Lecture 4: Principal Component Analysis (PCA)

School of Mathematical Sciences, Xiamen University

1. Setting

- The data set $\mathcal{D} = \left\{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \cdots, \mathbf{x}^{(p)}\right\}, \quad \mathbf{x}^{(j)} \in \mathbb{R}^n.$

  n: the number of components, or attributes, of the data.

  p: the number of data vectors.

- Given the data set $\mathcal{D}$, we address the questions on how to

  (1). remove possible hidden redundancies from the data,

  (2). reduce the dimensionality of the data,

  (3). visualize high-dimensional data maximizing the variability in the data set.

2. Removal of redundancies in data

- Given a data set $\mathcal{D}$, is it possible to determine its effective dimensionality, and if it is lower than the apparent dimensionality, how can one find a redundancy-free presentation?

- Start by arranging the data into a matrix,

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^{(1)} & \mathbf{x}^{(2)} & \cdots & \mathbf{x}^{(p)} \end{bmatrix} \in \mathbb{R}^{n \times p}.$$

Represent $\mathbf{X}$ in terms of its compact SVD

$$\mathbf{X} = \mathbf{U}_r \mathbf{\Sigma}_r \mathbf{V}_r^\top,$$

where $\mathbf{U}_r \in \mathbb{R}^{n \times r}, \mathbf{V}_r \in \mathbb{R}^{p \times r}$ are matrices with orthonormal columns,

$$\mathbf{U}_r = \begin{bmatrix} \mathbf{u}^{(1)} & \mathbf{u}^{(2)} & \cdots & \mathbf{u}^{(r)} \end{bmatrix}, \quad \mathbf{V}_r = \begin{bmatrix} \mathbf{v}^{(1)} & \mathbf{v}^{(2)} & \cdots \mathbf{v}^{(r)} \end{bmatrix}$$

and

$$\mathbf{\Sigma}_r = \text{diag}\{\sigma_1, \sigma_2, \cdots, \sigma_r\}, \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0.$$

It follows that

$$\mathbf{x}^{(j)} = \mathbf{X}\mathbf{e}_j = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^\top\mathbf{e}_j = \mathbf{U}_r\mathbf{z}^{(j)},$$

where

$$\mathbf{z}^{(j)} = \mathbf{\Sigma}_r\mathbf{V}_r^\top\mathbf{e}_j = \begin{bmatrix} \sigma_1 v_j^{(1)} \\ \sigma_2 v_j^{(2)} \\ \vdots \\ \sigma_r v_j^{(r)} \end{bmatrix}.$$

Hence, each data vector can be represented as a linear combination of the r left singular vectors,

$$\mathbf{x}^{(j)} = z_1^{(j)}\mathbf{u}^{(1)} + z_2^{(j)}\mathbf{u}^{(2)} + \cdots z_r^{(j)}\mathbf{u}^{(r)}, \quad 1 \leq j \leq p.$$

The scalars $z_k^{(j)}$, $1 \leq k \leq r$, are called the *principal components* of $\mathbf{x}^{(j)}$. The vectors $\mathbf{u}^{(j)}$ are called *feature vectors*.

- By the orthogonality of the singular vectors, the principal components can be computed as

$$z_k^{(j)} = (\mathbf{u}^{(k)})^\top \mathbf{x}^{(j)}.$$

Let

$$\mathbf{Z} = \begin{bmatrix} \mathbf{z}^{(1)} & \mathbf{z}^{(2)} & \dots & \mathbf{z}^{(p)} \end{bmatrix} \in \mathbb{R}^{r \times p}.$$

We have

$$\mathbf{Z} = \mathbf{U}_r^\top \mathbf{X},$$

and

$$\mathbf{X} = \mathbf{U}_r \mathbf{Z}.$$

If $r < n$, passing to principal components only compresses the data by removing redundancy.

3. PCA and model reduction

- Express the data matrix $\mathbf{X}$ as

$$\mathbf{X} = \sum_{\ell=1}^{r} \sigma_\ell \mathbf{u}^{(\ell)} (\mathbf{v}^{(\ell)})^\top.$$

It follows that

$$\mathbf{x}^{(j)} = \mathbf{X}\mathbf{e}_j = (\sigma_1 v_j^{(1)})\mathbf{u}^{(1)} + \cdots + (\sigma_r v_j^{(r)})\mathbf{u}^{(r)},$$

where

$$|z_\ell^{(j)}| = |\sigma_\ell v_j^{(\ell)}| \leq \sigma_\ell.$$

- Assume that the data are known to be corrupted by additive noise, i.e., any approximation $\widehat{\mathbf{x}}^{(j)} \in \mathbb{R}^n$ of the data vector $\mathbf{x}^{(j)}$ is within the noise level τ,

$$\|\mathbf{x}^{(j)} - \widehat{\mathbf{x}}^{(j)}\|_2 < \tau.$$

- Project $\mathbf{x}^{(j)}$ onto the subspace $\mathcal{H}_k = \text{span}\{\mathbf{u}^{(1)}, \cdots, \mathbf{u}^{(k)}\}$:

$$\mathbf{P}_k \mathbf{x}^{(j)} = \mathbf{U}_k \mathbf{U}_k^\top \mathbf{x}^{(j)} = (\sigma_1 v_j^{(1)})\mathbf{u}^{(1)} + \cdots + (\sigma_r v_j^{(k)})\mathbf{u}^{(k)}.$$

  It follows that

$$\begin{aligned} \|\mathbf{x}^{(j)} - \mathbf{P}_k \mathbf{x}^{(j)}\|_2^2 &= \left\| \sum_{\ell=k+1}^{r} z_\ell^{(j)} \mathbf{u}^{(j)} \right\|_2^2 \\ &= \sum_{\ell=k+1}^{r} (z_\ell^{(j)})^2 \leq \sum_{\ell=k+1}^{r} \sigma_\ell^2. \end{aligned}$$

- One strategy to clean the data from redundancy and retain only the components that are less likely to be contaminated by the noise is to replace $\mathbf{X}$ by $\mathbf{X} \approx \mathbf{U}_k \mathbf{\Sigma}_k \mathbf{V}_k^\top$, where k is chosen so that

$$\sum_{\ell=k}^{r} \sigma_\ell^2 \geq \tau^2 > \sum_{\ell=k+1}^{r} \sigma_\ell^2.$$

4. PCA and data visualization

- The PCA provides an effective way of visualizing high-dimensional data.
- Let $\mathbf{X} \in \mathbb{R}^{n \times p}$ denote the data matrix, and let $\mathbf{q} \in \mathbb{R}^n$ be a unit vector in the data space.
- The components of the data vectors $\mathbf{x}^{(j)}$ along the direction $\mathbf{q}$ are given by the components of the row vector

$$\mathbf{y}^\top = \mathbf{q}^\top \mathbf{X} = \begin{bmatrix} \mathbf{q}^\top \mathbf{x}^{(1)} & \mathbf{q}^\top \mathbf{x}^{(2)} & \cdots & \mathbf{q}^\top \mathbf{x}^{(p)} \end{bmatrix} = \begin{bmatrix} y_1 & y_2 & \cdots & y_p \end{bmatrix}.$$

- The *spread* of the components is defined as

$$\text{spread}(\mathbf{y}) = \left(\sum_{j=1}^{p} y_j^2 \right)^{1/2} = \|\mathbf{y}\|_2.$$

- Which direction $\mathbf{q}$ maximizes the spread?

It follows by the definition of induced matrix two-norm that

$$\max\{\text{spread}(\mathbf{y})\} = \max\{\|\mathbf{X}^\top \mathbf{q}\|_2 \mid \|\mathbf{q}\|_2 = 1\} = \|\mathbf{X}^\top\|_2.$$

Furthermore, $\max\{\text{spread}(\mathbf{y})\} = \sigma_1 = \|\mathbf{X}^\top \mathbf{u}^{(1)}\|_2$, that is, the projection direction that maximizes the spread of the data is given be the first left singular vector, $\mathbf{q} = \mathbf{u}^{(1)}$.

- Which projection direction, orthogonal to $\mathbf{u}^{(1)}$, gives the second largest spread?

  Subtract from the data matrix the rank-1 matrix corresponding to the first singular triplet,

$$\widetilde{\mathbf{X}} = \mathbf{X} - \sigma_1 \mathbf{u}^{(1)} (\mathbf{v}^{(1)})^\top = \sum_{j=2}^{r} \sigma_j \mathbf{u}^{(j)} (\mathbf{v}^{(j)})^\top,$$

  and define the spread of the components of $\widetilde{\mathbf{y}}^\top = \mathbf{q}^\top \widetilde{\mathbf{X}}$.

5. Data centering

- Subtract from each data point the data mean, so that average of the centered data is at the origin of the coordinate system.
- Let $\overline{\mathbf{x}}$ denote the mean value of the data, or the center of mass of the data,

$$\overline{\mathbf{x}} = \frac{1}{p}\sum_{j=1}^{p} \mathbf{x}^{(j)},$$

sometimes referred to also as the *centroid*. Defined the centered data matrix $\mathbf{X}_c$:

$$\begin{aligned}\mathbf{X}_c &= \begin{bmatrix}\mathbf{x}^{(1)} - \overline{\mathbf{x}} & \mathbf{x}^{(2)} - \overline{\mathbf{x}} & \cdots & \mathbf{x}^{(p)} - \overline{\mathbf{x}}\end{bmatrix} \\ &= \begin{bmatrix}\mathbf{x}_c^{(1)} & \mathbf{x}_c^{(2)} & \cdots & \mathbf{x}_c^{(p)}\end{bmatrix}.\end{aligned}$$

- Write $\mathbf{X}_c = \mathbf{U}_r\mathbf{\Sigma}_r\mathbf{V}_r^\top$, then we have

$$\mathbf{x}_c^{(j)} = \sum_{\ell=1}^{r} z_\ell^{(j)}\mathbf{u}_c^{(j)}, \quad z_\ell^{(j)} = (\mathbf{u}_c^{(\ell)})^\top \mathbf{x}_c^{(j)}.$$

- Example:

  The data are vectors in $\mathbb{R}^2$ whose components have been drawn independently from normal distributions with means and standard deviations $(4, 1)$ and $(4, 0.2)$, respectively, using the following MATLAB commands

```
% Center of the data cloud
c = [4;4];
% Gaussian cloud moved around the center
p = 1500;
X1 = randn(1,p);
X2 = 0.2*randn(1,p);
X = c*ones(1,p)+[X1;X2];
```

  Compute the SVD of both $\mathbf{X}$ and $\mathbf{X}_c$ and plot the two feature vectors for both cases. Explain your results.

6. Application: Handwritten digits from US postal envelopes

- Data set: the well-known MNIST data set. The data points consist of black and white pixel images of handwritten digits collected from US postal envelopes.
- Each image consists of 28×28 pixels, with grayscale values in the interval $[0, 1]$, with the value 1 representing white, and 0 black. Stack the pixel values of each image into a vector of length $n = 28^2 = 784$. The test sample contains $p = 10000$ images, and hence $\mathbf{X} \in \mathbb{R}^{784 \times 10000}$.
- To each vector, an annotation between 0 and 9 is given to indicate which digit the vector represents. Collect all the annotations into the vector $I \in \mathbb{N}^{10000}$.
- Write MATLAB code to visualize the data, selecting one digit of each type.
- PCA for the data