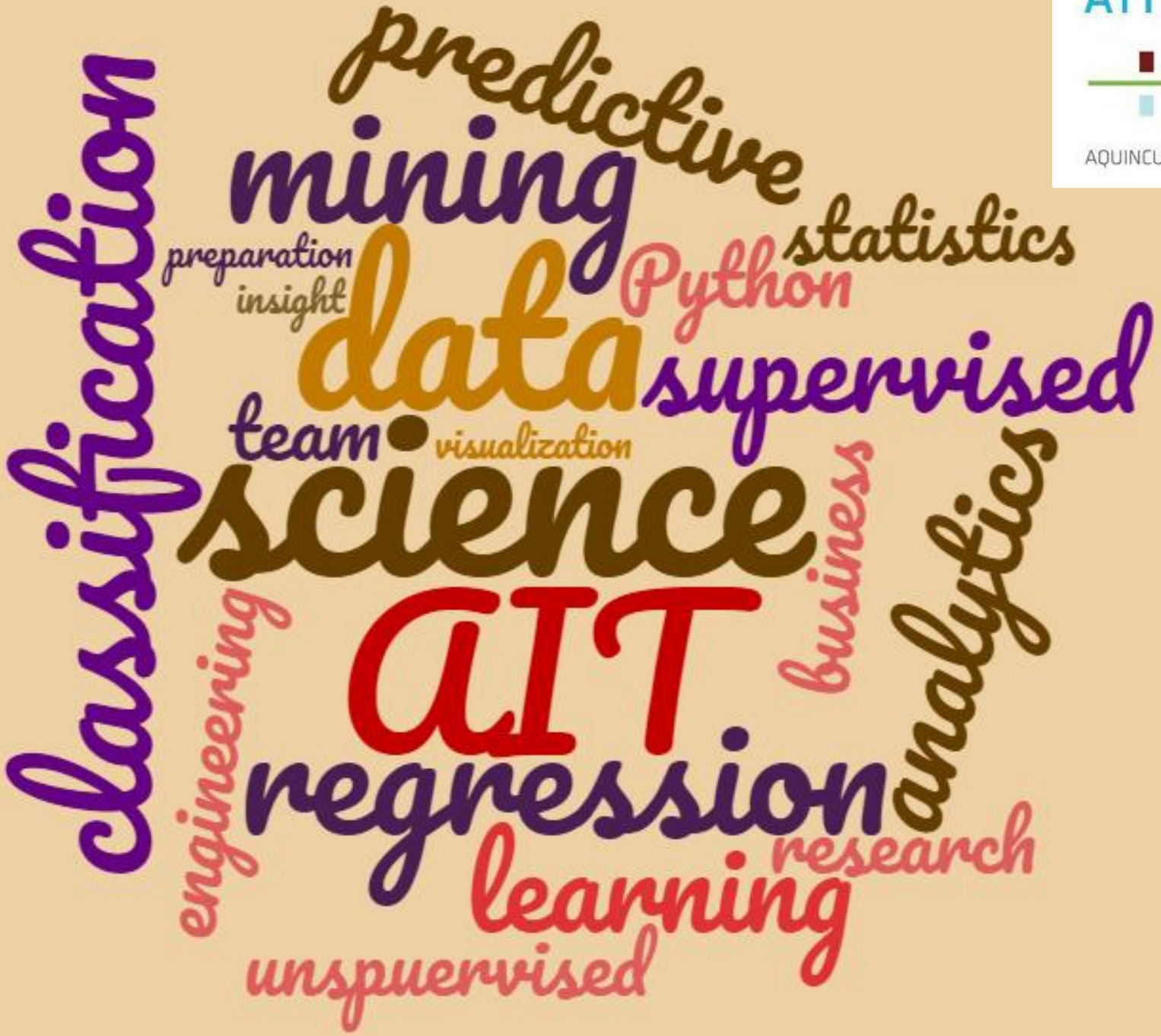
Data Science



February 24, 2023
kNN

Dr. Roland Molontay

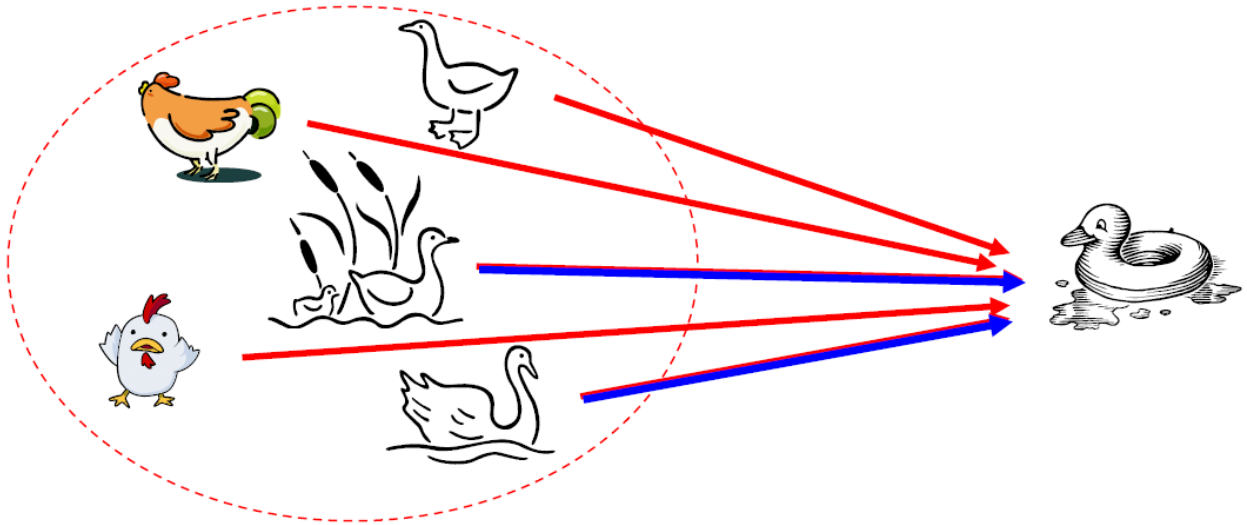
AIT-BUDAPEST
AQUINCUM INSTITUTE OF TECHNOLOGY

Schedule of the semester

	Monday midnight	*Tuesday class*	*Friday class*
W1 (02/06)			
W2 (02/13)		HW1 out	
W3 (02/20)			
W4 (02/27)	HW1 deadline + TEAMS	HW2 out	
W5 (03/06)	PROJECT PLAN		
W6 (03/13)	HW2 deadline	HW3 out	
W7 (03/20)			MIDTERM
SPRING BREAK		SPRING BREAK	SPRING BREAK
W8 (04/03)	HW3 deadline		GOOD FRIDAY
W9 (04/10)	MILESTONE 1	HW4 out	
W10 (04/17)			
W11 (04/24)	HW4 deadline		
W12 (05/01)	MILESTONE 2		
W13 (05/08)			
W14 (05/15)		FINAL	PROJECT presentations
W15 (05/22)		PROJECT presentations	

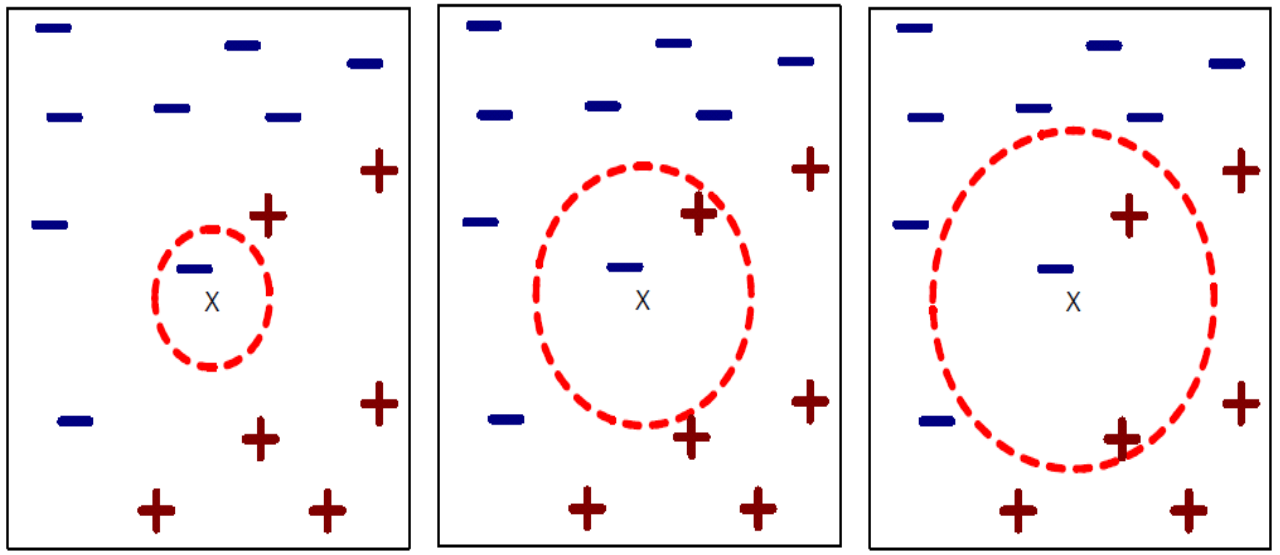
K Nearest Neighbors (kNN)

- kNN: k Nearest Neighbors
- Principle: „If it looks like a duck, swims like a duck, and quacks like a duck, then it probably *is* a duck.”
- It can be used for classification and regression as well



kNN approach

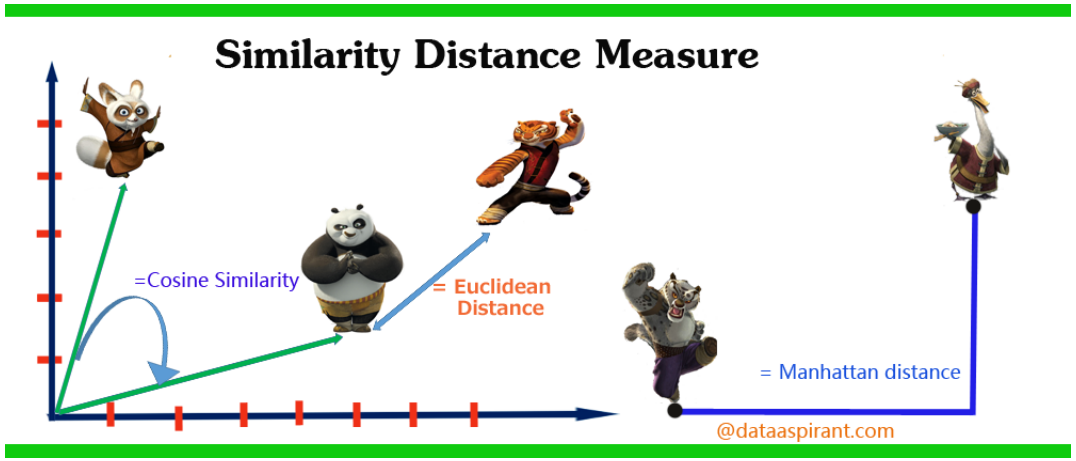
- Records: points in the *d*-dimensional space (where *d* is the number of attributes excluding the label)
- The label of a new record is determined by the labels of its *k* nearest neighbors in the training set
  - What similarity measure to use?
  - How to choose *k*?
  - How to decide on the target variable?



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

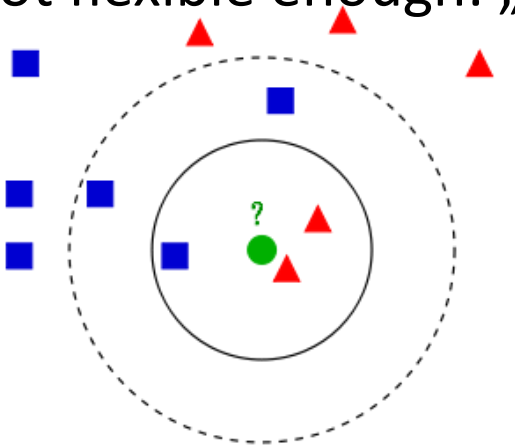
How to choose the right (dis)similarity measure?

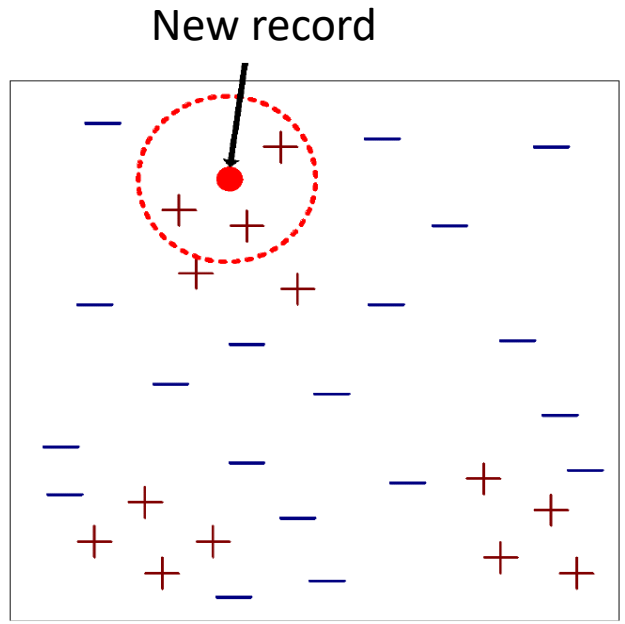
- Choose the (dis)similarity that is most suitable for the problem in hand
  - Sometimes we choose the suitable measure intuitively, ensuring that those objects are close according to the chosen measure that we think are similar indeed
  - We can try out more measures and test which has the best performance
  - For possible (dis)similarity measure, see Lecture 03



How to choose *k*?

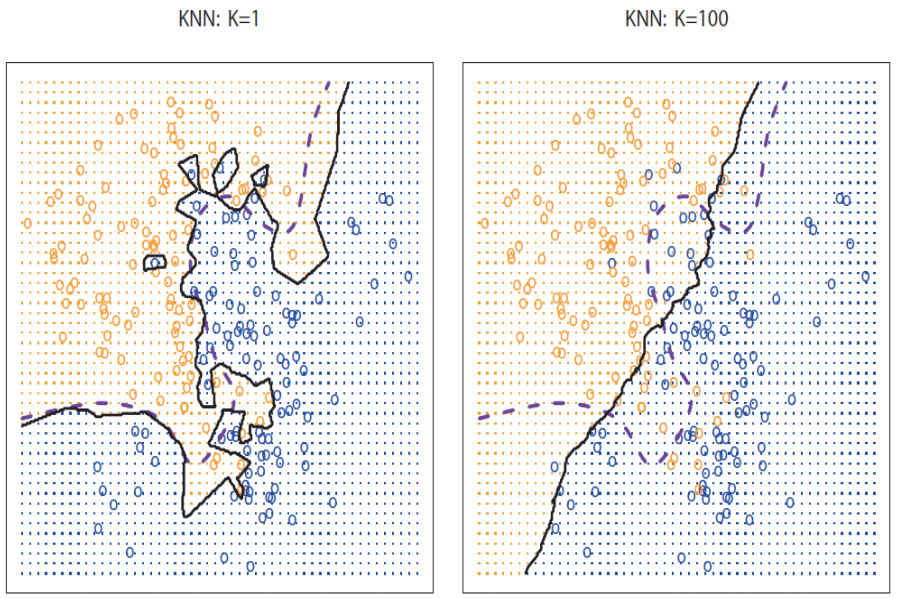
- If *k* is too small, it is too sensitive for noise, for local errors
  - It is sensitive to the training set itself: „high variance”
- If *k* is too large, then too dissimilar objects are also taken into consideration
  - The model is not flexible enough: „high bias”

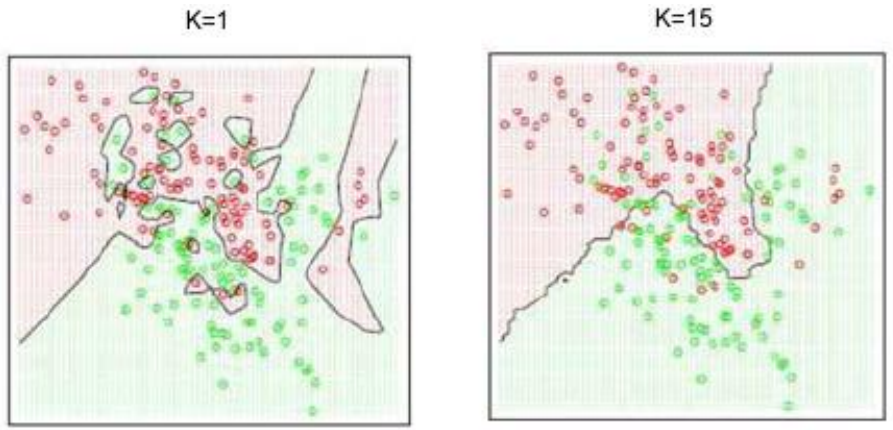




Effect of choosing *k*

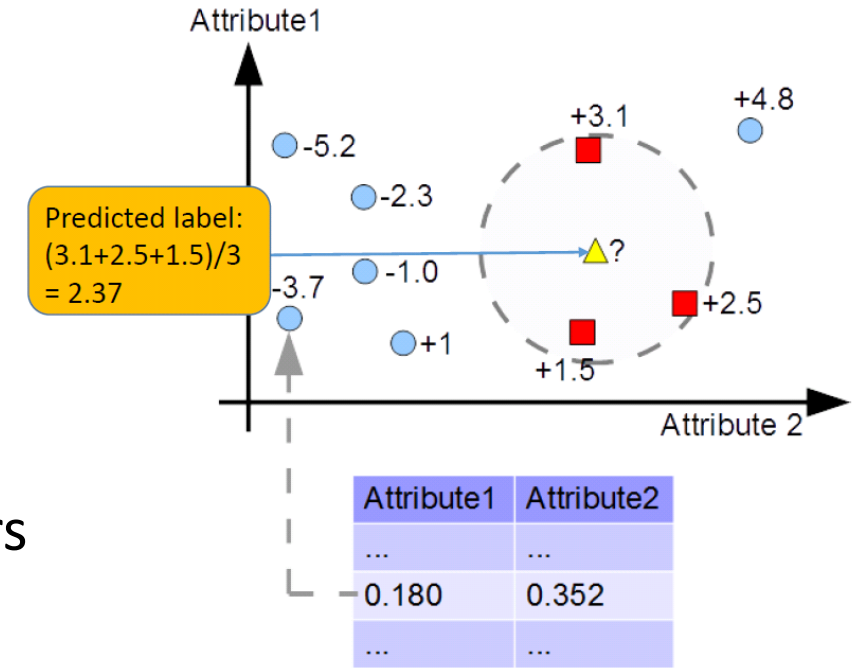
- The bigger the *k* value is the smoother the boundary is, the smaller the effect of noise is
- If *k* is too large, the objects that are far away also play a role
  - E.g. if *k=N*, we predict the majority class for every instance





How to decide on the output?

- For classification problems:
  - Majority voting among the labels of the *k* neighbors
    - For binary classification *k* should be odd
  - Weighted voting, a possible weight: $w_i = \frac{1}{d_i^2}$, where d_i is the distance of *i*th neighbor
- For regression problems:
  - Averaging the target variables of the *k* neighbors
  - Weighted averaging (similarly inversely proportional to the distance)



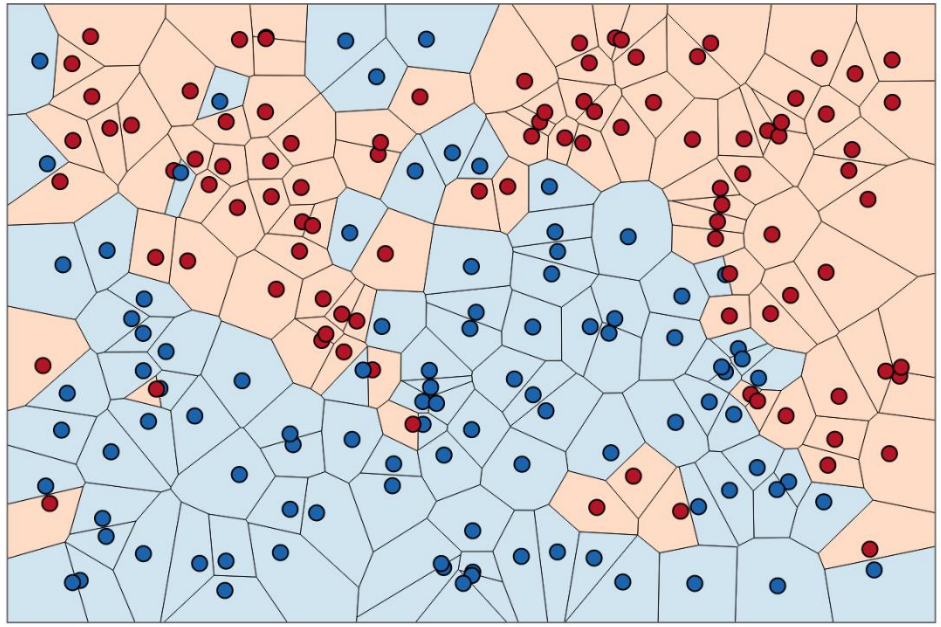
Advantages / disadvantages

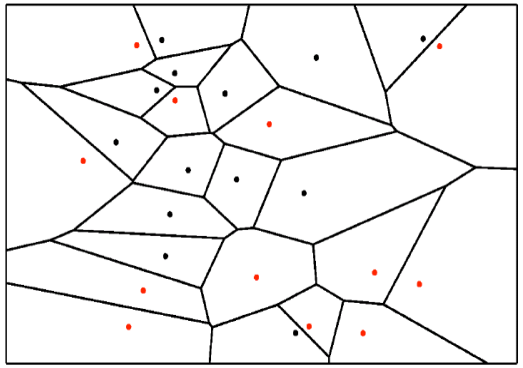
- Simple, easy to understand, easy to implement
- Widespread
- Theoretically efficient (for infinitely many training data points)
- „lazy learner”: it works only if a new unlabeled data point arrives
  - There is no explicitly built models, no long model building phase
- For *k=1* we can make the prediction very fast using some preparation
  - Voronoi diagram (Voronoi cells): Partitioning the space into regions based on distance to some seed points. A cell consists of all points closest to its seed. The seeds are the training points.

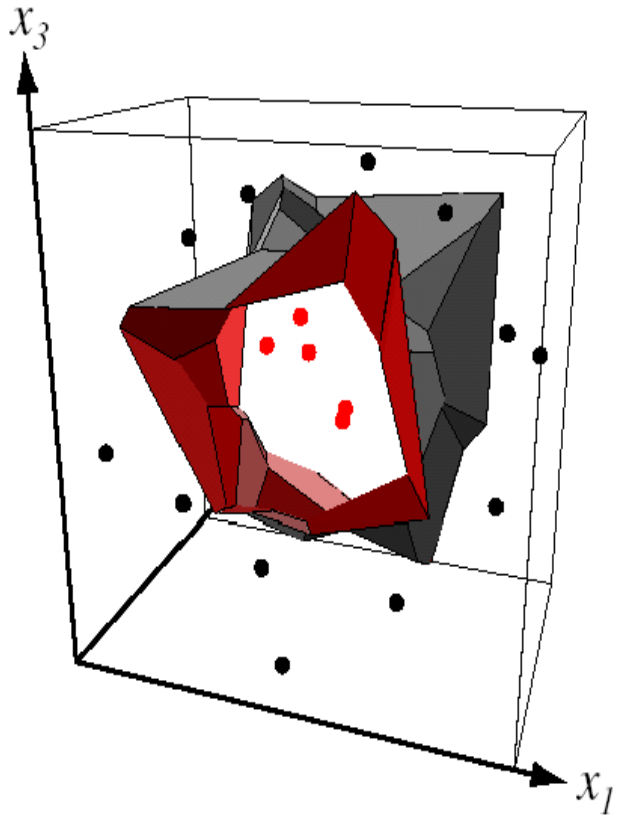
- Classifying one record is relatively slow for large training set
- For good performance an arbitrary point should have enough number of training points in its neighborhood
- For good performance, in increased dimension the number of training points should increase exponentially
- Sensitive for irrelevant and correlated attributes
  - Its sensitivity also depends on the chosen dissimilarity (distance) measure

Voronoi diagram (cells)

- Space partition by 1NN classifier





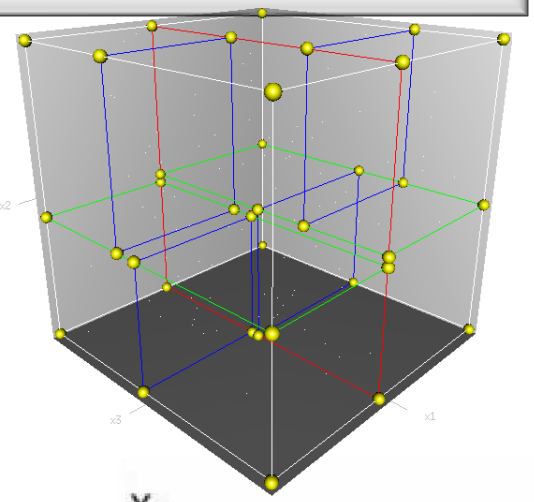


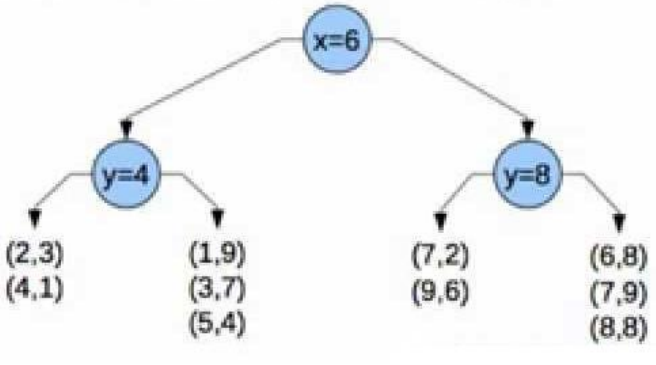
How to find the nearest neighbors?

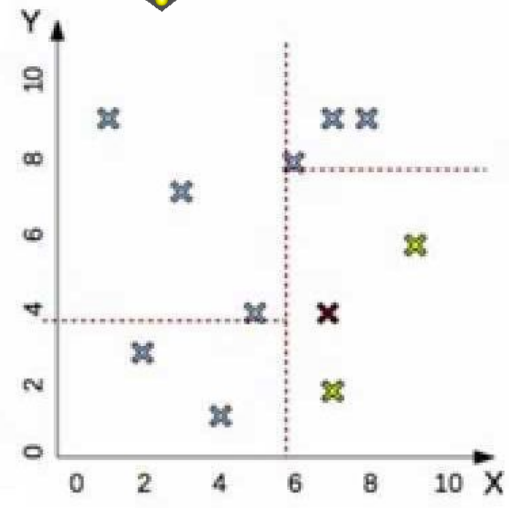
- Naive method: we calculate the distances between the record (that we aim to classify) and all the training points
  - Time complexity: *O(dN)* (*N*: number of training points, *d*: number of attributes, dimension)
- Using space-partitioning data structures for organizing points
  - E.g. using *k-d* tree
    - Average complexity of nearest neighbor search: *O(log N)*, in case of randomly distributed points and constant dimension
- There are also approximation algorithms
  - We can improve running time if we accept „good guesses” of the nearest neighbor. It doesn’t guarantee to return the actual nearest neighbor
    - For real-world problems, the approximation algorithms work (almost) as good as exact ones

K-d tree

- Build a *k-d* tree based on the training set
  - *{(1,9), (2,3), (3,7), (4,1), (5,4), (6,8), (7,2), (8,8), (7,9), (9,6)}*
  - Choose a random coordinate, calculate the median, divisive the data into two parts, repeat the procedure on both branches as long as we need
- Find the nearest neighbor of a new point: *(7,4)*
  - The point is consisted by which range?
  - Calculate the distance to the points in the range
  - We also check the points in the neighboring ranges







Classification error rate

- How can we measure the „goodness” of a classifier?
  - There are several methods (later)
  - The simplest method is the error rate
  - Error rate in the training set
    - Employ the model to the training points: $\{(x_1, y_1), \dots, (x_n, y_n)\}$
    - Error rate is the ratio of misclassified data points:

$$\frac{1}{n}\sum_{i=1}^{n} I(y_i \neq \hat{y}_i)$$

  - Error rate in the test set
    - We calculate the error rate for new data points that were not used for training
    - A good classifier has a small error rate on the test set

Bayes classifier

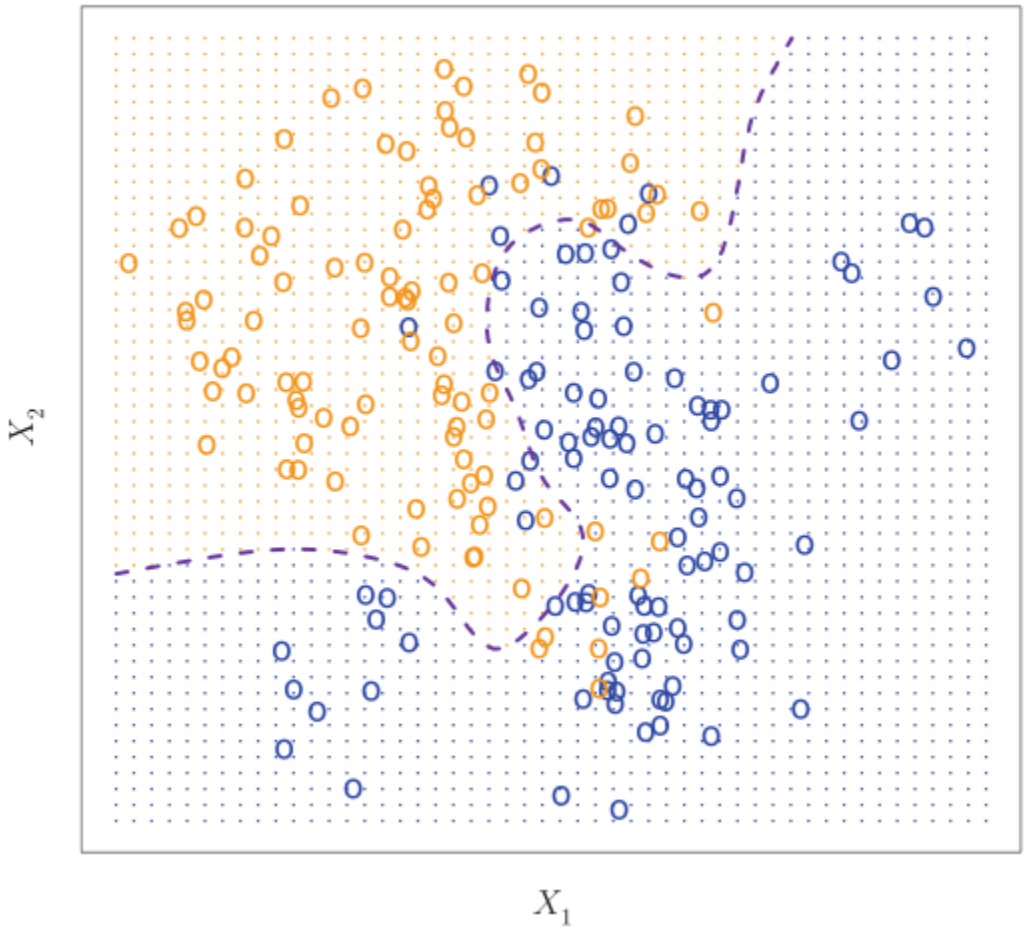
- The classifier that minimizes the test error rate is such a classifier that assigns the label to the observation with the highest probability given the attributes
- In other words: to an observation with x_0 feature-vector such a j class is assigned for which the following conditional probability is maximal:

$$P(Y = j|X = x_0)$$

  - For binary classification, label 1 is assigned if: $P(Y = 1|X = x_0) > 0.5$
- The conditional probabilities are not known, unless the data is generated from a given $p(X, Y)$ joint distribution (background distribution)

Bayes decision boundary

- For binary classification problems: it is the region of the attribute space in which the conditional probabilities are equal (0.5 – 0.5)
  - The classifier will classify all the points on one side of the decision boundary as belonging to one class and all those on the other side as belonging to the other class
- In the figure it is the purple dashed line



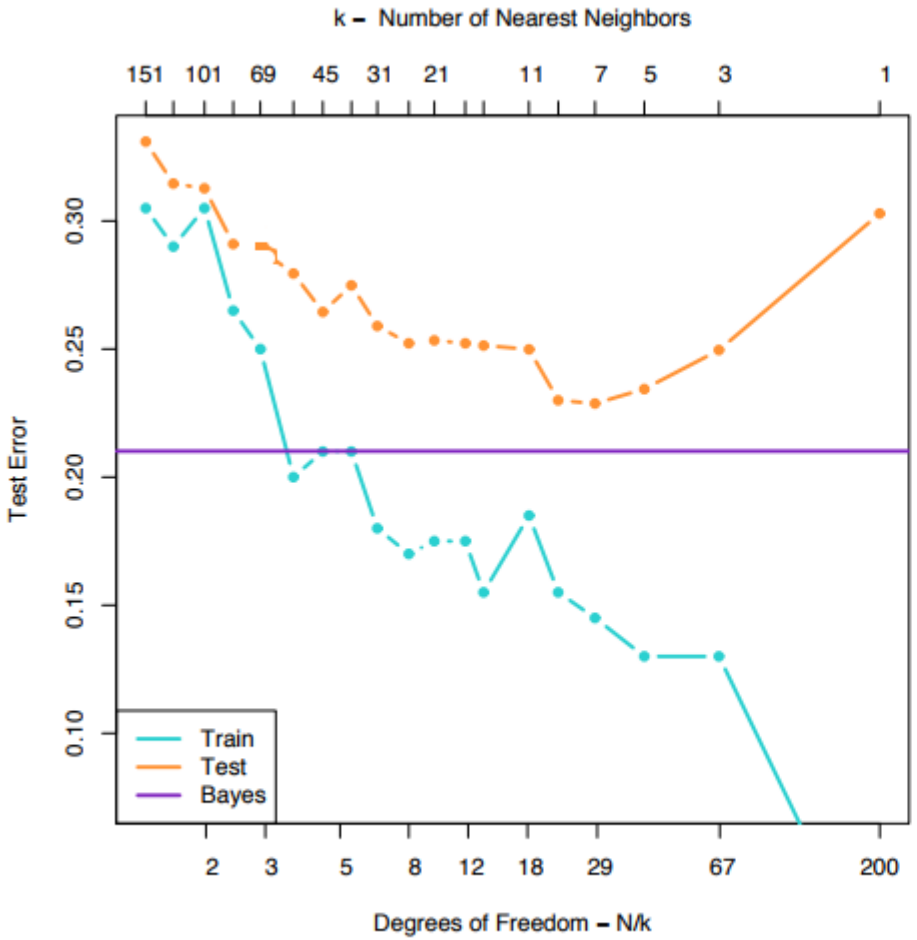
Bayes error rate

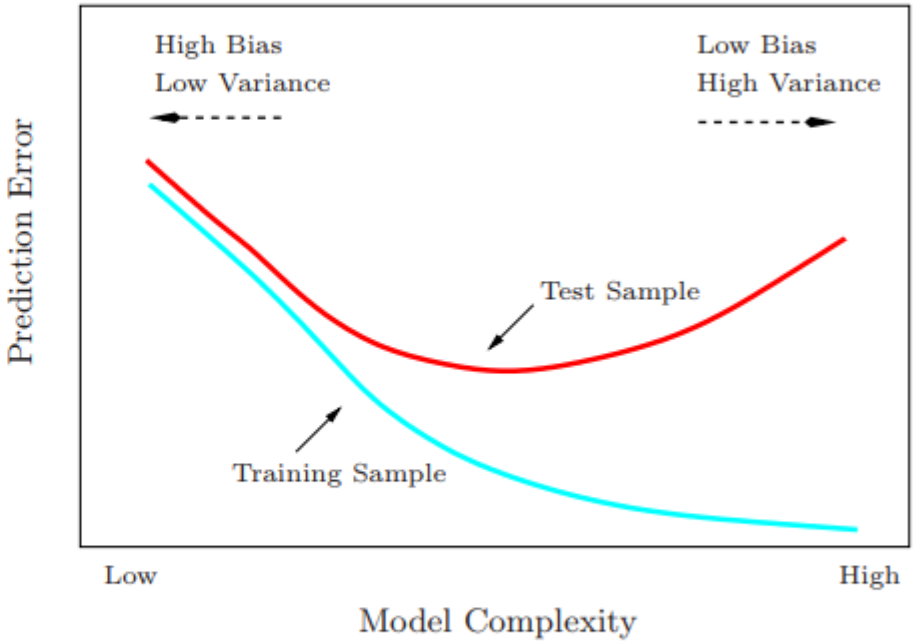
- The error rate of the Bayes classifier
- The lowest possible error rate for any classifier on the test set

$$1 - \mathbb{E}(\max_{j} P(Y = j|X))$$

  - Expectation is taken with respect to all possible values of *X*

Error rates for *k*NN algorithm





Acknowledgement

- András Benczúr, Róbert Pálovics, SZTAKI-AIT, DM1-2
- Krisztián Buza, MTA-BME, VISZJV68
- Bálint Daróczy, SZTAKI-BME, VISZAMA01
- Judit Csima, BME, VISZM185
- Gábor Horváth, Péter Antal, BME, VIMMD294, VIMIA313
- Lukács András, ELTE, MM1C1AB6E
- Tim Kraska, Brown University, CS195
- Dan Potter, Carsten Binnig, Eli Upfal, Brown University, CS1951A
- Erik Sudderth, Brown University, CS142
- Joe Blitzstein, Hanspeter Pfister, Verena Kaynig-Fittkau, Harvard University, CS109
- Rajan Patel, Stanford University, STAT202
- Andrew Ng, John Duchi, Stanford University, CS229