Foundations of Machine Learning
African Masters in Machine Intelligence

Principal Component Analysis

Marc Deisenroth

Quantum Leap Africa
African Institute for Mathematical Sciences, Rwanda

Department of Computing
Imperial College London

@mpd37
mdeisenroth@aimsammi.org

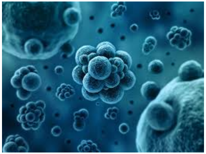
October 4, 2018

References

- Bishop: Pattern Recognition and Machine Learning, Chapter 12
- Deisenroth et al.: Mathematics for Machine Learning, Chapter 10 (`https://mml-book.com`)

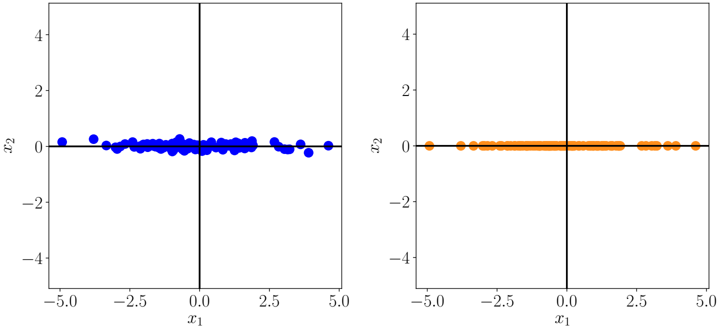
Overview

High-Dimensional Data

- Real-world data is often high dimensional
- **Challenges:**
  - Difficult to analyze
  - Difficult to visualize
  - Difficult to interpret

Properties of High-dimensional Data



- Many dimensions are unnecessary
- Data often lives on a low-dimensional manifold

▶▶ Dimensionality reduction finds the relevant dimensions.

Background: Coordinate Representations

Consider $\mathbb{R}^2$ with the canonical basis $\boldsymbol{e}_1 = [1, 0]^\top$, $\boldsymbol{e}_2 = [0, 1]^\top$.

$$\boldsymbol{x} = \begin{bmatrix} 5 \\ 3 \end{bmatrix} = 5\boldsymbol{e}_1 + 3\boldsymbol{e}_2 \qquad \text{Linear combination of basis vectors}$$

- **Coordinates** of $\boldsymbol{x}$ w.r.t. $(\boldsymbol{e}_1, \boldsymbol{e}_2)$: $[5, 3]$

Background: Coordinate Representations

Consider $\mathbb{R}^2$ with the canonical basis $\boldsymbol{e}_1 = [1, 0]^\top$, $\boldsymbol{e}_2 = [0, 1]^\top$.

$$\boldsymbol{x} = \begin{bmatrix} 5 \\ 3 \end{bmatrix} = 5\boldsymbol{e}_1 + 3\boldsymbol{e}_2 \qquad \text{Linear combination of basis vectors}$$

- **Coordinates** of $\boldsymbol{x}$ w.r.t. $(\boldsymbol{e}_1, \boldsymbol{e}_2)$: $[5, 3]$

Consider the vectors of the form

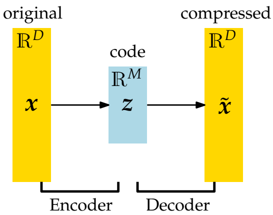
$$\tilde{\boldsymbol{x}} = \begin{bmatrix} 0 \\ z \end{bmatrix} \in \mathbb{R}^2, \quad z \in \mathbb{R}$$

▸▸ Write them as $0\boldsymbol{e}_1 + z\boldsymbol{e}_2$.

- Only remember/store the **coordinate/code** z of the $\boldsymbol{e}_2$ vector
  ▸▸ **Compression**
- Set of $\tilde{\boldsymbol{x}}$ vectors forms a vector subspace $U \subseteq \mathbb{R}^2$ with $\dim(U) = 1$ because $U = \text{span}[\boldsymbol{e}_2]$.

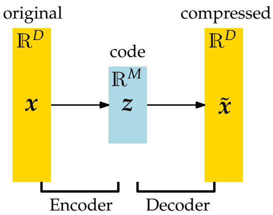
Overview

PCA Setting



- Dataset $\mathcal{X} := \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_N\}$, $\boldsymbol{x}_n \in \mathbb{R}^D$
- Data matrix $\boldsymbol{X} := [\boldsymbol{x}_1, \dots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ ▸▸ Often $N \times D$ matrix

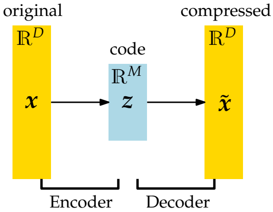
PCA Setting



- Dataset $\mathcal{X} := \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, $\boldsymbol{x}_n \in \mathbb{R}^D$
- Data matrix $\boldsymbol{X} := [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ ⏩ Often $N \times D$ matrix
- Without loss of generality: $\mathbb{E}[\mathcal{X}] = \boldsymbol{0}$ ⏩ Centered data

  ⏩ Data covariance matrix

PCA Setting



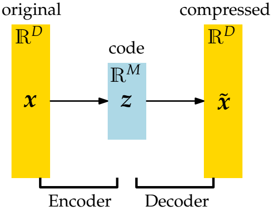
- Dataset $\mathcal{X} := \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$, $\boldsymbol{x}_n \in \mathbb{R}^D$
- Data matrix $\boldsymbol{X} := [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N] \in \mathbb{R}^{D \times N}$ ▶▶ Often $N \times D$ matrix
- Without loss of generality: $\mathbb{E}[\mathcal{X}] = \boldsymbol{0}$ ▶▶ Centered data

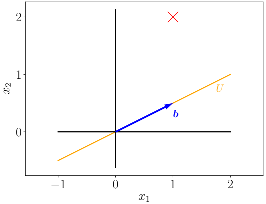
  ▶▶ Data covariance matrix $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top \in \mathbb{R}^{D \times D}$
- Linear relationships between latent code $\boldsymbol{z}$ and data $\boldsymbol{x}$:

$$\boldsymbol{z} = \boldsymbol{B}^\top \boldsymbol{x}\,, \quad \tilde{\boldsymbol{x}} = \boldsymbol{B}\boldsymbol{z}$$

- $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M] \in \mathbb{R}^{D \times M}$ is an orthogonal matrix

Low-Dimensional Embedding

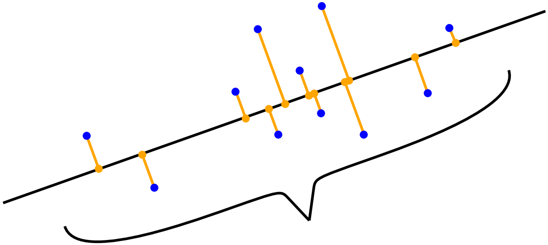




- Find an M-dimensional subspace $U \subset \mathbb{R}^D$ onto which we project the data
- $\tilde{\boldsymbol{x}} = \pi_U(\boldsymbol{x})$ is the projection of $\boldsymbol{x}$ onto U
- Find projections $\tilde{\boldsymbol{x}}$ that are as similar to $\boldsymbol{x}$ as possible
  - Find basis vectors $\boldsymbol{b}_1, \dots, \boldsymbol{b}_M$
- Compression loss incurs if $M \ll D$

Overview

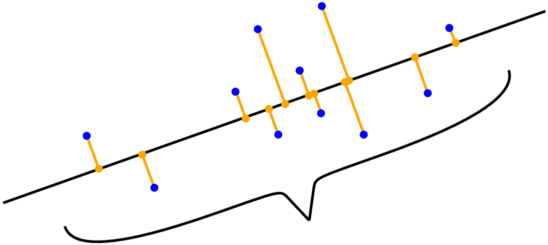
PCA Idea: Maximum Variance



- Project D-dimensional data $\boldsymbol{x}$ onto an M-dimensional subspace that retains as much information as possible

  ▶▶ Data compression

PCA Idea: Maximum Variance



- Project D-dimensional data $\boldsymbol{x}$ onto an M-dimensional subspace that retains as much information as possible
  - ▶▶ Data compression
- Informally: information = diversity = variance
  - ▶▶ **Maximize variance in projected space** (Hotelling 1933)

PCA Objective: Maximum Variance

- Linear relationships:

$$\boldsymbol{z} = \boldsymbol{B}^\top \boldsymbol{x}, \quad \tilde{\boldsymbol{x}} = \boldsymbol{B}\boldsymbol{z}$$

- $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M] \in \mathbb{R}^{D \times M}$ is an orthogonal matrix
- Columns of $\boldsymbol{B}$ are an ONB of an M-dimensional subspace of $\mathbb{R}^D$

PCA Objective: Maximum Variance

- Linear relationships:

$$\boldsymbol{z} = \boldsymbol{B}^\top \boldsymbol{x}\,, \quad \tilde{\boldsymbol{x}} = \boldsymbol{B}\boldsymbol{z}$$

- $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M] \in \mathbb{R}^{D\times M}$ is an orthogonal matrix
- Columns of $\boldsymbol{B}$ are an ONB of an M-dimensional subspace of $\mathbb{R}^D$
- Find $\boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M]$ so that the variance in the projected space is maximized

$$\max_{\boldsymbol{b}_1,\ldots,\boldsymbol{b}_M} \mathbb{V}[\boldsymbol{z}] = \max_{\boldsymbol{b}_1,\ldots,\boldsymbol{b}_M} \mathbb{V}[\boldsymbol{B}^\top \boldsymbol{x}]$$

$$\text{s.t. } \|\boldsymbol{b}_1\| = 1 = \ldots = \|\boldsymbol{b}_M\|$$

▸▸ Constrained optimization problem

Direction with Maximal Variance (1)

- Maximize variance of first coordinate of $\boldsymbol{z} \in \mathbb{R}^M$:

$$V_1 := \mathbb{V}[z_1] = \frac{1}{N}\sum_{n=1}^{N} z_{n1}^2$$

  ▸▸ Empirical variance of the training dataset

Direction with Maximal Variance (1)

- Maximize variance of first coordinate of $\boldsymbol{z} \in \mathbb{R}^M$:

$$V_1 := \mathbb{V}[z_1] = \frac{1}{N}\sum_{n=1}^{N} z_{n1}^2$$

  ▶▶ Empirical variance of the training dataset

- First coordinate of $\boldsymbol{z}_n$ is

$$z_{n1} = \boldsymbol{b}_1^\top \boldsymbol{x}_n$$

  ▶▶ Coordinate of orthogonal projection of $\boldsymbol{x}_n$ onto span$[\boldsymbol{b}_1]$ (1-dimensional subspace spanned by $\boldsymbol{b}_1$)

Direction with Maximal Variance (1)

- Maximize variance of first coordinate of $\boldsymbol{z} \in \mathbb{R}^M$:

$$V_1 := \mathbb{V}[z_1] = \frac{1}{N}\sum_{n=1}^{N} z_{n1}^2$$

▸▸ Empirical variance of the training dataset

- First coordinate of $\boldsymbol{z}_n$ is

$$z_{n1} = \boldsymbol{b}_1^\top \boldsymbol{x}_n$$

▸▸ Coordinate of orthogonal projection of $\boldsymbol{x}_n$ onto span$[\boldsymbol{b}_1]$ (1-dimensional subspace spanned by $\boldsymbol{b}_1$)

$$\mathbb{V}[z_1] =$$

Direction with Maximal Variance (1)

- Maximize variance of first coordinate of $\boldsymbol{z} \in \mathbb{R}^M$:

$$V_1 := \mathbb{V}[z_1] = \frac{1}{N}\sum_{n=1}^{N} z_{n1}^2$$

  ▶▶ Empirical variance of the training dataset

- First coordinate of $\boldsymbol{z}_n$ is

$$z_{n1} = \boldsymbol{b}_1^\top \boldsymbol{x}_n$$

  ▶▶ Coordinate of orthogonal projection of $\boldsymbol{x}_n$ onto span$[\boldsymbol{b}_1]$ (1-dimensional subspace spanned by $\boldsymbol{b}_1$)

$$\mathbb{V}[z_1] = \frac{1}{N}\sum_{n=1}^{N}(\boldsymbol{b}_1^\top \boldsymbol{x}_n)^2 = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{b}_1^\top \boldsymbol{x}_n \boldsymbol{x}_n^\top \boldsymbol{b}_1$$

$$= \boldsymbol{b}_1^\top \left(\frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_n \boldsymbol{x}_n^\top\right) \boldsymbol{b}_1 = \boldsymbol{b}_1^\top \boldsymbol{S} \boldsymbol{b}_1$$

Direction with Maximal Variance (2)

- Maximize variance

$$\max_{\boldsymbol{b}_1, \|\boldsymbol{b}_1\|^2=1} \mathbb{V}[z_1] = \max_{\boldsymbol{b}_1, \|\boldsymbol{b}_1\|^2=1} \boldsymbol{b}_1^\top \boldsymbol{S} \boldsymbol{b}_1$$

Direction with Maximal Variance (2)

- Maximize variance

$$\max_{\boldsymbol{b}_1, \|\boldsymbol{b}_1\|^2=1} \mathbb{V}[z_1] = \max_{\boldsymbol{b}_1, \|\boldsymbol{b}_1\|^2=1} \boldsymbol{b}_1^\top \boldsymbol{S} \boldsymbol{b}_1$$

- Lagrangian:

$$L(\boldsymbol{b}_1, \lambda) = \boldsymbol{b}_1^\top \boldsymbol{S} \boldsymbol{b}_1 + \lambda_1 (1 - \boldsymbol{b}_1^\top \boldsymbol{b}_1)$$

Discuss with your neighbors and find λ_1 and $\boldsymbol{b}_1$

Direction with Maximal Variance (2)

- Maximize variance

$$\max_{\boldsymbol{b}_1, \|\boldsymbol{b}_1\|^2=1} \mathbb{V}[z_1] = \max_{\boldsymbol{b}_1, \|\boldsymbol{b}_1\|^2=1} \boldsymbol{b}_1^\top \boldsymbol{S} \boldsymbol{b}_1$$

- Lagrangian:

$$L(\boldsymbol{b}_1, \lambda) = \boldsymbol{b}_1^\top \boldsymbol{S} \boldsymbol{b}_1 + \lambda_1 (1 - \boldsymbol{b}_1^\top \boldsymbol{b}_1)$$

Discuss with your neighbors and find λ_1 and $\boldsymbol{b}_1$

- Setting the gradients w.r.t. $\boldsymbol{b}_1$ and λ_1 to $\mathbf{0}$ yields

$$\boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1$$
$$\boldsymbol{b}_1^\top \boldsymbol{b}_1 = 1$$

- $\boldsymbol{b}_1$ is an eigenvector of the data covariance matrix $\boldsymbol{S}$
- λ_1 is the corresponding eigenvalue

Direction with Maximal Variance (3)

- $\boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1$

$$\mathbb{V}[z_1] = \boldsymbol{b}_1^\top \boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1^\top \boldsymbol{b}_1 = \lambda_1$$

▶▶ Variance retained by first coordinate corresponds to eigenvalue λ_1

Direction with Maximal Variance (3)

- $\boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1$

$$\mathbb{V}[z_1] = \boldsymbol{b}_1^\top \boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1^\top \boldsymbol{b}_1 = \lambda_1$$

▶▶ Variance retained by first coordinate corresponds to eigenvalue λ_1

▶▶ Choose eigenvector $\boldsymbol{b}_1$ associated with the largest eigenvalue

Direction with Maximal Variance (3)

- $\boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1$

$$\mathbb{V}[z_1] = \boldsymbol{b}_1^\top \boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1^\top \boldsymbol{b}_1 = \lambda_1$$

  ⏩ Variance retained by first coordinate corresponds to eigenvalue λ_1

  ⏩ Choose eigenvector $\boldsymbol{b}_1$ associated with the largest eigenvalue
- Projection:
- Coordinate:

Direction with Maximal Variance

Maximizing the variance means to choose the direction $\boldsymbol{b}_1$ as the eigenvector of the data covariance matrix $\boldsymbol{S}$ that is associated with the largest eigenvalue λ_1 of $\boldsymbol{S}$.

Direction with Maximal Variance (3)

- $\boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1$

$$\mathbb{V}[z_1] = \boldsymbol{b}_1^\top \boldsymbol{S}\boldsymbol{b}_1 = \lambda_1 \boldsymbol{b}_1^\top \boldsymbol{b}_1 = \lambda_1$$

▶▶ Variance retained by first coordinate corresponds to eigenvalue λ_1

▶▶ Choose eigenvector $\boldsymbol{b}_1$ associated with the largest eigenvalue

- Projection: $\tilde{\boldsymbol{x}}_n = \boldsymbol{b}_1 \boldsymbol{b}_1^\top \boldsymbol{x}_n$
- Coordinate: $z_{n1} = \boldsymbol{b}_1^\top \boldsymbol{x}_n$

Direction with Maximal Variance

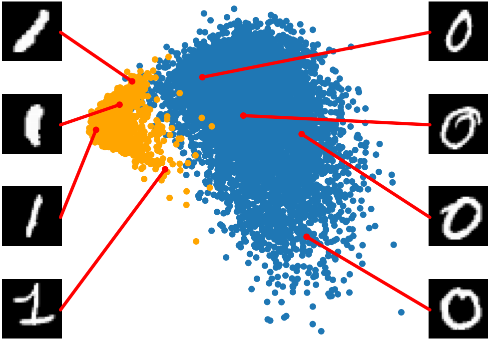
Maximizing the variance means to choose the direction $\boldsymbol{b}_1$ as the eigenvector of the data covariance matrix $\boldsymbol{S}$ that is associated with the largest eigenvalue λ_1 of $\boldsymbol{S}$.

M-dimensional Subspace with Maximum Variance

General Result

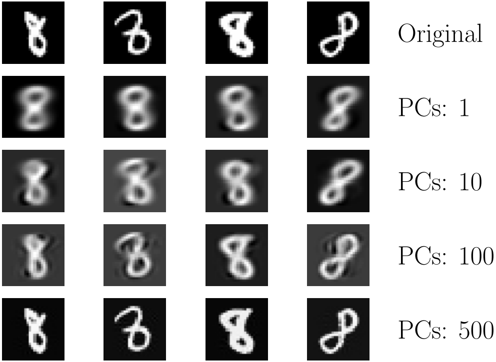
The M-dimensional subspace of $\mathbb{R}^D$ that retains the most variance is spanned by the M eigenvectors of the data covariance matrix $\boldsymbol{S}$ that are associated with the M largest eigenvalues of $\boldsymbol{S}$. (e.g., Bishop 2006)

Example: MNIST Embedding (Training Set)



- Embedding of handwritten ‘0’ and ‘1’ digits (28×28 pixels) into a two-dimensional subspace, spanned by the first two principal components.

Example: MNIST Reconstruction (Test Set)



- Reconstructions of original digits as the number of principal components increases

Overview

Refresher: Orthogonal Projection onto Subspaces

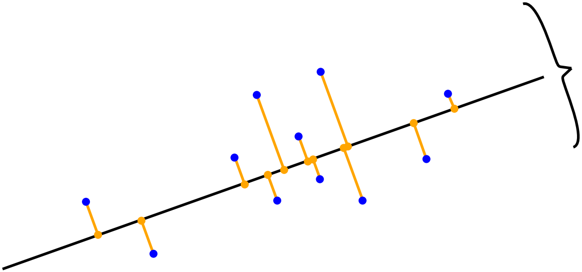
- Basis $\boldsymbol{b}_1, \dots, \boldsymbol{b}_M$ of a subspace $U \subset \mathbb{R}^D$
- Define $\boldsymbol{B} = [\boldsymbol{b}_1, ..., \boldsymbol{b}_M] \in \mathbb{R}^{D \times M}$
- Project $\boldsymbol{x} \in \mathbb{R}^D$ onto subspace U:

$$\pi_U(\boldsymbol{x}) = \tilde{\boldsymbol{x}} = \boldsymbol{B}(\boldsymbol{B}^\top \boldsymbol{B})^{-1} \boldsymbol{B}^\top \boldsymbol{x}$$

- If $\boldsymbol{b}_1, \dots, \boldsymbol{b}_M$ form an orthonormal basis ($\boldsymbol{b}_i^\top \boldsymbol{b}_j = \delta_{ij}$), then the projection simplifies to

$$\tilde{\boldsymbol{x}} = \boldsymbol{B}\boldsymbol{B}^\top \boldsymbol{x}$$

PCA Objective: Minimize Reconstruction Error



- Objective: Find orthogonal projection that minimizes the average squared projection/reconstruction error

$$J = \frac{1}{N}\sum_{n=1}^{N} \|\tilde{\boldsymbol{x}}_n - \tilde{\boldsymbol{x}}_n\|^2$$

where $\tilde{\boldsymbol{x}}_n = \pi_U(\boldsymbol{x}_n)$ is the projection of $\boldsymbol{x}_n$ onto U

Derivation (1)

- Assume an orthonormal basis of $\mathbb{R}^D = \text{span}[\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$, such that $\boldsymbol{b}_i^\top \boldsymbol{b}_j = \delta_{ij}$

Derivation (1)

- Assume an orthonormal basis of $\mathbb{R}^D = \text{span}[\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$, such that $\boldsymbol{b}_i^\top \boldsymbol{b}_j = \delta_{ij}$
- Every data point $\boldsymbol{x}$ can be written as a linear combination of the basis vectors:

$$\boldsymbol{x} = \sum_{d=1}^{D} \eta_d \boldsymbol{b}_d = \boldsymbol{B}\boldsymbol{\eta}\,, \quad \boldsymbol{B} = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$$

Derivation (1)

- Assume an orthonormal basis of $\mathbb{R}^D = \text{span}[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_D]$, such that $\boldsymbol{b}_i^\top \boldsymbol{b}_j = \delta_{ij}$
- Every data point $\boldsymbol{x}$ can be written as a linear combination of the basis vectors:

$$\boldsymbol{x} = \sum_{d=1}^{D} \eta_d \boldsymbol{b}_d = \boldsymbol{B}\boldsymbol{\eta}\,, \quad \boldsymbol{B} = [\boldsymbol{b}_1, \ldots, \boldsymbol{b}_D]$$

▶▶ Rotation of the standard coordinates to a new coordinate system defined by the basis $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_D)$.

Derivation (1)

- Assume an orthonormal basis of $\mathbb{R}^D = \text{span}[\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$, such that $\boldsymbol{b}_i^\top \boldsymbol{b}_j = \delta_{ij}$
- Every data point $\boldsymbol{x}$ can be written as a linear combination of the basis vectors:

$$\boldsymbol{x} = \sum_{d=1}^{D} \eta_d \boldsymbol{b}_d = \boldsymbol{B}\boldsymbol{\eta}\,, \quad \boldsymbol{B} = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$$

▶▶ Rotation of the standard coordinates to a new coordinate system defined by the basis $(\boldsymbol{b}_1, \dots, \boldsymbol{b}_D)$.

▶▶ Original coordinates x_d are replaced by η_d, $d = 1, \dots, D$

Derivation (1)

- Assume an orthonormal basis of $\mathbb{R}^D = \text{span}[\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$, such that $\boldsymbol{b}_i^\top \boldsymbol{b}_j = \delta_{ij}$
- Every data point $\boldsymbol{x}$ can be written as a linear combination of the basis vectors:

$$\boldsymbol{x} = \sum_{d=1}^{D} \eta_d \boldsymbol{b}_d = \boldsymbol{B}\boldsymbol{\eta}\,, \quad \boldsymbol{B} = [\boldsymbol{b}_1, \dots, \boldsymbol{b}_D]$$

▶▶ Rotation of the standard coordinates to a new coordinate system defined by the basis $(\boldsymbol{b}_1, \dots, \boldsymbol{b}_D)$.

▶▶ Original coordinates x_d are replaced by η_d, $d = 1, \dots, D$

- Obtain $\eta_d = \boldsymbol{x}^\top \boldsymbol{b}_d$, such that

$$\boldsymbol{x} = \sum_{d=1}^{D} (\boldsymbol{x}^\top \boldsymbol{b}_d) \boldsymbol{b}_d$$

Derivation (2)

Objective

Approximate

$$\boldsymbol{x} = \sum_{d=1}^{D} \eta_d \boldsymbol{b}_d \quad \text{with} \quad \tilde{\boldsymbol{x}} = \sum_{m=1}^{M} z_m \boldsymbol{b}_m$$

using $M \ll D$ many basis vectors

▶▶ **Projection** onto a lower-dimensional subspace

Derivation (2)

Objective

Approximate

$$\boldsymbol{x} = \sum_{d=1}^{D} \eta_d \boldsymbol{b}_d \quad \text{with} \quad \tilde{\boldsymbol{x}} = \sum_{m=1}^{M} z_m \boldsymbol{b}_m$$

using $M \ll D$ many basis vectors

▶▶ **Projection** onto a lower-dimensional subspace

Derivation (3): Objective

$$\tilde{\boldsymbol{x}}_n = \underbrace{\sum_{m=1}^{M} z_{mn} \boldsymbol{b}_m}_{\text{lower-dim. subspace}}$$

Derivation (3): Objective

$$\tilde{\boldsymbol{x}}_n = \underbrace{\sum_{m=1}^{M} z_{mn}\boldsymbol{b}_m}_{\text{lower-dim. subspace}}$$

- Choose coordinates z_{mn} and basis vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_D$ such that the average squared reconstruction error

$$J_M = \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n\|^2$$

is minimized

Derivation (3): Objective

$$\tilde{\boldsymbol{x}}_n = \underbrace{\sum_{m=1}^{M} z_{mn}\boldsymbol{b}_m}_{\text{lower-dim. subspace}}$$

- Choose coordinates z_{mn} and basis vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_D$ such that the average squared reconstruction error

$$J_M = \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n\|^2$$

is minimized

▶▶ Compute gradients of J_M w.r.t. all variables, set to $\mathbf{0}$, solve

Derivation (4): Optimal Coordinates

Necessary condition for optimum:

$$\frac{\partial J_M}{\partial z_{mn}} = 0 \quad \implies \quad z_{mn} = \boldsymbol{x}_n^\top \boldsymbol{b}_m\,, \qquad m = 1, \ldots, M$$

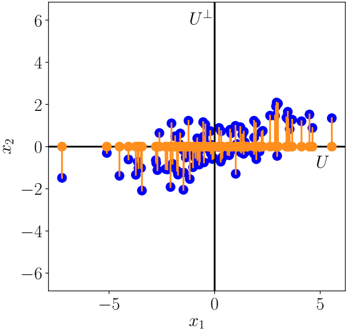
- The optimal projection is the orthogonal projection
- The optimal coordinate z_{mn} is the orthogonal projection of $\boldsymbol{x}_n$ onto the one-dimensional subspace spanned by $\boldsymbol{b}_m$
- $(\boldsymbol{b}_1, \ldots, \boldsymbol{b}_D)$ is ONB ⏩ $\text{span}[\boldsymbol{b}_{M+1}, \ldots, \boldsymbol{b}_D]$ is orthogonal complement of principal subspace ($\text{span}[\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M]$)
- If

$$\boldsymbol{x}_n = \sum\nolimits_{d=1}^{D} \eta_{dn} \boldsymbol{b}_d \quad \text{and } \tilde{\boldsymbol{x}}_n = \sum\nolimits_{m=1}^{M} z_{mn} \boldsymbol{b}_m$$

  then $\eta_{mn} = z_{mn}$ for $m = 1, \ldots, M$

⏩ Minimum error is given by the orthogonal projection of $\boldsymbol{x}_n$ onto the principal subspace spanned by $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_M$

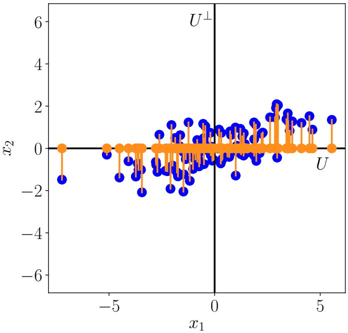
Derivation (5): Displacement Vector



Approximation error only plays a role in dimensions $M+1, \ldots, D$:

$$\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n = \sum_{j=M+1}^{D} \left(\boldsymbol{x}_n^\top \boldsymbol{b}_j\right) \boldsymbol{b}_j$$

Derivation (5): Displacement Vector



Approximation error only plays a role in dimensions $M+1,\dots,D$:

$$\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n = \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)\boldsymbol{b}_j$$

▸▸ Displacement vector $\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n$ lies in orthogonal complement $U^\perp$ of principal subspace U (linear combination of the $\boldsymbol{b}_j$ for $j = M+1,\dots,D$)

Derivation (5)

From the previous slide:

$$\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n = \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)\boldsymbol{b}_j$$

Derivation (5)

From the previous slide:

$$\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n = \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)\boldsymbol{b}_j$$

Let's compute our reconstruction error:

$$J_M = \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n\|^2 = \frac{1}{N}\sum_{n=1}^{N} (\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)^\top (\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)$$

Derivation (5)

From the previous slide:

$$\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n = \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)\boldsymbol{b}_j$$

Let's compute our reconstruction error:

$$J_M = \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n\|^2 = \frac{1}{N}\sum_{n=1}^{N} (\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)^\top(\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)$$

$$= \frac{1}{N}\sum_{n=1}^{N} \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)^2$$

Derivation (5)

From the previous slide:

$$\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n = \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)\boldsymbol{b}_j$$

Let's compute our reconstruction error:

$$\begin{aligned} J_M &= \frac{1}{N}\sum_{n=1}^{N} \|\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n\|^2 = \frac{1}{N}\sum_{n=1}^{N} (\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n)^\top(\boldsymbol{x}_n - \tilde{\boldsymbol{x}}_n) \\ &= \frac{1}{N}\sum_{n=1}^{N} \sum_{j=M+1}^{D} (\boldsymbol{x}_n^\top \boldsymbol{b}_j)^2 \\ &= \sum_{j=M+1}^{D} \boldsymbol{b}_j^\top \boldsymbol{S} \boldsymbol{b}_j \end{aligned}$$

where $\boldsymbol{S} = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_n \boldsymbol{x}_n^\top$ is the data covariance matrix

Derivation (6)

- What remains: Minimize J_M w.r.t. $\boldsymbol{b}_j$ under the constraint that the $\boldsymbol{b}_j$ form an orthonormal basis.
- Similar setting to maximum variance perspective: Instead of maximizing the variance in the principal subspace, we minimize the variance in the orthogonal complement of the principal subspace
- End up with **eigenvalue problem**:

$$\boldsymbol{S}\boldsymbol{b}_j = \lambda_j \boldsymbol{b}_j \,, \quad j = D+1, \ldots, M$$

Derivation (7)

- Find the eigenvectors $\boldsymbol{b}_j$ of the data covariance matrix $\boldsymbol{S}$

Derivation (7)

- Find the eigenvectors $\boldsymbol{b}_j$ of the data covariance matrix $\boldsymbol{S}$
- Corresponding value of the squared reconstruction error:

$$J_M = \sum_{j=M+1}^{D} \lambda_j$$

i.e., the sum of the eigenvalues associated with eigenvectors not in the principle subspace

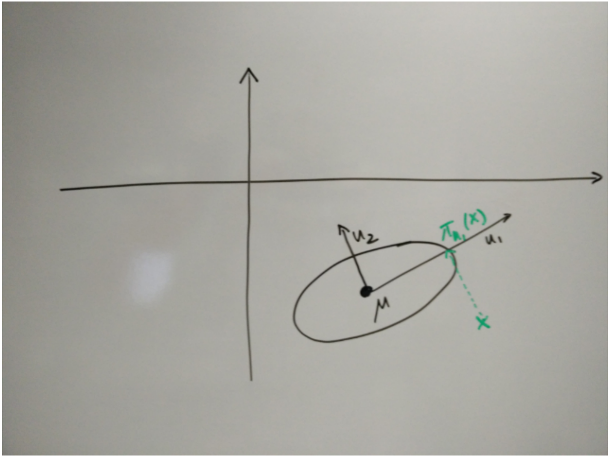
Derivation (7)

- Find the eigenvectors $\boldsymbol{b}_j$ of the data covariance matrix $\boldsymbol{S}$
- Corresponding value of the squared reconstruction error:

$$J_M = \sum_{j=M+1}^{D} \lambda_j$$

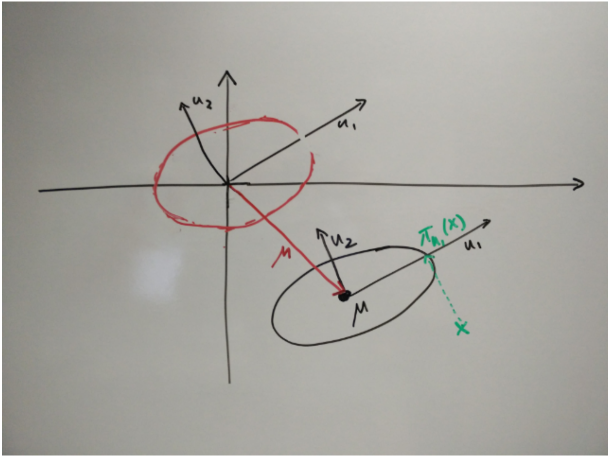
  i.e., the sum of the eigenvalues associated with eigenvectors not in the principle subspace
- Minimizing J_M requires us to choose the M eigenvectors as the principle subspace that are associated with the M largest eigenvalues.

Geometric Interpretation



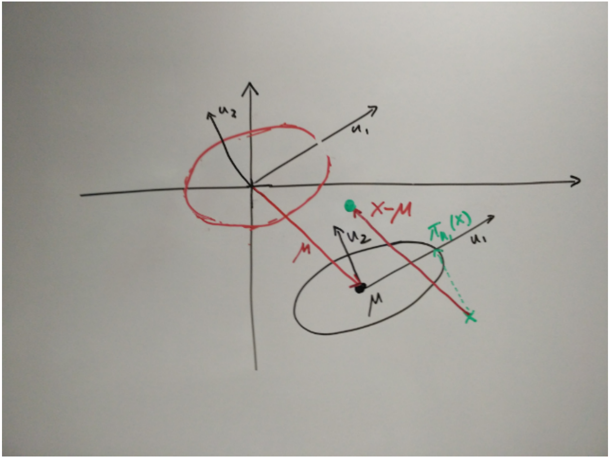
- Objective: Project $\boldsymbol{x}$ onto an affine subspace $\boldsymbol{\mu} + \text{span}[\boldsymbol{b}_1]$.

Geometric Interpretation



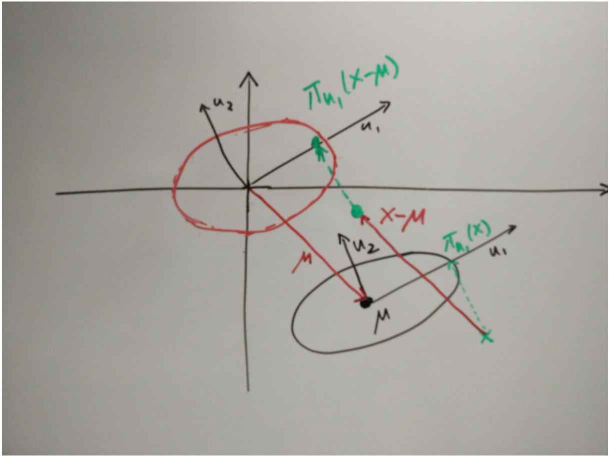
- Shift scenario to the origin (affine subspace ⇝ vector subspace)

Geometric Interpretation



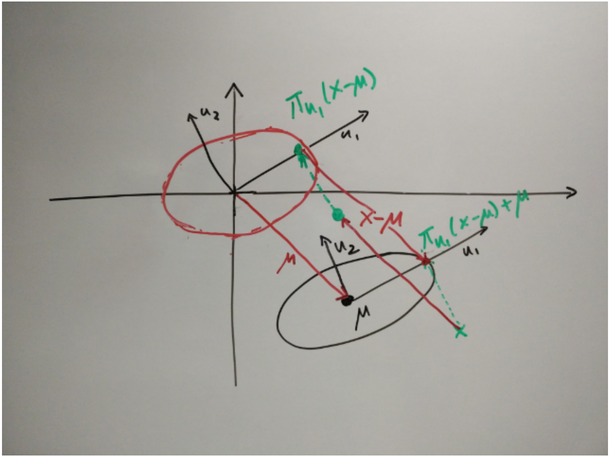
- Shift $\boldsymbol{x}$ as well (onto $\boldsymbol{x} - \boldsymbol{\mu}$).

Geometric Interpretation



- Orthogonal projection of $\boldsymbol{x} - \boldsymbol{\mu}$ onto subspace spanned by $\boldsymbol{b}_1$

Geometric Interpretation



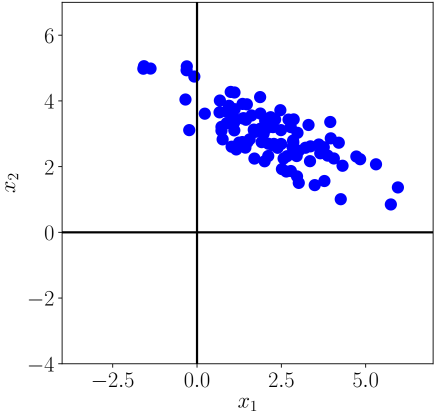
- Move projected point $\pi_{U_1}(\boldsymbol{x})$ back into original (affine) setting.

Overview

Key Steps of PCA

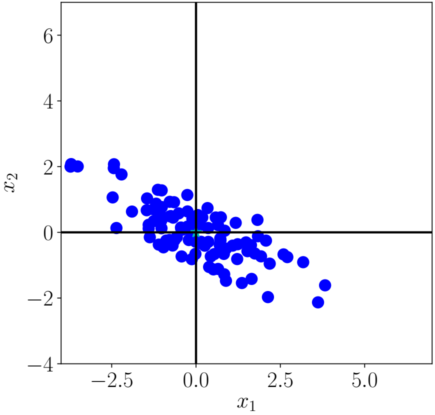
1. Compute the empirical mean $\boldsymbol{\mu}$ of the data
2. Mean subtraction: Replace all data points $\boldsymbol{x}_i$ with $\bar{\boldsymbol{x}}_i = \boldsymbol{x}_i - \boldsymbol{\mu}$.
3. Standardization: Divide the data by its standard deviation in each dimension: $\hat{\boldsymbol{X}}^{(d)} = \bar{\boldsymbol{X}}/\sigma(\boldsymbol{X}^{(d)})$ for $d = 1, \ldots, D$.
4. Eigendecomposition of the data covariance matrix: Compute the eigenvectors (orthonormal) and eigenvalues of the data covariance matrix $\boldsymbol{S}$
5. Orthogonal projection: Choose the eigenvectors associated with the M largest eigenvalues to be the basis of the principal subspace. Obtain $\tilde{\boldsymbol{X}}$
6. Moving back to original data space: $\tilde{\boldsymbol{X}}^{(d)} = \tilde{\boldsymbol{X}}^{(d)}\sigma(\boldsymbol{X}^{(d)}) + \mu_d$

PCA Algorithm



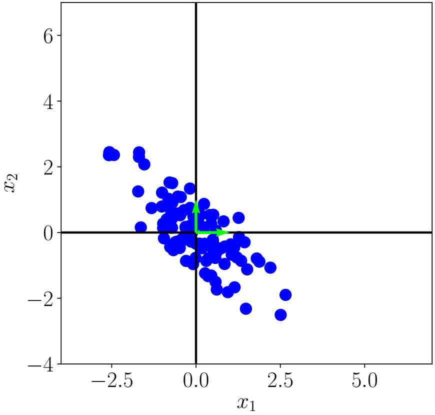
- Dataset

PCA Algorithm: Step 1



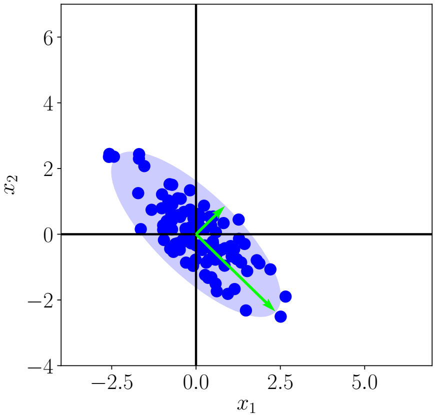
- Mean subtraction

PCA Algorithm: Step 2



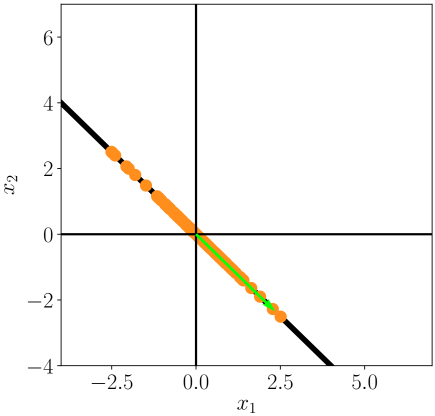
- Standardization (variance 1 in each direction)

PCA Algorithm: Step 3



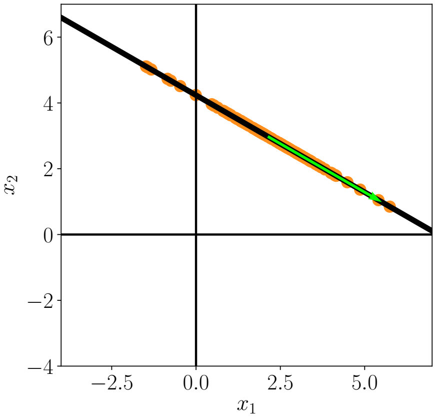
- Eigendecomposition of the data covariance matrix

PCA Algorithm: Step 4



- Orthogonal projection onto the principal subspace

PCA Algorithm: Step 5



- Moving back to the original data space

Overview

PCA for High-Dimensional Data

- Fewer data points than dimensions, i.e., $N < D$.
- At least $D - N + 1$ eigenvalues 0.
- Computation time for computing eigenvalues of data covariance matrix $\boldsymbol{S}$: $\mathcal{O}(D^3)$
- Rephrase PCA

Reformulating PCA

- Define $\boldsymbol{X}$ to be the $D \times N$-dimensional centered data matrix, whose nth row is $(\boldsymbol{x}_n - \mathbb{E}[\boldsymbol{x}])^\top$ ▸▸ Mean normalization

Reformulating PCA

- Define $\boldsymbol{X}$ to be the $D \times N$-dimensional centered data matrix, whose nth row is $(\boldsymbol{x}_n - \mathbb{E}[\boldsymbol{x}])^\top$ ⏩ Mean normalization
- Corresponding covariance: $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top$

Reformulating PCA

- Define $\boldsymbol{X}$ to be the $D \times N$-dimensional centered data matrix, whose nth row is $(\boldsymbol{x}_n - \mathbb{E}[\boldsymbol{x}])^\top$ ▸▸ Mean normalization
- Corresponding covariance: $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top$
- Corresponding eigenvector equation:

$$\boldsymbol{S}\boldsymbol{b}_i = \lambda_i \boldsymbol{b}_i \iff \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{b}_i = \lambda_i \boldsymbol{b}_i$$

Reformulating PCA

- Define $\boldsymbol{X}$ to be the $D \times N$-dimensional centered data matrix, whose nth row is $(\boldsymbol{x}_n - \mathbb{E}[\boldsymbol{x}])^\top$ ▶▶ Mean normalization
- Corresponding covariance: $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top$
- Corresponding eigenvector equation:

$$\boldsymbol{S}\boldsymbol{b}_i = \lambda_i \boldsymbol{b}_i \iff \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{b}_i = \lambda_i \boldsymbol{b}_i$$

- Transformation (left-multiply by $\boldsymbol{X}^\top$):

$$\frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top \boldsymbol{b}_i = \lambda_i \boldsymbol{b}_i \quad \iff \quad \frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X} \underbrace{\boldsymbol{X}^\top \boldsymbol{b}_i}_{=:\boldsymbol{v}_i} = \lambda_i \underbrace{\boldsymbol{X}^\top \boldsymbol{b}_i}_{=:\boldsymbol{v}_i}$$

▶▶ $\boldsymbol{v}_i$ is an eigenvector of the $N \times N$-matrix $\frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X}$, which has the same non-zero eigenvalues as the original covariance matrix.

▶▶ Get eigenvalues in $\mathcal{O}(N^3)$ instead of $\mathcal{O}(D^3)$.

Recovering the Original Eigenvectors

- The new eigenvalue/eigenvector equation is

$$\frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

where $\boldsymbol{v}_i = \boldsymbol{X}^\top \boldsymbol{b}_i$

Recovering the Original Eigenvectors

- The new eigenvalue/eigenvector equation is

$$\frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

  where $\boldsymbol{v}_i = \boldsymbol{X}^\top \boldsymbol{b}_i$
- We want to recover the original eigenvectors $\boldsymbol{b}_i$ of the data covariance matrix $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top$

Recovering the Original Eigenvectors

- The new eigenvalue/eigenvector equation is

$$\frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

  where $\boldsymbol{v}_i = \boldsymbol{X}^\top \boldsymbol{b}_i$
- We want to recover the original eigenvectors $\boldsymbol{b}_i$ of the data covariance matrix $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top$
- Left-multiply eigenvector equation by $\boldsymbol{X}$ yields

$$\underbrace{\frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top}_{=\boldsymbol{S}} \boldsymbol{X}\boldsymbol{v}_i = \lambda_i \boldsymbol{X}\boldsymbol{v}_i$$

  and we recover $\boldsymbol{X}\boldsymbol{v}_i$ as an eigenvector of $\boldsymbol{S}$ associated with eigenvalue λ_i

Recovering the Original Eigenvectors

- The new eigenvalue/eigenvector equation is

$$\frac{1}{N}\boldsymbol{X}^\top \boldsymbol{X}\boldsymbol{v}_i = \lambda_i \boldsymbol{v}_i$$

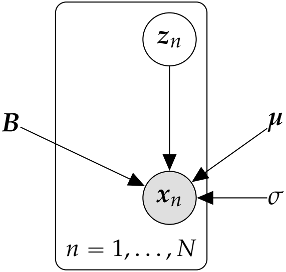
  where $\boldsymbol{v}_i = \boldsymbol{X}^\top \boldsymbol{b}_i$
- We want to recover the original eigenvectors $\boldsymbol{b}_i$ of the data covariance matrix $\boldsymbol{S} = \frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top$
- Left-multiply eigenvector equation by $\boldsymbol{X}$ yields

$$\underbrace{\frac{1}{N}\boldsymbol{X}\boldsymbol{X}^\top}_{=\boldsymbol{S}} \boldsymbol{X}\boldsymbol{v}_i = \lambda_i \boldsymbol{X}\boldsymbol{v}_i$$

  and we recover $\boldsymbol{X}\boldsymbol{v}_i$ as an eigenvector of $\boldsymbol{S}$ associated with eigenvalue λ_i
- Make sure to normalize $\boldsymbol{X}\boldsymbol{v}_i$ so that $\|\boldsymbol{X}\boldsymbol{v}_i\| = 1$

Overview

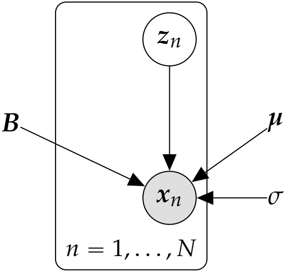
Latent Variable Perspective



- Model:

$$x = Bz + \mu + \epsilon\,, \quad \epsilon \sim \mathcal{N}\big(0,\, \sigma^2 I\big)$$

Latent Variable Perspective



- Model:

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}\,, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\big(\boldsymbol{0},\, \sigma^2 \boldsymbol{I}\big)$$

- Generative process:

$$\boldsymbol{z} \sim \mathcal{N}\big(\boldsymbol{0},\, \boldsymbol{I}\big)$$
$$\boldsymbol{x}|\boldsymbol{z} \sim \mathcal{N}\big(\boldsymbol{x} \,|\, \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu},\, \sigma^2 \boldsymbol{I}\big)$$

Why is this useful?

- “Standard” PCA as a special case,

Why is this useful?

- "Standard" PCA as a special case,
- Comes with a likelihood function, and we can explicitly deal with noisy observations

Why is this useful?

- "Standard" PCA as a special case,
- Comes with a likelihood function, and we can explicitly deal with noisy observations
- Allow for Bayesian model comparison via the marginal likelihood

Why is this useful?

- "Standard" PCA as a special case,
- Comes with a likelihood function, and we can explicitly deal with noisy observations
- Allow for Bayesian model comparison via the marginal likelihood
- PCA as a generative model, which allows us to simulate new data

Why is this useful?

- "Standard" PCA as a special case,
- Comes with a likelihood function, and we can explicitly deal with noisy observations
- Allow for Bayesian model comparison via the marginal likelihood
- PCA as a generative model, which allows us to simulate new data
- Straightforward connections to related algorithms and models (e.g., ICA)

Why is this useful?

- "Standard" PCA as a special case,
- Comes with a likelihood function, and we can explicitly deal with noisy observations
- Allow for Bayesian model comparison via the marginal likelihood
- PCA as a generative model, which allows us to simulate new data
- Straightforward connections to related algorithms and models (e.g., ICA)
- Deal with data dimensions that are missing at random by applying Bayes' theorem

Likelihood

- Model:

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}\,, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\big(\boldsymbol{0},\, \sigma^2 \boldsymbol{I}\big)$$

Likelihood

- Model:

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}\,, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\big(\boldsymbol{0},\, \sigma^2 \boldsymbol{I}\big)$$

- PPCA Likelihood (integrate out the latent variables):

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{B}, \boldsymbol{\mu}, \sigma^2) &= \int p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\mu}, \sigma^2) p(\boldsymbol{z}) d\boldsymbol{z} \\ &= \int \mathcal{N}\big(\boldsymbol{x} \,|\, \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu},\, \sigma^2 \boldsymbol{I}\big) \mathcal{N}\big(\boldsymbol{z} \,|\, \boldsymbol{0},\, \boldsymbol{I}\big) \mathrm{d}\boldsymbol{z} \end{aligned}$$

Likelihood

- Model:

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}\,, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\big(\boldsymbol{0},\, \sigma^2 \boldsymbol{I}\big)$$

- PPCA Likelihood (integrate out the latent variables):

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{B},\boldsymbol{\mu},\sigma^2) &= \int p(\boldsymbol{x}|\boldsymbol{z},\boldsymbol{\mu},\sigma^2)p(\boldsymbol{z})d\boldsymbol{z} \\ &= \int \mathcal{N}\big(\boldsymbol{x}\,|\,\boldsymbol{B}\boldsymbol{z}+\boldsymbol{\mu},\, \sigma^2\boldsymbol{I}\big)\mathcal{N}\big(\boldsymbol{z}\,|\,\boldsymbol{0},\, \boldsymbol{I}\big)\mathrm{d}\boldsymbol{z} \end{aligned}$$

▶▶ Is Gaussian with mean and covariance

Likelihood

- Model:

$$\boldsymbol{x} = \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}\big(\boldsymbol{0},\, \sigma^2 \boldsymbol{I}\big)$$

- PPCA Likelihood (integrate out the latent variables):

$$\begin{aligned} p(\boldsymbol{x}|\boldsymbol{B}, \boldsymbol{\mu}, \sigma^2) &= \int p(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{\mu}, \sigma^2) p(\boldsymbol{z}) d\boldsymbol{z} \\ &= \int \mathcal{N}\big(\boldsymbol{x} \,|\, \boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu},\, \sigma^2 \boldsymbol{I}\big) \mathcal{N}\big(\boldsymbol{z} \,|\, \boldsymbol{0},\, \boldsymbol{I}\big) \mathrm{d}\boldsymbol{z} \end{aligned}$$

  ▸▸ Is Gaussian with mean and covariance

$$\begin{aligned} \mathbb{E}[\boldsymbol{x}] &= \mathbb{E}_{\boldsymbol{z}}[\boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}] = \boldsymbol{\mu} \\ \mathbb{V}[\boldsymbol{x}] &= \mathbb{V}_{\boldsymbol{z}}[\boldsymbol{B}\boldsymbol{z} + \boldsymbol{\mu} + \boldsymbol{\epsilon}] = \boldsymbol{B}\boldsymbol{B}^\top + \sigma^2 \boldsymbol{I} \end{aligned}$$

Joint Distribution and Posterior

- Joint distribution of observed and latent variables

$$p(\boldsymbol{x}, \boldsymbol{z} | \boldsymbol{B}, \boldsymbol{\mu}, \sigma^2) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{z}\end{bmatrix} \,\middle|\, \begin{bmatrix}\boldsymbol{\mu}\\ \boldsymbol{0}\end{bmatrix}, \begin{bmatrix}\boldsymbol{B}\boldsymbol{B}^\top + \sigma^2\boldsymbol{I} & \boldsymbol{B}\\ \boldsymbol{B}^\top & \boldsymbol{I}\end{bmatrix}\right)$$

- Posterior via Gaussian conditioning:

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{B}, \boldsymbol{\mu}, \sigma^2) &= \mathcal{N}\big(\boldsymbol{z} \,|\, \boldsymbol{m}, \boldsymbol{C}\big) \\ \boldsymbol{m} &= \boldsymbol{B}^\top(\boldsymbol{B}\boldsymbol{B}^\top + \sigma^2\boldsymbol{I})^{-1}(\boldsymbol{x} - \boldsymbol{\mu}) \\ \boldsymbol{C} &= \boldsymbol{I} - \boldsymbol{B}^\top(\boldsymbol{B}\boldsymbol{B}^\top + \sigma^2\boldsymbol{I})^{-1}\boldsymbol{B} \end{aligned}$$

▶▶ For a new observation $\boldsymbol{x}_*$ compute the posterior on $p(\boldsymbol{z}_*|\boldsymbol{x}_*, \boldsymbol{X})$ and examine it (e.g., variance).

- Generate new (plausible) data from this posterior

Maximum Likelihood Estimation

- In PPCA, we can determine the parameters $\boldsymbol{\mu}, \boldsymbol{B}, \sigma^2$ via maximum likelihood estimation. PPCA Likelihood: $p(\boldsymbol{X}|\boldsymbol{\mu}, \boldsymbol{B}, \sigma^2)$
- Result (e.g., Tipping & Bishop (1999)):

$$\boldsymbol{\mu}_{\text{ML}} = \frac{1}{N}\sum_{n=1}^{N} \boldsymbol{x}_n$$ ▶▶Sample mean

$$\boldsymbol{B}_{\text{ML}} = \boldsymbol{T}(\boldsymbol{\Lambda} - \sigma^2 \boldsymbol{I})^{\frac{1}{2}} \boldsymbol{R}$$

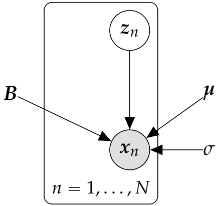
$$\sigma^2_{\text{ML}} = \frac{1}{D-M}\sum_{j=M+1}^{D} \lambda_j$$ ▶▶Average variance in orth. complement

- For $\sigma \to 0$ the maximum likelihood solution gives the same result as PCA (see `mml-book.com`)

Overview

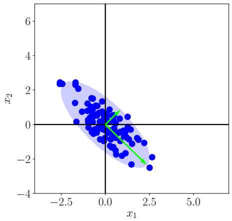
Related Models

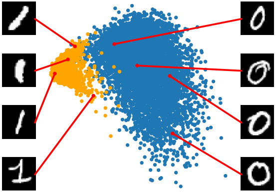
- Factor analysis: Axis-aligned noise (instead of isotropic)
- Independent component analysis: Non-Gaussian prior $p(\boldsymbol{z}) = \prod_m p_m(z_m)$
- Kernel PCA
- Bayesian PCA: Priors on parameters $\boldsymbol{B}, \boldsymbol{\mu}, \sigma^2$ ▶▶ Approximate inference

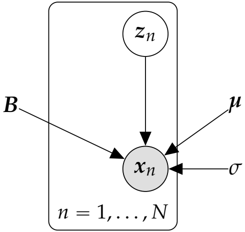


- Gaussian process latent variable model (GP-LVM): Replace linear mapping in Bayesian PCA with Gaussian process. Point estimate of $\boldsymbol{z}$
- Bayesian GP-LVM maintains a distribution on $\boldsymbol{z}$ ▶▶ Approximate inference

Summary







- PCA: Algorithm for linear dimensionality reduction
- Orthogonal projection of data onto a lower-dimensional subspace
  - Maximizes the variance of the projection
  - Minimizes the average squared projection/reconstruction error
- High-dimensional data
- Probabilistic PCA