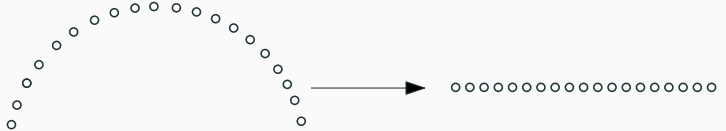
Dimension reduction and manifold learning

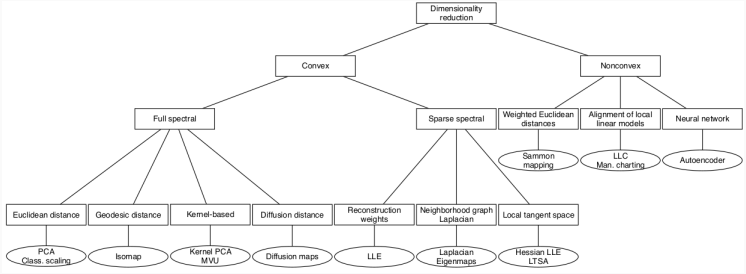
A non-exhaustive tour into nonlinear dimension reduction

Eddie Aamari
Département de mathématiques et applications
CNRS, ENS PSL

Master MASH — Dauphine PSL

Dimensionality Reduction Overview





More Graph and Spectral Methods

Laplacian Eigenmaps

Laplacian Eigenmaps originates from Belkin and Niyogi 2003.

Initially, the method was intended to remedy some shortcomings of other spectral methods like Isomap and LLE.

Related methods in MDS

LE is reminiscent of the spectral graph drawing of (Hall 1970).
See also Klimenta 2012, Section 3.4.3.

Laplacian Eigenmaps

Step 1: Graph. Build neighbors $\mathcal{N}_i$ of each point.
Given scale parameter $t > 0$, set $K \in \mathbb{R}^{n \times n}$ with

$$k_{i,j} = e^{-\frac{\|x_j - x_i\|^2}{t}} \mathbb{I}\{j \in \mathcal{N}_i\} \text{ for all } i, j \in \{1, \ldots, n\} .$$

Laplacian Eigenmaps

Step 1: Graph. Build neighbors $\mathcal{N}_i$ of each point.
Given scale parameter $t > 0$, set $K \in \mathbb{R}^{n \times n}$ with

$$k_{i,j} = e^{-\frac{\|x_j - x_i\|^2}{t}} \mathbb{I}\{j \in \mathcal{N}_i\} \text{ for all } i, j \in \{1, \ldots, n\} .$$

Step 2: Laplacian. Compute the Laplacian matrix

$$D := \operatorname{diag}(K 1_n), \qquad L := D - K.$$

Laplacian Eigenmaps

Step 1: Graph. Build neighbors $\mathcal{N}_i$ of each point.
Given scale parameter $t > 0$, set $K \in \mathbb{R}^{n \times n}$ with

$$k_{i,j} = e^{-\frac{\|x_j - x_i\|^2}{t}} \mathbb{I}\{j \in \mathcal{N}_i\} \text{ for all } i, j \in \{1, \dots, n\} .$$

Step 2: Laplacian. Compute the Laplacian matrix

$$D := \operatorname{diag}(K 1_n), \qquad L := D - K.$$

Step 3: Embedding. Compute the $d + 1$ *smallest* generalized eigenvectors $v_0, \dots, v_d \in \mathbb{R}^n$ associated to (L, D):

$$L v_k = \lambda_k D v_k$$

and set $Y_{\mathrm{LE}} := (\lambda_1 v_1 \mid \cdots \mid \lambda_d v_d) \in \mathbb{R}^{n \times d}$.

Insights behind Laplacian Eigenmaps

For all candidate embedding $Y = (y_1 \mid \cdots \mid y_n)^\top \in \mathbb{R}^{n \times d}$, it turns out as soon as K is symmetric,

$$\frac{1}{2} \sum_{i,j=1}^{n} k_{i,j} \|y_i - y_j\|^2 = \mathrm{Tr}(Y^\top L Y),$$

where $L = D - K$ as before.

Insights behind Laplacian Eigenmaps

For all candidate embedding $Y = (y_1 \mid \cdots \mid y_n)^\top \in \mathbb{R}^{n\times d}$, it turns out as soon as K is symmetric,

$$\frac{1}{2}\sum_{i,j=1}^{n} k_{i,j}\|y_i - y_j\|^2 = \mathrm{Tr}(Y^\top L Y),$$

where $L = D - K$ as before. Indeed,

$$\begin{aligned}
\frac{1}{2}\sum_{i,j} k_{i,j}\|y_i - y_j\|^2 &= \frac{1}{2}\sum_{i,j=1}^{n}(k_{i,j}y_i y_i^\top + y_j y_j^\top - 2y_i y_j^\top) \\
&= \frac{1}{2}\sum_{i} D_{i,i}y_i y_i^\top + \frac{1}{2}\sum_{j} D_{j,j}y_j y_j^\top - \sum_{i,j} k_{i,j}y_i y_j^\top \\
&= \mathrm{Tr}(Y^\top(D-K)Y).
\end{aligned}$$

Constraints

Without a constraint, $\arg\min_{Y \in \mathbb{R}^{n \times d}} \operatorname{Tr}(Y^\top L Y) = 0$.

Constraints

Without a constraint, $\arg\min_{Y \in \mathbb{R}^{n \times d}} \operatorname{Tr}(Y^\top L Y) = 0$.

Similar to LLE, put the covariance constraint $Y^T D Y = I_{d \times d}$.

(D weights the vertices: the larger $D_{i,i}$, the more "important" is that vertex.)

Vector $y = 1 \in \mathbb{R}^n$ is a trivial eigenvector of L.
Eliminate it with an orthogonality constraint $Y^\top D 1 = 0$.

Constraints

Without a constraint, $\arg\min_{Y \in \mathbb{R}^{n \times d}} \operatorname{Tr}(Y^\top L Y) = 0$.

Similar to LLE, put the covariance constraint $Y^T D Y = I_{d \times d}$.

(D weights the vertices: the larger $D_{i,i}$, the more "important" is that vertex.)

Vector $y = 1 \in \mathbb{R}^n$ is a trivial eigenvector of L.
Eliminate it with an orthogonality constraint $Y^\top D 1 = 0$.

$$E_{LE} = \underset{\substack{Y^\top D Y = I_{d \times d} \\ Y^\top D 1 = 0}}{\arg\min} \operatorname{Tr}(Y^\top L Y).$$

This is a generalized eigenvalue problem with solution

$$Y_{LE} = (\lambda_1 v_1 \mid \cdots \mid \lambda_d v_d).$$

What Laplacian Eigenmaps does

For $n \to \infty$ and $t \to 0$, convergence to the eigenstructure towards the Laplace-Beltrami operator Δ_M of the sampled manifold $M \subset \mathbb{R}^p$:

$$\begin{aligned} \Delta_M : \mathcal{C}^2(M) &\longrightarrow L^2(M) \\ f &\longmapsto -\mathrm{div}(\nabla f), \end{aligned}$$

Pointwise results in Belkin and Niyogi 2006.

General uniform convergence in García Trillos et al. 2020.

Refined versions in Wahl 2024.

Functional Formulation

Embed $M \subset \mathbb{R}^p$ in dimension $d = 1$ "as best as possible"

$\Leftrightarrow$

Find $f : M \to \mathbb{R}$ that preserves geodesic distances

Functional Formulation

Embed $M \subset \mathbb{R}^p$ in dimension $d = 1$ "as best as possible"

$\Leftrightarrow$

Find $f : M \to \mathbb{R}$ that preserves geodesic distances

If $f \in \mathcal{C}^2(M)$, then for all $x, x' \in M$,

$$\begin{aligned} |f(x) - f(x')| &\leq \big(\max_{\gamma_{x \to x'}} \|\nabla f\|\big) \mathrm{d}_M(x, x') \\ &\leq \big(\|\nabla_x f\| + o(1)\big) \mathrm{d}_M(x, x'). \end{aligned}$$

Functional Formulation

Embed $M \subset \mathbb{R}^p$ in dimension $d = 1$ "as best as possible"

$\Leftrightarrow$

Find $f : M \to \mathbb{R}$ that preserves geodesic distances

If $f \in \mathcal{C}^2(M)$, then for all $x, x' \in M$,

$$\begin{aligned} |f(x) - f(x')| &\leq \big(\max_{\gamma_{x \to x'}} \|\nabla f\|\big) \mathrm{d}_M(x, x') \\ &\leq \big(\|\nabla_x f\| + o(1)\big) \mathrm{d}_M(x, x'). \end{aligned}$$

$\Rightarrow$ $\|\nabla_x f\|$ testifies of how far apart f maps nearby points.

Functional Formulation

We may try to find f that preserves locality on average over M:

$$\arg\min_{f \in \mathcal{C}^2(M)} \int_M \|\nabla f\|^2 \mathrm{dvol}_M$$

Functional Formulation

We may try to find f that preserves locality on average over M:

$$\underset{f \in \mathcal{C}^2(M)}{\arg\min} \int_M \|\nabla f\|^2 \mathrm{dvol}_M$$

- $f \mapsto \nabla f$ is invariant by addition of constants $\Rightarrow$ Impose $\int_M f \mathrm{dvol}_M = 0$
- Taking $f = 0$ trivially solves the problem $\Rightarrow$ Impose $\int_M f^2 \mathrm{dvol}_M = 1$

Functional Formulation

We may try to find f that preserves locality on average over M:

$$\underset{f \in \mathcal{C}^2(M)}{\arg\min} \int_M \|\nabla f\|^2 \mathrm{dvol}_M$$

- $f \mapsto \nabla f$ is invariant by addition of constants $\Rightarrow$ Impose $\int_M f \mathrm{dvol}_M = 0$
- Taking $f = 0$ trivially solves the problem $\Rightarrow$ Impose $\int_M f^2 \mathrm{dvol}_M = 1$

We end up considering

$$\underset{\substack{\int_M f^2 \mathrm{dvol}_M = 1 \\ \int_M f \mathrm{dvol}_M = 0}}{\arg\min} \int_M \|\nabla f\|^2 \mathrm{dvol}_M$$

From Gradient to Laplace Operator

For all $f \in \mathcal{C}^2(M)$,

$$\int_M \|\nabla f\|^2 \mathrm{dvol}_M = \int_M \langle \nabla f, \nabla f \rangle \mathrm{dvol}_M = \int_M f \Delta_M f \mathrm{dvol}_M,$$

which leads to the energy minimization

$$\underset{\substack{\int_M f^2 \mathrm{dvol}_M = 1 \\ \int_M f \mathrm{dvol}_M = 0}}{\arg\min} \int_M f \Delta_M f \mathrm{dvol}_M.$$

From Laplace Operator to its Eigenstructure

$$f_1^* \in \underset{\substack{\int_M f^2 \mathrm{dvol}_M = 1 \\ \int_M f \mathrm{dvol}_M = 0}}{\arg\min} \int_M f \Delta_M f \mathrm{dvol}_M$$

$$\int_M f_1^* \mathrm{dvol}_M = 0 \qquad \int_M (f_1^*)^2 \mathrm{dvol}_M = 1$$

$$\int_M g \Delta_M f_1^* \mathrm{dvol}_M = \lambda_1 \int_M g f_1^* \mathrm{dvol}_M \text{ for all } g \in \mathcal{C}^2(M)$$

with associated energy

$$\int_M f_1^* \Delta_M f^* \mathrm{dvol}_M = \lambda_1 \int_M (f_1^*)^2 \mathrm{dvol}_M = \lambda_1.$$

Higher Order Eigenstructure

The *second* best such function naturally solves

$$f_2^* \in \underset{\substack{\int_M f^2 \mathrm{dvol}_M = 1 \\ \int_M f \mathrm{dvol}_M = 0 \\ \int_M f f_1^* \mathrm{dvol}_M = 0}}{\arg\min} \int_M f \Delta_M f \mathrm{dvol}_M$$

Applying Lagrange multipliers again

$$\int_M f_2^* \mathrm{dvol}_M = 0 \qquad \int_M (f_2^*)^2 \mathrm{dvol}_M = 1$$

$$\int_M g \Delta_M f_2^* \mathrm{dvol}_M = \lambda \int_M g f_2^* \mathrm{dvol}_M \text{ for all } g \in \mathcal{C}^2(M)$$

$$\text{and } \int_M f_1^* f_2^* \mathrm{dvol}_M = 0.$$

Discretization

Writing $K_t(x_i, x_j) = e^{-\frac{\|x_j - x_i\|^2}{t}} \mathbb{I}\{j \in \mathcal{N}_i\}$ and $y = f(x)$,

$$\begin{aligned}
\sum_{i,j} k_{i,j} \|y_j - y_i\|^2 &= \sum_i \sum_j \|f(x_j) - f(x_i)\|^2 K_t(x_i, x_j) \\
&\simeq_{n\to\infty} \int_M \int_M \|f(x') - f(x)\|^2 K_t(x, x') \mathrm{dvol}_M(x') \mathrm{dvol}_M(x) \\
&\simeq_{t\to 0} \int_M \int_{B(x,t)} \langle \nabla f(x), x' - x \rangle^2 \mathrm{dvol}_M(x') \mathrm{dvol}_M(x) \\
&\propto_{t\to 0} \int_M \|\nabla f\|^2 \mathrm{dvol}_M
\end{aligned}$$

Dictionnary

Object	Continuous	Discrete
Set	M	$\mathcal{X} \simeq \{1, \dots, n\}$
Point	$x \in M$	$i \in \{1, \dots, n\}$
Map	$f \in \mathbb{R}^M$	$y \in \mathbb{R}^{1 \times n}$
Gradient	$\nabla f(x)$	$\left(\sqrt{k_{i,j}}(f(x_j) - f(x_i))\right)_{j \in \mathcal{N}_i}$
Loss	$\int_M \|\nabla f\|^2 \mathrm{dvol}_M$	$\sum_{i,j} k_{i,j}(y_j - y_i)^2$

Hessian LLE – Hessian eigenmaps

Hessian LLE – Hessian eigenmaps originates from Donoho and Grimes 2003.

- Variant of LLE that minimizes the "curviness" of the high-dimensional manifold when embedding it into a low-dimensional space,
- Shares many characteristics with Laplacian Eigenmaps:

$\hookrightarrow$ it replaces the gradient by the Hessian.

$$\arg\min_{f} \int_{M} \|H_M f\|_{\mathrm{F}}^2 \mathrm{dvol}_M$$

Heuristic

If M is isometric to an open connected subset of $\mathbb{R}^d$, then the quadratic form

$$\mathcal{H}(f) := \int_M \|H_M f(x)\|^2 \mathrm{dvol}_M(x)$$

has a $(d+1)$-dimensional null space consisting of:

- the constant functions; (as for Laplacian Eigenmaps)
- a d-dimensional space of functions spanned by the original isometric coordinates.

Hence, the isometric coordinates can be recovered, up to a rigid motion, from the null space of $\mathcal{H}$.

Heuristic

If M is isometric to an open connected subset of $\mathbb{R}^d$, then the quadratic form

$$\mathcal{H}(f) := \int_M \|H_M f(x)\|^2 \mathrm{dvol}_M(x)$$

has a $(d+1)$-dimensional null space consisting of:

- the constant functions; (as for Laplacian Eigenmaps)
- a d-dimensional space of functions spanned by the original isometric coordinates.

Hence, the isometric coordinates can be recovered, up to a rigid motion, from the null space of $\mathcal{H}$.

Drawbacks

H-LLE is computationally heavy, as it requires to estimate tangent spaces at each data point.

H-LLE Example

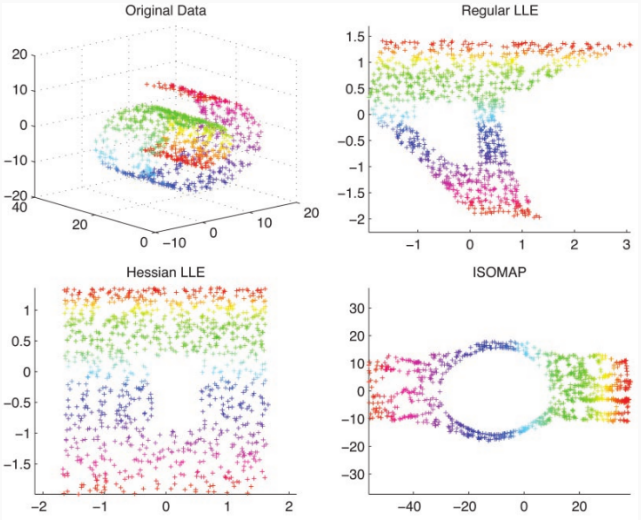


Figure 1: from Donoho and Grimes 2003.

Local Tangent Space Alignment

Local Tangent Space Alignment comes from Zhang and Zha 2004

Idea

If correctly unfolded, transformed data should have all its tangent spaces aligned

Related Methods

As Hessian LLE, LTSA works with charts from tangent spaces

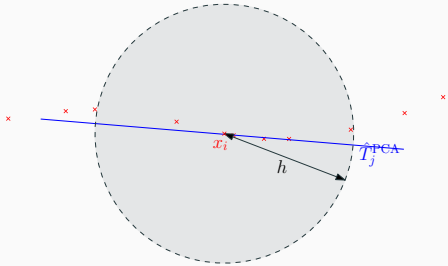
See also Parallel transport unfolding of Budninskiy et al. 2019.

Local Tangent Space Alignment

Step 1: Graph. Build neighbors $\mathcal{N}_i$ of each point x_i.

Step 2: Tangent Space Estimation. Estimate $T_{x_i}M$ by local PCA on neighborhood $\mathcal{N}_i$.

Step 3: Global coordinates. Find a global coordinate system that best represents neighborhoods in these tangent spaces.

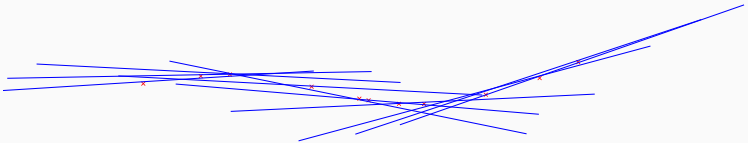


Local Tangent Space Alignment

Step 1: Graph. Build neighbors $\mathcal{N}_i$ of each point x_i.

Step 2: Tangent Space Estimation. Estimate $T_{x_i}M$ by local PCA on neighborhood $\mathcal{N}_i$.

Step 3: Global coordinates. Find a global coordinate system that best represents neighborhoods in these tangent spaces.



Kernel Methods

Kernel PCA

Kernel PCA formally arises from Schölkopf, Smola, and Müller 1998

It is a nonlinear generalization of PCA, already in use in applied statistics when introducing new variables to enrich the model.

Idea

- Transform data into features living in a higher (possibly infinite) dimensional space
- Apply PCA to them

Kernel PCA

Let $\mathcal{F}$ be a Hilbert space.

Let $\Phi : \mathbb{R}^p \to \mathcal{F}$ be the feature map.

Step 1: Featurization
Write $\tilde{\mathcal{X}} = \{\Phi(x_1), \ldots, \Phi(x_n)\}$.

Step 2: Principal components
Perform PCA on $\tilde{\mathcal{X}}$.

Here, Φ is fixed and problem-specific. Choice can be driven by:

- how linear $\Phi(\mathcal{X})$ is;
- how small $\dim(\mathcal{F})$ is.

Intended Principle of Kernel PCA

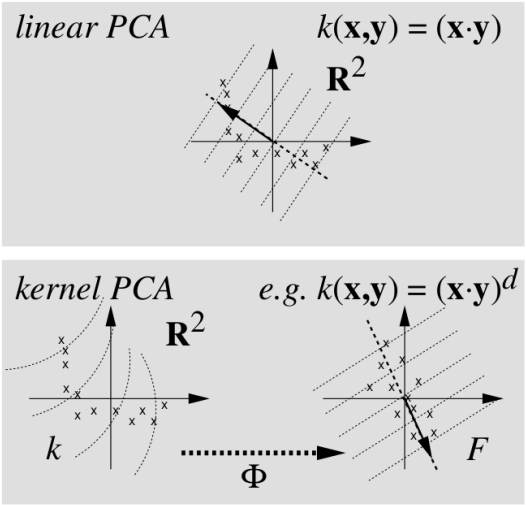


Figure 3: from Schölkopf, Smola, and Müller 1998

Why "Kernel" PCA?

Recall that given a point cloud $X = USV^\top \in \mathbb{R}^{n \times p}$,

PCA	$\Leftrightarrow$	**Classical scaling**
Covariance $X^\top X = VS^\top SV^\top$		Gram $XX^\top = USS^\top U^\top$

$$\hookrightarrow Y = US_{*,[d]} \hookleftarrow$$

Principal components in $\mathcal{F}$ can be computed only from $K(x_i, x_j) = \langle \Phi(x_i), \Phi(x_j) \rangle$, which can sometimes be done without computing any $\Phi(x)$.

This is often referred to as the kernel trick.

Step 1: Kernel Matrix: Compute $K := (K(x_i, x_j))_{1 \le i,j \le n}$

Step 2: SVD: Write $K = US^2U^\top$

Step 3: Truncation: Output $Y_{KPCA} := US_{*,[d]}$

Kernel Trick

- Polynomial kernel $K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^r$

Kernel Trick

- Polynomial kernel $K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^r$

$$K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^r = \sum_{\ell=1}^{r} \binom{r}{\ell} \langle x_i, x_j \rangle^\ell,$$

with

$$\langle x_i, x_j \rangle^\ell = \left(\sum_{k=1}^{d} x_i^{(k)} x_j^{(k)} \right)^\ell = \sum_{u_1 + \ldots + u_d = \ell} \frac{\ell!}{u_1! \ldots u_d!} (x_i^{(k)} x_j^{(k)})^{u_1} \ldots (x_i^{(k)} x_j^{(k)})^{u_d}$$

Kernel Trick

- Polynomial kernel $K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^r$

$$K(x_i, x_j) = (1 + \langle x_i, x_j \rangle)^r = \sum_{\ell=1}^{r} \binom{r}{\ell} \langle x_i, x_j \rangle^\ell,$$

with

$$\langle x_i, x_j \rangle^\ell = \left(\sum_{k=1}^{d} x_i^{(k)} x_j^{(k)} \right)^\ell = \sum_{u_1 + \ldots + u_d = \ell} \frac{\ell!}{u_1! \ldots u_d!} (x_i^{(k)} x_j^{(k)})^{u_1} \ldots (x_i^{(k)} x_j^{(k)})^{u_d}$$

Here, $\mathcal{F} = \mathbb{R}_{\leq r}[x^{(1)}, \ldots, x^{(d)}]$ and $\dim(\mathcal{F}) \asymp (d + r)^r$

Kernel Trick

- Gaussian kernel $K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$

Kernel Trick

- Gaussian kernel $K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$
  For $d = 1$ and $\sigma^2 = 1$,

$$K(x_i, x_j) = \left\langle e^{-x_i^2/2}\left(1, x_i, \frac{x_i^2}{\sqrt{2!}}, \dots\right), e^{-x_j^2/2}\left(1, x_j, \frac{x_j^2}{\sqrt{2!}}, \dots\right)\right\rangle_{\ell^2(\mathbb{N})}$$

Kernel Trick

- Gaussian kernel $K(x_i, x_j) = \exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma^2}\right)$
  For $d = 1$ and $\sigma^2 = 1$,

$$K(x_i, x_j) = \left\langle e^{-x_i^2/2}\left(1, x_i, \frac{x_i^2}{\sqrt{2!}}, \dots\right), e^{-x_j^2/2}\left(1, x_j, \frac{x_j^2}{\sqrt{2!}}, \dots\right)\right\rangle_{\ell^2(\mathbb{N})}$$

  Here, $\mathcal{F} = \ell^2(\mathbb{N})$ and $\dim(\mathcal{F}) = \infty$

KPCA in the initial space

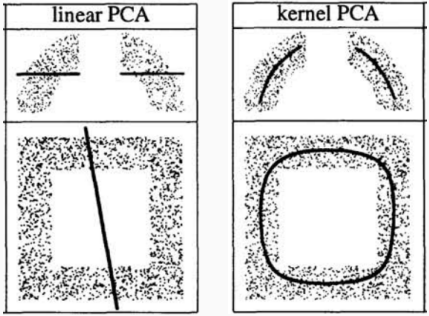


Figure 4: from Mika et al. 1998

Convergence of Kernel PCA

In infinite dimensions, formal link between covariance and convolution in Blanchard, Bousquet, and Zwald 2007, Theorem 2.3

Let $x_1, \ldots, x_n \sim_{\text{iid}} P$ on $\mathbb{R}^p$, with $\mathbb{E}_x \|\Phi(x)\|^2 < \infty$.

Convergence of Kernel PCA

In infinite dimensions, formal link between covariance and convolution in Blanchard, Bousquet, and Zwald 2007, Theorem 2.3

Let $x_1, \ldots, x_n \sim_{\text{iid}} P$ on $\mathbb{R}^p$, with $\mathbb{E}_x \|\Phi(x)\|^2 < \infty$.

In the limit $n \to \infty$, KPCA amounts to use the eigenstructure of

- The covariance operator $C_\Phi : \mathcal{F} \to \mathcal{F}$ defined as
$$C_\Phi z = \int_{\mathbb{R}^p} \Phi(x)\Phi(x)^\top z \, \mathrm{d}P(x)$$
- The convolution operator $K_\Phi : L^2(P) \mapsto L^2(P)$ defined as
$$(K_\Phi f)(x') = \int_{\mathbb{R}^p} f(x) \langle \Phi(x), \Phi(x') \rangle \, \mathrm{d}P(x)$$

(In fact, $C_\Phi = T^*T$ and $K_\Phi = TT^*$ with $T : \mathcal{F} \ni z \mapsto z^\top \Phi(\cdot) \in L^2(P)$)

Maximum Variance Unfolding / Semidefinite Embedding

Maximum Variance Unfolding arises in Weinberger, Sha, and Saul 2004.
(formerly known as Semidefinite Embedding)

Limitations of kernel PCA

KPCA is versatile and powerful, but the choice of the kernel is crucial

Different kernels are bound to reveal different types of low dimensional structure.

Idea

Learn a kernel matrix that reveals when high dimensional inputs lie on or near a low dimensional manifold.

Building Maximum Variance Unfolding: Constraints

We want a matrix $K \in \mathbb{R}^{n \times n}$

Building Maximum Variance Unfolding: Constraints

We want a matrix $K \in \mathbb{R}^{n \times n}$

- Interpretable as a Gram matrix $K = \big(\langle \Phi(x_i), \Phi(x_j) \rangle\big)_{1 \leq i,j \leq n}$
  - Symmetry $K = K^\top$
  - Positive semi-definiteness $\alpha^\top K \alpha \geq 0$ for all $\alpha \in \mathbb{R}^n$
- Arising from centered data $\sum_{i=1}^n \Phi(x_i) = 0$
  - This translates to $\sum_{i,j} K(x_i, x_j) = 0$

Building Maximum Variance Unfolding: Constraints

We want a matrix $K \in \mathbb{R}^{n \times n}$

- Interpretable as a Gram matrix $K = \big(\langle \Phi(x_i), \Phi(x_j) \rangle\big)_{1 \leq i,j \leq n}$
  - Symmetry $K = K^\top$
  - Positive semi-definiteness $\alpha^\top K \alpha \geq 0$ for all $\alpha \in \mathbb{R}^n$
- Arising from centered data $\sum_{i=1}^n \Phi(x_i) = 0$
  - This translates to $\sum_{i,j} K(x_i, x_j) = 0$
- That locally preserves distances in neighborhoods

  For all x_i, x_j sharing a neighbor (or themselves neighbors),

  $$\|\Phi(x_i) - \Phi(x_j)\|^2 = \|x_i - x_j\|^2.$$

  - Writing $G = \big(\langle x_i, x_j \rangle\big)_{i,j \leq n}$, this translates to

  $$K(x_i, x_i) + K(x_j, x_j) - K(x_i, x_j) - K(x_j, x_i) = G_{i,i} + G_{j,j} - G_{i,j} - G_{j,i}$$

Building Maximum Variance Unfolding: Objective Function

To unfold the manifold, pull the $\Phi(x_i)$'s as far apart as possible.

Formally, this can be done by maximizing

$$\begin{aligned} T(\Phi) &:= \frac{1}{2} \sum_{i,j} \|\Phi(x_i) - \Phi(x_j)\|^2 \\ &= \frac{1}{2} \sum_{i,j} K(x_i, x_i) + K(x_j, x_j) - 2K(x_i, x_j) \end{aligned}$$

From the centering constraint, $\sum_{i,j} K(x_i, x_j) = 0$, so that

$$T(\Phi) = \text{trace}(K).$$

Building Maximum Variance Unfolding

We get the following *semidefinite programming* (SDP) problem:

$$
\begin{aligned}
\text{maximize} \quad & \operatorname{trace}(K) \text{ over } K \in \mathbb{R}^{n \times n} \\
\text{subject to} \quad & K_{i,i} + K_{j,j} - 2K_{i,j} = G_{i,i} + G_{j,j} - 2G_{i,j}, \\
& \forall (i,j) \in \mathcal{E} \\
& \sum_{i,j} K_{ij} = 0 \\
& K \succeq 0
\end{aligned}
$$

An d-dimensional embedding is then obtained by a truncated SVD of the solution to this problem. (i.e. Kernel PCA applied to K)

Building Maximum Variance Unfolding

We get the following *semidefinite programming* (SDP) problem:

$$
\begin{aligned}
\text{maximize} \quad & \operatorname{trace}(K) \text{ over } K \in \mathbb{R}^{n \times n} \\
\text{subject to} \quad & K_{i,i} + K_{j,j} - 2K_{i,j} = G_{i,i} + G_{j,j} - 2G_{i,j}, \\
& \forall (i,j) \in \mathcal{E} \\
& \sum_{i,j} K_{ij} = 0 \\
& K \succeq 0
\end{aligned}
$$

An d-dimensional embedding is then obtained by a truncated SVD of the solution to this problem. (i.e. Kernel PCA applied to K)

Drawback: Size $O(n^2)$ SDP with $O(\sum_i |\mathcal{N}_i|^2)$ linear constraints.

Why "Maximum Variance Unfolding"?

The centering condition $\sum_{i=1}^{n} \Phi(x_i) = 0$ allows to interpret the eigenvalues of the kernel matrix as measures of variance along principal components in the feature space $\mathcal{F}$.

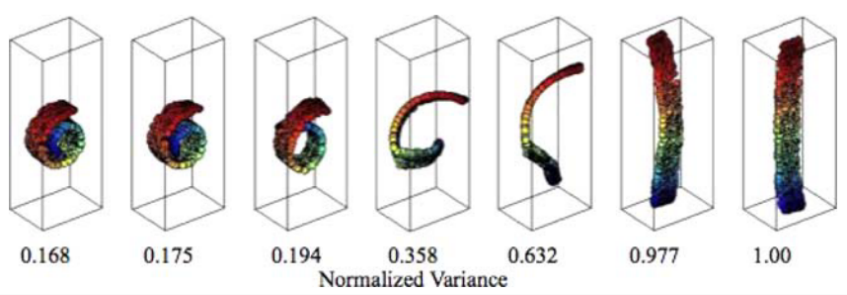


Figure 5: from Weinberger and Saul 2006

Convergence of MVU

Arias-Castro and Pelletier 2013 study the following formulation of MVU when $x_1, \dots, x_n \sim_{\text{iid}} P$ is random.

- Choose a neighborhood radius $r > 0$
- Write

$$\mathcal{Y}_{n,r} := \left\{ y_1, \dots, y_n \in \mathbb{R}^p \mid \|y_i - y_j\| \mathbf{1}_{\|x_i - x_j\| \le r} \le \|x_i - x_j\| \right\}$$

Discrete MVU

$$\begin{aligned} \text{maximize} \quad & E(Y) := \frac{1}{n(n-1)} \sum_{i,j} \|y_j - y_i\|^2 \\ \text{over} \quad & Y = (y_1 \cdots y_n)^\top \in \mathbb{R}^{n \times p} \\ \text{subject to} \quad & Y \in \mathcal{Y}_{n,r} \end{aligned}$$

Discrete MVU

$$\begin{aligned} \text{maximize} \quad & E(Y) := \frac{1}{n(n-1)} \sum_{i,j} \|y_j - y_i\|^2 \\ \text{over} \quad & Y = (y_1 \cdots y_n)^\top \in \mathbb{R}^{n \times p} \\ \text{subject to} \quad & Y \in \mathcal{Y}_{n,r} \end{aligned}$$

Convergence of MVU

Discrete MVU

$$\begin{aligned} \text{maximize} \quad & E(Y) := \frac{1}{n(n-1)} \sum_{i,j} \|y_j - y_i\|^2 \\ \text{over} \quad & Y = (y_1 \cdots y_n)^\top \in \mathbb{R}^{n \times p} \\ \text{subject to} \quad & Y \in \mathcal{Y}_{n,r} \end{aligned}$$

Continuum MVU

$$\begin{aligned} \text{maximize} \quad & E(f) := \int_{M \times M} \|f(x) - f(x')\|^2 P(\mathrm{d}x) P(\mathrm{d}x') \\ \text{over} \quad & f : \mathbb{R}^p \to \mathbb{R}^p \\ \text{subject to} \quad & \|f\|_{\mathrm{Lip}} \leq 1 \end{aligned}$$

Convergence of MVU

Theorem (Arias-Castro and Pelletier 2013)

Assume that $r = r_n \to 0$ slowly enough, and that

- *M is compact and connected*
- *For all $x, x' \in M$, $\mathrm{d}_M(x, x') \leq \big(1 + o(\|x - x'\|)\big)\|x - x'\|$*

Then as $n \to \infty$, almost surely, we have

$$\sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \to \sup_{\|f\|_{\mathrm{Lip}} \leq 1} E(f).$$

Furthermore, for all solution $\hat{Y} = (\hat{y}_1 \cdots \hat{y}_n)^\top$ of Discrete MVU,

$$\inf_{\|f\|_{\mathrm{Lip}} \leq 1} \max_{i \leq n} \|\hat{y}_i - f(x_i)\| \to 0$$

Convergence of MVU: Proof Ideas (Convergence of Energy)

Coverage: With high probability, $\sup_{x \in M} \min_{1 \leq i \leq n} \|x - x_i\| \leq \eta \to 0$

Convergence of MVU: Proof Ideas (Convergence of Energy)

Coverage: With high probability, $\sup_{x \in M} \min_{1 \le i \le n} \|x - x_i\| \le \eta \to 0$

Interpolation: All $Y \in \mathcal{Y}_{n,r}$ writes as $y_i = f(x_i)$, with $\|f\|_{\mathrm{Lip}} \lesssim 1 + \eta/r$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \le \sup_{\|f\|_{\mathrm{Lip}} \le 1 + \eta/r} E(Y_n(f))$$

Convergence of MVU: Proof Ideas (Convergence of Energy)

Coverage: With high probability, $\sup_{x \in M} \min_{1 \le i \le n} \|x - x_i\| \le \eta \to 0$

Interpolation: All $Y \in \mathcal{Y}_{n,r}$ writes as $y_i = f(x_i)$, with $\|f\|_{\mathrm{Lip}} \lesssim 1 + \eta/r$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \le \sup_{\|f\|_{\mathrm{Lip}} \le 1 + \eta/r} E(Y_n(f))$$

Comparison: For $\|f\|_{\mathrm{Lip}} \lesssim 1 - o(r)$, $Y_n(f) \in \mathcal{Y}_{n,r}$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \ge \sup_{\|f\|_{\mathrm{Lip}} \le 1 - o(r)} E(Y_n(f))$$

Convergence of MVU: Proof Ideas (Convergence of Energy)

Coverage: With high probability, $\sup_{x \in M} \min_{1 \le i \le n} \|x - x_i\| \le \eta \to 0$

Interpolation: All $Y \in \mathcal{Y}_{n,r}$ writes as $y_i = f(x_i)$, with $\|f\|_{\mathrm{Lip}} \lesssim 1 + \eta/r$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \le \sup_{\|f\|_{\mathrm{Lip}} \le 1 + \eta/r} E(Y_n(f))$$

Comparison: For $\|f\|_{\mathrm{Lip}} \lesssim 1 - o(r)$, $Y_n(f) \in \mathcal{Y}_{n,r}$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \ge \sup_{\|f\|_{\mathrm{Lip}} \le 1 - o(r)} E(Y_n(f))$$

Stability: $|E(f) - E(g)|$ and $|E(Y_n(f)) - E(Y_n(g))| \lesssim \|f - g\|_\infty$

Convergence of MVU: Proof Ideas (Convergence of Energy)

Coverage: With high probability, $\sup_{x \in M} \min_{1 \le i \le n} \|x - x_i\| \le \eta \to 0$

Interpolation: All $Y \in \mathcal{Y}_{n,r}$ writes as $y_i = f(x_i)$, with $\|f\|_{\mathrm{Lip}} \lesssim 1 + \eta/r$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \le \sup_{\|f\|_{\mathrm{Lip}} \le 1 + \eta/r} E(Y_n(f))$$

Comparison: For $\|f\|_{\mathrm{Lip}} \lesssim 1 - o(r)$, $Y_n(f) \in \mathcal{Y}_{n,r}$

$$\Rightarrow \sup_{Y \in \mathcal{Y}_{n,r}} E(Y) \ge \sup_{\|f\|_{\mathrm{Lip}} \le 1 - o(r)} E(Y_n(f))$$

Stability: $|E(f) - E(g)|$ and $|E(Y_n(f)) - E(Y_n(g))| \lesssim \|f - g\|_\infty$

Concentration: Hoeffding's Inequality for U-statistics and chaining yield

$$\sup_{\|f\|_{\mathrm{Lip}} \le 1} |E(Y_n(f)) - E(f)| \to 0$$

Convergence of MVU: Proof Ideas (Convergence of Solutions)

From the previous proof, any output $\hat{Y}_n$ of *Discrete MVU* writes as

$$\hat{y}_i = \hat{f}_n(x_i),$$

where $\|\hat{f}_n\|_{\mathrm{Lip}} \leq 1 + O(\eta/r)$ and $\eta/r \to 0$.

Convergence: One can show that $\inf_{\|f\|_{\mathrm{Lip}} \leq 1} \|\hat{f}_n - f\|_\infty \to 0$

Conclusion: By construction, for all $\|f\|_{\mathrm{Lip}} \leq 1$, we have

$$\max_{1 \leq i \leq n} \|\hat{y}_i - f(x_i)\| = \max_{1 \leq i \leq n} \|\hat{f}_n(x_i) - f(x_i)\| \leq \|\hat{f}_n - f\|_\infty$$

Convergence of MVU: Proof Ideas (Convergence of Solutions)

From the previous proof, any output $\hat{Y}_n$ of *Discrete MVU* writes as

$$\hat{y}_i = \hat{f}_n(x_i),$$

where $\|\hat{f}_n\|_{\mathrm{Lip}} \leq 1 + O(\eta/r)$ and $\eta/r \to 0$.

Convergence: One can show that $\inf_{\|f\|_{\mathrm{Lip}} \leq 1} \|\hat{f}_n - f\|_\infty \to 0$

Conclusion: By construction, for all $\|f\|_{\mathrm{Lip}} \leq 1$, we have

$$\max_{1 \leq i \leq n} \|\hat{y}_i - f(x_i)\| = \max_{1 \leq i \leq n} \|\hat{f}_n(x_i) - f(x_i)\| \leq \|\hat{f}_n - f\|_\infty$$

Other Convergence Result in Paprotny and Garcke 2012

MVU asymptotically recovers a geodesic distance matrix of data.

MVU vs Isomap

MVU can be seen as a regularized version of the shortest path problem on a graph (Paprotny and Garcke 2012).

MVU vs Isomap

MVU can be seen as a regularized version of the shortest path problem on a graph (Paprotny and Garcke 2012). Write

$$E_{i,j} := (e_i - e_j)(e_i - e_j)^\top$$

Given the neighborhood graph $G = (\{1, \dots, n\}, \mathcal{E})$, MVU writes as

$$\begin{aligned} \text{maximize} \quad & \operatorname{trace}(K) \text{ over } K \in \mathbb{R}^{n \times n} \\ \text{subject to} \quad & \operatorname{trace}\big(E_{i,j}^\top K\big) \leq \|x_i - x_j\|^2 \quad \forall (i,j) \in \mathcal{E} \\ & \operatorname{trace}\big((11^\top)^\top K\big) = 0 \\ & K \succeq 0 \end{aligned}$$

($E_{i,j}$ transforms scalar products (back) to squared distances)

MVU vs Isomap

Paprotny and Garcke 2012 establish another formulation of MVU in terms of squared distance matrices $D = \Delta^{\circ 2}$.

$$\begin{aligned} \text{maximize} \quad & \operatorname{trace}\big((11^\top)^\top D\big) \text{ over } D \in \mathbb{R}^{n\times n} \\ \text{subject to} \quad & \operatorname{trace}\big(e_j e_i^\top D\big) \le \|x_i - x_j\|^2 \quad \forall (i,j) \in \mathcal{E} \\ & D \in \mathcal{D}_{\text{Eucl}} \end{aligned}$$

Here, $\mathcal{D}_{\text{Eucl}}$ stands for the (squared) Euclidean distance matrices of order n:

$$\mathcal{D}_{\text{Eucl}} = \bigcup_{d\ge 1} \left\{ \left(\|y_j - y_i\|^2\right)_{i,j\le n} \mid y_1, \ldots, y_n \in \mathbb{R}^d \right\}.$$

$\mathcal{D}_{\text{Eucl}}$ is a closed convex cone.

MVU vs Isomap

This suggests a broad metric generalization of MVU.

Given any metric graph $(\{1,\dots,n\},\mathcal{E},\Delta)$, non-Euclidean MVU writes as

$$\begin{aligned} \text{maximize} \quad & \operatorname{trace}\big((11^\top)^\top D\big) \text{ over } D \in \mathbb{R}^{n\times n} \\ \text{subject to} \quad & \operatorname{trace}\big(e_j e_i^\top D\big) \le \delta_{i,j}^2 \quad \forall (i,j) \in \mathcal{E} \\ & D \in \mathcal{D}_{\text{Metric}} \end{aligned}$$

Here, $\mathcal{D}_{\text{Metric}}$ stands for the (squared) distance matrices of order n:

$$\mathcal{D}_{\text{Metric}} = \left\{ \left(\delta_{i,j}^2\right)_{i,j\le n} \succeq 0 \mid \delta_{i,i} = 0,\ \delta_{i,j} = \delta_{j,i} \text{ and } \delta_{i,j} \le \delta_{i,k} + \delta_{k,j} \right\}.$$

$\mathcal{D}_{\text{Metric}}$ is a closed convex cone.

The optimization can also be restricted to any $\mathcal{C} \subset \mathcal{D}_{\text{Metric}}$

MVU vs Isomap

Theorem (Paprotny and Garcke 2012)

Let $\mathcal{C} \subset \mathcal{D}_{\text{Metric}}$, and assume that $G = (\{1, \dots, n\}, \mathcal{E}, \Delta)$ is connected. Then non-Euclidean MVU over $\mathcal{C}$ is equivalent to

$$\begin{aligned} &\textit{minimize} && \|D - \Delta_G^{\circ 2}\|_{\ell^1} \textit{ over } D \in \mathbb{R}^{n \times n} \\ &\textit{subject to} && \operatorname{trace}\left(e_j e_i^\top D\right) \leq \delta_{i,j}^2 \quad \forall (i,j) \in \mathcal{E} \\ & && D \in \mathcal{C} \end{aligned}$$

where

$\Delta_G^{\circ 2}$ *is the squared geodesic distance matrix over* $(\{1, \dots, n\}, \mathcal{E}, \Delta)$.

MVU vs Isomap

$$\begin{aligned} \text{minimize} \quad & \|D - \Delta_G^{\circ 2}\|_{\ell^1} \text{ over } D \in \mathbb{R}^{n \times n} \\ \text{subject to} \quad & \operatorname{trace}\big(e_j e_i^\top D\big) \leq \delta_{i,j}^2 \quad \forall (i,j) \in \mathcal{E} \\ & D \in \mathcal{C} \end{aligned}$$

As a corollary:

- (Generalized) MVU actually solved a shortest path problem
- Isomap writes similarly with $\mathcal{C} = \mathcal{D}_{\text{Eucl}}^{(d)}$, but a ℓ_2 norm
  $\Rightarrow$ MVU $\simeq$ ℓ_1-regularized Isomap
- MVU implicitely incorporates global geodesic distances

MVU vs Laplacian Eigenmaps

In the same vein, variants of Laplacian Eigenmaps can be cast as the following modified MVU program.

$$
\begin{aligned}
\text{maximize} \quad & \operatorname{trace}(K) \text{ over } K \in \mathbb{R}^{n \times n} \\
\text{subject to} \quad & \operatorname{trace}\big(E_{i,j}^{\top} K\big) \leq \mathrm{d}_G(i,j)^2 \quad \forall (i,j) \in \mathcal{E} \\
& \operatorname{trace}\big((11^{\top})^{\top} K\big) = 0 \\
& K \succeq 0 \\
& \|K\|_{\mathrm{op}} \leq 1
\end{aligned}
$$

$$
\Rightarrow \text{Laplacian Eigenmaps} \simeq \text{MVU} + \|K\|_{\mathrm{op}} \leq 1
$$

Diffusion Maps

Diffusion Maps originate from Coifman and Lafon 2006.

Very related to Laplacian Eigenmaps

Idea

Interpret kernel K as a node affinity representing transition probabilities from neighboring points.

Use eigenfunctions of the associated Markov chain (random walk) for embedding.

Markovian Framework

Let

$$p(x_i, x_j) := \frac{K(x_i, x_j)}{\sum_{k=1}^{n} K(x_i, x_k)}$$

represent the probability of transition in one time step from node x_i to node x_j.

Unlike K, p is no longer symmetric, but can be interpreted as the transition kernel of a reversible Markov chain.
That is, writing $\mu_{\mathcal{X}} = \sum_{i=1}^{n} \delta_{x_i}$ for the empirical measure,

$$p : L^2(\mathcal{X}, \mu_{\mathcal{X}}) \ni f \mapsto \sum_{j=1}^{n} p(\cdot, x_j) f(x_j).$$

Markovian Framework

Let

$$p(x_i, x_j) := \frac{K(x_i, x_j)}{\sum_{k=1}^{n} K(x_i, x_k)}$$

represent the probability of transition in one time step from node x_i to node x_j.

Unlike K, p is no longer symmetric, but can be interpreted as the transition kernel of a reversible Markov chain.
That is, writing $\mu_{\mathcal{X}} = \sum_{i=1}^{n} \delta_{x_i}$ for the empirical measure,

$$p : L^2(\mathcal{X}, \mu_{\mathcal{X}}) \ni f \mapsto \sum_{j=1}^{n} p(\cdot, x_j) f(x_j).$$

For all $t \in \mathbb{N}$, the probability of transition from x_i to x_j t time steps is $p_t(x_i, x_j)$.
Matricially, $p_t = p^t$.

Spectral Decomposition of the Markov chain

Under mild assumptions on K, p has a discrete sequence of eigenfunctions $\psi_\ell \in L^2(\mathcal{X}, \mu_\mathcal{X})$ and eigenvalues $1 = \lambda_0 \geq \lambda_1 \geq \ldots \geq \lambda_n \geq 0$, with

$$p\psi_\ell = \lambda_\ell \psi_\ell.$$

Diffusion Maps

For all $d \geq 1$, we associate embeddings indexed by $t \in \mathbb{N}$, called diffusion maps:

$$\Psi_t^{(d)}(x_i) := \left(\lambda_1^t \psi_1(x_i), \cdots, \lambda_d^t \psi_d(x_i)\right)^\top \in \mathbb{R}^d.$$

Stationary Distribution

This Markov chain has a stationary distribution ($\pi p = \pi$) given by

$$\pi(x_i) := \frac{\sum_{j=1}^{n} K(x_i, x_j)}{\sum_{k,j=1}^{n} K(x_k, x_j)}.$$

Diffusion Distances

For all $t \in \mathbb{N}$, the Diffusion Distances are defined by

$$\begin{aligned} \mathrm{D}_t(x_i, x_j)^2 &:= \|p_t(x_i, \cdot) - p_t(x_j, \cdot)\|^2_{L^2(\mathcal{X}, \mu_{\mathcal{X}}/\pi)} \\ &= \sum_{k=1}^{n} \big(p_t(x_i, x_k) - p_t(x_j, x_k)\big)^2 \frac{1}{\pi(x_k)}. \end{aligned}$$

In fact, we also have

$$\mathrm{D}_t(x_i, x_j)^2 = \sum_{\ell \geq 1} \lambda_\ell^{2t} (\psi_\ell(x_i) - \psi_\ell(x_j))^2.$$

Truncation

For all $d \geq 1$, truncate the sum to obtain the Truncated Diffusion Distance

$$\mathrm{D}_t^{(d)}(x_i, x_j)^2 := \sum_{\ell=1}^{d} \lambda_\ell^{2t} (\psi_\ell(x_i) - \psi_\ell(x_j))^2.$$

Isometry

By construction, diffusion map Ψ_t embeds the data into the Euclidean space $\mathbb{R}^d$, isometrically with respect to the diffusion distance (up to eigenvector d):

$$\|\Psi_t^{(d)}(x_i) - \Psi_t^{(d)}(x_j)\| = \mathrm{D}_t^{(d)}(x_i, x_j).$$

Insights Behind Diffusion Maps: What do Diffusion Distances Measure?

- $\mathrm{D}_t(x_i, x_j)$ small $\Leftrightarrow$ large number of short paths $x_i \leftrightarrow x_j$
- $\mathrm{D}_t(x_i, x_j)$ involves all paths of length t connecting $x_i \leftrightarrow x_j$
  $\Rightarrow$ Robustness to noise (better than geodesic distance)
- t plays the role of a scale parameter. The larger t:
  - The more spread the support of $p_t(x_i, \cdot)$ (diffusion)
  - The less significant eigenvalues $\Rightarrow$ We can take d smaller.

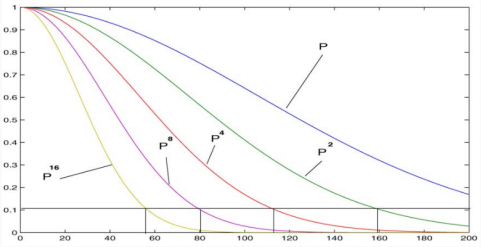


Figure 6: from Coifman and Lafon 2006

Continuous Limit

Diffusion Maps are strongly related to the Heat kernel

$$k_M(t,\cdot,\cdot) : M \times M \to \mathbb{R},\ t \geq 0.$$

For all $x_0 \in M$, $k_M(t, x_0, x)$ is the minimal positive solution of the heat equation

$$\begin{cases} \dfrac{\partial k_M(t,x_0,\cdot)}{\partial t} = \Delta_M k_M(t,x_0,\cdot) \\ \lim_{t\to 0^+} k_M(t,x_0,x) = 1_{x_0=x} \end{cases}$$

Continuous Limit

Diffusion Maps are strongly related to the Heat kernel

$$k_M(t,\cdot,\cdot) : M \times M \to \mathbb{R},\ t \geq 0.$$

For all $x_0 \in M$, $k_M(t, x_0, x)$ is the minimal positive solution of the heat equation

$$\begin{cases} \dfrac{\partial k_M(t,x_0,\cdot)}{\partial t} = \Delta_M k_M(t,x_0,\cdot) \\ \lim_{t\to 0^+} k_M(t,x_0,x) = 1_{x_0=x} \end{cases}$$

Theorem (Grigor'yan, Hu, and Lau 2014)

For arbitrary smooth Riemannian manifold M,

$$\log k_M(t,x_0,x) \sim_{t\to 0} -\mathrm{d}_M(x_0,x)^2/(4t).$$

Bayesian Methods

Stochastic Neighbor Embedding

Stochastic Neighbor Embedding (SNE) orginates from Hinton and Roweis 2002.

It has a probabilistic approach.

Idea

- Define a probability distribution over all "potential neighbors of each point" of $X \in \mathbb{R}^{n \times p}$
- Align this distribution as well as possible, when doing the same operation on the low-dimensional embedding $Y \in \mathbb{R}^{n \times d}$
- Measure distances between probability distributions with Kullback Leibler divergence

Stochastic Neighbor Embedding

Step 1: Input Probabilities. *(don't ask of what)*

Given data $X = (x_1 \mid \cdots \mid x_n)^\top \in \mathbb{R}^{n \times p}$, write

$$p_{i|j} = \frac{\exp(-\|x_i - x_j\|^2/(2\sigma_i^2))}{\sum_{\ell \neq i} \exp(-\|x_\ell - x_j\|^2/(2\sigma_\ell^2))} \text{ for all } i \neq j \in \{1, \ldots, n\}$$

Step 2: Latent Probabilities.

Given candidate embedding $Y = (y_1 \mid \cdots \mid y_n)^\top \in \mathbb{R}^{n \times d}$, write

$$q_{i|j}(Y) = \frac{\exp(-\|y_i - y_j\|^2/(2\sigma_i^2))}{\sum_{\ell \neq i} \exp(-\|y_\ell - y_j\|^2/(2\sigma_\ell^2))},$$

Step 3: Alignment. Embed points with

$$Y \in \operatorname*{arg\,min}_{Y \in \mathbb{R}^{n \times d}} \sum_{i=1}^{n} \mathrm{KL}(P_i || Q_i(Y)) = \sum_{i \neq j} p_{i|j} \log\left(\frac{p_{i|j}}{q_{i|j}(Y)}\right)$$

SNE

Strengths

- Explicit (stochastic) gradient method
- Good at preserving local distances

Weaknessess

- Not so good for global representation
- Does not handle well high dimensional data (preliminary PCA and feature selection)
- Sensitive to the calibration of the hyperparameter

SNE

Strengths

- Explicit (stochastic) gradient method
- Good at preserving local distances

Weaknessess

- Not so good for global representation
- Does not handle well high dimensional data (preliminary PCA and feature selection)
- Sensitive to the calibration of the hyperparameter

Variants

- With symmetrized probabilities $p_{i,j} = (p_{i|j} + p_{j|i})/(2n)$ and $q_{i,j} = (q_{i|j} + q_{j|i})/(2n)$
- With other probability families: t-SNE, UMAP, LargeVis

Variants of SNE

t-**SNE** originates in Van Der Maaten 2009

The method adds robustness by using Student's distribution:

Input probabilities:
$$p_{i|j} = \frac{\exp(-\|x_i - x_j\|^2/(2\sigma_i^2))}{\sum_{\ell \neq i} \exp(-\|x_\ell - x_j\|^2/(2\sigma_\ell^2))},$$

Latent probabilities:
$$q_{i|j} = \frac{(1 + \|y_i - y_j\|^2/\delta)^{-(\delta+1)/2}}{\sum_{\ell \neq i}(1 + \|y_i - y_\ell\|^2/\delta)^{-(\delta+1)/2}}.$$

Variants of SNE

t-**SNE** originates in Van Der Maaten 2009

The method adds robustness by using Student's distribution:

$$\textit{Input probabilities:} \qquad p_{i|j} = \frac{\exp(-\|x_i - x_j\|^2/(2\sigma_i^2))}{\sum_{\ell \neq i} \exp(-\|x_\ell - x_j\|^2/(2\sigma_\ell^2))},$$

$$\textit{Latent probabilities:} \qquad q_{i|j} = \frac{(1 + \|y_i - y_j\|^2/\delta)^{-(\delta+1)/2}}{\sum_{\ell \neq i}(1 + \|y_i - y_\ell\|^2/\delta)^{-(\delta+1)/2}}.$$

From t-SNE to SNE

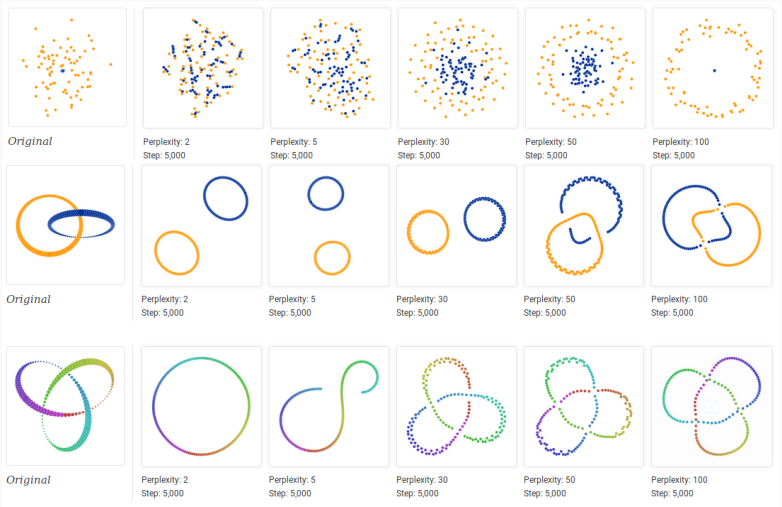
When $\delta \to \infty$, we recover regular SNE.

LargeVis

(Tang et al. 2016) approximates k-NN to accelerate computations.

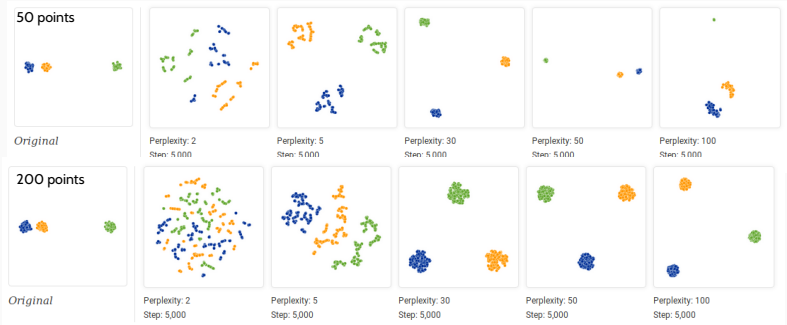
t-SNE on Synthetic Data

Catching Complex Geometries

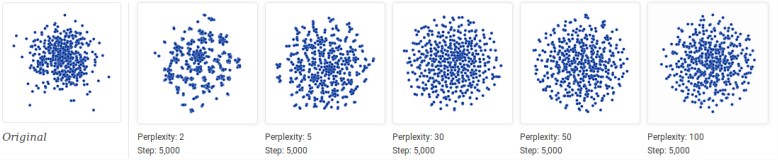


t-SNE on Synthetic Data

tSNE does not account for between-cluster distance



What about random noise ?



Variants of SNE

Uniform Manifold Approximation and Projection

UMAP originates in McInnes, Healy, and Melville 2018.

The original paper claims that it functorializes embeddings.

Uniform Manifold Approximation and Projection

UMAP originates in McInnes, Healy, and Melville 2018.

The original paper claims that it functorializes embeddings.

Definition 8. *Define the functor* FinSing : $\mathbf{FinEPMet} \to \mathbf{Fin\text{-}sFuzz}$ *by*

$$\mathsf{FinSing}(Y) : ([n], [0, a)) \mapsto \hom_{\mathbf{FinEPMet}}(\mathsf{FinReal}(\Delta^n_{<a}), Y).$$

We then have the following theorem.

Theorem 1. *The functors* FinReal : $\mathbf{Fin\text{-}sFuzz} \to \mathbf{FinEPMet}$ *and* FinSing : $\mathbf{FinEPMet} \to \mathbf{Fin\text{-}sFuzz}$ *form an adjunction with* FinReal *the left adjoint and* FinSing *the right adjoint.*

The proof of this is by construction. Appendix B provides a full proof of the theorem.

Figure 9: from McInnes, Healy, and Melville 2018

UMAP

Take *input probabilities* as

$$p_{ij} = p_{j|i} + p_{i|j} - p_{j|i}p_{i|j},$$

where

$$p_{j|i} \propto \exp\left(-\frac{\|x_i - x_j\| - \rho_i}{\sigma_i}\right) \text{ and } \rho_i = \min_{j \neq i} \|x_i - x_j\|,$$

and *latent probabilities* as

$$q_{ij} \propto \left(1 + a\|y_i - y_j\|_2^{2b}\right)^{-1}$$

UMAP

Take *input probabilities* as

$$p_{ij} = p_{j|i} + p_{i|j} - p_{j|i}p_{i|j},$$

where

$$p_{j|i} \propto \exp\left(-\frac{\|x_i - x_j\| - \rho_i}{\sigma_i}\right) \text{ and } \rho_i = \min_{j \neq i} \|x_i - x_j\|,$$

and *latent probabilities* as

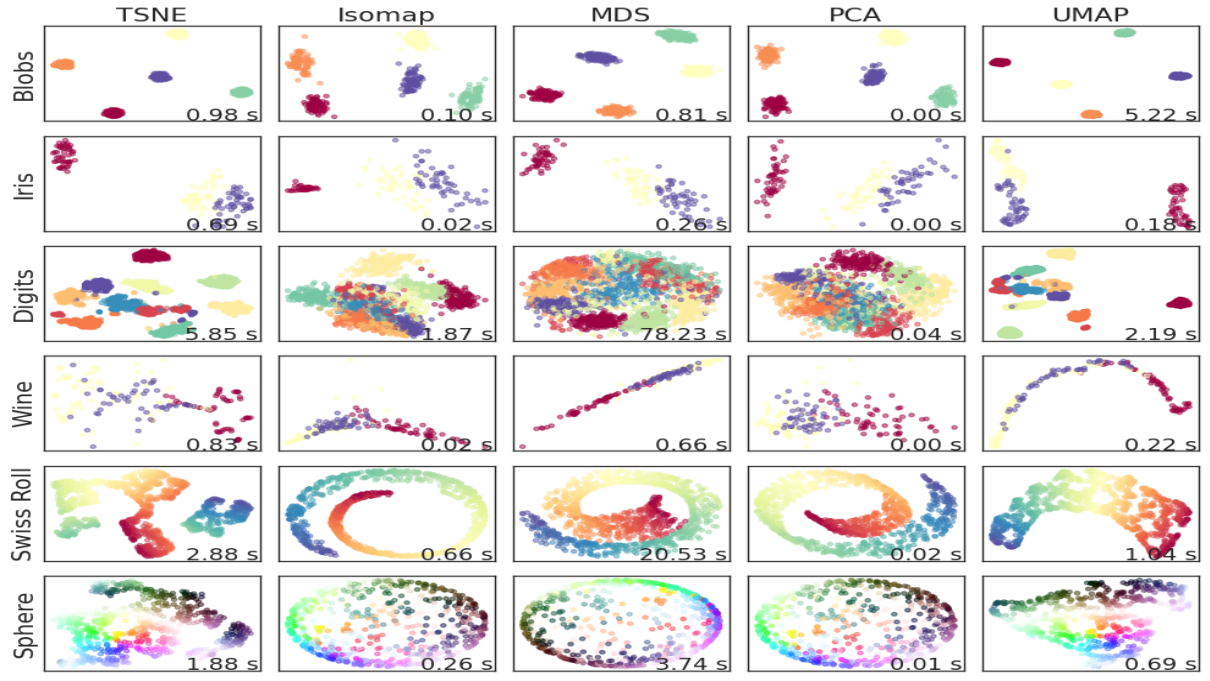
$$q_{ij} \propto \left(1 + a\|y_i - y_j\|_2^{2b}\right)^{-1}$$

The criterion is the cross-entropy:

$$Y \in \operatorname*{arg\,min}_{Y \in \mathbb{R}^{n \times d}} - \sum_{i<j} p_{ij} \log q_{ij} + (1 - p_{ij}) \log(1 - q_{ij}).$$

Visual comparison

- **Blobs:** A set of five separated gaussian blobs in 10 dimensional space. This should be a prototypical example
- **Iris:** a classic small dataset with one distinct class and two classes that are not clearly separated.
- **Digits:** handwritten digit. Due to the nature of handwriting, digits may have several forms (crossed or uncrossed sevens, capped or straight line "0", etc.)
- **Wine:** wine characteristics ideally used for a toy regression. Ultimately the data is essentially one dimensional in nature.
- **Swiss Roll:** data is essentially a rectangle, but has been "rolled up" like a swiss roll in three dimensional space.
- **Sphere:** the two dimensional unit sphere. It has been coloured with hue around the equator and black to white from the south to north pole.


TSNE
Isomap
MDS
PCA
UMAP
Blobs
Iris
Digits
Wine
Swiss Roll
Sphere
0.98 s
0.10 s
0.81 s
0.00 s
5.22 s
0.69 s
0.02 s
0.26 s
0.00 s
0.18 s
5.85 s
1.87 s
78.23 s
0.04 s
2.19 s
0.83 s
0.02 s
0.66 s
0.00 s
0.22 s
2.88 s
0.66 s
20.53 s
0.02 s
1.04 s
1.88 s
0.26 s
3.74 s
0.01 s
0.69 s

Unified Framework: Graph Coupling

Van Assel et al. 2022 proposed a unifying framework based on a Bayesian Method and Graph Couplings.

Idea

- *Model:* Observations X and Y are structured by two latent (weighted) graphs W_X and W_Y.
- *Prior:* Consider random graphs (W_X, W_Y) distributed according to some predefined prior distributions.
- *Posterior Alignment:* Match the posterior distributions of $W_X|X$ and $W_Y|Y$ with a *cross-entropy criterion*.
  The associated minimization is called graph coupling.

See also (Damrich and Hamprecht 2021) for links with force-directed graph drawing.

Unified Framework: Graph Coupling

Fix a symmetric kernel $k : \mathbb{R}^p \to \mathbb{R}_{\geq 0}$.

Model for $X|W$

Given W, consider the (unnormalized) density $X|W$ given by

$$\begin{array}{rccc} f_K : & \mathbb{R}^{n\times p} \times [0:n]^{n\times n} & \longrightarrow & \mathbb{R}_{\geq 0} \\ & (X|W) & \longmapsto & \prod_{1\leq i,j\leq n} k\big((x_i - x_j)/\tau_i\big)^{W_{i,j}} \end{array}$$

$f_K(x|W)$ large $\Leftrightarrow$ $(x, x') \mapsto k(x - x')$ varies smoothly on W

Unified Framework: Graph Coupling

Fix a symmetric kernel $k : \mathbb{R}^p \to \mathbb{R}_{\geq 0}$.

Model for $X|W$

Given W, consider the (unnormalized) density $X|W$ given by

$$\begin{array}{rccc} f_K\colon & \mathbb{R}^{n\times p} \times [0:n]^{n\times n} & \longrightarrow & \mathbb{R}_{\geq 0} \\ & (X|W) & \longmapsto & \prod_{1\leq i,j\leq n} k\big((x_i - x_j)/\tau_i\big)^{W_{i,j}} \end{array}$$

$f_K(x|W)$ large $\Leftrightarrow$ $(x, x') \mapsto k(x - x')$ varies smoothly on W

Given $W \in [0:n]^{n\times n}$, define the conditional distribution on $\mathbb{R}^{n\times p}$

$$\mathbb{P}(\mathrm{d}X|W) := C_K(W)^{-1} f_K(X|W) \mathrm{d}\lambda_{\mathbb{R}^{n\times p}}(\mathrm{d}X)$$

(Hiding integrability issues due to translation invariance depending on W)

Unified Framework: Graph Coupling

Priors for W

Build conjugate priors constraining the topology of the graph:

(B) Binary	$\Omega_B(W) = \prod_{i,j} \mathbb{I}\{W_{i,j} \leq 1\}$
(D) Unitary out-degree	$\Omega_D(W) = \prod_i \mathbb{I}\{\sum_j W_{i,j} = 1\}$
(E) With n-edges	$\Omega_E(W) = \mathbb{I}\{\sum_{i,j} W_{i,j} = n\} \prod_{1 \leq i,j \leq n} (W_{i,j}!)^{-1}$

For $\mathcal{P} \in \{B, D, E\}$, $\pi \in \left(\mathbb{R}_{\geq 0}\right)^{n \times n}$ and $\alpha \geq 0$,

$$\mathbb{P}_{\mathcal{P},K}(W; \pi, \alpha) \propto C_K(W)^\alpha \Omega_{\mathcal{P}}(W) \prod_{1 \leq i,j \leq n} \pi_{i,j}^{W_{i,j}}.$$

Unified Framework: Graph Coupling

Priors for W

When $W \sim \mathbb{P}_{\mathcal{P},K}(\cdot; \pi, \alpha = 0)$,

(B) $W_{i,j} \overset{\perp\!\!\!\perp}{\sim} \text{Bernoulli}(\pi_{i,j}/(1+\pi_{i,j}))$
$\hookrightarrow$ Each edge independent

(D) $W_i \overset{\perp\!\!\!\perp}{\sim} \text{Multinomial}(1, \pi_i/\sum_j \pi_{i,j})$
$\hookrightarrow$ Each node chooses its (unique) neighbor independently

(E) $W \sim \text{Multinomial}(n, \pi/\sum_{i,j} \pi_{i,j})$
$\hookrightarrow$ n edges overall, chosen multinomially

Unified Framework: Graph Coupling

Posterior $W|X$

Theorem (Van Assel et al. 2022)

Under mild assumptions, if $W \sim \mathbb{P}_{\mathcal{P}}(\cdot, \pi, \alpha = 1)$, *then*

$$W|X \sim \mathbb{P}_{\mathcal{P}}(\cdot, \pi \odot K_X),$$

where $\odot$ *is the Hadamard product, and* $K_X = \big(k(x_i - x_j)\big)_{i,j \leq n}$

Posterior $W|X$

Theorem (Van Assel et al. 2022)

Under mild assumptions, if $W \sim \mathbb{P}_{\mathcal{P}}(\cdot, \pi, \alpha = 1)$*, then*

$$W|X \sim \mathbb{P}_{\mathcal{P}}(\cdot, \pi \odot K_X),$$

where $\odot$ *is the Hadamard product, and* $K_X = \big(k(x_i - x_j)\big)_{i,j \leq n}$

(B) $W_{i,j}|X \overset{\perp\!\!\!\perp}{\sim} \text{Bernoulli}\left(\frac{\pi_{i,j} k(x_i - x_j)}{1 + \pi_{i,j} k(x_i - x_j)}\right)$

(D) $W_i|X \overset{\perp\!\!\!\perp}{\sim} \text{Multinomial}\left(1, \left(\frac{\pi_{i,j} k(x_i - x_j)}{\sum_{\ell \leq n} \pi_{i\ell} k(x_i - x_\ell)}\right)_{j \leq n}\right)$

(E) $W|X \sim \text{Multinomial}\left(n, \left(\frac{\pi_{i,j} k(x_i - x_j)}{\sum_{\ell,t \leq n} \pi_{\ell,t} k(x_\ell - x_t)}\right)_{i,j \leq n}\right)$

Graph Coupling

Consider both graph priors with $\pi = (1)_{1 \leq i,j \leq n}$ and $\alpha = 1$.

For $(\mathcal{P}_X, \mathcal{P}_Y) \in \{B, D, E\}^2$, minimize the cross entropy between the posteriors:

$$Y \in \underset{Y \in \mathbb{R}^{n \times d}}{\arg\min} \left\{ \mathcal{H}_{X,Y} := -\mathbb{E}_{W_X \sim \mathbb{P}_{\mathcal{P}_X}(\cdot; K_X)} \left[\log \mathbb{P}_{\mathcal{P}_Y}(W_X; K_Y) \right] \right\}$$

Graph Coupling

Consider both graph priors with $\pi = (1)_{1 \leq i,j \leq n}$ and $\alpha = 1$.

For $(\mathcal{P}_X, \mathcal{P}_Y) \in \{B, D, E\}^2$, minimize the cross entropy between the posteriors:

$$Y \in \underset{Y \in \mathbb{R}^{n \times d}}{\arg\min} \left\{ \mathcal{H}_{X,Y} := -\mathbb{E}_{W_X \sim \mathbb{P}_{\mathcal{P}_X}(\cdot; K_X)} \left[\log \mathbb{P}_{\mathcal{P}_Y}(W_X; K_Y) \right] \right\}$$

$\mathcal{P}_X$ \ $\mathcal{P}_Y$	B	D	E
$\tilde{B}$	UMAP	$(*)$	$(*)$
D	LargeVis	SNE	T-SNE
E	$(*)$	$(*)$	Tang et al. 2016

$(*)$ yield $\text{Support}(\mathbb{P}_{\mathcal{P}_X}) \not\subset \text{Support}(\mathbb{P}_{\mathcal{P}_Y})$, so that $\mathcal{H}_{X,Y} = \infty$.

Attraction / Repulsion

Decomposing and simplifying $\mathcal{H}_{X,Y}$ with Bayes', we get

$$\arg\min_{Y \in \mathbb{R}^{n \times d}} - \sum_{1 \le i,j \le n} P_{i,j}^{\mathcal{P}_X} \log k_y(y_i - y_j) + \log \mathbb{P}(Y).$$

- $P_{i,j}^{\mathcal{P}_X}$ is the posterior expectation of W_X
  $\hookrightarrow$ Tends to bring y_i's close.
  For the Gaussian kernel $-\log k_y(t) \propto t^2$,
  $$- \sum_{1 \le i,j \le n} P_{i,j}^{\mathcal{P}_X} \log k_y(y_i - y_j) = \operatorname{trace}\left(Y^\top \mathbb{E}_{W \sim \mathbb{P}_{\mathcal{P}_X}(\cdot, K_X)}[L(W)] Y\right)$$
  is reminiscent of Laplacian Eigenmaps

- $\mathbb{P}(Y) = \sum_W \mathbb{P}(Y, W)$ with $\mathbb{P}(Y, W) \propto f_K(Y|W) \Omega_{\mathcal{P}_Y}(W)$
  $\hookrightarrow$ Prevents singular solutions ($Y|W_Y$ modal at $Y = (y \cdots y)^\top$)

PCA as Graph Coupling

Theorem (Van Assel et al. 2022)

For $\nu \geq n$, let $\Theta_X \sim \text{Wishart}(\nu, I_n)$ and $\Theta_Y \sim \text{Wishart}(\nu + p - d, I_n)$ be random precision matrices. Assume that

$$X|\Theta_X \sim \mathcal{N}(0, \Theta_X^{-1} \otimes I_p)$$
$$Y|\Theta_Y \sim \mathcal{N}(0, \Theta_Y^{-1} \otimes I_d)$$

Then the solution of the precision coupling problem:

$$\min_{Y \in \mathbb{R}^{n \times d}} -\mathbb{E}_{\Theta_X|X} \left[\log \mathbb{P}(\Theta_X = \Theta_Y|Y)\right]$$

is a PCA embedding of X with d components.

Van Assel et al. 2022 define a hierarchical graph model to capture both global (PCA) and local (SNE) structures.

References

Arias-Castro, Ery and Bruno Pelletier (2013). **"On the convergence of maximum variance unfolding".** In: *The journal of machine learning research* 14.1, pp. 1747–1770.

Belkin, M. and P. Niyogi (2003). **"Laplacian eigenmaps for dimensionality reduction and data representation".** In: *Neural computation* 15.16, pp. 1373–1396.

Belkin, Mikhail and Partha Niyogi (2006). **"Convergence of laplacian eigenmaps".** In: *Advances in neural information processing systems* 19.

Blanchard, Gilles, Olivier Bousquet, and Laurent Zwald (2007). **"Statistical properties of kernel principal component analysis".** In: *Machine learning* 66.2, pp. 259–294.

Budninskiy, Max, Gloria Yin, Leman Feng, Yiying Tong, and Mathieu Desbrun (2019). **"Parallel transport unfolding: a connection-based manifold learning approach".** In: *Siam journal on applied algebra and geometry* 3.2, pp. 266–291.

Coifman, R.R. and S. Lafon (2006). **"Diffusion maps".** In: *Applied and computational harmonic analysis* 21.1, pp. 5–30.

Damrich, Sebastian and Fred A Hamprecht (2021). **"On umap's true loss function".** In: *Advances in neural information processing systems* 34, pp. 5798–5809.

Donoho, D.L. and C. Grimes (2003). **"Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data".** In: *P. natl. acad. sci. usa* 100.10, pp. 5591–5596.

García Trillos, Nicolás, Moritz Gerlach, Matthias Hein, and Dejan Slepčev (2020). **"Error estimates for spectral convergence of the graph laplacian on random geometric graphs toward the laplace–beltrami operator".** In: *Foundations of computational mathematics* 20.4, pp. 827–887.

Grigor'yan, Alexander, Jiaxin Hu, and Ka-Sing Lau (2014). **"Heat kernels on metric measure spaces".** In: *Geometry and analysis of fractals*. Springer, pp. 147–207.

Hall, Kenneth M (1970). **"An r-dimensional quadratic placement algorithm".** In: *Management science* 17.3, pp. 219–229.

Hinton, Geoffrey E and Sam Roweis (2002). **"Stochastic neighbor embedding".** In: *Advances in neural information processing systems* 15.

Klimenta, Mirza (2012). **"Extending the usability of multidimensional scaling for graph drawing".** PhD thesis. Universität Konstanz.

McInnes, Leland, John Healy, and James Melville (2018). **"Umap: uniform manifold approximation and projection for dimension reduction".** In: *Arxiv preprint arxiv:1802.03426*.

Mika, Sebastian, Bernhard Schölkopf, Alex Smola, Klaus-Robert Müller, Matthias Scholz, and Gunnar Rätsch (1998). **"Kernel pca and de-noising in feature spaces".** In: *Advances in neural information processing systems* 11.

Paprotny, Alexander and Jochen Garcke (2012). **"On a connection between maximum variance unfolding, shortest path problems and isomap".** In: *Artificial intelligence and statistics*, pp. 859–867.

Schölkopf, Bernhard, Alexander Smola, and Klaus-Robert Müller (1998). **"Nonlinear component analysis as a kernel eigenvalue problem".** In: *Neural computation* 10.5, pp. 1299–1319.

Tang, Jian, Jingzhou Liu, Ming Zhang, and Qiaozhu Mei (2016). **"Visualizing large-scale and high-dimensional data".** In: *Proceedings of the 25th international conference on world wide web*, pp. 287–297.

Van Assel, Hugues, Thibault Espinasse, Julien Chiquet, and Franck Picard (2022). **"A probabilistic graph coupling view of dimension reduction".** In: *Arxiv preprint arxiv:2201.13053*.

Van Der Maaten, Laurens (2009). **"Learning a parametric embedding by preserving local structure".** In: *Artificial intelligence and statistics*. PMLR, pp. 384–391.

Wahl, Martin (2024). **A kernel-based analysis of laplacian eigenmaps.** arXiv: 2402.16481 [math.ST]. URL: https://arxiv.org/abs/2402.16481.

Weinberger, Kilian Q and Lawrence K Saul (2006). **"Unsupervised learning of image manifolds by semidefinite programming".** In: *International journal of computer vision* 70.1, pp. 77–90.

Weinberger, K.Q., F. Sha, and L.K. Saul (2004). **"Learning a kernel matrix for nonlinear dimensionality reduction".** In: *International confernence on machine learning (icml)*, p. 106.

Zhang, Zhenyue and Hongyuan Zha (2004). **"Principal manifolds and nonlinear dimensionality reduction via tangent space alignment".** In: *Siam journal on scientific computing* 26.1, pp. 313–338.