Intro to ML

marco milanesio

UNIVERSITÉ CÔTE D'AZUR

The problem

- *n* samples
- predict properties of the unknown
- that is: learn what the properties are
- learning:
  - supervised
    - we know some of the attributes
  - unsupervised
    - we know nothing (almost)

ML in a nutshell

- supervised learning
  - classification
    - finite set of labels
  - regression
    - “classification” in the continuum
- unsupervised learning:
  - clustering
    - “similarity”
  - density estimation
    - distribution
  - dimensionality reduction

pipeline

- gather the data
- clean the data
- create a model
- fit a model
- predict
- evaluate

training/testing

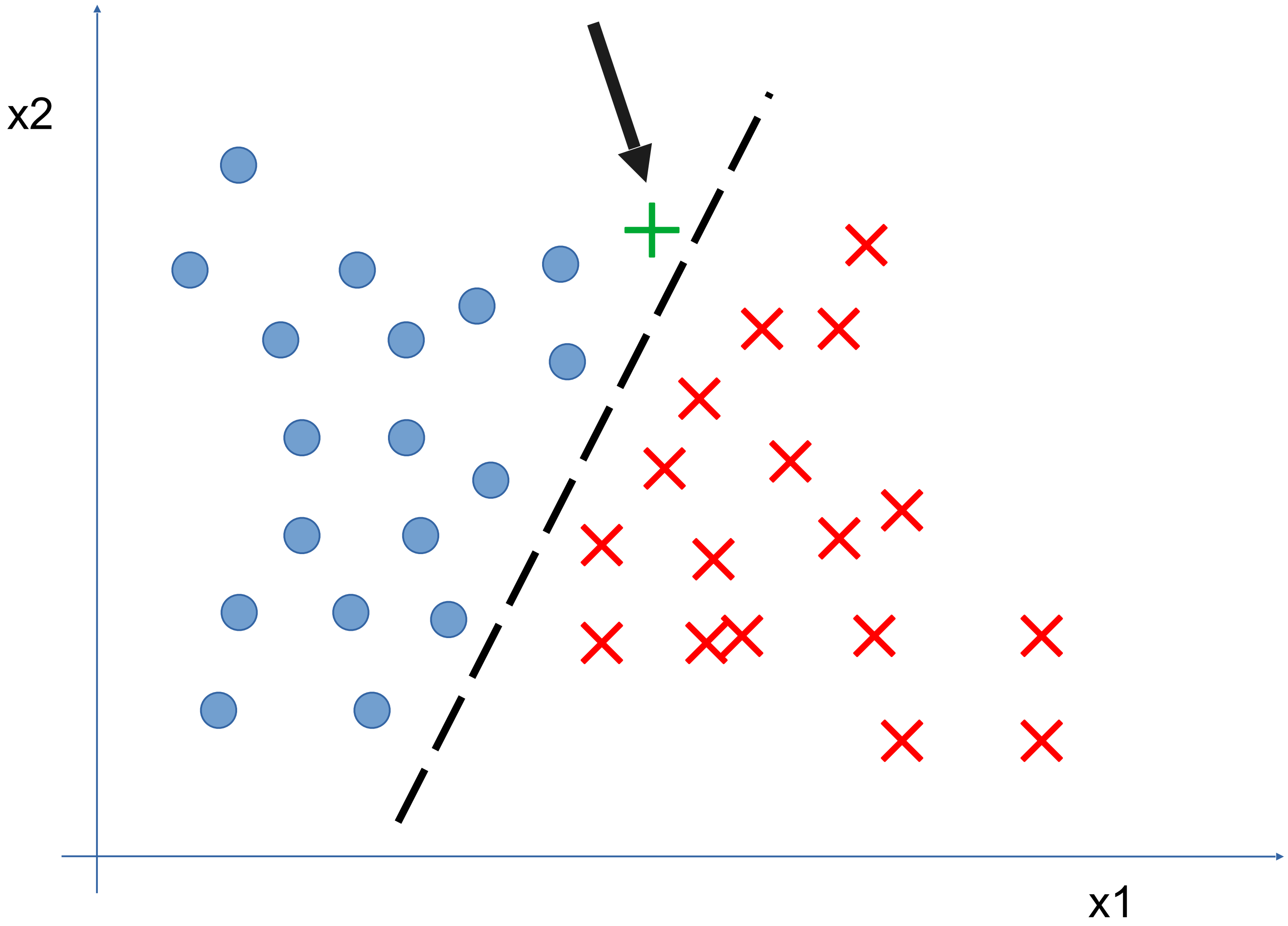
- learning from training set
- predicting on testing set (unknown)
- 80-20 / 70-30
- overfitting
- imbalanced datasets:
  - oversampling
  - undersampling

supervised learning

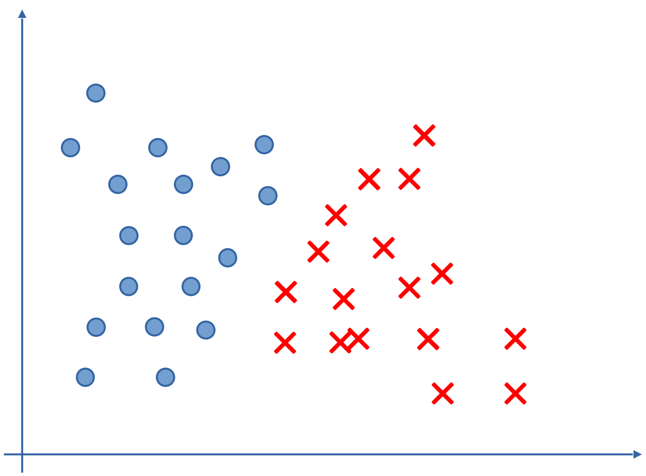
classification

- Goal: predict the **categorical** class labels
  - discrete
  - unordered
  - group membership
- Binary classification
  - spam / no spam
  - cat / no cat
- Multi-class classification
  - handwritten digits

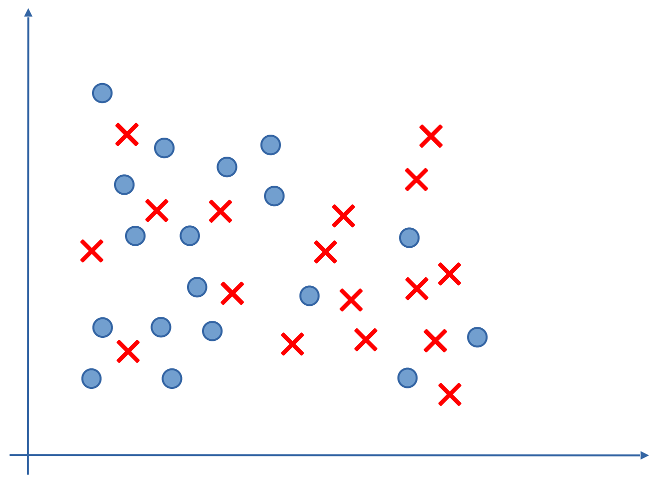
classification



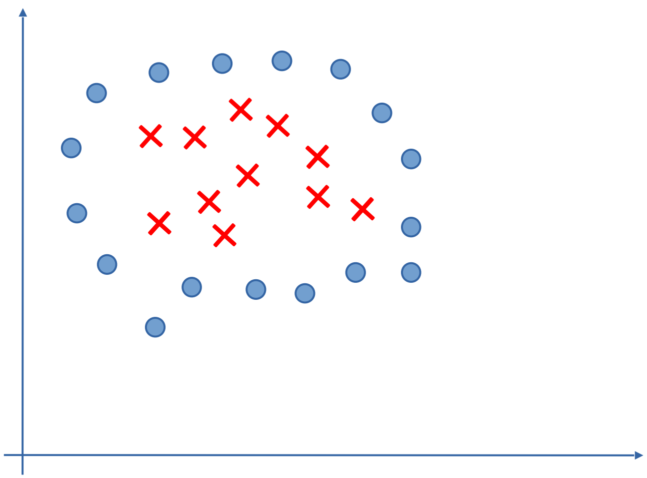
classification



linearly separable



non linearly separable



non linearly separable

classification

- logistic regression
- support vector machine
- decision tree
- random forest
- KNN

logistic regression

- perfect for linearly separable
- can be extended to multiclass

$$\text{logit}(P) = \log \frac{P}{1 - P}$$

logistic regression

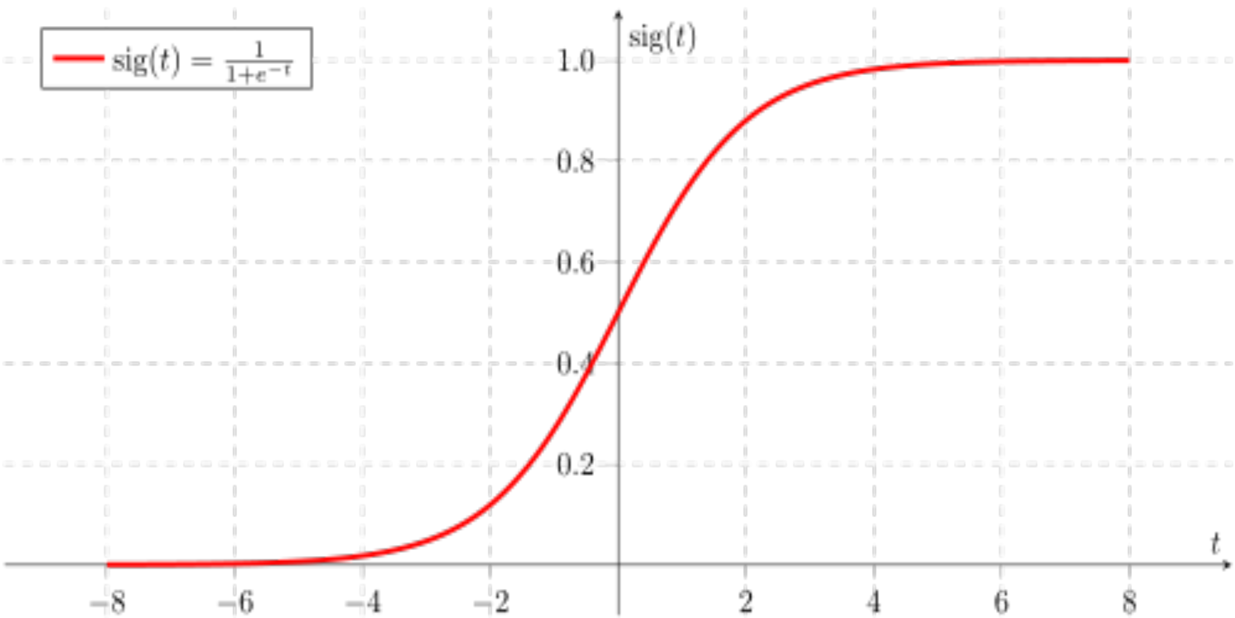
- the logit function takes input in [0,1] and returns in (-inf, +inf)
- express linear relationships between feature values and the log-odds

$$\text{logit}(P(y=1|x)) = \text{sum}(W_i X_i) = W^T X$$

- where is the conditional probability that a particular sample belongs to class 1 given its features x.

sigmoid function

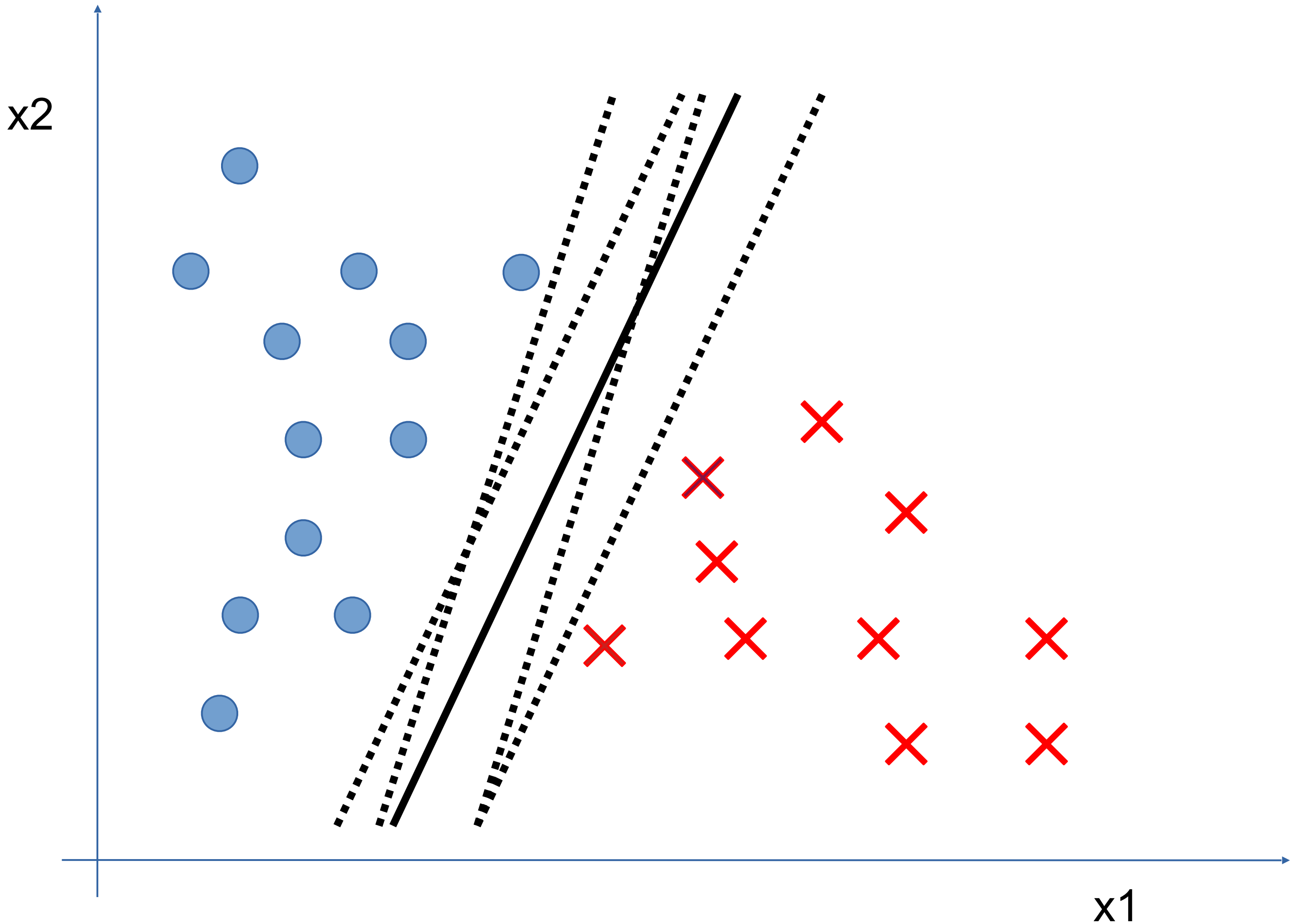
- the inverse of the logit function
- sigmoid(logit(p)) = p



sigmoid

- from (-inf, +inf) to [0,1]
- takes real values and transform them in the [0,1] range with an intercept at 0.5
- THIS IS WHAT THE logit function does while trained.
- the output of the sigmoid is the probability of a certain sample to be of class 1, given its feature **x** parametrised by the weights **w**

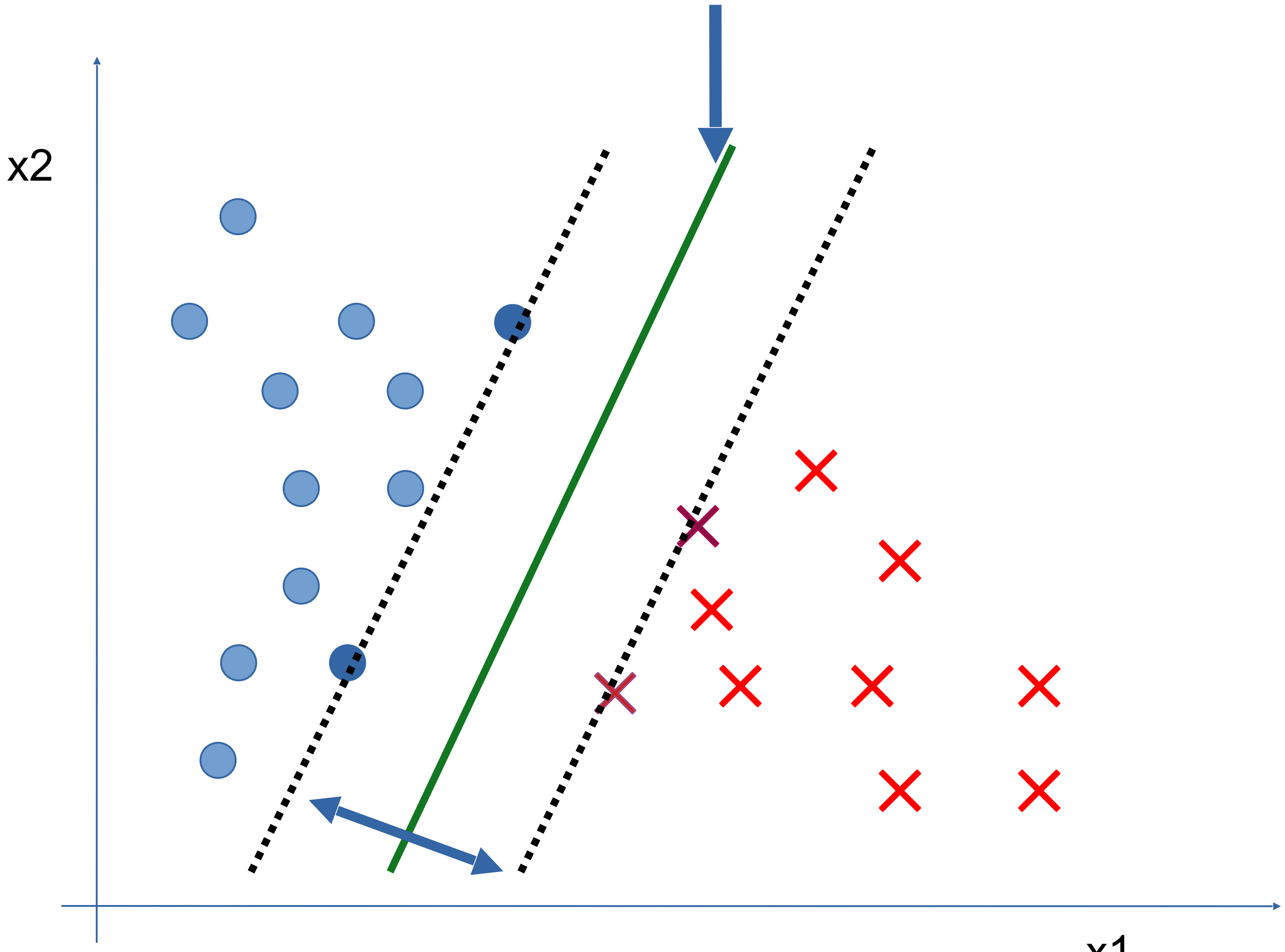
Support Vector Machine



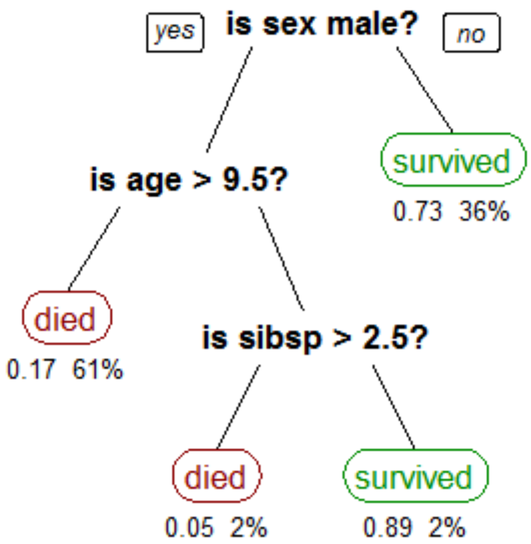
Support Vector Machine

- find a hyperplane in an N-dimensional space that distinctly classifies the data points.
- many possible hyperplanes that could be chosen.
- find a plane that has the maximum margin, i.e., the maximum distance between data points of both classes.

Support Vector Machine



Decision Tree



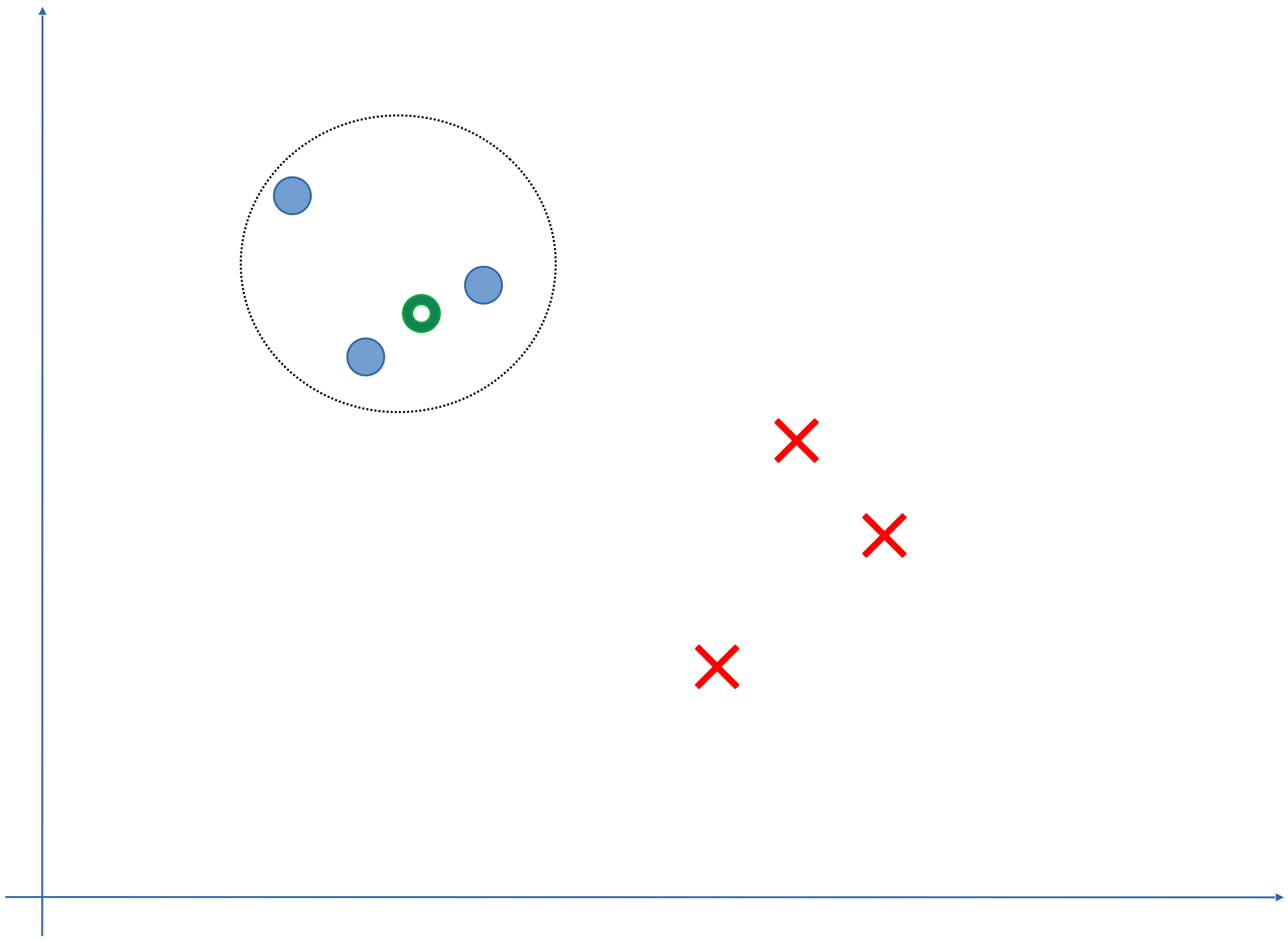
source: wikipedia

Decision Tree

- feature importance is KEY
- *n* features → *n* candidates splits
- calculate how much accuracy is lost for each split
- the split that costs least is chosen

- WHEN DO WE STOP???
  - max depth
  - min number of training inputs for each leaf
  - …

KNN



KNN

- Load the data
- Choose **K**
- For each point **p** in test data:
  - Compute distance between **p** and each training data
  - Sort in ascending order
  - Choose the top **K** rows
  - Assign the most frequent class
- Done.

unsupervised learning

unsupervised

- No labels given
- GOAL: find structure
  - discovering hidden patterns in data

unsupervised

- trickier
  - no answer labels (no ground truth)
  - external evaluation vs internal evaluation
    - experts vs objective function
- but:
  - annotating large datasets is very costly (Speech Recognition)
  - we don’t know how many classes can be (Data Mining)
  - gain some insight into the structure of the data before designing a classifier

clustering

- more problems:
  - define distance
  - define similarity
  - define clusters
- Examples:
  - Kmeans
  - Fuzzy Kmeans
  - GMM
  - Hierarchical
  - …

K-means

- Group input data into K groups
- Define K centers
- While “not converged”:
  - Take each point and assign it to the “closest” center
  - Recompute centers
    - minimize inter-cluster distances