数据挖掘导论

Introduction to Data Mining

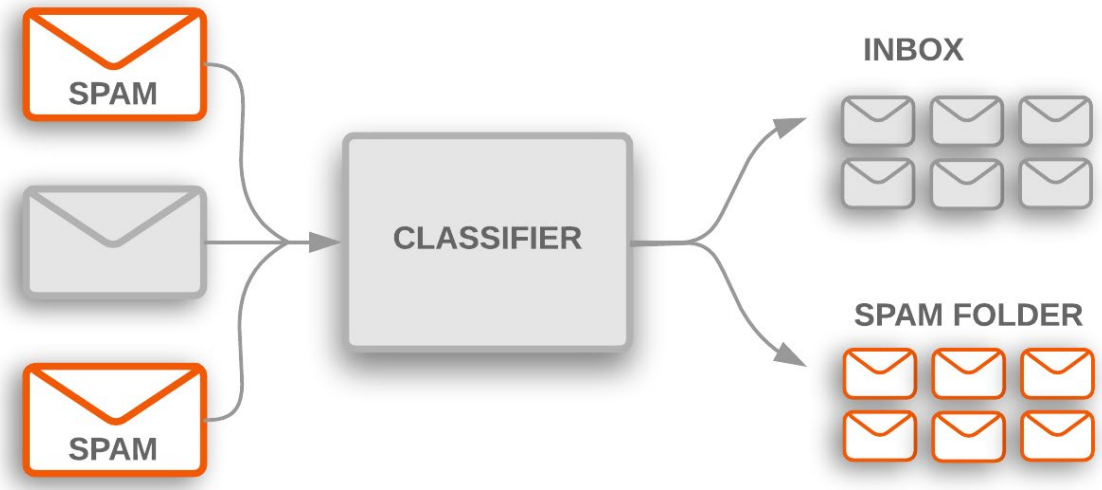
Basic Classification

数据智能实验室
DATA INTELLIGENCE LABORATORY
DI-LAB
浙江大学
Zhejiang University

Agenda

- Classification: Basic Concepts
- kNN Classifier
- Decision Tree
- Bayes Classification
- Support Vector Machine

Classification on Text Data



系统通知

admin： 用户 zhangdongxiang@zju.edu.cn 维护原因 为进一步提升邮件系统的安全性，我部门于近日陆续更新

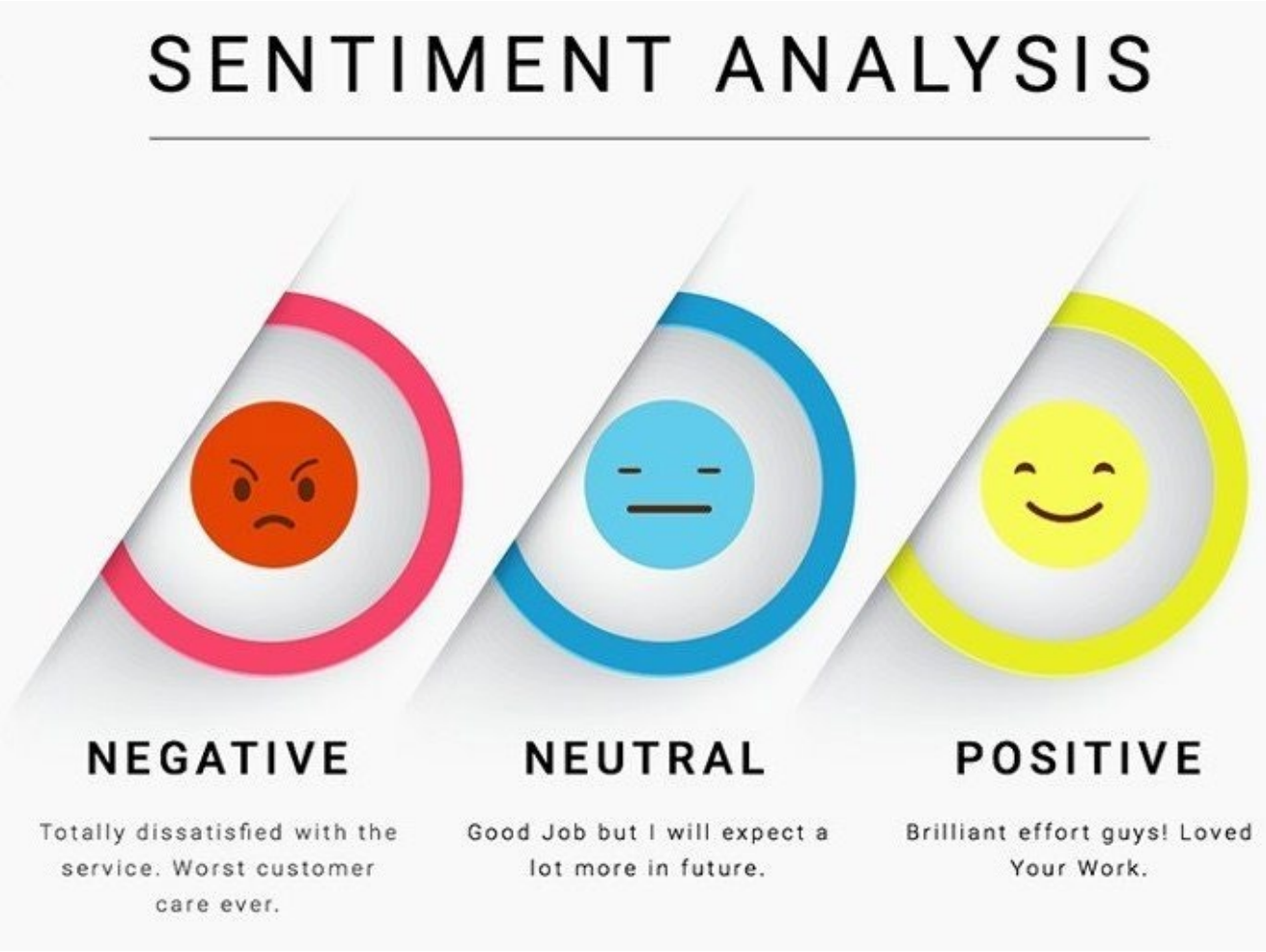
本次将继承往届会议的优点，继续为领域内的专家学者提供高水准的国际交流平台

EI-- ICNISC-9th： 计算机网络-信息系统会/议/时/间：2023年10月27日-29日会/议/地/点：中国武汉会/议/语/言

早 (12)

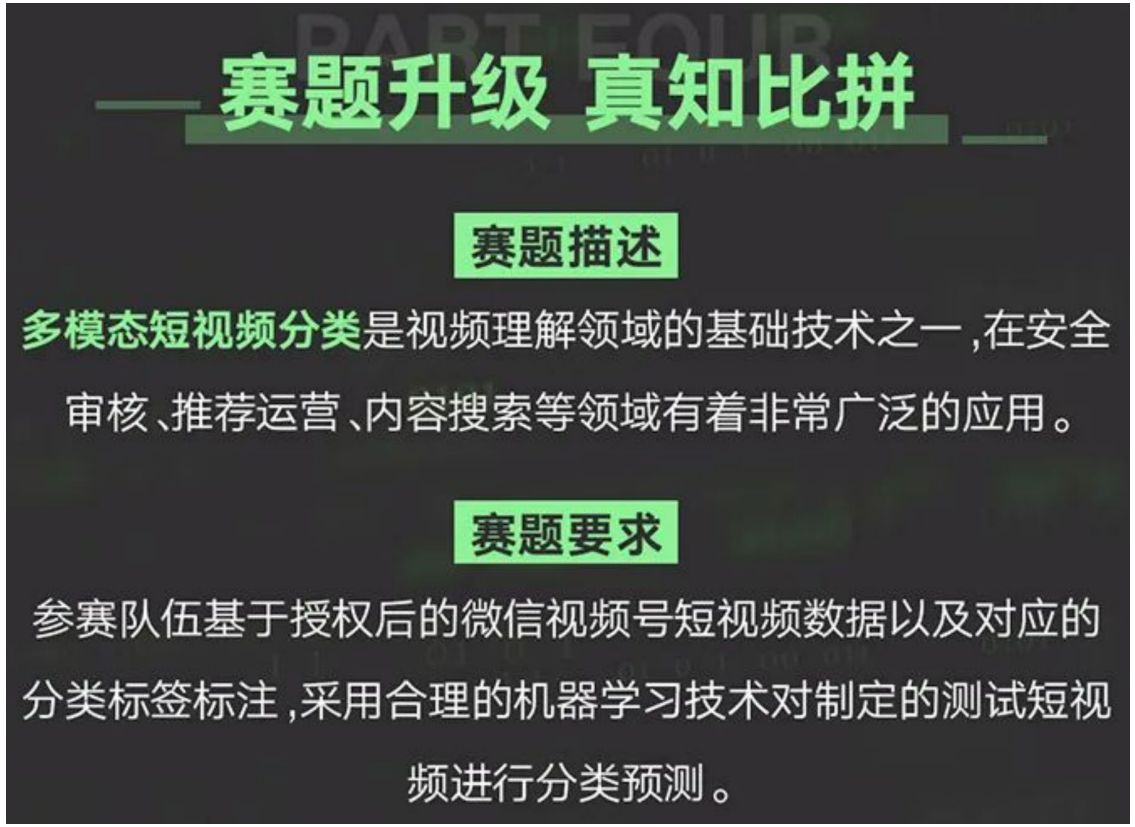
精美配图助力冲击高水平期刊

松迪科研绘图/科学可视化： 如果邮件内容无法正常显示请点击这里 退订 投诉 培训安排：5月10日-5月12日 合肥

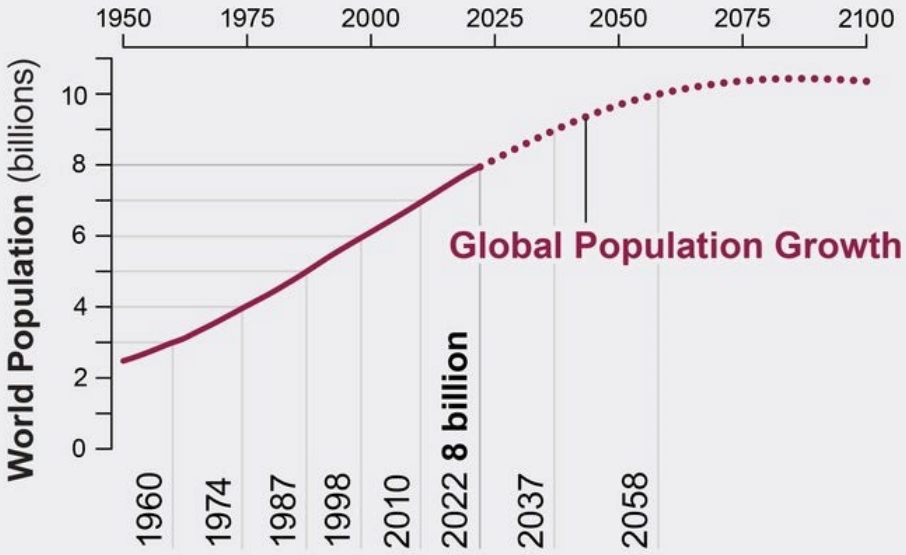


Classification on Image Data

Classification on Video Data



Regression



Global Population Prediction

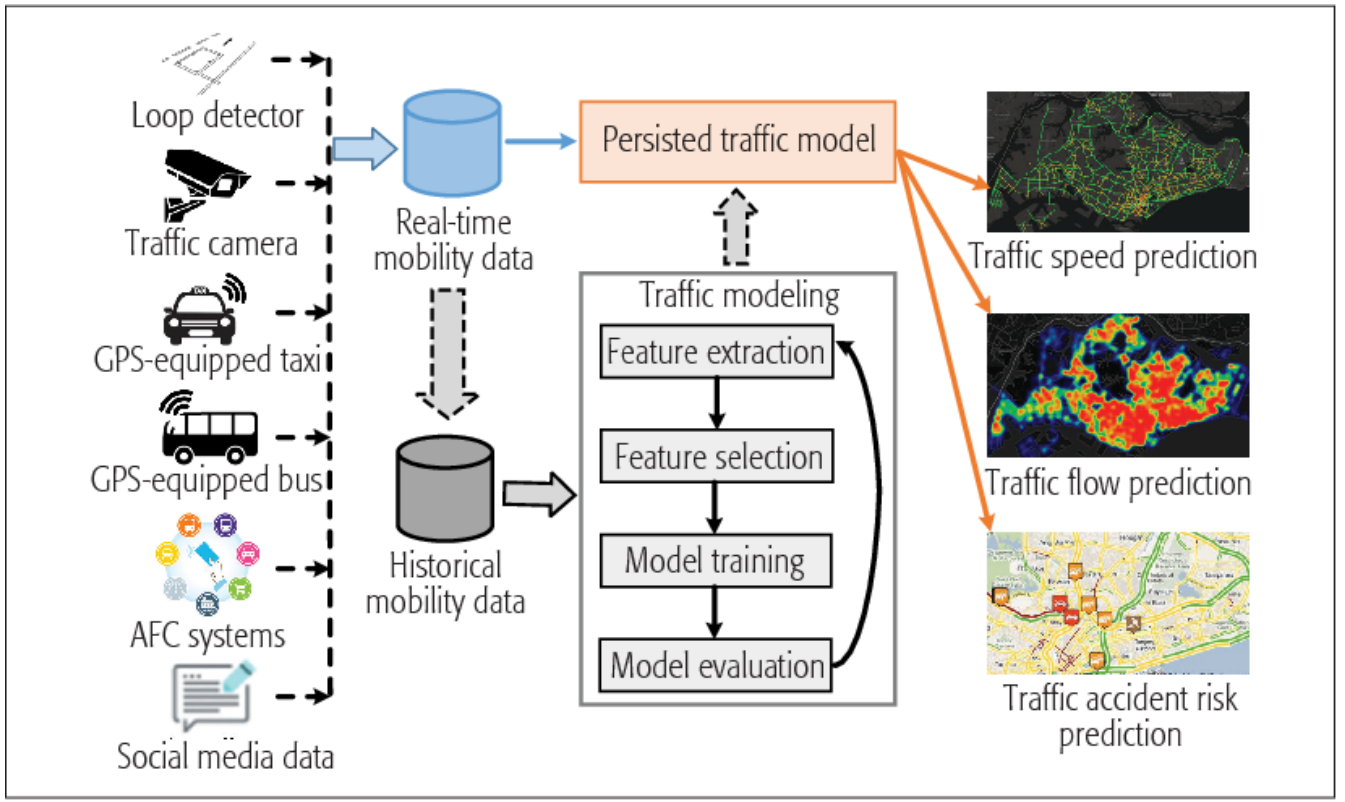
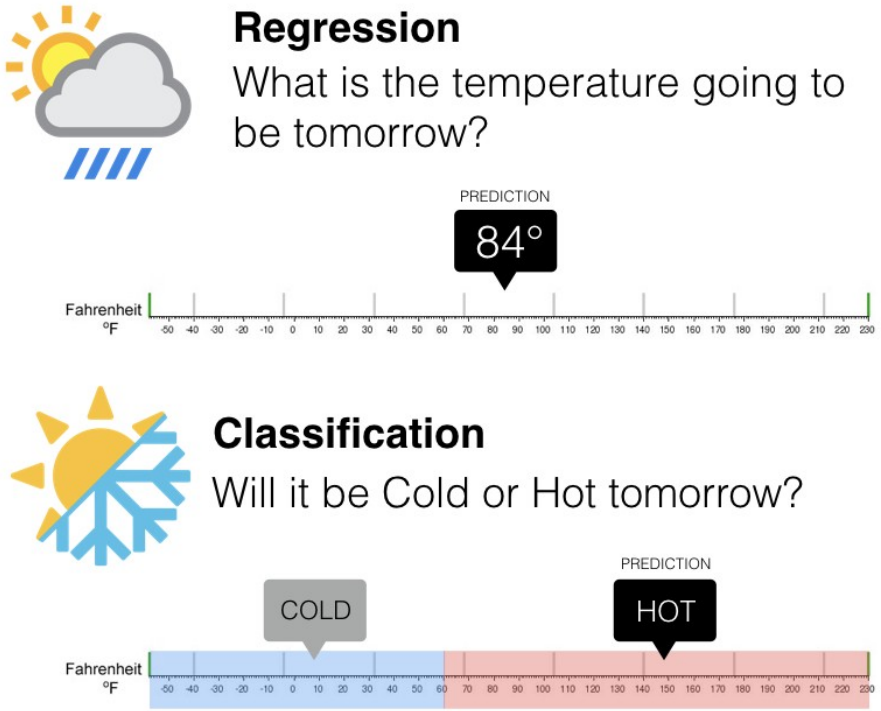


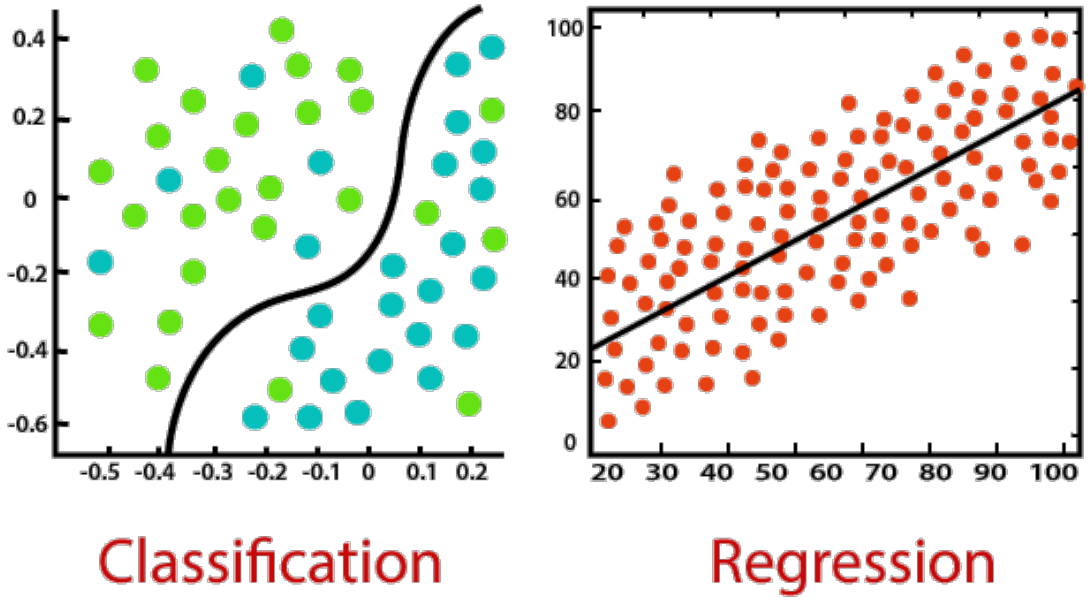
FIGURE 1. The basic components of urban traffic prediction.

Traffic Prediction

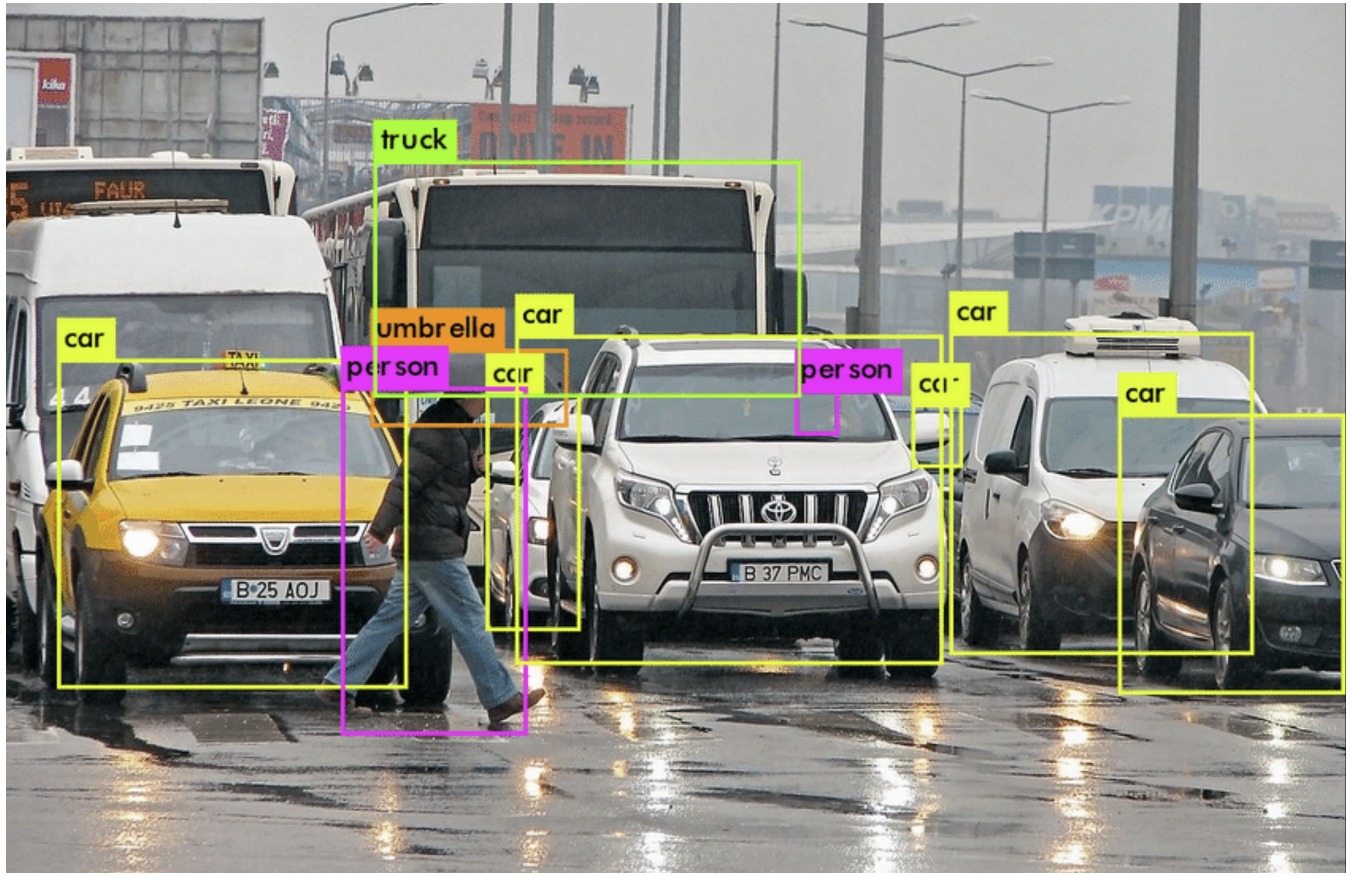
Prediction: Classification vs. Regression

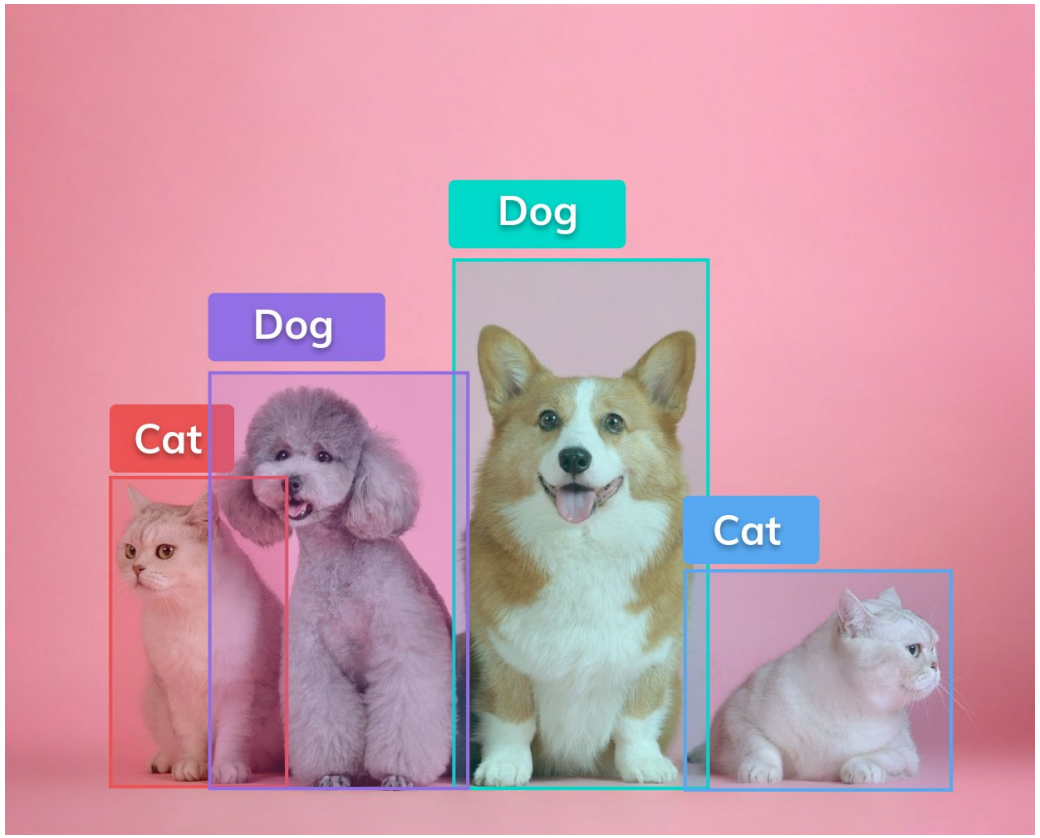
- **Classification**
  - ✓ predicts categorical class labels
- **Regression**
  - ✓ models continuous-valued functions





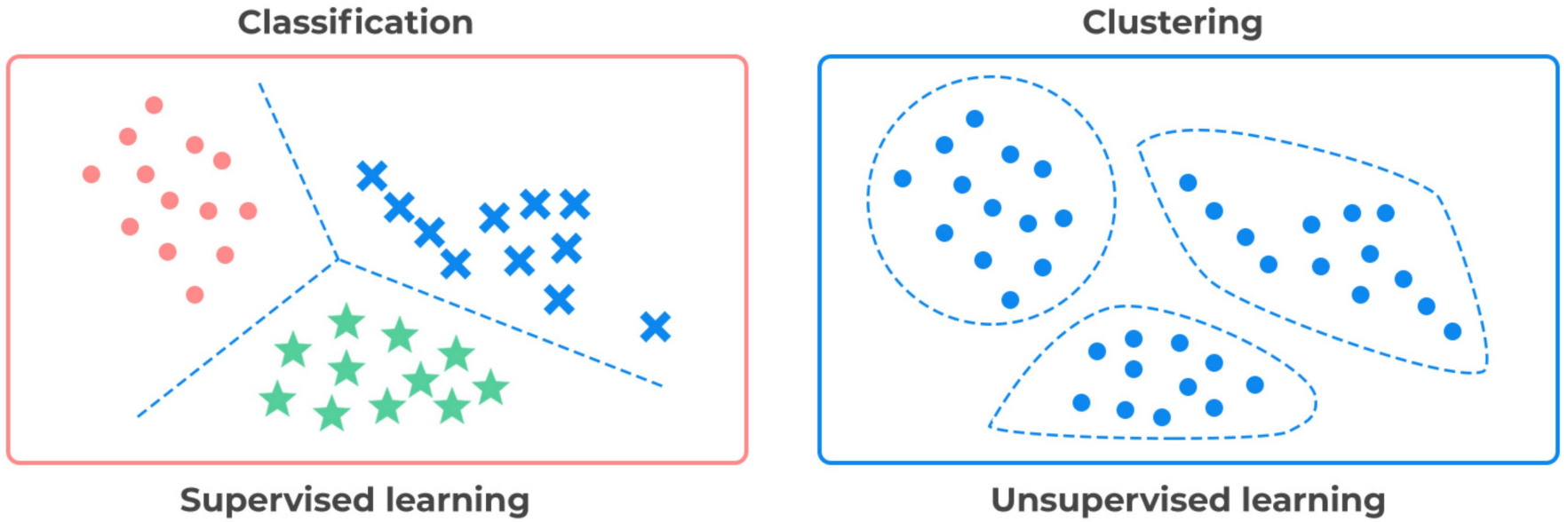
Object Detection



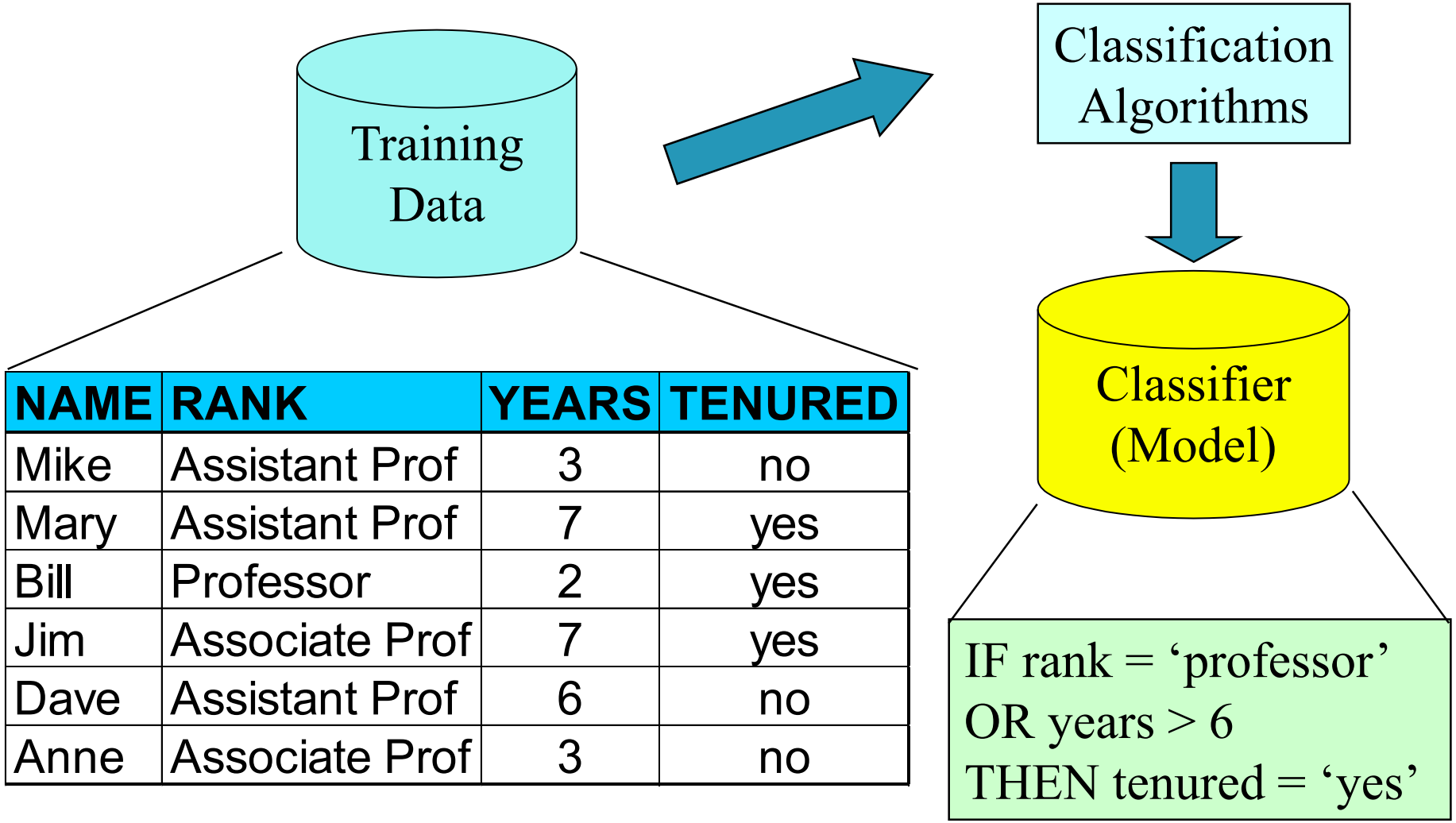


Supervised vs. Unsupervised Learning

- **Supervised learning**
  - ✓ Supervision: The training data (observations, measurements, etc.) are accompanied by labels indicating the class of the observations
  - ✓ New data is classified based on the training set
- **Unsupervised learning**
  - ✓ The class labels of training data is not available

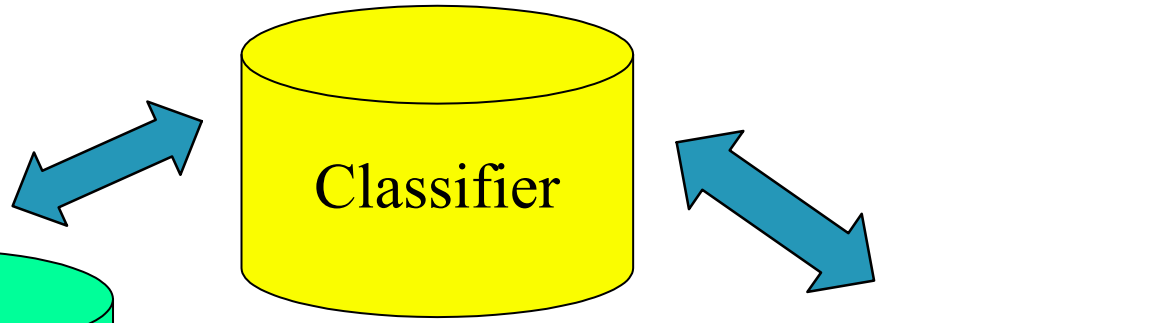


Process (1): Model Construction



NAME	RANK	YEARS	TENURED
Mike	Assistant Prof	3	no
Mary	Assistant Prof	7	yes
Bill	Professor	2	yes
Jim	Associate Prof	7	yes
Dave	Assistant Prof	6	no
Anne	Associate Prof	3	no

Process (2): Using the Model in Prediction



NAME	RANK	YEARS	TENURED
Tom	Assistant Prof	2	no
Merlisa	Associate Prof	7	no
George	Professor	5	yes
Joseph	Assistant Prof	7	yes

(Jeff, Professor, 4)

Tenured?

Yes

Data Splitting

- **Training Set**
  - ✓ The training set is the set of data we use to train a model
- **Validation Set**
  - ✓ The validation set is a set of data that we did not use when training our model that we use to assess how well the model performs on new data
  - ✓ The class labels of data in the validation set are known in advance
- **Test Set**
  - ✓ The test set is a set of data we did not use to train our model or use in the validation set to inform our choice of parameters/input features
  - ✓ In many Kaggle tasks, you are unaware of the labels for the test samples

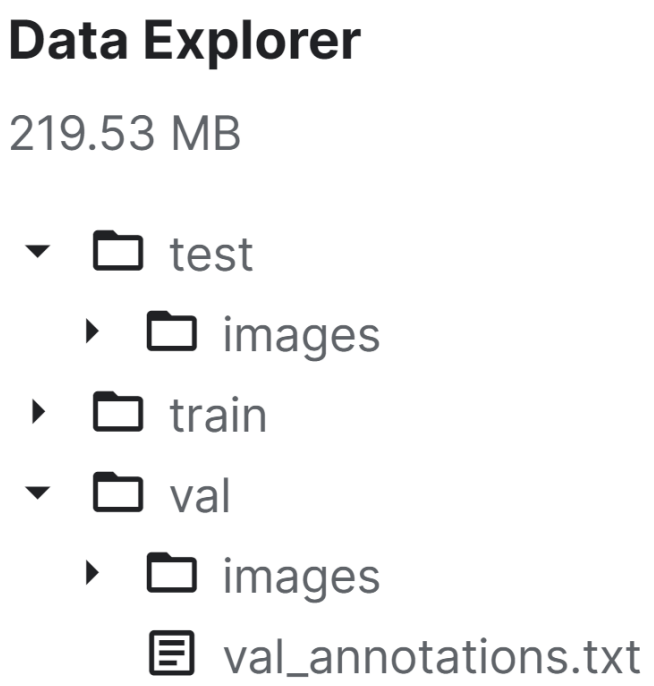
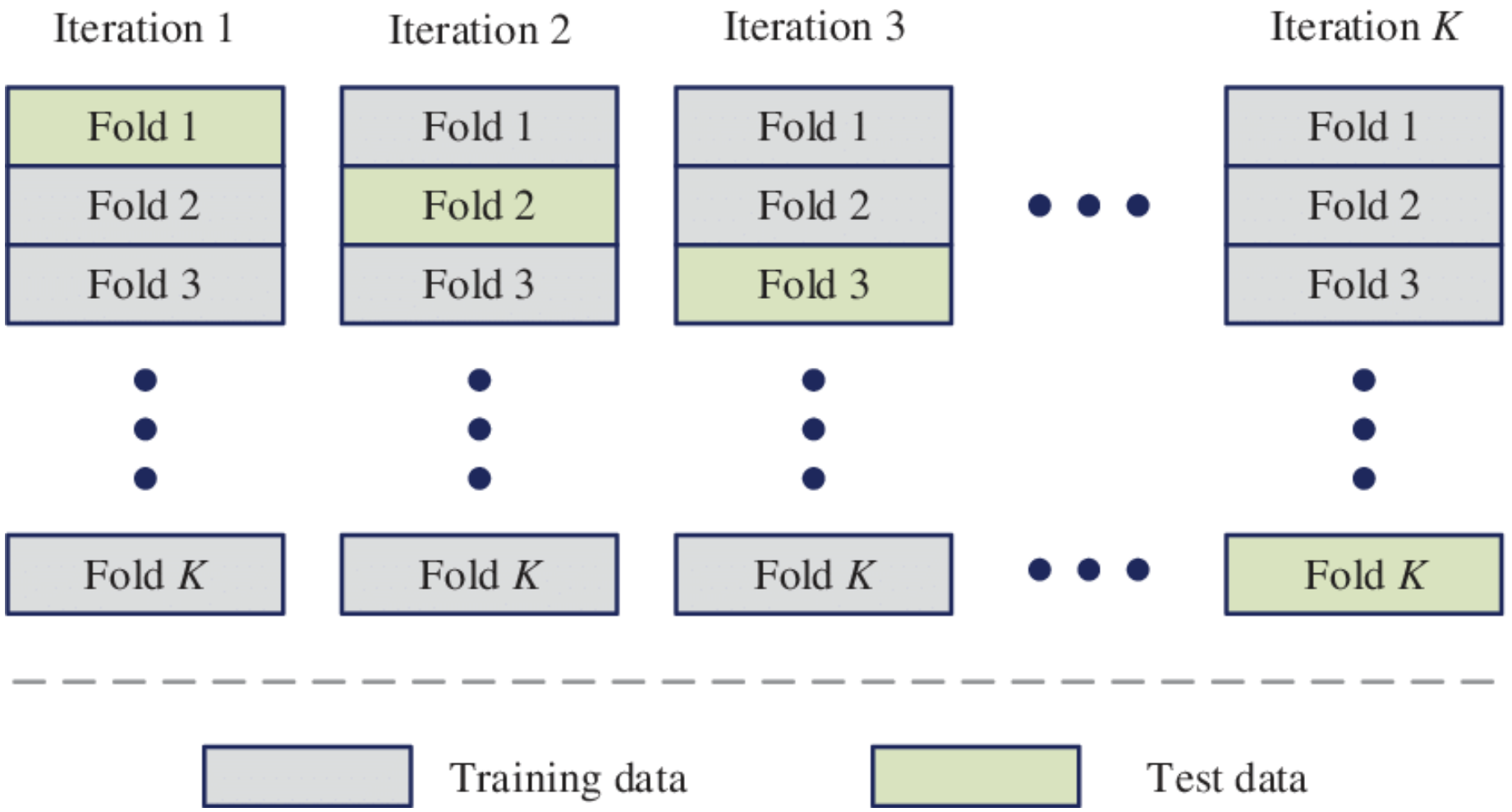


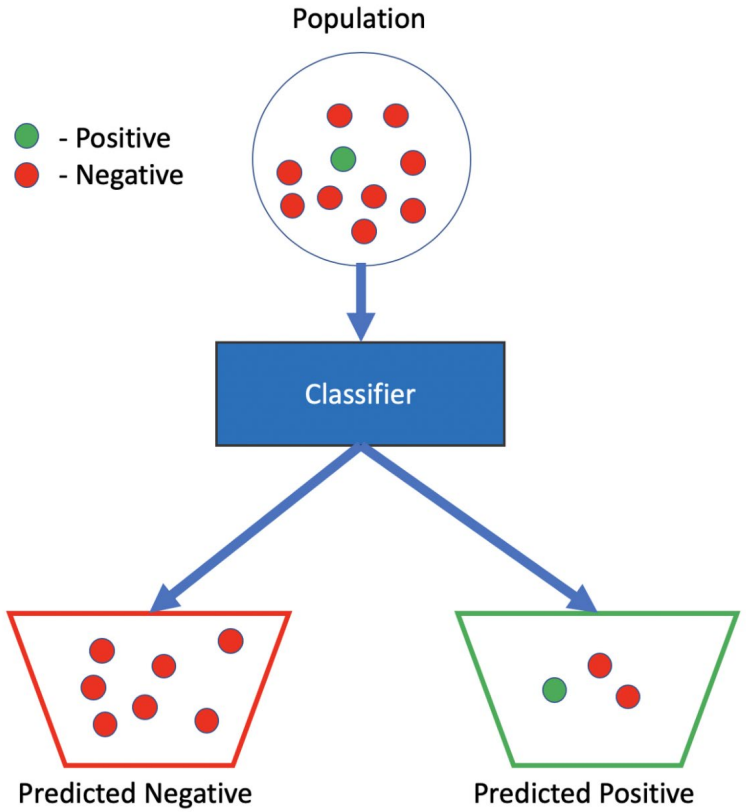
Image source: https://www.kaggle.com/c/image-detect/data

Cross Validation



Performance Metrics of Binary Classification

一个医院新开发了一套癌症AI诊断系统，想评估其性能好坏。我们把病人得了癌症定义为Positive，没得癌症定义为Negative。那么， 到底该用什么指标进行评估呢？



True Positive (TP): 把正样本成功预测为正。

True Negative (TN)：把负样本成功预测为负。

False Positive (FP)：把负样本错误地预测为正。

False Negative (FN)：把正样本错误的预测为负。

		Real	
		Positive	Negative
Predicted	Positive	1	2
	Negative	0	7

Performance Metrics of Binary Classification

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

$$F1\text{-}score = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

		Real	
		Positive	Negative
Predicted	Positive	1	2
	Negative	0	7

在诊断为癌症的一堆人中，到底有多少人真得了癌症？ → $\boldsymbol{precision = \frac{tp}{tp + fp} = \frac{1}{3} = 33\%}$

在一堆真的癌症病人中，有多少人被成功检测出癌症？ → $\boldsymbol{recall = \frac{tp}{tp + fn} = \frac{1}{1} = 100\%}$

在一堆癌症病人和正常人中，有多少人被系统给出了正确诊断结果（患癌或没患癌）？

参考：https://www.cvmart.net/community/detail/2840

Performance Metrics of Multiclass Classification

Actual \ Predicted	Cat	Dog	Pig
Cat	40	20	10
Dog	35	85	40
Pig	0	10	20

	TP	FN	FP
Cat	40	30	35
Dog	85	75	30
Pig	20	10	50

20只Cat和10只Pig被错误预测成Dog

10只Pig被错误预测成Dog

Accuracy: (40+85+20)/260=0.558

$P_{cat} = 8/15, P_{dog} = 17/23, P_{pig} = 2/7$ (P代表Precision)

$R_{cat} = 4/7, R_{dog} = 17/32, R_{pig} = 2/3$ (R代表Recall)

$$\text{Macro-Precision} = \frac{P_{cat} + P_{dog} + P_{pig}}{3} = 0.5194$$

$$\text{Macro-Recall} = \frac{R_{cat} + R_{dog} + R_{pig}}{3} = 0.5898$$

Performance Metrics of Regression

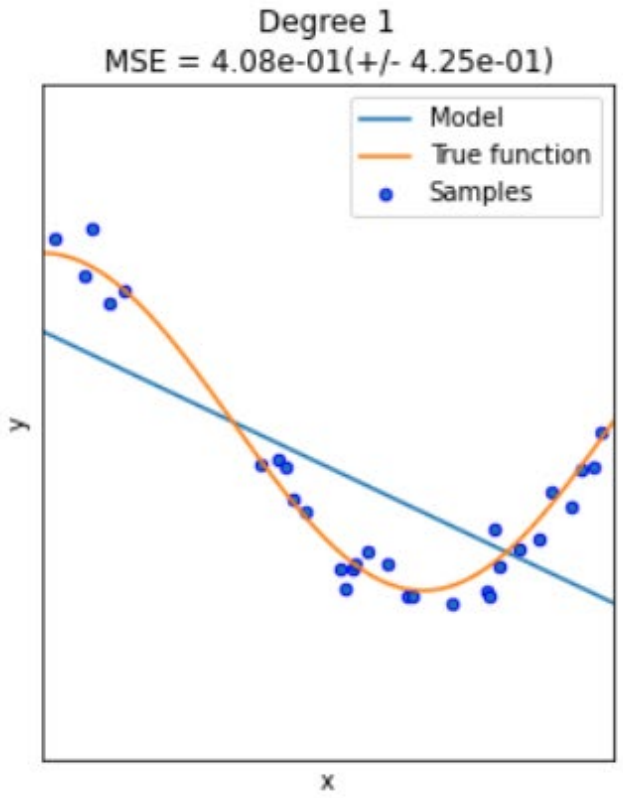
$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\hat{y}_i - y_i|$$

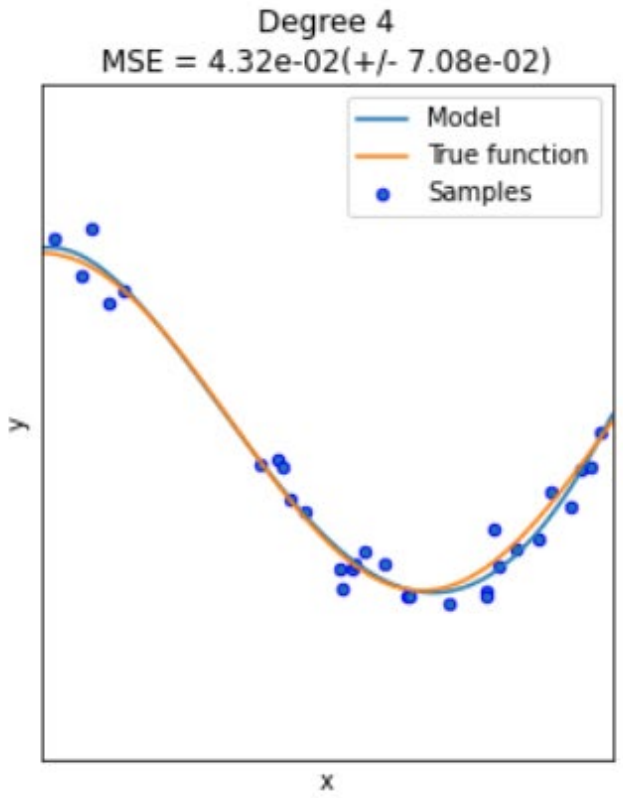
$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$$

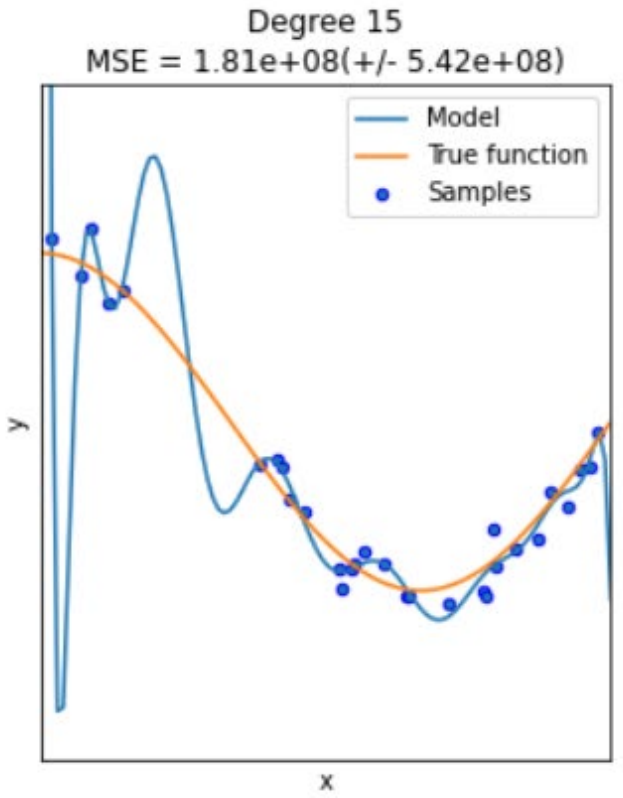
$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$$

Overfitting and Underfitting

- **Overfitting**
  - ✓ The model performs well on the training data but does not perform well on the evaluation data
- **Underfitting**
  - ✓ The model performs poorly on the training







Agenda

- Classification: Basic Concepts
- kNN Classifier
- Decision Tree
- Bayes Classification
- Support Vector Machine

kNN Classifier

- **Non-parametric algorithm, which means it does not make any assumption on underlying data**

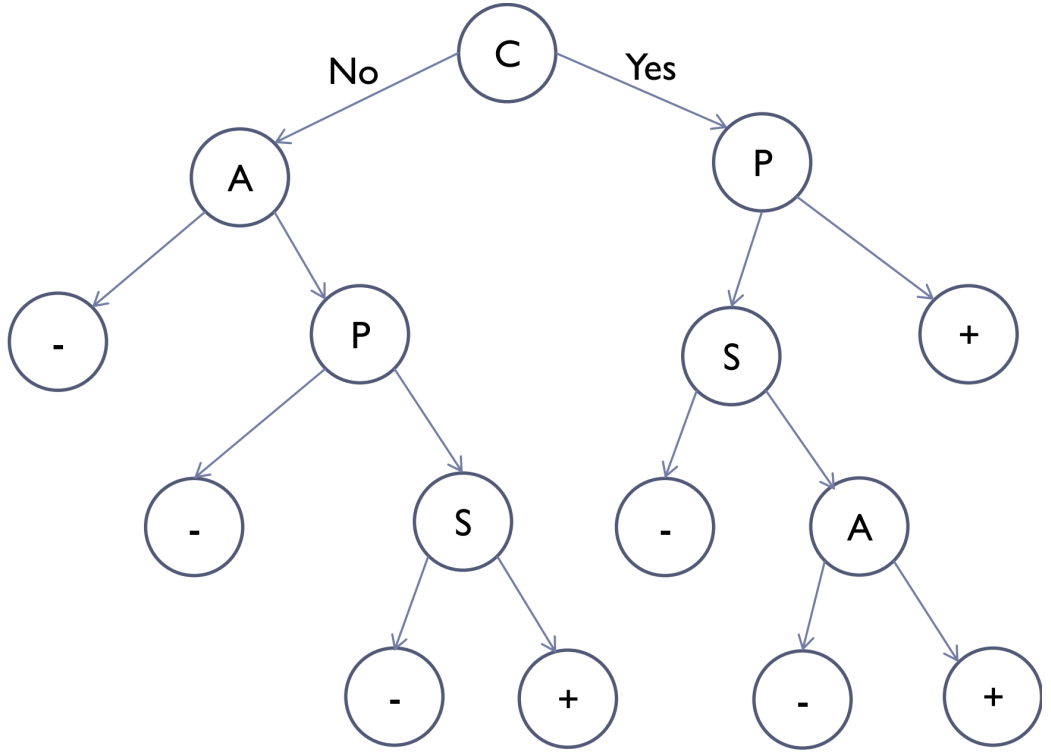
- **Step-1:** Select the number K of the neighbors
- **Step-2:** Calculate the Euclidean distance of **K number of neighbors**
- **Step-3:** Take the K nearest neighbors as per the calculated Euclidean distance.
- **Step-4:** Among these k neighbors, count the number of the data points in each category.
- **Step-5:** Assign the new data points to that category for which the number of the neighbor is maximum.

Agenda

- Classification: Basic Concepts
- kNN Classifier
- Decision Tree
- Bayes Classification
- Support Vector Machine

Decision Tree Structure

- Each leaf node represents a class label
- Each internal node corresponds to denotes a test on an attribute
  - ✓ Edges to children for each of the possible values of that attribute



- Attributes:
  - A: age>40
  - C: chest pain
  - S: smoking
  - P: physical test
- Label:
  - Heart disease (+), No heart disease (-)

Decision Tree Learning

- The most common strategy for DT learning is a greedy top-down approach

Basic decision tree building algorithm:

- Pick some feature/attribute (how to pick the "best"?)
- Partition the data based on the value of this attribute
- Recurse over each new partition (when to stop?)

Slide reference: Hungyi-Lee

Attribute Selection

- **Max-Gain: Choose the attribute that has the largest expected information gain**

Information Gain (IG)

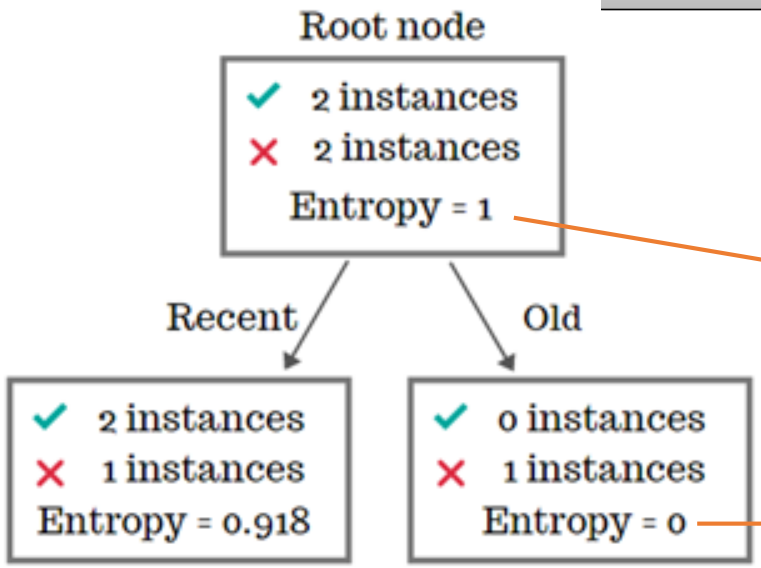
IG = Entropy (Parent) - weighted_avg * Entropy (Children)

Entropy (Parent) = 1

Entropy $H(x) = -\sum_{i=1}^{n} P(x=i)\log_2 P(x=i)$

Information gain for Age

Age	Mileage	Road Tested	Buy
Recent	Low	Yes	Buy ✓
Recent	High	Yes	Buy ✓
Old	Low	No	Don't buy ✗
Recent	High	No	Don't buy ✗



-(0.5*-1 + 0.5*-1)=1

-(1*0)=0

Children Entropy: 0.75*0.918+0.25*0=0.688

Information Gain: 1 – 0.688 = 0.312

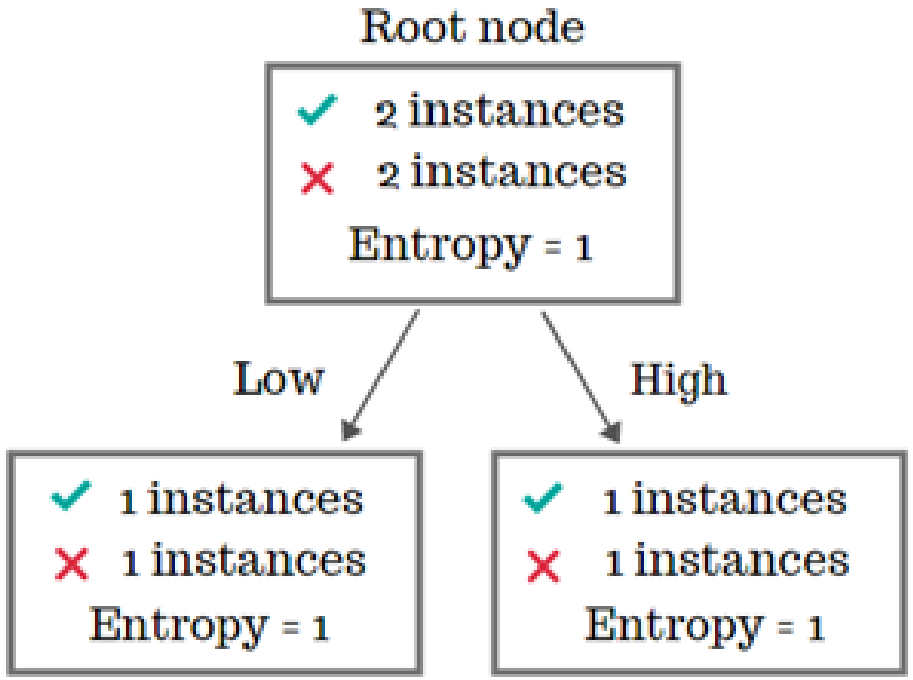
Example from: https://towardsdatascience.com/decision-tree-part-2-34b31b1dc328

Attribute Selection

- **Max-Gain: Choose the attribute that has the largest expected information gain**

Information gain for Milage

Age	Mileage	Road Tested	Buy
Recent	Low	Yes	Buy ✓
Recent	High	Yes	Buy ✓
Old	Low	No	Don't buy ✗
Recent	High	No	Don't buy ✗



Children Entropy = 2/4 (1) + 2/4 (1) = 1

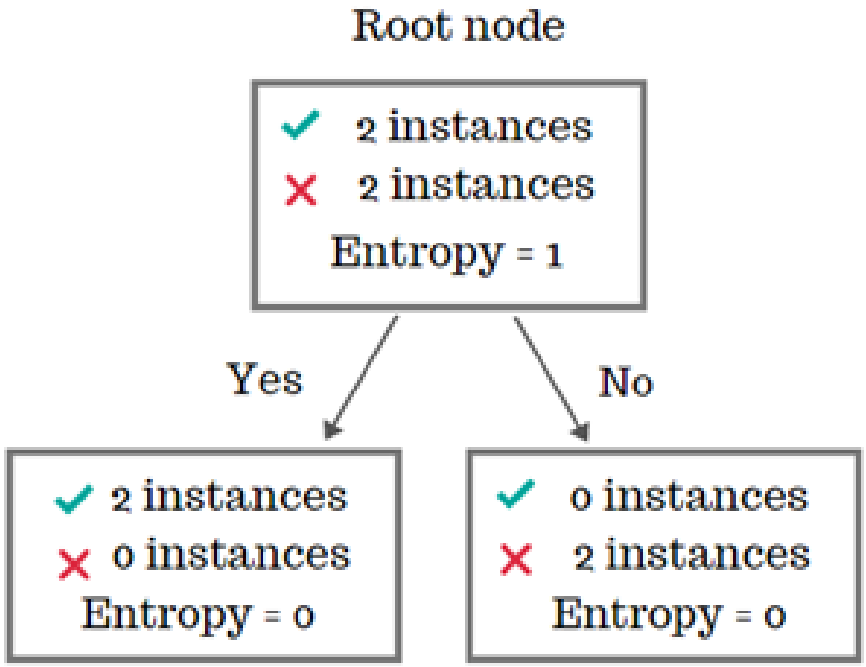
Information gain = 1 - 1 = 0

Attribute Selection

- **Max-Gain: Choose the attribute that has the largest expected information gain**

Information gain for Road Tested

Age	Mileage	Road Tested	Buy
Recent	Low	Yes	Buy ✓
Recent	High	Yes	Buy ✓
Old	Low	No	Don't buy ✗
Recent	High	No	Don't buy ✗

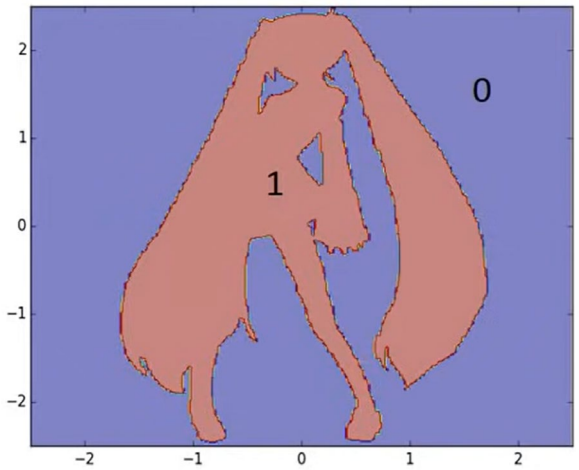


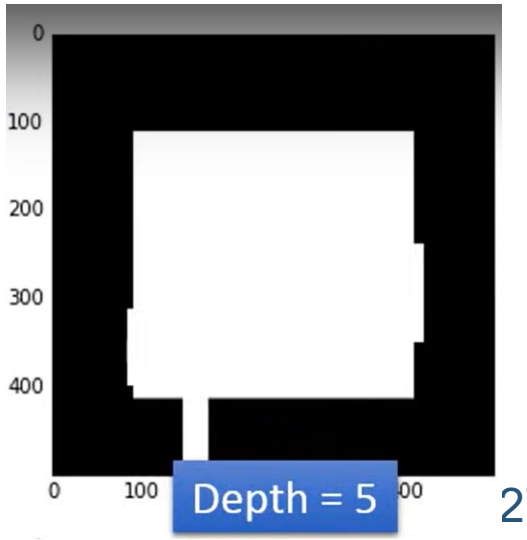
Children Entropy = 2/4 (0) + 2/4 (0) = 0

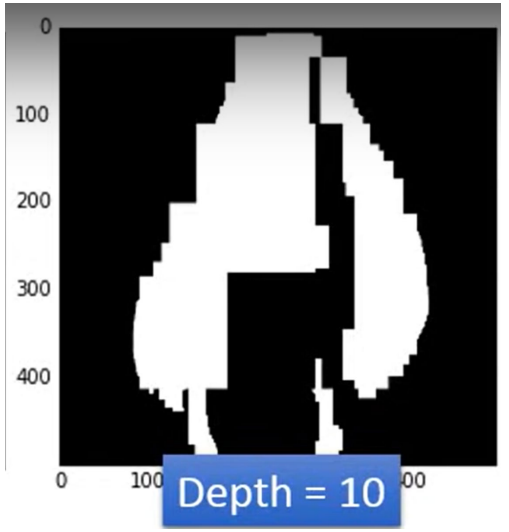
Information gain = 1 - 0 = 1

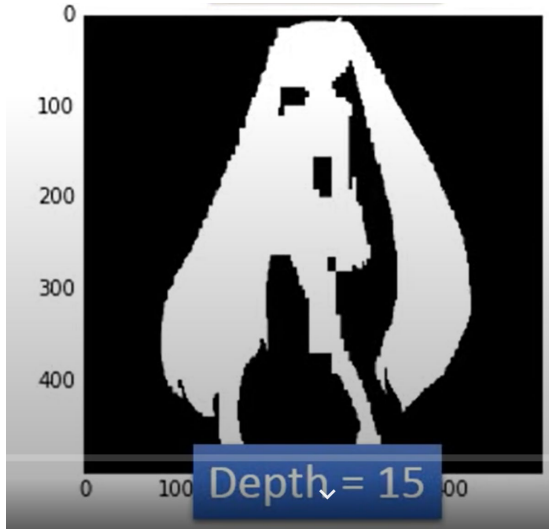
Decision Tree Learning

- **What functions can be represented by decision trees?**
  - ✓ Every function can be represented by a sufficiently complicated decision tree









Slide reference: Hungyi-Lee

When to Stop

- **It stops at smallest acceptable tree**
  - ✓ Occam's razor: prefer the simplest hypothesis that fits the data
  - ✓ If all attributes have small information gain, then don't recurse

Agenda

- Classification: Basic Concepts
- kNN Classifier
- Decision Tree
- Bayes Classification
- Support Vector Machine

Naïve Bayesian Classification

	Animals	Size of Animal	Body Color	Can we Pet them
0	Dog	Medium	Black	Yes
1	Dog	Big	White	No
2	Rat	Small	White	Yes
3	Cow	Big	White	Yes
4	Cow	Small	Brown	No
5	Cow	Big	Black	Yes
6	Rat	Big	Brown	No
7	Dog	Small	Brown	Yes
8	Dog	Medium	Brown	Yes
9	Cow	Medium	White	No
10	Dog	Small	Black	Yes
11	Rat	Medium	Black	No
12	Rat	Small	Brown	No
13	Cow	Big	White	Yes

Test Data: (Cow, Medium, Black)

P(Yes|Cow,Medium,Black)

P(No|Cow,Medium,Black)

Which probability is higher?

Example from: https://www.analyticsvidhya.com/blog/2021/09/naive-bayes-algorithm-a-complete-guide-for-data-science-enthusiasts/

Naïve Bayesian Classification

- **Why is it called Naïve?**
  - ✓ It assumes features are independent of each other, which is not true in real life
- **Why is it popular?**
  - ✓ It is a probabilistic approach and the predictions can be made instantly
  - ✓ It can be used for both binary and multi-class classification problems

$$P(B|A) = \frac{P(A|B)P(B)}{P(A)}.$$

$$P(Y = k|X1, X2 \ldots . Xn) = \frac{P(X1|Y = k) * P(X2|Y = k) \ldots \ldots * P(Xn|Y = k) * P(Y = k)}{P(X1) * P(X2) \ldots . * P(Xn)}$$

Example from: https://towardsdatascience.com/decision-tree-part-2-34b31b1dc328

Naïve Bayesian Classification

	Animals	Size of Animal	Body Color	Can we Pet them
0	Dog	Medium	Black	Yes
1	Dog	Big	White	No
2	Rat	Small	White	Yes
3	Cow	Big	White	Yes
4	Cow	Small	Brown	No
5	Cow	Big	Black	Yes
6	Rat	Big	Brown	No
7	Dog	Small	Brown	Yes
8	Dog	Medium	Brown	Yes
9	Cow	Medium	White	No
10	Dog	Small	Black	Yes
11	Rat	Medium	Black	No
12	Rat	Small	Brown	No
13	Cow	Big	White	Yes

Test Data: (Cow, Medium, Black)

P1=P(Cow |Yes)=3/8
P2=P(Medium |Yes)=2/8
P3=P(Black |Yes)=3/8
P(Yes)=8/14

P(Yes|Cow,Medium,Black)
=P1*P2*P3*P(Yes)/P(Cow,Medium,Black)
=3/8*2/8*3/8*8/14/P(Cow,Medium,Black)
=0.02/P(Cow,Medium,Black)

P(No|Cow,Medium,Black)
=2/6*2/6*1/6*6/14/P(Cow,Medium,Black)
=0.0079/P(Cow,Medium,Black)

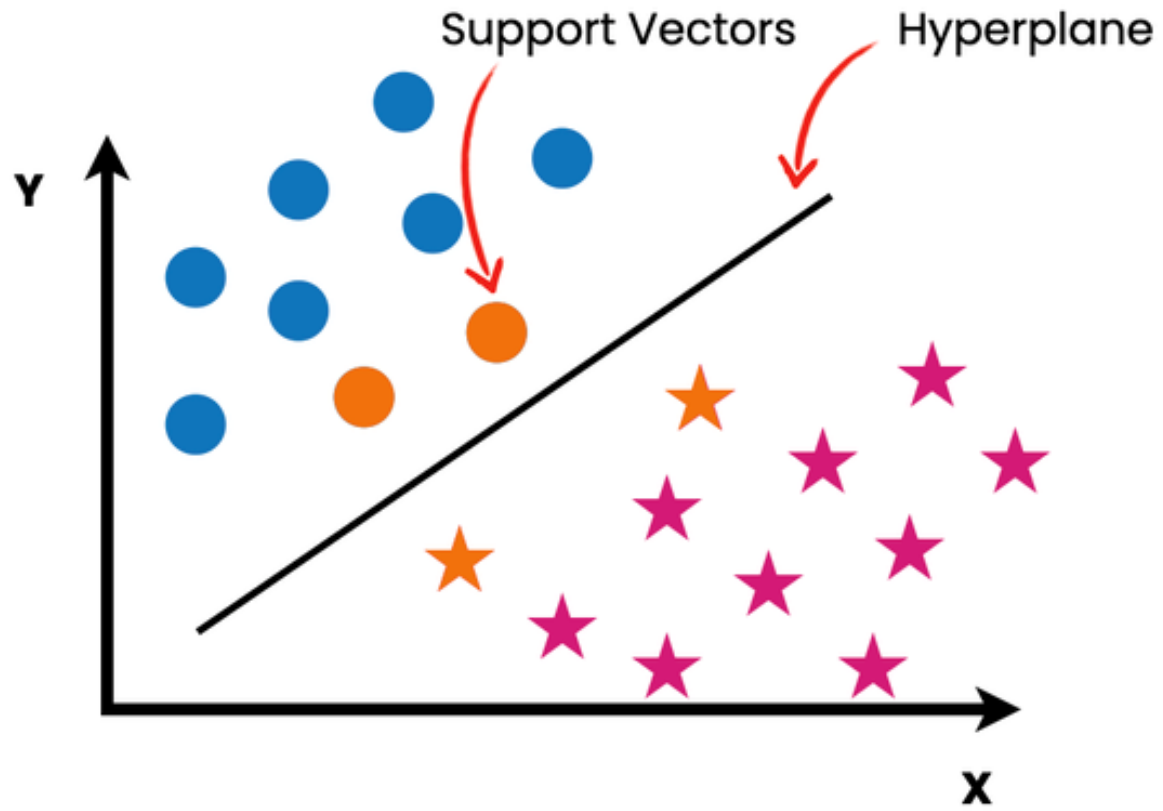
Example from: https://www.analyticsvidhya.com/blog/2021/09/naive-bayes-algorithm-a-complete-guide-for-data-science-enthusiasts/

Agenda

- Classification: Basic Concepts
- kNN Classifier
- Decision Tree
- Bayes Classification
- Support Vector Machine

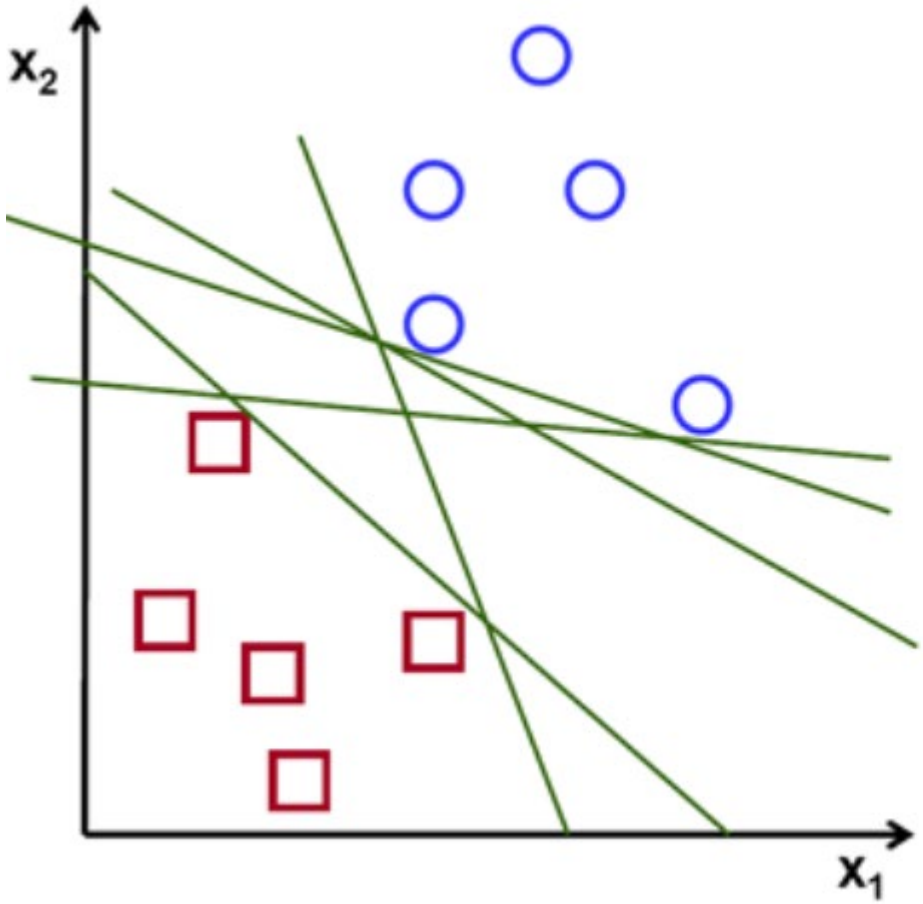
Support Vector Machine

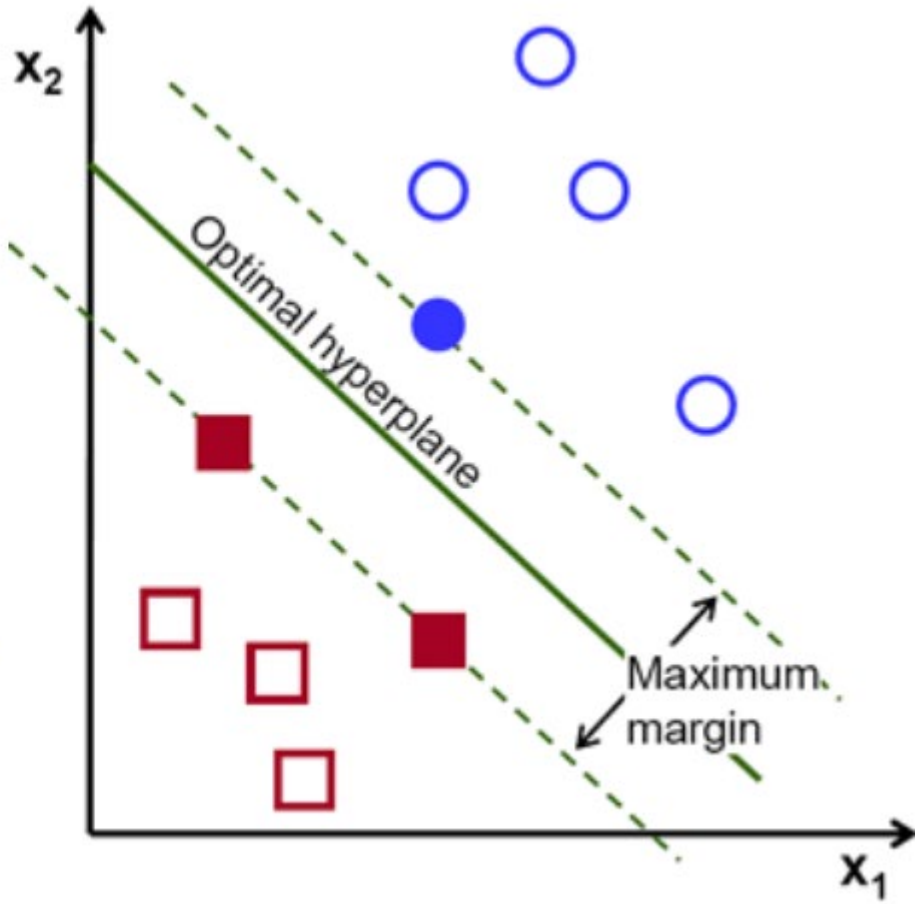
- Basic idea



Support Vector Machine

- Which line is the best?

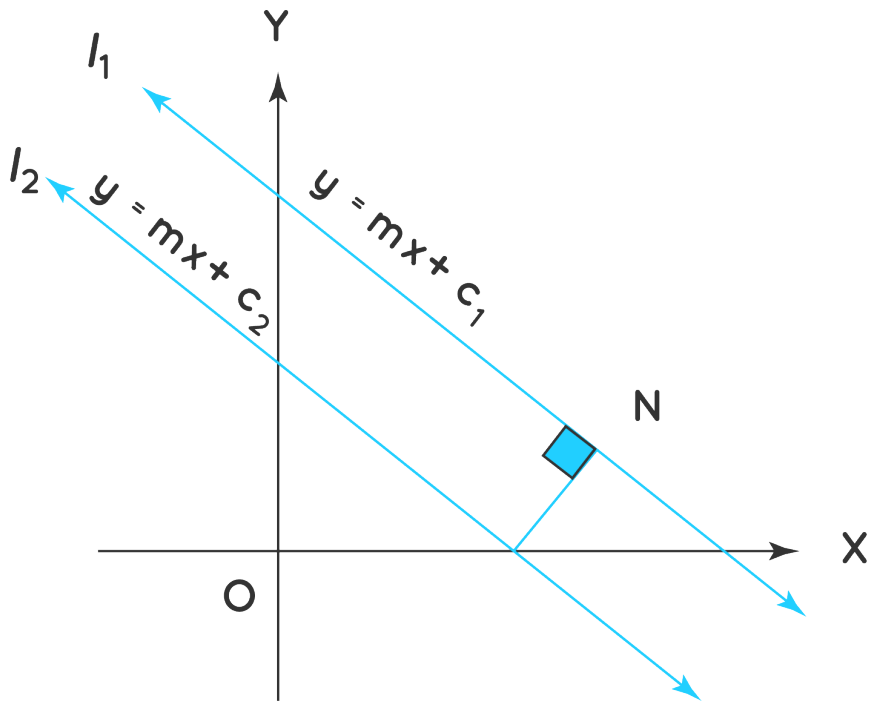




Support Vector Machine

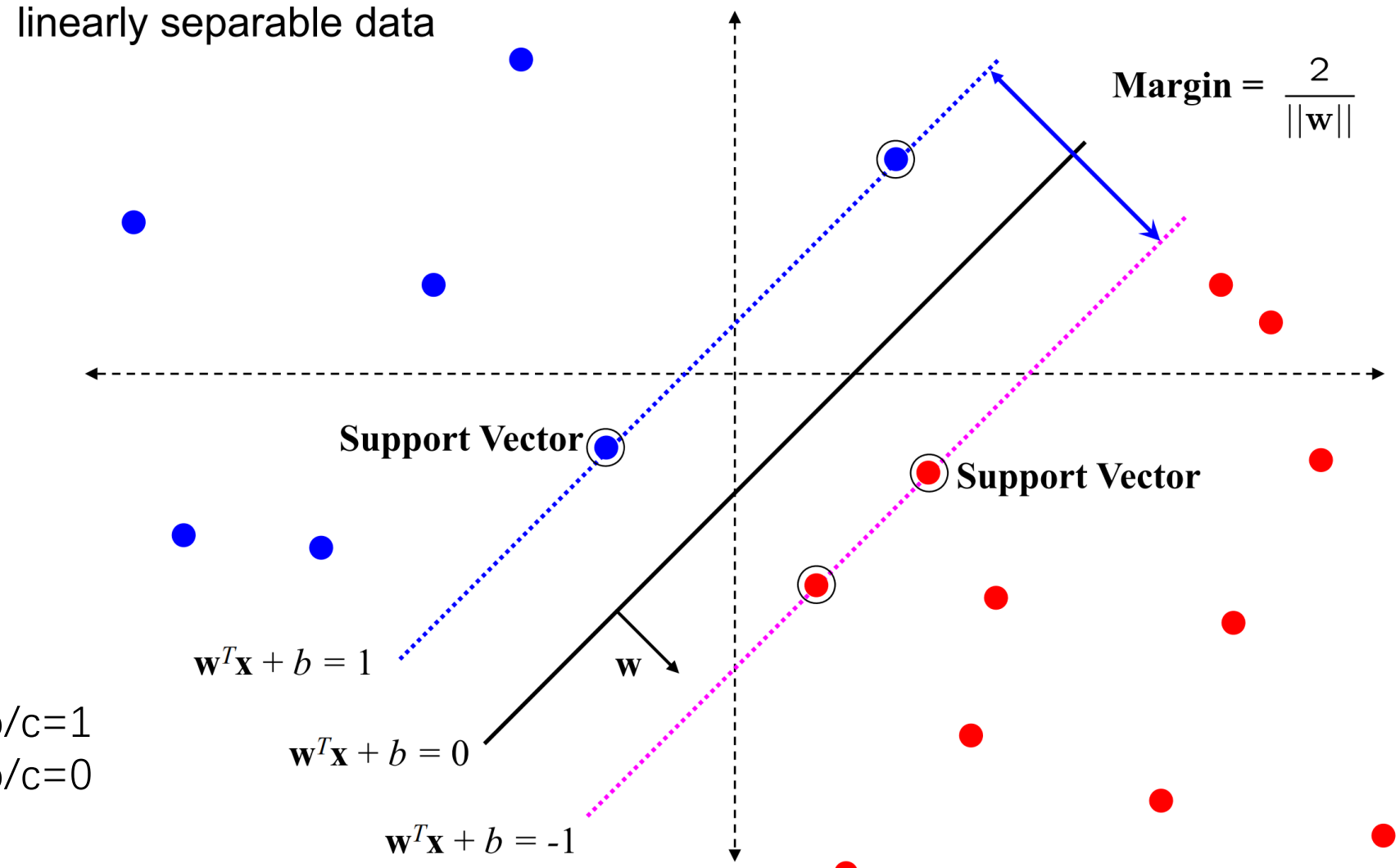
- **How to represent a line?**
  - ✓ We can use y=ax+b
  - ✓ We can rewrite it as $[a,-1]\begin{pmatrix}x\\y\end{pmatrix}+b=0$
- **The distance between two parallel lines**

Distance Between Two Lines



$$d = \frac{|c_2 - c_1|}{\sqrt{1 + m^2}}$$

Support Vector Machine



$\mathbf{w}^T\mathbf{x}+b=c$
$\mathbf{w}^T\mathbf{x}+b=0$

$(\mathbf{w}/c)^T\mathbf{x}+b/c=1$
$(\mathbf{w}/c)^T\mathbf{x}+b/c=0$

Example from: https://www.robots.ox.ac.uk/~az/lectures/ml/lect2.pdf

Support Vector Machine

- Learning the SVM can be formulated as an optimization:

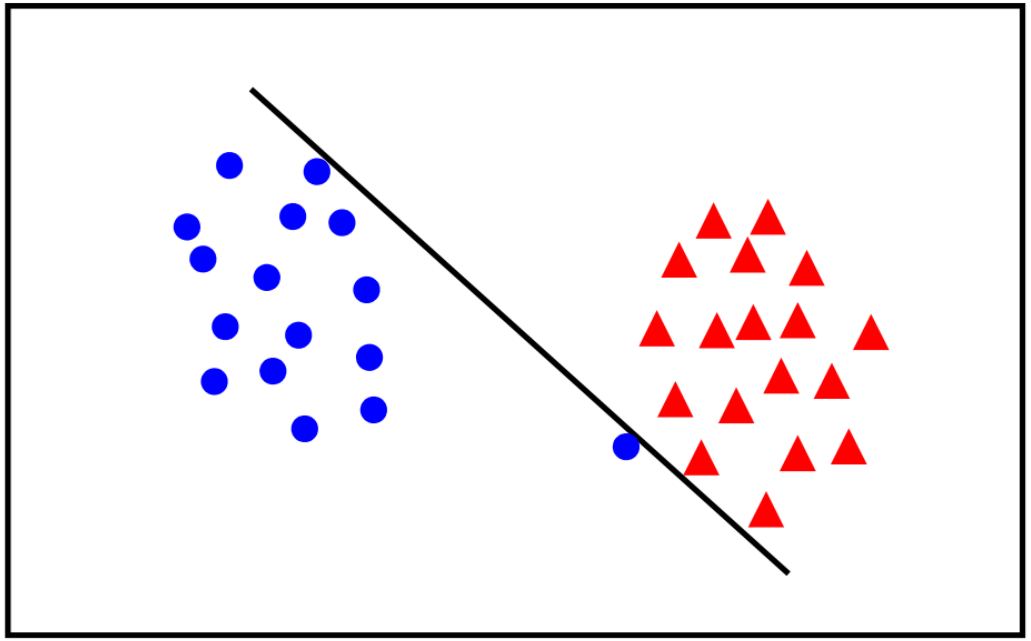
$$\max_{\mathbf{w}} \frac{2}{||\mathbf{w}||} \text{ subject to } \mathbf{w}^\top \mathbf{x}_i + b \begin{array}{ll} \geq 1 & \text{if } y_i = +1 \\ \leq -1 & \text{if } y_i = -1 \end{array} \text{ for } i = 1 \ldots N$$

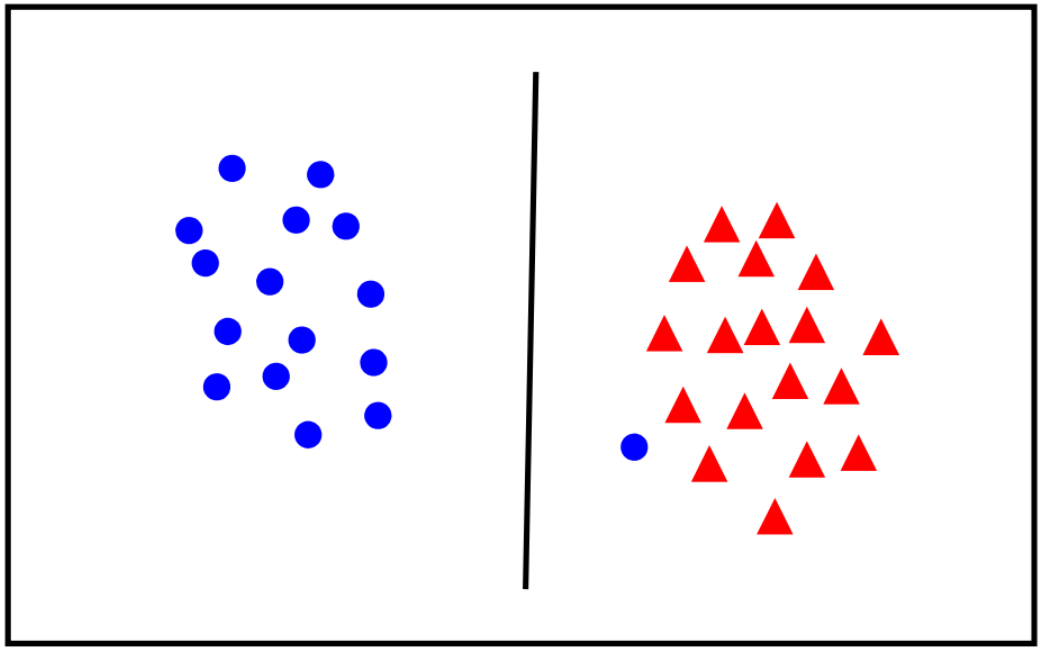
- Or equivalently

$$\min_{\mathbf{w}} ||\mathbf{w}||^2 \text{ subject to } y_i \left(\mathbf{w}^\top \mathbf{x}_i + b \right) \geq 1 \text{ for } i = 1 \ldots N$$

- This is a quadratic optimization problem subject to linear constraints and there is a unique minimum

Support Vector Machine





Support Vector Machine (Soft Margin Solution)

The optimization problem becomes

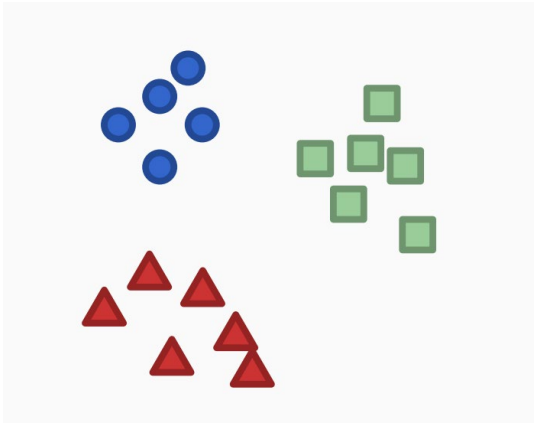
$$\min_{\mathbf{w}\in\mathbb{R}^d,\xi_i\in\mathbb{R}^+} ||\mathbf{w}||^2 + C\sum_i^N \xi_i$$

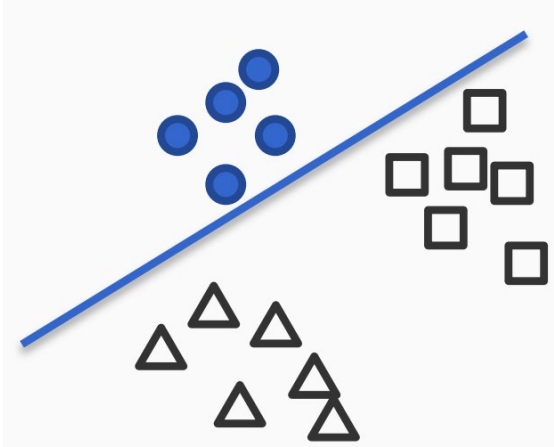
subject to

$$y_i\left(\mathbf{w}^\top \mathbf{x}_i + b\right) \geq 1 - \xi_i \text{ for } i = 1\ldots N$$

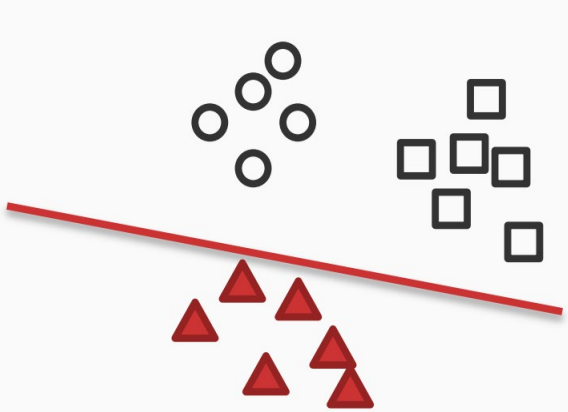
- Every constraint can be satisfied if ξ_i is sufficiently large
- C is a regularization parameter:
  - small C allows constraints to be easily ignored $\rightarrow$ large margin
  - large C makes constraints hard to ignore $\rightarrow$ narrow margin
  - $C = \infty$ enforces all constraints: hard margin

Support Vector Machine

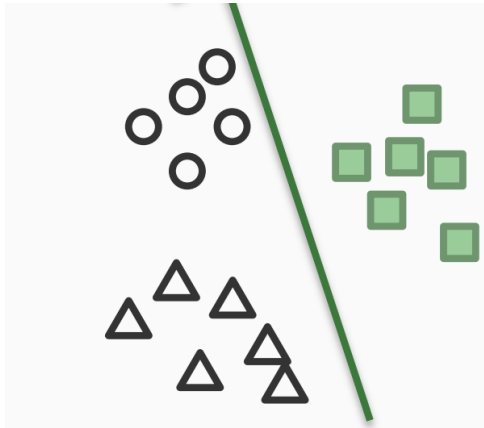




$w_{blue}^T x > 0$ for blue circle inputs



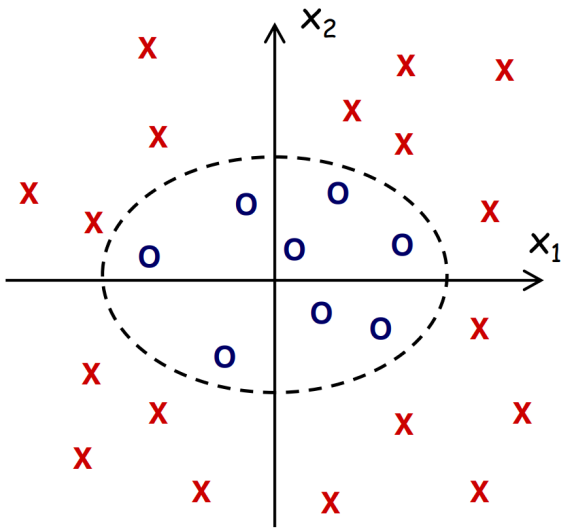
$w_{red}^T x > 0$ for red triangle inputs

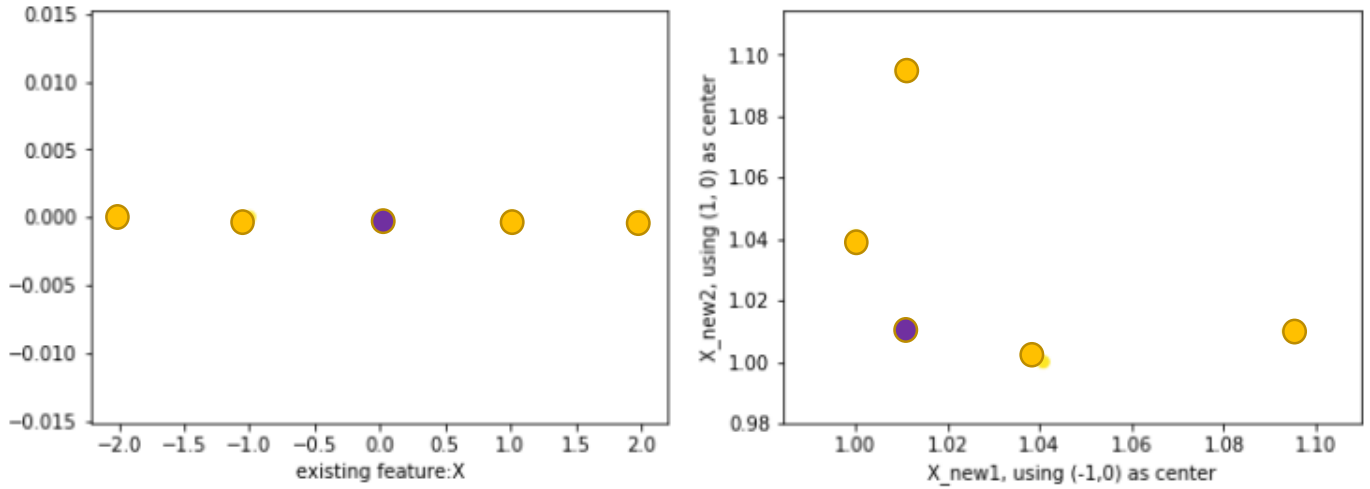


$w_{green}^T x > 0$ for green square inputs

Example from: https://svivek.com/teaching/lectures/slides/multiclass/multiclass-full.pdf

Support Vector Machine





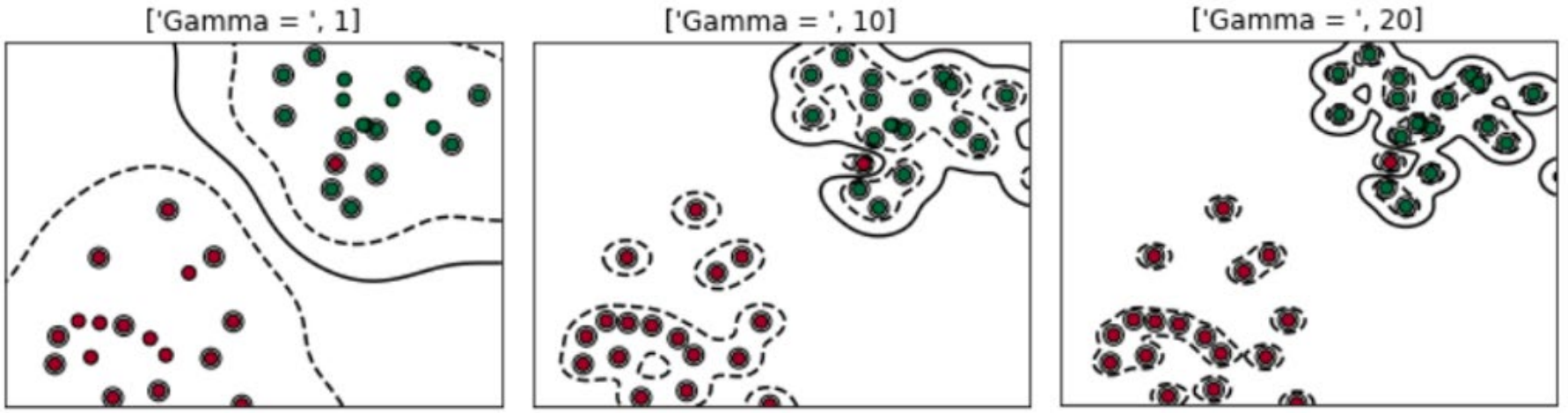


Image source: https://towardsdatascience.com/support-vector-machine-simply-explained-fee28eba5496