Predictive Modelling

k-Nearest Neighbors (kNN)

Raquel Sebastião

Departamento de Electrónica, Telecomunicações, Informática
Universidade de Aveiro

raquel.sebastiao@ua.pt

2022/2023

k-Nearest Neighbors (kNN)

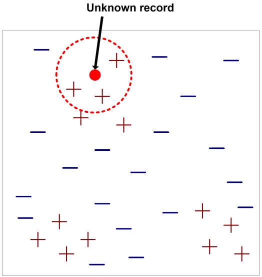
k-Nearest Neighbor belongs to the class of **instance-based** often known as ”lazy learner”

- No algorithm to extract information from labeled data.
- The labeled data is stored to classify new data.
- It does not learn a function to map the predictor variables into a target variable
- it does not make any assumption on the unknown functional form we are trying to approximate, it means that with sufficient data they are applicable to any problem

k-Nearest Neighbors (kNN)

The decision about label of new data

- looking for the **most similar examples (neighbors)** within the stored data
- the **label** is decided according to the **label of neighbors**



The hyper-parameters of the classifier are: *k* **and criterium** to find the neighbors.

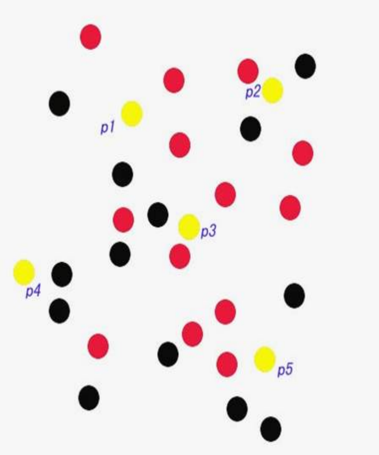
k-Nearest Neighbors (kNN)

Method:

- Choose the number **k** and the distance metric **d**
- For a test case **x**
  - find the **k** nearest cases in the training data according to **d**
  - use the target variable values of these cases to obtain the prediction for **x**
  - the prediction is the majority class

k-Nearest Neighbors (kNN): example

$2 - D$ data set belonging to two classes



● Class A

● Class B

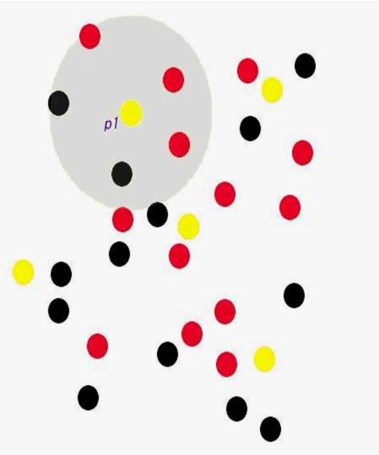
● New data

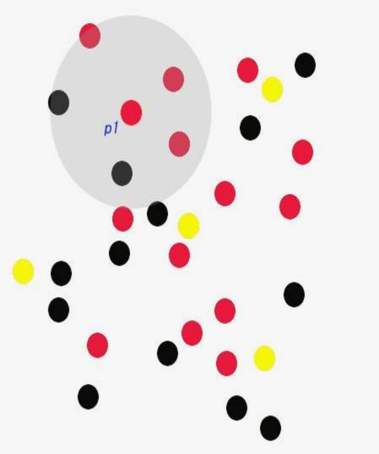
What is the label of **new (yellow) points** $\mathbf{p}_i, i = 1 \ldots 5$**?**

KNN: choose $K = 5$

k-Nearest Neighbors (kNN): example

Looking for neighbors of $\mathbf{p}_1$

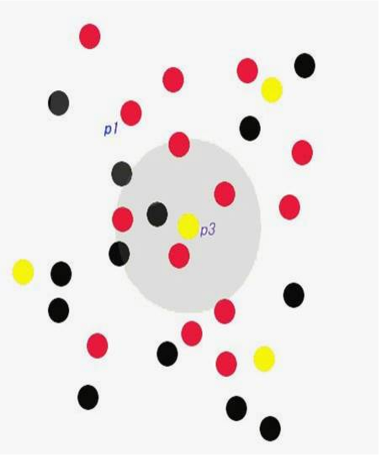


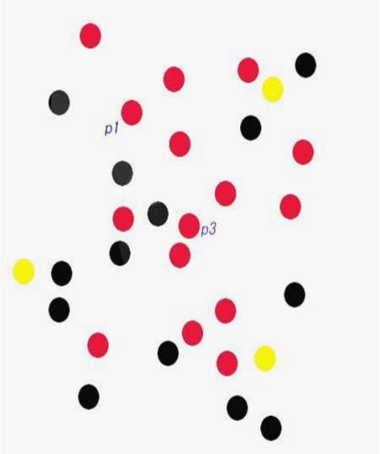


Within 5 neighbors: 3 are of class A then Class A

k-Nearest Neighbors (kNN): example

Looking for neighbors of $\mathbf{p}_3$





Within 5 neighbors: 3 are of class A then Class A

k-Nearest Neighbors (kNN): basic principles

- The **class membership** of a new object is estimated by **majority vote within nearest neighbors**.
  - Note $k = 1$, is the label of the **nearest neighbor.**
- The definition of neighborhood depends on a proper measure

Different measures to find neighbors

- **Euclidean distance**
- Manhattan distance
- Chebyshev distance
- Cossine distance
- ...

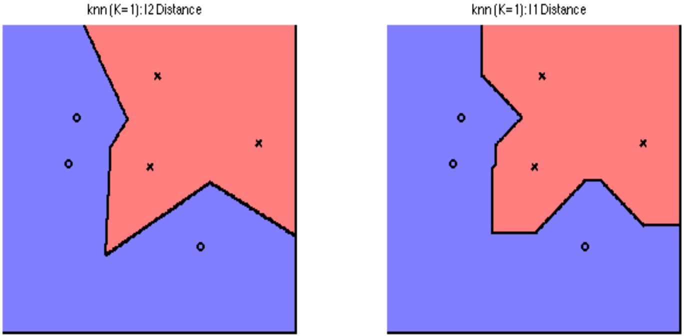
k-Nearest Neighbors (kNN): Choosing k

What should be the value of **k** ?

- typically, 3, 5 and 7
- odd numbers to avoid draws
- it can be estimated experimentally
  - **global** estimation searches for the ideal k for a given data set
  - **local** estimation methods try to estimate the ideal **k** for each test case (computationally very demanding!)

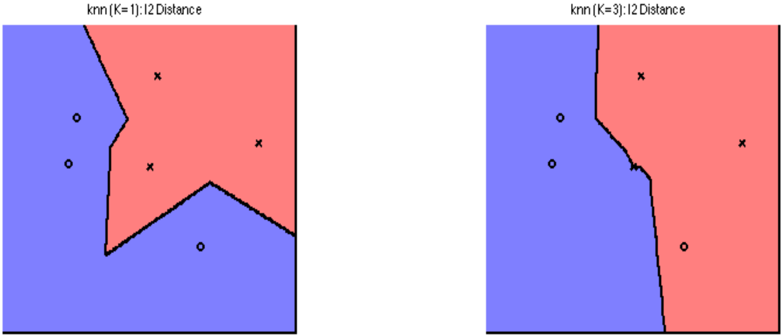
k-Nearest Neighbors (kNN): toy example

$K = 1$ and Euclidian distance versus Manhattan



k-Nearest Neighbors (kNN): toy example

Euclidian distance with $k = 1$ versus $k = 3$



k-Nearest Neighbors (kNN): Advantages

Algorithm provides a highly effective inference method for noisy training data

Algorithm is easy to interpret

New classes can be added without re-training

Different metrics provide flexibility

Works well for online learning as new data is constantly arriving

k-Nearest Neighbors (kNN): Disadvantages

Requires good choices!

- Results depend on choice of **k**
- Results depend on choice of metric, especially in high-dim spaces
  - normalization, irrelevant variables, unknown values, outliers may have a strong impact on the performance

”Training set” consumes much main memory

- Complexity grows linearly with the number of cases

Classification is time consuming

- Fast training time, but slow testing time

References

Introduction to Data Mining, Pang-Ning Tan, Michael Steinbach, Anuj Karpatne, Vipin Kumar, Pearson, 2019 (chap 6.3)

Data Mining, the Textbook, Charu C. Aggarwal, Springer, 2015 (chap 10.8)