Machine Learning

Principal Component Analysis

Lecturer: *Dr*. Wan-Lei Zhao

Autumn Semester 2025

Contact: wlzhao@xmu.edu.cn

Outline

Opening Example (1)

- Given students record with two properties

Name	Gender	Height (cm)
Ada	0	168
Anna	0	163
Dennis	1	172
Diana	0	165
Jake	1	170
Tom	1	175

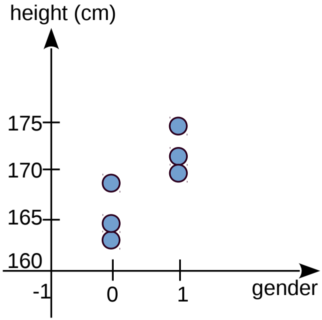


Figure: Visualization of the record.

- If you are allowed to keep single property
- Which one you will choose to best distinguish different students

Opening Example (2)

- Let's try to keep "gender" only, see what happens

Name	Gender
Ada	0
Anna	0
Dennis	1
Diana	0
Jake	1
Tom	1

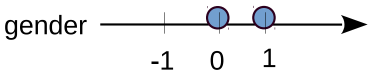


Figure: Visualization of the record.

- Observation: six points have been reduced to two points

Opening Example (3)

- Let's try to keep "height" only, see what happ

Name	Height (cm)
Ada	168
Anna	163
Dennis	172
Diana	165
Jake	170
Tom	175

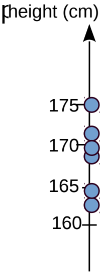


Figure: Visualization of the record.

- Observation: six points are still visible

Principal Content Analysis (PCA): the concept (1)

- Given a d-dimensional vector,
- We project it to lower dimension **m** via matrix **P**

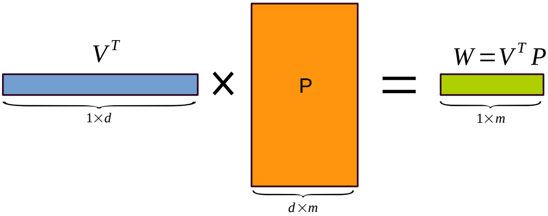


Figure: PCA projection.

- Aim: W preserves the **principal content** of vector V
- This is where the name comes

Principal Content Analysis (PCA): the concept (1)

- Given a *d*-dimensional vector,
- We project it to lower dimension **m** via matrix **P**

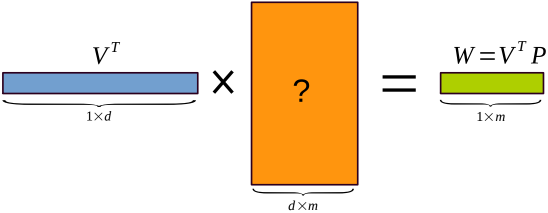
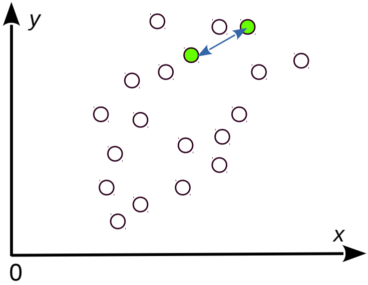


Figure: PCA projection.

- Problem: how to work out this project matrix (vector)

The idea (1)

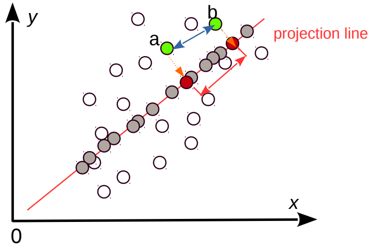
- Let's start with a simple example
- Given a group of 2D data



- We want to project them to 1D
- While preserving their relative distances as much as possible

The idea (2)

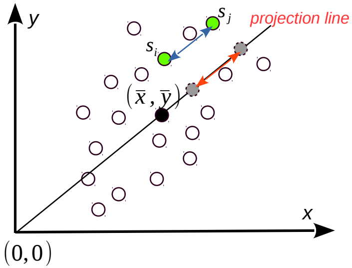
- Let's start with a simple example
- Given a group of 2D data



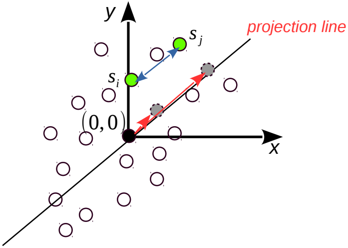
- We want to project them to 1D
- While preserving their relative distances as much as possible

The idea (3)

- Let's first move the axis origin to the center of the distribution



(a) $dist({s^{(i)}}^T \cdot P,\ {s^{(j)}}^T \cdot P)$

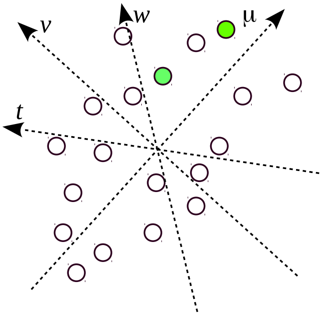


(b) ${s^{(i)}}^T \cdot P + {s^{(j)}}^T \cdot P$

- This is done by $x' = x - \bar{x}$, $y' = y - \bar{y}$
- $dist({s^{(i)}}^T \cdot P,\ {s^{(j)}}^T \cdot P) \approx {s^{(i)}}^T \cdot P + {s^{(j)}}^T \cdot P$, the bias from this new origin

The idea (4)

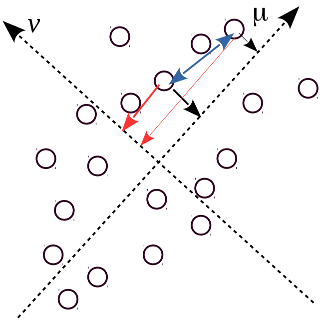
- We try to use different projection lines t, u, v and w



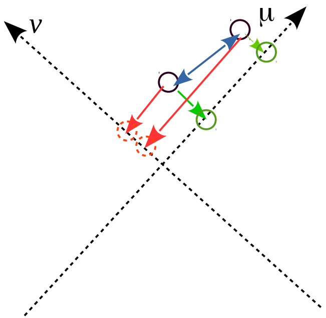
- Based on your observation, which one we should choose?

The model (1)

- Projection along μ is better



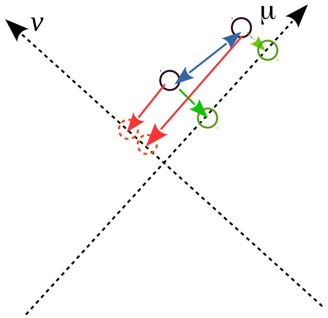
(c)



(d)

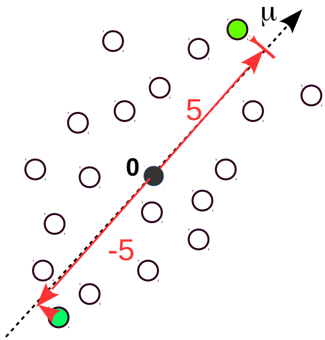
- We should find out projection line like μ that maximizes: $s^T\mu$

The model (2)



- Since we have $s^{(1)} \cdots s^{(i)} \cdots s^{(m)}$, we should maximize: $\sum_i^m s^{(i)^T} \mu$

The model (3)



- Two projections will be canceled by each other
- To alleviate this problem, we should $\sum_i^m s^{(i)^T}\mu \Rightarrow \sum_i^m (s^{(i)^T}\mu)^2$

The model (4)

- Given we have m samples $s^{(1)} \cdots s^{(i)} \cdots s^{(m)}$

$$\begin{array}{cc} \text{Maximize} & \sum_{i=1}^{m} ({s^{(i)}}^T \mu)^2 \\ \text{sub. to} & \mu^T \mu = 1 \end{array} \tag{1}$$

- The constraint $\mu^T \mu = 1$ is to require μ is in **unit length**

The model (5)

- Given there are m instances $s^{(1)} \dots s^{(i)} \dots s^{(m)}$

$$\begin{array}{ll} \text{Maximize} & \sum_{i=1}^{m} ({s^{(i)}}^T \mu)^2 \\ \text{sub. to} & \mu^T \mu = 1 \end{array} \quad (1)$$

- We further divide the target by m

$$\begin{array}{ll} \text{Maximize} & \frac{1}{m} \sum_{i=1}^{m} ({s^{(i)}}^T \mu)^2 \\ \text{sub. to} & \mu^T \mu = 1 \end{array} \quad (2)$$

- Otherwise, we can make the target bigger by increasing m

The model (6)

- Given there are m instances $s^{(1)} \cdots s^{(i)} \cdots s^{(m)}$

$$\begin{array}{cc} \text{Maximize} & \frac{1}{m}\sum_{i=1}^{m}({s^{(i)}}^T\mu)^2 \\ \text{sub. to} & \mu^T\mu = 1 \end{array} \tag{2}$$

- Above function is rewritten as

$$\begin{array}{cc} \text{Maximize} & \mu^T(\frac{1}{m}\sum_{i=1}^{m} s^{(i)}{s^{(i)}}^T)\mu \\ \text{sub. to} & \mu^T\mu = 1 \end{array} \tag{3}$$

- Can be easily solved by **Lagrangian multiplier**

The solution (1): solving by Lagrangian multiplier

- Given following problem:

$$\begin{aligned} \text{Maximize} \quad & \mu^T(\tfrac{1}{m}\textstyle\sum_{i=1}^{m} s^{(i)} {s^{(i)}}^T)\mu \\ \text{sub. to} \quad & \mu^T\mu = 1 \end{aligned} \quad (3)$$

- We define its Lagrangian function as:

$$L(\mu, \lambda) = \mu^T(\frac{1}{m}\sum s^{(i)} {s^{(i)}}^T)\mu + \lambda[\mu^T\mu - 1] \quad (4)$$

- Take partial derivative on μ and λ, we have

$$\frac{\partial L}{\partial \mu} = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T \mu - \lambda\mu = 0 \quad \tfrac{\partial L}{\partial \lambda} = \mu^T\mu - 1 = 0$$

The solution (2): solving by Lagrangian multiplier

- Take partial derivative on μ and λ, we have

$$\frac{\partial L}{\partial \mu} = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T \mu - \lambda\mu = 0 \quad \frac{\partial L}{\partial \lambda} = \mu^T \mu - 1 = 0$$

- With above results, we have

$$\begin{cases} \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T \mu = \lambda\mu \\ \mu^T \mu = 1 \end{cases} \tag{4}$$

The solution (3): solving by Lagrangian multiplier

$$\begin{cases} \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T \mu = \lambda\mu \\ \mu^T \mu = 1 \end{cases} \tag{4}$$

- Given $A = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T$

$$\begin{cases} A\mu = \lambda\mu \\ \mu^T \mu = 1 \end{cases} \tag{5}$$

- It is an **eigenvalue decomposition** problem on matrix **A**
  1. λs are the eigenvalues of matrix A
  2. μs are the eigenvectors of matrix A
- Question: what $A = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T$ looks like?

The solution (4): what A looks like

- Given $A = \frac{1}{m}\sum s^{(i)}{s^{(i)}}^T$, $s^{(i)} = [x^{(i)} \quad y^{(i)}]^T$

$$A = \frac{1}{m}\sum_{i=1}^{m} \begin{bmatrix} x^{(i)} \\ y^{(i)} \end{bmatrix} \begin{bmatrix} x^{(i)} & y^{(i)} \end{bmatrix} \tag{6}$$

$$\Downarrow$$

$$A = \frac{1}{m}\sum_{i=1}^{m} \begin{bmatrix} x^{(i)}.x^{(i)} & x^{(i)}.y^{(i)} \\ x^{(i)}.y^{(i)} & y^{(i)}.y^{(i)} \end{bmatrix} \tag{7}$$

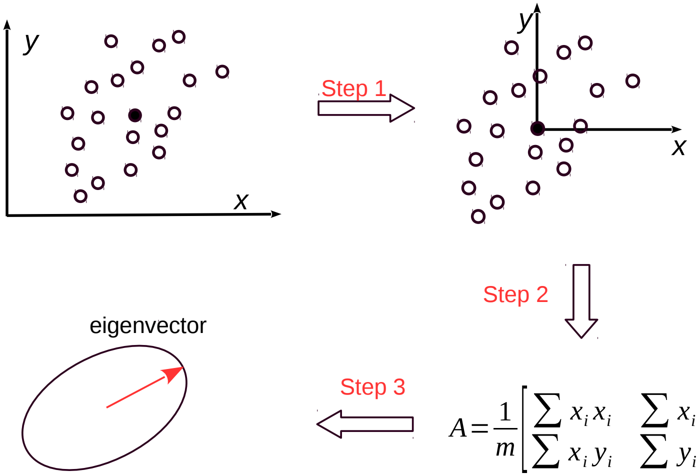
$$\Downarrow$$

$$A = \frac{1}{m} \begin{bmatrix} \sum_{i=1}^{m} x^{(i)}.x^{(i)} & \sum_{i=1}^{m} x^{(i)}.y^{(i)} \\ \sum_{i=1}^{m} x^{(i)}.y^{(i)} & \sum_{i=1}^{m} y^{(i)}.y^{(i)} \end{bmatrix} \tag{8}$$

- A is the covariance matrix of samples $s^{(1)} \cdots s^{(i)} \cdots s^{(m)}$

The procedure

- Now we summarize the PCA learning steps:



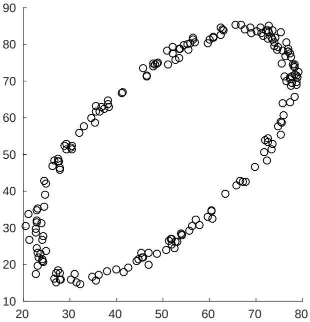
$$A = \frac{1}{m}\begin{bmatrix} \sum x_i x_i & \sum x_i y_i \\ \sum x_i y_i & \sum y_i y_i \end{bmatrix}$$

Outline

Implementation with Matlab (1)

```
dat   = load('dat1.txt');
scatter(dat(:,1), dat(:,2), 'k');
```

Listing 1: Load the data into memory



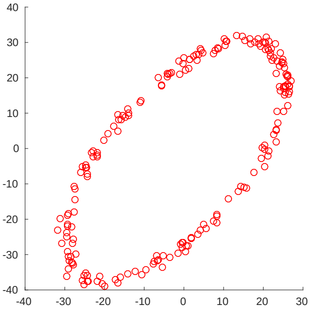
Implementation with Matlab (2)

```
dat   = load('dat1.txt');
scatter(dat(:,1), dat(:,2), 'k');

avg   = mean(dat)
N     = max(size(dat));
avg   = repmat(avg, N, 1);
mdat = (dat-avg); %Step 1
scatter(mdat(:,1), mdat(:,2), 'r');

```

Listing 2: Normalize the data



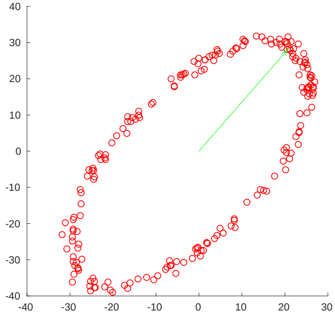
Implementation with Matlab (3)

```
dat   = load('dat1.txt');
scatter(dat(:,1), dat(:,2), 'k');

avg   = mean(dat)
N     = max(size(dat));
avg   = repmat(avg, N, 1);
mdat  = (dat-avg);
scatter(mdat(:,1), mdat(:,2), 'r');
hold on;

covr    = (dat-avg)/sqrt(N-1); %Step 1
covr    = covr'*covr; %Step 2
[P, e]  = eigs(covr); %Step 3
quiver(0,0,35*P(2,1), 35*P(2,2), 0, 'g')
```

Listing 3: Visualize the projection vector



Extended to d-dimensional case (1)

$$\begin{cases} \frac{1}{m}\sum s^{(i)} {s^{(i)}}^{T} \mu = \lambda\mu \\ \mu^{T}\mu = 1 \end{cases} \tag{4}$$

- Given $A = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^{T}$

$$\begin{cases} A\mu = \lambda\mu \\ \mu^{T}\mu = 1 \end{cases} \tag{5}$$

- It is an **eigenvalue decomposition** problem on matrix **A**
  1. λs are the eigenvalues of matrix A
  2. μs are the eigenvectors of matrix A
- Given $S_{d\times m}$, one column in S keeps a $s^{(i)}$
- How the problem looks like

Extended to d-dimensional case (2)

- Given $S_{d\times m}$, one column in S keeps a $s^{(i)}$
- How the problem looks like

$$\begin{cases} \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T \mu = \lambda\mu \\ \mu^T\mu = 1 \end{cases} \quad (4)$$

- Given $A = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T$

$$\begin{cases} A\mu = \lambda\mu \\ \mu^T\mu = 1 \end{cases} \quad (5)$$

$$\begin{cases} \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T \mu = \lambda\mu \\ \mu^T\mu = 1 \end{cases} \quad (6)$$

- Given
  $A = \frac{1}{m}\sum s^{(i)} {s^{(i)}}^T = \frac{1}{m}SS^T$

$$\begin{cases} \frac{1}{m}SS^T\mu = \lambda\mu \\ \mu^T\mu = 1 \end{cases} \quad (7)$$

Extended to d-dimensional case (3)

- Given $A = \frac{1}{m}\sum s^{(i)}{s^{(i)}}^T = \frac{1}{m}SS^T$

$$\begin{cases} \frac{1}{m}SS^T\mu = \lambda\mu \\ \mu^T\mu = 1 \end{cases} \tag{7}$$

$$C = \frac{1}{m}\begin{bmatrix} \sum_{i=1}^{m} s_1^{(i)} \cdot s_1^{(i)} & \sum_{i=1}^{m} s_1^{(i)} \cdot s_2^{(i)} & \cdots & \sum_{i=1}^{m} s_1^{(i)} \cdot s_d^{(i)} \\ \sum_{i=2}^{m} s_2^{(i)} \cdot s_1^{(i)} & \sum_{i=1}^{m} s_2^{(i)} \cdot s_2^{(i)} & \cdots & \sum_{i=1}^{m} s_2^{(i)} \cdot s_d^{(i)} \\ . & . & \cdots & . \\ \sum_{i=1}^{m} s_d^{(i)} \cdot s_1^{(i)} & \sum_{i=1}^{m} s_d^{(i)} \cdot s_2^{(i)} & \cdots & \sum_{i=1}^{m} s_d^{(i)} \cdot s_d^{(i)} \end{bmatrix}_{d \times d} \tag{8}$$

- C is called the second moment matrix or covariance matrix