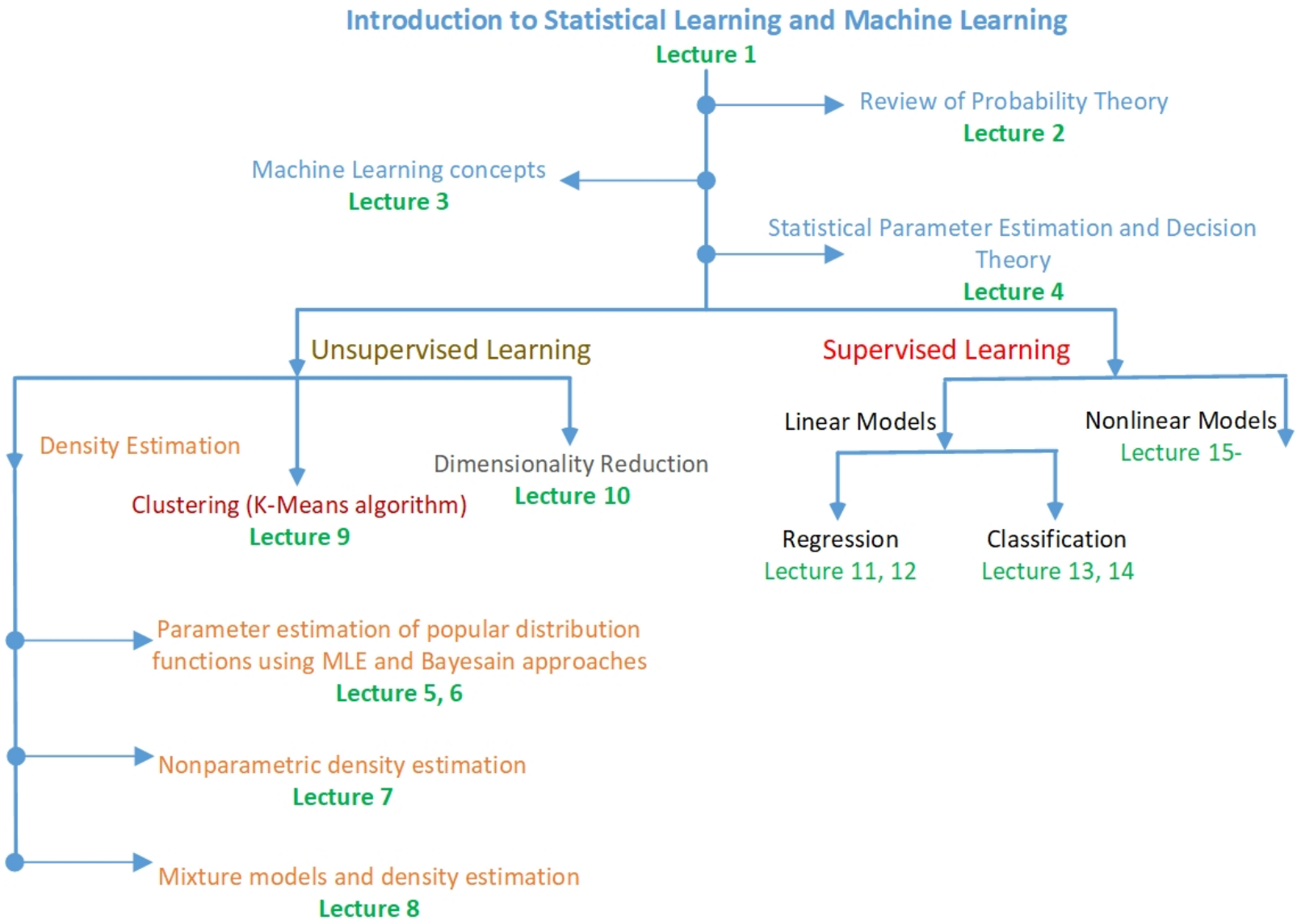
Statistical Learning and Machine Learning

Lecture 10 - Principal Component Analysis

September 18, 2021

Course overview and where do we stand



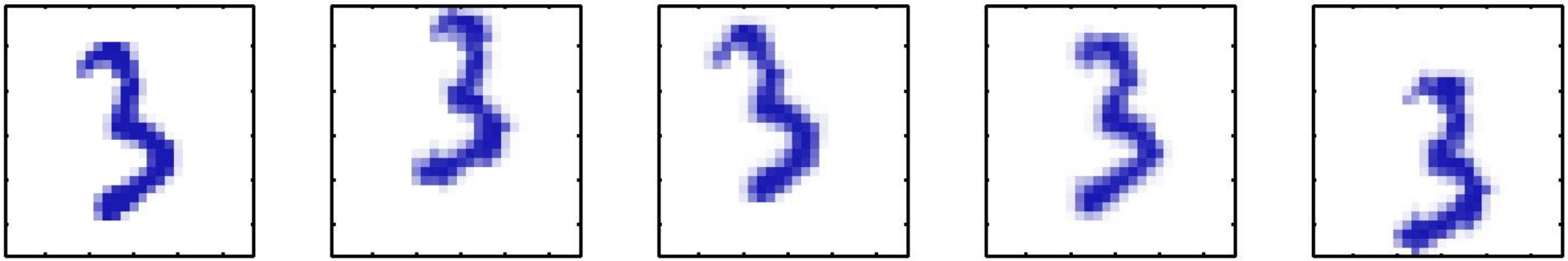
Main topics of the lecture

- Principal Component Analysis (PCA) and applications
- Minimum Variance Formulation of PCA
- Probabilistic PCA

Intrinsic dimensionality of data

Data forming data sets usually lie close to a manifold of much lower dimensionality than that of the original data space. In this example:

- translation
- rotation
- zero-padding



Principal Component Analysis

PCA is a technique for determining a linear data transformation (data projection) of the form:

$$y = U^T x. \tag{18}$$

It is widely used for:

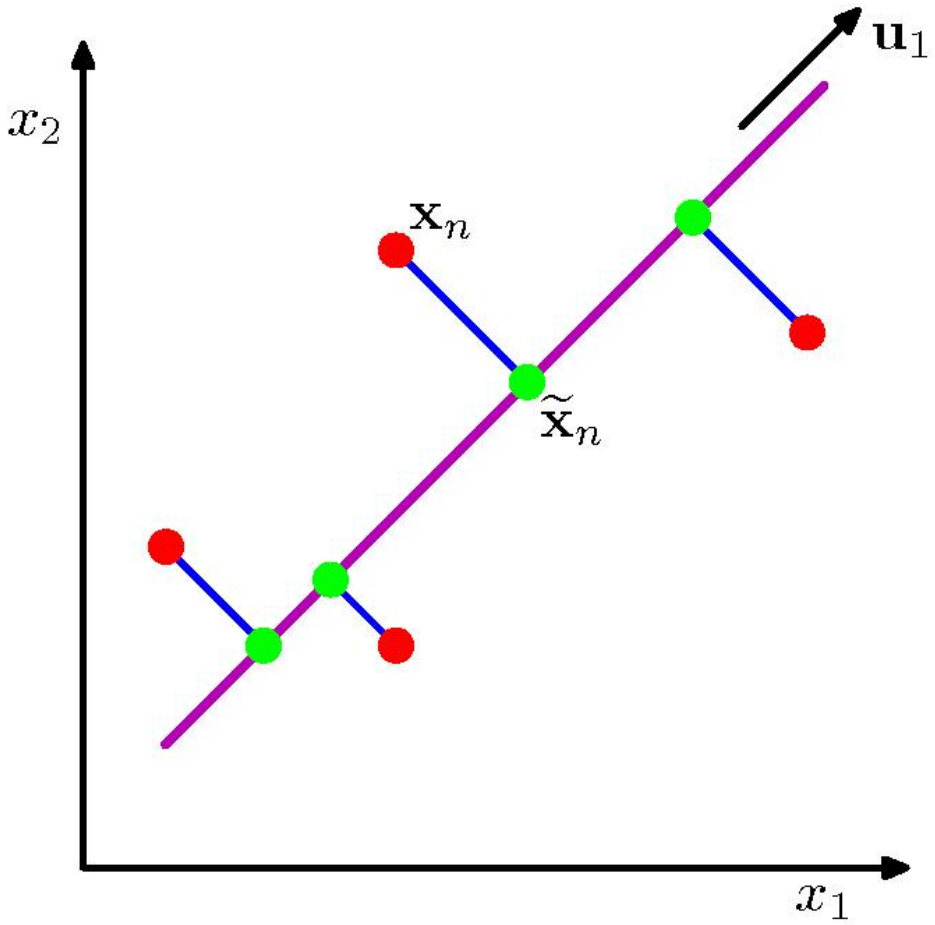
- dimensionality reduction
- lossy data compression
- feature extraction
- data visualization.

There are two ways to define it as:

- the data projection preserving the maximum variance of the data set
- the data projection minimizing the average transformation cost.

Principal Component Analysis

The space on which the data are projected is called principal subspace. In this example the 2D data points are projected to the magenta line.



PCA: Maximum variance formulation

Let $\mathcal{D} = \{x_1, \ldots, x_N\}$ be a set of N data points $x_n \in \mathbb{R}^D$. Our goal is to:

- project the data onto a space with dimensionality $M < D$ (we assume that M is given)
- maximize the variance of the projected data.

PCA: Maximum variance formulation ($M = 1$)

We use a vector $u_1 \in \mathbb{R}^D$ to project the vector $x_n \in \mathbb{R}^D$ to a value y_n:

- we are interested only in the direction of the line on which $y_n,\ n = 1, \ldots, N$ lie
- we choose a unit vector: $u_1^T u_1 = 1$.

The mean value and the variance of the projected data are:

$$\bar{y} = \frac{1}{N} \sum_{n=1}^{N} u_1^T x_n = u_1^T \bar{x} \tag{19}$$

$$var[y] = \frac{1}{N} \sum_{n=1}^{N} (u_1^T x_n - u_1^T \bar{x})^2 = u_1^T S u_1 \tag{20}$$

where S is the covariance matrix of the original data:

$$S = \frac{1}{N} \sum_{n=1}^{N} (x_n - \bar{x})(x_n - \bar{x})^T \tag{21}$$

PCA: Maximum variance formulation ($M = 1$)

We want to maximize $var[y]$ under the constraint that $\mathrm{u}_1^T \mathrm{u}_1 = 1$. To do so, we introduce a Lagrange multiplier and maximize for:

$$\mathcal{J}_{PCA} = \mathrm{u}_1^T \mathrm{S} \mathrm{u}_1 + \lambda_1 (1 - \mathrm{u}_1^T \mathrm{u}_1). \tag{22}$$

Setting $\frac{\theta \mathcal{J}_{PCA}}{\theta \mathrm{u}_1} = 0$:

$$\mathrm{S}\mathrm{u}_1 = \lambda_1 \mathrm{u}_1 \tag{23}$$

Thus, u_1 is an eigenvector of S.

By multiplying both sides of (23) with u_1^T we get: $\mathrm{u}_1^T \mathrm{S} \mathrm{u}_1 = \lambda_1$

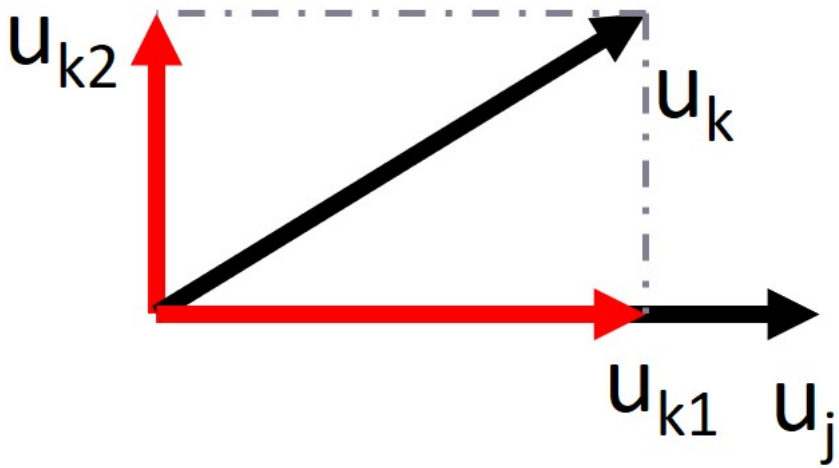
To maximize the variance, we set u_1 equal to the eigenvector of S corresponding to the largest eigenvalue λ_1.

u_1 is called the *first principal component*.

PCA: Maximum variance formulation ($M > 1$)

We define additional principal components in an incremental fashion:

- u_1 is the first principal component of S
- u_k, $k > 1$, such that
  - $u_k^T u_k = 1$
  - $u_k^T u_j = 0$, $\forall j < k$.



The data projection matrix is formed by the eigenvectors corresponding to the M largest eigenvalues of S ($U = [u_1, \ldots, u_M]$).

PCA: Minimum-error formulation

We define a new coordinate system $\{\mathrm{u}_j\}$, $i = 1, \ldots, D$ that satisfy $\mathrm{u}_j^T \mathrm{u}_j = 1$, and $\mathrm{u}_i^T \mathrm{u}_j = 0$ for $i \neq j$.

The data points x_n, $n = 1, \ldots, N$ can be expressed in the new coordinate system by:

$$\mathrm{x}_n = \sum_{i=1}^{D} \alpha_{ni} \mathrm{u}_i \tag{24}$$

The above corresponds to:

- a coordinate change from $\mathrm{x}_n \in \mathbb{R}^D$ to $\boldsymbol{\alpha}_n \in \mathbb{R}^D$
- $\alpha_{nj} = \mathrm{u}_j^T \mathrm{x}_n$ being the dimension of the n-th data point in the new coordinate system

PCA: Minimum-error formulation

Thus the original data points can be reconstructed by:

$$\mathrm{x}_n = \sum_{i=1}^{D} (\mathrm{u}_i^T \mathrm{x}_n) \mathrm{u}_i \tag{25}$$

However, we want to approximate the data points using only M data point-dependent dimensions:

$$\tilde{\mathrm{x}}_n = \sum_{i=1}^{M} z_{ni} \mathrm{u}_i + \sum_{i=M+1}^{D} b_i \mathrm{u}_i \tag{26}$$

where $\{z_{ni}\}$ depend on the particular data point and $\{b_i\}$ are the same for all points.

PCA: Minimum-error formulation

We use the following cost function:

$$\begin{aligned} \mathcal{J} &= \frac{1}{N}\sum_{n=1}^{N}\|\mathrm{x}_n - \tilde{\mathrm{x}}_n\|^2 \\ &= \frac{1}{N}\sum_{n=1}^{N}\|\mathrm{x}_n - \left(\sum_{i=1}^{M} z_{ni}\mathrm{u}_i + \sum_{i=M+1}^{D} b_i\mathrm{u}_i\right)\|^2 \end{aligned} \tag{27}$$

which is a function of $\{\mathrm{u}_i\}$, $\{z_{ni}\}$ and $\{b_i\}$, $n = 1, \ldots, N$, $i = 1, \ldots, D$.

PCA: Minimum-error formulation

Minimization w.r.t. $\{z_{ni}\}$:

$$\frac{\theta \mathcal{J}}{z_{nj}} = 0 \Rightarrow z_{nj} = \mathrm{x}_n^T \mathrm{u}_j, \;\; j = 1, \ldots, M \tag{28}$$

Minimization w.r.t. $\{b_j\}$:

$$\frac{\theta \mathcal{J}}{b_j} = 0 \Rightarrow b_j = \bar{\mathrm{x}}^T \mathrm{u}_j, \;\; j = M+1, \ldots, D \tag{29}$$

Substituting the above to $\mathcal{J}$ and using $\mathrm{x}_n = \sum_{i=1}^{D} (\mathrm{u}_i^T \mathrm{x}_n) \mathrm{u}_i$ we get:

$$\mathrm{x}_n - \tilde{\mathrm{x}}_n = \sum_{i=M+1}^{D} \{(\mathrm{x}_n - \bar{\mathrm{x}})^T \mathrm{u}_i\} \mathrm{u}_i \tag{30}$$

PCA: Minimum-error formulation

Thus:

$$\mathcal{J} = \frac{1}{N}\sum_{n=1}^{N}\sum_{i=M+1}^{D}(\mathrm{x}_n^T\mathrm{u}_i - \bar{\mathrm{x}}^T\mathrm{u}_i)^2 = \sum_{i=M+1}^{D}\mathrm{u}_i^T\mathrm{S}\mathrm{u}_i \tag{31}$$

For u_j, $j = M+1, \ldots, D$, we want to minimize $\mathcal{J}$ under the constraint that $\mathrm{u}_j^T\mathrm{u}_j = 1$. To do so, we introduce a Lagrange multiplier and minimize for:

$$\tilde{\mathcal{J}} = \mathrm{u}_j^T\mathrm{S}\mathrm{u}_j + \lambda_j(1 - \mathrm{u}_j^T\mathrm{u}_j). \tag{32}$$

To minimize $\tilde{\mathcal{J}}$, we set u_i, $i = M+1, \ldots, D$ equal to the eigenvectors of S corresponding to the smallest eigenvalues λ_i.

The distortion is equal to: $\mathcal{J} = \sum_{i=M+1}^{D}\lambda_i$.

PCA for high-dimensional data

The application of eigen-analysis to the matrix $\mathsf{S} \in \mathbb{R}^{D \times D}$ has a time complexity of $O(D^3)$. When D is large, application of PCA is very slow.

Let $\mathsf{X} \in \mathbb{R}^{N \times D}$ be the (centered) data matrix. Then we have:

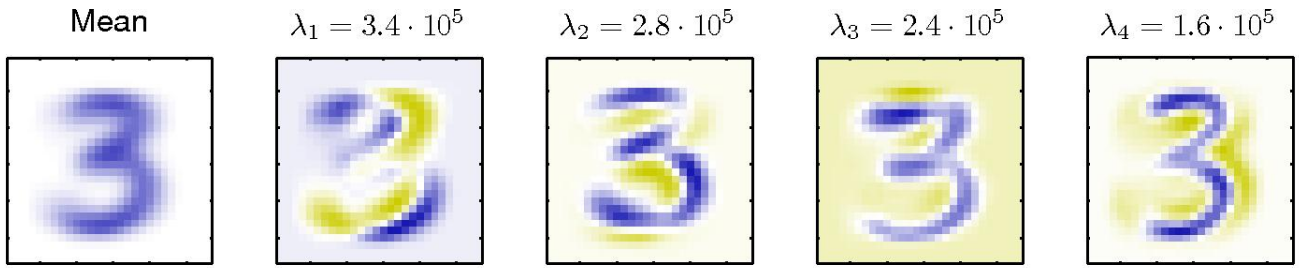
- $\mathsf{S} = \frac{1}{N}\mathsf{X}^T\mathsf{X}$
- The original eigen-analysis problem is: $\frac{1}{N}\mathsf{X}^T\mathsf{X}\mathsf{u}_i = \lambda_i \mathsf{u}_i$
- Multiply with X both sides: $\frac{1}{N}\mathsf{X}\mathsf{X}^T(\mathsf{X}\mathsf{u}_i) = \lambda_i(\mathsf{X}\mathsf{u}_i)$
- Set $(\mathsf{X}\mathsf{u}_i) = \mathsf{v}_i$ and we get: $\frac{1}{N}\mathsf{X}\mathsf{X}^T\mathsf{v}_i = \lambda_i \mathsf{v}_i$
- Solve the above problem which has time complexity $O(N^3)$
- Calculate the original eigenvectors u_i by: $\mathsf{u}_i = \frac{1}{(N\lambda_i)^{1/2}}\mathsf{X}^T\mathsf{v}_i$

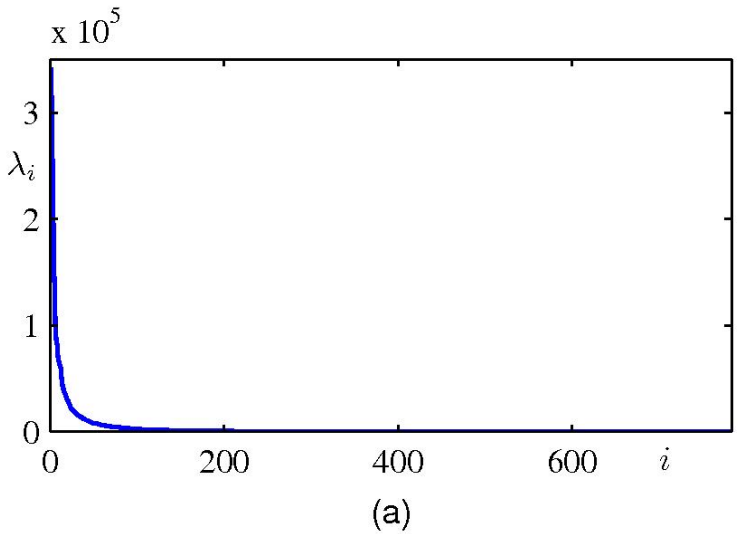
Thus, when $N < D$ we can apply PCA faster (with time complexity $O(N^3)$).

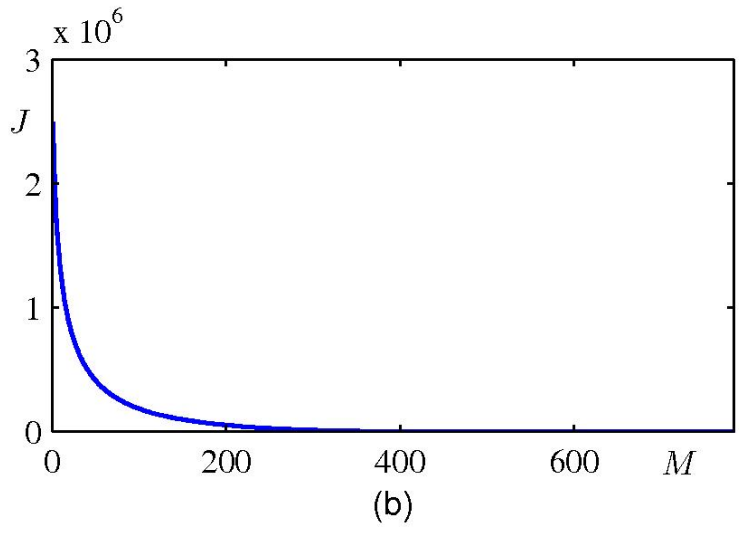
PCA: Applications

Data compression:

$$\tilde{x}_n = \hat{x} + \sum_{i=1}^{M} (x_n^T u_i - \hat{x}^T u_i) u_i \tag{33}$$

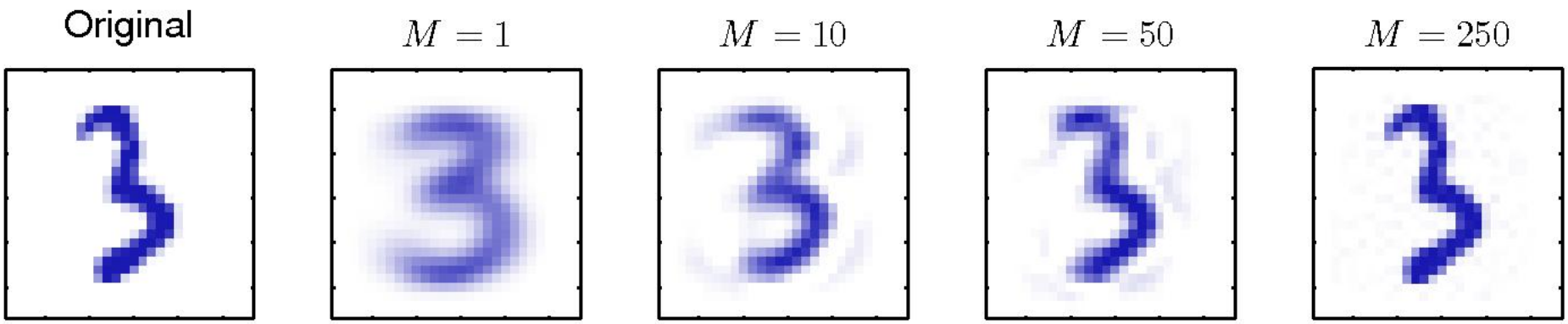






PCA: Applications

Data reconstruction for a varying number of principal components



PCA: Applications

Data pre-processing:

- when the data dimensions correspond to measurements of different types (leading to different scaling)
- normalization helps in appropriately scaling the various dimensions

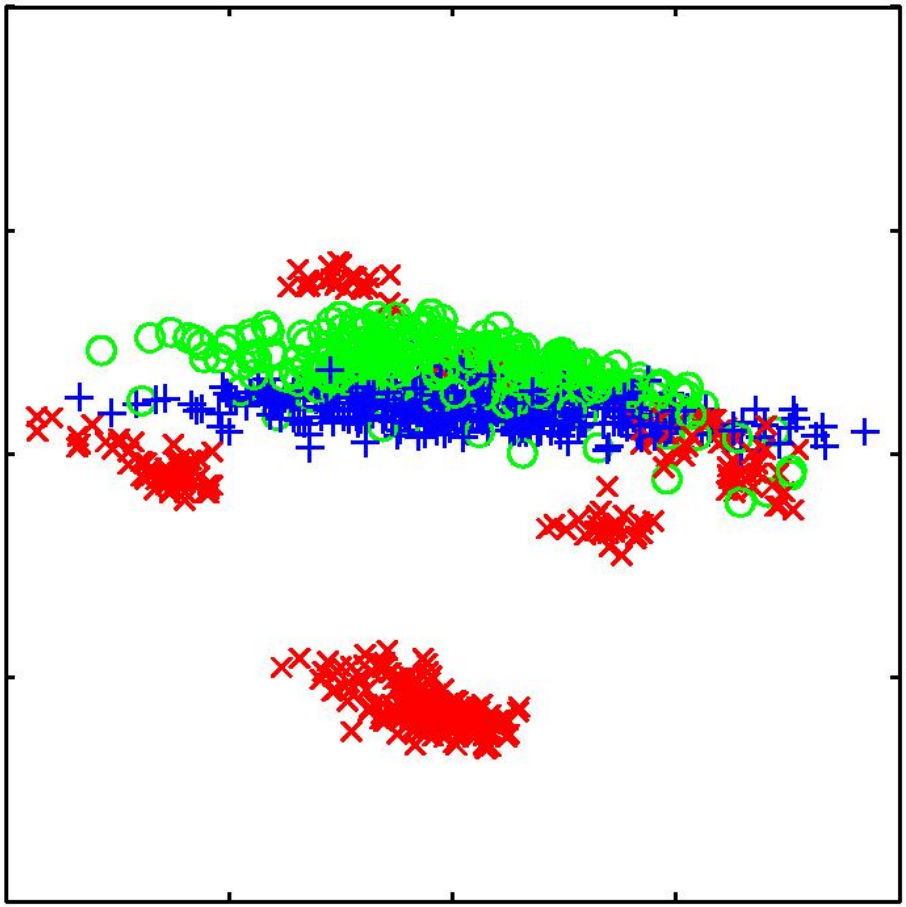
PCA-based normalization (leading to de-correlated dimensions)

1. Calulate the data covariance matrix S
2. Perform eigen-analysis: $\mathsf{SU} = \mathsf{UL}$
3. Transform the data: $\mathsf{y}_n = \mathsf{L}^{-1/2}\mathsf{U}^T(\mathsf{x}_n - \bar{\mathsf{x}})$
4. Now: $\frac{1}{N}\sum_{n=1}^{N} \mathsf{y}_n\mathsf{y}_n^T = \mathsf{I}$

The above data transformation is called *data whitening*.

PCA: Applications

Data visualization: Project the data to 2D or 3D.



Probabilistic PCA

We introduce variable z corresponding to the principal component subspace and the distributions:

$$p(\mathsf{z}) = \mathcal{N}(\mathsf{z}|0, \mathsf{I}) \tag{34}$$

$$p(\mathsf{x}|\mathsf{z}) = \mathcal{N}(\mathsf{x}|\mathsf{W}\mathsf{z} + \boldsymbol{\mu}, \sigma^2\mathsf{I}) \tag{35}$$

where $\mathsf{W} \in \mathbb{R}^{D\times M}$ and $\boldsymbol{\mu} \in \mathbb{R}^D$

The marginal distribution of x is $p(\mathsf{x}) = \mathcal{N}(\mathsf{x}|\boldsymbol{\mu}, \mathsf{C})$, where:

$$\boldsymbol{\mu} = \mathbb{E}[\mathsf{W}\mathsf{z} + \boldsymbol{\mu} + \epsilon] = \mathbb{E}[\mathsf{x}] \tag{36}$$

$$\begin{aligned} \mathsf{C} &= \mathbb{E}[(\mathsf{W}\mathsf{z} + \boldsymbol{\epsilon})(\mathsf{W}\mathsf{z} + \boldsymbol{\epsilon}^T)] \\ &= \mathbb{E}[\mathsf{W}\mathsf{z}\mathsf{z}^T\mathsf{W}^T] + \mathbb{E}[\boldsymbol{\epsilon}\boldsymbol{\epsilon}^T] = \mathsf{W}\mathsf{W}^T + \sigma^2\mathsf{I} \end{aligned} \tag{37}$$

Probabilistic PCA: Maximum Likelihood

The parameters of the Probabilistic PCA model are:

$$\boldsymbol{\mu}, \qquad \mathrm{W} \qquad \text{and} \qquad \sigma^2 \tag{38}$$

Given a set of data points $\mathrm{X} = \{\mathrm{x}_n\}$ we want to estimate the above parameters.

The log-likelihood function is:

$$\begin{aligned} \ln p(\mathrm{X}|\boldsymbol{\mu}, \mathrm{W}, \sigma^2) &= \sum_{n=1}^{N} \ln p(\mathrm{x}_n|\mathrm{W}, \boldsymbol{\mu}, \sigma^2) = -\frac{ND}{2}\ln(2\pi) - \frac{N}{2}\ln|\mathrm{C}| \\ &- \frac{1}{2}\sum_{n=1}^{N}(\mathrm{x}_n - \boldsymbol{\mu})^T \mathrm{C}^{-1}(\mathrm{x}_n - \boldsymbol{\mu}) \end{aligned} \tag{39}$$

Setting $\frac{\theta \ln p(\mathrm{X}|\boldsymbol{\mu},\mathrm{W},\sigma^2)}{\theta \boldsymbol{\mu}} = 0$, we get $\boldsymbol{\mu} = \bar{\mathrm{x}}$.

Probabilistic PCA: Maximum Likelihood

Maximization w.r.t. W and σ is more complex, but has an exact solution:

$$\mathrm{W}_{ML} = \mathrm{U}_M(\mathrm{L}_M - \sigma^2\mathrm{I})^{1/2}\mathrm{R} \tag{40}$$

$$\sigma^2_{ML} = \frac{1}{D-M}\sum_{i=M+1}^{D}\lambda_i \tag{41}$$

where:

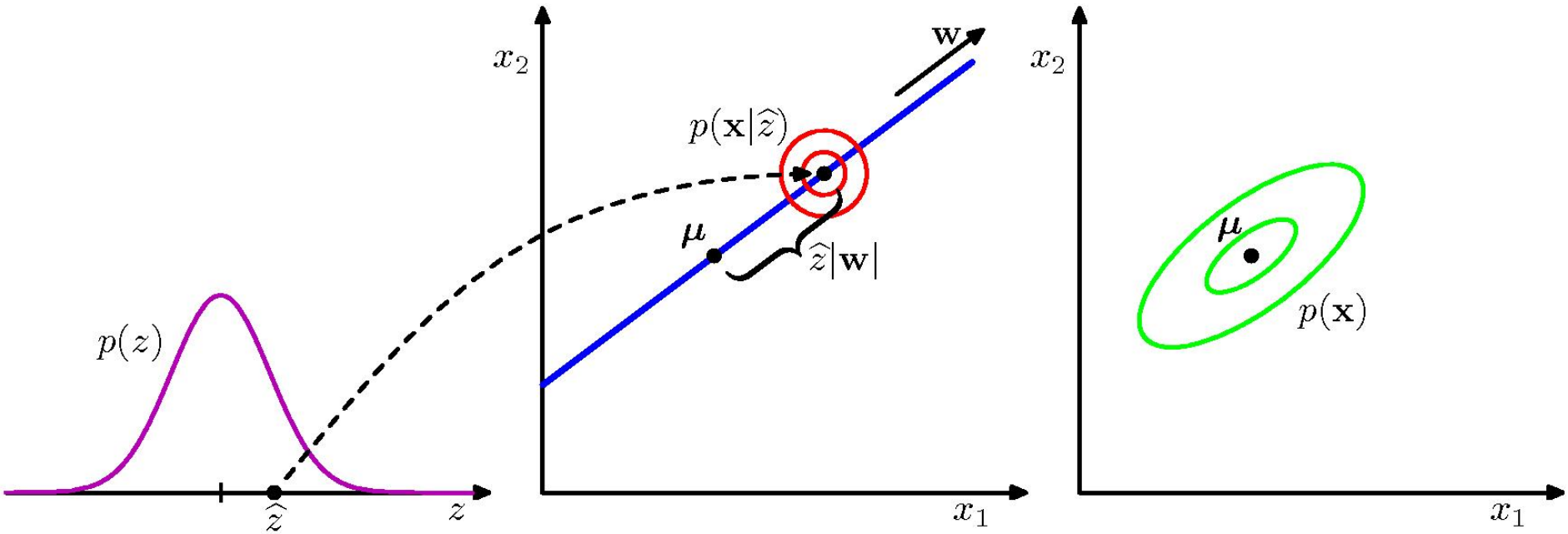
- $\mathrm{U}_M \in \mathbb{R}^{D\times M}$ is formed by the M eigenvectors corresponding to the largest eigenvalues $\lambda_i,\ i = 1,\ldots,M$
- $\mathrm{L}_M = diag(\lambda_i)$ is formed by the M largest eigenvalues
- $\mathrm{R} \in \mathbb{R}^{M\times M}$ is an arbitrary orthogonal matrix.

,

Probabilistic PCA: Generative model

Generate a data point x by:

1. drawing a random value $\hat{z}$ using $p(z)$
2. using $\hat{z}$, draw a vector x from an isotropic Gaussian $\mathcal{N}(\mathrm{W}\hat{z} + \boldsymbol{\mu}, \sigma^2\mathrm{I})$



EM for Probabilistic PCA

In cases of working on high-dimensional spaces (D is large) it may be easier to use an iterative method to estimate the parameters of PPCA.

The log-likelihood function for $\mathrm{X} = [\mathrm{x}_1, \ldots, \mathrm{x}_N]$ and $\mathrm{Z} = [\mathrm{z}_1, \ldots, \mathrm{z}_N]$ is:

$$\ln p(\mathrm{X}, \mathrm{Z}|\boldsymbol{\mu}, \mathrm{W}, \sigma^2) = \sum_{n=1}^{N} \{\ln p(\mathrm{x}_n|\mathrm{z}_n) + \ln p(\mathrm{z}_n)\} \tag{42}$$

When $\sigma^2 \to 0$, we obtain the standard PCA model.

EM for Probabilistic PCA

E step:

$$\mathbb{E}[z_n] = M^{-1} W^T (x_n - \bar{x}) \tag{43}$$

$$\mathbb{E}[z_n z_n^T] = \sigma^2 M^{-1} + \mathbb{E}[z_n]\,\mathbb{E}[z_n]^T \tag{44}$$

where $M = W^T W + \sigma^2 I$.

M step:

$$W_{new} = \left[\sum_{n=1}^{N} (x_n - \bar{x})\,\mathbb{E}[z_n]^T\right] \left[\sum_{n=1}^{N} \mathbb{E}[z_n z_n^T]\right]^{-1} \tag{45}$$

$$\begin{aligned} \sigma^2_{new} &= \frac{1}{ND} \sum_{n=1}^{N} \{ \|x_n - \bar{x}_n\|^2 - 2\,\mathbb{E}[z_n]^T W_{new} (x_n - \bar{x}) \\ &+ Tr(\mathbb{E}[z_n z_n^T] W_{new}^T W_{new}) \} \end{aligned} \tag{46}$$