LEARNING STOCHASTIC REPRESENTATIONS OF PHYSICAL SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning representations of physical systems is an important problem at the interface of statistical physics and machine learning. Recently, there has been a growing interest in devising methods to analyze high-dimensional simulation data generated by unbiased or biased samplers. As statistical physics systems consisting of $N \gg 1$ objects tend to have many degrees of freedom, dimensionality reduction methods are of particular interest. Here, we use a new method, multiscale reweighted stochastic embedding (MRSE), to analyze handwritten digits data sets and a biased trajectory of alanine tetrapeptide, and show that we can reconstruct low-dimensional representations of these data sets while retaining the most informative characteristics of their high-dimensional representation.

1 INTRODUCTION

Learning low-dimensional representations of physical systems is a fundamental task in statistical learning. Usually, such systems are high-dimensional as they consist of many objects, rendering too many degrees of freedom to sift through. Because of that, data sets generated by unbiased and biased simulations require dimensionality reduction. In the context of molecular simulations, a dimensionality reduction of a statistical system amounts to finding generalized degrees of freedom, the so-called collective variables (CVs). The CVs can be used for the analysis of the studied system, or for biasing the CVs by enhanced sampling. Enhanced sampling methods aim at exploring the long-timescale behavior of complex dynamical systems (Valsson et al., 2016). Some of widely used techniques for enhancing the sampling of the CVs are metadynamics (Laio & Parrinello, 2002; Barducci et al., 2008; Bussi & Laio, 2020), variationally enhanced sampling (Valsson et al., 2016), and umbrella sampling (Torrie & Valleau, 1977; Maragakis et al., 2009).

Methods at the interaction of machine learning (ML) and statistical physics can be used to embed the high-dimensional feature space (e.g., microscopic coordinates, distances, angles) into the low-dimensional latent space. Several such techniques have been introduced to alleviate the dimensionality problem in simulations of complex systems, for instance, autoencoders (Wehmeyer & Noé, 2018), stochastic kinetic embedding (Zhang & Chen, 2018) and sketch-map (Ceriotti et al., 2011). Recently, a new method has been proposed to that aim, multiscale reweighted stochastic embedding (MRSE) (Rydzewski & Valsson, 2021), on which we focus in this work.

In this paper, we demonstrate the ability of MRSE to learn the low-dimensional embeddings spanned in the latent space. We show that MRSE is applicable to data sets from molecular simulations and a handwritten digits data set considered often to test dimensionality reduction methods. More importantly, we show that MRSE can learn CVs of a statistical physics system, alanine tetrapeptide, characterized by a biased data set, and bias the CVs on the fly during an enhanced sampling simulation. This feature enables us to reconstruct the free energy surface of the physical system composed of many metastable states of different sizes.

2 GENERALIZED DEGREES OF FREEDOM: COLLECTIVE VARIABLES

We first introduce the concept of generalized degrees of freedom. In statistical physics, we consider a physical system described entirely by microscopic coordinates, $\mathbf{R} \in \mathbb{R}^{3N}$, where N is the number of atoms in the system, and a potential energy function $U(\mathbf{R})$. To sample $\mathbf{R}$ of such systems

molecular dynamics may be used. In the canonical ensemble (NVT), the microscopic coordinates at equilibrium follow the Boltzmann distribution:

$$P(\mathbf{R}) = \frac{\mathrm{e}^{-\beta U(\mathbf{R})}}{\int \mathrm{d}\mathbf{R}\, \mathrm{e}^{-\beta U(\mathbf{R})}}, \tag{1}$$

where the inverse temperature (the inverse of the product of the Boltzmann constant and temperature) is $\beta = (k_\mathrm{B}T)^{-1}$. Note that in the ML literature, β is often set to 1 so that the characteristic Boltzmann function is $\mathrm{e}^{-U(\mathbf{R})}$. Here, however, to be exact, we use the notation from the physics literature. In Equation 1, the denominator is the so-called partition function $Z = \int \mathrm{d}\mathbf{R}\, \mathrm{e}^{-\beta U(\mathbf{R})}$ which normalizes the solution.

To obtain a more useful representation that has a lower number of degrees of freedom, we transform the equilibrium distribution of the microscopic coordinates to the equilibrium marginal distribution of generalized degrees of freedom, i.e., CVs (Valsson et al., 2016), denoted below as $\mathbf{z} \in \mathbb{R}^d$, where d is the number of CVs. It is done by integrating out all other degrees of freedom:

$$P(\mathbf{z}) = \int \mathrm{d}\mathbf{R}\, \delta\big[\mathbf{z} - \mathbf{z}(\mathbf{R})\big] P(\mathbf{R}) = \Big\langle \delta\big[\mathbf{z} - \mathbf{z}(\mathbf{R})\big] \Big\rangle_U, \tag{2}$$

where $\delta[\cdot]$ is the Dirac delta function and $\langle\cdot\rangle_U$ is an ensemble average under the potential $U(\mathbf{R})$. Connected to calculating $P(\mathbf{z})$ is the inverse problem of reconstructing the free energy $F(\mathbf{z}) = -\beta^{-1} \log P(\mathbf{z})$ from $\mathbf{z}$. Another way to express the equilibrium probability distribution in the $\mathbf{z}$ space, is $P(\mathbf{z}) = \mathrm{e}^{-\beta F(\mathbf{z})} / \int \mathrm{d}\mathbf{z}\, \mathrm{e}^{-\beta F(\mathbf{z})}$, where $U(\mathbf{R})$ is replaced by $F(\mathbf{z})$ (see Equation 1).

One should note that the dependence of CVs on the microscopic degrees of freedom may either be explicit or implicit, and so it is often helpful to initially map the microscopic degrees of freedom to features (e.g., internal representation) and then to CVs (see Figure 1). In our formalism, features may be any function depending on $\mathbf{R}$. Therefore it is possible to base learning CVs on a set of manually selected features instead of $\mathbf{R}$. This kind of selection impacts finding CVs, however, it is easier first to select a general set of features than a two- or three-dimensional CVs (including an additional variable has a drastic impact on the convergence of an enhanced sampling simulation).

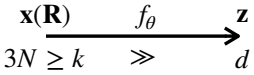

Figure 1: Embedding function f_θ, parametrized by θ is $\mathbf{z} \equiv f_\theta(\mathbf{x}(\mathbf{R}))$, i.e., a feature sample of dimension $k \leq 3N$, as $\mathbf{x}$ may be as large as $\mathbf{R}$, is reduced to a latent sample of dimension $d \ll k$. Here, the dimensionality reduction works by getting a selection of CVs that retains characteristics of the system.

When the free energy has many metastable states separated by high free energy barriers ($\gg k_\mathrm{B}T$), the system is kinetically stuck within a single state, rendering the simulation time to sample the CV space too high to be simulated directly (Valsson et al., 2016). In optimization theory, this means that a local minimum cannot be reached in a finite number of steps. We can use enhanced sampling methods that apply an additional bias potential V to the dynamics to alleviate this sampling problem. Heavily inspired by importance sampling, this concept was first coined by Torrie & Valleau (1977), and it is based on introducing V into a simulation, regardless of its construction, which leads to the following biased distribution at convergence:

$$P_V(\mathbf{z}) = \Big\langle \delta\big[\mathbf{z} - \mathbf{z}(\mathbf{R})\big] \Big\rangle_{U+V}, \tag{3}$$

where the system evolves now at the total potential $U + V$. Due to the non-physical characteristics of V, each $\mathbf{R}$ carries a statistical weight defined as $w(\mathbf{z}) \equiv P(\mathbf{z})/P_V(\mathbf{z})$ and has to be accounted for when reweighting ("unbiasing") the biased distribution.

The functional form of the weights depends on the type of V used in the simulation. For instance, when the bias is static, $w(\mathbf{z}) = \mathrm{e}^{\beta V(\mathbf{z})}$, otherwise the bias has to be modified by an additional time-dependent constant. This is the case in well-tempered metadynamics (Laio & Parrinello, 2002; Barducci et al., 2008) where the time-dependent bias potential is added to a simulation by periodically depositing Gaussian kernels at the current location in the CV space. As the Gaussian height

decreases over time and in the long-time limit approaches zero, the bias potential converges to the free energy:

$$V(\mathbf{z}, t \to \infty) = -\left(1 - \frac{1}{\gamma}\right) F(\mathbf{z}), \tag{4}$$

where $\gamma > 1$ is a bias factor. The time-dependence of the bias introduces a modification to the statistical weights: $w(\mathbf{z}, t) = \exp[\beta(V(\mathbf{z}, t) - c(t))]$, where $c(t)$ is a time-dependent constant defined as:

$$c(t) = \frac{1}{\beta} \log \frac{\int \mathrm{d}\mathbf{z} \exp\left[\frac{\gamma}{\gamma-1}\beta V(\mathbf{z}, t)\right]}{\int \mathrm{d}\mathbf{z} \exp\left[\frac{1}{\gamma-1}\beta V(\mathbf{z}, t)\right]}. \tag{5}$$

The statistical weights defined in this way will be important for learning a low-dimensional representation of a physical system. For more details regarding reweighting methods, see (Tiwary & Parrinello, 2015; Valsson et al., 2016).

Next, we describe how to construct the low-dimensional CVs based on a data set of collected feature samples.

3 Learning Latent Variables

3.1 Feature Representation

We initially have a set $\{\mathbf{x}_k\}_{k=1}^K$ of feature samples. To each pair of feature samples $\mathbf{x}_i$ and $\mathbf{x}_j$, we assign a pairwise value modeled using the Gaussian kernel $G_\varepsilon(\mathbf{x}_i, \mathbf{x}_j) \in \mathbb{R}^{K \times K}$, characterized by a scale parameter ε:

$$G_\varepsilon(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\varepsilon \|\mathbf{x}_i - \mathbf{x}_j\|^2) \tag{6}$$

The distance $\|\cdot\|$ between feature samples is measured using the Euclidean distance. Next, using Eq. 6, we build a Gaussian mixture as a sum over different values of scale parameters $\boldsymbol{\varepsilon} = \{\varepsilon\}$ that are unique for the i-th sample:

$$G(\mathbf{x}_i, \mathbf{x}_j) = \sum_i G_{\varepsilon_i}(\mathbf{x}_i, \mathbf{x}_j), \tag{7}$$

where each ε_i is estimated by fitting $G_{\varepsilon_i}(\mathbf{x}_i, \mathbf{x}_j)$ to the data so that the Shannon entropy of the Gaussian kernel is approximately $\log_2 p$. Here, p is a parameter called perplexity, defined as an exponential of the Shannon entropy. We can view p as the effective number of neighbors in a manifold. For details regarding the estimation of ε_i, we refer to Rydzewski & Valsson (2021).

Then we can define an intermediate feature pairwise probability as:

$$g_{ij} = \frac{G(\mathbf{x}_i, \mathbf{x}_j)}{\sum_m G(\mathbf{x}_i, \mathbf{x}_m)}, \tag{8}$$

where $1 \leq i, j \leq K$ is the number of feature samples and the diagonal elements are zeros.

Inspired by diffusion maps (Coifman et al., 2005; Coifman & Lafon, 2006; Coifman et al., 2008), we want the representation to account for the data density. To this aim, we introduce to the feature representation a density normalization factor, $g(\mathbf{x}_i) = \sum_j G(\mathbf{x}_i, \mathbf{x}_j)$. Next, we construct a Laplacian kernel $L(\mathbf{x}_i, \mathbf{x}_j)$ as:

$$L(\mathbf{x}_i, \mathbf{x}_j) = \frac{G(\mathbf{x}_i, \mathbf{x}_j)}{[g(\mathbf{x}_i)]^\alpha [g(\mathbf{x}_j)]^\alpha}, \tag{9}$$

where α is a parameter dependent on the studied system. For $\alpha = \frac{1}{2}$, the density normalization corresponds to the Markov chain that is an approximation of the diffusion of a Fokker-Planck equation with the density $\propto \mathrm{e}^{-\beta U}$, allowing to approximate the long-time behavior of a system described by a certain stochastic differential equation. Other values of α are also possible, e.g., for $\alpha = 0$, we get the classical normalized graph Laplacian; for $\alpha = 1$, we ignore the underlying probability density (Coifman et al., 2005).

Using Eq. 9, we rewrite the feature pairwise probability distribution in terms of L:

$$\mathbf{L} = \left(l_{ij} \right) \quad \text{where} \quad l_{ij} = \frac{L(\mathbf{x}_i, \mathbf{x}_j)}{\sum_m L(\mathbf{x}_i, \mathbf{x}_m)}, \tag{10}$$

where $\mathbf{L}$ is the Markov probability matrix that we base the feature representation on.

3.2 LATENT REPRESENTATION

To represent distances between low-dimensional latent variables $\mathbf{z}_i$ and $\mathbf{z}_j$, we use a parametric version of t-distribution kernel as in McInnes et al. (2018):

$$Q(\mathbf{z}_i, \mathbf{z}_j) = \left(1 + a\|\mathbf{z}_i - \mathbf{z}_j\|^{2b}\right)^{-1}, \tag{11}$$

where a and b are parameters of the kernel. Then, the latent pairwise probability distribution is defined:

$$\mathbf{Q} = \left(q_{ij} \right) \quad \text{where} \quad q_{ij} = \frac{Q(\mathbf{z}_i, \mathbf{z}_j)}{\sum_m Q(\mathbf{z}_i, \mathbf{z}_m)}, \tag{12}$$

similarily to Eq. 10.

As shown in van der Maaten & Hinton (2008); van der Maaten (2009), a heavy-tailed t-distribution used in the latent space overcomes the so-called "crowding problem" by grouping close data samples and separating data samples that are far away from each other.

The latent variables are obtained via the embedding function, i.e., $\mathbf{z}_i = f(\mathbf{x}_i)$ for i denoting feature samples in the training datasets.

3.3 EMBEDDING FUNCTION

Embedding function f_θ, parametrized with θ, is defined as $\mathbf{z} = f_\theta(\mathbf{x}(\mathbf{R}))$, i.e., a feature sample of dimension $k \leq 3N$, as $\mathbf{x}$ may be as large as $\mathbf{R}$, is reduced to a latent sample of dimension $d \ll k$ (Figure 1). As a parametric function, we use a deep neural network.

To find optimal θ for the embedding function, we use the following loss: $D = D_{\text{CE}} + D_{\text{W}}$, where D_{CE} is the cross-entropy between $\mathbf{L}$ and $\mathbf{Q}$ and D_{W} is the Brownian correlation between $\mathbf{x}$ and $\mathbf{z}$. Below we explain the loss in detail and provide an interpretation of each term.

The cross-entropy between $\mathbf{L}$ and $\mathbf{Q}$ is:

$$D_{\text{CE}} = \sum_{i \neq j} l_{ij} \log\left(\frac{l_{ij}}{q_{ij}}\right) + \sum_{i \neq j} (1 - l_{ij}) \log\left(\frac{1 - l_{ij}}{1 - q_{ij}}\right), \tag{13}$$

where $D_{\text{CE}} \geq 0$ and equal to zero if $\mathbf{L} = \mathbf{Q}$. Minimizing the cross-entropy enforces that the feature pairwise probability distribution matches the latent pairwise distribution probability. Therefore, the low-dimensional latent space has to have a probability between latent samples similar to their high-dimensional counterpart.

The Brownian correlation between $\mathbf{x}$ and $\mathbf{z}$, D_{W} is a coefficient suitable for finding a correlation between stochastic processes and capable of indicating both linear and nonlinear correlation between two random variables. D_{W} is defined as (Székely & Rizzo, 2009):

$$D_{\text{W}} = -\frac{\text{cov}_{\text{W}}(\mathbf{x}, \mathbf{z})}{\sigma(\mathbf{x})\,\sigma(\mathbf{z})}, \tag{14}$$

where $\text{cov}_{\text{W}}(\mathbf{x}, \mathbf{z})$ denotes the Brownian covariance of $\mathbf{x}$ and $\mathbf{z}$ such that $0 \leq D_{\text{W}} \leq 1$, and $\sigma(\cdot)$ is the standard deviation of $\mathbf{x}$ and $\mathbf{z}$, respectively. The minimization of D_{W} increases the correlation between the pairwise distances in the feature space and the latent space. The interpretation is that such procedure increases the density conservation in the latent space. For details on how to calculate $\text{cov}_{\text{W}}(\mathbf{x}, \mathbf{z})$, see Appendix A.

4 RESULTS

4.1 UCI HANDWRITTEN DIGITS DATA SET

Before we show the results for a physical system, we apply MRSE to the UCI handwritten digits data set (Dua & Graff, 2017) to show that the method is not only applicable to simulation data. The data set consists of 1797 digit images, each of size 8×8 pixels. Each class (from 0 to 9) has around 180 samples. The data set is downloaded using the `scikit-learn` library (Pedregosa et al., 2011).

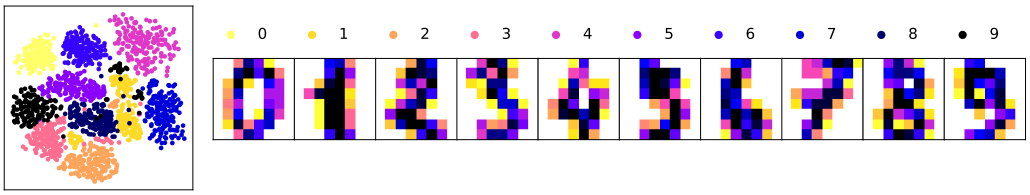

Figure 2: Low-dimensional embedding calculated for the UCI handwritten digits data set (https://archive.ics.uci.edu/ml/machine-learning-databases/optdigits/). The data set consists of 1797 digit images, each of size 8×8 pixels.

The embedding function for this data set is modeled as a neural network of size [64, 5000, 2] with the ReLU activation functions and the linear function for the output and is trained through 100 epochs using the Adam optimizer with default parameters. To construct the feature pairwise probability distribution, we use $\alpha = 0$ (Eq. 9) (e.g., a normalized graph Laplacian) and a Gaussian mixture over the following perplexities: 256, 32, 4. For the latent pairwise probability distribution, we use $a = 1.93$ and $b = 1.58$ as parameters in Eq. 11 which are found by fitting to the latent pairwise distances using the limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm in the initial steps of the training of the embedding function.

Overall, we find that MRSE correctly embeds the digits data set into a low-dimensional representation. Apart from few samples that are placed far from their main clusters, which may be due to the low resolution of the digits (compare "3" and "5" in Figure 2), the majority of samples is grouped well (Figure 2).

4.2 Physical System Data Set

As a physical system in this study, we take an alanine tetrapeptide (Ace-Ala_3-Nme) data set from Rydzewski & Valsson (2021). This data set is generated using GROMACS 2019.2 patched with PLUMED employing well-tempeted metadynamics to sample 6 dihedral angles and enhance the fluctuations of the Φ dihedral angles (see Figure 3). For additional information about the simulation setup, see Appendix B. The data set consists of 10^5 feature samples of sines and cosines of the dihedral angles (2×6 features).

A neural network of size [12, 64, 2] is used to represent the embedding function. We use ReLU activation functions apart from the output of the network that has a linear unit. The network is trained using the Adam optimizer with default parameters during 100 epochs. For the feature pairwise probability distribution, we use $\alpha = \frac{1}{2}$ (Eq. 9), to account for the long-time dynamics of our system. The Gaussian mixture is constructed by averaging over the following perplexities: 256, 128, 64, 32, 16, 8, 4. For the latent space, we use $a = 1$ and $b = 1$ in Eq. 11.

As the feature samples resulting from the well-tempered metadynamics simulation are biased, we need to reweight the feature pairwise probability distribution. Since each sample has a statistical weight, we use them to redefine Eq. 8 (Rydzewski & Valsson, 2021):

$$G(\mathbf{x}_i, \mathbf{x}_j) \equiv \sqrt{w(\mathbf{x}_i)\, w(\mathbf{x}_j)} \cdot G(\mathbf{x}_i, \mathbf{x}_j) \tag{15}$$

where the weights correspond to the i-th and j-th samples, respectively. Apart from this change, the rest of the protocol is without changes.

After the alanine tetrapeptide is trained, we run another well-tempered metadynamics simulation using the embedding to sample CVs on the fly during 100 ns. Figure 3 shows the obtained results. We can see that the trained embedding can be used to sample additional conformations which aggregate to the free energy surface shown in Figure 3. Several important metastable states characterizes the free energy surface, similarily to restults obtained recently in the literature (Giberti et al., 2019; Rydzewski & Valsson, 2021). Therefore, the embedding can be used to analyze the metastable states visible in the latent space without looking at all combinations of the dihedral angles. Moreover, the biasing of the learned low-dimensional embedding during a biased simulation can be used to

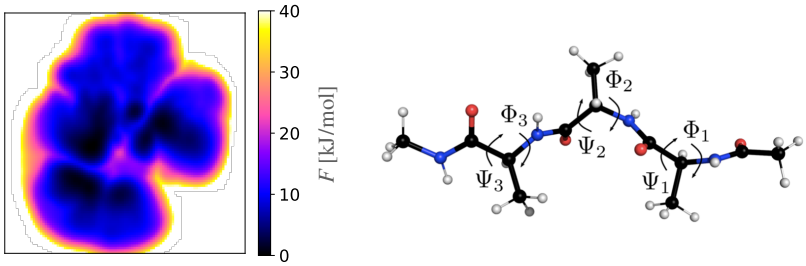

Figure 3: Low-dimensional embedding of the alanine tetrapeptide data set consisting of 10^5 feature samples where each sample is of 12 dimensions. The sines and cosines of the Φ and Ψ dihedral angles are used to describe each sample. The data set comes from a biased simulation run that enhances the fluctuations of the Φ dihedral angles using well-tempered metadynamics. The learned CVs are subsequently used to enhance sampling during a 100-ns well-tempered metadynamics simulation. This setup enables us to see several metastable states in the free energy surface embedded in the latent space. As we bias only two CVs, the simulation takes less computational time than sampling the Φ dihedral angles.

reconstruct the free energy surface, which takes less computational time than biasing the Φ dihedral angles to achieve an ergodic behavior.

5 Conclusions

In this short work, we show new results obtained by MRSE, which is used to learn the low-dimensional embeddings of the UCI handwritten digits and alanine tetrapeptide data sets. Overall, this work provides a generalization of MRSE to learn from synthetic ML data sets. Additionally, we show the ability of the MRSE embeddings to be used during a biased simulation which is important to calculate a free energy surface, arguably, one of the most informative characteristics of statistical physics systems. The theory and experiments behind MRSE are a proof-of-principle. If they can be extented to aid sampling of low-dimensional CVs in systems like proteins, it would be of great benefit. This aspect will be explored in future work.

Ethics Statement

The authors adhere to the ICLR Code of Ethics (`https://iclr.cc/public/CodeOfEthics`).

Reproducibility Statement

MRSE is implemented in an additional module called `LowLearner` (Rydzewski & Valsson, 2021) in a development version (2.7.0-dev) of the open-source PLUMED library (Tribello et al., 2014; The PLUMED Consortium, 2019) and it uses the LibTorch library (PyTorch C++ API) (Paszke et al., 2019). The initial version of the implementation along the alanine tetrapeptide data set are available at Zenodo (DOI: `https://doi.org/10.5281/zenodo.4756093`).

Acknowledgments

To be added after peer review.

References

A. Barducci, G. Bussi, and M. Parrinello. Well-tempered metadynamics: A smoothly converging and tunable free-energy method. *Phys. Rev. Lett.*, 100(2):020603, 2008. doi: Well-temperedmetadynamics:Asmoothlyconvergingandtunablefree-energymethod.

G. Bussi and A. Laio. Using metadynamics to explore complex free-energy landscapes. *Nat. Rev. Phys.*, pp. 1, 2020. doi: https://doi.org/10.1038/s42254-020-0153-0.

G. Bussi and M. Parrinello. Accurate Sampling using Langevin Dynamics. *Phys. Rev. E*, 75(5): 056707, 2007. doi: 10.1103/PhysRevE.75.056707.

M. Ceriotti, G. A. Tribello, and M. Parrinello. Simplifying the representation of complex free-energy landscapes using sketch-map. *Proc. Natl. Acad. Sci. U.S.A.*, 108(32):13023–13028, 2011. doi: https://doi.org/10.1073/pnas.1108486108.

R. R. Coifman and S. Lafon. Diffusion Maps. *Appl. Comput. Harmon. Anal.*, 21(1):5–30, 2006. doi: https://doi.org/10.1016/j.acha.2006.04.006.

R. R. Coifman, S. Lafon, A. B. Lee, M. Maggioni, B. Nadler, F. Warner, and S. W. Zucker. Geometric Diffusions as a Tool for Harmonic Analysis and Structure Definition of Data: Diffusion Maps. *Proc. Natl. Acad. Sci. U.S.A.*, 102(21):7426–7431, 2005. doi: https://doi.org/10.1073/pnas.0500334102.

R. R. Coifman, I. G. Kevrekidis, S. Lafon, M. Maggioni, and B. Nadler. Diffusion maps, reduction coordinates, and low dimensional representation of stochastic systems. *Multiscale Model. Simul.*, 7(2):842–864, 2008. doi: https://doi.org/10.1137/070696325.

Dheeru Dua and Casey Graff. UCI Machine Learning Repository, 2017. URL https://archive.ics.uci.edu/ml/machine-learning-databases/optdigits/.

F. Giberti, B. Cheng, G. A. Tribello, and M. Ceriotti. Iterative unbiasing of quasi-equilibrium sampling. *J. Chem. Theory Comput.*, 16(1):100–107, Nov 2019. ISSN 1549-9626. doi: 10.1021/acs.jctc.9b00907. URL http://dx.doi.org/10.1021/acs.jctc.9b00907.

B. Hess. P-LINCS: A Parallel Linear Constraint Solver for Molecular Simulation. *J. Chem. Theory Comput.*, 4(1):116–122, 2008. doi: https://doi.org/10.1021/ct700200b.

V. Hornak, R. Abel, A. Okur, B. Strockbine, A. Roitberg, and C. Simmerling. Comparison of multiple Amber force fields and development of improved protein backbone parameters. *Proteins*, 65(3):712–725, Nov 2006. ISSN 1097-0134. doi: 10.1002/prot.21123. URL http://dx.doi.org/10.1002/prot.21123.

A. Laio and M. Parrinello. Escaping free-energy minima. *Proc. Natl. Acad. Sci. U.S.A.*, 99(20): 12562–12566, 2002. doi: https://doi.org/10.1073/pnas.202427399.

Paul Maragakis, Arjan van der Vaart, and Martin Karplus. Gaussian-mixture umbrella sampling. *J. Phys. Chem. B*, 113(14):4664–4673, Apr 2009. ISSN 1520-5207. doi: 10.1021/jp808381s. URL http://dx.doi.org/10.1021/jp808381s.

L. McInnes, J. Healy, and J. Melville. UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction. *arXiv preprint arXiv:1802.03426*, 2018. URL https://arxiv.org/abs/1802.03426.

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. *NeurIPS*, 33:8024–8035, 2019. URL url={http://papers.nips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf}.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. scikit-learn: Machine learning in Python. *J. Mach. Lear. Res.*, 12(85):2825–2830, 2011. URL http://jmlr.org/papers/v12/pedregosa11a.html.

J. Rydzewski and O. Valsson. Multiscale Reweighted Stochastic Embedding: Deep Learning of Collective Variables for Enhanced Sampling. *J. Phys. Chem. A*, 125(28):6286–6302, 2021.

G. J Székely and M. L. Rizzo. Brownian Distance Covariance. *Ann. Appl. Stat.*, 3(4):1236–1265, 2009.

The PLUMED Consortium. Promoting transparency and reproducibility in enhanced molecular simulations. *Nat. Methods*, 16:670–673, 2019. doi: https://doi.org/10.1038/s41592-019-0506-8. For the full list of researches from the PLUMED Consortium, see https://www.plumed-nest.org/consortium.html.

P. Tiwary and M. Parrinello. A time-independent free energy estimator for metadynamics. *J. Phys. Chem. B*, 119(3):736–742, 2015. doi: 10.1021/jp504920s. URL http://dx.doi.org/10.1021/jp504920s.

G. M. Torrie and J. P. Valleau. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *J. Comp. Phys.*, 23(2):187–199, 1977. doi: https://doi.org/10.1016/0021-9991(77)90121-8.

G. A. Tribello, M. Bonomi, D. Branduardi, C. Camilloni, and G. Bussi. PLUMED 2: New feathers for an old bird. *Comp. Phys. Commun.*, 185(2):604–613, 2014. doi: https://doi.org/10.1016/j.cpc.2013.09.018.

O. Valsson, P. Tiwary, and M. Parrinello. Enhancing important fluctuations: Rare events and metadynamics from a conceptual viewpoint. *Ann. Rev. Phys. Chem.*, 67(1):159–184, 2016. doi: 10.1146/annurev-physchem-040215-112229.

L. van der Maaten. Learning a parametric embedding by preserving local structure. *J. Mach. Learn. Res.*, 5:384–391, 2009. URL http://proceedings.mlr.press/v5/maaten09a.html.

L. van der Maaten and G. Hinton. Visualizing data using t-SNE. *J. Mach. Learn. Res.*, 9:2579–2605, 2008. URL http://www.jmlr.org/papers/v9/vandermaaten08a.html.

C. Wehmeyer and F. Noé. Time-lagged autoencoders: Deep learning of slow collective variables for molecular kinetics. *J. Chem. Phys.*, 148(24):241703, 2018. doi: https://doi.org/10.1063/1.5011399.

J. Zhang and M. Chen. Unfolding hidden barriers by active enhanced sampling. *Phys. Rev. Lett.*, 121(1):010601, 2018. doi: 10.1103/PhysRevLett.121.010601.

A Diffusion Covariance

We start with the definition of the diffusion covariance. Let $\{\mathbf{x}_k\}_{k=0}^K$ and $\{\mathbf{z}_k\}_{k=0}^K$ be samples in the feature and latent spaces, respectively. We fist compute matrices containing pairwise Euclidean distances in both spaces:

$$\mathbf{x}_{ij} \equiv \|\mathbf{x}_i - \mathbf{x}_j\| \tag{16}$$

and

$$\mathbf{z}_{ij} \equiv \|\mathbf{z}_i - \mathbf{z}_j\| \tag{17}$$

for $i, k = 1, \ldots, K$. Next, we perform double centering by:

$$\mathbf{x}^c_{ij} = \mathbf{x}_{ij} - \bar{\mathbf{x}}_{i\cdot} - \bar{\mathbf{x}}_{\cdot j} + \bar{\mathbf{x}}_{\cdot\cdot}, \tag{18}$$

and

$$\mathbf{z}^c_{ij} = \mathbf{z}_{ij} - \bar{\mathbf{z}}_{i\cdot} - \bar{\mathbf{z}}_{\cdot j} + \bar{\mathbf{z}}_{\cdot\cdot}, \tag{19}$$

where $\bar{\mathbf{x}}_{i\cdot}$ ($\bar{\mathbf{z}}_{i\cdot}$) is the i-th row mean, $\bar{\mathbf{x}}_{\cdot j}$ ($\bar{\mathbf{z}}_{\cdot j}$) is the j-th column mean, and $\bar{\mathbf{x}}_{\cdot\cdot}$ ($\bar{\mathbf{z}}_{\cdot\cdot}$) is the grand mean of pairwise distances. Then, the diffusion covariance of $\mathbf{x}$ and $\mathbf{z}$ is defined as (Székely & Rizzo, 2009):

$$\mathrm{cov}_{\mathrm{W}}(\mathbf{x}, \mathbf{z}) = \frac{1}{K}\sqrt{\sum_{i=1}^{K}\sum_{j=1}^{K}\mathbf{x}^c_{ij}\mathbf{z}^c_{ij}}. \tag{20}$$

B SIMULATIONS OF ALANINE TETRAPEPTIDE

To model the dynamics of alanine tetrapeptide in vacuum, we use the Amber99-SB force field (Hornak et al., 2006). We perform the simulations in the canonical ensemble using a time step of 2 fs, the stochastic velocity rescaling thermostat (Bussi & Parrinello, 2007) with temperature 300 K and a relaxation time of 0.1 fs, and LINCS (Hess, 2008) to constrain hydrogen bonds. The simulations are done without periodic boundary conditions and cut-offs for electrostatic and non-bonded van der Waals interactions.

The data set for training is obtained by biasing the Φ backbone dihedral angles and a bias factor of 5 using well-tempered metadynamics (Barducci et al., 2008). The simulation biasing the low-dimensional embedding found by MRSE is also performed with a bias factor of 5. We use an initial Gaussian height of 1.2 kJ/mol and deposit Gaussians every 1 ps in both simulations. We calculate $c(t)$ every 50 ps.