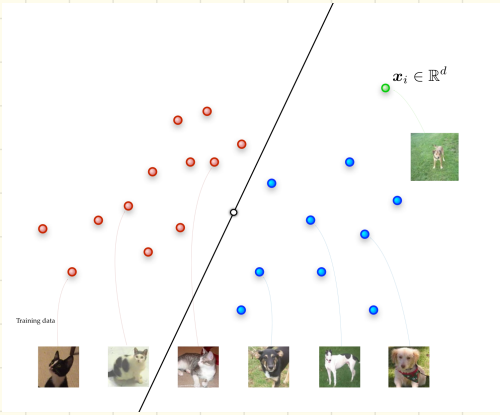
Topic 6: Classification

- Classification problem
- Non-parametric classification methods
- K-nearest neighbours classification
- The curse of dimensionality
- Logistic regression

Section 1: Introduction to classification problem

In the previous parts of the module we dealt with the regression problem of the form: find a function $f: \mathbb{R}^d \to \mathbb{R}^m$ such that $f(x^{(i)}) \approx y^{(i)}$ for a given samples $\{x^{(i)}, y^{(i)}\}_{i=1}^{s}$. In these problems $x^{(i)}$, $y^{(i)}$ were taken to be any d and m dimensional vectors.

What about cat vs dog problem? Can the outcome of the prediction be a number? Moreover, can it be any number from a continuous set?



As one can see from the example above, we expect to have a continuous variable for an input, but the output should be discrete, i.e. taken from a finite range of class labels (which can be a number as well as in regression problems)

Does it look like a regression problem? Yes and No
The key difference is a finite number of possible outcomes, but the methodology is very similar.

Def. Let $C = \{C_1, C_2, \dots, C_n\}$ be a finite set of class labels that are numerical values associated with n individual classes. Suppose we have a set of S input and output samples $\{x^{(i)}, y_i\}_{i=1}^{S}$ such that for every $i = \overline{1,S}$ one has $x^{(i)} \in \mathbb{R}^d$, $y_i \in C$. Then a classification problem is the problem that aims at finding a function $f: \mathbb{R}^d \to C$ such that for every $i = \overline{1,S}$ one has $f(x^{(i)}) \approx y_i$.

Open questions:
- what is the meaning of $\approx$?
- how to define a function with a finite number of possible outputs?

Remark: there is no ordering in labels. 1 is not supposed to be in any way better than 0.

Variety of class labels

Def. We say that a classification problem is a binary classification problem if the number of class labels is 2. For example,
- $C = \{0, 1\}$
- $C = \{-1, 1\}$
- etc.

We say that a classification problem is a multiclass classification problem if the number of class labels satisfies .
For example,
- $C = \{-1, 0, 1\}$
- $C = \{1, 2, \dots, n\}$
- etc.

Examples

Surviving the Titanic disaster

Input:
- Ticket class
- Wealth
- Ticket price
- Distance to the boats

Output:
Survived (1) or not(0)

AIM: To build a model to predict whether the person would survive or not.

Train delays prediction

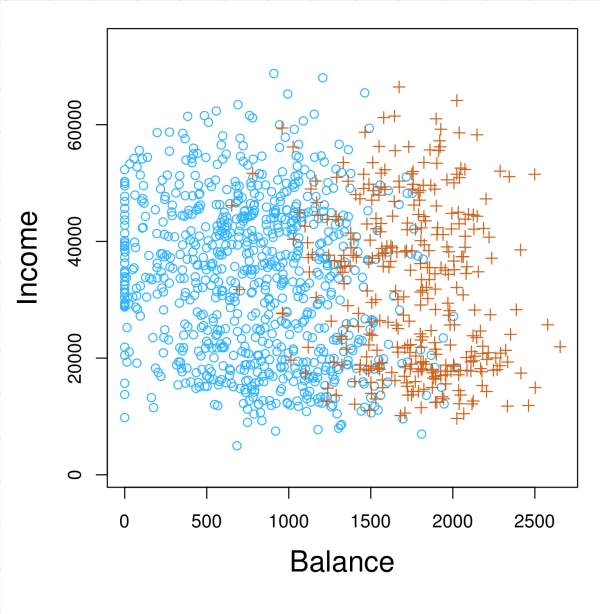
Input:
- Delays per last week
- Load of trains
- Load of stations
- Signaling problems along the route

Output:
- delay time (regression problem)
- delay repay (classification)

If delay time is less than 30 mins-> no repay.
If delay time is more than 30 mins-> repay.

AIM: To predict whether a passenger will be payed.

Credit default prediction



Input:
- Person's income
- Credit balance

Output:
- Whether the person is likely to default (1) or not (0).

Observation:
Default happens to persons with the balance above an absolute, not relative treshold

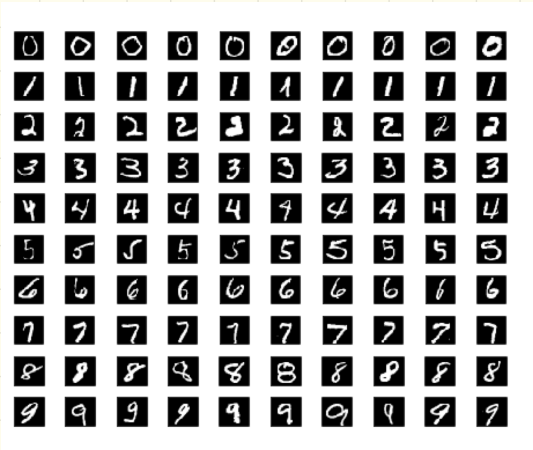
AIM: To build a prediction whether the person is likely to default based on its current income and credit balance.

Targeted marketing

Input: age, gender, employment, status.

Output: whether to offer the goods (1) or not (0)

Handwritten numbers recognition



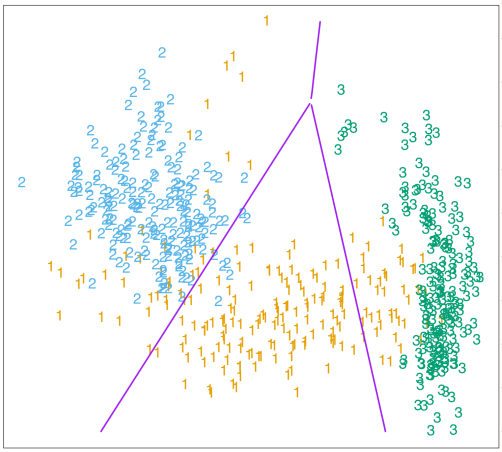
Input:
- 28x28 pixels with BW data

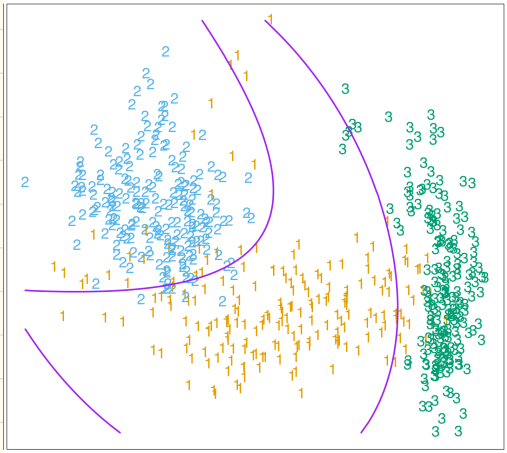
Output:
- A number in the range 0-9

AIM: To write a model to recognise a handwritten number/letter

Classifier

In order to classify the data we need to introduce a classifier - the method which splits the input space into several regions, each of which corresponds to one class only. The boundaries of these regions are called decision boundaries. The classifiers could be linear and non-linear as well.





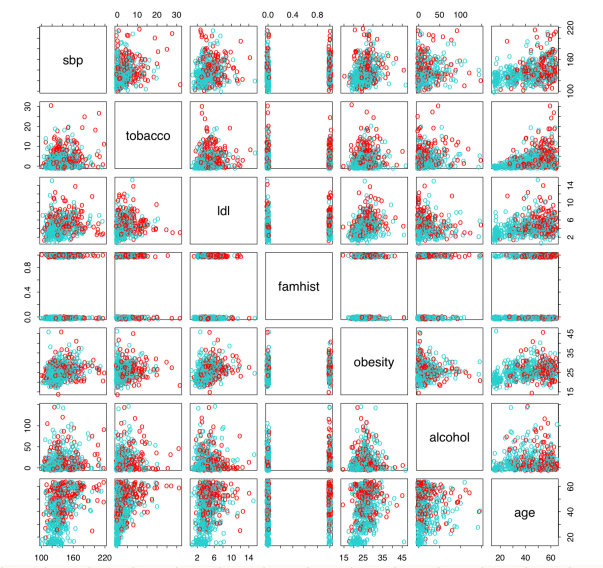
Aims of classification

Classification itself: we are constructing a predictor based on a training data and are interested in applying it to a new data.
Example: credit card company aims to predict what is the credit limit which the customer shouldn't overcome not to default.

Understanding the 'cause' of something: we are interested in the understanding of reasons of concrete prediction.
Example: credit card company aims to understand what is the most important reason for a customer to default, small income or large debt.

Remark: for the second task it is important to have simple models.

Example:



South Africa Heart disease data
Blood pressure, smoking habits, cholesterol level, family history, obesity, alcohol, age.

Classification as regression

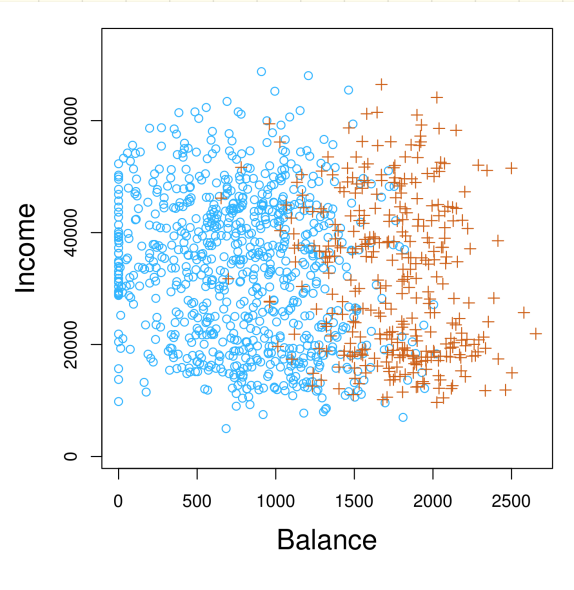
Let us consider a binary classification problem as a regression one. I.e.

$$f_w(x) : \mathbb{R}^d \to \mathbb{R}$$

$$\hat{w} = \arg\min \left\{ \frac{1}{2s} \sum_{i=1}^{s} \left| f_w(x^{(i)}) - y_i \right|^2 + \alpha \|w\|^2 \right\}$$

$$\hat{f}_{\hat{w}}(x) : \mathbb{R}^d \to C$$

$$\hat{f}_{\hat{w}}(x) = \begin{cases} 1, & f_{\hat{w}}(x) \geq 1/2 \\ 0, & f_{\hat{w}}(x) < 1/2 \end{cases}$$

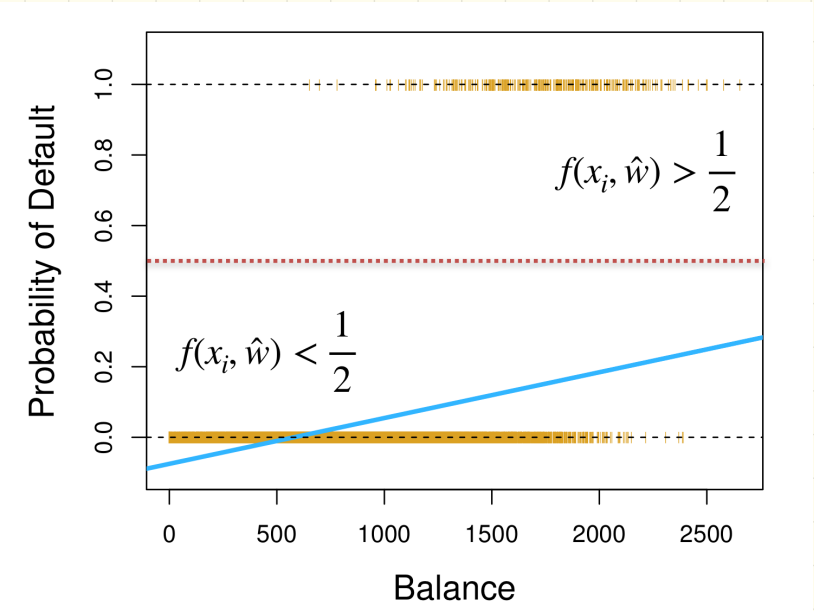


There is an evidence that the balance is the dominant reason for a default. Let us try a linear model

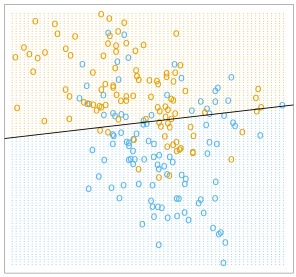
$$f_w(x) = w_0 + w_1 x$$

where x is the balance.

$$y_i = \begin{cases} 0, & \text{no default} \\ 1, & \text{default} \end{cases}$$


Probability of Default
0.0
0.2
0.4
0.6
0.8
1.0
$f(x_i, \hat{w}) > \frac{1}{2}$
$f(x_i, \hat{w}) < \frac{1}{2}$
0
500
1000
1500
2000
2500
Balance

Section 2: Non-parametric classification



In regression analysis we have always searched for the parametrised prediction function, with the parameters being selected to minimise an error. The same principle could be applied in classification problems. But we will start our discussion of classification methods with non-parametric ones.

K-nearest neighbours classification

Suppose we are given S data points $\{x^{(i)}, y_i\} \in \mathbb{R}^d \times C$ sampled from the distribution $\mathcal{D}$. What is the appropriate class for a new data point x? Let $N_k(x, \mathcal{D})$ be a set of k closest data inputs to x.

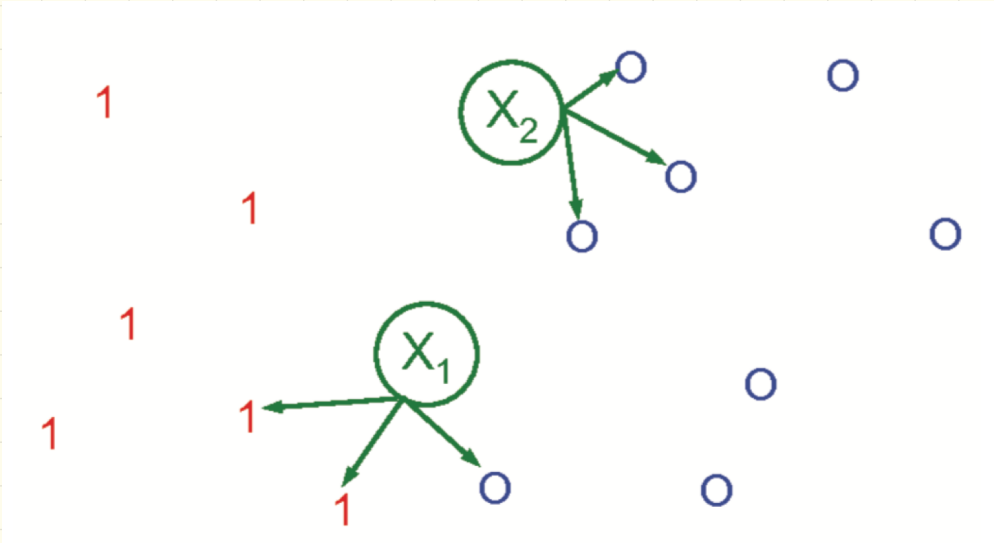
Idea: we define the probability of the point to belong to class C_i as the ratio of its neighbours belonging to class C_i.

$$p(f(x)=c) = \frac{1}{k} \sum_{i \in N_k(x,\mathcal{D})} \ell(y_i = c), \quad \forall c \in C$$

$$\ell(z) = \begin{cases} 1, & z \text{ is true} \\ 0, & z \text{ is false} \end{cases}$$

$$f(x) = \arg\max_c \; p(f(x)=c)$$

Example (K =3)



x_1

$p(y=0 \mid x_1, \mathcal{D}, K=3) = \frac{1}{3}(0+0+1) = \frac{1}{3}$

$p(y=1 \mid x_1, \mathcal{D}, K=3) = \frac{1}{3}(1+1+0) = \frac{2}{3}$

$f(x_1) = 1$

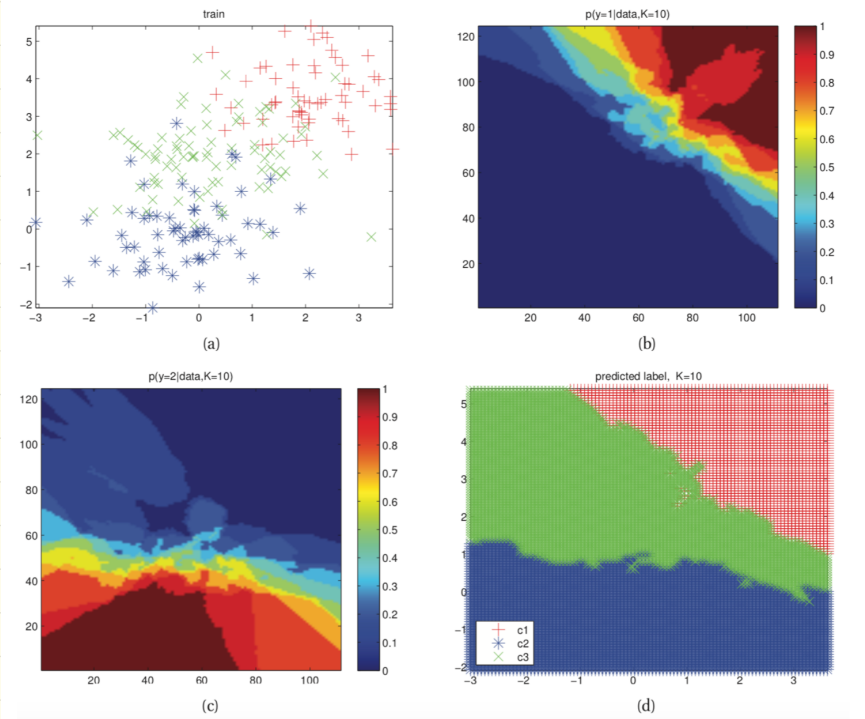
x_2

$p(y=0 \mid x_2, \mathcal{D}, K=3) = \frac{1}{3}(1+1+1) = 1$

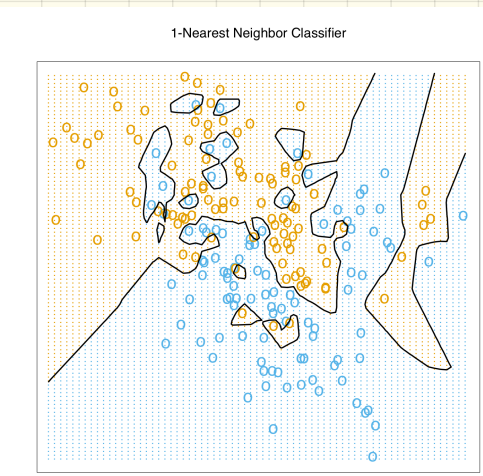
$p(y=1 \mid x_2, \mathcal{D}, K=3) = \frac{1}{3}(0+0+0) = 0$

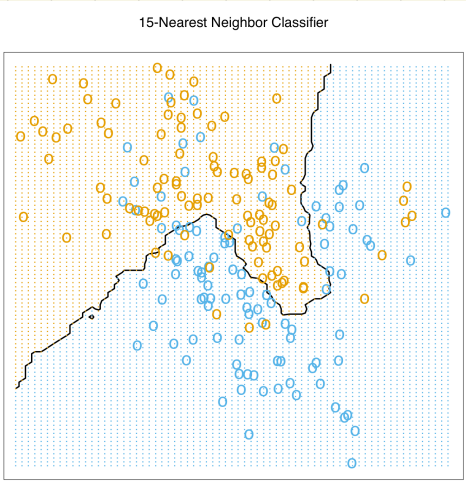
$f(x_2) = 0$

Example



Example





Section 3: The curse of dimensionality

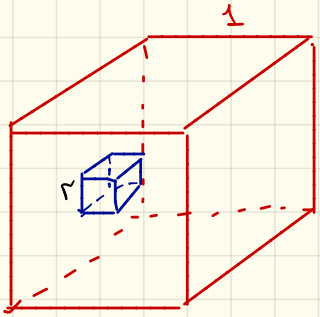
How the dimensionality of input could interfere the result of prediction?

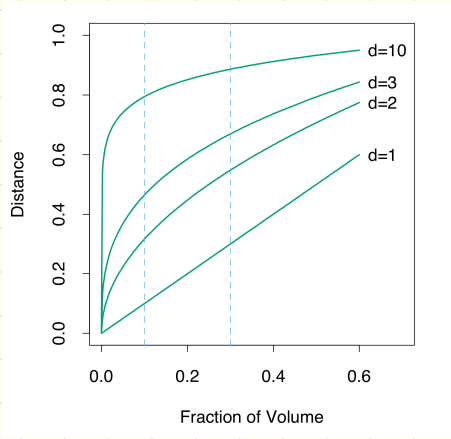
Size of training region

Generalising becomes exponentially harder with the dimension growth: a fixed size training sets contain vanishing amount of data points. So to get a fixed number of points inside the training set, one needs to take larger sizes of training regions.

Imagine all point lie in the unit cube $[0,1]^d$. Let $0 \le a \le 1$. What is the ratio of points that belong to the cube $[a, a+r]^d$?

$$\#\left\{x^{(i)} \in [a, a+r]^d\right\} = S \cdot r^d \xrightarrow[d \to \infty]{} 0$$





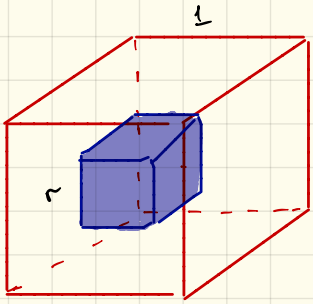
1% for $d = 10 \Rightarrow r = 0.63$

10% for $d = 10 \Rightarrow r = 0.8$

Random choice of training points

As the dimension increases the selection of training points become quite random. This is due to the larger distances in high dimensions between points.

Let all points belong to the unit cube. Assume they are uniformly distributed. We consider a cube with side size r centred at $(1/2, 1/2, \ldots, 1/2)$.
What is the probability that this cube contains no points at all?



$$\mathbb{P}[\text{a point} \notin \text{cube}] = 1 - r^d$$

$$\mathbb{P}[\text{no points}] = (1-r^d)^s \qquad \downarrow r \to 1$$

$$r^* : \ \mathbb{P}[\text{no points}] = 1/2$$

$$r^* = \left(1 - 2^{-1/s}\right)^{1/d}$$

$d = 10$
$s = 500 \Rightarrow r \approx 0.52$