Machine Learning, Lecture 2: k-nearest neighbours

S. Nõmm

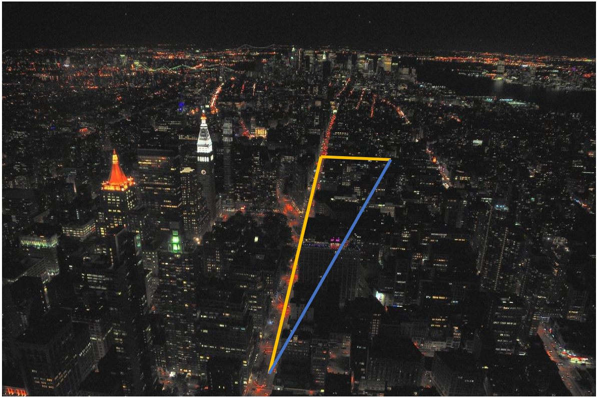
[1]Department of Computer Science, Tallinn University of Technology

07.02.2017

Distance and/or Similarity

Let x and y are two elements (objects). Define measure of distance/similarity between x and y

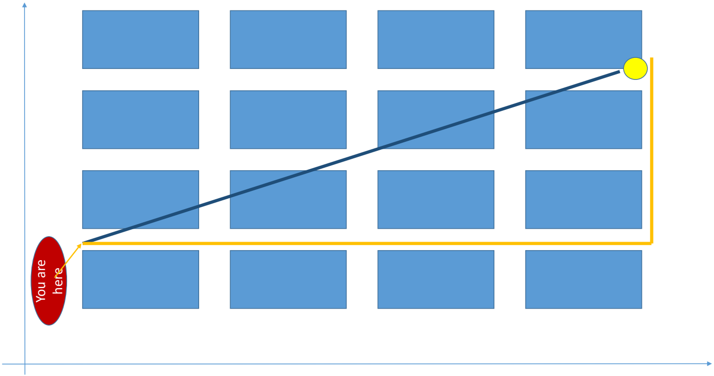
Distance ?



This is the distance used to compute the price of a taxi ride

Actual distance between the starting end ending points of your journey

Distance ?



Metric (some times referred as distance function)

Definition

A function $d : X \times X \to \mathbb{R}$ is called metric if for any elements x, y and z of X the following conditions are satisfied.

1. Non-negativity or separation axiom

$$d(x, y) \geq 0$$

2. Identity of indiscernibles, or coincidence axiom

$$d(x, y) = 0 \Leftrightarrow x = y$$

3. Symmetry

$$d(x, y) = d(y, x)$$

4. Subadditivity or triangle inequality)

$$d(x, z) \leq d(x, y) + d(y, z)$$

Examples: distances in the Euclidean space 1

Do you remember what Euclidean space is?

- Euclidean distance

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

- Manhattan distance also referred as city block distance or taxicab distance

$$d(x,y) = \sum_{i=1}^{n} \mid x_i - y_i \mid$$

- Chebyshev distance

$$d(x,y) = \lim_{k\to\infty}\Big(\sum_{i=1}^{n} \mid x_i - y_i \mid^k\Big)^{\frac{1}{k}} = \max_i\big(\mid x_i - y_i \mid\big)$$

Examples 2

- Mahalanobis distance

$$S(x, y) = \sqrt{(x-y)^T C^{-1} (x-y)}$$

  where C is the covariance matrix. Takes into account impact of data distribution.

- Cosine distance Cosine similarity is the measure of the angle between two vectors

$$S_c(x, y) = \frac{x \cdot y}{\|x\| \|y\|}$$

  Usually used in high dimensional positive spaces, ranges from -1 to 1. Cosine distance is defined as follows

$$S_C(x, y) = 1 - S_c(x, y)$$

L_p norms

- The real valued function f defined in a vector space V over the subfield F is called a norm if for any $a \in F$ and all $u, v \in V$ it satisfies following three conditions
  - $f(av) = \mid a \mid f(v)$
  - $f(u + v) \leq f(u) + f(v)$
  - $f(v) = 0 \Rightarrow v = 0$
- L_p is defined as follows

$$S(\bar{X}\bar{Y}) = \left(\sum_{i=1}^{d} \mid x_i - y_i \mid^p \right)^{\frac{1}{p}}$$

- In case of $p = 1$ we are dealing with already known to you Manhattan distance. In case of $p = 2$ Euclidean.

Examples 3: Distances between strings

- Levenshtein or SED distance. SED - minimal number of single -charter edits required to change one string into another. Edit operations are as follows:
  - insertions
  - deletions
  - substitutions
- SED(delta, delata)$=1$ delete "a" or SED(kitten,sitting)$=3$: substitute "k" with "s",substitute "e" with "i", insert "g".
- Hamming distance Similar to Levenshtein but with substitution operation only. Frequently used with categorical and binary data.

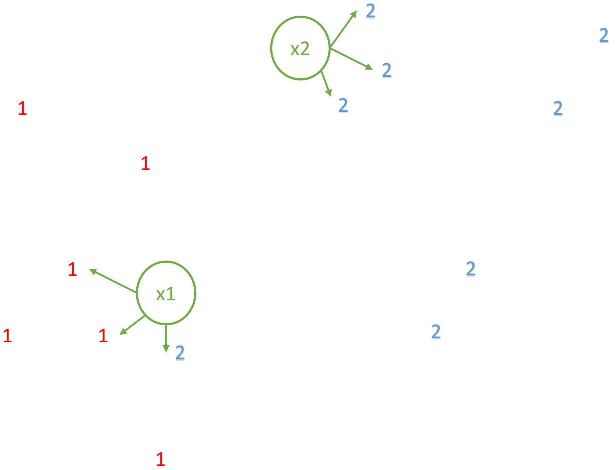
k-nearest neighbour (k-NN) classification

- Let N be a labeled set of points belonging to c different classes such that

$$\sum_{i=1}^{c} N_i = N$$

- Classification of a given point x
  - Find k - nearest points to the point x.
  - Assign x the majority label of neighbouring (k-nearest) points

Example



(k-NN) classification

- k-NN is a supervised learning method
- it is nonparametric learning method (number of the parameters grows with the amount of data)
- k-NN is a memory (or instance) -based learning, (algorithm memorizes the training data).
- k is the hyperparameter.

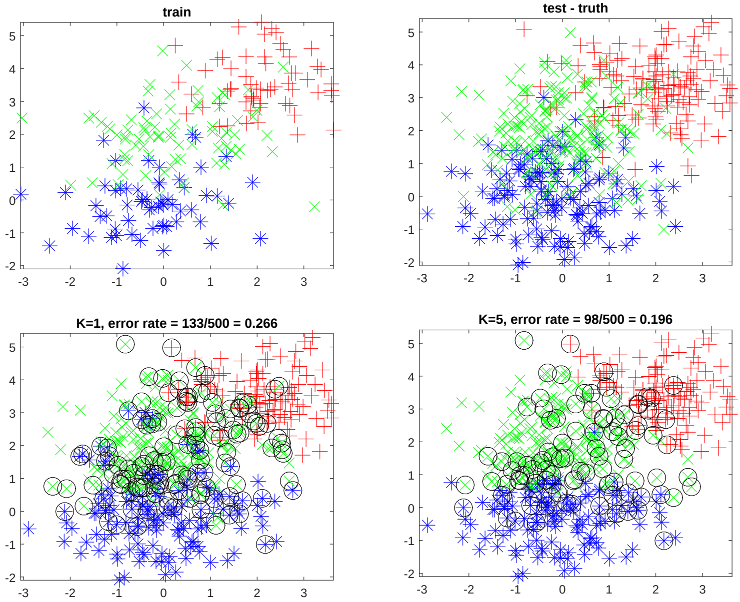
(k-NN) classification

- For an arbitrary point x the probability to belong to the class c is given by

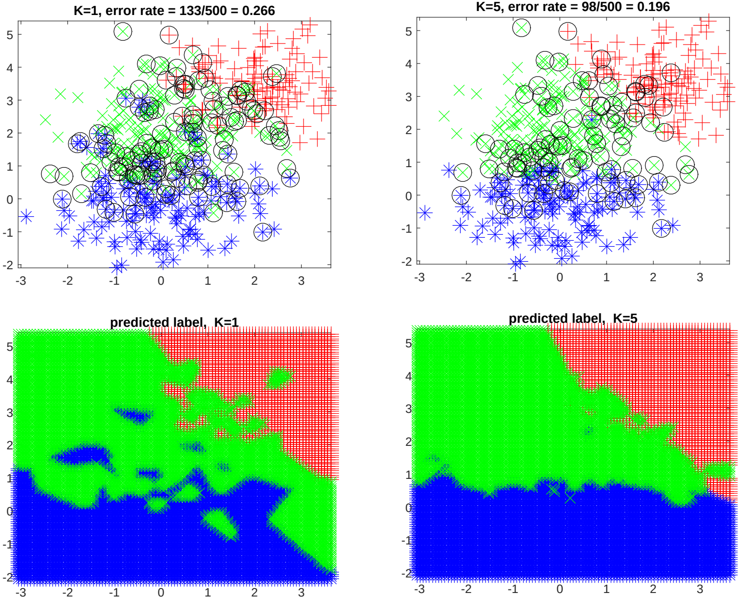
$$p(y = c \mid x, \mathcal{D}, k) = \frac{1}{k} \sum_{i \in N_{k(x,\mathcal{D})}} \mathbb{I}(y_i = c)$$

here $N_{k(x,\mathcal{D})}$ denotes the indexes of the k nearest points to x in $\mathcal{D}$

Example


train
test - truth
K=1, error rate = 133/500 = 0.266
K=5, error rate = 98/500 = 0.196
-3
-2
-1
0
1
2
3
4
5

Example


K=1, error rate = 133/500 = 0.266
K=5, error rate = 98/500 = 0.196
predicted label, K=1
predicted label, K=5
-3 -2 -1 0 1 2 3
-2 -1 0 1 2 3 4 5

Decision boundary

- Decision boundary or decision surface (the lines between different colors on the previous slide) is a "hypersurface" that partition the vector space in accordance to two classes it separates.
- Not necessarly surface in the strict sense of this word.
- Decision boundaries characterize the complexity of the model
  - Decision boundary is too "complex" - overfitting.
  - Decision boundary is too "smooth" - underfitting.
- the value k is used to control the complexity of the decision boundary
- Cross-validation may be used to select value k

Data normalization

Normalization - is the process of adjusting values measured on different scales to a common scale. There are different ways to normalize the data:

- Standard score Works well for normally distributed data. For each dimension j compute

$$x'_{i,j} = \frac{x_{i,j-\bar{\mu}_j}}{\sigma_j}.$$

- Feature scaling used to bring all values into the range $[0, 1]$.

$$x' = \frac{x - min(x)}{max(x) - min(x)}$$

  may be generalized to bring the values in to and closed interval $[a, b]$

$$x' = a + \frac{\big(x - min(x)\big)(b - a)}{max(x) - min(x)}$$

Note x' denotes normalization, not to be confused with derivative.

Impact of High Dimensionality (Curse of Dimensionality)

Curse of dimensionality - term introduced by Richard Bellman. Referred to the phenomenon of efficiency loss by distance based data-mining methods. Let us consider the following example.

- Consider the unit cube in d - dimensional space, with one corner at the origin.
- What is the Manhattan distance from the arbitrary chosen point inside the cube to the origin?

$$S(\bar{0}, \bar{Y}) = \sum_{i=1}^{d} (Y_i - 0)$$

Note that Y_i is random variable in $[0, 1]$

- The result is random variable with a mean $\mu = d/2$ and standard deviation $\sigma = \sqrt{d/12}$
- The ratio of the variation in the distances to the mean value is referred as *contrast*

$$G(d) = \frac{S_{max} - S_{min}}{\mu} = \sqrt{\frac{12}{d}}$$