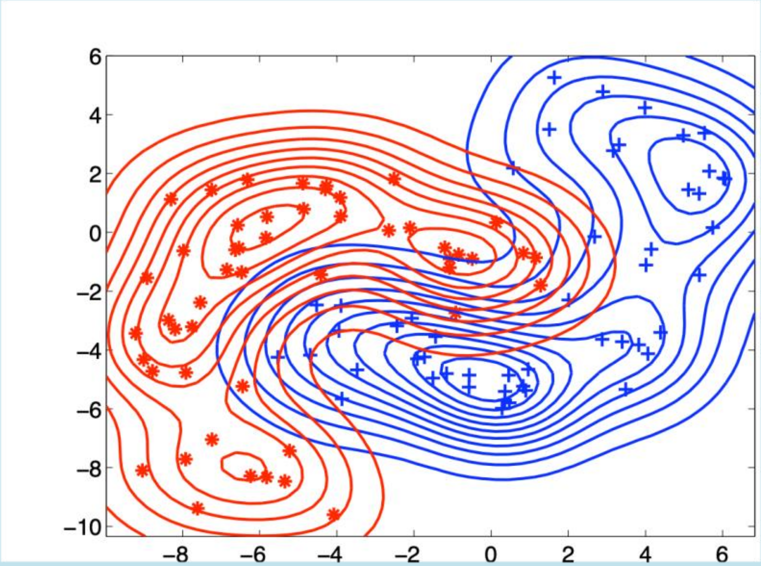
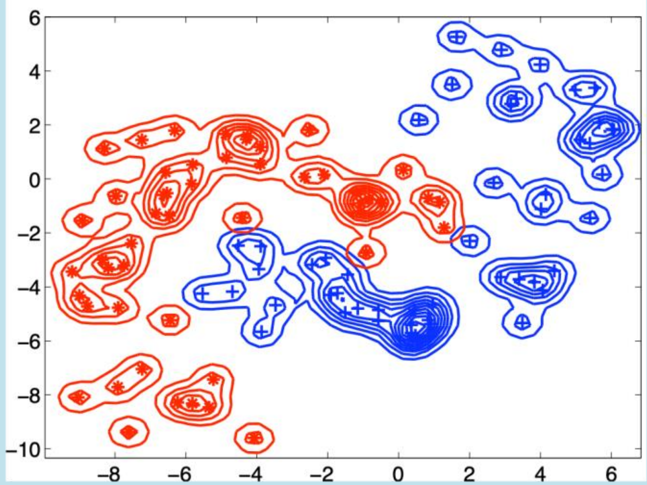
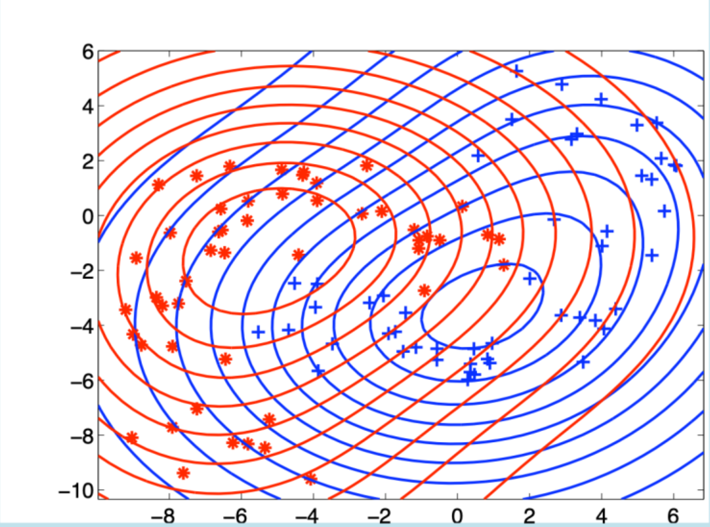
Non-parametric density estimation

Gosia Migut







After practicing with the concepts of this lecture you should be able to:

- Explain the difference between parametric and non-parametric density estimation
- Explain Parzen/Kernel, k-Nearest Neighbour and Naïve Bayes density estimation and classification in detail.
- Explain the advantages and disadvantages of those methods.
- Implement k-nn classifier in Python

Literature

- Chapter 2 section 2.5 from:

  Bishop (2006). Pattern Recognition and Machine Learning. Springer, UK.

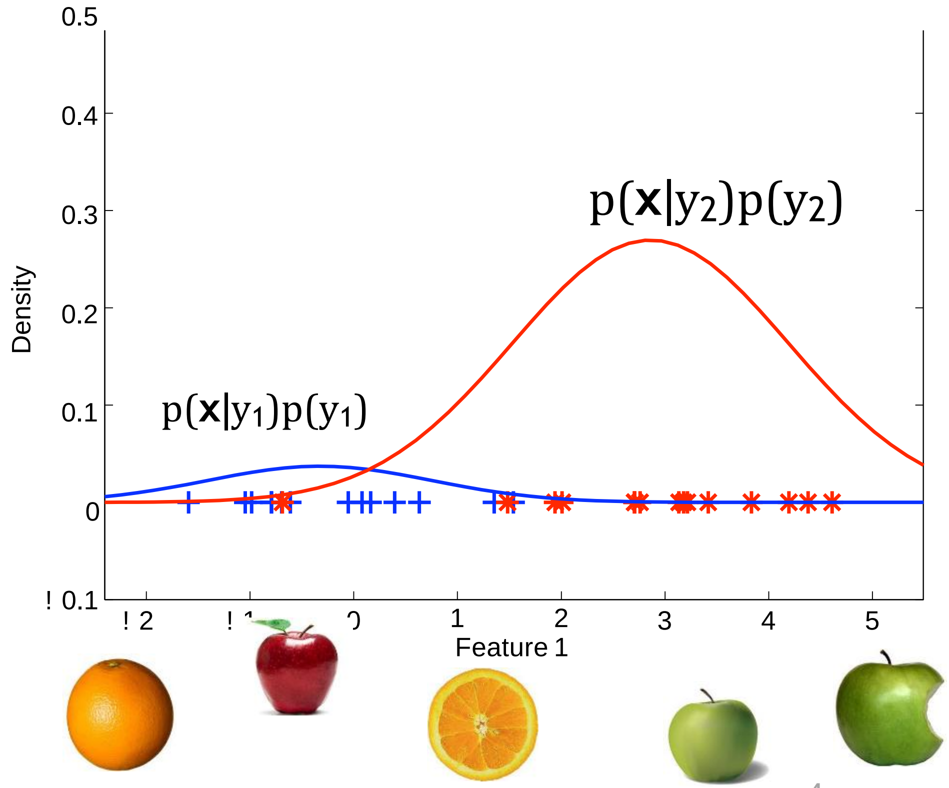
- Lecture notes CS229: section 2 and 2.1 (excluding 2.2). Andrew Ng, Standford University. http://cs229.stanford.edu/notes/cs229-notes2.pdf

Last week: parametric density estimation

- Known distribution, eg.: assume a single Gaussian distribution for each of the classes:
  - $\hat{p}(x|y_i) = N(x|\mu_i, \Sigma_i)$

- Estimate the **global** parameters on training set, eg.:
  - estimate μ_i and Σ_i for each of the classes

- For classification use Bayes rule
  - $p(x|y_1)p(y_1) > p(x|y_2)p(y_2)$

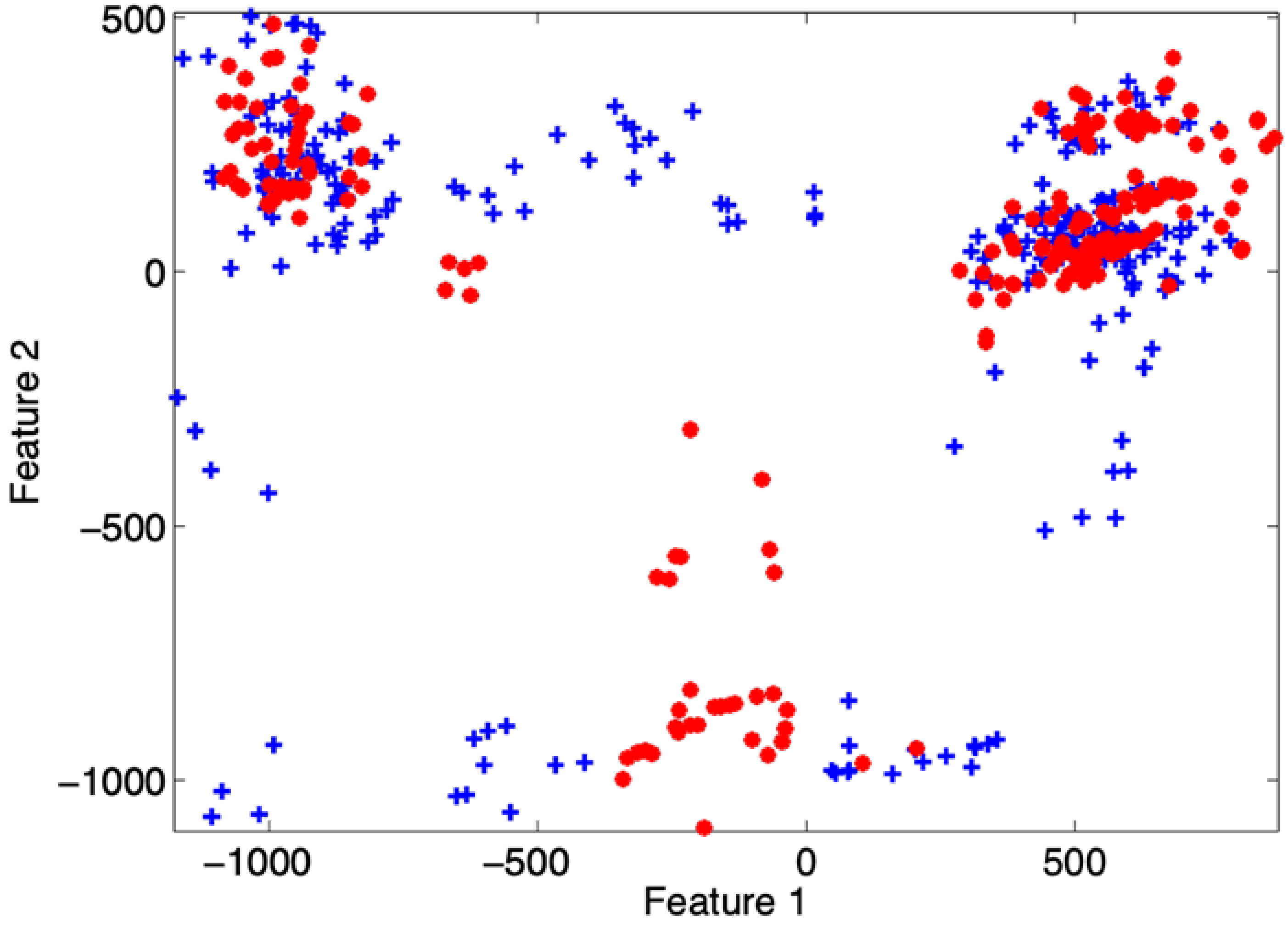


TUDelft

The real life…

Q: Which distribution to assume?

A: We don't know the distribution of data = no global parameters to estimate

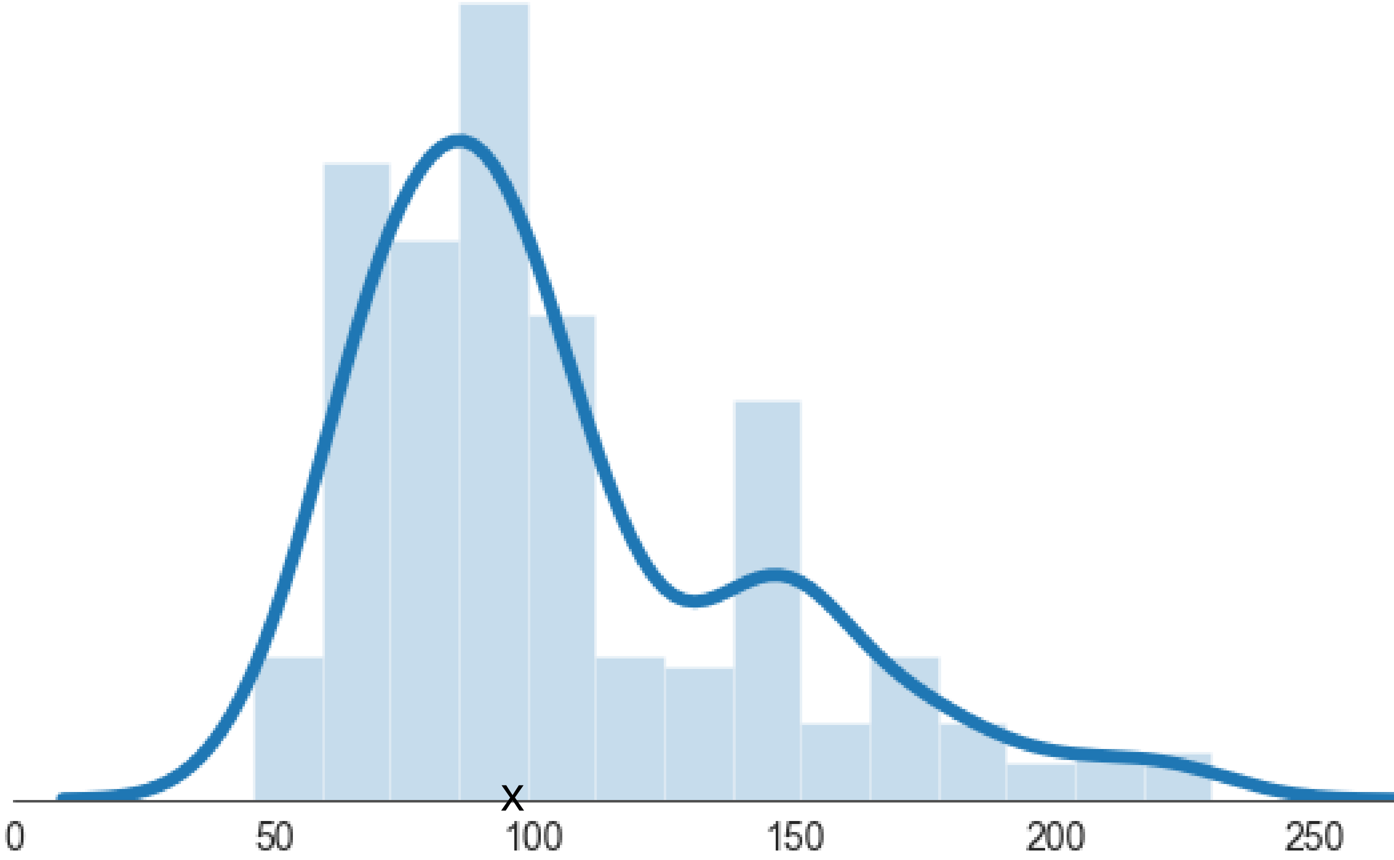


TU Delft

Simple non-parametric density estimation

- Example: we have one feature and N samples
- How to estimate the probability density?
  - Histogram
- Split the feature in subregions (bins) of width h
- Count the number of objects in each bin: k_B
- Probability density estimate at point x:

$$\hat{p}(x) = \frac{1}{h}\frac{k_B}{N}$$



Can we do better than histogram?

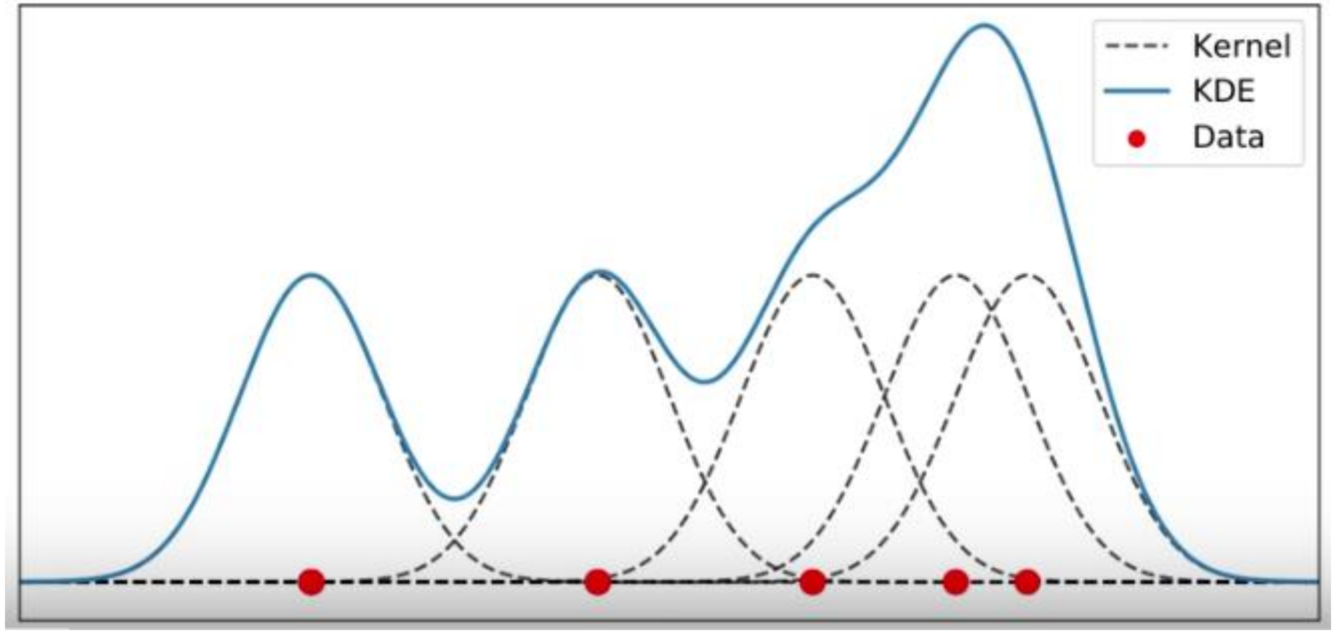
- Histogram puts all samples between boundaries of each bin.
- Bins location is arbitrary (no unique solution).

- In practice, two very related methods are used:
  - **Parzen (kernel) density estimate**
  - k-Nearest-neighbor density estimate (next lecture: theory and lab)

Parzen (window) Density Estimation

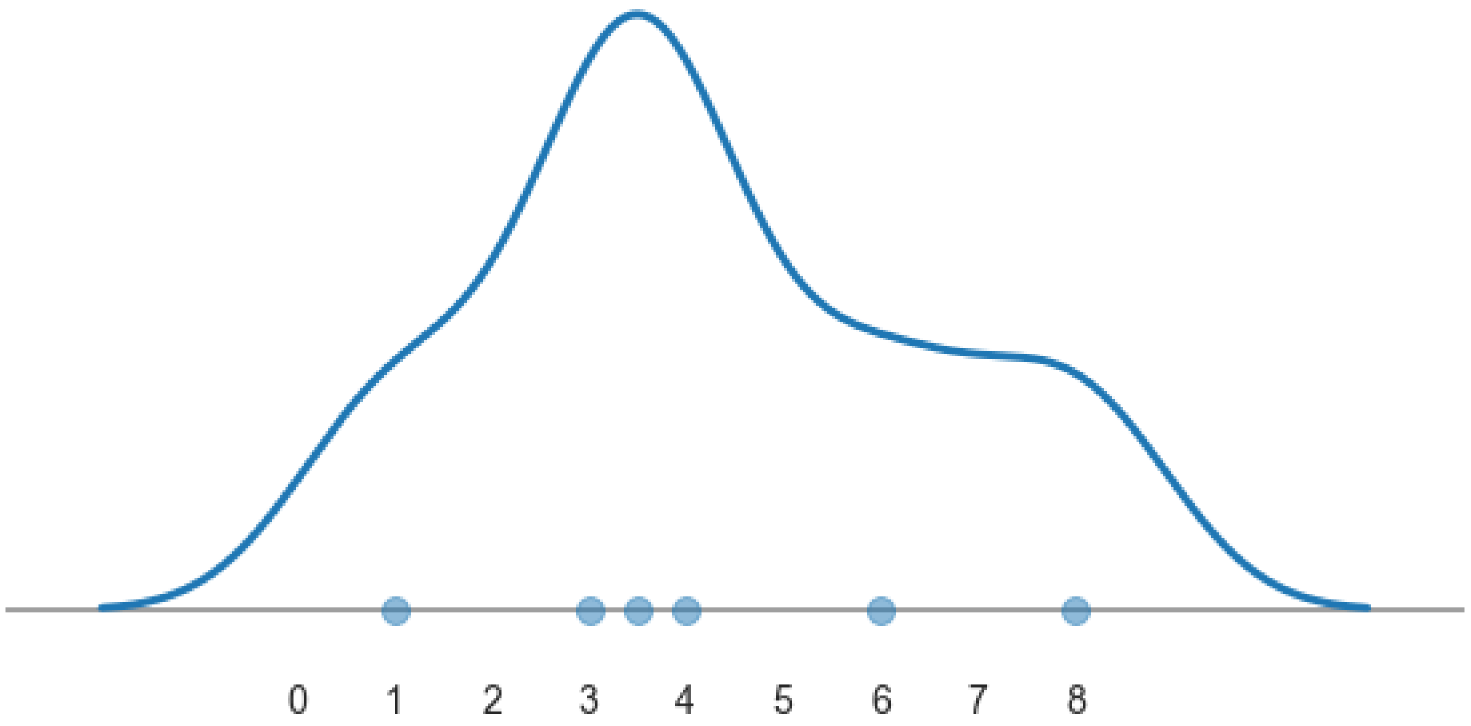
or

Kernel Density Estimation (KDE)



TU Delft

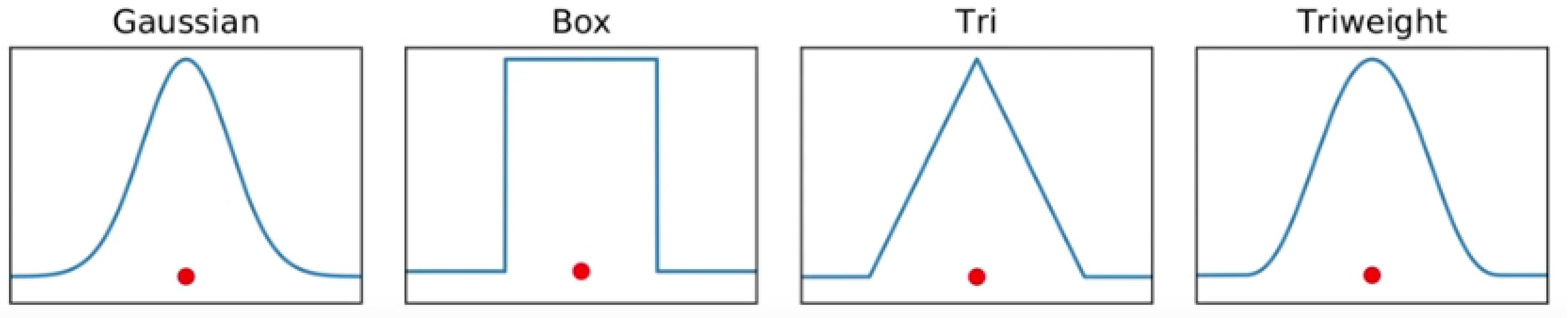
Parzen density estimation: intuition



Q: Do you know this type of distribution?

Parzen density estimation: intuition

- Define cell shape (kernel/window function), eg. Gaussian
- Fix size of kernel function (h), eg. $\sigma^2 = 1$



- Q: Is the KDE with a box kernel the same as a histogram?

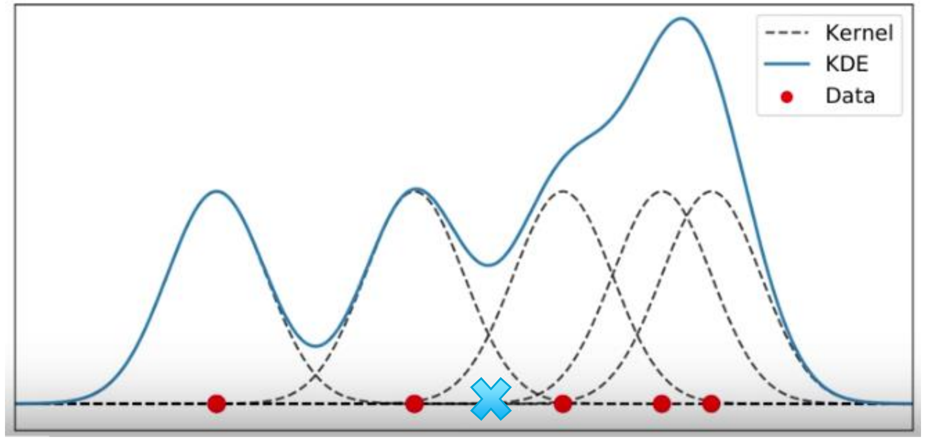
Lets do some drawing

- Draw histogram with bin size 2
- Draw KDE with box kernel with h=2

Lets do some drawing

- Draw histogram with bin size 2
- Draw KDE with box kernel with h=2

How to find Parzen probability density function estimate at x?



$$\hat{p}(x) = \frac{1}{nh}\sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

TU Delft

Question

Given a set of four data points:

$$\boldsymbol{x_1 = 2, x_2 = 2.5, x_3 = 3.5, x_4 = 0.5}$$

find Parzen probability density function (pdf) estimate at ***x*=3** using the kernel function with width ***h*=1**:

$$\boldsymbol{K(x) = \begin{cases} 0.5 & if\ |x| < 1 \\ 0 & otherwise \end{cases}}$$

Parzen pfd:

$$\hat{p}(x) = \frac{1}{nh} \sum_{i=1}^{n} K(\frac{x - x_i}{h})$$

Solution

- $x_1 = 2, x_2 = 2.5, x_3 = 3.5, x_4 = 0.5$
- $x = 3$ and $\mathrm{h} = 1$
- $K(x) = \begin{cases} 0.5 & if\ |x| < 1 \\ 0 & otherwise \end{cases}$

- $\hat{p}(x) = \frac{1}{nh}\sum_{i=1}^{n} K(\frac{x-x_i}{h}) = \frac{1}{4}(K\left(\frac{3-2}{1}\right) + K\left(\frac{3-2.5}{1}\right) + K\left(\frac{3-3.5}{1}\right) + K\left(\frac{3-0.5}{1}\right)) = \frac{1}{4}(0 + 0.5 + 0.5 + 0) = \frac{1}{4}$

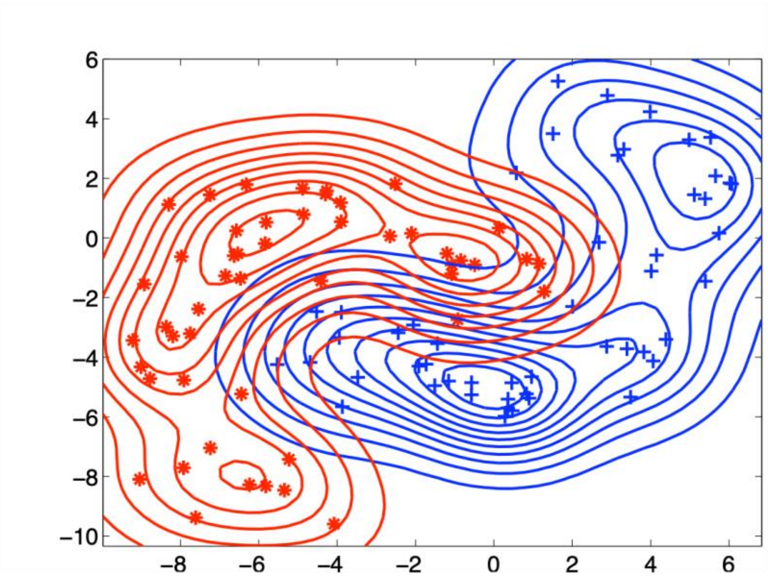
TU Delft

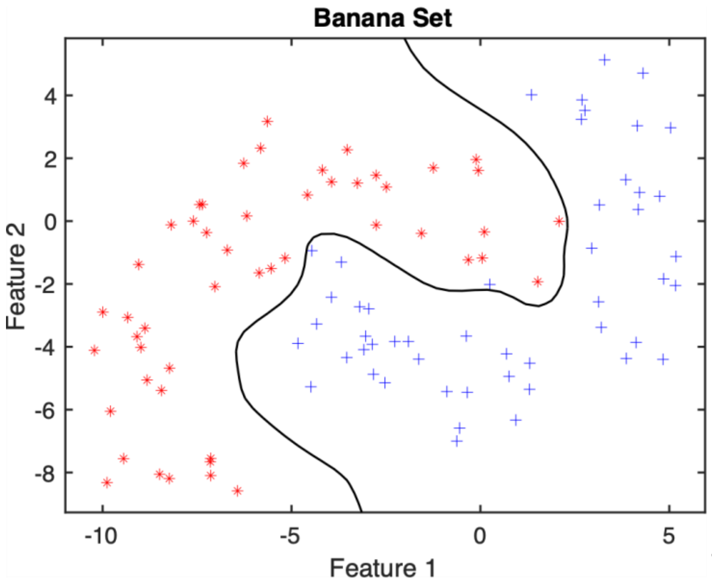
Parzen classifier

- Fix shape and size of the kernel:
  - Gaussian kernal and Identity matrix as covariance matrix

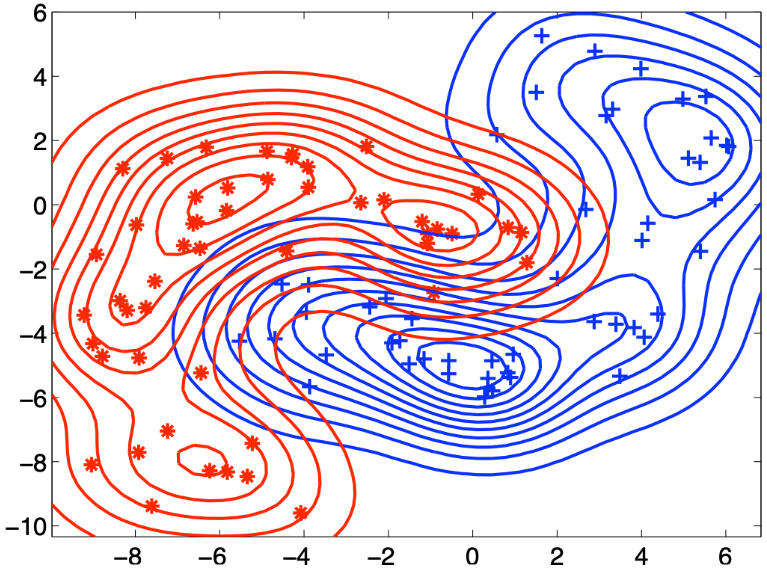
$$(x|y_i) = \frac{1}{n_i}\sum_{j=1}^{n_i} N(x|x_j^{(i)}, hI)$$

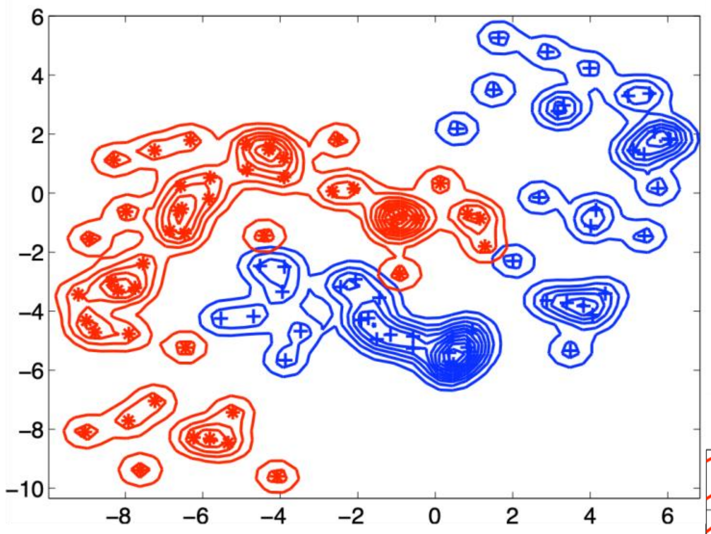
- For classification use Bayes rule
  - $p(x|y_1)p(y_1) > p(x|y_2)p(y_2)$

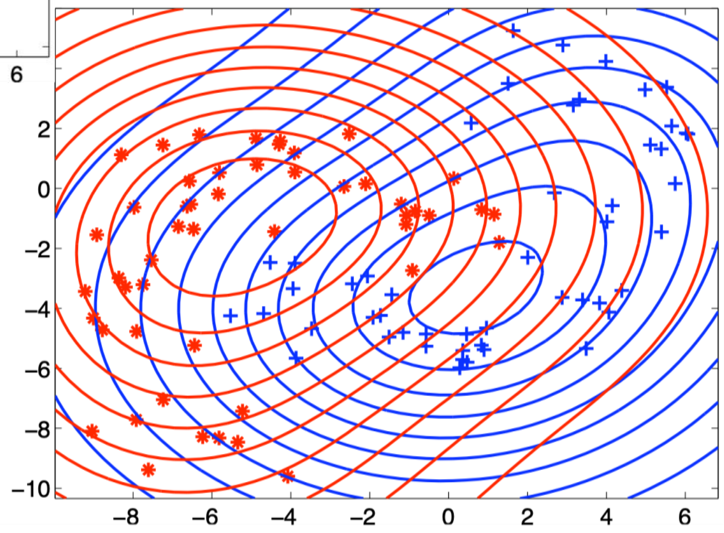




Does h matter? Intuition on parzen width parameter







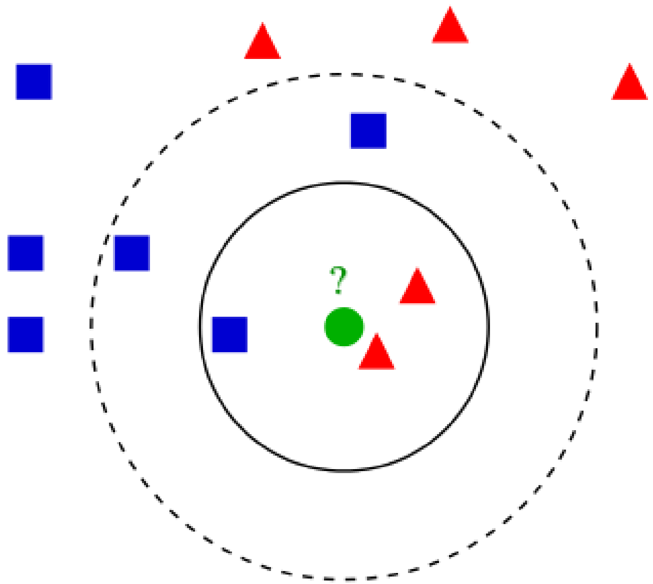
TUDelft

Summary of Parzen/Kernel density estimation

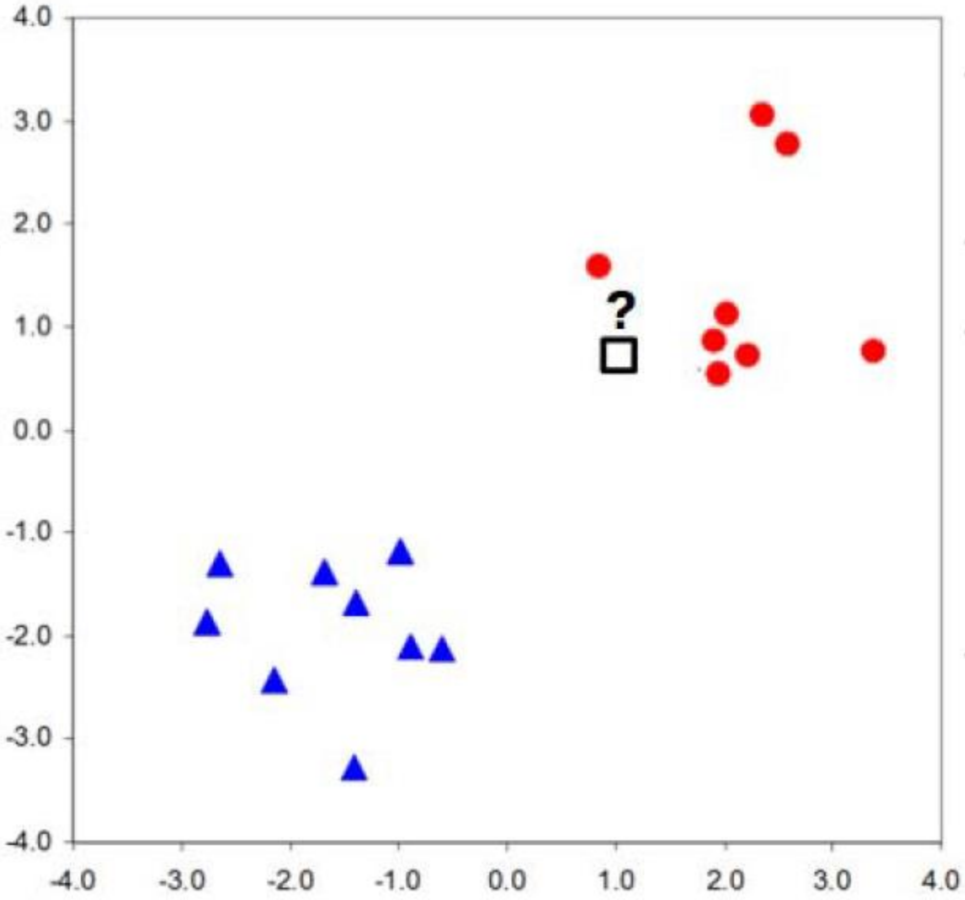
- Does not assume known distribution
- Estimates probability densities using kernel function
- Uses kernel funtion of fixed shape and width
- Width matters

TU Delft

K-nearest Neighbours



K-nearest neighbour intuition

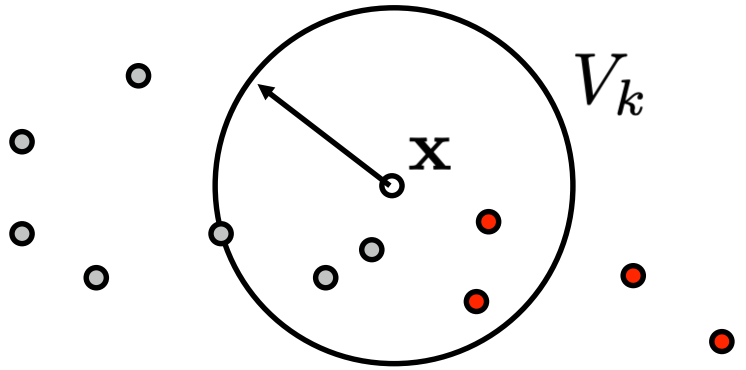


- Is the box red or blue?

- How did you do it?
- Nearby points are red

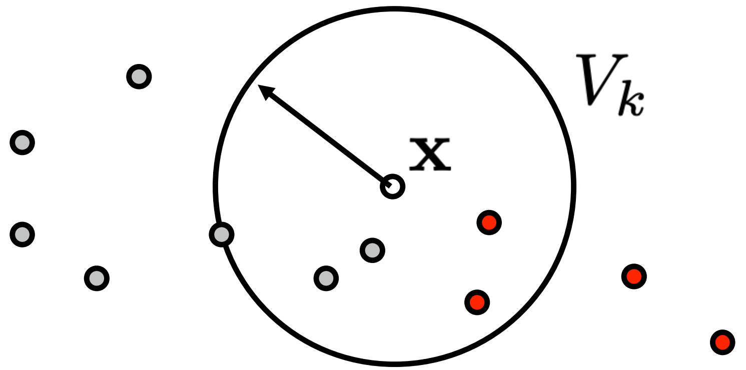
K-nearest neighbour density estimation

- Locate the cell on the new point x
- Do **not** fix the volume of the cell (V)
- Grow the cell until it covers k objects (find the k-th neighbor)



e.g. k=5

K-nn density estimation

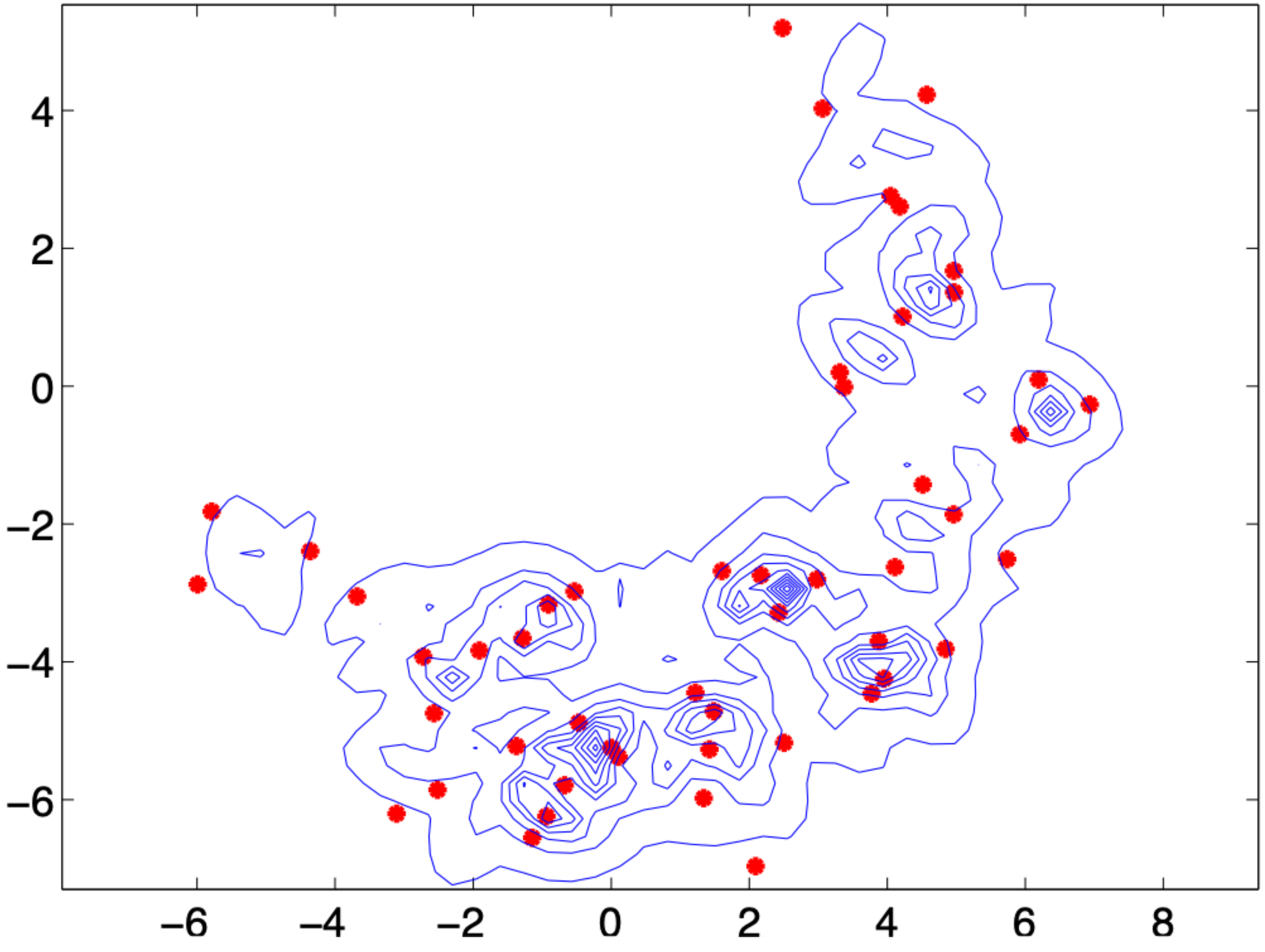


$$\hat{p}(x|y_i) = \frac{k_i}{n_i V_k}$$

- Where V_k is the volume of the sphere centered at x
  - with radius r, being the distance to the k-th nearest neighbour;
- k_i is the number of neighbours of class i within V_k
- n_i is the number of data points of class i in the dataset

- Bayes: $\hat{p}(x|y_i)\hat{p}(y_i) > \hat{p}(x|y_j)\hat{p}(y_j) \rightarrow\ k_i > k_j$

K-nn density estimate

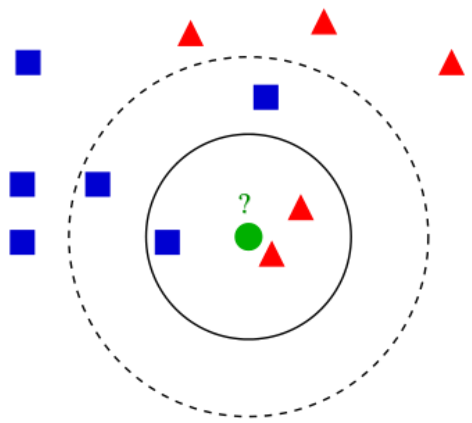


K-nn classification algorithm (lab)

- Given:
  - training examples $\{x_i, y_i\}$
    - x_i attribute-value representation of examples
    - y_i class label: {apple, pear}, digit {0,1, ... 9} etc.
  - testing point x that we want to classify
- Algorithm:
  - compute distance $D(x, x_i)$ to every training example x_i
  - select k closest instances $x_{i1} \dots x_{ik}$ and their labels $y_{i1} \dots y_{ik}$
  - output the class y^* which is most frequent in $y_{i1} \dots y_{ik}$ (**majority vote**)

TU Delft

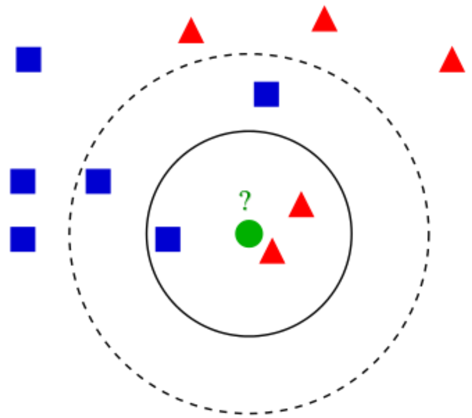
What is the influence of k?



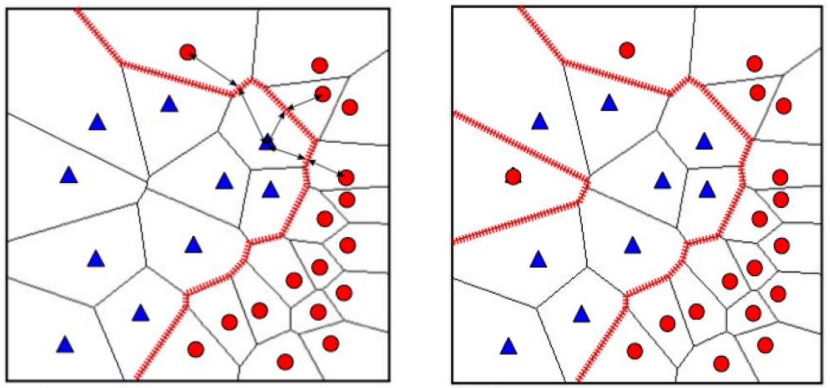
- What is the largest/smallest value of k that you can choose?
  - What will be the classification error then?

TU Delft

What is the influence of k?

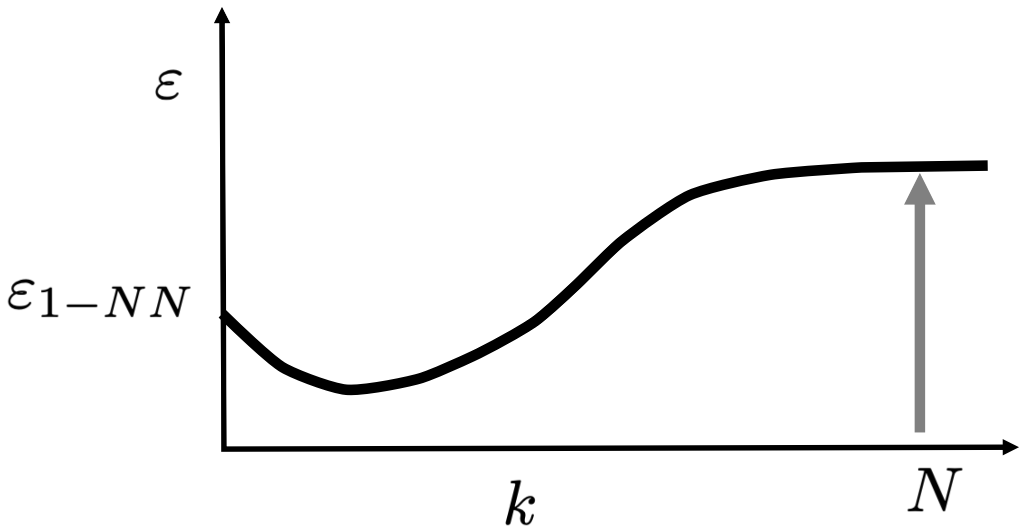


- Value of k has strong effect on k-nn performance
  - Large value ➔ everything classified as the most probable class
  - Small value ➔ highly variable, unstable decision boundaries, eg. for 1-nn:



Choosing the value of k

- Selecting the value of k
  - set aside a portion of the training data (validation set)
  - vary k
  - Pick k that gives best generalization performance



K-nn resolving ties

- Equal number of positive/negative neighbours ?

- Resolving ties:
  - use odd k (doesn't solve multi-class)
  - breaking ties:
    - random: flip the coin to decide positive/negative
    - prior: pick class with greater prior
    - nearest: use 1-nn classifier to decide

Distance measures

- The key component of the kNN algorithm
  - defines which examples are similar and which aren't
  - can have strong effect on performance

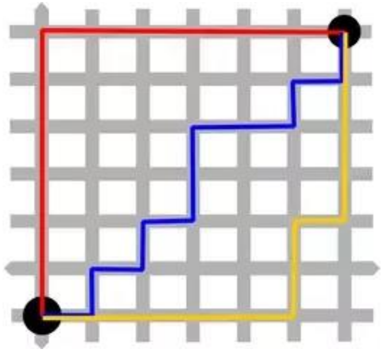
- Euclidean (numeric features):

$$D(x, x') = \sqrt{\sum_d |\, x_d - x'_d |^{\,2}}$$

Distance measures

- Manhattan distance

$$D(x,x') = \sum_d |x_d - x'_d|$$



- Hamming (categorical features):
  - number of features where x and x' differ

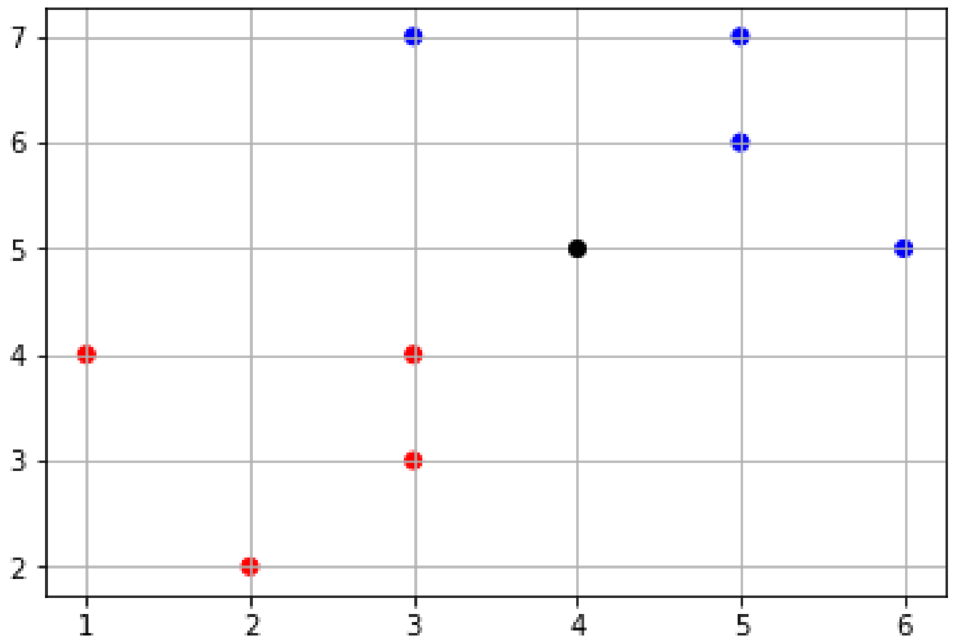
$$D(x,x') = \sum_d 1_{x_d \neq x'_d}$$

- Other (out of scope), eg.:
  - Kullback-Leibler (KL) divergence (for histograms)
  - Custom distance measures (BM25 for text)

K-nn example

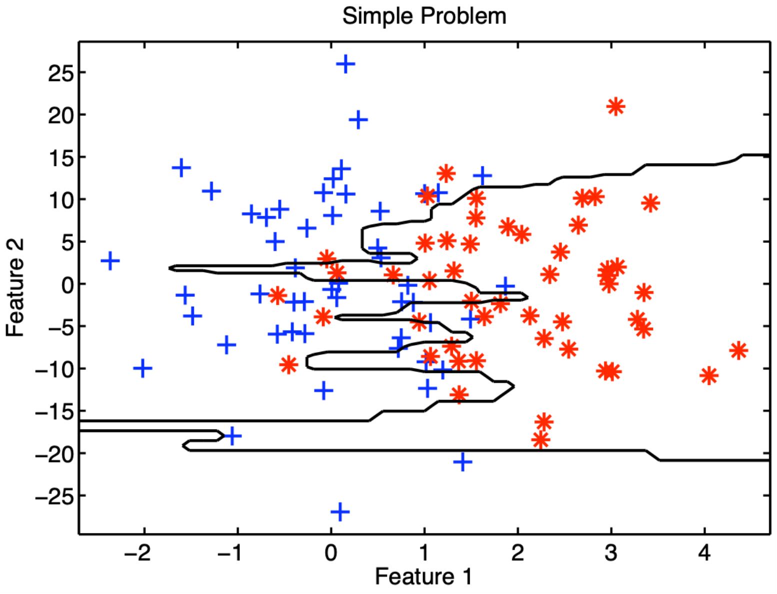
- Given a labeled two-dimensional data set:
  - Red label: (1,4); (2,2); (3,3); (3,4);
  - Blue label: (3,7); (5,7); (5,6); (6,5);
- Predict the label of a new black point (4, 5) using 3-nn classifier with Manhattan distance.

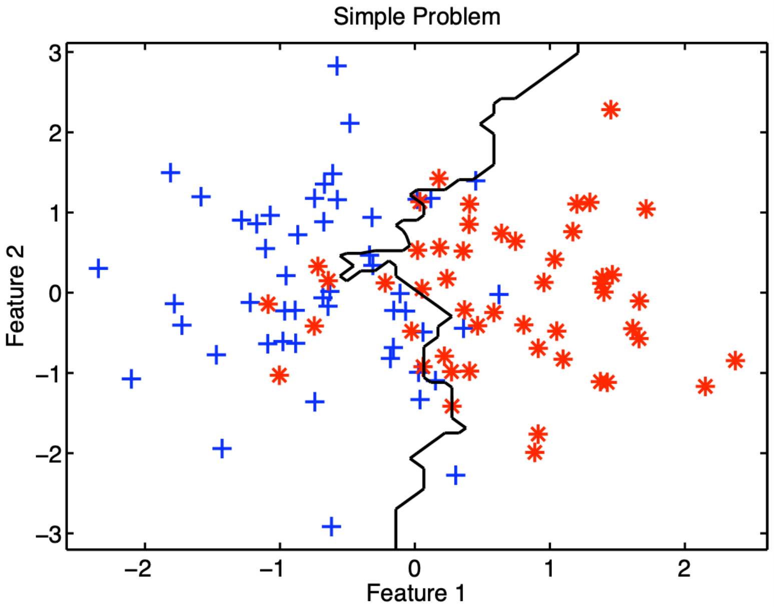
A. Red label
B. Blue label



Sometimes strange results

- How is this possible?





Scale your features!

K-nn pros and cons

- Simple and flexible classifiers
- Often a very good classification performance
- It is simple to adapt the complexity of the classifier

- Relatively large training sets are needed
- The complete training set has to be stored
- Distances to all training objects have to be computed
- The features have to be scaled sensibly
- The value for k has to be optimized

Naïve Bayes Classifier

TU Delft

Recap Bayes classifer

- For classification we need $p(y|x)$
- We can use Bayes' theorem if we can estimate $p(y)$ and $p(x|y)$

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

- Assigning an object to the class with the maximum posterior probability gives the Bayes' classifier

$$p(x|y_1)p(y_1) > p(x|y_2)p(y_2)$$

TUDelft

Warming up question

- Suppose we have trained a generative model and now get a new test example x. Our model tells us that: $p(x|y_o) = 0.01$, $p(x|y_1) = 0.03$ and $p(y_0) = p(y_1) = 0.5$

- What is $p(y_1|x)$?
  A. 0.015
  B. 0.25
  C. 0.75
  D. Insufficient information to compute. We also need to know the p(x).

Solution

- $p(y_1|x) = \frac{p(x|y_1)p(y_1)}{p(x)}$
- $p(x) = p(x|y_1)p(y_1) + p(x|y_0)p(y_0)$

- $p(y_1|x) = \frac{0.03*0.5}{0.03*0.5+0.01*0.5} = 0.75$

Density estimation

- So, we want to estimate a class conditional probability density function:

$$p(x|y)$$

- Typically, each feature vector **x** has many features:

$$p(x|y) = p(x_1, x_2, x_3, x_4, \dots, x_d|y)$$

- To estimate this joint pdf (conditional on the class), we need LOTS of data... (curse of dimensionality)

Naive Bayes: conditional independence assumption

- We make a strong assumption: all features are independent
- We assume conditional independence given y
- We just estimate $p(x_i|y)$ per feature and multiply them.

$$p(x|y) = p(x_1, x_2, x_3, x_4, \dots, x_d|y) = \prod_{i=1}^{d} p(x_i|y)$$

$$= p(x_1|y)p(x_2|y) \dots p(x_d|y)$$

- No curse of dimensionality!

Conditional independence example

- We assume conditional independence of two variables given a third variable.
- Example: probablity of going to the beach and having a heartstroke may be independent if we know the wheather is hot

$$p(B,S|H) = p(B|H)p(S|H)$$

- Hot weather “explains” all the dependence between beach and heartstroke
- In classification: class value explains all the dependence between features

TU Delft

Naive Bayes: conditional independence assumption

- We make a strong assumption: all features are independent
- We assume conditional independence given y
- We just estimate $p(x_i|y)$ per feature and multiply them.

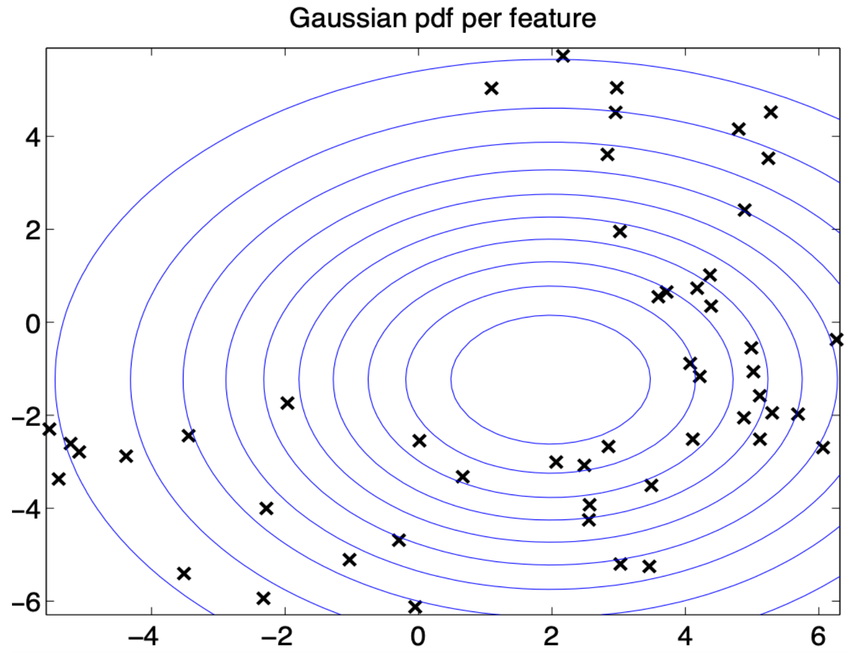
$$p(x|y) = p(x_1, x_2, x_3, x_4, \dots, x_d|y) = \prod_{i=1}^{d} p(x_i|y)$$

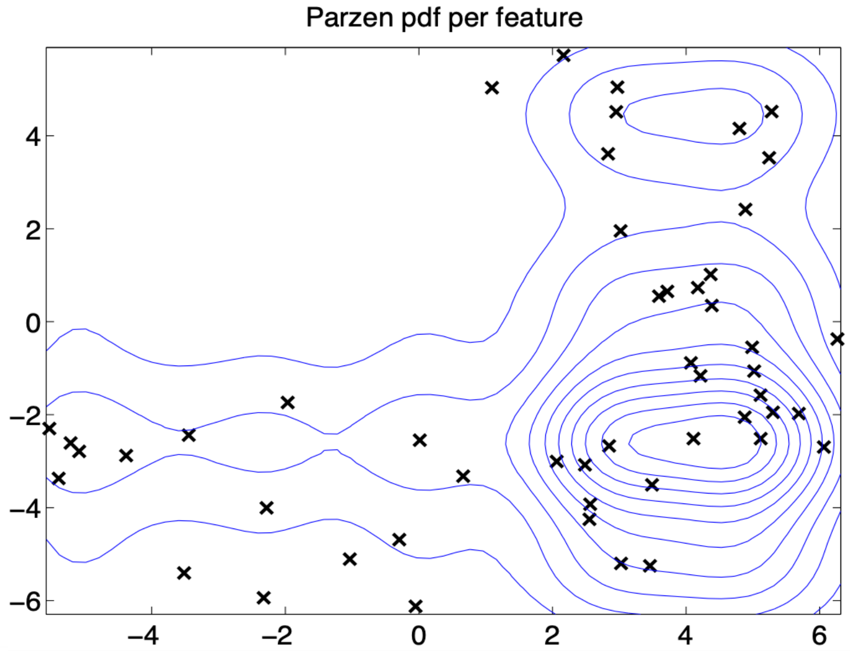
$$= p(x_1|y)p(x_2|y) \dots p(x_d|y)$$

- No curse of dimensionality!

Parametric vs. non-parametric

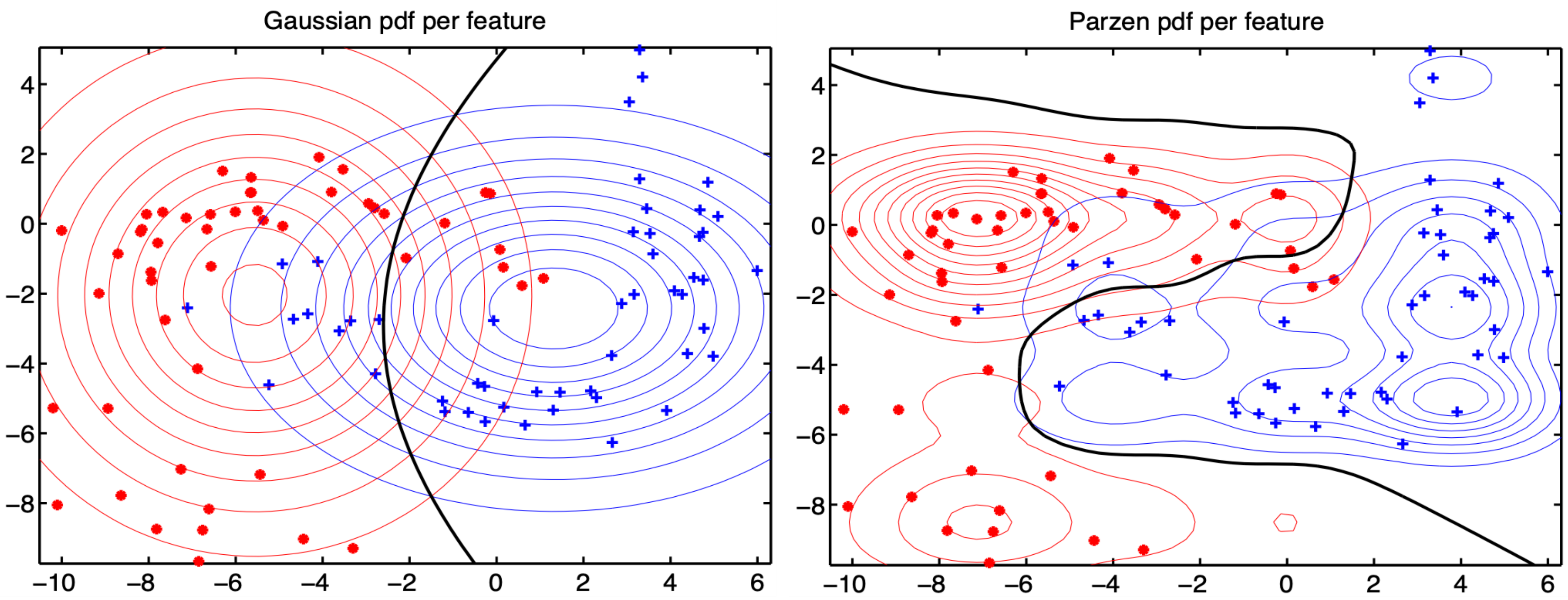
- You still have to choose a model for $p(x_i|y)$





TUDelft

Naive Bayes classifier



Continuous data example

- Distinguish children from adults based on size
  - Classes: y = {a, c}, features: x = {height (cm), weight (kg)}
  - Training examples: 4 adults, 12 children
- Class probabailities $p(a) = \frac{4}{4+12} = 0.25,\ p(c) = 0.75$
- Model for adults:
  - Assume height and weight are independent
  - Height, estimate Gaussian with mean, variance

$$\begin{cases} \mu_{h,a} = \frac{1}{4} \sum_{i:y_i=a} h_i \\ \sigma_{h,a}^2 = \frac{1}{4} \sum_{i:y_i=a} (h_i - \mu_{h,a})^2 \end{cases}$$

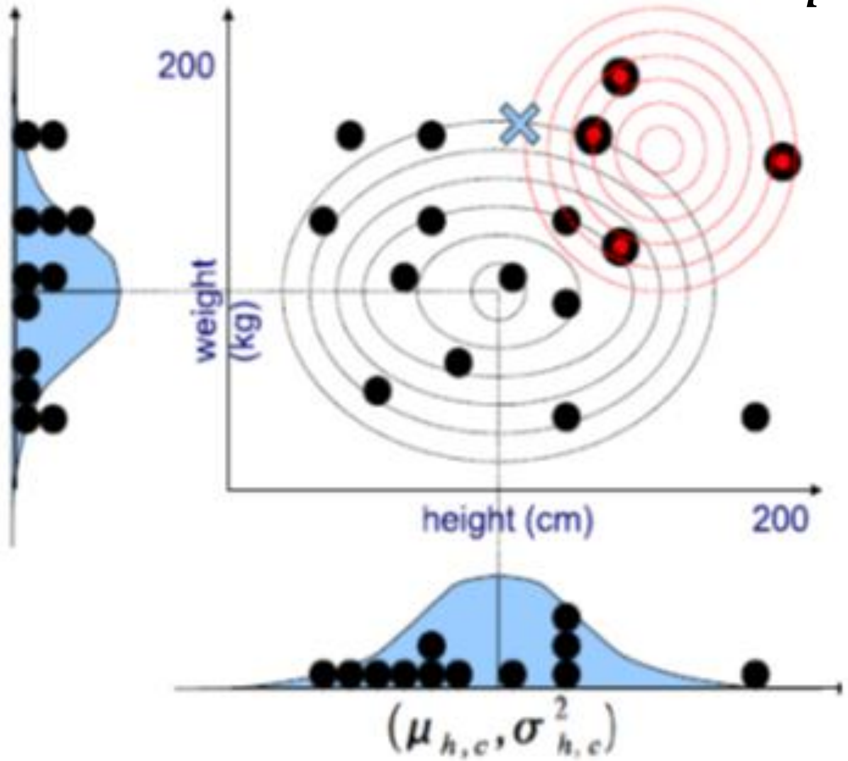
  - Weight, estimate Gaussian $(\mu_{w,a}, \sigma_{w,a}^2)$
- Model for children: use $(\mu_{h,c}, \sigma_{h,c}^2), (\mu_{w,c}, \sigma_{w,c}^2)$

TU Delft

Continuous example

$$p(w|a) = \frac{1}{\sqrt{2\pi\sigma_{w,a}^2}} exp - (\frac{w \; - \; \mu_{w,a}{}^2}{2\sigma_{w,a}^2})$$

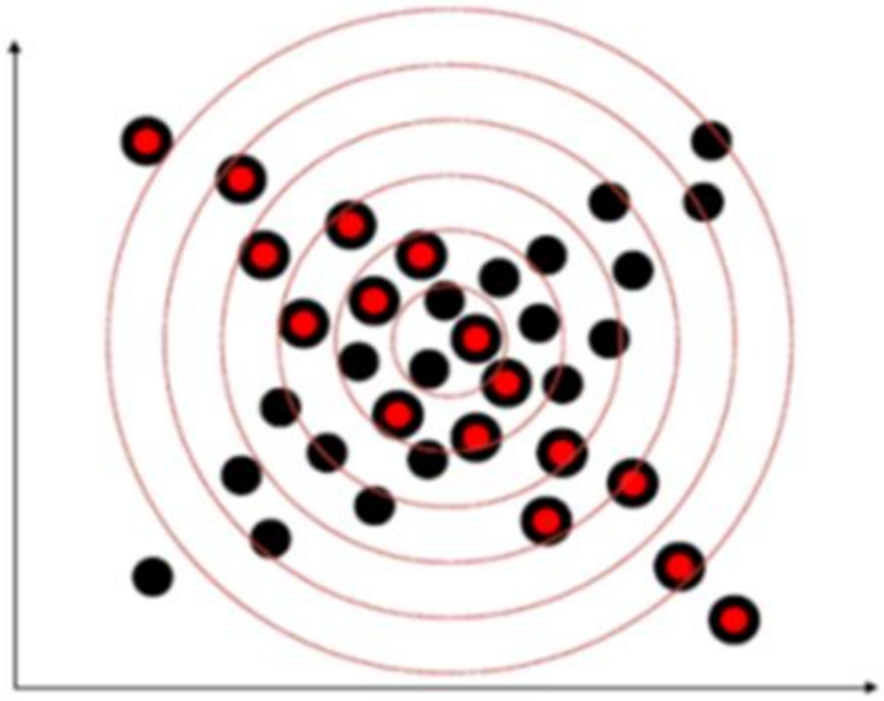
$$p(h|a) = \frac{1}{\sqrt{2\pi\sigma_{h,a}^2}} exp - (\frac{h \; - \; \mu_{h,a}{}^2}{2\sigma_{h,a}^2})$$



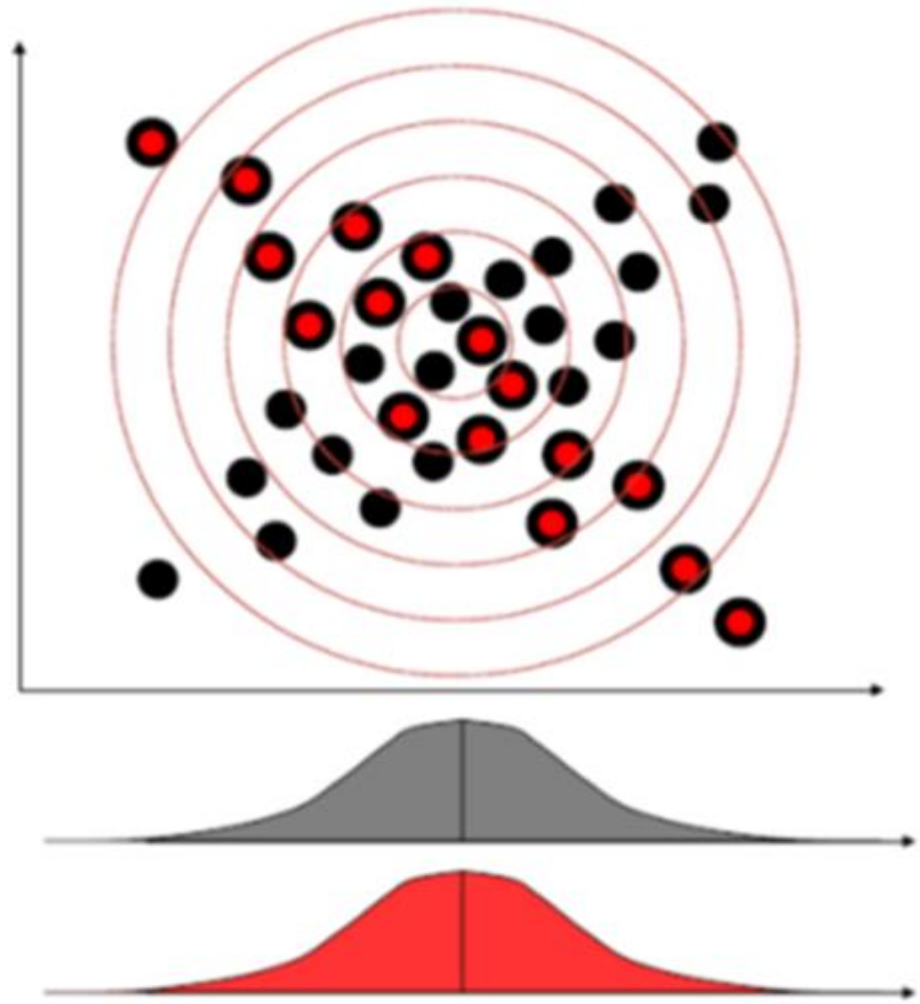
$$p(x|a) = p(w|a)p(h|a)$$

$$p(a|x) = \frac{p(x|a)p(a)}{p(x)}$$

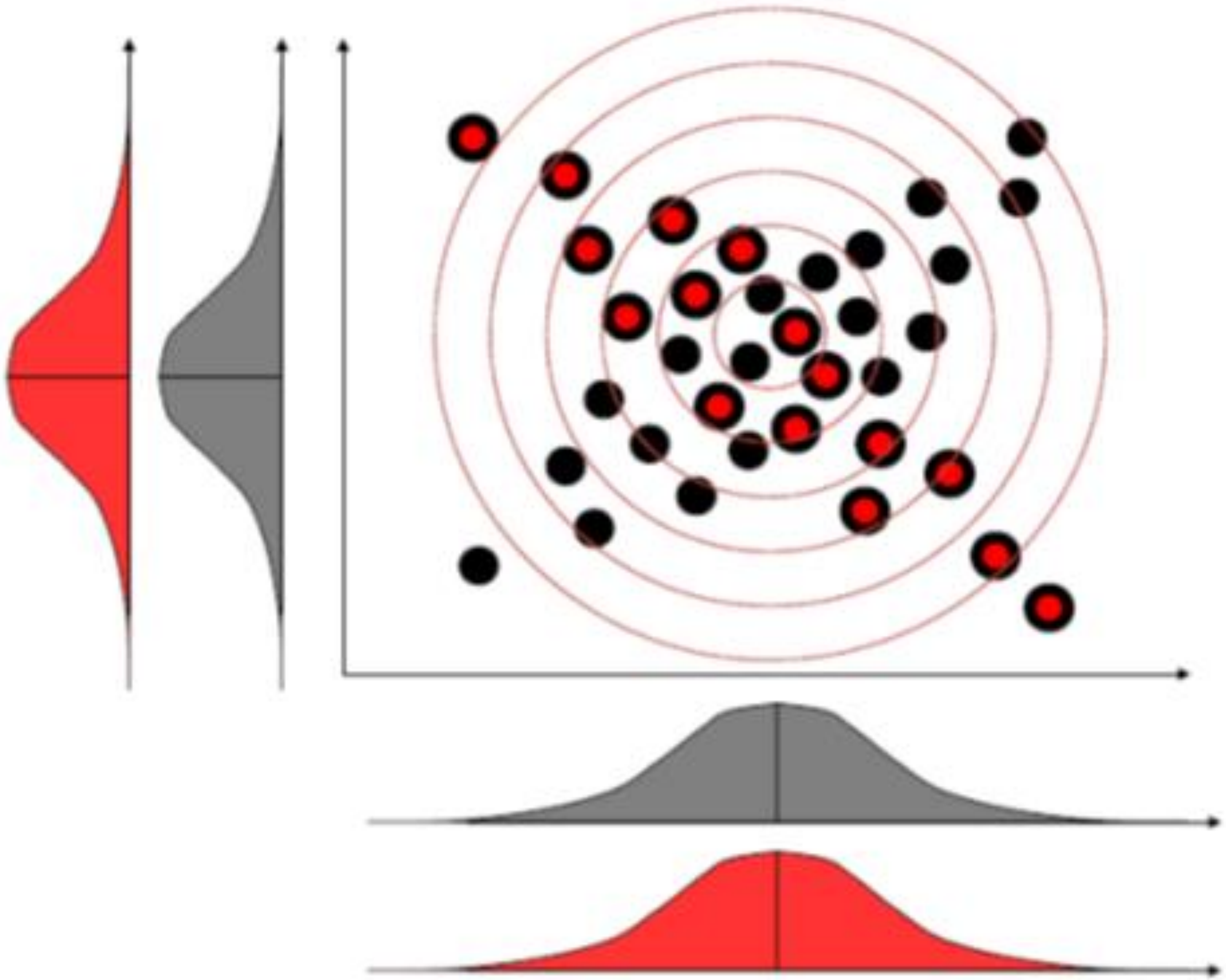
Problem with Naive Bayes



Problem with Naive Bayes



Problem with Naive Bayes



Discrete example

- Separate spam from valid email (features = words)

D1: “send us your password”	spam
D2: “send us review”	valid
D3: “review your password”	valid
D4: “review us”	spam
D5: “send your password”	spam
D6: “send us your account”	spam

p(spam) = 4/6 p(valid) = 2/6		
	spam	valid
Password	2/4	½
Review	1/4	2/2
Send	3/4	½
Us	3/4	½
Your	3/4	½
Account	1/4	0/2

- New email “review us now”

Discrete example

p(spam) = 4/6 p(valid) = 2/6		
	spam	valid
Password	2/4	½
Review	1/4	2/2
Send	3/4	½
Us	3/4	½
Your	3/4	1/2
Account	1/4	0/2

- New email: “review us now”

- p(“review us”|spam) =
  p([0, 1, 0, 1, 0, 0]|spam) =
  $(1-\frac{2}{4})(\frac{1}{4})(1-\frac{3}{4})(\frac{3}{4})(1 - \frac{3}{4})(1-\frac{1}{4})$ = 0.0044

- p(“review us”|valid) =
  p([0, 1, 0, 1, 0, 0]|valid) =
  $\left(1-\frac{1}{2}\right)\left(\frac{2}{2}\right)\left(1-\frac{1}{2}\right)\left(\frac{1}{2}\right)\left(1 - \frac{1}{2}\right)\left(1-\frac{0}{2}\right)$ = 0.0625

Solution

p(spam) = 4/6 p(valid) = 2/6		
	spam	valid
Password	2/4	½
Review	1/4	2/2
Send	3/4	½
Us	3/4	½
Your	3/4	½
Account	1/4	0/2

- p(“review us”|spam) = 0.0044
- p(“review us”|valid) = 0.0625

- p(“review us”|spam)p(spam) = 0.0044 * 4/6 = 0.0029
- p(“review us”|valid)p(valid) = 0.0625 * 2/6 = 0.02

- Note: identical example!

D1: “send us your password”	spam
D2: “send us review”	valid
D3: “review your password”	valid
D4: “review us”	spam
D5: “send your password”	spam
D6: “send us your account”	spam

Zero frequency problem

p(spam) = 4/6 p(valid) = 2/6		
	spam	valid
Password	2/4	½
Review	1/4	2/2
Send	3/4	½
Us	3/4	½
Your	3/4	½
Account	1/4	0/2

- No email containing "account" is valid

  p("account"|valid) = 0/2

- Solution: never allow zero probabilities
  - Laplace smoothing: add a small positive number to the counts (K-> number of classes)

$$p(w|c) = \frac{num(w,c) + \varepsilon}{num(c) + K\varepsilon}$$

TU Delft

Fooling Naive Bayes

- Every word contributes independently to p(spam|email)
- Add lots of valid words into spam email.

Naive Bayes pros and cons

- Can handle high dimensional feature spaces
- Fast training time
- Can handle continuous and discrete data

- Can't deal with correlated features

Exercise Naive Bayes

- Predict if Bob will default his loan

<u>Bob:</u>

Homeowner: no

Maritial status: married

Job experience: 3

Home owner	Maritial status	Job experience	Deafulted
Yes	Single	3	No
No	Married	4	No
No	Single	5	No
Yes	Married	4	No
No	Divorced	2	Yes
No	Married	4	No
Yes	Divorced	2	No
No	Married	3	Yes
No	Married	3	No
Yes	Single	2	Yes

TU Delft

After practicing with the concepts of this lecture you should be able to:

- Explain the difference between parametric and non-parametric density estimation
- Explain Parzen, k-Nearest Neighbour and Naïve Bayes density estimation and classification in detail.
- Explain the advantages and disadvantages of those methods.
- Implement k-nn classifier in Python

TUDelft