QUEEN'S UNIVERSITY BELFAST
SCHOOL OF ELECTRONICS, ELECTRICAL ENGNIEERING AND COMPUTER SCIENCE

Data Analysis & Visualisation

CSC3062

BEng (CS & SE), MEng (CS & SE), BIT & CIT

Dr Reza Rafiee

Semester 1 2019

QUEEN'S UNIVERSITY BELFAST

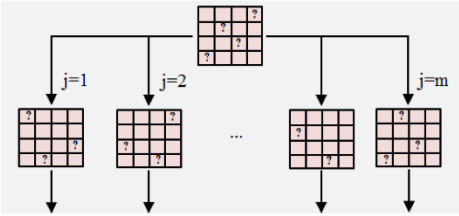
SCHOOL OF ELECTRONICS, ELECTRICAL ENGNIEERING AND COMPUTER SCIENCE

Missingness and multiple imputation

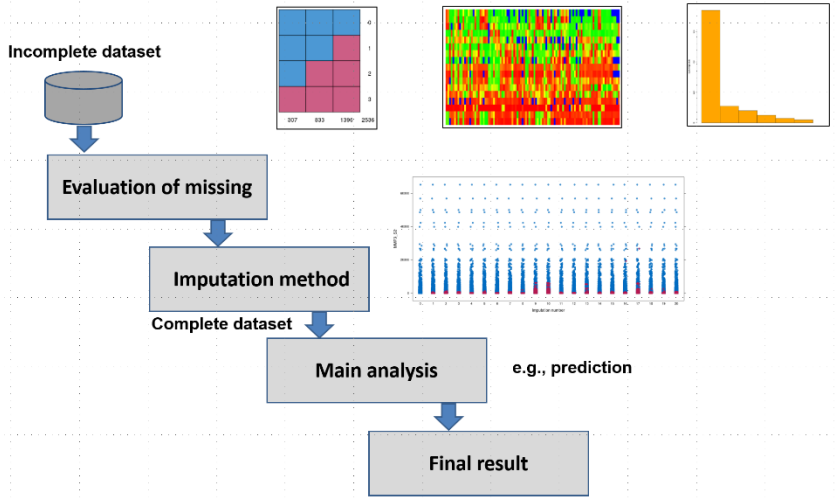
What is multiple imputation?

- This statistical technique (algorithm) takes the incomplete dataset (i.e., including missing data) and returns m imputed datasets with no missing values.

m is a user-selected parameter



Impact of imputation on final result



QUEEN'S UNIVERSITY BELFAST | SCHOOL OF ELECTRONICS, ELECTRICAL ENGNIEERING AND COMPUTER SCIENCE

Principal component analysis (PCA)

PCA analysis using *prcomp()* package

	England	N Ireland	Scotland	Wales
Alcoholic drinks	375	135	458	475
Beverages	57	47	53	73
Carcase meat	245	267	242	227
Cereals	1472	1494	1462	1582
Cheese	105	66	103	103
Confectionery	54	41	62	64
Fats and oils	193	209	184	235
Fish	147	93	122	160
Fresh fruit	1102	674	957	1137
Fresh potatoes	720	1033	566	874
Fresh Veg	253	143	171	265
Other meat	685	586	750	803
Other Veg	488	355	418	570
Processed potatoes	198	187	220	203
Processed Veg	360	334	337	365
Soft drinks	1374	1506	1572	1256
Sugars	156	139	147	175

Input_dataset

PCA →

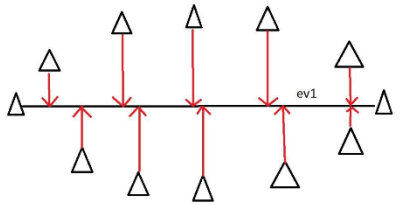
	England	N Ireland	Scotland	Wales
PC1	-144.993	477.3916	-91.8693	-240.529
PC2	2.532999	58.90186	-286.082	224.6469
PC3	105.7689	-4.8779	-44.4155	-56.4756

Reduced dataset

Summarises of features

PC | eigenvector and eigenvalue

The horizontal line is therefore the principal component in this example.



The direction of this line is called **eigenvector**.

An **eigenvalue** is a number telling us how spread out the data is on the line.

Consider 9126 samples with 440 features

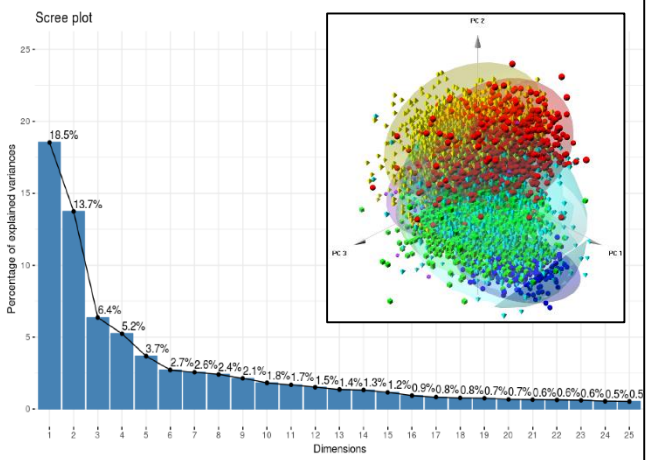
RNA-seq (FPKM), **6 immune subgroups**
440 genes (5 signatures)

- MultipleImputationModelling()
- Dataset_Splitter() including
- TCGA_Cancer_Immune_Subtypes()
- TCGA_Gene_Signature()
- Index_of_Sample_When_Split()

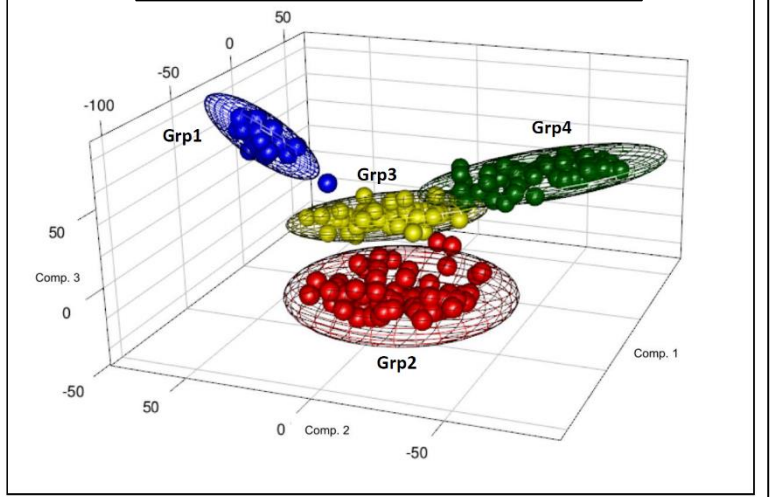
Training/validation dataset

Test dataset
(Unobserved data)

*80% – 20% of each class



220 samples with 17 features



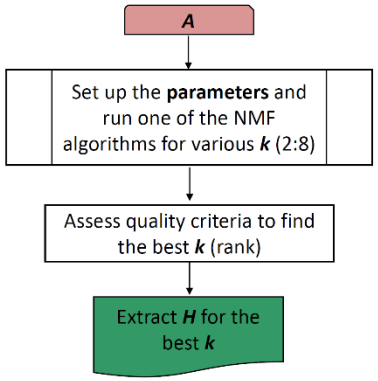
PCA visualisation of groups identified using a consensus NMF clustering

Non-negative matrix factorisation (NMF)

What is the best NMF rank?

Factorising matrix A into two matrices with positive entries

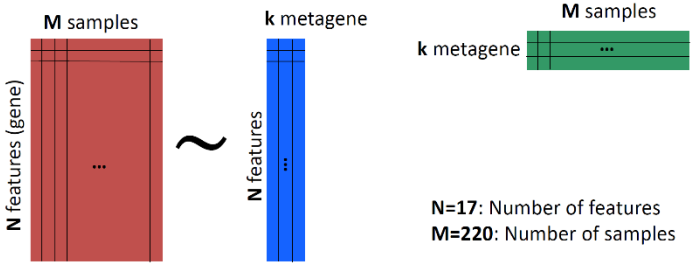
For **any rank k**, the NMF algorithm **groups** the samples into clusters.



$A \sim WH$

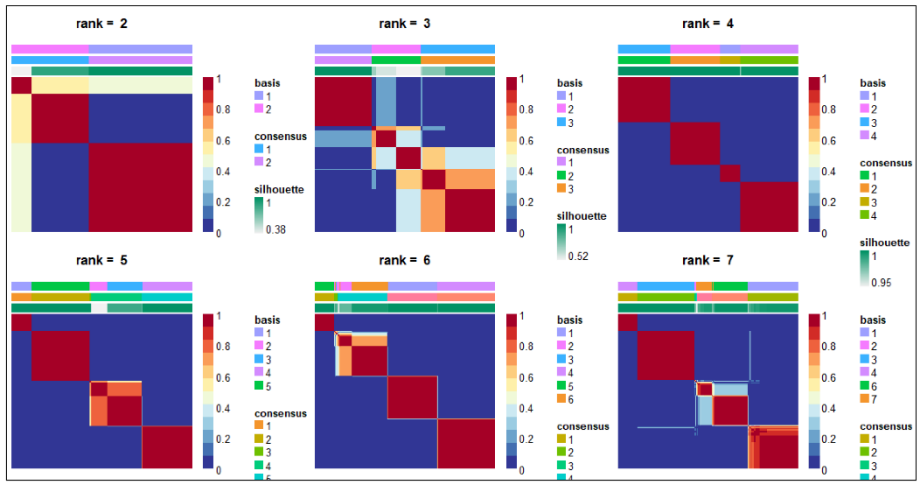
17×220 $\quad 17 \times k \quad k \times 220$

k is the rank of H

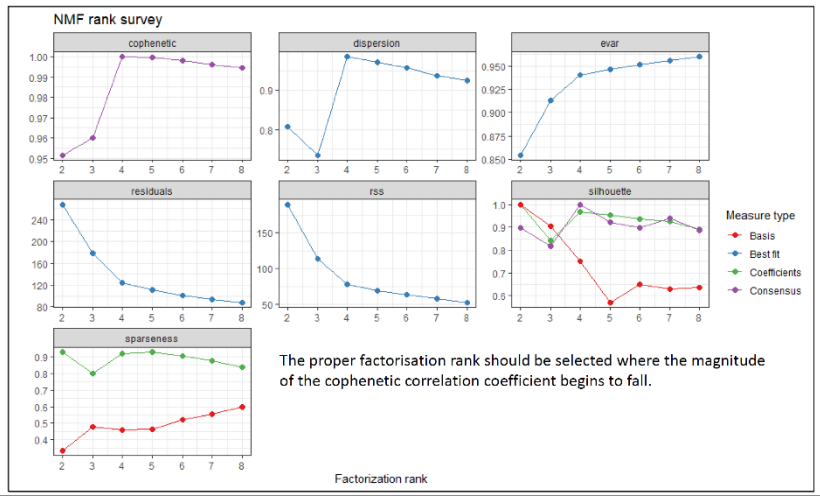


N=17: Number of features
M=220: Number of samples

Consensus matrix for different ranks [2:7]



NMF rank



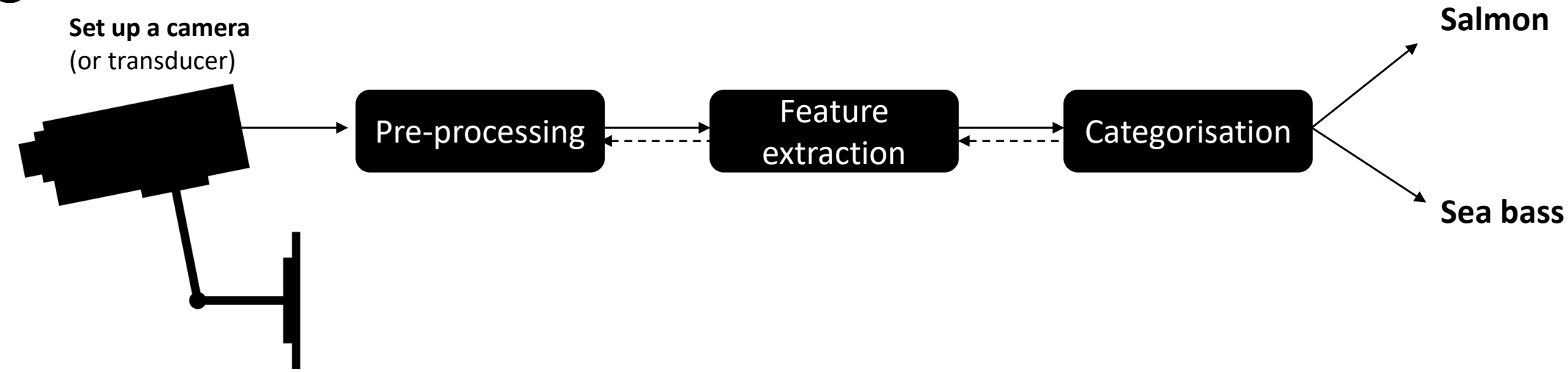
The proper factorisation rank should be selected where the magnitude of the cophenetic correlation coefficient begins to fall.

Pattern recognition systems

Pattern recognition systems

Assume a system: measurement & observation

- A fish packing factory aims to automate the process of sorting incoming fish on a conveyor belt according to species.
- Pilot project: separating sea bass from salmon using optical sensing



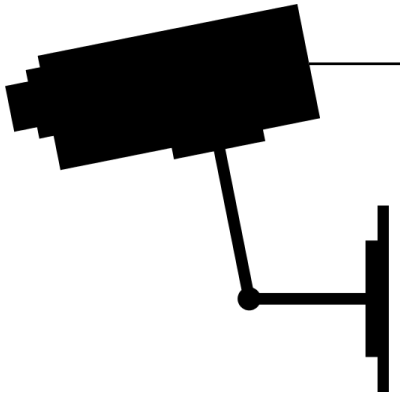
Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

Pattern recognition systems

Assume a system: measurement & observation

- A fish packing factory aims to automate the process of sorting incoming fish on a conveyor belt according to species.
- Pilot project: separating sea bass from salmon using optical sensing

Set up a camera (or transducer)



Pre-processing → Feature extraction → Categorisation → **Salmon** / **Sea bass**

Object(s)

Measurement

Filtering, removing noise and so on

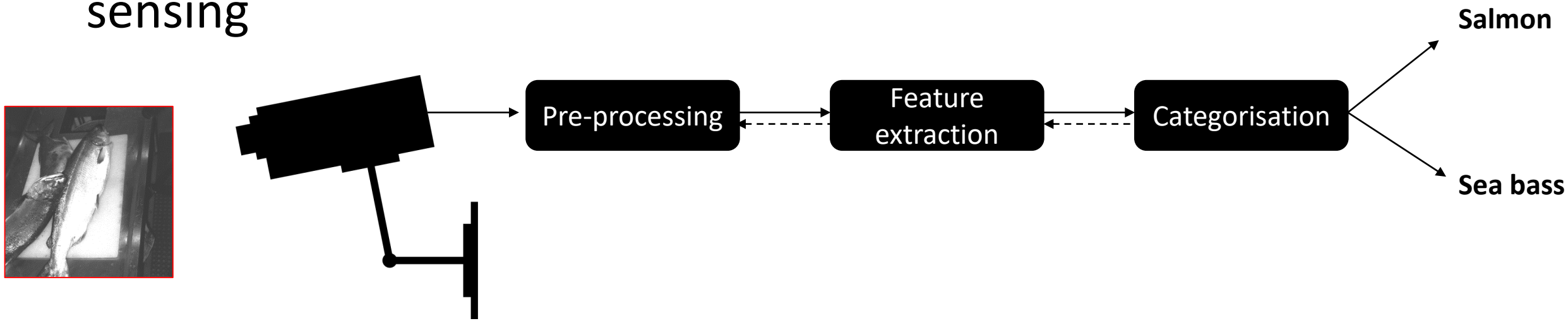
Choosing the best features

Clustering
Classification

Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

Pattern recognition systems

- Pilot project: separating sea bass from salmon using optical sensing

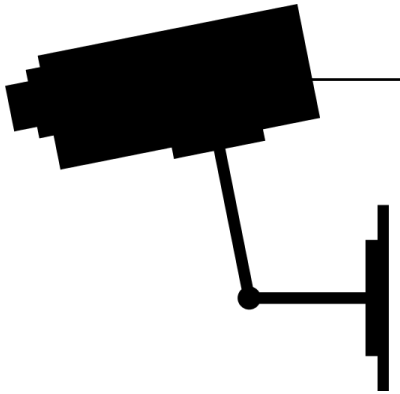


Think and discuss about the three first stages of this pattern recognition system. What would you suggest for selecting features from an image?

Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

Pattern recognition systems

Object(s)



Measurement

1

Pre-processing

Filtering, removing noise and so on

2

Feature extraction

Choosing the best features

3

- One fish per image (using a *Segmentation* technique a single fish extracted)
- No colouring information
- In our measurement using the camera, we could get different parameters of an image object such as the size and the lightness
- We know that **there are two classes** (groups) for each observation/object (i.e., fish): salmon vs. sea bass

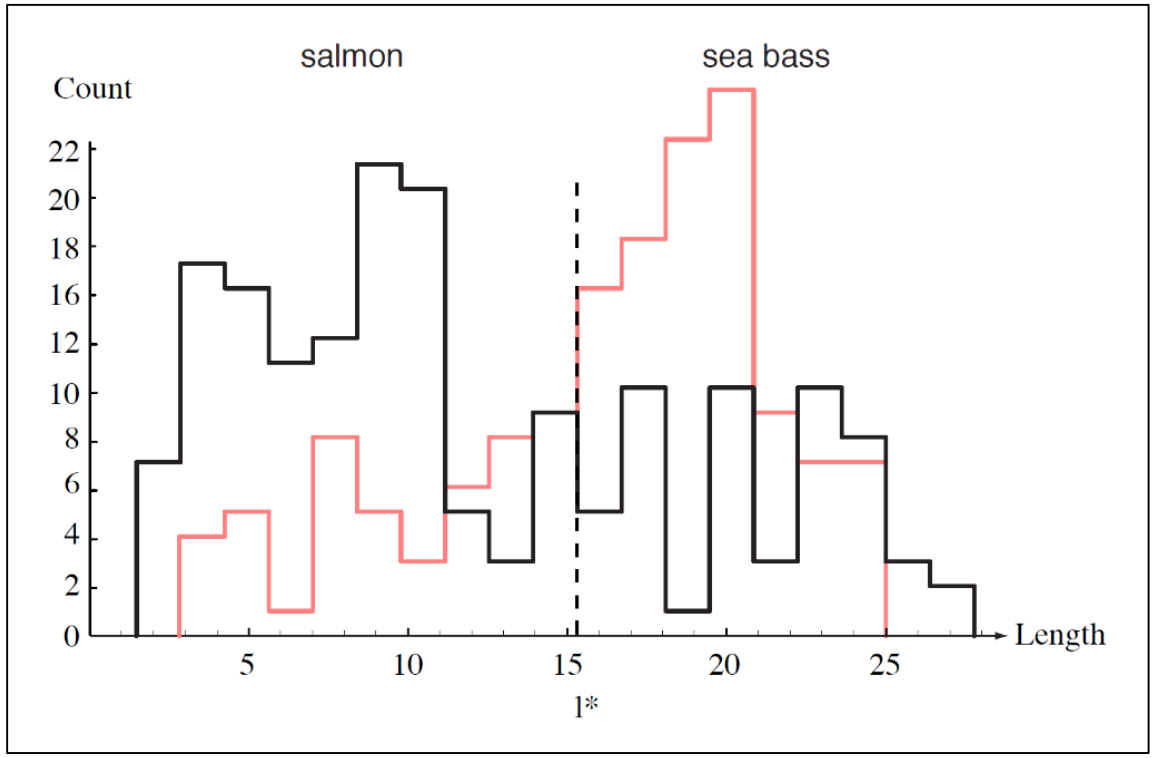
➡ **Supervised classification** or **classification**

Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

How to choose a feature?

No single threshold value of the ***length*** will serve to unambiguously discriminate between the two categories

There would be some errors if we use only ***length*** property as a feature



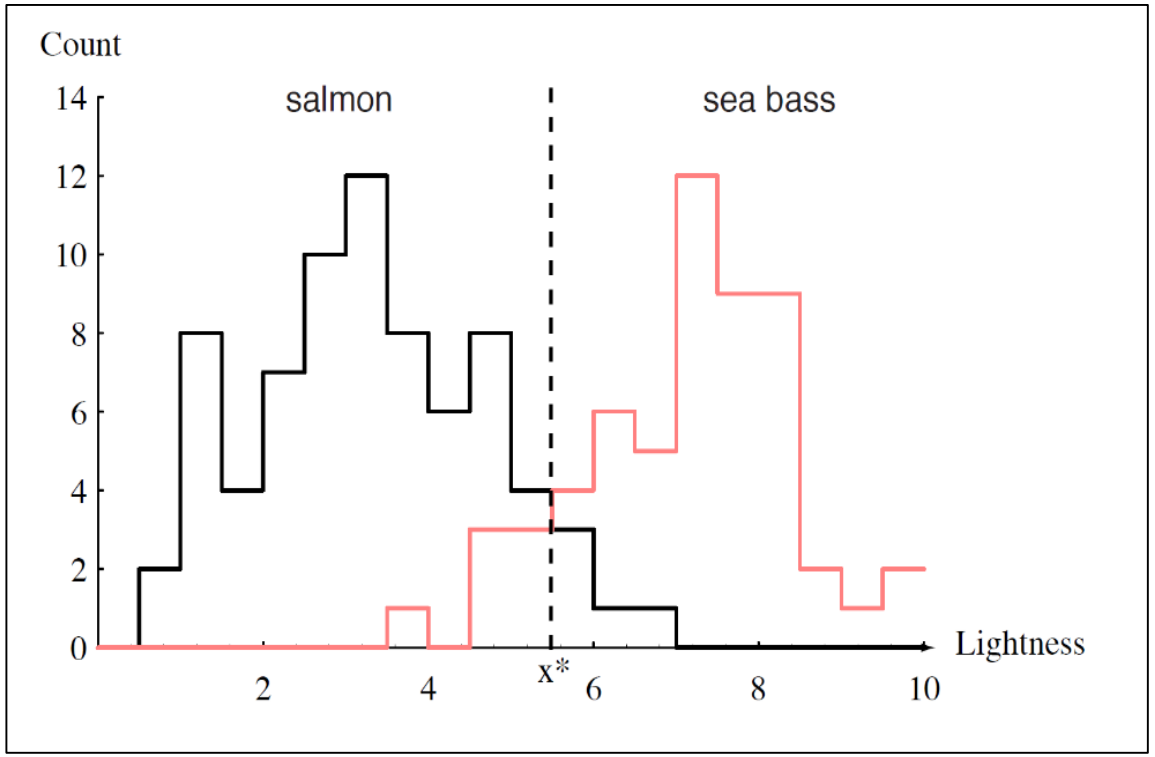
Histograms for the fish length for the two categories

Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

How to choose a feature?

No single threshold value of the ***lightness*** will serve to unambiguously discriminate between the two categories

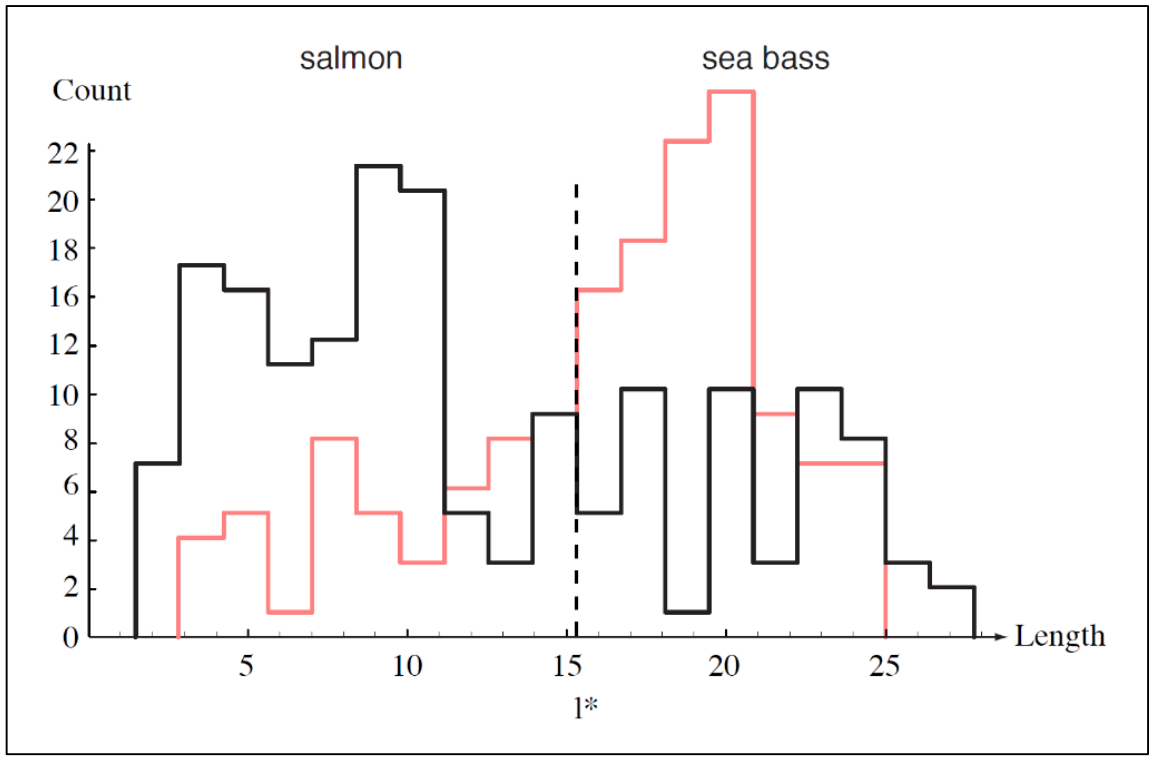
There would be some errors if we use only ***lightness*** property as a feature



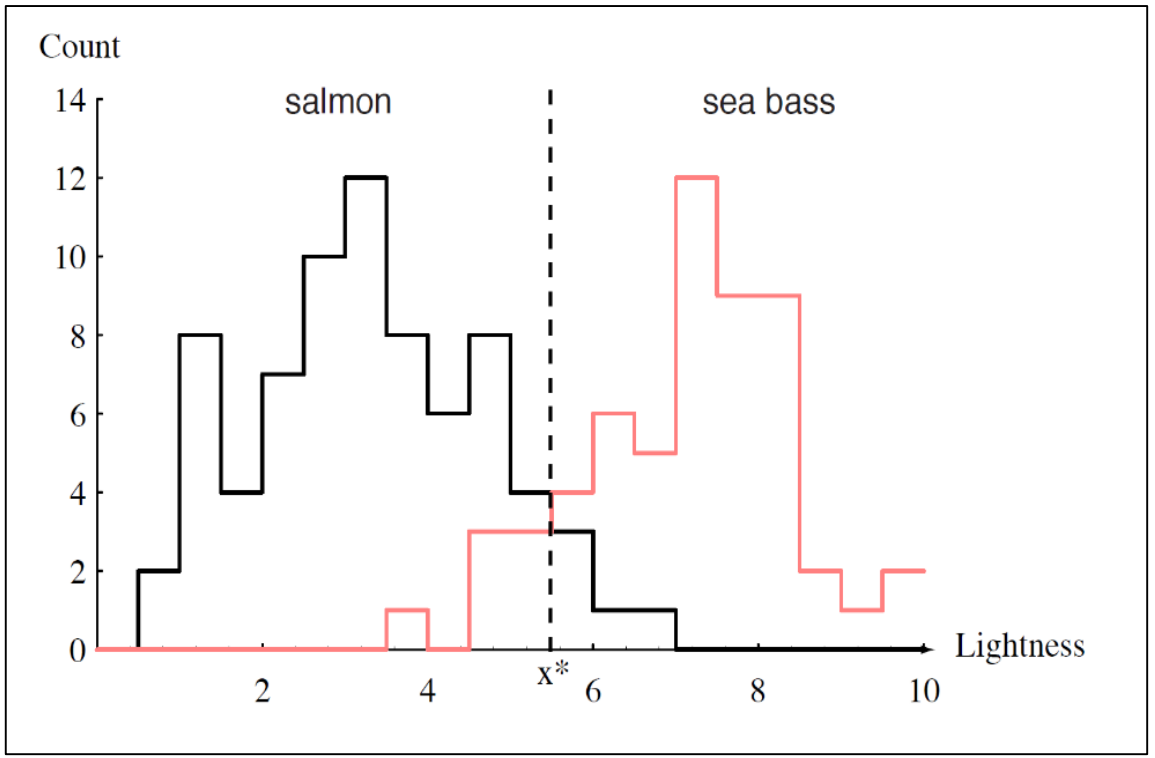
Histograms for the fish lightness for the two categories

Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

How to choose a feature?



Histograms for the fish length for the two categories

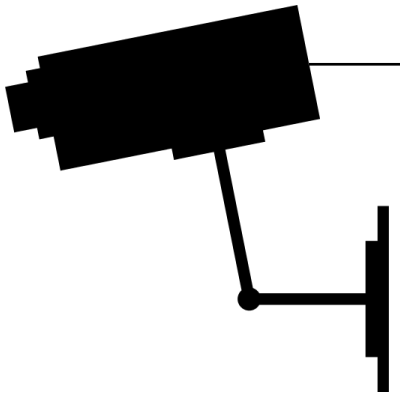


Histograms for the fish lightness for the two categories

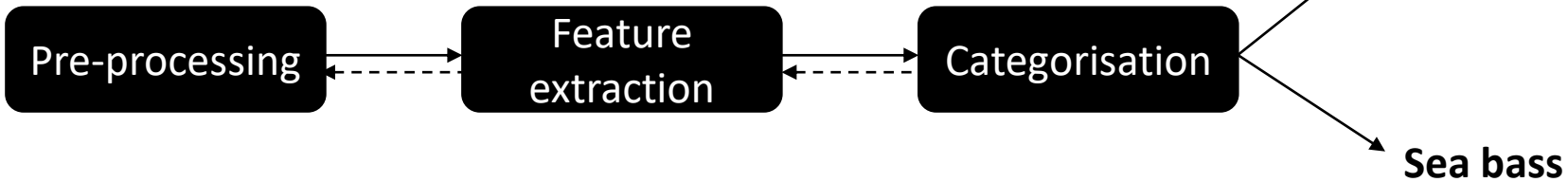
Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

Pattern recognition systems

Object(s) (samples)



Measurement



Filtering, removing noise and so on

Choosing the best features

Clustering
Classification

x_1: **lightness**
x_2: **width**

The aim is to partition the feature space into two regions

Feature space: two dimensions $X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

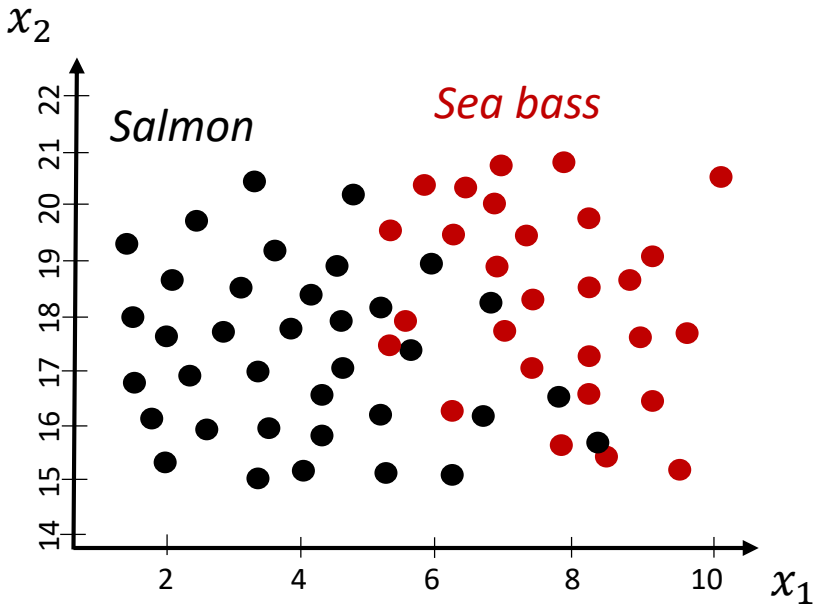
Suppose that we measure the feature vectors for our samples

Reference: Pattern Classification, R. O. Duda, P. E. Hart, D. G. Stork; Wiley

Feature space; lightness & width

x_1: lightness
x_2: width

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

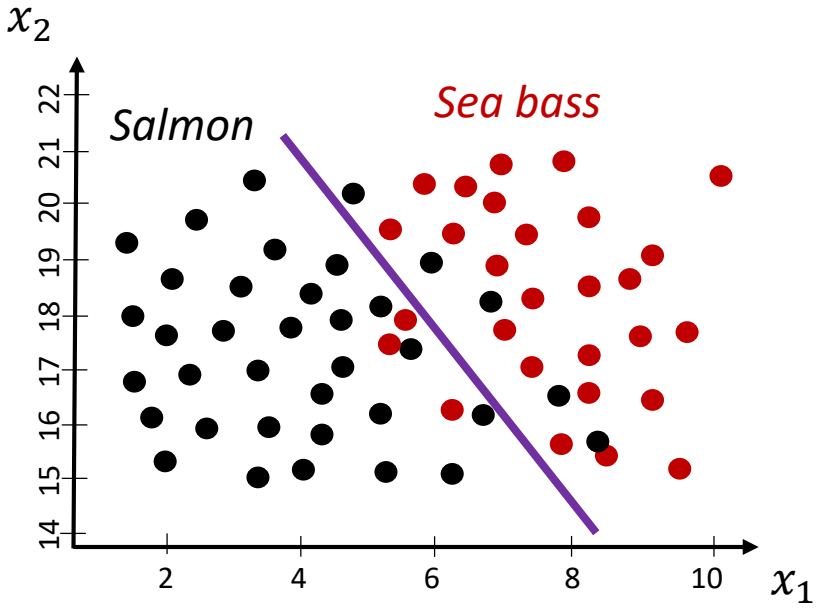


Decision boundary (line)

x_1: lightness
x_2: width

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Linear decision boundary



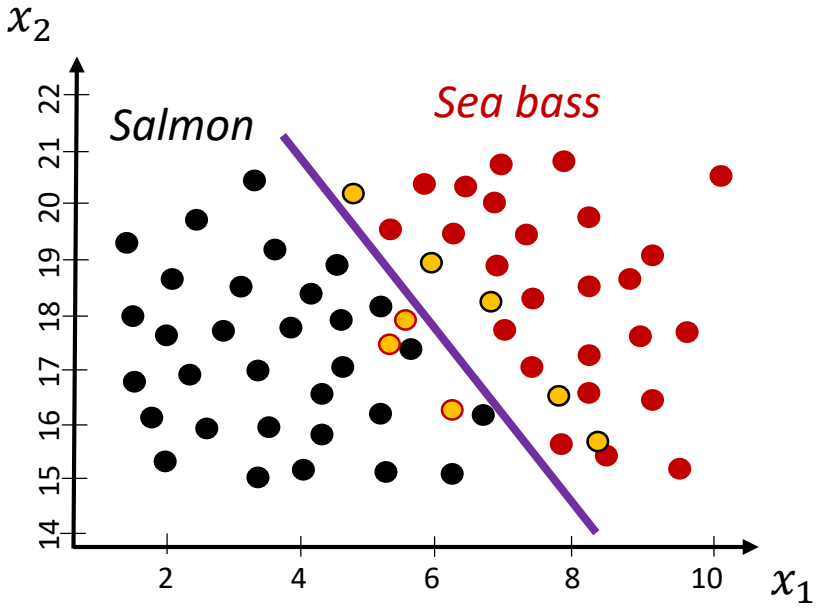
This plot suggests the following rule for categorising (separating) a fish:
Classify a fish as *salmon* if the feature vector of this fish **falls below** the line (this line is called decision boundary)

Classification error

x_1: lightness
x_2: width

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Linear decision boundary



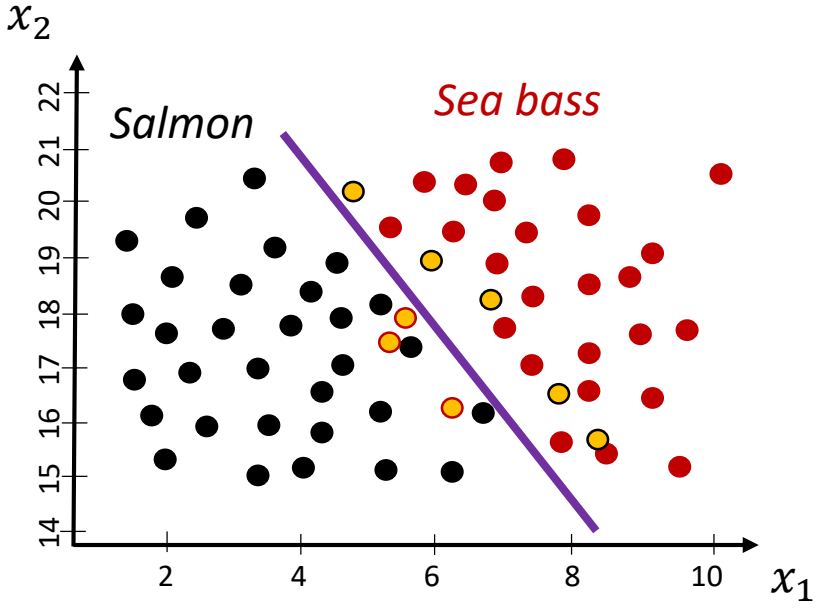
This plot suggests the following rule for categorising (separating) a fish:
Classify a fish as *sea bass* if the feature vector of this fish **falls above** the line (this line is called decision boundary)

How to reduce the classification error?

x_1: lightness
x_2: width

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Linear decision boundary



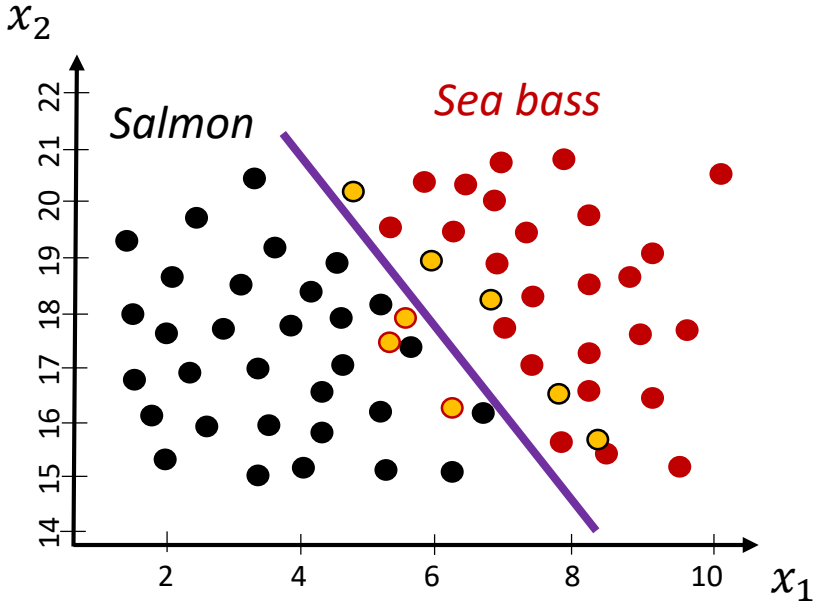
Any suggestions to reduce the classification error (i.e., to improve the accuracy of the classification)?

How to reduce the classification error?

x_1: lightness
x_2: width

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Linear decision boundary



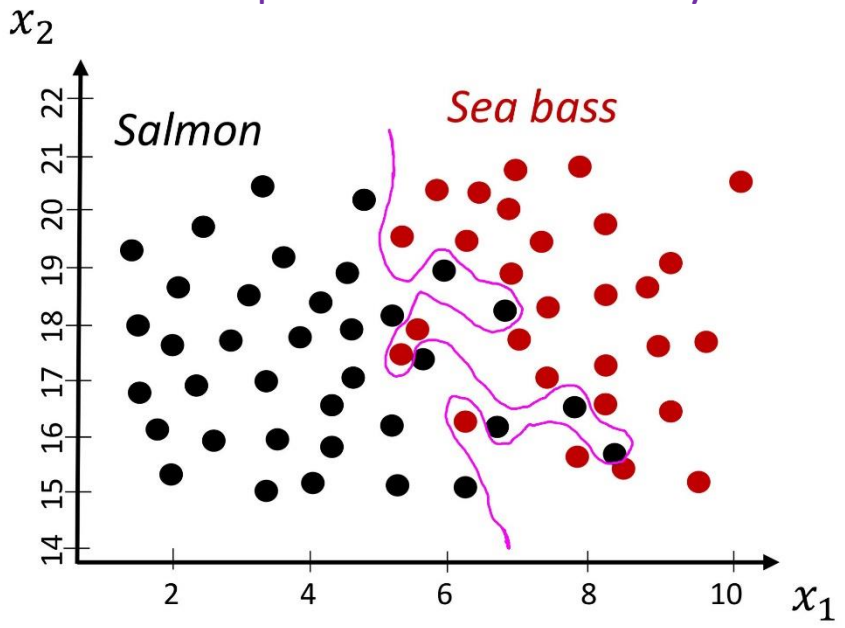
- **Include extra features such as the shape parameters of the fish**
  - E.g., the placement of the eyes (as expressed as a proportion of the mouth-to-tail distance)
  - Some features might be redundant
- **Choose a non-linear decision boundary instead of using a simple straight line!?**

How to reduce the classification error?

x_1: lightness
x_2: width

$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

Complex decision boundary



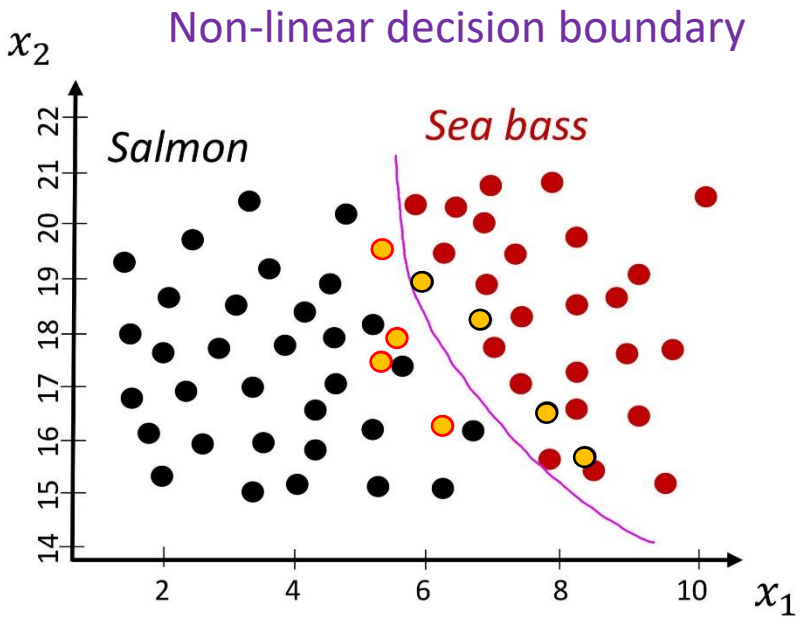
- **Include extra features such as the shape parameters of the fish**
  - E.g., the placement of the eyes (as expressed as a proportion of the mouth-to-tail distance)
  - Some features might be redundant
- **Choose a complex or non-linear decision boundary instead of using a simple straight line!?**

There is an issue of ***generalisation*** when we are using a complex decision boundary to perfectly separate the objects

How to reduce the classification error?

x_1: lightness
x_2: width

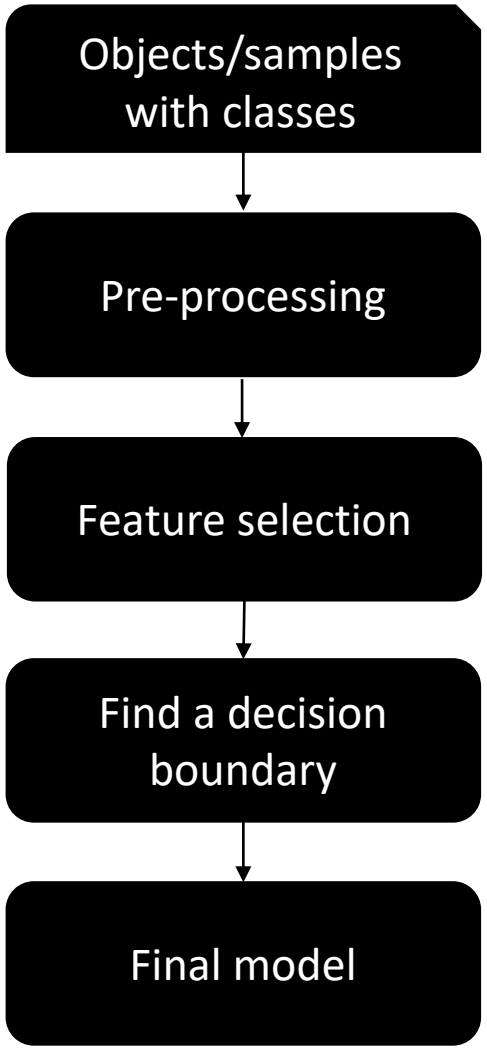
$$X = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$



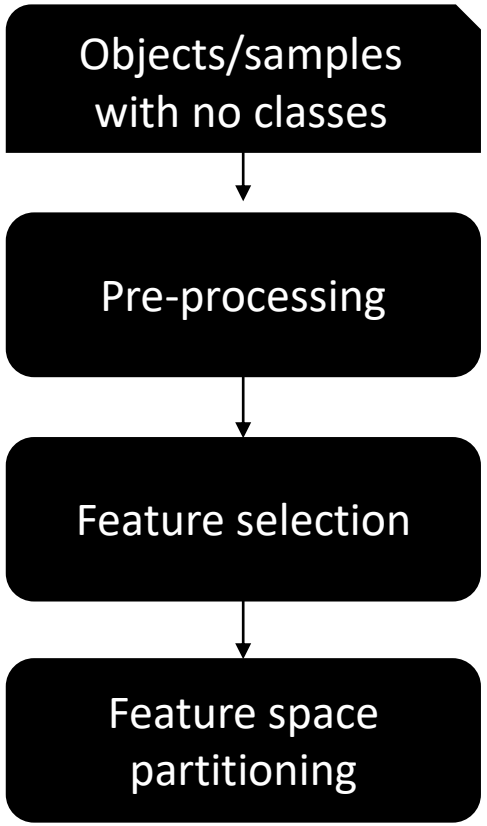
- **Include extra features such as the shape parameters of the fish**
  - E.g., the placement of the eyes (as expressed as a proportion of the mouth-to-tail distance)
  - Some features might be redundant
- **Choose a complex or non-linear decision boundary instead of using a simple straight line!?**

There is an issue of ***generalisation*** when we are using a complex decision boundary to perfectly separate the objects

Classification vs. clustering



It's called model training or classifier training stage

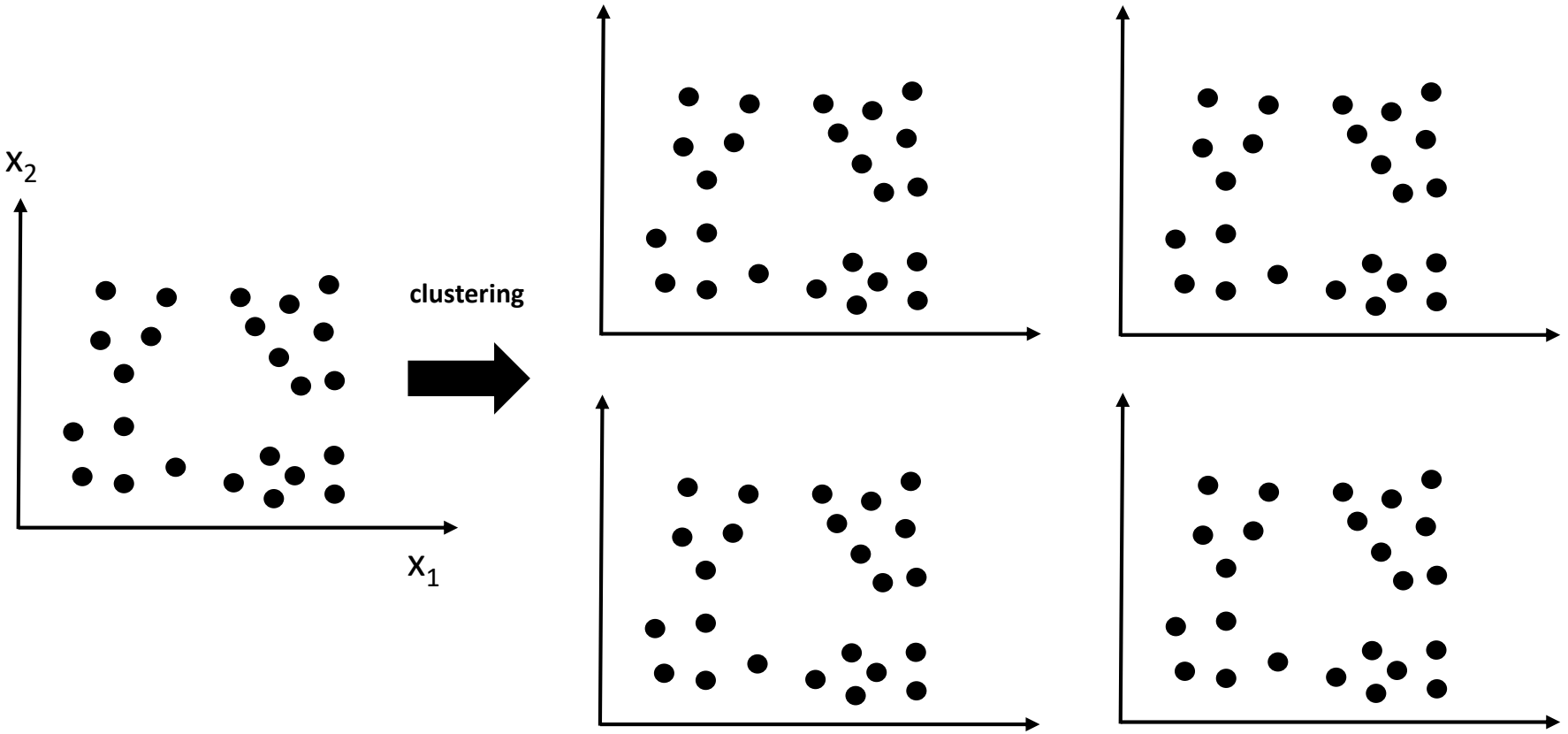


Feature selection for high separability

Unsupervised learning

What is clustering?

Clustering concept

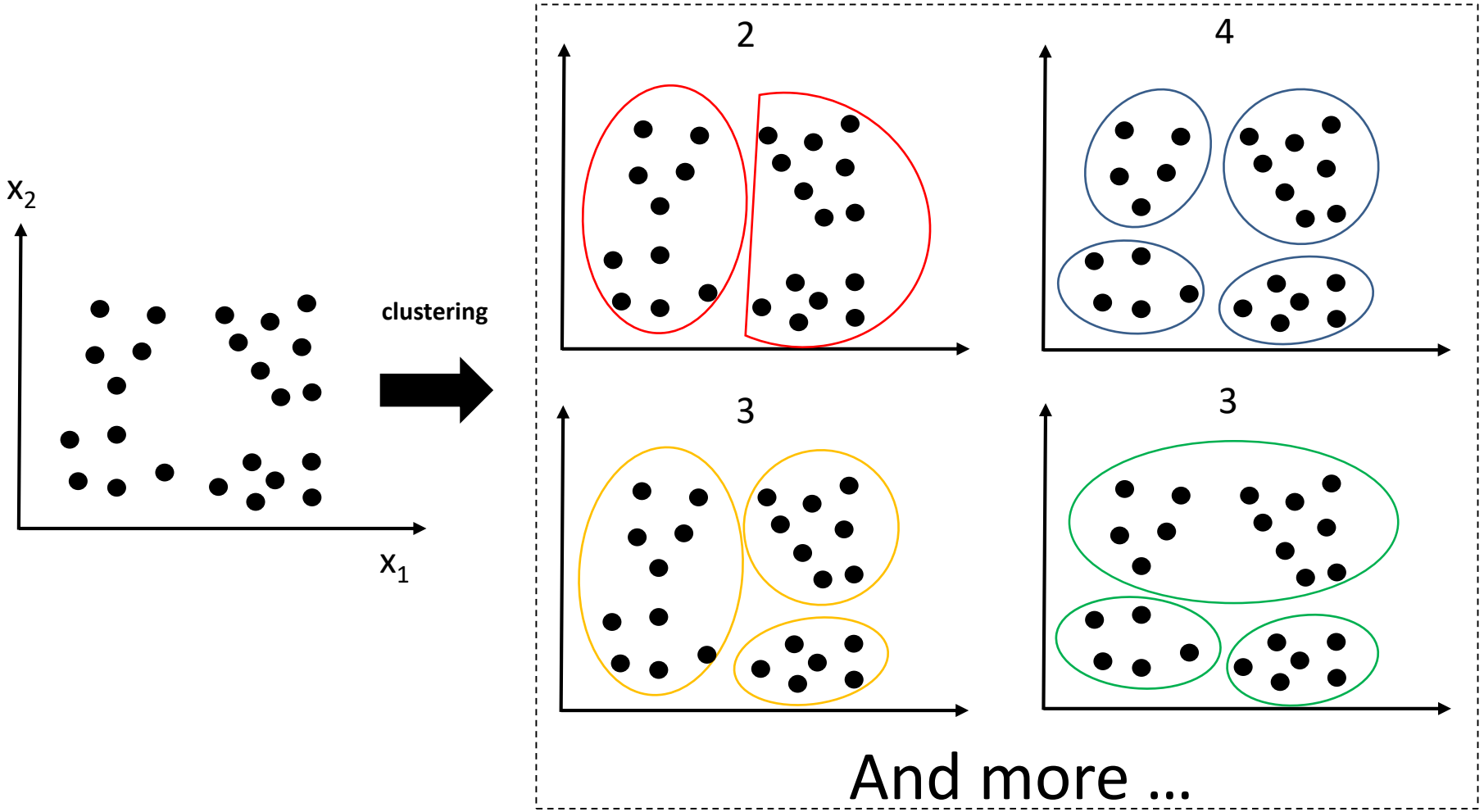


And more …

In case of applying an appropriate clustering method (with well-adjusted parameters/initialisation, bootstrapping and cross-validation techniques), we could have distinct groups (with possibly different number of clusters) but they might be meaningless!

What is clustering?

The aim of clustering is to group objects into meaningful groups/classes



In case of applying an appropriate clustering method (with well-adjusted parameters/initialisation, bootstrapping and cross-validation techniques), we could have distinct groups (with possibly different number of clusters) but they might be meaningless!

Any Questions?