Machine learning numpy, school of AI Kuala Lumpur

Husein Zolkepli

Bayes theorem text classification

Likelihood probability, probability of vector X when class C

Prior probability, probability of class C going to occur

$$P(C \mid X) = \frac{P(X|C)\,P(C)}{P(X)}$$

Posterior probability, probability of class C going to happen when vector is X

Marginal probability, probability of vector X, most of the case, its unobserve

Rebranding bayes theorem

$$P(C \mid X) \propto P(X \mid C)\,P(C)$$

Rebranding bayes theorem

$$P(C \mid X) \propto P(X \mid C)\, P(C)$$

$$P(C \mid X) = k\, P(X \mid C)\, P(C)$$

Rebranding bayes theorem

$$P(C \mid X) \propto P(X \mid C)\, P(C)$$

$$P(C \mid X) = k\, P(X \mid C)\, P(C)$$

$$P(C' \mid X) = k\, P(X \mid C')\, P(C')$$

Rebranding bayes theorem

$$P(C \mid X) \propto P(X \mid C)\, P(C)$$

$$P(C \mid X) = k\, P(X \mid C)\, P(C)$$

$$P(C' \mid X) = k\, P(X \mid C')\, P(C')$$

$$1 = k\, P(X \mid C)\, P(C) + k\, P(X \mid C')\, P(C')$$

Rebranding bayes theorem

$$k = \frac{1}{P(X|C)\,P(C) + P(X|C')\,P(C')}$$

$$k = \frac{1}{P(X)}$$

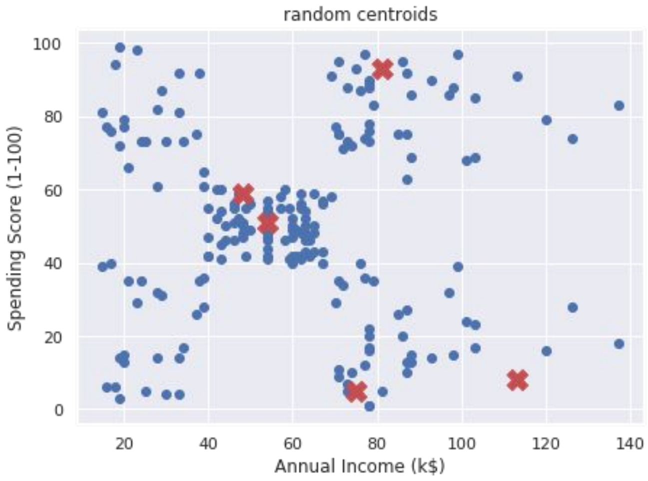
$$P(C \mid X) = \frac{P(X|C)\,P(C)}{P(X|C)\,P(C) + P(X|C')\,P(C')}$$

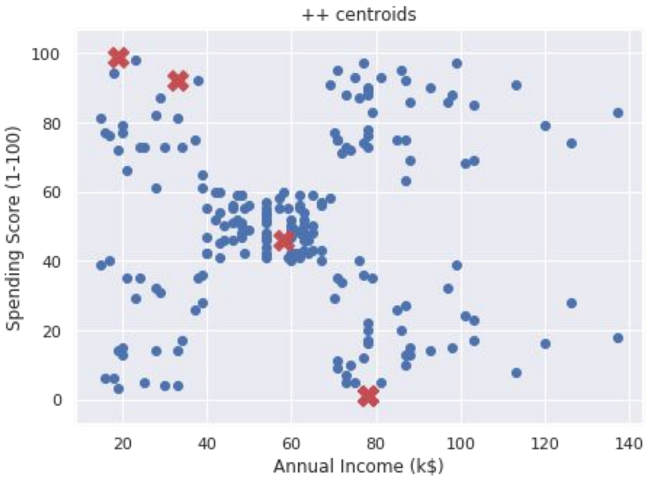
Text classification

index	i	like	chicken	meat	label
1	1	1	1	0	0
2	1	1	0	1	1

$$P(like \mid 0) = \frac{1+count}{\sum count+|V|}$$

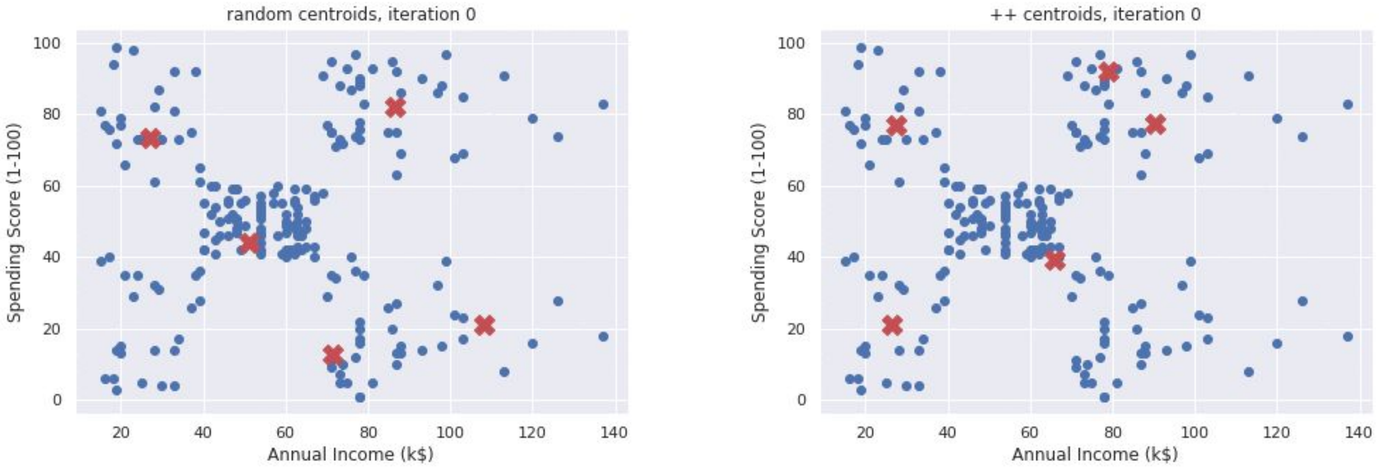
Kmean





1. Initiate random centroids, or use kmeans++.

Kmean



2. Keep iterating to calculate distances between individuals and centroids, and mean clustered individuals.

Kmean

3. To calculate ELBOW,

Iterate N K-means, every iteration, calculate sum of distances between centroids and grouped individuals, and plot.

Principal Component Analysis

Principal Component Analysis

1. Visualization

Principal Component Analysis

1. Visualization

Height, x	Weight, y	Bmi, z	Score, a	Hair length, b	Age, c	Steps, d

Principal Component Analysis

1. Visualization

Height, x	Weight, y	Bmi, z	Score, a	Hair length, b	Age, c	Steps, d

It does not makes sense if you want to plot this table into a vector space, we have 7 dimensions!

Principal Component Analysis

2. Reduce noise

Let say you want to study stress level of a student, based on,

Principal Component Analysis

2. Reduce noise

Let say you want to study stress level of a student, based on,

Height, x	Weight, y	Bmi, z	Score, a	Hair length, b	Age, c	Steps, d

Not all these 7 dimensions bring important information! We want to reject some attributes.

Principal Component Analysis

2. Reduce noise

Let say you want to study stress level of a student, based on,

Height, x	Weight, y	Bmi, z	Score, a	Hair length, b	Age, c	Steps, d

Not all these 7 dimensions bring important information! We want to reject some attributes. Maybe 7 does not hurt much. What happen if you have 512 * 512 * 3 (image) dimension?! insane!

Principal Component Analysis

3. Reduce memory (computer science)

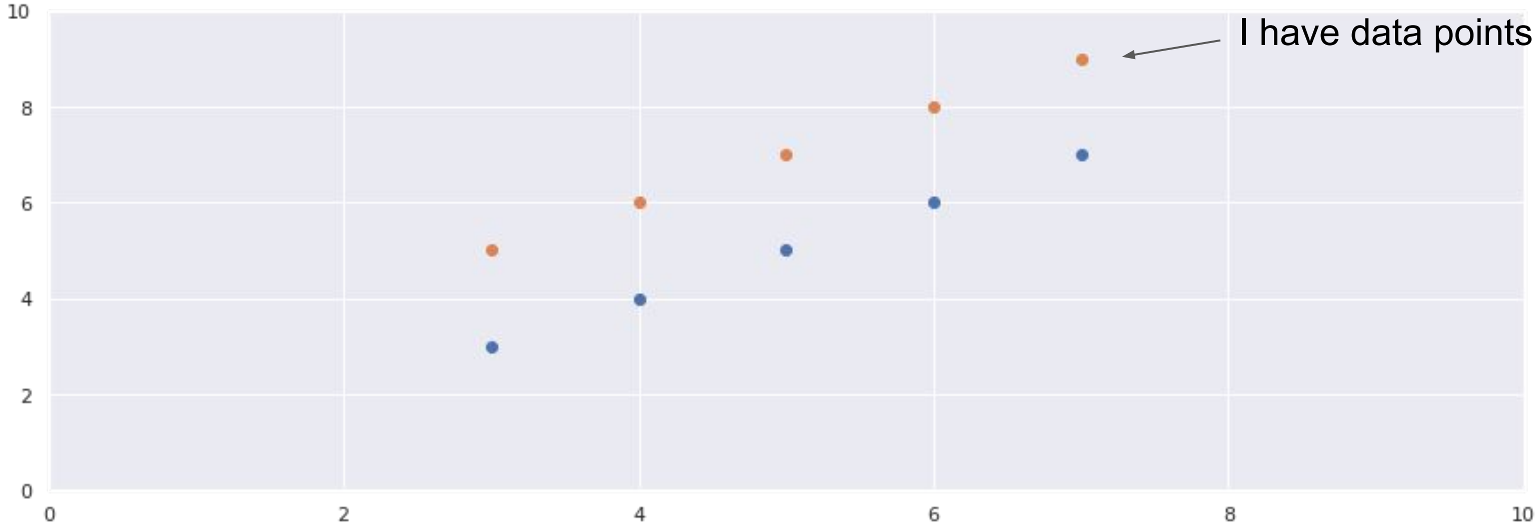
Height, x	Weight, y	Bmi, z	Score, a	Hair length, b	Age, c	Steps, d

Let say a float took 1 bytes, we have 7 columns and 1 billion of rows.

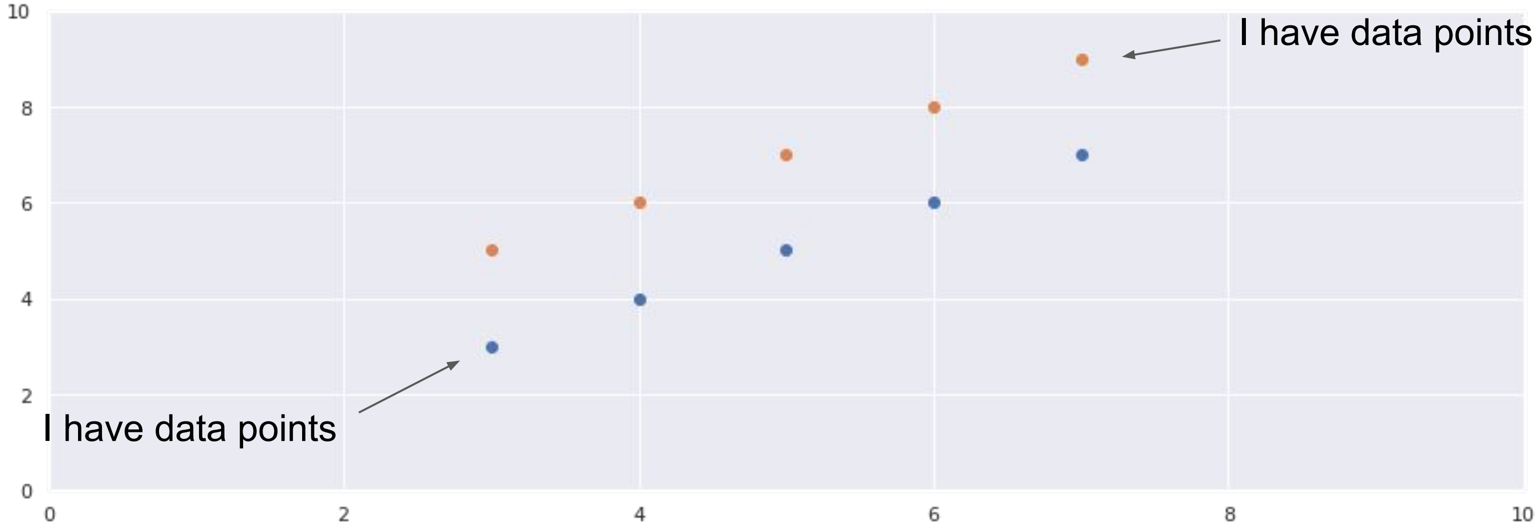
7 * 1,000,000,000 * 1 = 7,000,000,000 bytes == 70 GB!

Drop a column will save us 10 GB of memory!

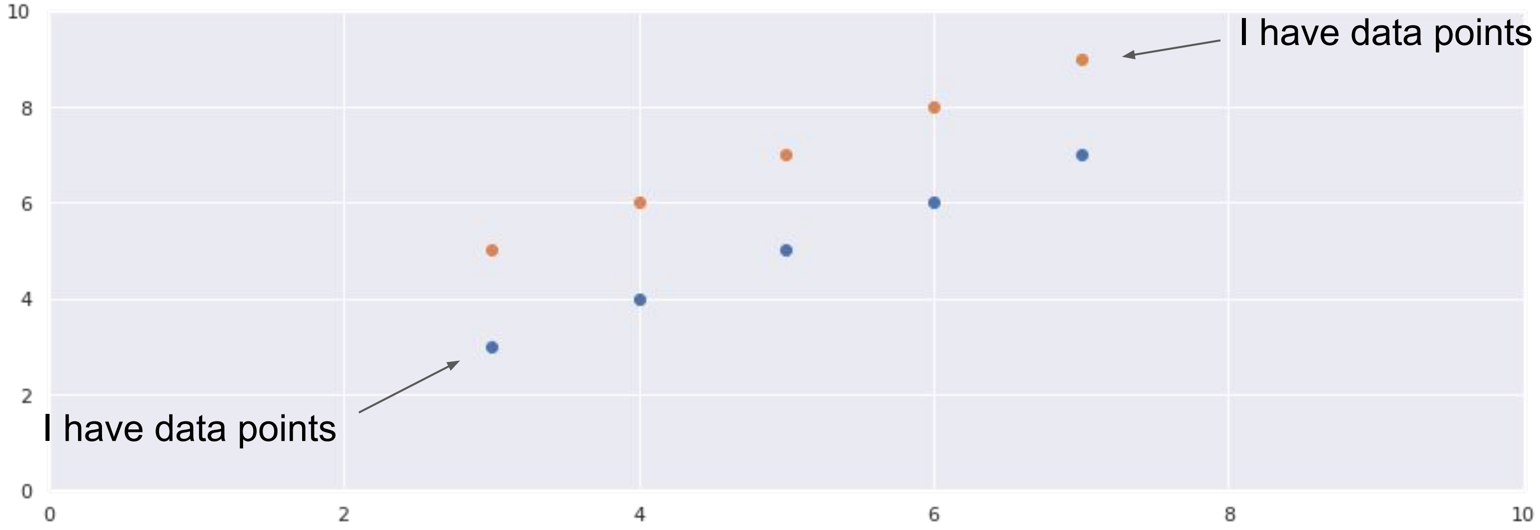
Principal Component Analysis



Principal Component Analysis

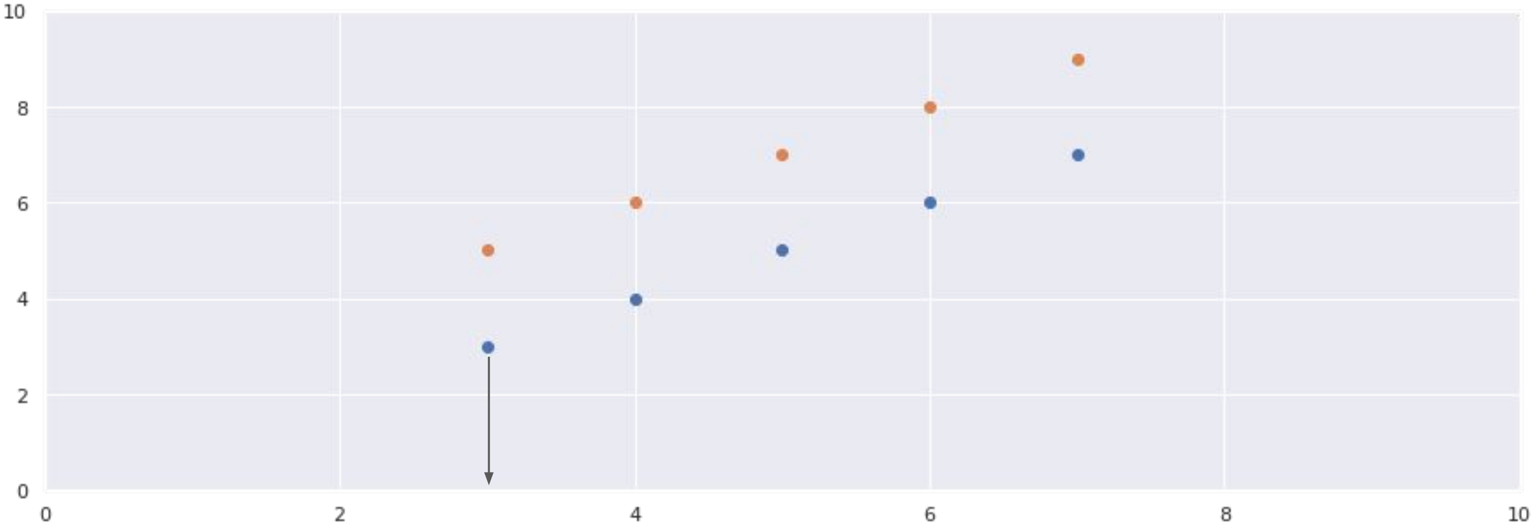


Principal Component Analysis

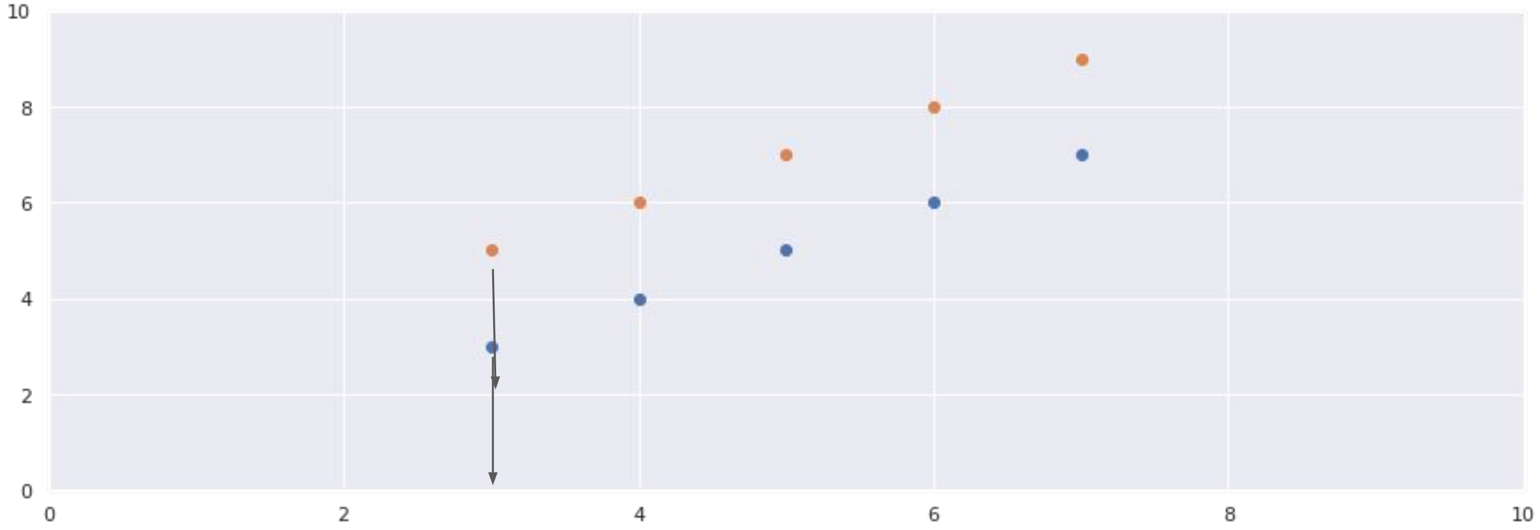


Let say, this plane is R^n , we only visualize it on R^2 , I want to visualize the data points at axis-0, which is x-axis.

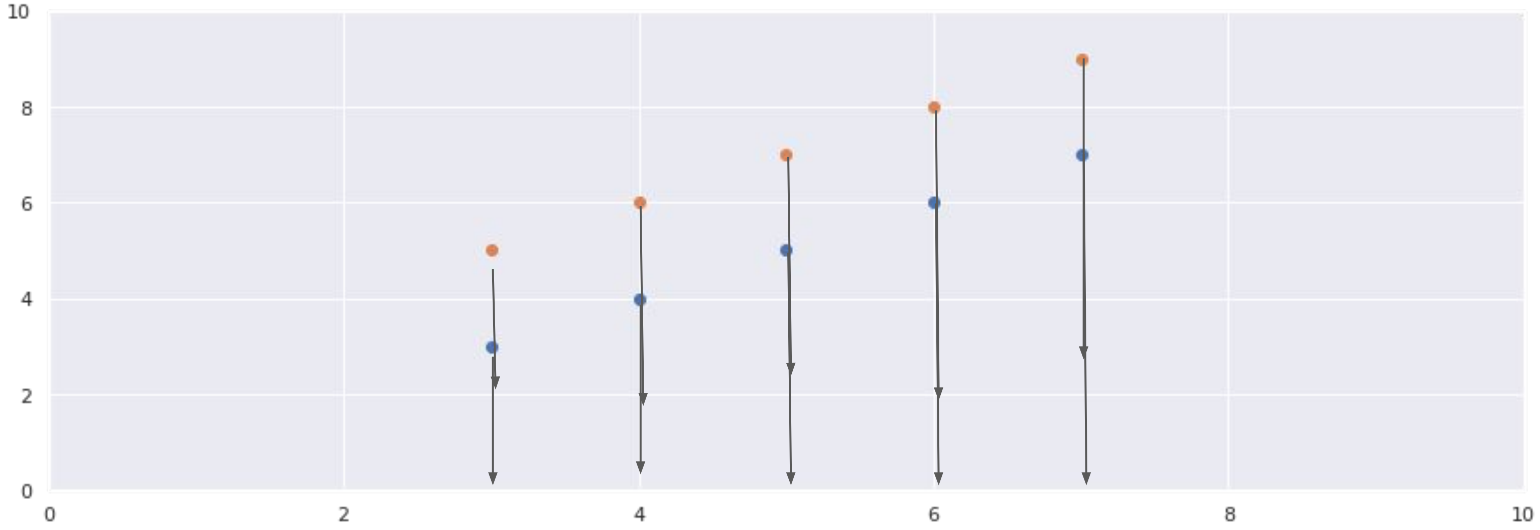
Principal Component Analysis



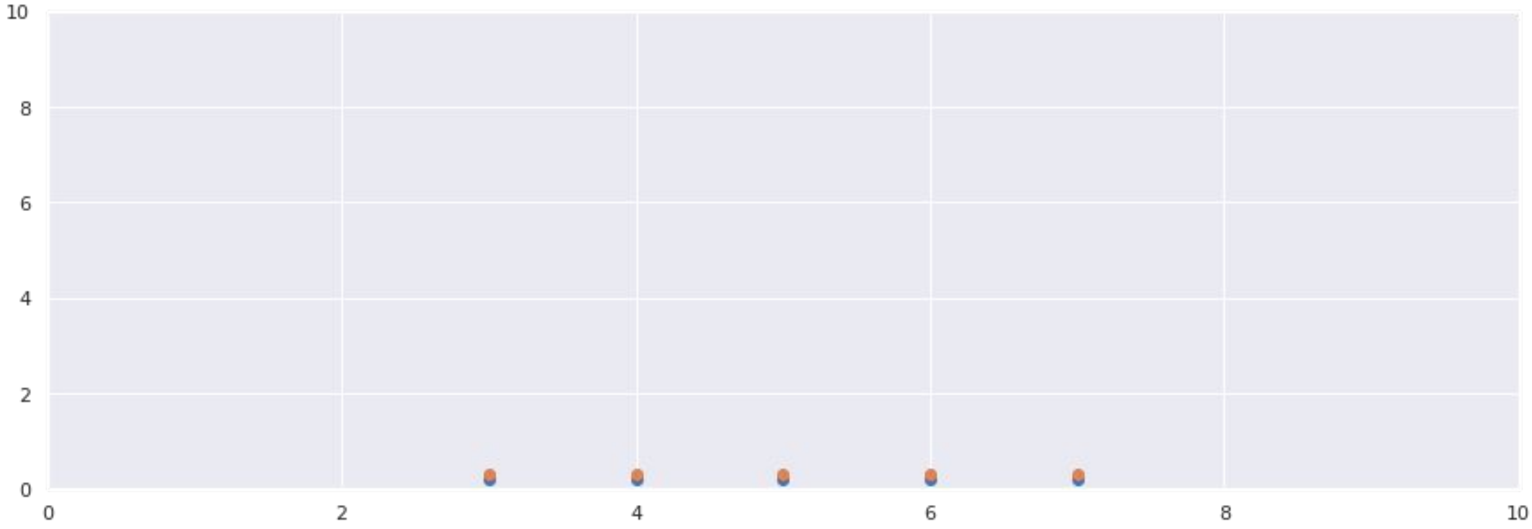
Principal Component Analysis



Principal Component Analysis

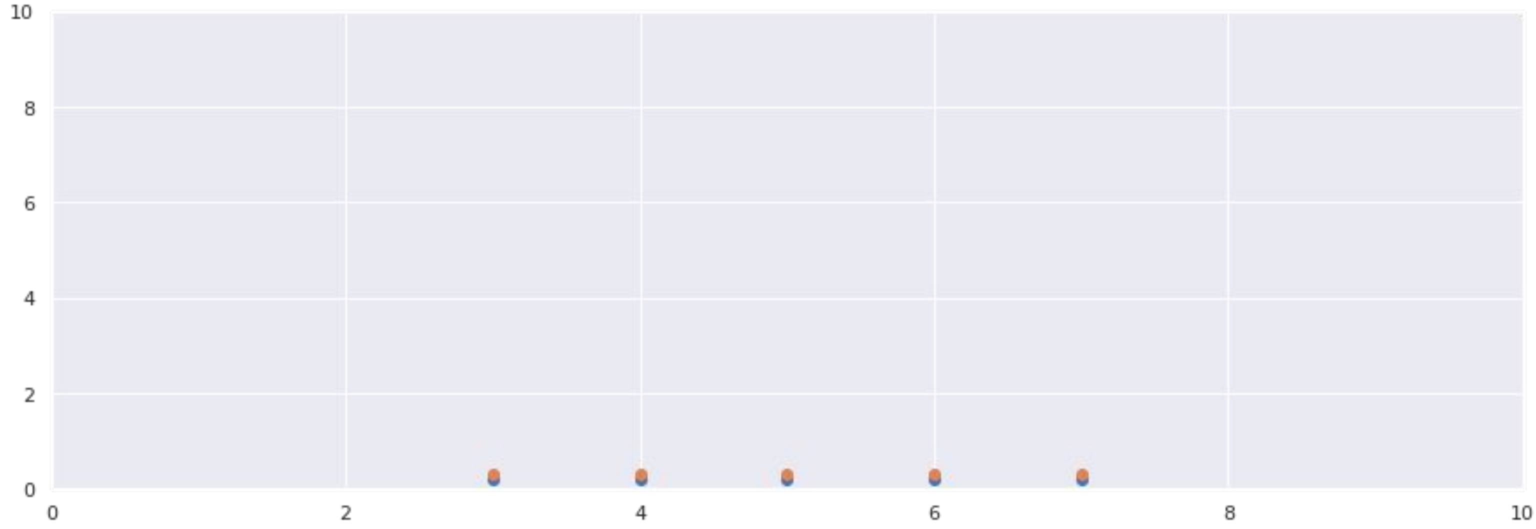


Principal Component Analysis



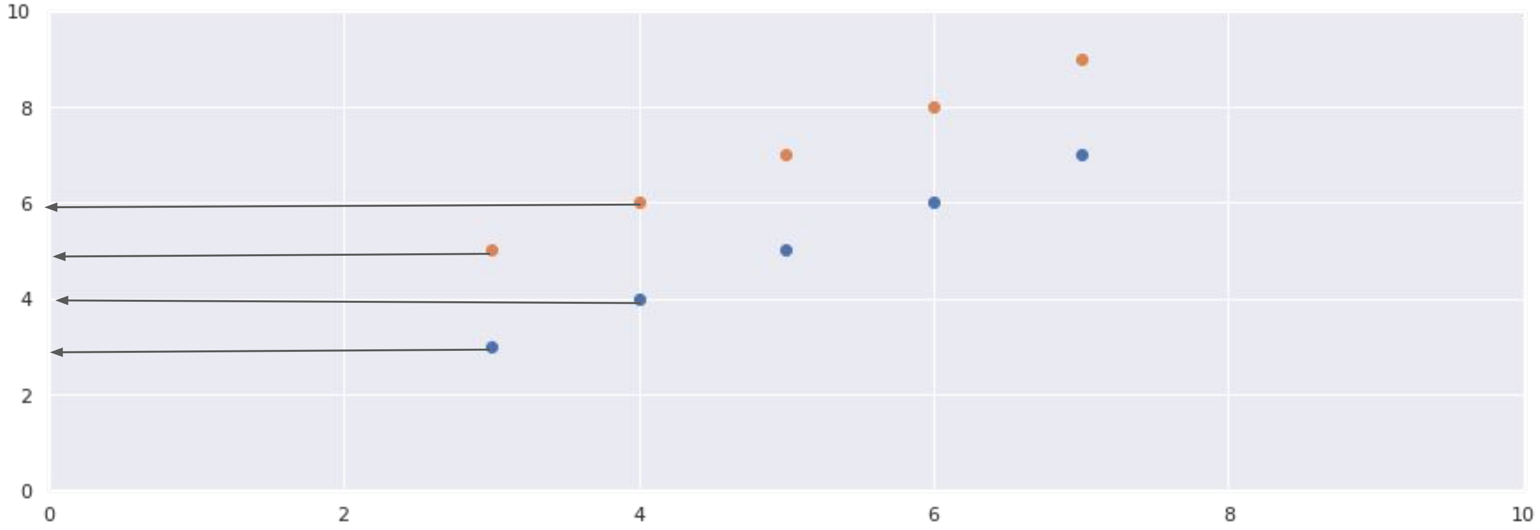
We cannot distinguish between oranges and blues!

Principal Component Analysis

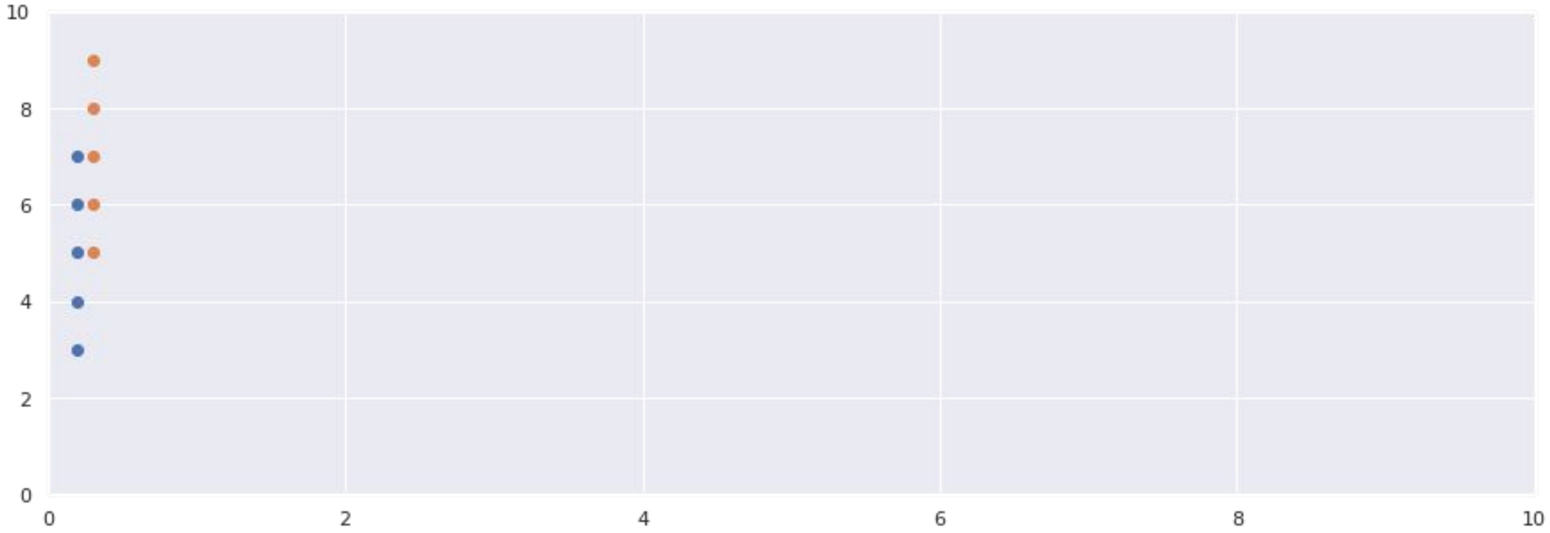


We cannot distinguish between oranges and blues! How about axis-2, which is, axis-y?

Principal Component Analysis

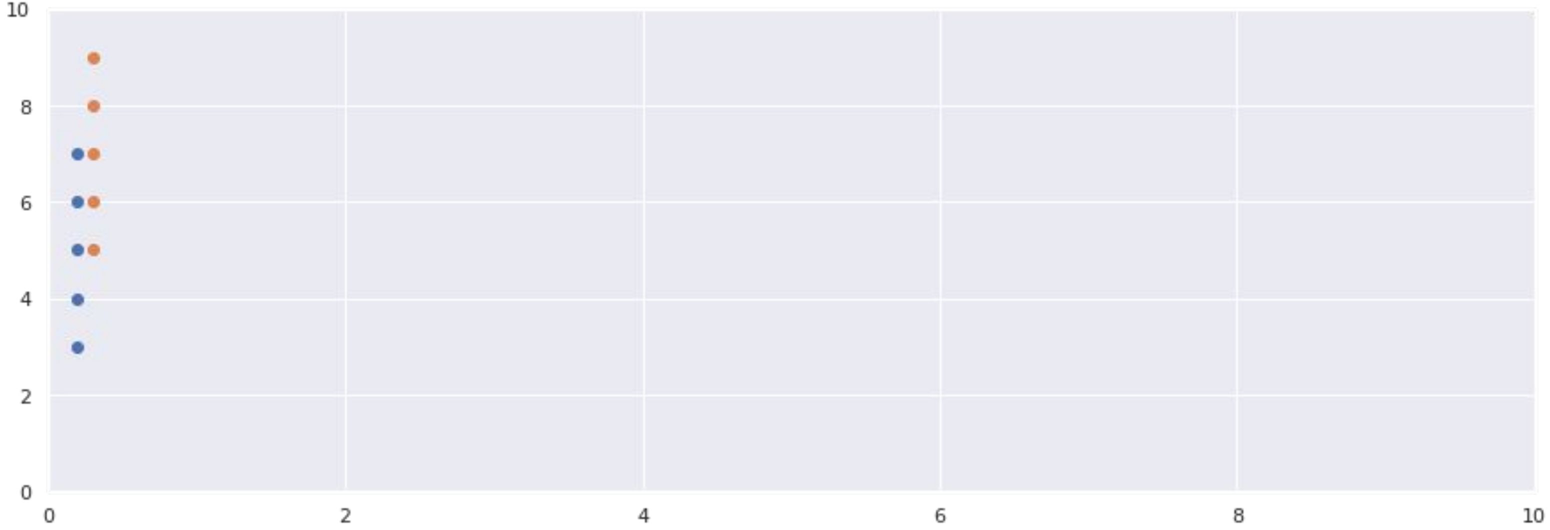


Principal Component Analysis



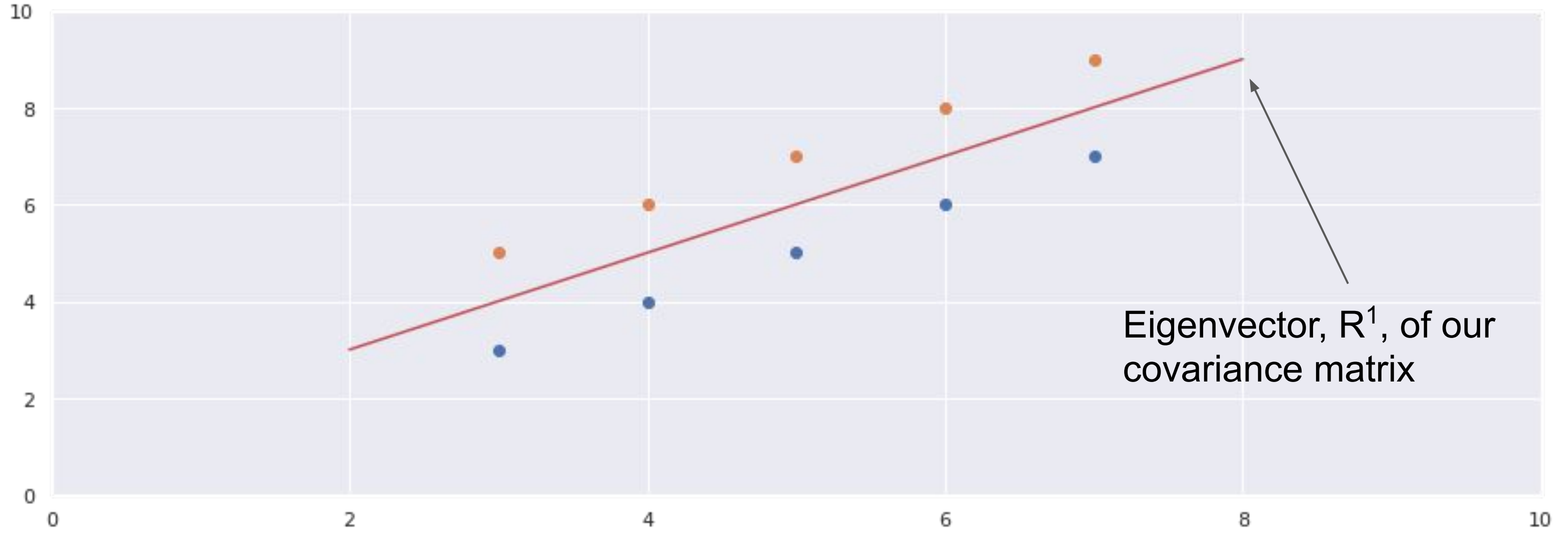
It is quite okay, just a few data points overlapped each others.

Principal Component Analysis

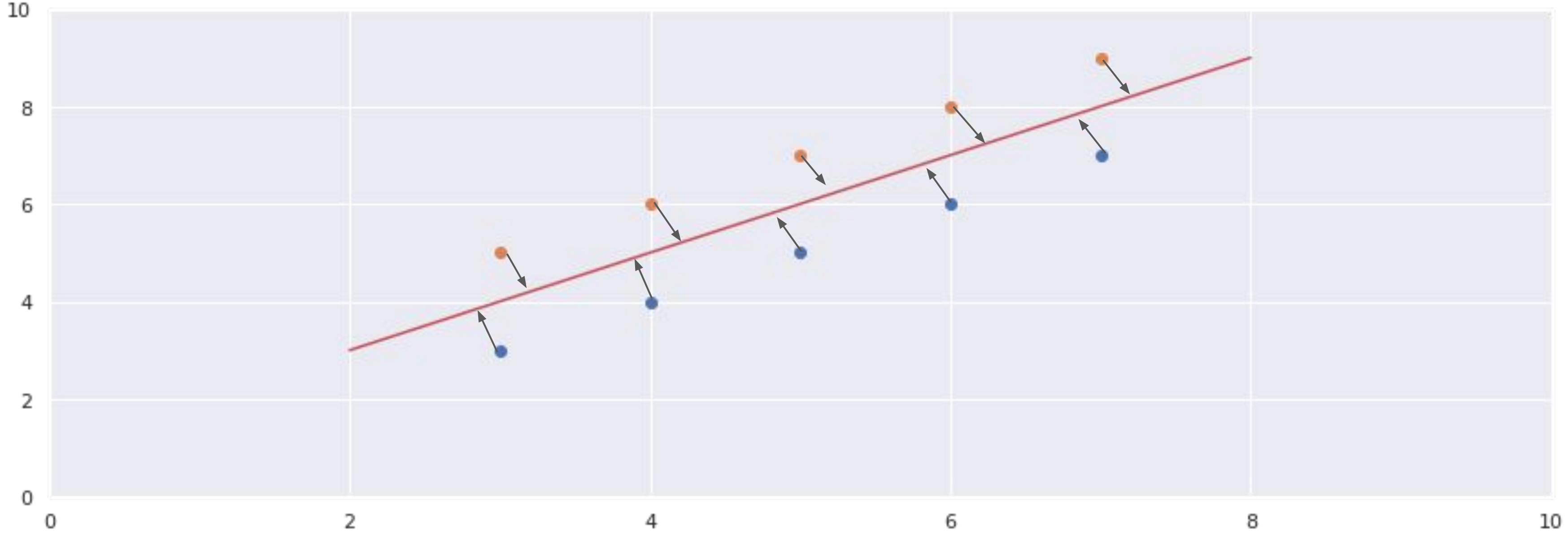


It is quite okay, just a few data points overlapped each others. **But we don't overlapping right?!**

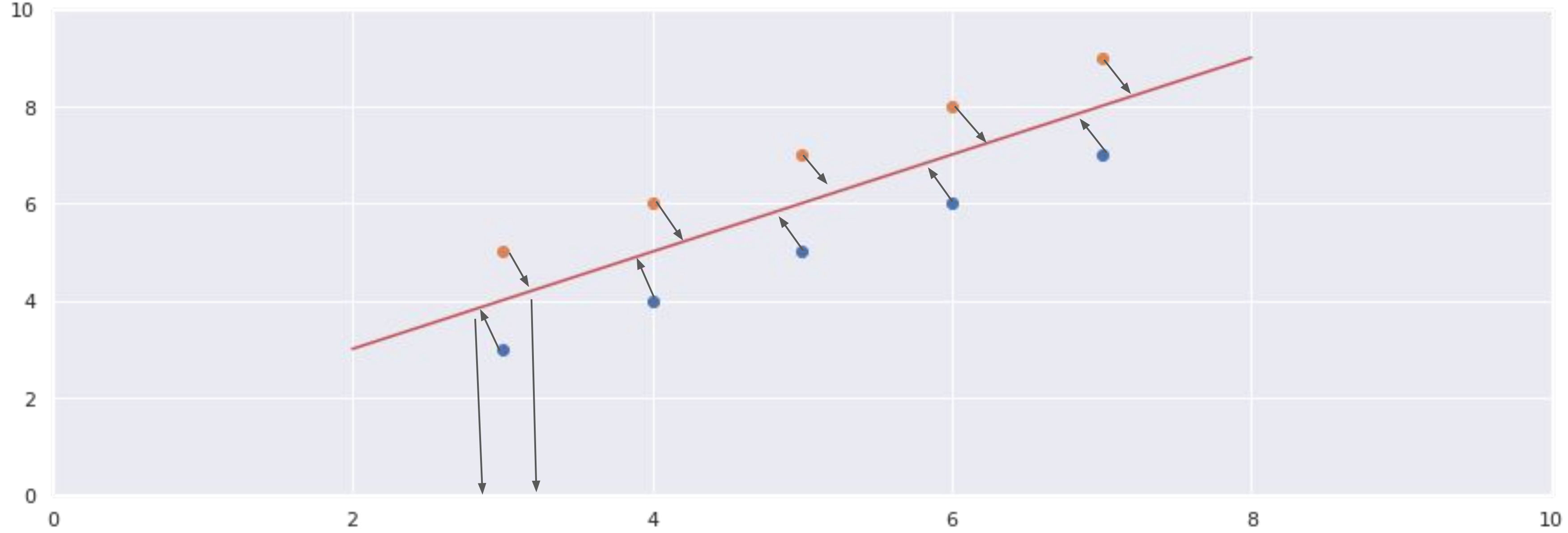
Principal component analysis



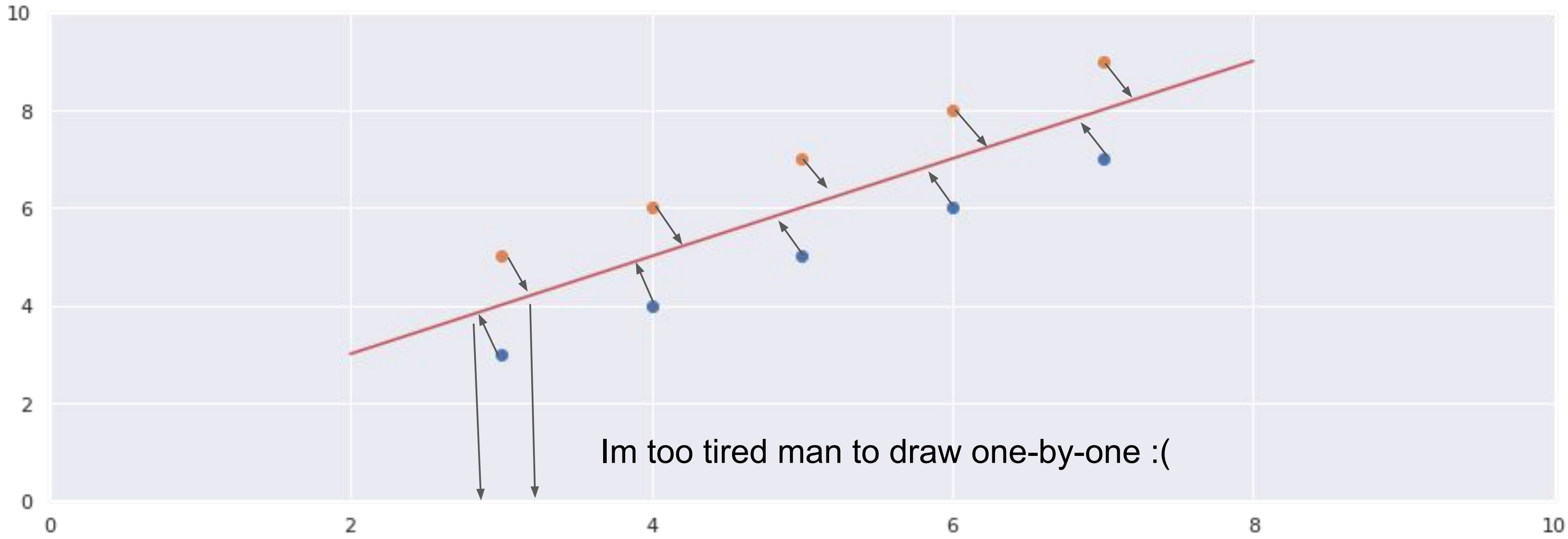
Principal component analysis



Principal component analysis

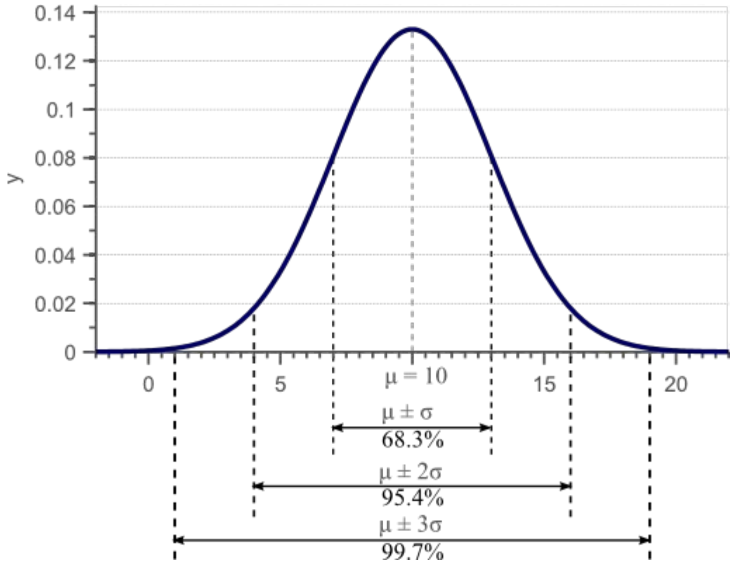


Principal component analysis



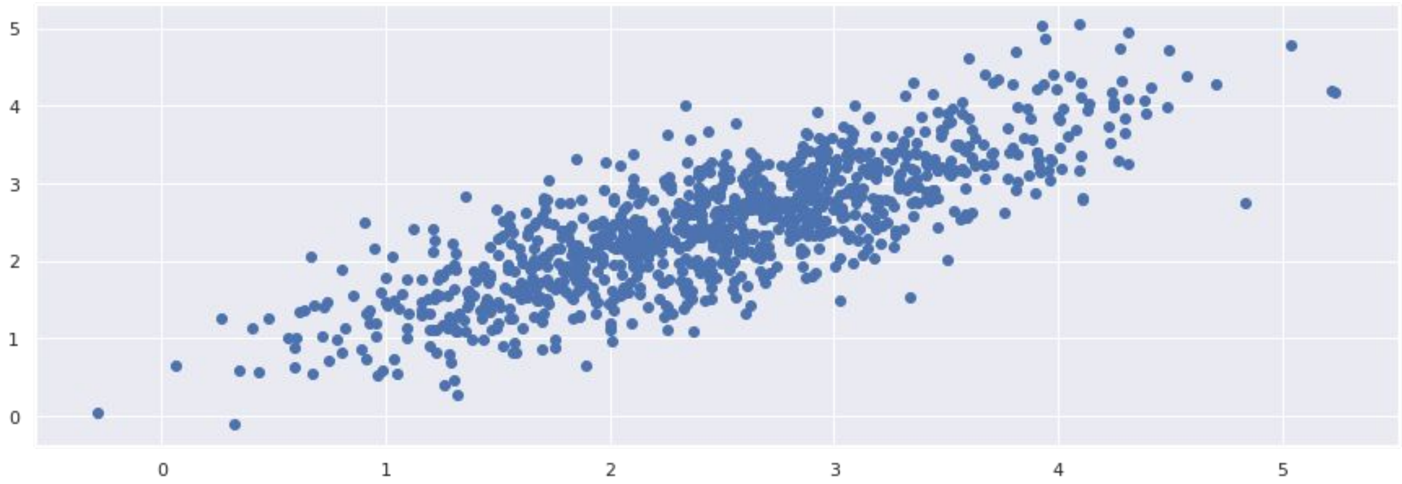
Principal component analysis

How to make sense of it?



$$s^2 = \frac{\Sigma X^2 - \frac{(\Sigma X)^2}{N}}{N-1}$$

Principal component analysis



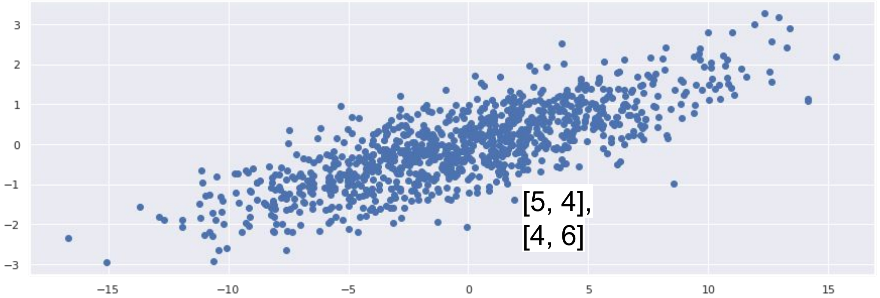
$$\mathrm{conv}(x, y) = \frac{\sum (x - \overline{x})(y - \overline{y})}{n}$$

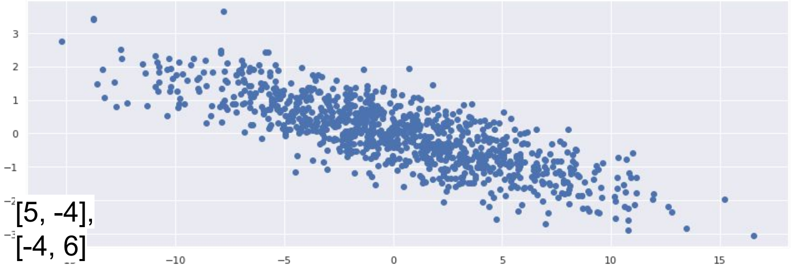
Principal component analysis

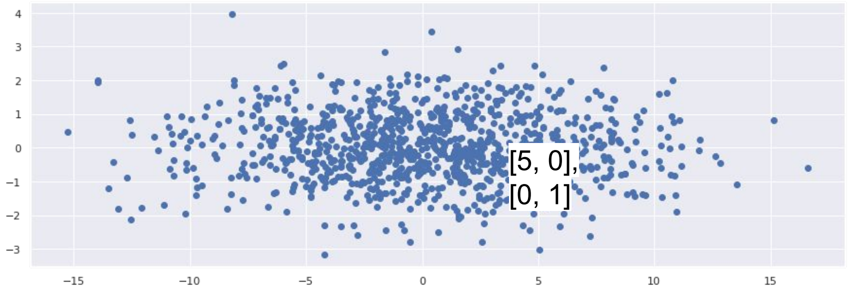
$$\Sigma = \begin{bmatrix} \sigma(x,x) & \sigma(x,y) \\ \sigma(y,x) & \sigma(y,y) \end{bmatrix}$$

$$X.T \cdot X$$

Principal component analysis

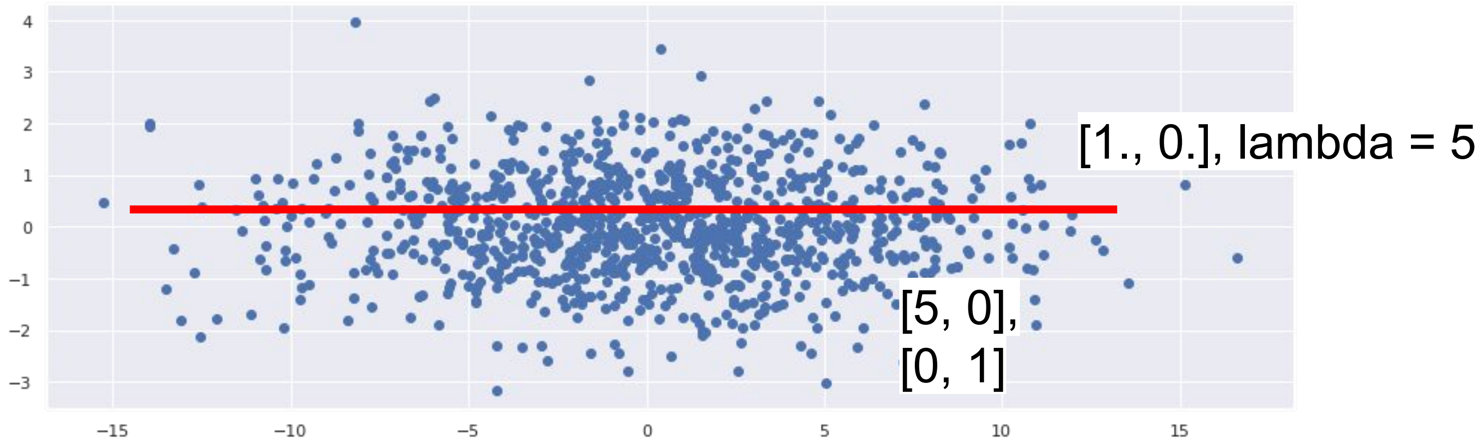






Value 1 is y axis, 0 correlation

Principal component analysis




```
l, v = np.linalg.eig(np.array([[5,0],[0,1]]))
l, v
(array([5., 1.]), array([[1., 0.],
                          [0., 1.]]))
```