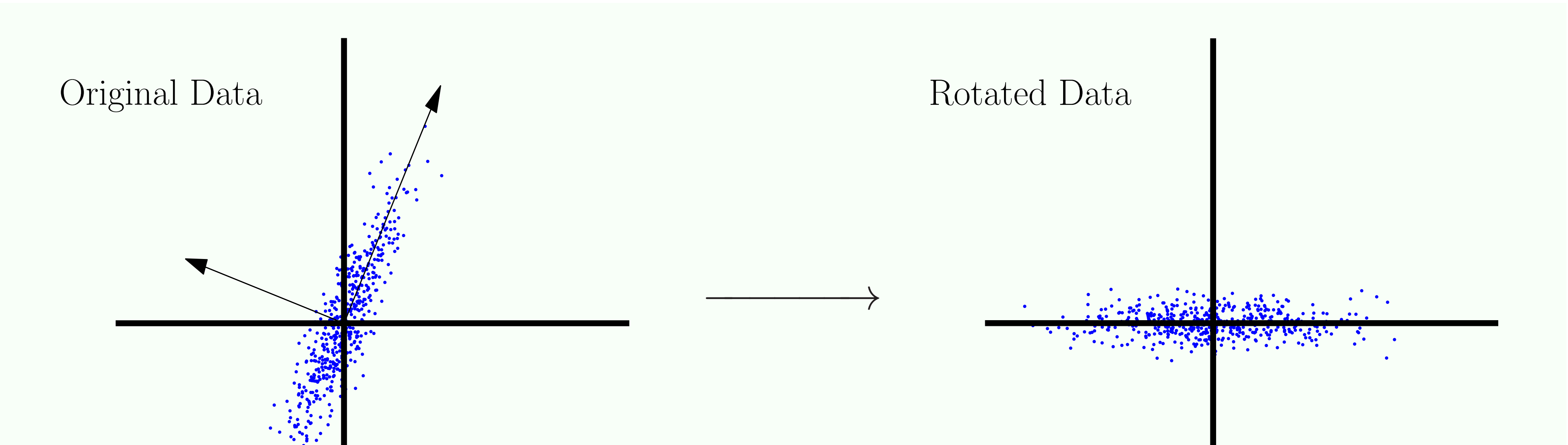
Principal Components Analysis (PCA)

Chapter e-9

Motivation

Principal components: new features constructed as linear combinations of the given features.

Choose combinations of features that achieve high variance.

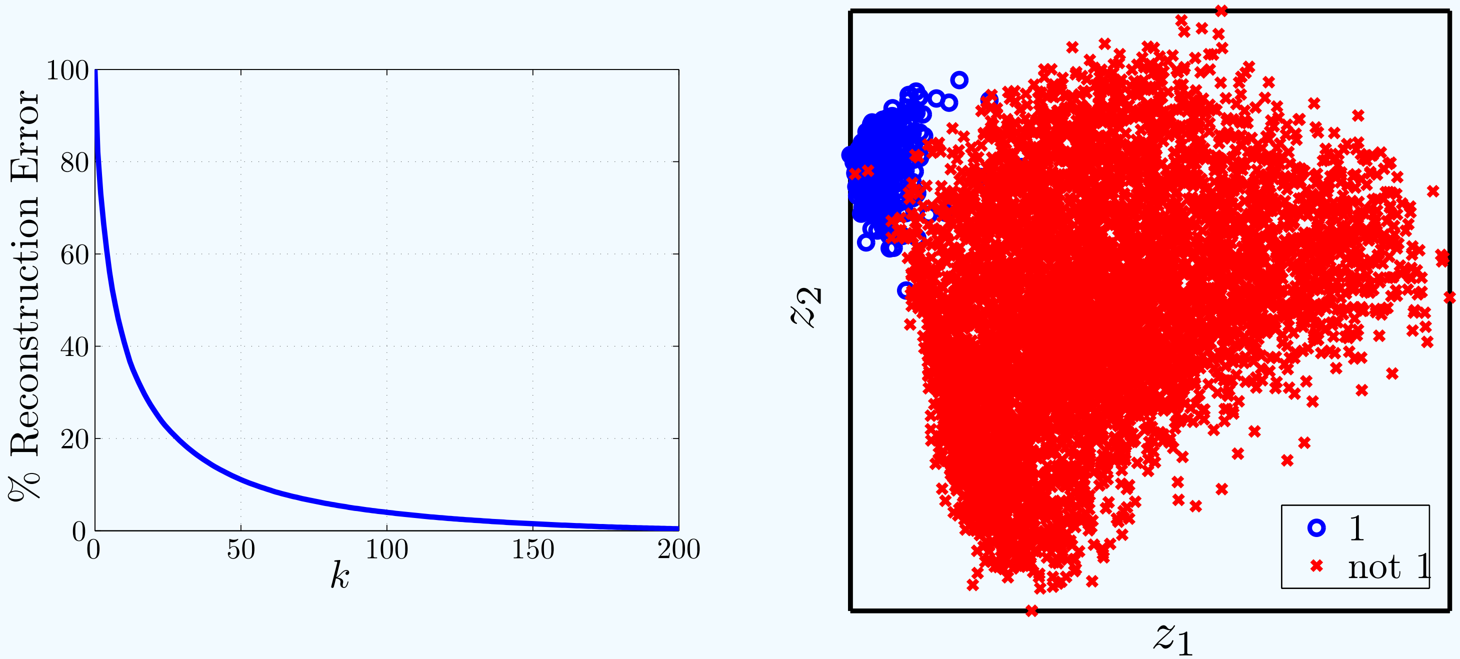


$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \rightarrow \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \rightarrow \begin{bmatrix} z_1 \end{bmatrix}$$

Motivation

Principal components: new features constructed as linear combinations of the given features.

Choose combinations of features that achieve high variance.



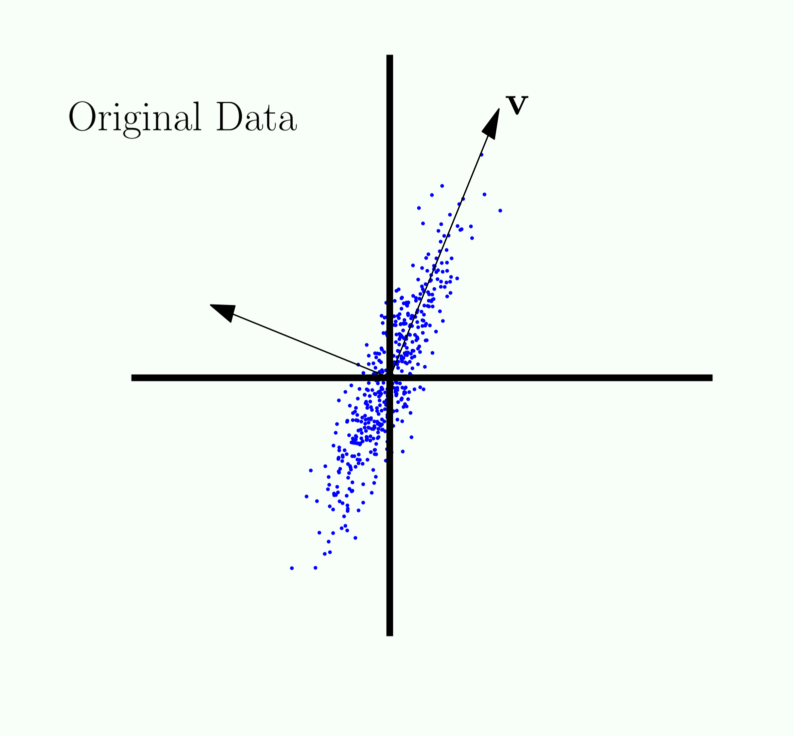
(a) Reconstruction error (b) Top-2 PCA-features

Pre-processing

- Center the data
- Standardize if features have different scales

First principal component

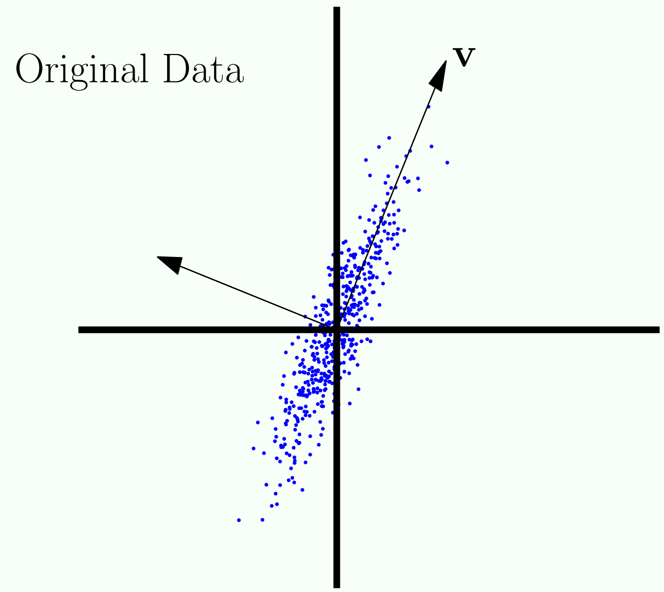
Look for a direction v that maximizes the variance of $z = \mathbf{x}_n^{\mathrm{T}}\mathbf{v}$



First principal component

Look for a direction v that maximizes the variance of $z = \mathbf{x}_n^{\mathrm{T}}\mathbf{v}$

$$\begin{aligned}
\mathrm{var}[z] &= \frac{1}{N}\sum_{n=1}^{N} z_n^2 \\
&= \frac{1}{N}\sum_{n=1}^{N} \mathbf{v}^{\mathrm{T}}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\mathbf{v} \\
&= \mathbf{v}^{\mathrm{T}}\left(\frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\right)\mathbf{v} \\
&= \mathbf{v}^{\mathrm{T}}\Sigma\mathbf{v}.
\end{aligned}$$

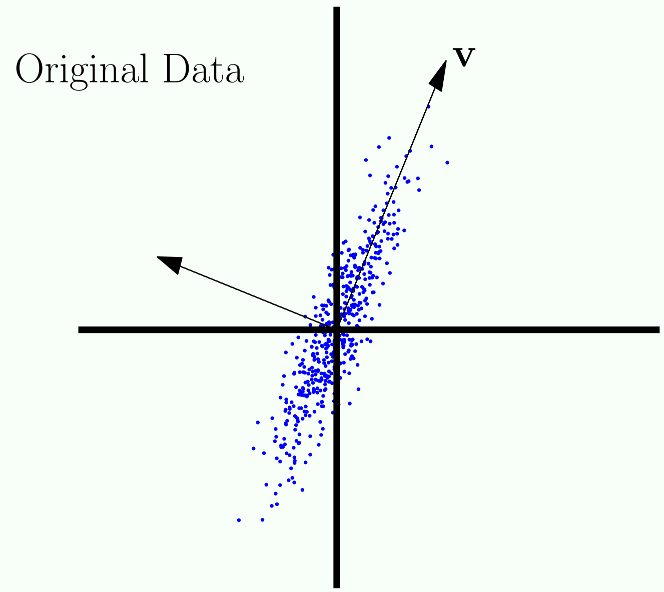


First principal component

Look for a direction v that maximizes the variance of $z = \mathbf{x}_n^{\mathrm{T}}\mathbf{v}$

$$\begin{aligned}
\text{var}[z] &= \frac{1}{N}\sum_{n=1}^{N} z_n^2 \\
&= \frac{1}{N}\sum_{n=1}^{N} \mathbf{v}^{\mathrm{T}}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\mathbf{v} \\
&= \mathbf{v}^{\mathrm{T}}\left(\frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_n\mathbf{x}_n^{\mathrm{T}}\right)\mathbf{v} \\
&= \mathbf{v}^{\mathrm{T}}\Sigma\mathbf{v}.
\end{aligned}$$

covariance matrix



First principal component

Objective:

$$\text{maximize} \quad \mathbf{v}^\intercal \Sigma \mathbf{v}$$

$$\text{such that} \quad \mathbf{v}^\intercal \mathbf{v} = 1$$

Use the Lagrange multipliers method with

$$\Lambda(\mathbf{v}) = \mathbf{v}^\intercal \Sigma \mathbf{v} + \lambda(1 - \mathbf{v}^\intercal \mathbf{v})$$

Taking derivative with respect to **v** and setting to zero gives:

$$\Sigma \mathbf{v} = \lambda \mathbf{v}$$

I.e. **v** is an eigenvector of the covariance matrix.

Which one should we choose?

Principal components

Solution is an eigenvector of the covariance matrix:

$$\Sigma \mathbf{v} = \lambda \mathbf{v}$$

Since the covariance matrix is symmetric, it has real eigenvalues, and the eigenvectors form a basis.

Therefore, if we want k directions, choose the k eigenvectors with the largest eigenvalues.

Coordinate systems

Coordinate system: an orthonormal basis.

Our standard Euclidean basis: $\mathbf{u}_1, \ldots, \mathbf{u}_d$

where u_i is a unit vector with a single non-zero coefficient

Expressing **x** in terms of our basis:

$$\mathbf{x} = \sum_{i=1}^{d} x_i \mathbf{u}_i = \sum_{i=1}^{d} (\mathbf{x}^\mathsf{T} \mathbf{u}_i)\mathbf{u}_i$$

Can do that in terms of the principal components:

$$\begin{aligned} z_1 &= \mathbf{x}^{\mathrm{T}} \mathbf{v}_1 \\ z_2 &= \mathbf{x}^{\mathrm{T}} \mathbf{v}_2 \\ z_3 &= \mathbf{x}^{\mathrm{T}} \mathbf{v}_3 \\ &\vdots \end{aligned}$$

Principal components

To represent $\mathbf{x}_i$ in the basis of the principal components:

$$\mathbf{z}_i = \begin{pmatrix} \mathbf{x}_i^\top \mathbf{v}_1 \\ \vdots \\ \mathbf{x}_i^\top \mathbf{v}_k \end{pmatrix}$$

$\mathbf{z}_i$ is a k-dimensional vector; $\mathbf{x}_i$ is d-dimensional

Alternative interpretation

PCA can also be derived as the basis that minimizes the reconstruction error arising from projecting the data onto a k-dimensional subspace.

Reconstruction error

If we kept all the PCs:

$$\mathbf{x} = \sum_{i=1}^{d} z_i \mathbf{v}_i$$

The reconstructed vector using k PCs:

$$\hat{\mathbf{x}} = \sum_{i=1}^{k} z_i \mathbf{v}_i$$

The reconstruction error:

$$||\mathbf{x} - \hat{\mathbf{x}}||^2 = ||\sum_{i=k+1}^{d} z_i \mathbf{v}_i||^2 = \sum_{i=k+1}^{d} z_i^2$$

Computing principal components using SVD

SVD: singular value decomposition

Any n x d matrix X can be expressed as:

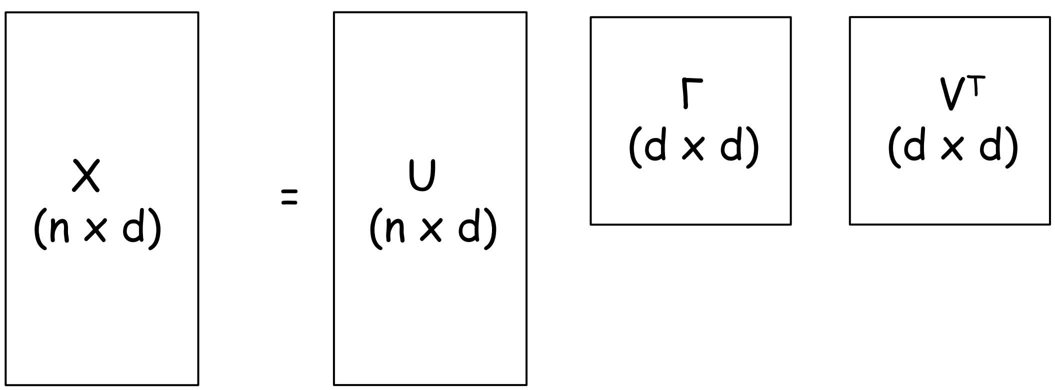
$$X = U \Gamma V^{\top}$$

Where:

U : n x d matrix (orthonormal columns)

V : d x d matrix (orthonormal columns)

Γ : d x d matrix (diagonal)



Computing principal components using SVD

SVD: singular value decomposition

Any n x d matrix X can be expressed as:

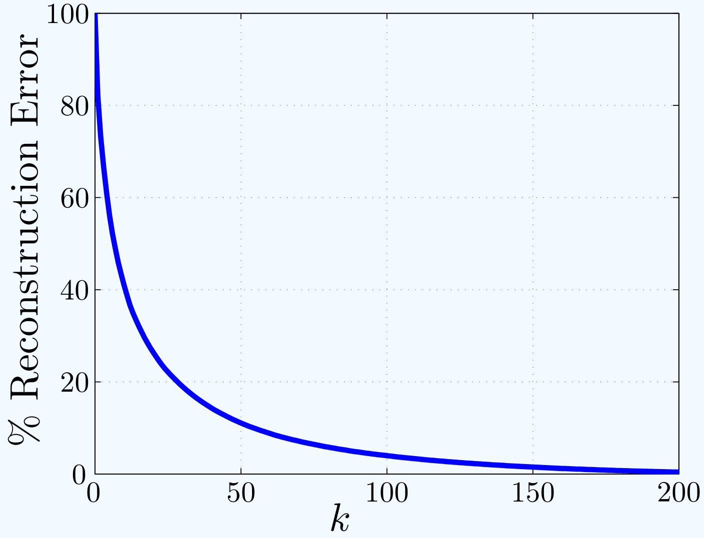
$$X = U\Gamma V^\intercal$$

Relationship to the scatter/covariance matrix:

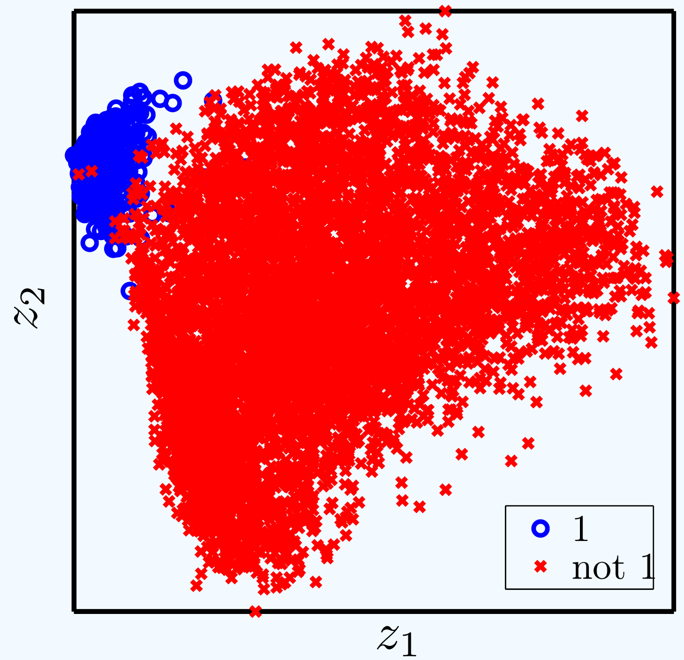
$$\Sigma = X^\intercal X = (U\Gamma V^\intercal)^\intercal (U\Gamma V^\intercal)$$
$$= (V\Gamma U^\intercal)^\intercal U\Gamma V^\intercal = V\Gamma^2 V^\intercal$$

I.e., the matrix V we obtain from SVD is the matrix of eigenvectors of the covariance matrix.

PCA for the digits data



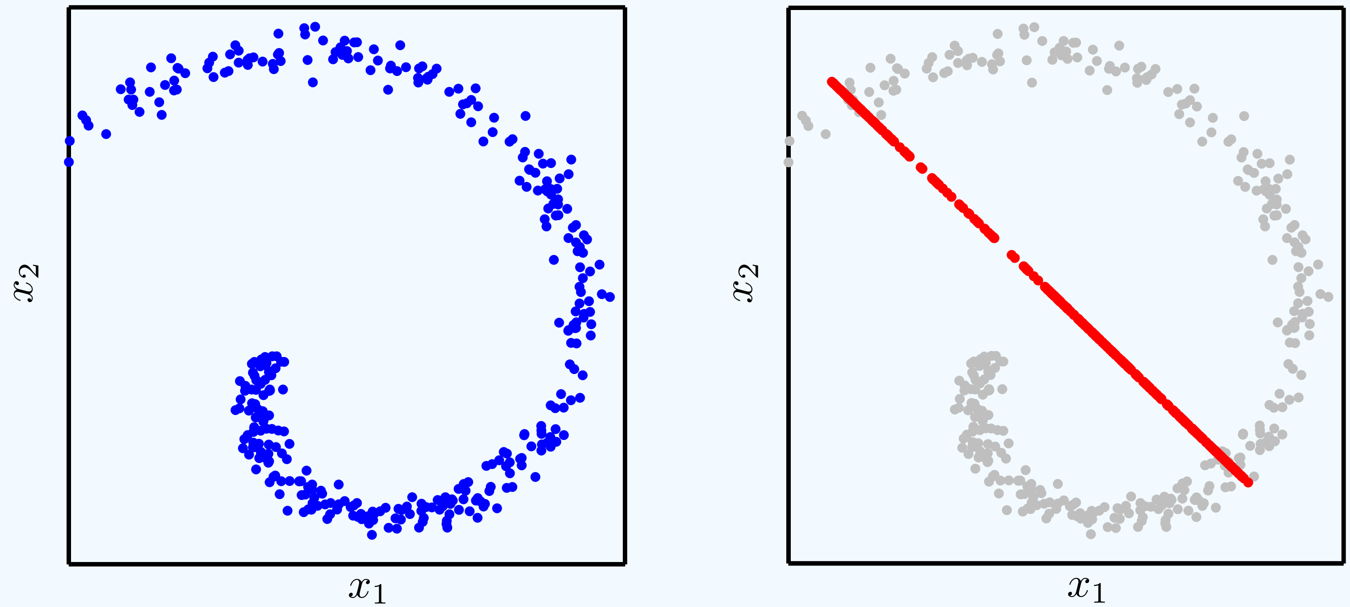
(a) Reconstruction error



(b) Top-2 PCA-features

How to choose the number of principal components?

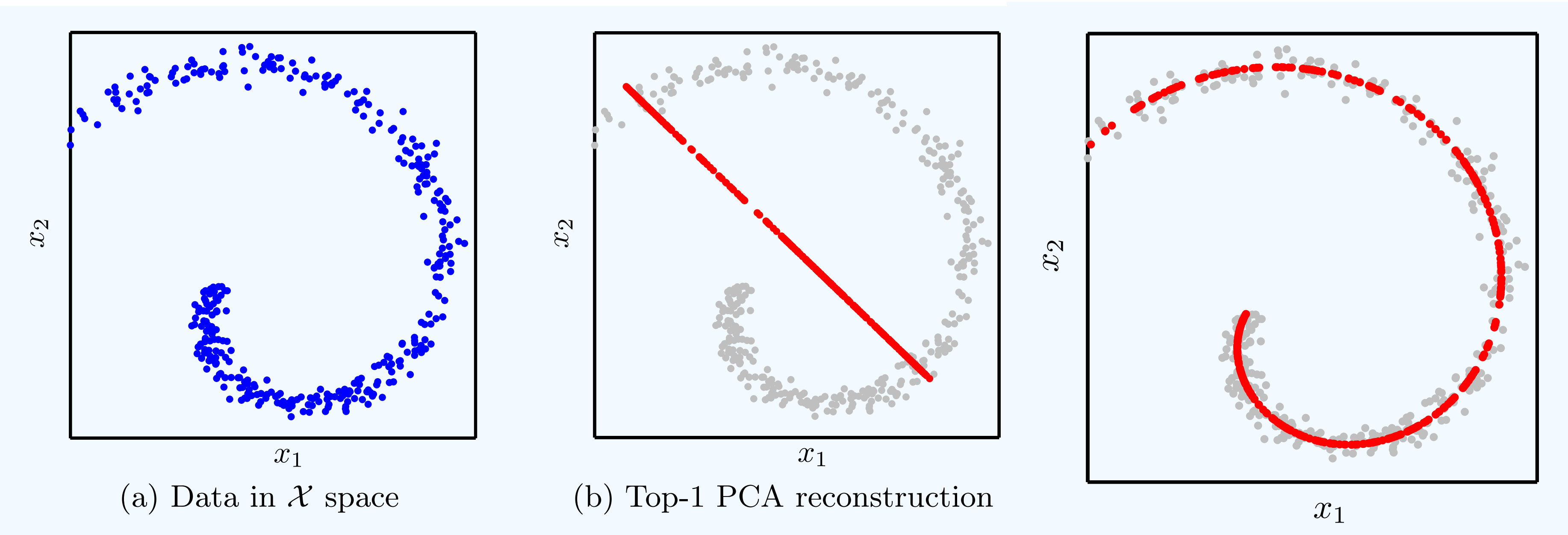
Nonlinear PCA



(a) Data in $\mathcal{X}$ space (b) Top-1 PCA reconstruction

Linear PCA won't work here.

Nonlinear PCA



(a) Data in $\mathcal{X}$ space

(b) Top-1 PCA reconstruction

Nonlinear PCA can be performed in kernel space

Applications of PCA

- Data visualization
- Data compression
- Dimensionality reduction before applying other forms of learning
- Can be viewed as performing noise-reduction

Domains where it is commonly used:

- Face recognition (eigenfaces)
- Text categorization (LSA)

Face Representation using PCA

Input face

EigenFaces

56.4 38.6 -19.7 9.8 -45.9 19.6 - 14.2 ...

PCA

Reconstructed face

Minimize reconstruction error

Comments about PCA

- One of the most widely used techniques for data analysis
- The basis for "latent semantic analysis" for representing text.

- There are many other dimensionality reduction techniques:
  - Canonical correlation analysis (CCA)
  - Independent component analysis (ICA)
  - Non-negative matrix factorization (NMF)
  - Autoencoders

PCA in Python

```
X = X - np.mean(X, axis=0)
[u,s,v] = numpy.linalg.svd(X)
v = v.transpose()
v = v[:,:numcomp]
return numpy.dot(X, v)
```