Non-Isometric Manifold Learning

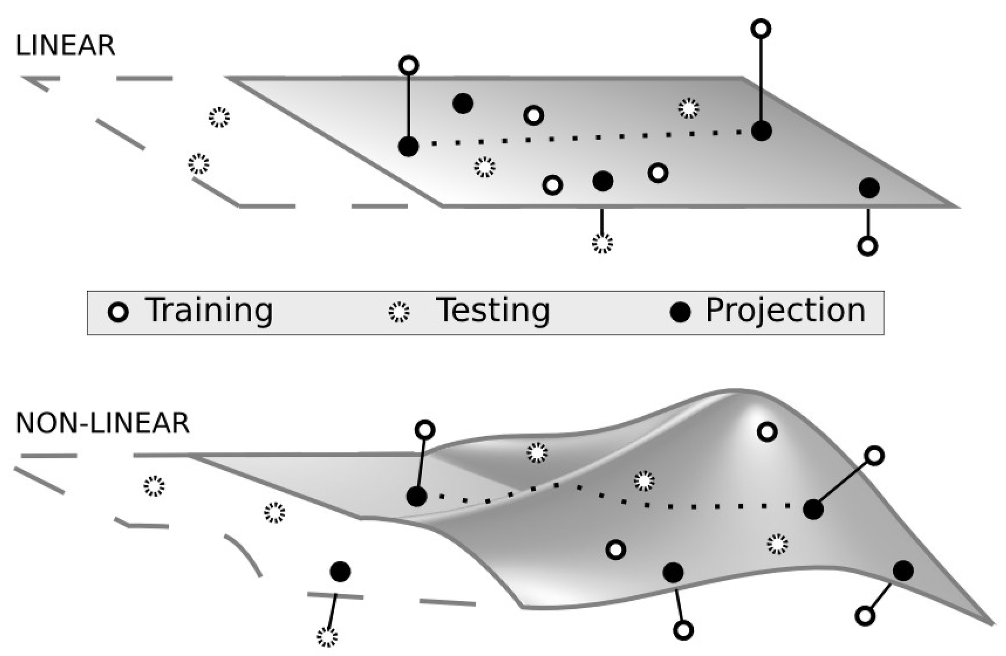
Analysis and an Algorithm

Piotr Dollár
Computer Science and Engineering
University of California, San Diego
pdollar@cs.ucsd.edu

Vincent Rabaud
Computer Science and Engineering
University of California, San Diego
vrabaud@cs.ucsd.edu

Serge Belongie
Computer Science and Engineering
University of California, San Diego
sjb@cs.ucsd.edu

Introduction



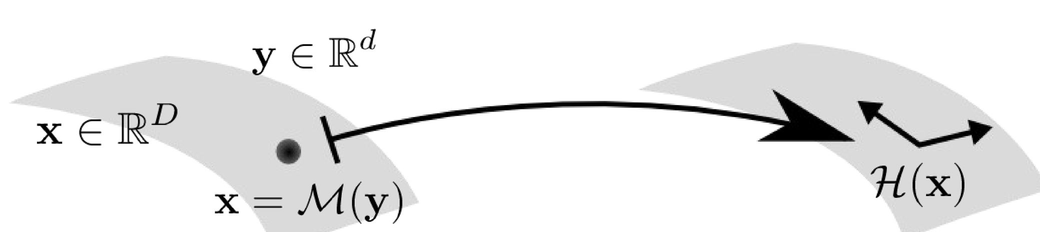
Typical operations on a linear subspace (PCA):
- Project onto subspace
- Distance between point and subspace
- Distance between projected points
- Predict structure of space where no data given

Extend above to nonlinear manifolds (LSML):
- Need new manifold representation
- Deal with non-isometric manifolds
- *Generalize* to unseen portions of manifold

Goal of LSML

Problem Formulation

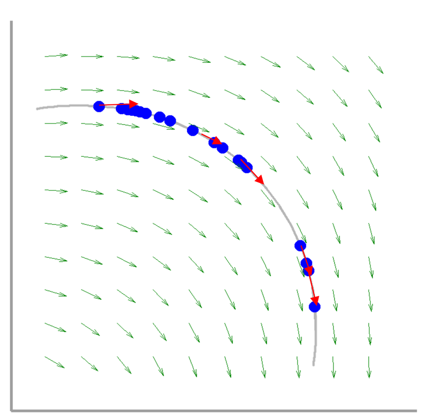
- *d*-dimensional manifold(s) in *D*-dimensional space
- Learn a mapping from a point to its tangent basis
- LSML = locally smooth manifold learning



$$\mathcal{H}: \begin{cases} \mathbb{R}^D & \longrightarrow \quad \mathbb{R}^{D\times d} \\ \mathbf{x} & \longmapsto \quad \left[\frac{\partial}{\partial \mathbf{y}_1}\mathcal{M}(\mathbf{y}) \quad \cdots \quad \frac{\partial}{\partial \mathbf{y}_d}\mathcal{M}(\mathbf{y})\right] \end{cases}$$

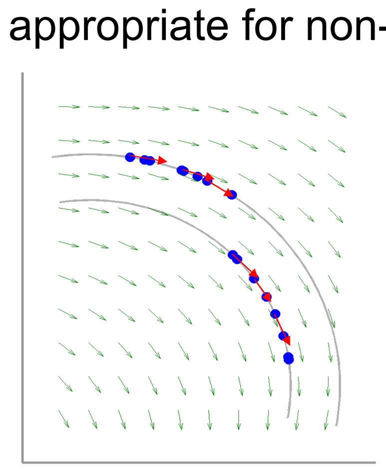
Intuition

- Suppose *training* tangent vectors are given:
- Could *learn* a regression from coordinates to the *d* tangent vectors
- Could then *test* regression in other places on manifold:



Benefits

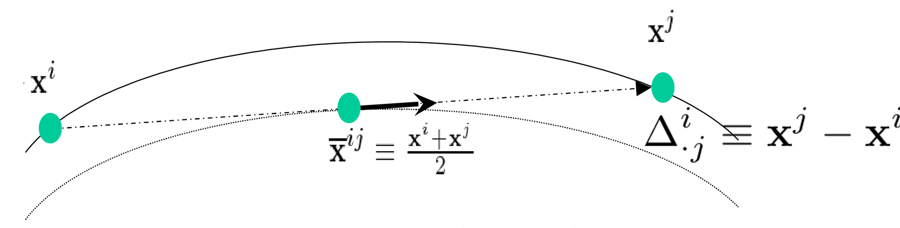
- Representation learned by LSML (tangent space) appropriate for non-isometric manifolds



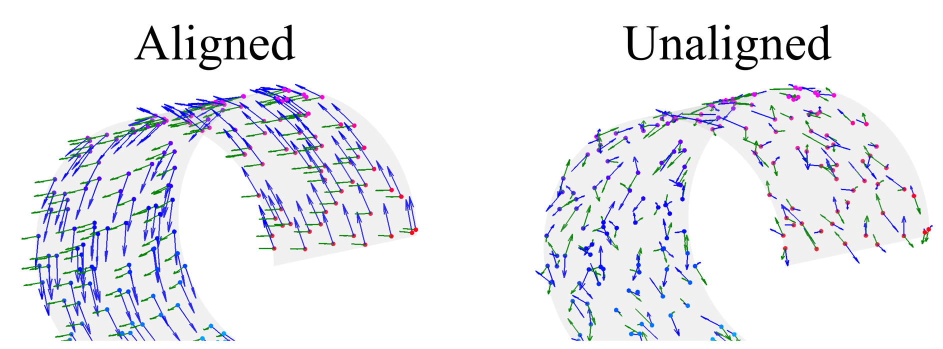
- Learns a mapping over $\mathbb{R}^D$. Can be applied beyond support of original data (*generalization*).
- Can also learn from multiple manifolds, generalize to new manifolds: *manifold transfer*

Error Function

$\mathbf{x}^i$	$[D \times 1]$	$i \in [n]$, data point
$\overline{\mathbf{x}}^{ij}$	$[D \times 1]$	$\overline{\mathbf{x}}^{ij} = \frac{\mathbf{x}^i + \mathbf{x}^j}{2}$
$\mathbf{f}^{ij}$	$[f \times 1]$	features of $\overline{\mathbf{x}}^{ij}$
$\mathcal{N}^i$		indices of neighbors of $\mathbf{x}^i$
$\mathcal{H}_\theta$		$\mathcal{H}_\theta : \mathbb{R}^D \to \mathbb{R}^{D\times d}$
H^{ij}	$[D \times d]$	$H^{ij} = \mathcal{H}_\theta(\overline{\mathbf{x}}^{ij})$ (for θ fixed)
$\boldsymbol{\epsilon}^{ij}$	$[d \times 1]$	alignment free parameter



$$\mathcal{H}_\theta(\overline{\mathbf{x}}^{ij})\boldsymbol{\epsilon}^{ij} \approx \Delta^i_{\cdot j}$$



$$\mathrm{err}(\theta) = \min_{\{\boldsymbol{\epsilon}^{ij}\}} \sum_{i,j \in \mathcal{N}^i} \left\| \mathcal{H}_\theta(\overline{\mathbf{x}}^{ij})\boldsymbol{\epsilon}^{ij} - \Delta^i_{\cdot j} \right\|_2^2$$

Add regularization terms:

$$+\lambda_E \sum \left\|\boldsymbol{\epsilon}^{ij}\right\|_2^2 + \lambda_\theta \sum \left\| \mathcal{H}_\theta(\overline{\mathbf{x}}^{ij}) - \mathcal{H}_\theta(\overline{\mathbf{x}}^{ij'}) \right\|_F^2$$

Optimization Procedure

Linear form: $\mathcal{H}_\theta(\overline{\mathbf{x}}^{ij}) = \left[\Theta^1 \mathbf{f}^{ij} \cdots \Theta^D \mathbf{f}^{ij}\right]^\top$

$$\mathrm{err}(\theta) = \min_{\{\boldsymbol{\epsilon}^{ij}\}} \sum_{i,j \in \mathcal{N}^i} \sum_{k=1}^{D} \left(\mathbf{f}^{ij\top} \Theta^{k\top} \boldsymbol{\epsilon}^{ij} - \Delta^i_{kj} \right)^2$$

Initialize Θ randomly.
while $\mathrm{err}(\Theta)$ decreases **do**
$\forall i, j$, solve for the best $\boldsymbol{\epsilon}^{ij}$ given the Θ^ks:

$$\boldsymbol{\epsilon}^{ij} = (H^{ij\top} H^{ij} + \lambda_\epsilon I)^{+} H^{ij\top} \Delta^i_{\cdot j}$$

$\forall k$, solve for the best Θ^k given the $\boldsymbol{\epsilon}^{ij}$'s:

$$\text{Let: } A = \begin{bmatrix} \vdots \\ \boldsymbol{\epsilon}^{ij\top} \otimes \mathbf{f}^{ij\top} \\ \vdots \end{bmatrix}, \mathbf{b}^k = \begin{bmatrix} \vdots \\ \Delta^i_{kj} \\ \vdots \end{bmatrix}$$

$$\mathrm{vec}\left(\Theta^{k\top}\right) = (A^\top A + \lambda_\Theta (I \otimes (\Delta_F{}^\top \Delta_F)))^{-1} A^\top \mathbf{b}^k$$

end while

Analyzing Manifold Learning Methods

Need evaluation methodology:
- To objectively compare methods
- To extend to non-isometric manifolds

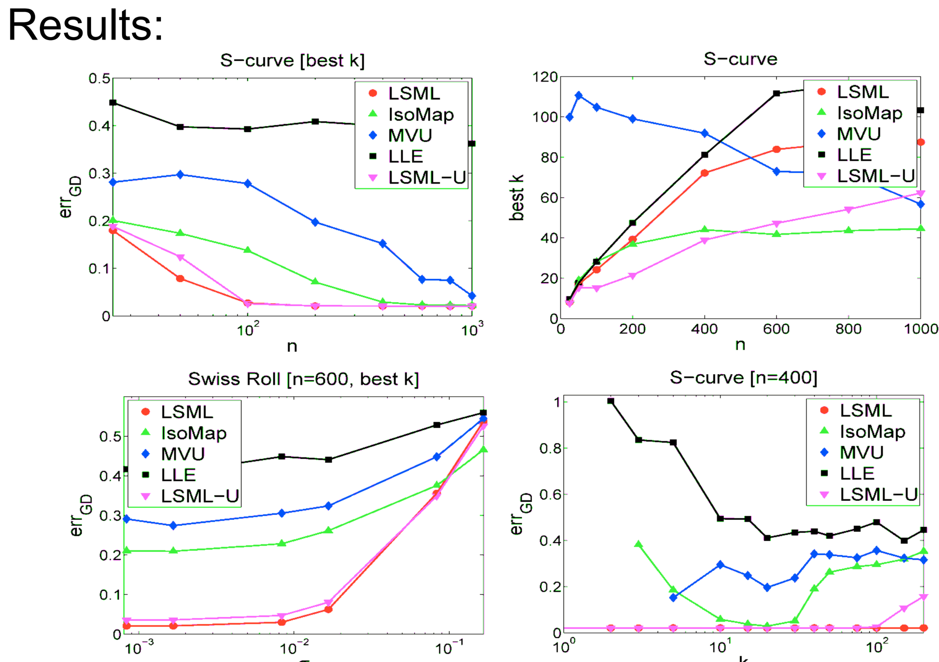
By definition, for isometric manifolds embedding should preserve distance

$$\mathrm{err}_{\mathrm{GD}} \equiv \frac{1}{n^2} \sum_{ij} \frac{|d_{ij} - d'_{ij}|}{d_{ij}}$$

(d'_{ij}: Estimated dist; d_{ij}: True distance)

Requires two sets of samples from manifold:
1. S_n – for training
2. S_∞ – for computing "true" geodesics (Isomap)

Results:



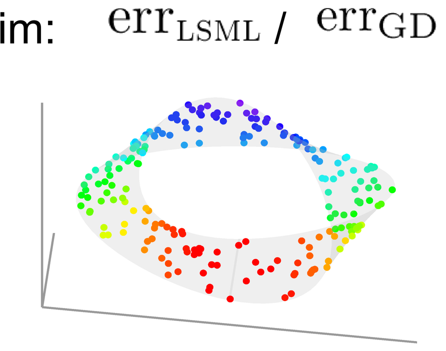
- Finite Sample performance: LSML > ISOMAP > MVU > LLE
- Model Complexity
  - All methods have at least one parameter: k
  - Bias-Variance tradeoff
- Similar for non-isometric (LSML / Isomap only)

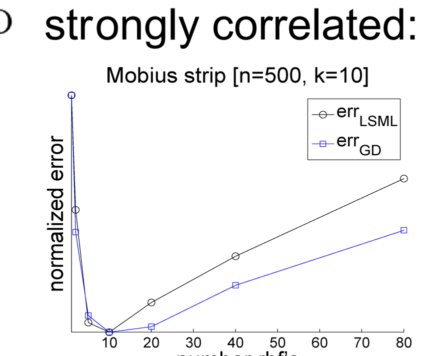
LSML Test Error

Typically, S_∞ not available
- Need notion of generalization: **testable prediction**
- Model assessment / Model selection

Define: $\mathrm{err}_{\mathrm{LSML}} \equiv \sum_i \min_{\boldsymbol{\epsilon}^{ii'}} \left\| \mathcal{H}_\theta(\overline{\mathbf{x}}^{ii'})\boldsymbol{\epsilon}^{ii'} - (\mathbf{x}^i - \mathbf{x}^{i'}) \right\|_2^2$

Claim: $\mathrm{err}_{\mathrm{LSML}}$ / $\mathrm{err}_{\mathrm{GD}}$ strongly correlated:





- Use much as test error in supervised learning
- Cannot be used to select *d*
- Can also measure error for manifold transfer

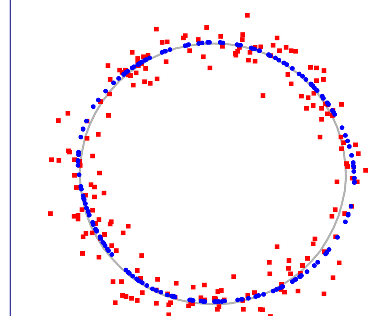
Using the Tangent Space

Projection

A projection x' of x must satisfy: $\min_{\mathbf{x}'} \|\mathbf{x} - \mathbf{x}'\|_2^2$

Perform gradient descent after initializing x' to a close point on manifold: $\mathbf{x}' \leftarrow \mathbf{x}' + \alpha H' H'^\top (\mathbf{x} - \mathbf{x}')$

Manifold De-noising



x^i is noisy and its clean version χ^i needs to be close and to satisfy local linearity assumption:

$$\mathrm{err}_{\mathcal{M}}(\chi) = \min_{\{\boldsymbol{\epsilon}^{ij}\}} \sum_{i,j \in \mathcal{N}^i} \|\mathcal{H}_\theta(\overline{\chi}^{ij})\boldsymbol{\epsilon}^{ij} - (\chi^i - \chi^j)\|_2^2$$

$$\mathrm{err}_{\mathrm{orig}}(\chi) = \sum_{i=1}^{n} \|\chi^i - \mathbf{x}^i\|_2^2$$

Apply gradient descent to minimize overall error:

$$\frac{\partial \mathrm{err}_{\mathcal{M}}}{\partial \chi^i}(\chi) = 4 \sum_{j \in \mathcal{N}^i} B^{ij\top} B^{ij} (\chi^i - \chi^j)$$

$$\frac{\partial \mathrm{err}_{\mathrm{orig}}(\chi)}{\partial \chi^i} = 2(\chi^i - \mathbf{x}^i)$$

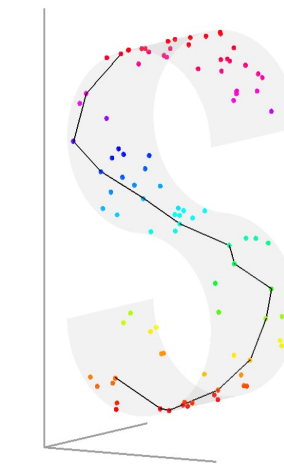
Geodesic Distance

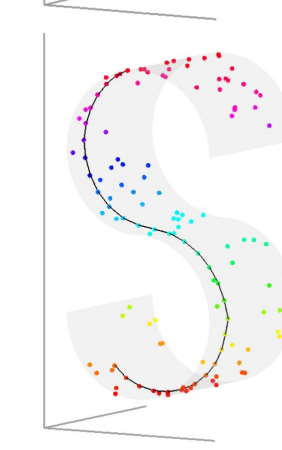
Is the length of an optimal path χ^i between two points, with local tangents on the manifold.

$$\mathrm{err}_{\mathrm{length}}(\chi) = \sum_{i=2}^{m} \|\chi^i - \chi^{i-1}\|_2^2$$

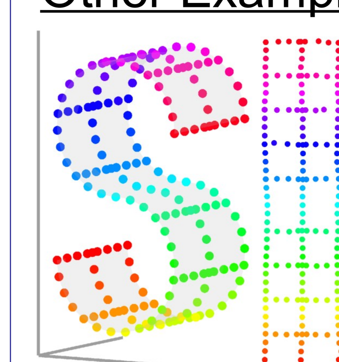
Alternative optimization of above + $\mathrm{err}_{\mathcal{M}}(\chi)$ by projected gradient descent:

$$\frac{\partial \mathrm{err}_{\mathrm{length}}}{\partial \chi^i}(\chi) = 2 \sum_{j = i \pm 1} (\chi^i - \chi^j)$$

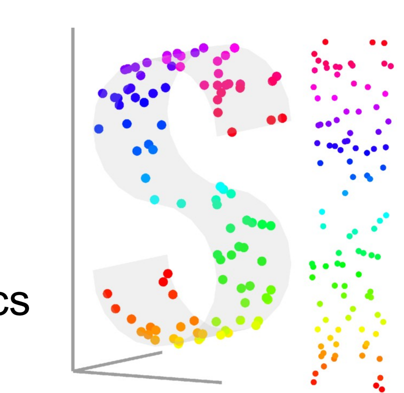


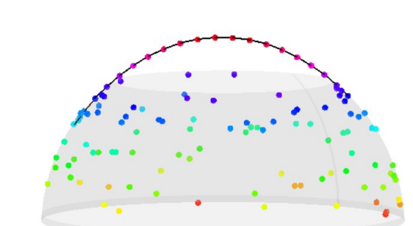


Other Example Uses

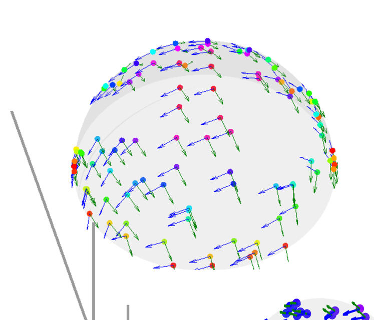


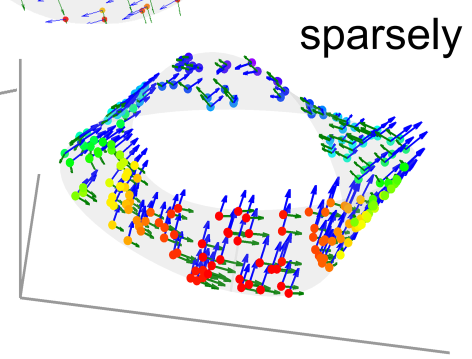
Accurate embedding of structured and sparse data using geodesics





Generalization beyond support of training data





Tangent field learned for non-isometric manifolds; Error low even in sparsely sampled regions