MTU
Ollscoil Teicneolaíochta na Mumhan
Munster Technological University

Machine Learning

Lecture 6: Feature extraction

What is a feature?

- Definition:
  A **feature** is an individual measurable property or characteristic of a phenomenon being observed

  - It is NOT necessary the raw data being measured
  - Domain knowledge (e.g. medical, linguistic, audio, photogrammetry, etc.) can go a long way in determining the right features to extract for a given application
  - Therefore, this step is where domain experts should be involved in a machine learning project

  - Selecting and extracting the “right” features can determine if a machine learning problem is difficult or easy to solve

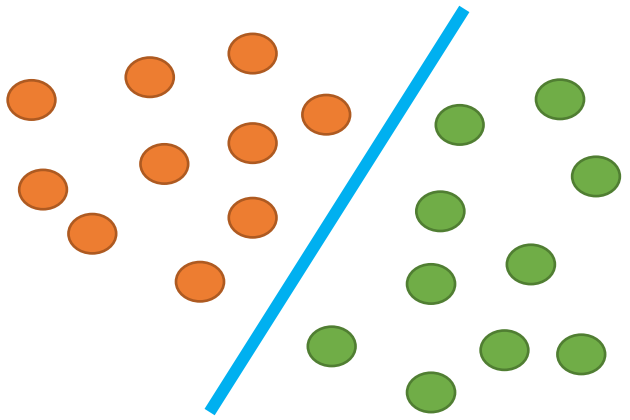
What is a feature?

- Definition:
  A **feature** is an individual measurable property or characteristic of a phenomenon being observed

  - Typically it is encoded in a numerical **feature vector** representing a single object
  - The vector space associated with these vectors is call the **feature space**

Feature space

- Every object (**feature vector**) is a point in **feature space**
- The feature space should be designed so that
  - Objects with the same label should be close together in feature space
  - Object with different labels should be far apart in feature space
  - In particular: The notion of “distance” has to make sense in feature space for this to be meaningful
- The surface separating objects with different labels in feature space is called a **decision surface**
- Decision surfaces should be “simple”



Example: Video processing

- A 4K video generates 24 frames of 3840 × 2160 pixels every second, with every pixel being a 3 dimensional (R,G,B) colour information
- The raw data generated in a 10 second 4K video is a 6 billion dimensional vector

- Likely a lot of this information is redundant
- Also, distinguishing objects using Euclidean distances in 6 billion dimensions is very difficult

- Therefore, this raw data is not very suitable as input for most machine learning algorithm
- Image processing algorithms can reduce the amount of data to be processed

Feature extraction

- Given a high dimensional raw data vector (e.g. a video) there are two fundamental problems to be addressed:
  - Find the locations inside the raw data vector where there is interesting information to be extracted
  - Create a meaningful descriptor of this localised region in the raw data to extract the feature seen there

Convolution filters

- An approach solving both of these issues simultaneously are convolution filters

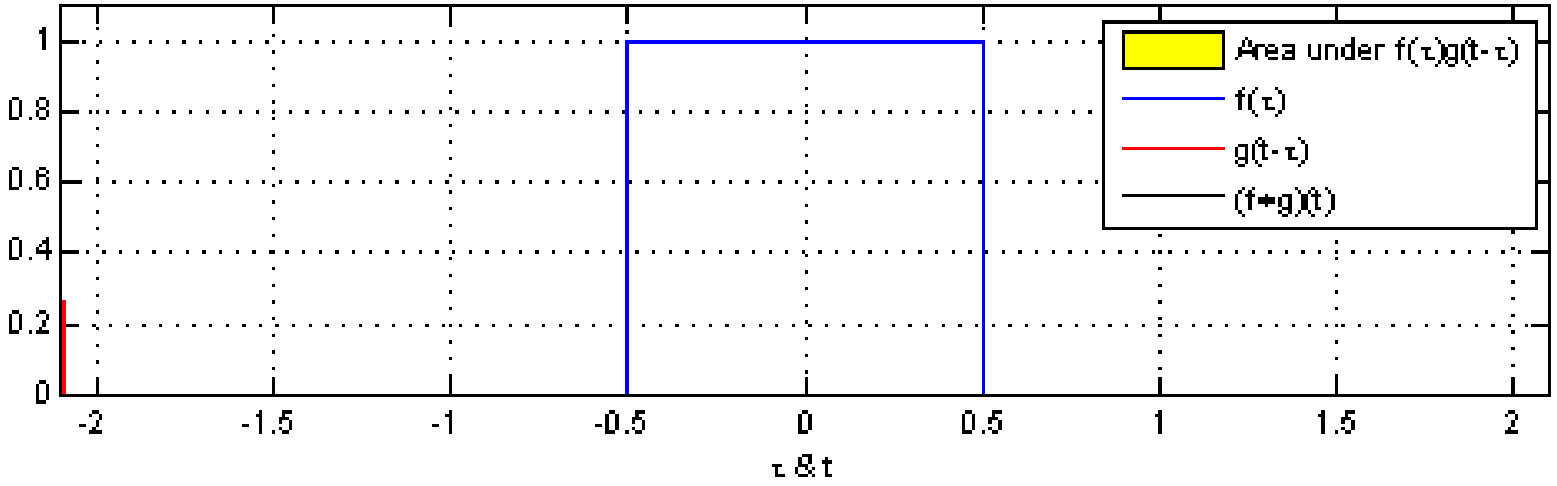
- In signal processing the convolution of a **signal** function $f[x]$ with a **kernel** $\omega[x]$ is defined as

$$f * \omega = \int_t f[t]\omega[x-t]dt$$

- Intuitively, this tells us how similar the signal function looks to the kernel and where
- How does this help?

Convolution filters

- The kernel is essentially a mask, that is shifted over the signal and creates a strong response where the signal looks like the mask



Convolution filters

- Relation between convolution filters and scale:

- A Gaussian kernel creates a smoothed (blurred) version of the original function

$$\int_t f[t] \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-t)^2}{2\sigma^2}} dt$$

- Selecting the right scale σ^2 on which a feature is visible is often crucial for feature extraction
- Convolution is associative, i.e.

$$(f * G_{\sigma^2}) * \omega = f * (G_{\sigma^2} * \omega)$$

- Scale selection is therefore by applying a Gaussian filter to the kernel pattern we are interested in

Convolution filters

- Relation between convolution filters and derivatives:

- Remember: the derivative of a function is defined as

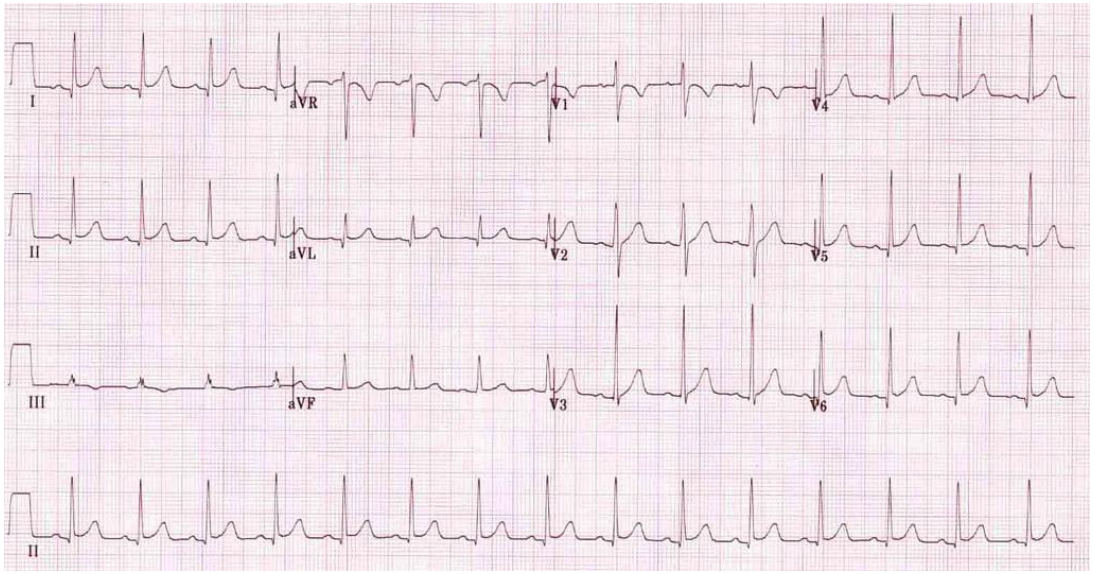
$$f'[x] = \lim_{h \to 0} \frac{f[x+h] - f[x]}{h}$$

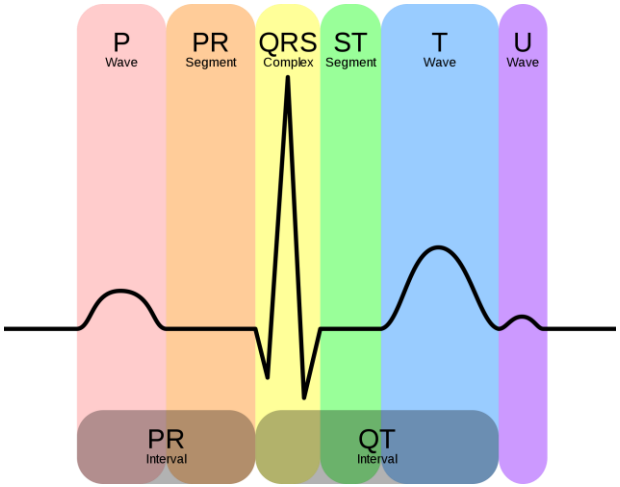
$$f''[x] = \lim_{h \to 0} \frac{f[x+h] - 2f[x] + f[x-h]}{h^2}$$

- Computing a discrete version of a derivative can be achieved with a convolution kernel [-1, 1]
- A discrete second derivative is computed with a convolution kernel [1, -2, 1]

Example: medical data

- Medical science has developed a good understanding of different feature masks in the electrocardiogram (ECG)
- This domain knowledge can be used explicitly to generate good features using convolution filter masks



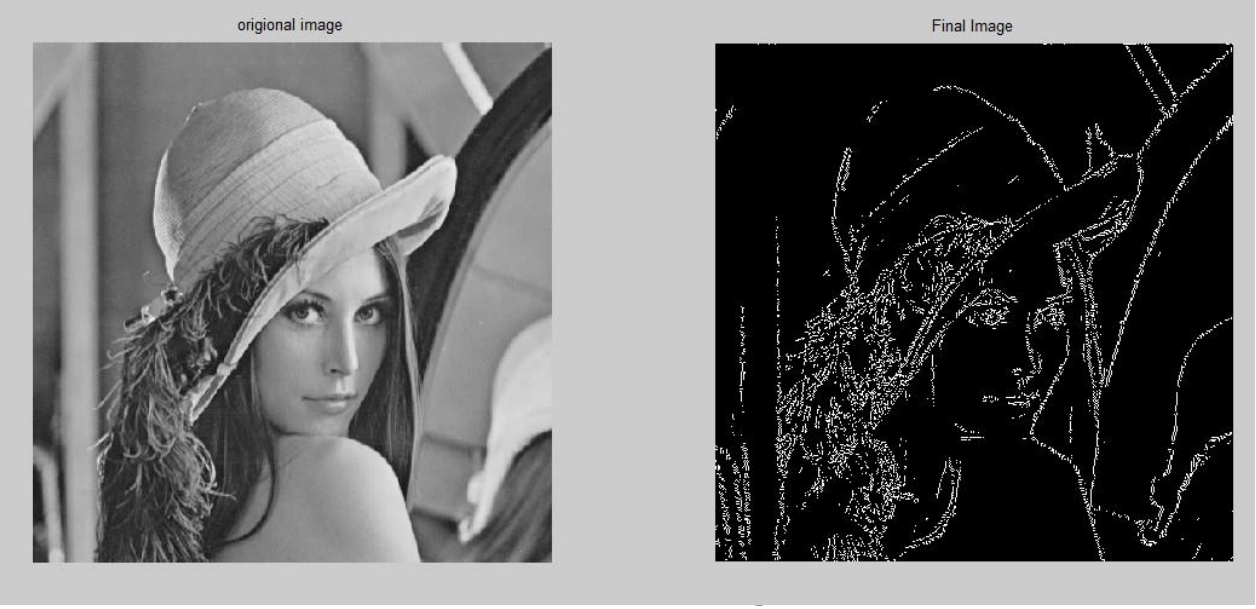


Convolution filters

- Convolution also works with 2D signals $f[x, y]$, i.e. images, and kernels $\omega[x, y]$ and is defined as

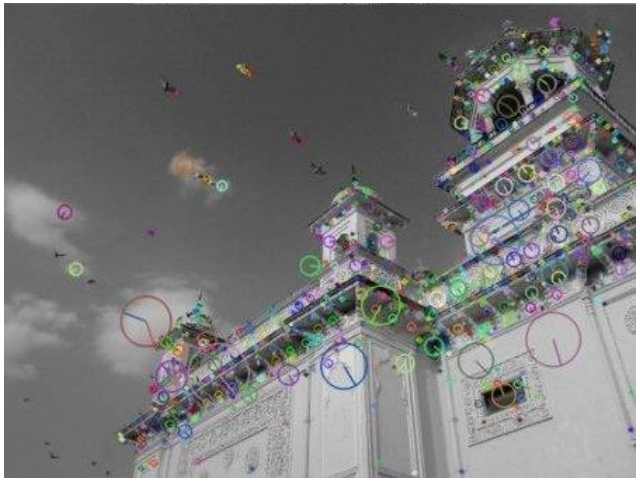
$$f * \omega = \int_t \int_s f[s, t]\omega[x - s, y - t]ds\, dt$$

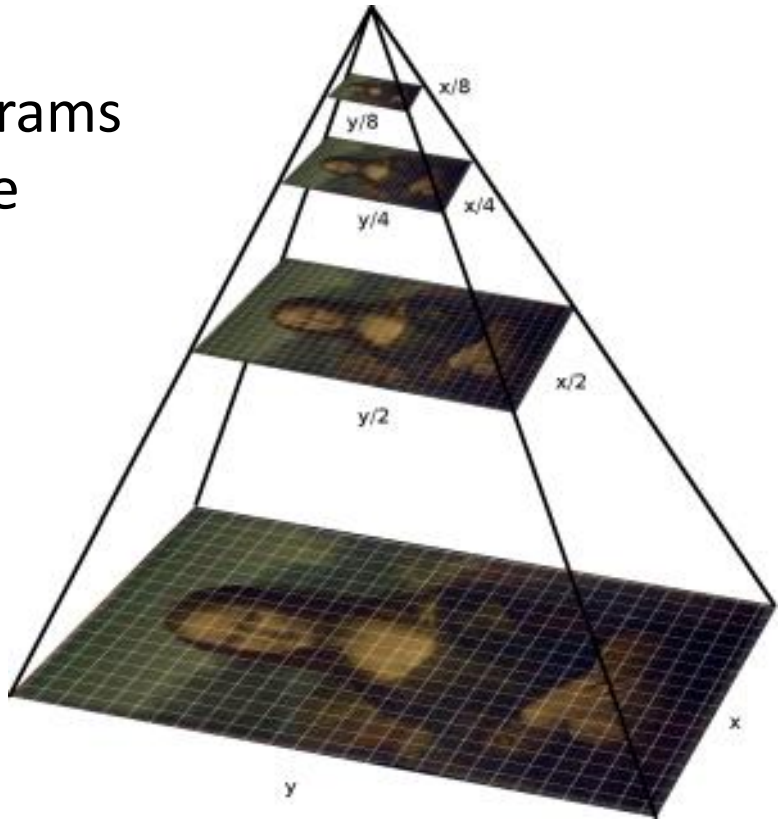
- A gradient filter finds the edges in the image



Example: Image processing

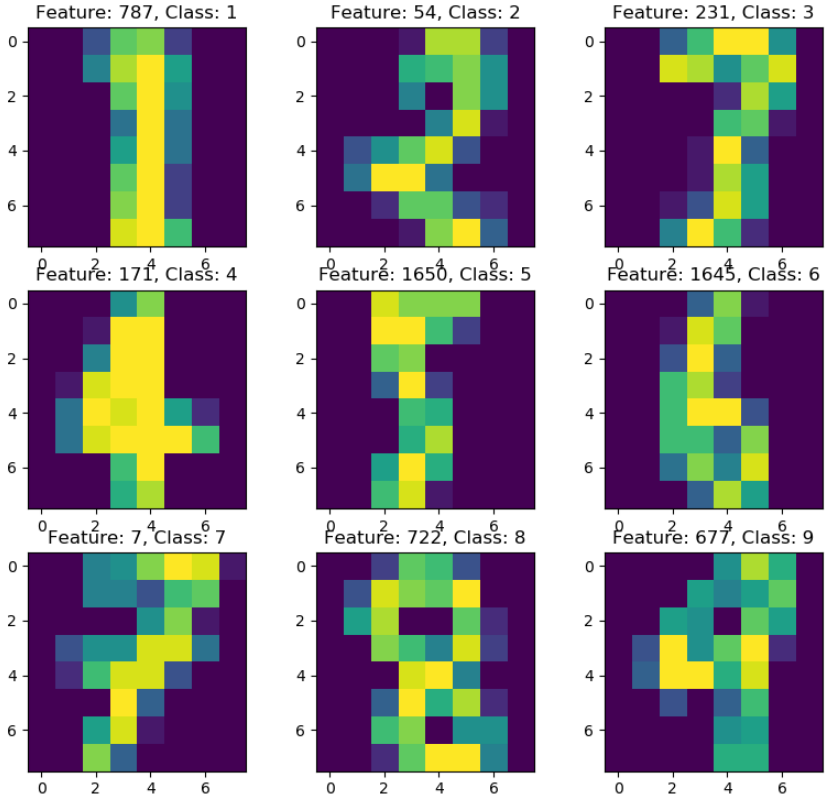
- Scale-Invariant Feature Transformation (SIFT)
  - An Image Pyramid considers the image at different sizes (scales)
  - “Interesting structures” can be found where the gradient is non-zero
  - Local normalised image histograms around these features describe the structure





Example: Optical Character Recognition

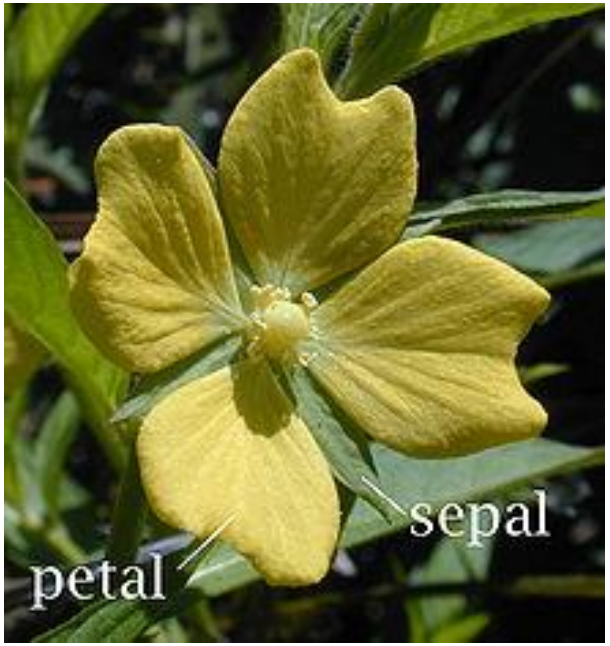
- Low resolution normalised images, adjusted for orientation and size to exactly fit the written character



Example: Botanical classification

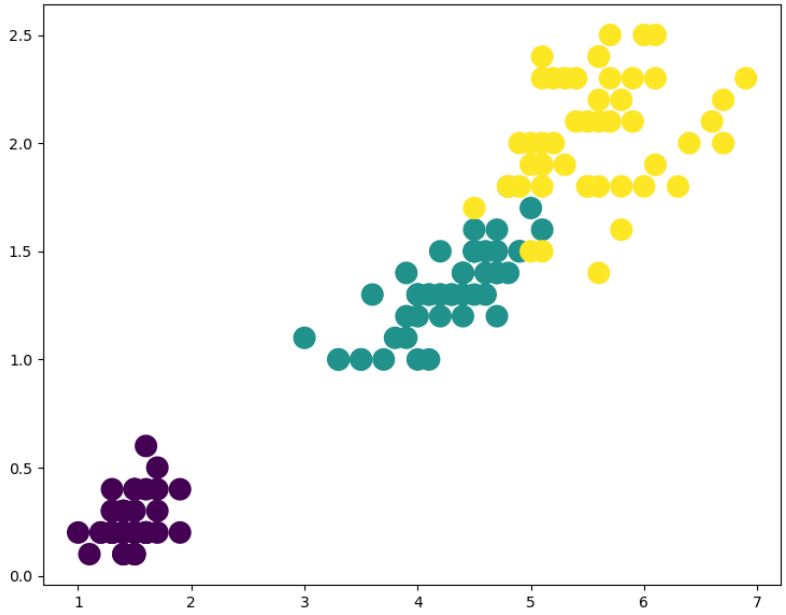
- We have already seen the Iris dataset

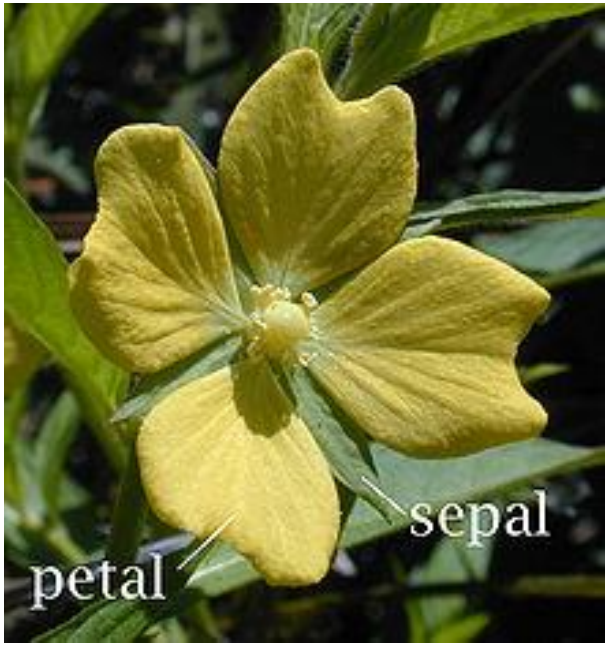
- In this case a domain expert decided that these four features are best suited to distinguish the different species of plants:
  - sepal length in cm
  - sepal width in cm
  - petal length in cm
  - petal width in cm



Example: Botanical classification

- We already found the petal length and petal width to be highly correlated
- Sometimes, simply removing a redundant feature improves the performance of a ML algorithm





Example: Document Classification

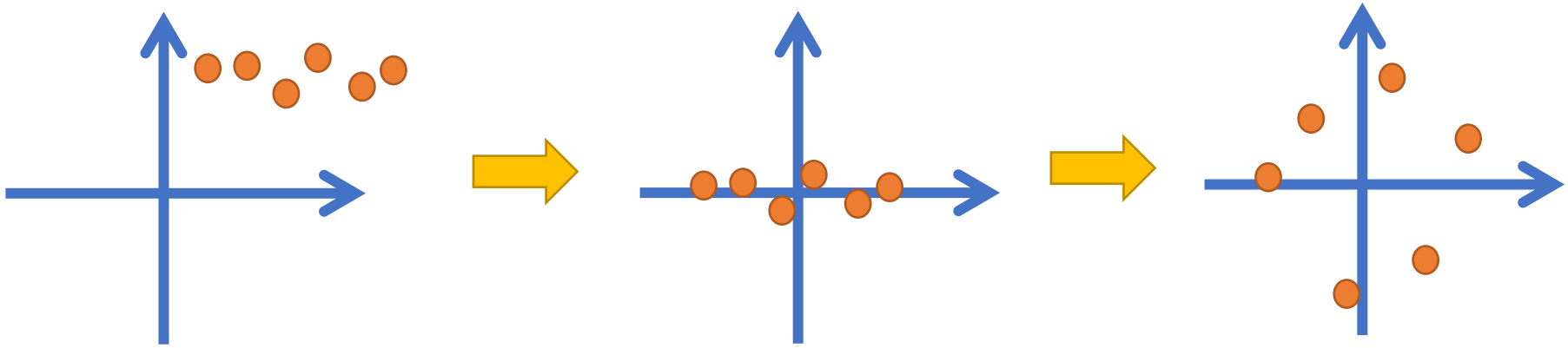
- In document classification **word occurrence** can be used as feature

- There are two commonly used approaches for counting words:
  - (Bernoulli model) **Set of words**, counts the number of documents where a word occurs
  - (Multinomial Model) **Bag of words**, counts the total occurrences of a a word across all documents.

- The Bernoulli model uses **binary occurrence** information, ignoring the number of occurrences of a word in a document , whereas the multinomial model keeps track of multiple occurrences in a single document.
- The models also differ in how **non-occurring terms** are used in classification. They do not affect the classification decision in the multinomial model; but in the Bernoulli model the probability of non-occurrence is factored in

Normalisation

- To make “distances” in feature space more meaningful it is useful if the objects of interest fill the feature space evenly
- Subtracting the mean or median from all features helps to centre the features around the origin and have a good distribution of positive and negative values
- Dividing all features by their standard deviation or maximum value helps to avoid differing scales between the features and have them all evenly distributed in the unit sphere around the origin



Noise / outliers / bias

- Typically we consider our data, and therefore our extracted features, to be a combination of a **signal** and a **noise** component
- The signal represents the “truth” generated by some underlying (potentially random and unknown) process
- Learning this “true” underlying process from observed input data is the goal of machine learning

Noise / outliers / bias

- A **bias** is a systematic error that is perturbing the distribution of datapoints

- It can be introduced by
  - the measurement technology
  - artefacts of the feature extraction process
  - the sampling method used for obtaining datapoints

- Making sure the training data is bias-free is an important aspect of engineering a machine learning application

Noise / outliers / bias

- In contrast to the bias, random errors and uncertainties of the process generate **noise**
- Bias-free noise is called **white noise**

- We consider a feature to be generated by
  1. A “true” value is sampled from a distribution determined by the underlying process
  2. The value is the perturbed by random noise

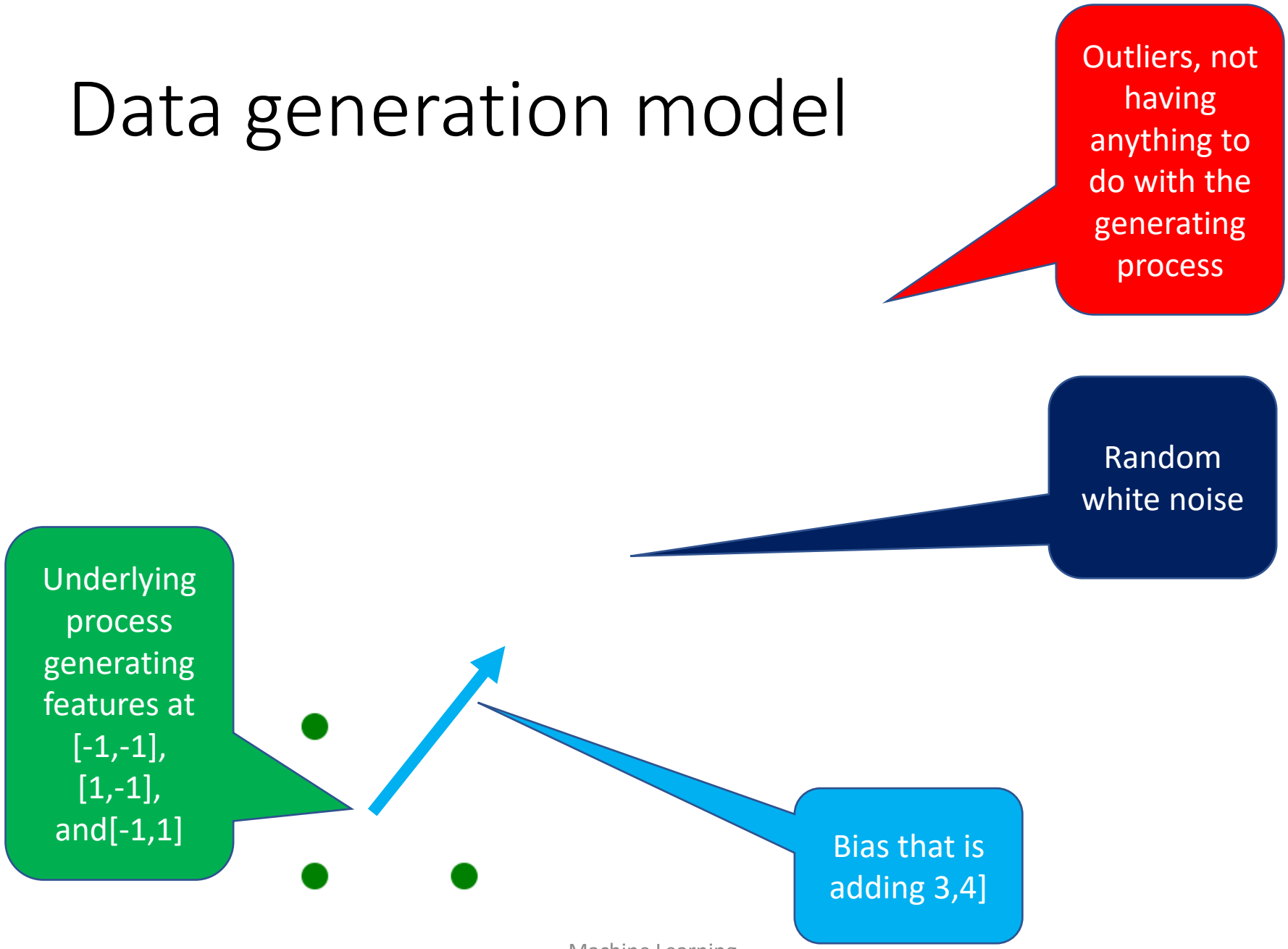
- Noise is introduced by the uncertainty of the measurement process
- Our algorithms should be able to cope with noise and extract the underlying “true” process regardless

Noise / outliers / bias

- An **outlier** is a data point that cannot be explained by the underlying signal generating process, the bias, or the noise process

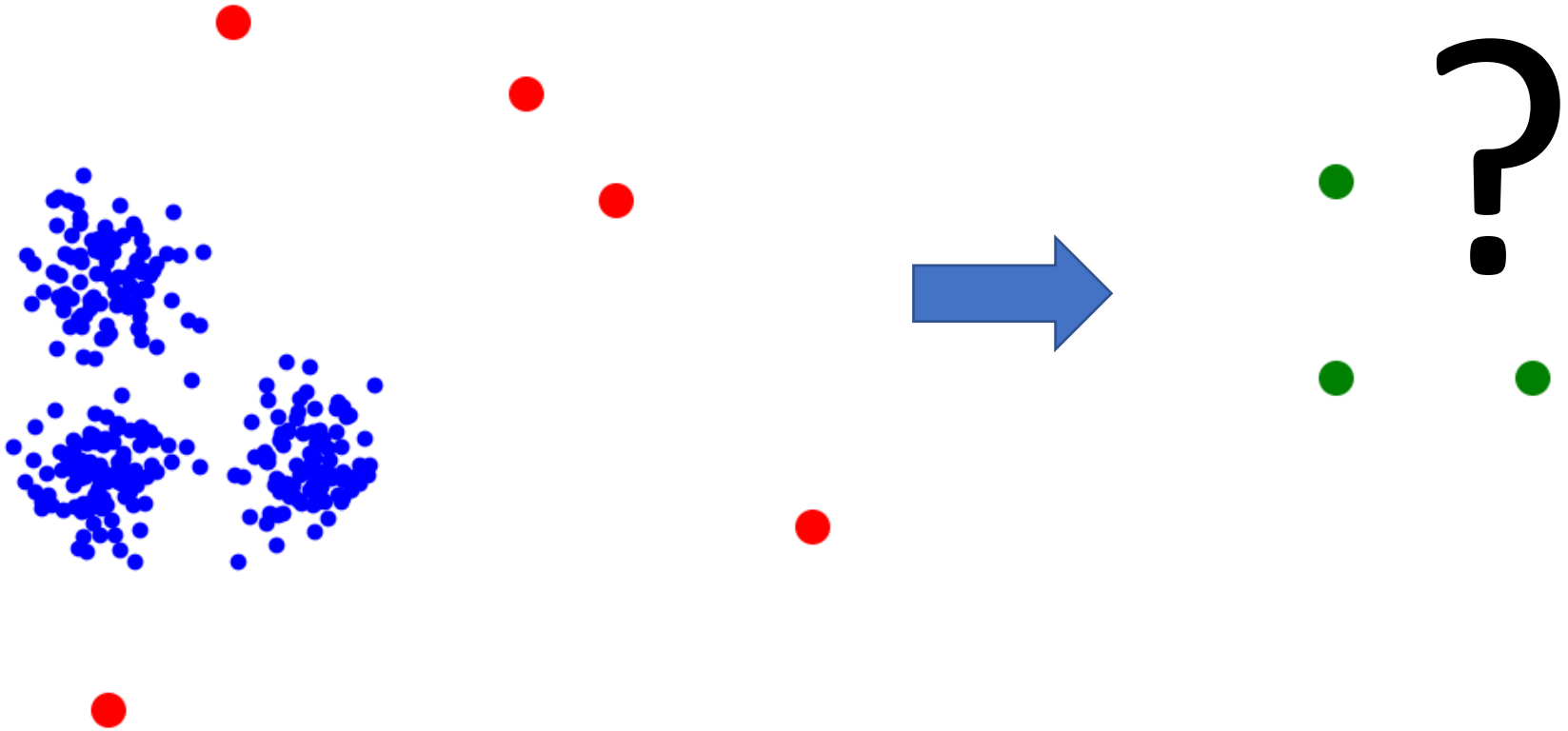
- Outliers typically occur because of
  - Mistakes or wrongly input training data
  - Very unusual samples in the training data combined with small overall sample size
  - In case the input is already the output of some other algorithms, which might have misclassified or wrongly detected a feature

Data generation model



Inferring the structure

The goal of ML is to reverse this process and infer the underlying structure that generated the features, even if our data contains noise and outliers



Outlier removal: RANSAC

- While white noise is unavoidable and not a big issue, removing outliers prior to processing is usually important

- Sanity checks on the plausibility of the input data can help

- Alternatively, methods such as RANdom SAmple Consensus can be applied
- The idea of the RANSAC algorithm is to select a random (small) subset of samples from the input and verify that the other datapoints are consistent
- The subset with the largest consensus is then used to eliminate datapoint which do not fit that model

Missing values

- Missing values in the training data can be a problem, not only for the classifier but also for feature extraction, outlier removal and normalisation
- Sometimes it makes sense to simply drop a datapoint that contains a missing value in any feature dimension
- It is also possible to substitute missing values by median, mean, or localised averages of the surrounding datapoints (or any other data model that we might have)
- This creates artificial correlations in the data set, so we need to be very careful with this

Thank you for your attention