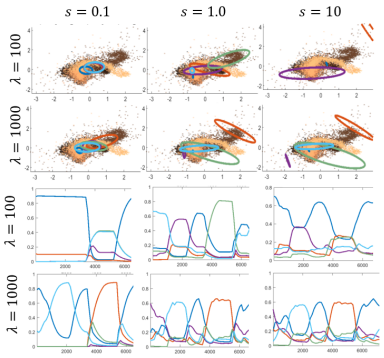

Figure 7: (a) Estimated pure state distributions and (b) estimated state for MSR data varying λ, s

$\lambda \backslash s$	0.01	0.1	1.0	10	100
0.1	596	601	581	395	434
1.0	597	593	455	389	378
10	572	533	284	203	214
100	521	362	**139**	157	173
1000	410	175	128	137	137

Table 2: Mean e_B for 25 MSR time series varying λ, s. Reported MSR and BT results use $\lambda = 100, s = 1.0$.

Sensitivity to initialization. The supplement includes plot showing similar results to Fig. 6 obtained when initializing $\mathbf{\Theta}$ using ground truth activity labels showing our choice of initialization is unbiased.

7 Conclusions and Future Work

Addressing recent trends in technology where the sampling rate of sensors can capture both stationary and transient behaviors of the system, we propose a novel method to learn both pure-state emission distributions and the time-varying state vector under a displacement-interpolation transition model between states in an unsupervised setting. For applications such as human activity recognition where transitions are often gradual, the displacement-interpolation given by the Wasserstein barycenter fits data more accurately than the mixture transition model commonly used in the literature. The proposed method can be applied to a wide range of time-series problems including segmentation, clustering, classification, and estimation.

As further contributions, we provide a dynamical state-evolution model of barycentric weight vectors over time. Inspired by previous work in state-space and Bayesian domains, this simplex random walk applies to other temporal modeling applications requiring simplex-valued representations. We also show how tailoring the optimization geometry to the problem leads to improved convergence speeds.

Limitations. Due to the need to estimate pure-state distributions in a time-sensitive manner, our method is primarily useful for applications where the data is low-dimensional and densely sampled. We have assumed knowledge of the true number of states; in practice at least an upper bound might be known but model size selection remains an open problem. Moreover, we have made a strong assumption that all distributions are Gaussian primarily to leverage the associated geometry and simplify the barycenter computation and optimization.

Future Work. More generally, because the Wasserstein barycenter in (3) is defined for any set of distributions with finite second moment (Peyré and Cuturi, 2019), the displacement-interpolation data model outlined in this work can be applied to any general set of such distributions. Constructing tractable algorithms based on non-parametric pure state models is certainly an interesting task. We also observe that the simplex random walk is amenable to natural extensions. For example, in our work, we assume that the transition parameters $\boldsymbol{\gamma}_t$ are IID over time. However, by coupling these parameters, we can obtain higher-order smoothness to the simplex-state vector.

Finally, the only barrier for making (9) a true likelihood is building a probabilistic model for the model fit based on the Wasserstein distance to the observed data. This has proven difficult due to the reasons discussed in (Panaretos and Zemel, 2019). However, with this modification, we can use posterior analysis to properly assess uncertainty in the model and aid in setting the model parameters including the total number of states K, which currently is assumed to be known a-priori.

References

P.-A. Absil, R. Mahony, and R. Sepulchre. 2008. *Optimization algorithms on matrix manifolds*. Princeton University Press, Princeton, N.J. ; Woodstock. OCLC: ocn174129993.

Saeed Aghabozorgi, Ali Seyed Shirkhorshidi, and Teh Ying Wah. 2015. Time-series clustering – A decade review. *Information Systems* 53 (Oct. 2015), 16–38. https://doi.org/10.1016/j.is.2015.04.007

Martial Agueh and Guillaume Carlier. 2011. Barycenters in the Wasserstein Space. *SIAM Journal on Mathematical Analysis* 43, 2 (Jan. 2011), 904–924. https://doi.org/10.1137/100805741

Vincent Arsigny, Pierre Fillard, Xavier Pennec, and Nicholas Ayache. 2007. Geometric Means in a Novel Vector Space Structure on Symmetric Positive-Definite Matrices. *SIAM J. Matrix Anal. Appl.* 29, 1 (Jan. 2007), 328–347. https://doi.org/10.1137/050637996

Rajendra Bhatia, Tanvi Jain, and Yongdo Lim. 2017. On the Bures-Wasserstein distance between positive definite matrices. *arXiv:1712.01504 [math]* (Dec. 2017). http://arxiv.org/abs/1712.01504 arXiv: 1712.01504.

H. Bi, M. Perello-Nieto, R. Santos-Rodriguez, and P. Flach. 2021. Human Activity Recognition Based on Dynamic Active Learning. *IEEE Journal of Biomedical and Health Informatics* 25, 4 (April 2021), 922–934. https://doi.org/10.1109/JBHI.2020.3013403 Conference Name: IEEE Journal of Biomedical and Health Informatics.

Yoosoon Chang, Robert K. Kaufmann, Chang Sik Kim, J. Isaac Miller, Joon Y. Park, and Sungkeun Park. 2020. Evaluating trends in time series of distributions: A spatial fingerprint of human effects on climate. *Journal of Econometrics* 214, 1 (Jan. 2020), 274–294. https://doi.org/10.1016/j.jeconom.2019.05.014

Yoosoon Chang, Chang Sik Kim, and Joon Y. Park. 2016. Nonstationarity in time series of state densities. *Journal of Econometrics* 192, 1 (May 2016), 152–167. https://doi.org/10.1016/j.jeconom.2015.06.025

Yongxin Chen, Tryphon T. Georgiou, and Allen Tannenbaum. 2018. Optimal transport for Gaussian mixture models. *arXiv:1710.07876 [cs, math]* (Jan. 2018). http://arxiv.org/abs/1710.07876 arXiv: 1710.07876.

Kevin C. Cheng, Shuchin Aeron, Michael C. Hughes, Erika Hussey, and Eric L. Miller. 2020a. Optimal Transport Based Change Point Detection and Time Series Segment Clustering. *arXiv:1911.01325 [cs, eess]* (Feb. 2020). http://arxiv.org/abs/1911.01325 arXiv: 1911.01325.

Kevin C. Cheng, Eric L. Miller, Michael C. Hughes, and Shuchin Aeron. 2020b. On Matched Filtering for Statistical Change Point Detection. *IEEE Open Journal of Signal Processing* 1 (2020), 159–176. https://doi.org/10.1109/OJSP.2020.3035070 Conference Name: IEEE Open Journal of Signal Processing.

Mark Gales and Steve Young. 2007. The Application of Hidden Markov Models in Speech Recognition. *Foundations and Trends® in Signal Processing* 1, 3 (2007), 195–304. https://doi.org/10.1561/2000000004

Reshad Hosseini and Suvrit Sra. 2015. Manifold Optimization for Gaussian Mixture Models. *arXiv:1506.07677 [cs, math, stat]* (June 2015). http://arxiv.org/abs/1506.07677 arXiv: 1506.07677.

Jiang Hu, Xin Liu, Zai-Wen Wen, and Ya-Xiang Yuan. 2020. A Brief Introduction to Manifold Optimization. *Journal of the Operations Research Society of China* 8, 2 (June 2020), 199–248. https://doi.org/10.1007/s40305-020-00295-9

R. E. Kalman. 1960. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering* 82, 1 (March 1960), 35–45. https://doi.org/10.1115/1.3662552

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs]* (Jan. 2017). http://arxiv.org/abs/1412.6980 arXiv: 1412.6980.

Zhenhua Lin. 2019. Riemannian Geometry of Symmetric Positive Definite Matrices via Cholesky Decomposition. *arXiv:1908.09326 [math, stat]* (Aug. 2019). https://doi.org/10.1137/18M1221084 arXiv: 1908.09326.

Luigi Malagò, Luigi Montrucchio, and Giovanni Pistone. 2018. Wasserstein Riemannian Geometry of Positive Definite Matrices. *arXiv:1801.09269 [math, stat]* (Sept. 2018). http://arxiv.org/abs/1801.09269 arXiv: 1801.09269.

Robert J. McCann. 1997. A Convexity Principle for Interacting Gases. *Advances in Mathematics* 128, 1 (June 1997), 153–179. https://doi.org/10.1006/aima.1997.1634

Dan Morris, T. Scott Saponas, Andrew Guillory, and Ilya Kelner. 2014. RecoFit: using a wearable sensor to find, recognize, and count repetitive exercises. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, Toronto Ontario Canada, 3225–3234. https://doi.org/10.1145/2556288.2557116

Goshu Nagino and Makoto Shozakai. 2006. Distance Measure Between Gaussian Distributions for Discriminating Speaking Styles. (2006), 4.

Tuan-Minh Nguyen and Stanislav Volkov. 2020. On a class of random walks in simplexes. *Journal of Applied Probability* 57, 2 (June 2020), 409–428. https://doi.org/10.1017/jpr.2020.19 arXiv: 1709.00174.

Victor M. Panaretos and Yoav Zemel. 2019. Statistical Aspects of Wasserstein Distances. *Annual Review of Statistics and Its Application* 6, 1 (2019), 405–431. https://doi.org/10.1146/annurev-statistics-030718-104938 _eprint: https://doi.org/10.1146/annurev-statistics-030718-104938.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. 2017. Automatic differentiation in PyTorch. (2017), 4.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12, 85 (2011), 2825–2830. http://jmlr.org/papers/v12/pedregosa11a.html

Gabriel Peyré and Marco Cuturi. 2019. Computational Optimal Transport: With Applications to Data Science. *Foundations and Trends® in Machine Learning* 11, 5-6 (Feb. 2019), 355–607. https://doi.org/10.1561/2200000073 Publisher: Now Publishers, Inc.

L. R. Rabiner. 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* 77, 2 (Feb. 1989), 257–286. https://doi.org/10.1109/5.18626 Conference Name: Proceedings of the IEEE.

Lu Ren, David B. Dunson, and Lawrence Carin. 2008. The dynamic hierarchical Dirichlet process. In *Proceedings of the 25th international conference on Machine learning - ICML '08*. ACM Press, Helsinki, Finland, 824–831. https://doi.org/10.1145/1390156.1390260

Walter Rudin. 1976. *Principles of mathematical analysis* (3d ed ed.). McGraw-Hill, New York.

Bharath K. Sriperumbudur, Arthur Gretton, Kenji Fukumizu, Bernhard Schölkopf, and Gert R. G. Lanckriet. 2010. Hilbert Space Embeddings and Metrics on Probability Measures. *Journal of Machine Learning Research* 11, 50 (2010), 1517–1561. http://jmlr.org/papers/v11/sriperumbudur10a.html

Asuka Takatsu. 2011. Wasserstein geometry of Gaussian measures. *Osaka Journal of Mathematics* 48, 4 (Dec. 2011), 1005–1026. https://doi.org/10.18910/4973 Publisher: Osaka University and Osaka City University, Departments of Mathematics.

Cédric Villani. 2009. *Optimal transport: old and new*. Number 338 in Grundlehren der mathematischen Wissenschaften. Springer, Berlin. OCLC: ocn244421231.

Greg Welch. 1997. An Introduction to the Kalman Filter. (1997), 16.

Roy D. Yates and David J. Goodman. 2005. *Probability and Stochastic Processes; a Friendly Introduction for Electrical and Computer Engineers* (2nd ed ed.). John Wiley & Sons, Hoboken, NJ.

Pedro C. Álvarez Esteban, E. del Barrio, J. A. Cuesta-Albertos, and C. Matrán. 2016. A fixed-point approach to barycenters in Wasserstein space. *arXiv:1511.05355 [math, stat]* (April 2016). http://arxiv.org/abs/1511.05355 arXiv: 1511.05355.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] **Sec. 7**
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] **Supplement**
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] **Code, instructions, and MSR data are included in the supplement**
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] **Sec: 4, 6.1 and supplement**
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] **Supplement**
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] **Sec. 6.1**
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] **Supplement**
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] **Supplement**
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]