AARHUS
UNIVERSITET

Optimization
and
Data Analytics

Alexandros Iosifidis
@
Aarhus University, Department of Engineering

Nearest Prototype Classification

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels l_i, we can define the class mean vectors:

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{i, l_i = k} \mathbf{x}_i, \; k = 1, \ldots, K$$

We use $\boldsymbol{\mu}_k$, k=1,…,K to represent the K classes.

Nearest Prototype Classification

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels l_i, we can define the class mean vectors:

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{i, l_i = k} \mathbf{x}_i, \; k = 1, \ldots, K$$

We use $\boldsymbol{\mu}_k$, k=1,…,K to represent the K classes.

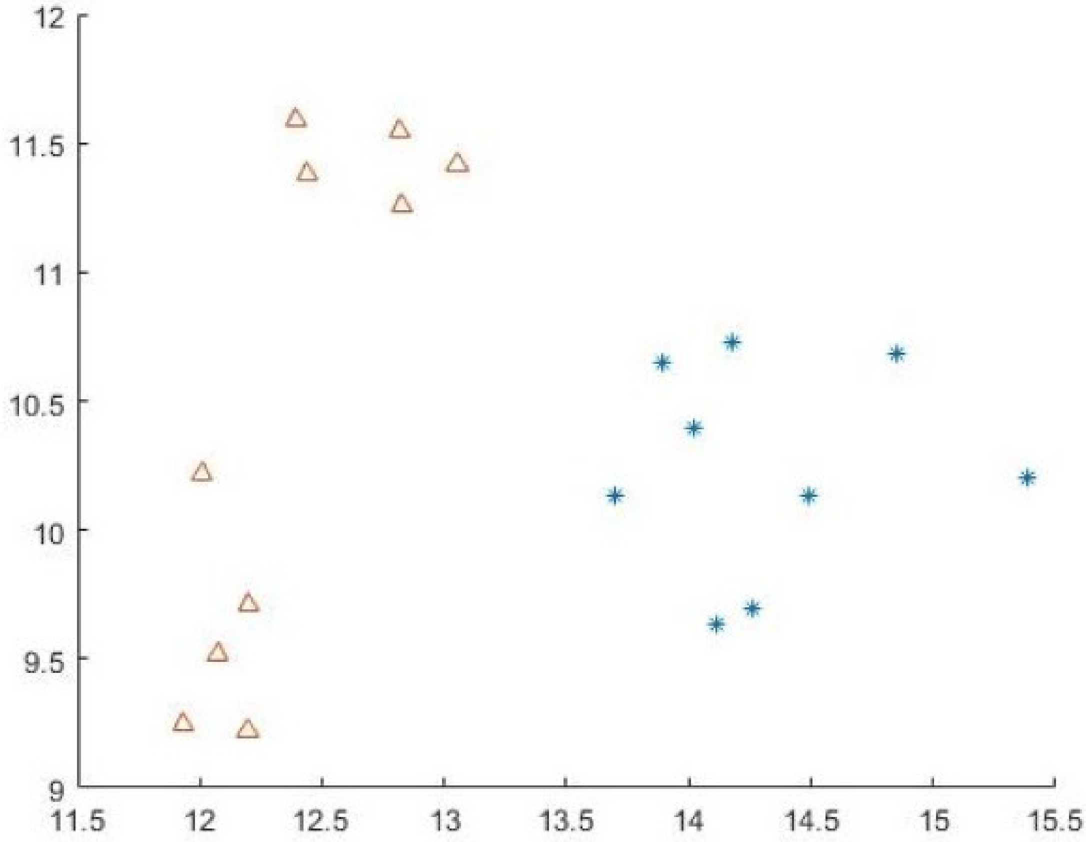
Then, a new vector $\mathbf{x}_*$ can be classified based on the minimal distance from $\boldsymbol{\mu}_k$

$$d(\mathbf{x}_*, \boldsymbol{\mu}_k) = \|\mathbf{x}_* - \boldsymbol{\mu}_k\|_2^2$$

This can be interpreted as a probalistic classifcation if we say that p(x|c_k) is defined in such a way that the Sigma is identity and all prior probabilities P(c_k) are the same. If P(c_k) is larger than any of the other classes then naturally it will have more influence which moves the decision boundary!

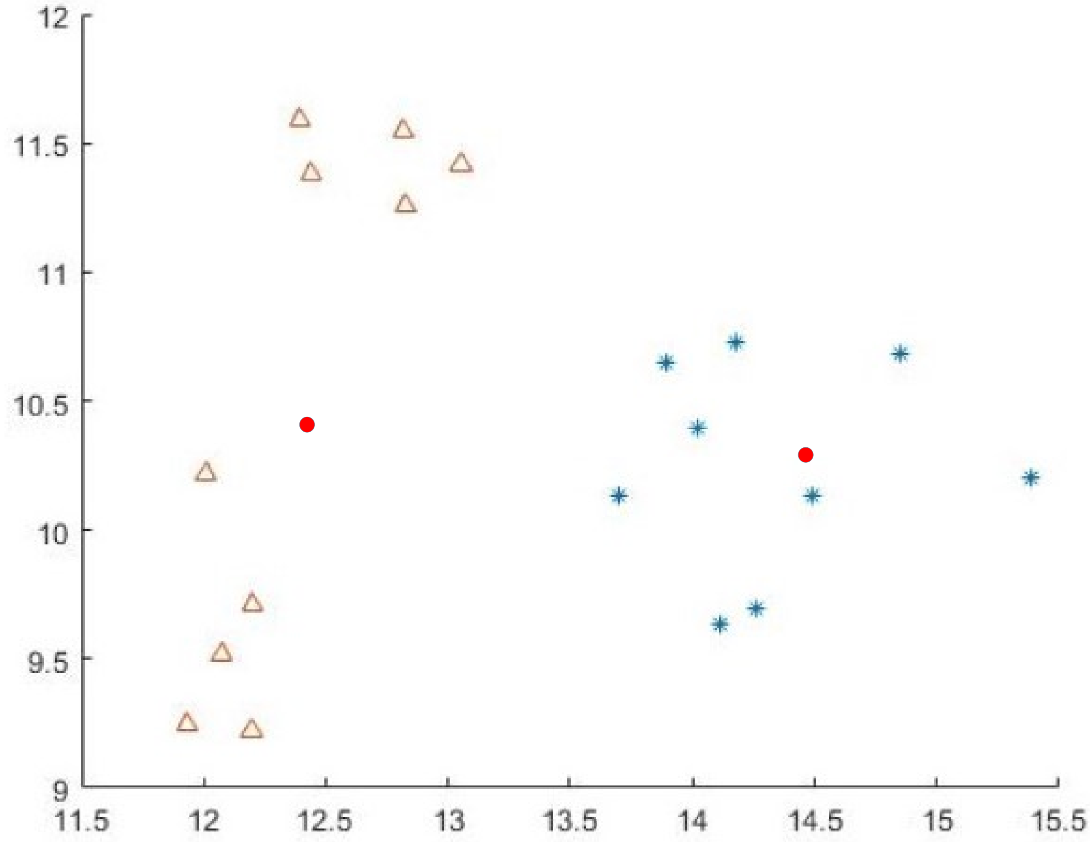
Nearest Prototype Classification

Example



Nearest Prototype Classification

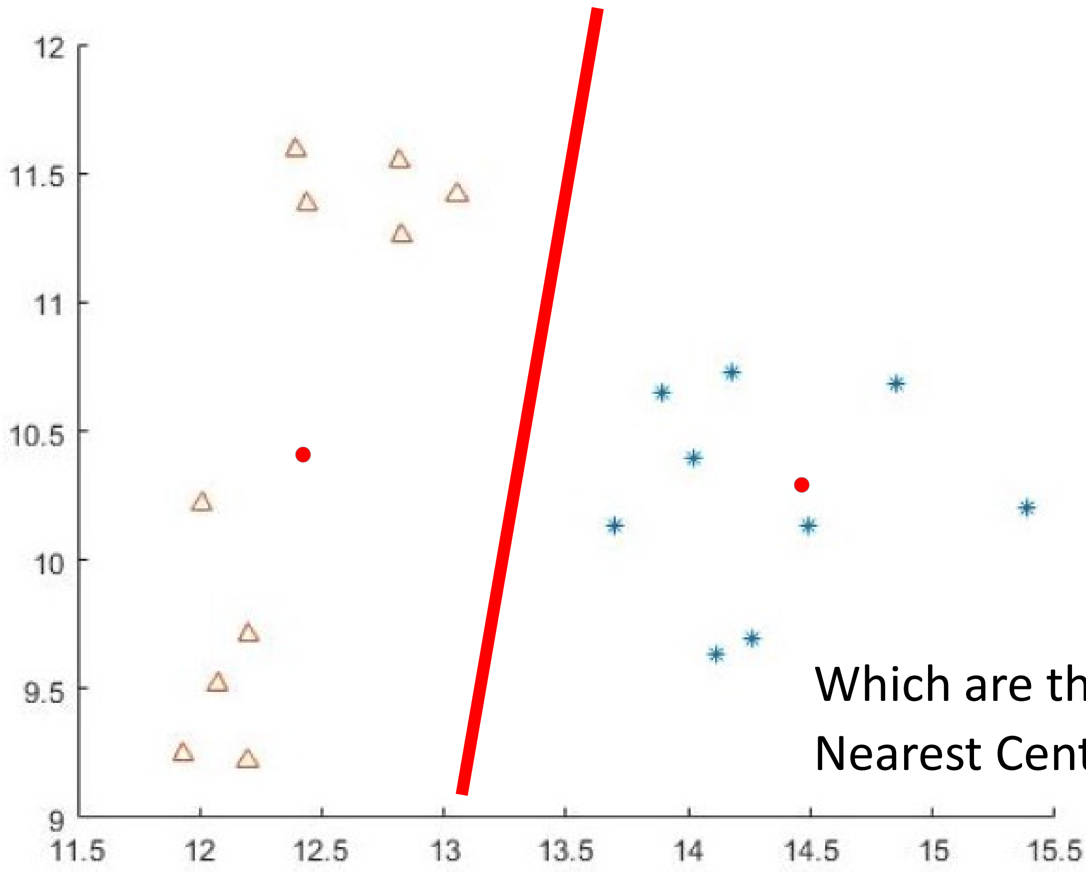
Example



Karstoft-Schultz-Iosifidis
Department of Engineering

Nearest Prototype Classification

Example



Which are the assumptions of Nearest Centroid classifier?

Nearest Prototype Classification

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels l_i, we can define the class mean vectors, we can define clusters on each class.

Nearest Prototype Classification

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels l_i, we can define the class mean vectors, we can define clusters on each class.

This means that we apply K times a clustering algorithm (e.g. K-Means). At each time, we use the samples of one class.

Nearest Prototype Classification

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels l_i, we can define the class mean vectors, we can define clusters on each class.

This means that we apply K times a clustering algorithm (e.g. K-Means). At each time, we use the samples of one class.

We obtain multiple prototypes for each class

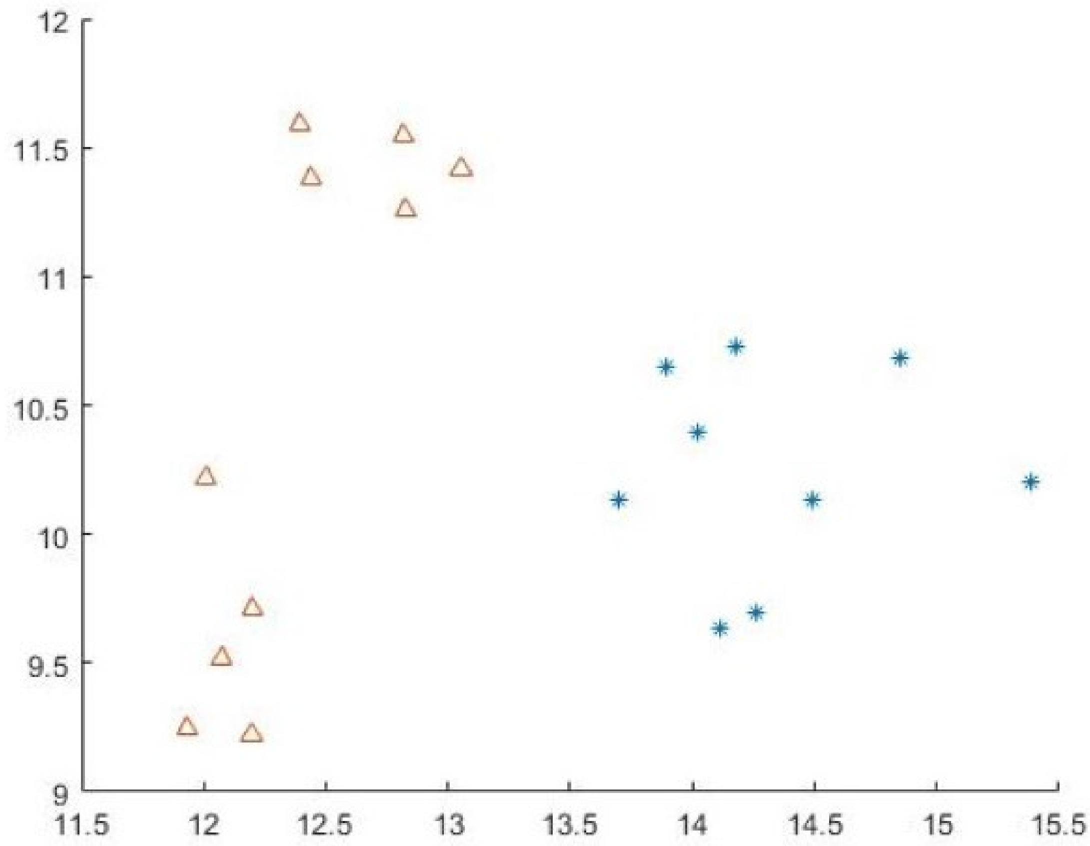
$$\boldsymbol{\mu}_{km} = \frac{1}{N_{km}} \sum_{i, l_i = k, q_i = m} \mathbf{x}_i$$

Then, a new vector $\mathbf{x}_*$ can be classified based on the minimal distance from $\boldsymbol{\mu}_{km}$

$$d(\mathbf{x}_*, \boldsymbol{\mu}_{km}) = \|\mathbf{x}_* - \boldsymbol{\mu}_{km}\|_2^2$$

Nearest Prototype Classification

Example

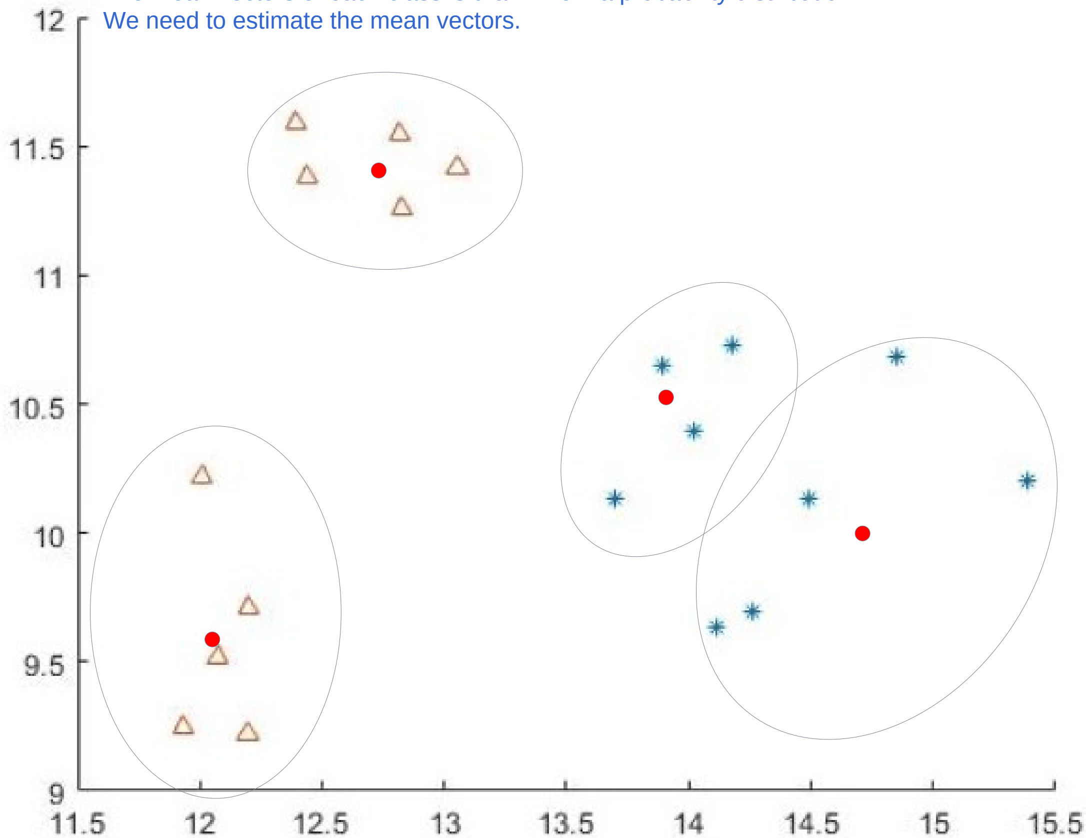


Nearest Prototype Classification

Interpreting this as probability-based classifier is complicated because we do not know the mean vectors.
The mean vectors are found by applying k-means. Applying k-means more than once, we get different resuts.
p(x|c1) = integral of N(mu,) * P(x)

The mean vectors of each class is drawn from a probability distribution.
We need to estimate the mean vectors.

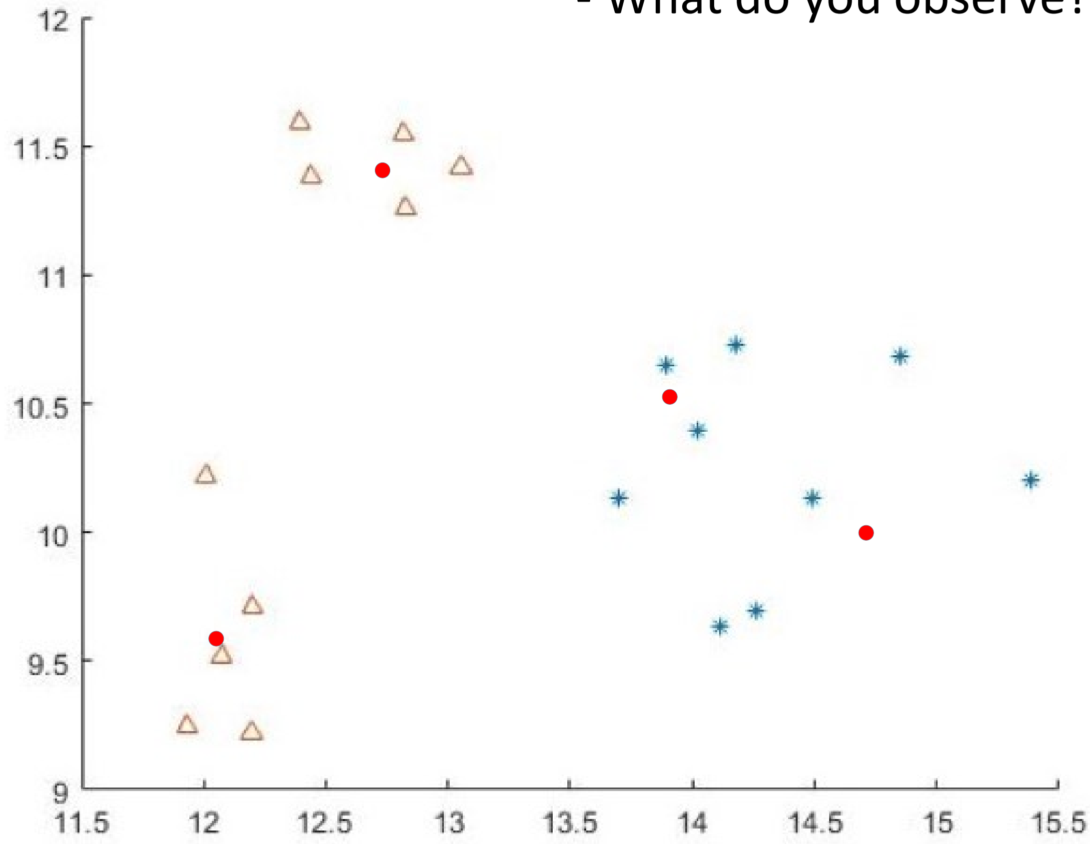
Example



Nearest Prototype Classification

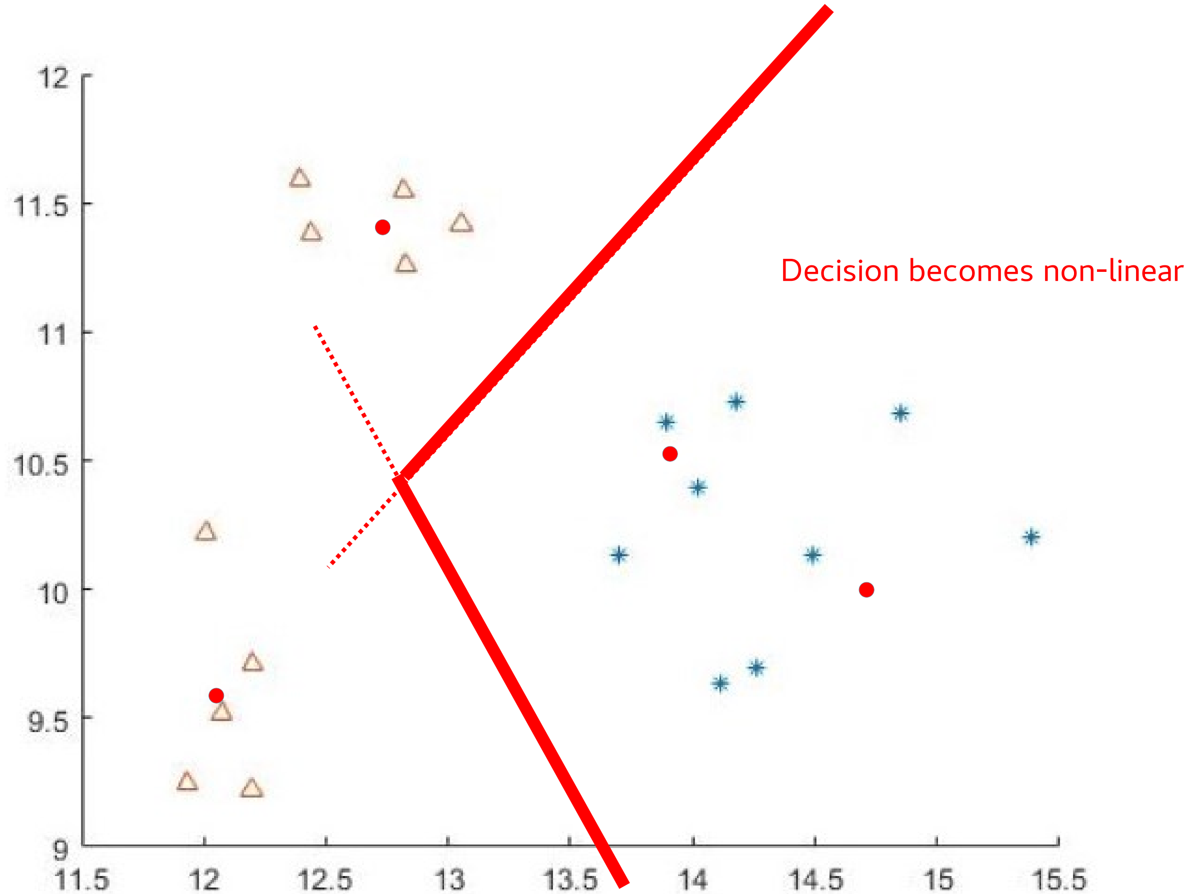
Example

- Which will be the decision hyperplane?
- What do you observe?



Nearest Prototype Classification

Example



Nearest Neighbor-based Classification

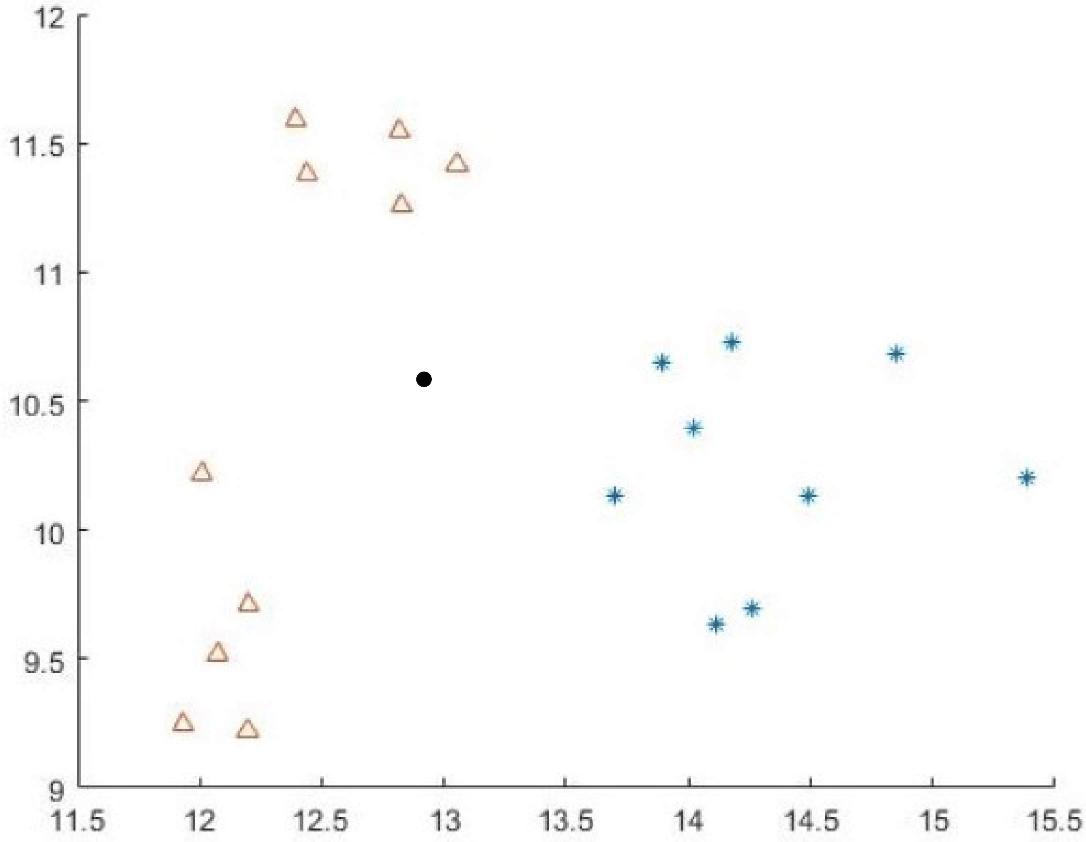
In the limit case where we assume that each training sample is a prototype, we end up calculating the distance of $\mathbf{x}_*$ with all training vectors $\mathbf{x}_i$, i=1,…,N and classify it to the class of the closest training sample.

How can we use multiple nearest neighbors for classification?

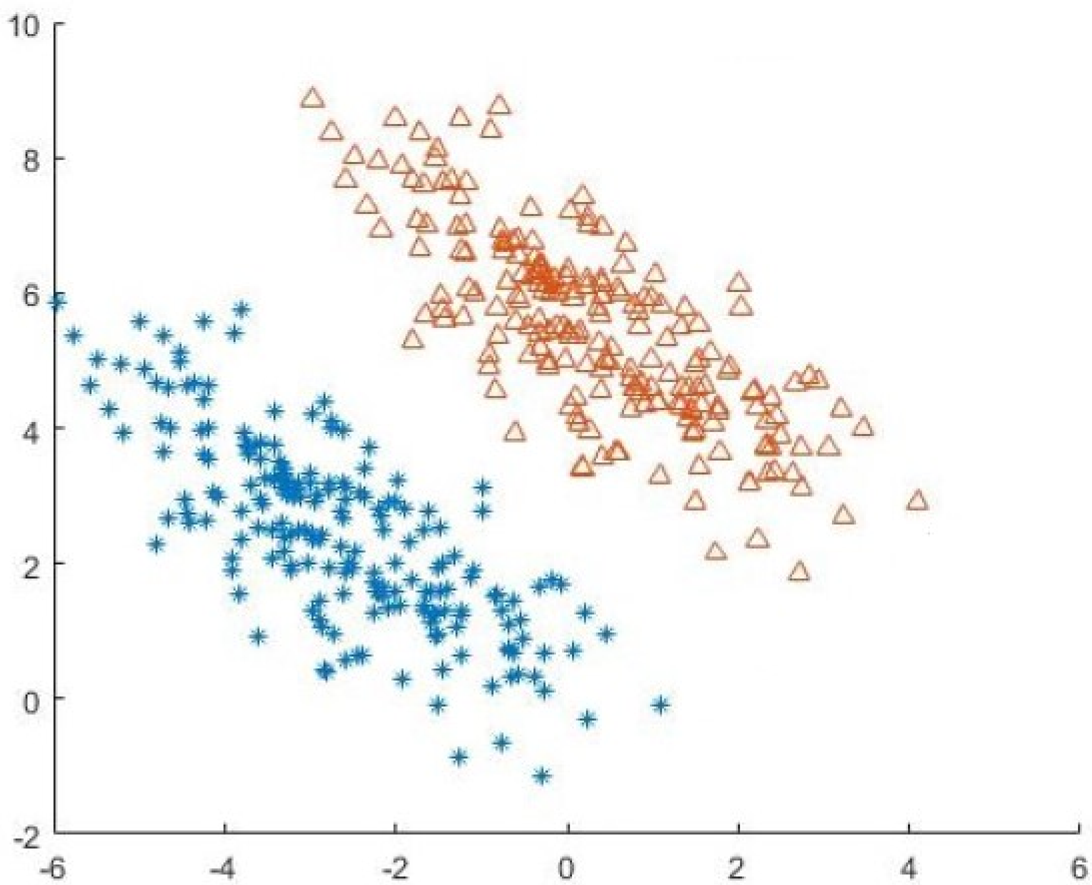
Demo: http://vision.stanford.edu/teaching/cs231n-demos/knn/

Nearest Neighbor-based Classification

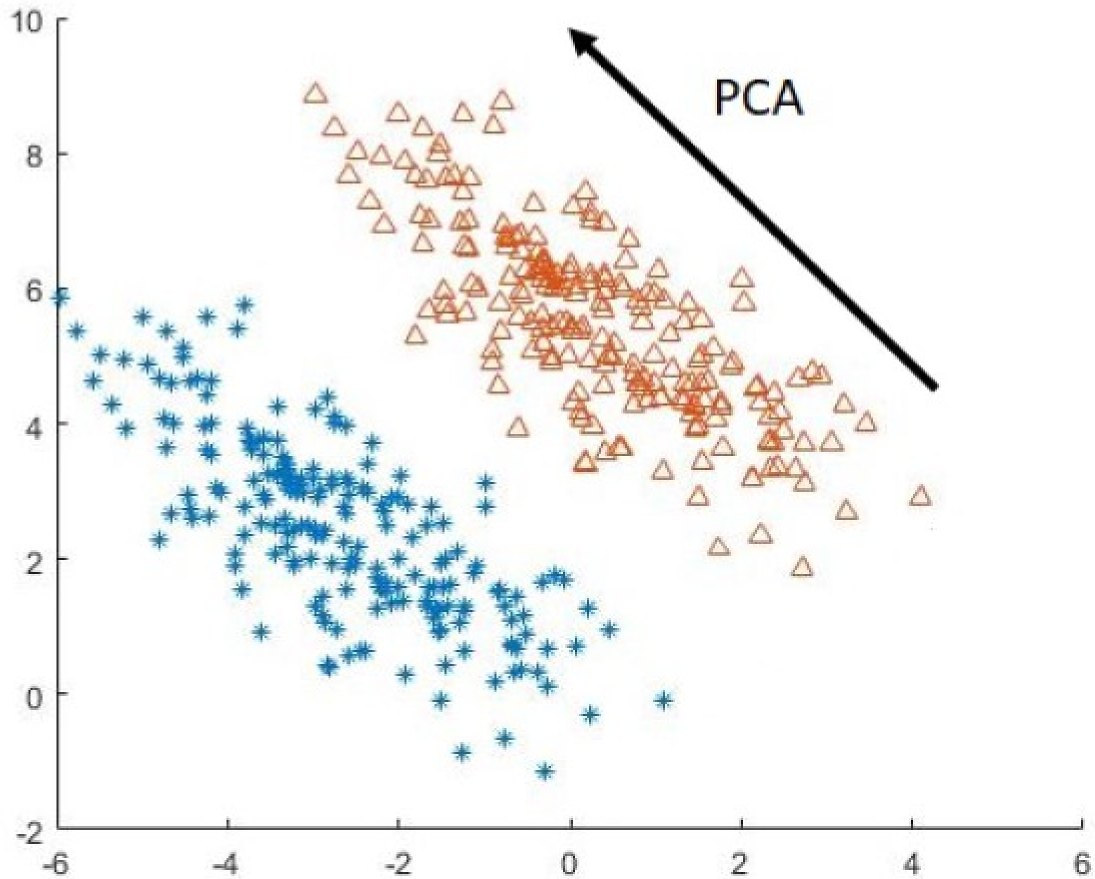
Example



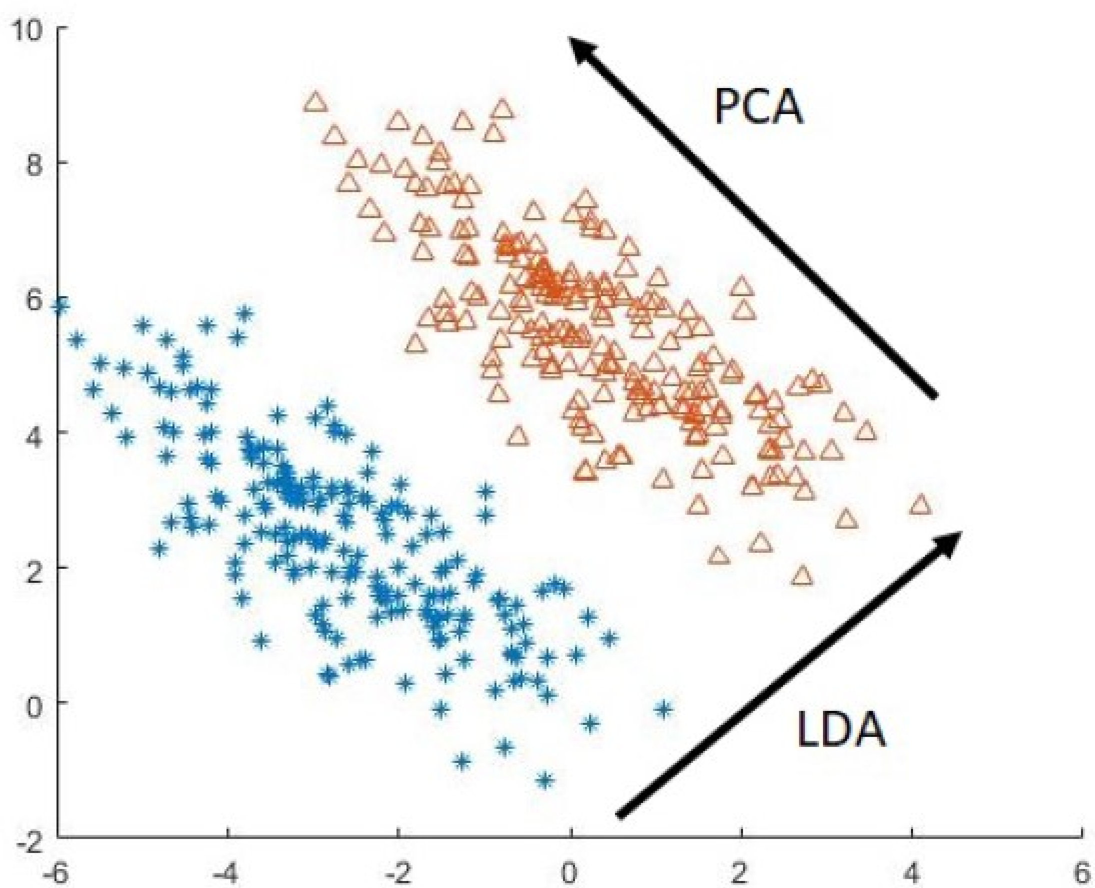
Fisher Discriminant Analysis



Fisher Discriminant Analysis



Fisher Discriminant Analysis



Fisher Discriminant Analysis

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels $l_i = \{1,2\}$ we can define a linear projection of the form

$$y_i = \mathbf{w}^T \mathbf{x}_i$$

where $\mathbf{w} \in \mathbb{R}^D$ is a (projection) vector mapping the D-dimensional space to a line.

Demo: https://calerga.com/projects/fm20170202/lda.html

Fisher Discriminant Analysis

Given a set of N samples, each represented by a vector $\mathbf{x}_i \in \mathbb{R}^D$, and the corresponding labels $l_i = \{1,2\}$ we can define a linear projection of the form

$$y_i = \mathbf{w}^T \mathbf{x}_i$$

where $\mathbf{w} \in \mathbb{R}^D$ is a (projection) vector mapping the D-dimensional space to a line.

Assuming that each class is unimodal and follows a Normal Distribution, how can we define the optimal vector **w**?

The mean vector

Fisher Discriminant Analysis

We define the class mean vectors $\boldsymbol{\mu}_k \in \mathbb{R}^D$, k=1,…,K

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{i, l_i = k} \mathbf{x}_i$$

Then, the mean values of each class in the projection space (line) are

$$m_k = \frac{1}{N_k} \sum_{i, l_i = k} y_i = \frac{1}{N_k} \sum_{i, l_i = k} \mathbf{w}^T \mathbf{x}_i = \mathbf{w}^T \boldsymbol{\mu}_k$$

Fisher Discriminant Analysis

We define the class mean vectors $\boldsymbol{\mu}_k \in \mathbb{R}^D$, k=1,…,K

$$\boldsymbol{\mu}_k = \frac{1}{N_k} \sum_{i, l_i = k} \mathbf{x}_i$$

Then, the mean values of each class in the projection space (line) are

$$m_k = \frac{1}{N_k} \sum_{i, l_i = k} y_i = \frac{1}{N_k} \sum_{i, l_i = k} \mathbf{w}^T \mathbf{x}_i = \mathbf{w}^T \boldsymbol{\mu}_k$$

The variance of each class in the line is
$$\sigma_k^2 = \frac{1}{N_k} \sum_{i, l_i = k} (y_i - m_k)^2$$

Fisher Discriminant Analysis

Since classes are unimodal and follow a Normal Distribution, they are better discriminated when:

1. The two mean values are as far as possible, i.e. their distance is as large as possible
2. The variances of the classes in the line are as small as possible

Fisher Discriminant Analysis

Since classes are unimodal and follow a Normal Distribution, they are better discriminated when:

1. The two mean values are as far as possible, i.e. their distance is as large as possible
2. The variances of the classes in the line are as small as possible

The distance of the centers can be expressed as a function of **w**

Similar to the expressing the objective function in PCA

$$
\begin{aligned}
(m_1 - m_2)^2 &= (\mathbf{w}^T \boldsymbol{\mu}_1 - \mathbf{w}^T \boldsymbol{\mu}_2)^2 \\
&= \mathbf{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \mathbf{w} \\
&= \mathbf{w}^T \mathbf{S}_b \mathbf{w}
\end{aligned}
$$

Fisher Discriminant Analysis

Since classes are unimodal and follow a Normal Distribution, they are better discriminated when:

1. The two mean values are as far as possible, i.e. their distance is as large as possible
2. The variances of the classes in the line are as small as possible

The distance of the centers can be expressed as a function of **w**

$$
\begin{aligned}
(m_1 - m_2)^2 &= (\mathbf{w}^T \boldsymbol{\mu}_1 - \mathbf{w}^T \boldsymbol{\mu}_2)^2 \\
&= \mathbf{w}^T (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^T \mathbf{w} \\
&= \mathbf{w}^T \mathbf{S}_b \mathbf{w}
\end{aligned}
$$

The variance can be written as

$$
\begin{aligned}
\sigma^2 &= \sigma_1^2 + \sigma_2^2 = \sum_{k=1}^{2} \sum_{i, l_i = k} (\mathbf{w}^T \mathbf{x}_i - \mathbf{w}^T \boldsymbol{\mu}_k)^2 \\
&= \sum_{k=1}^{2} \sum_{i, l_i = k} \mathbf{w}^T (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T \mathbf{w} = \mathbf{w}^T \mathbf{S}_w \mathbf{w}
\end{aligned}
$$

Fisher Discriminant Analysis

Since classes are unimodal and follow a Normal Distribution, they are better discriminated when:

1. The two mean values are as far as possible, i.e. their distance is as large as possible
2. The variances of the classes in the line are as small as possible

After expressing the two objectives above as functions of **w**, we can formulate an optimization problem which is a function of **w**

$$\mathcal{J}(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$

maximise the numerator
while
minimising the denominator

Fisher Discriminant Analysis

After expressing the two objectives above as functions of **w**, we can formulate an optimization problem which is a function of **w**

$$\mathcal{J}(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$

The above optimization problem is equivalent to the following problem

We can arrive at the next expressiong if we set the derivative of J(w) = 0. Once we take the derivative then the denominator cannot be zero. Look at the camera picture.

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

Fisher Discriminant Analysis

After expressing the two objectives above as functions of **w**, we can formulate an optimization problem which is a function of **w**

$$\mathcal{J}(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}$$

The above optimization problem is equivalent to the following problem

$$\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$$

Notice that w has not
the same scale here.
Therefore, w must be normalised

Assuming that $\mathbf{S}_w$ is non-singular $\quad \mathbf{S}_w^{-1}\mathbf{S}_b \mathbf{w} = \lambda \mathbf{w} \Rightarrow \mathbf{w} = \mathbf{S}_w^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is the extension of Fisher Discriminant Analysis for the case where $K > 2$.

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is the extension of Fisher Discriminant Analysis for the case where K > 2.

In LDA, $\mathbf{S}_w$ is a straightforward extension of the one used in FDA

$$\mathbf{S}_k = \sum_{i, l_i = k} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

$$\mathbf{S}_w = \sum_{k=1}^{K} \sum_{i, l_i = k} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

Linear Discriminant Analysis

In order to define the between-class scatter, we have

Total scatter matrix? Scatter matrix of all samples!

$$\begin{aligned}
\mathbf{S}_T &= \sum_{k=1}^{K} \sum_{i,l_i=k} (\mathbf{x}_i - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu})^T \\
&= \sum_{k=1}^{K} \sum_{i,l_i=k} (\mathbf{x}_i - \boldsymbol{\mu}_k + \boldsymbol{\mu}_k - \boldsymbol{\mu})(\mathbf{x}_i - \boldsymbol{\mu}_k + \boldsymbol{\mu}_k - \boldsymbol{\mu})^T \\
&= \sum_{k=1}^{K} \sum_{i,l_i=k} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T + \sum_{k=1}^{K} \sum_{i,l_i=k} (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T \\
&= \mathbf{S}_w + \sum_{k=1}^{K} N_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T \\
&= \mathbf{S}_w + \mathbf{S}_b.
\end{aligned}$$

N_k is because we have double sum here
Essentially,

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is the extension of Fisher Discriminant Analysis for the case where K > 2.

Thus, the within-class and between-class scatter matrices are defined as

$$\mathbf{S}_k = \sum_{i, l_i = k} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

$$\mathbf{S}_w = \sum_{k=1}^{\text{K}} \sum_{i, l_i = k} (\mathbf{x}_i - \boldsymbol{\mu}_k)(\mathbf{x}_i - \boldsymbol{\mu}_k)^T$$

$$\mathbf{S}_b = \sum_{k=1}^{K} N_k (\boldsymbol{\mu}_k - \boldsymbol{\mu})(\boldsymbol{\mu}_k - \boldsymbol{\mu})^T$$

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is the extension of Fisher Discriminant Analysis for the case where K > 2.

The optimization problem of LDA is

$$\mathcal{J}(\mathbf{W}) = \frac{Tr(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{Tr(\mathbf{W}^T \mathbf{S}_w \mathbf{W})}$$

What are the dimensions of **W**?

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is the extension of Fisher Discriminant Analysis for the case where K > 2.

The optimization problem of LDA is

$$\mathcal{J}(\mathbf{W}) = \frac{Tr(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{Tr(\mathbf{W}^T \mathbf{S}_w \mathbf{W})}$$

W is obtained by solving for: $\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$

Linear Discriminant Analysis

Linear Discriminant Analysis (LDA) is the extension of Fisher Discriminant Analysis for the case where K > 2.

The optimization problem of LDA is

$$\mathcal{J}(\mathbf{W}) = \frac{Tr(\mathbf{W}^T \mathbf{S}_b \mathbf{W})}{Tr(\mathbf{W}^T \mathbf{S}_w \mathbf{W})}$$

W is obtained by solving for: $\mathbf{S}_b \mathbf{w} = \lambda \mathbf{S}_w \mathbf{w}$

Solving this eigenproblem we get K-1 eigenvectors because the rank of S_b is K-1

We usually add a constraint $\mathbf{W}^T\mathbf{W} = \mathbf{I}$, why?

Because we need the vectors in W to be orthonormal. The practical problem that we are solving is we don't want to have irrelavent information.