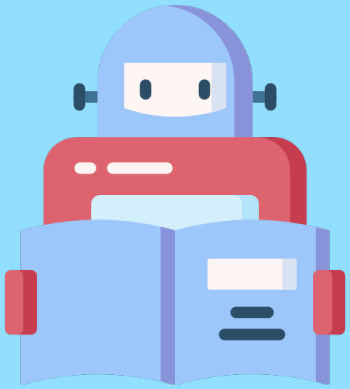
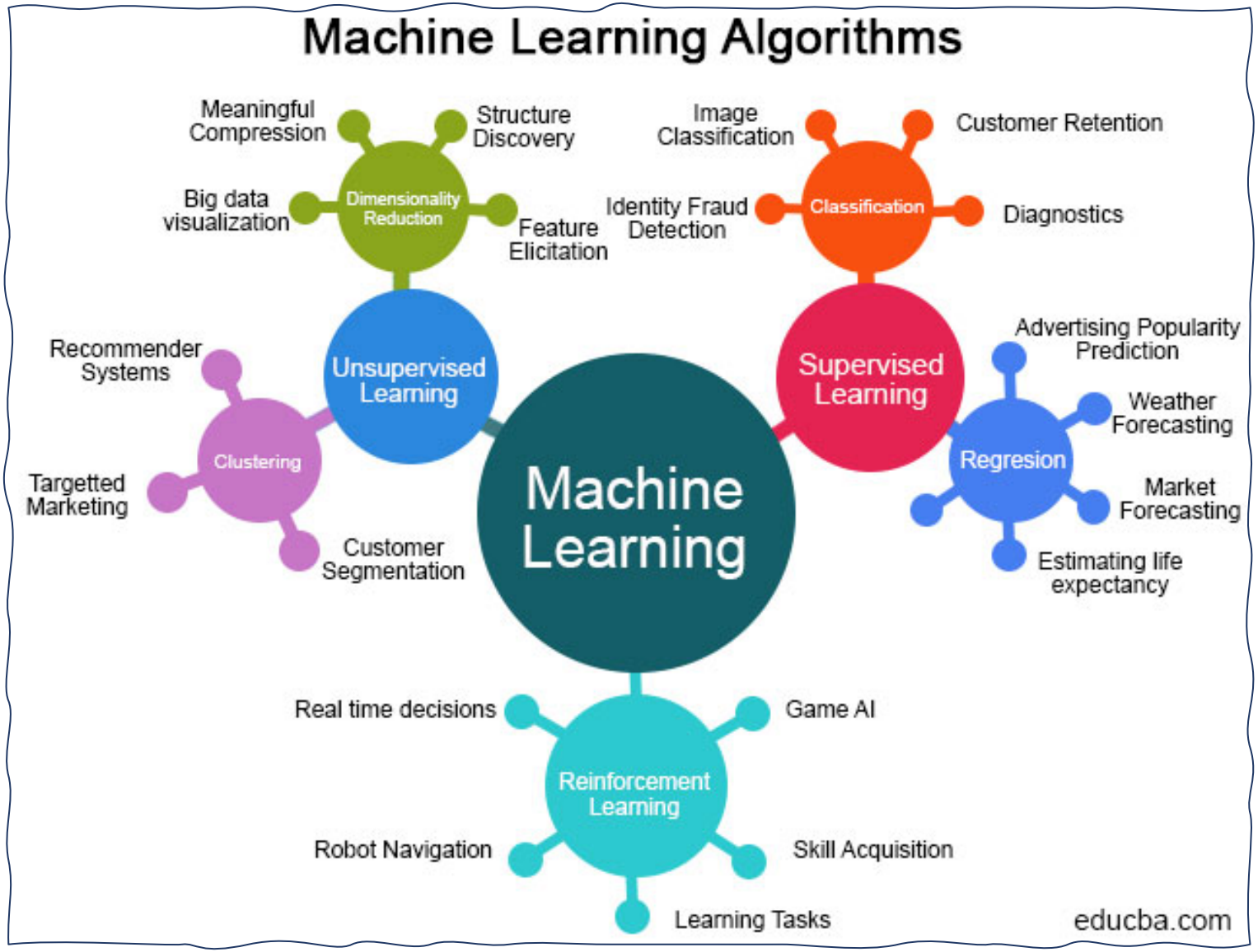
Mathematics for Machine Learning



Special Mathematics Lecture
Nagoya University, Fall 2020

Lecture 6
Naive Bayes II & Support vector machines I

https://www.henrikbachmann.com/mml_2020.html


Machine Learning Algorithms
Machine Learning
Unsupervised Learning
Supervised Learning
Reinforcement Learning
Dimensionality Reduction
Classification
Clustering
Regresion
Meaningful Compression
Structure Discovery
Big data visualization
Feature Elicitation
Image Classification
Customer Retention
Identity Fraud Detection
Diagnostics
Recommender Systems
Targetted Marketing
Customer Segmentation
Advertising Popularity Prediction
Weather Forecasting
Market Forecasting
Estimating life expectancy
Real time decisions
Game AI
Robot Navigation
Skill Acquisition
Learning Tasks
educba.com

Generative vs Discriminative learning algorithm

Notation: P(A|B) refers to the conditional probability that event A occurs, given that event B has occurred.

x: feature (e.g. hours of studying)
y: label (e.g. passing or failing exam)

Logistic regression

Want to find a hypothesis which describes P(y|x)

$$P(y = 1 \mid x; \theta) = h_\theta(x) \, .$$

Learning P(y|x) is an example of a **discriminative learning algorithm.**

Generative learning algorithm: Learn P(x | y) and P(y).

Generative learning algorithm

Notation: P(A|B) refers to the conditional probability that event A occurs, given that event B has occurred.

Generative learning algorithm: Learn P(x | y) and P(y).

Use Bayes rule & Law of total probability:

Bayes rule: $$P(y = 1 \mid x) = \frac{P(x \mid y = 1)P(y = 1)}{P(x)}$$

Law of total probability: $$P(x) = P(x \mid y = 1)P(y) + P(x \mid y = 0)P(y = 0)$$

Naive Bayes

An example for a generative learning algorithm: Naive Bayes

Example: Spam filter

- Feature: Email
- Label: Spam & No Spam

$$\mathcal{X} = \{0,1\}^d \qquad d : \text{words in a dictionary}$$

$$\mathcal{Y} = \{0,1\}$$

Naive Bayes assumption

Naive Bayes assumption:

The features are “conditionally independent” given the label.

conditionally independent

A and B are conditionally independent given C if and only if, given knowledge that C occurs, knowledge of whether A occurs provides no information on the likelihood of B occurring, and knowledge of whether B occurs provides no information on the likelihood of A occurring.

Naive Bayes assumption

Naive Bayes assumption:

The features are "conditionally independent" given the label.

If x_1, x_2 are conditionally independent given y, then we have

$$P(x_1 \mid y, x_2) = P(x_1|y)\,.$$

We want to calculate $P(x|y) = P(x_1, \ldots, x_d|y)$.

Chain rule of probabilities: $P(A, B) = P(A|B)P(B)$

Naive Bayes assumption

Naive Bayes assumption:

$$P(x_1 \mid y, x_2) = P(x_1|y)\,.$$

Chain rule of probabilities:

$$P(A, B) = P(A|B)P(B)$$

We want to calculate $P(x|y) = P(x_1, \ldots, x_d|y)$.

By the naive Bayes assumption we obtain

$$P(x \mid y) = P(x_1, \ldots, x_d|y) = \prod_{i=1}^{d} P(x_j \mid y)$$

Our model is parametrized (the stuff we need to remember after training) by

$$\begin{aligned}
\phi_{i|y=1} &= P(x_i = 1 \mid y = 1)\,, \\
\phi_{i|y=0} &= P(x_i = 1 \mid y = 0)\,, \\
\phi_{y=1} &= P(y = 1)\,.
\end{aligned}$$

Naive Bayes classifier: Training

$$\phi_{i|y=1} = P(x_i = 1 \mid y = 1)\,,$$
$$\phi_{i|y=0} = P(x_i = 1 \mid y = 0)\,,$$
$$\phi_{y=1} = P(y = 1)\,.$$

Indicator function

$$I(S) = \begin{cases} 1\,, & S \text{ is true} \\ 0\,, & S \text{ is false} \end{cases}.$$

Given a training set $\mathcal{T} = \big((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\big)$ we can calculate them by

$$\phi_{i|y=1} = \frac{\sum_{j=1}^{n} I(x_i^{(j)} = 1 \wedge y^{(j)} = 1)}{\sum_{j=1}^{n} I(y^{(j)} = 1)}$$

$$\phi_{i|y=0} = \frac{\sum_{j=1}^{n} I(x_i^{(j)} = 1 \wedge y^{(j)} = 0)}{\sum_{j=1}^{n} I(y^{(j)} = 0)}$$

$$\phi_{y=1} = \frac{1}{n} \sum_{j=1}^{n} I(y^{(j)} = 1)$$

What to do with a new mail?

$$P(x \mid y) = P(x_1, \ldots, x_d|y) = \prod_{i=1}^{d} P(x_j \mid y)$$

After training we have the following numbers:

$$\begin{aligned} \phi_{i|y=1} &= P(x_i = 1 \mid y = 1)\,, \\ \phi_{i|y=0} &= P(x_i = 1 \mid y = 0)\,, \\ \phi_{y=1} &= P(y = 1)\,. \end{aligned}$$

Now assume we get a new feature (i.e. someone sends us an email) $x \in \mathcal{X}$. Then we can calculate for each word $i = 1, \ldots, d$ the probabilities

$$\begin{aligned} P(x_i = 1 \mid y = 1) &= \phi_{i|y=1}\,, & P(x_i = 1 \mid y = 0) &= \phi_{i|y=0}\,, \\ P(x_i = 0 \mid y = 1) &= 1 - \phi_{i|y=1}\,, & P(x_i = 0 \mid y = 0) &= 1 - \phi_{i|y=0}\,. \end{aligned}$$

The probability of the new email being spam is then

$$\begin{aligned} P(y = 1 \mid x) &= \frac{P(x \mid y = 1)P(y = 1)}{P(x)} \\ &= \frac{\prod_{i=1}^{d} P(x_i \mid y = 1) \cdot \phi_{y=1}}{\prod_{i=1}^{d} P(x_i \mid y = 1) \cdot \phi_{y=1} + \prod_{i=1}^{d} P(x_i \mid y = 0)(1 - \phi_{y=1})}\,. \end{aligned}$$

What to do with a new mail?

$$P(x \mid y) = P(x_1, \ldots, x_d|y) = \prod_{i=1}^{d} P(x_j \mid y)$$

After training we have the following numbers:

$$\begin{aligned} \phi_{i|y=1} &= P(x_i = 1 \mid y = 1)\,, \\ \phi_{i|y=0} &= P(x_i = 1 \mid y = 0)\,, \\ \phi_{y=1} &= P(y = 1)\,. \end{aligned}$$

Alternative: Compare the y = 1 and y = 0 case.

Now assume we get a new feature (i.e. someone sends us an email) $x \in \mathcal{X}$. Then we can calculate for each word $i = 1, \ldots, d$ the probabilities

$$\begin{aligned} P(x_i = 1 \mid y = 1) &= \phi_{i|y=1}\,, & P(x_i = 1 \mid y = 0) &= \phi_{i|y=0}\,, \\ P(x_i = 0 \mid y = 1) &= 1 - \phi_{i|y=1}\,, & P(x_i = 0 \mid y = 0) &= 1 - \phi_{i|y=0}\,. \end{aligned}$$

The probability of the new email being spam is then

$$\begin{aligned} P(y = 1 \mid x) &= \frac{P(x \mid y = 1)P(y = 1)}{P(x)} \\ &= \frac{\prod_{i=1}^{d} P(x_i \mid y = 1) \cdot \phi_{y=1}}{\prod_{i=1}^{d} P(x_i \mid y = 1) \cdot \phi_{y=1} + \prod_{i=1}^{d} P(x_i \mid y = 0)(1 - \phi_{y=1})}\,. \end{aligned}$$

Naive Bayes classifier: Problem!

There is a simple to solve problem with the Naïve Bayes classifier described so far.

Assume you get an email containing a word from the dictionary which never appear in the Trainings set.

$$P(y = 1 \mid x) = \frac{\prod_{i=1}^{d} P(x_i \mid y = 1) \cdot \phi_{y=1}}{\prod_{i=1}^{d} P(x_i \mid y = 1) \cdot \phi_{y=1} + \prod_{i=1}^{d} P(x_i \mid y = 0)(1 - \phi_{y=1})}.$$

Naive Bayes classifier: Laplace/additive smoothing

To solve this problem, we will assume at least a small probability for any event.

$$\widetilde{\phi}_{i|y=1} = \frac{1 + \sum_{j=1}^{n} I(x_i^{(j)} = 1 \wedge y^{(j)} = 1)}{2 + \sum_{j=1}^{n} I(y^{(j)} = 1)}$$

$$\widetilde{\phi}_{i|y=0} = \frac{1 + \sum_{j=1}^{n} I(x_i^{(j)} = 1 \wedge y^{(j)} = 0)}{2 + \sum_{j=1}^{n} I(y^{(j)} = 0)}$$

$$\widetilde{\phi}_{y=1} = \frac{1 + \sum_{j=1}^{n} I(y^{(j)} = 1)}{2 + n}$$

Possible interpretation: We assume that each word appeared at least once in a spam and in a non spam email.

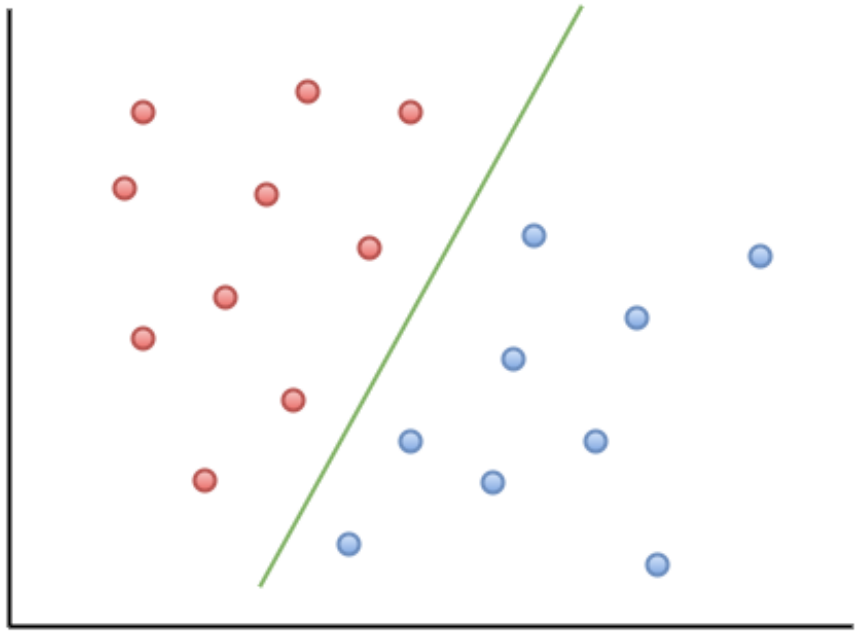
Homework 2 (coming in the next days)

- In Homework 2 you should implement the Naïve Bayes algorithm to make a simple spam filter as described above.
- The template is still under construction.
- Will be posted on the homepage and announced in the discord.

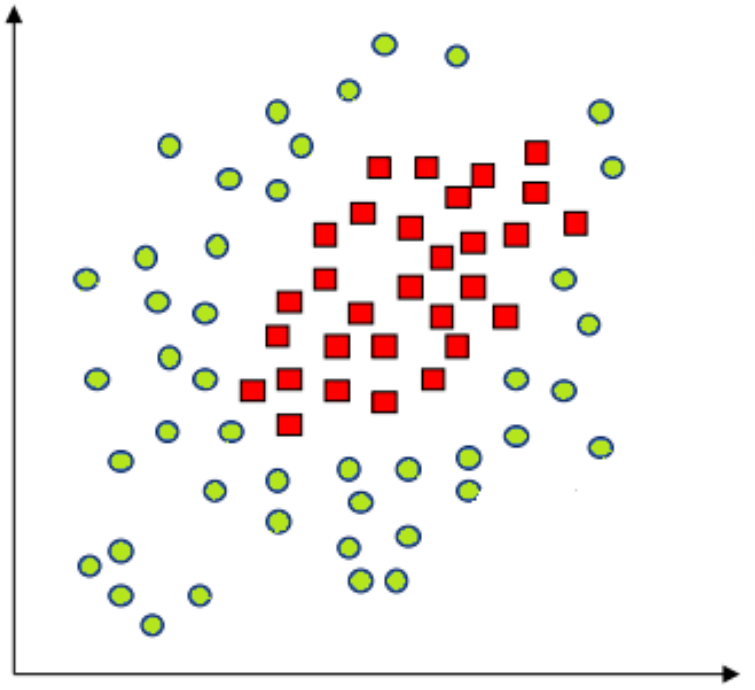
Support vector machines

Features: $\mathcal{X} = \mathbb{R}^d$

Labels: $\mathcal{Y} = \{-1, 1\}$

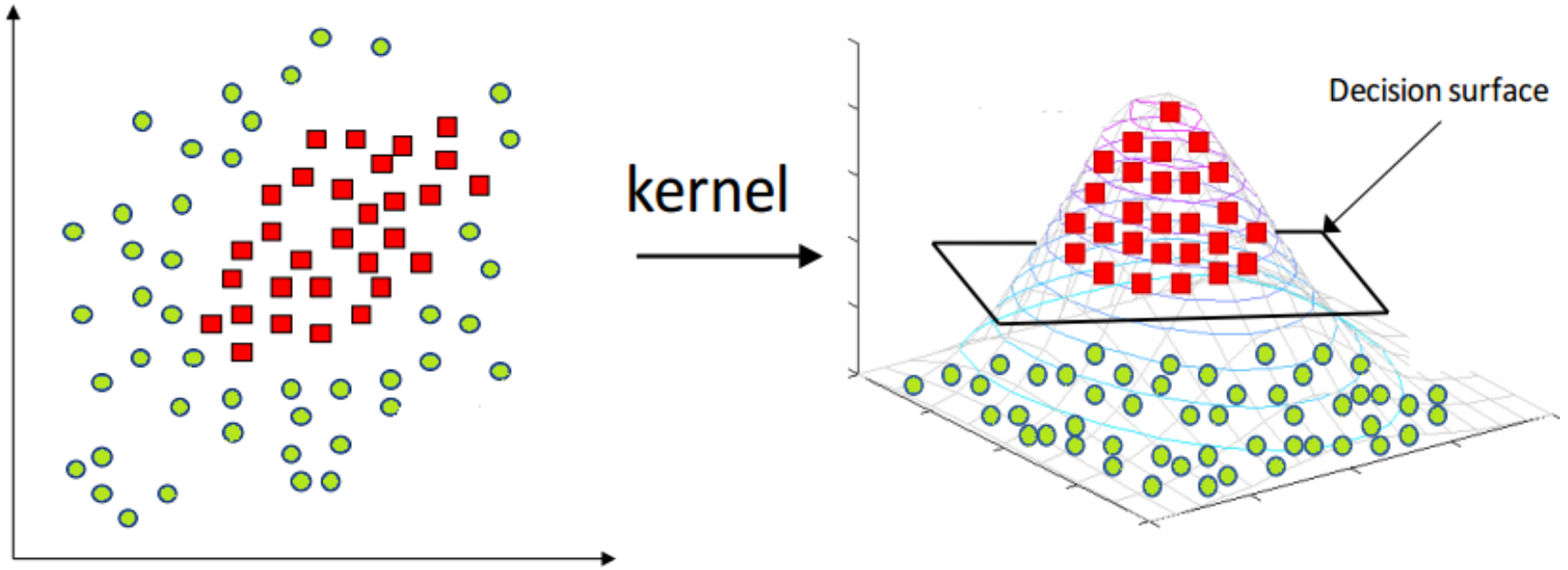


Support vector machines



We can not separate these by a line!

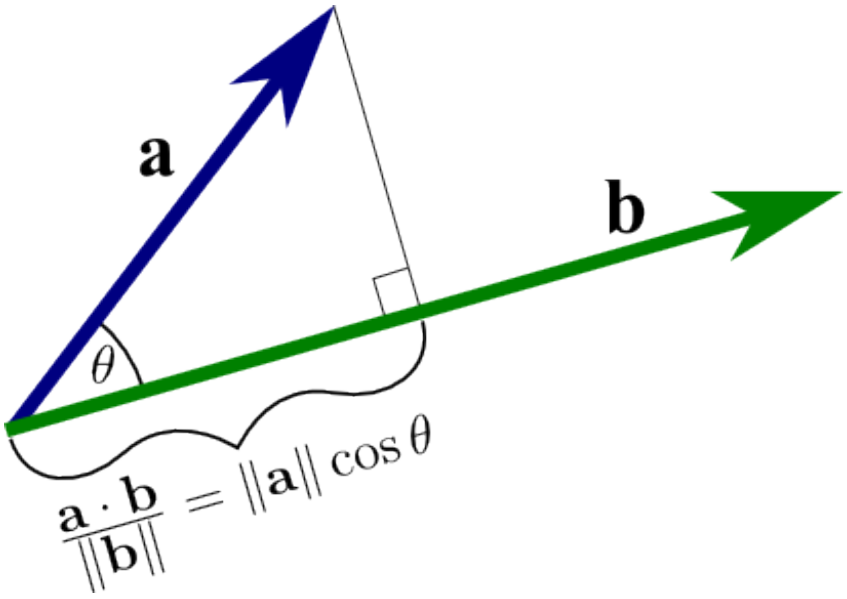
Support vector machines: Kernel trick



In 3-dimensions we can!

First some basics: Dot product & Hesse normal form

Geometric interpretation of the dot product



First some basics: Dot product & Hesse normal form

How to describe a hyperplane: Hesse normal form

Separate a training set by a hyperplane

For $w \in \mathbb{R}^d$, $b \in \mathbb{R}$ we define the hyperplane $H(w, b)$ by

$$H(w, b) = \left\{ x \in \mathbb{R}^d \mid w^T x - b = 0 \right\} .$$

$\mathcal{X} = \mathbb{R}^d$, $\mathcal{Y} = \{-1, 1\}$

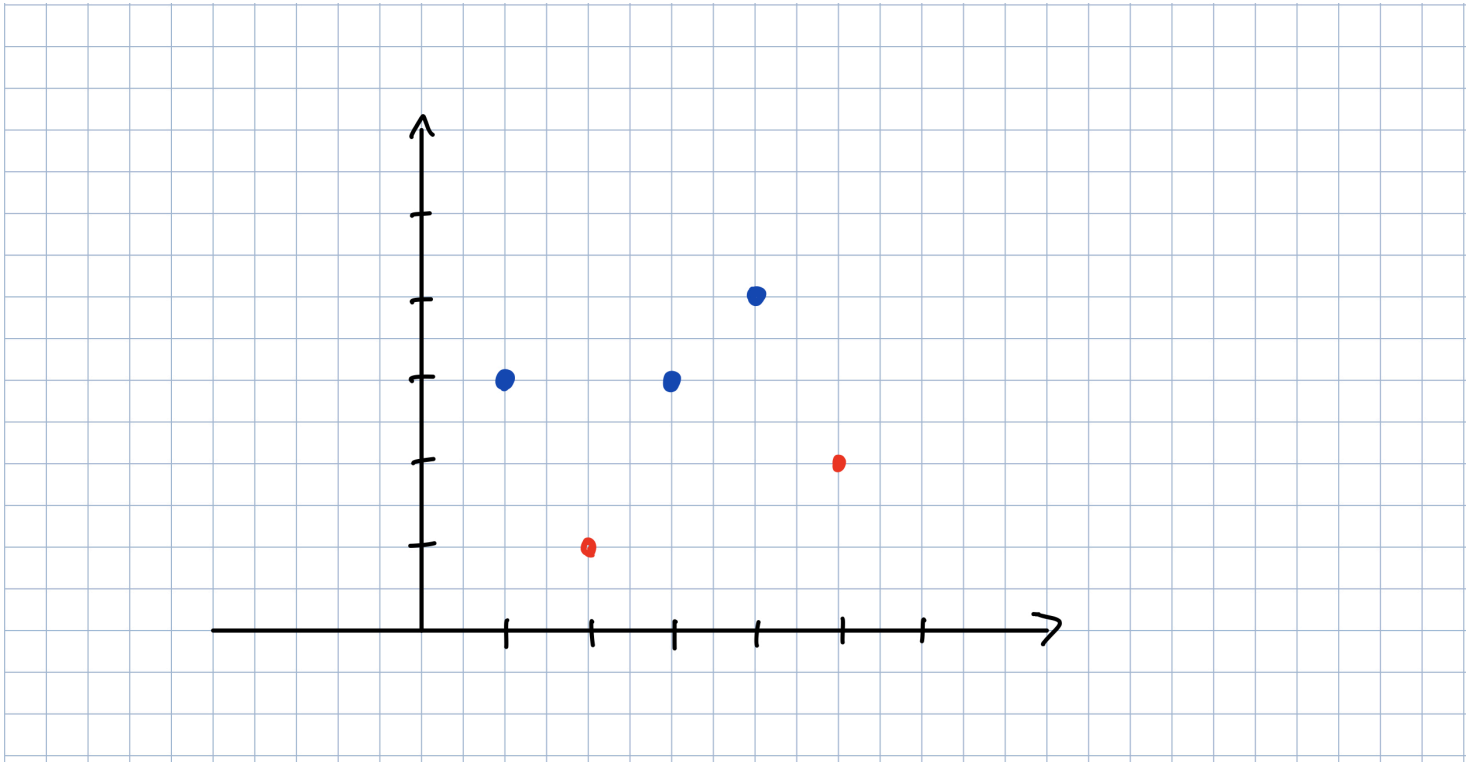
Let $\mathcal{T} = \left((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)}) \right)$ be a training set.

Goal: Find a hyperplane which separates the training examples in the "best" possible way.

Separate a training set by a hyperplane

Let $d = 2$ and consider the training set

$$\mathcal{T} = \left(\left(\begin{pmatrix} 1 \\ 3 \end{pmatrix}, 1 \right), \left(\begin{pmatrix} 3 \\ 3 \end{pmatrix}, 1 \right), \left(\begin{pmatrix} 4 \\ 4 \end{pmatrix}, 1 \right), \left(\begin{pmatrix} 2 \\ 1 \end{pmatrix}, -1 \right), \left(\begin{pmatrix} 5 \\ 2 \end{pmatrix}, -1 \right) \right)$$



What would be a good hyperplane to separate these points?
How do we measure if it is good?

Functional margin of a hyperplane

We define the **functional margin of the hyperplane** $H(w, b)$ with respect to the training example $(x^{(j)}, y^{(j)})$ by

$$\hat{\gamma}^{(j)} = y^{(i)}(w^T x^{(j)} + b)\,.$$

The functional margin of the hyperplane $H(w, b)$ with respect to the training set $\mathcal{T}$ is then defined by

$$\hat{\gamma} = \min_{j=1,\ldots,n} \gamma^{(j)}\,.$$

Big functional margin = good

… but..

Support vector machines: Optimal margin

We will normalize the functional margin to 1 and will try to maximize the **geometric margin**.

Let $\mathcal{T} = \left((x^{(1)}, y^{(1)}), \ldots, (x^{(n)}, y^{(n)})\right)$ be a training set.

Goal: Find a Hyperplane $H(w, b)$, such that

i) $\|w\|$ is minimal.

ii) For $j = 1, \ldots, n$ we have $\hat{\gamma}^{(j)} = y^{(i)}(w^T x^{(j)} + b) \geq 1$.