Pattern Recognition: Week 1

Introduction to Pattern Recognition

John Quinn

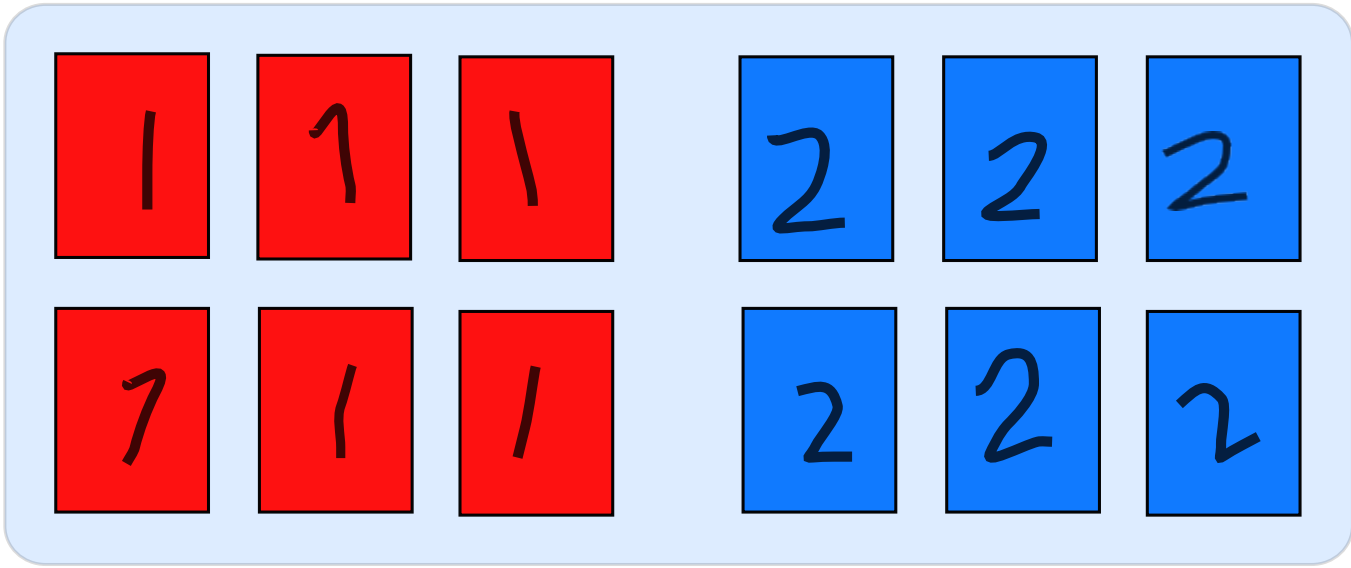
February 19, 2008

Patterns

- Patterns are **regularities** in data.
- Humans are good at seeing patterns, even when they don't really exist.
- There are many situations where we need automatic identification of patterns: in medicine, finance, robotics...
- In this course we will look at common techniques for recognising patterns. There will be a mixture of theory and practical programming excercises.

Handwriting recognition

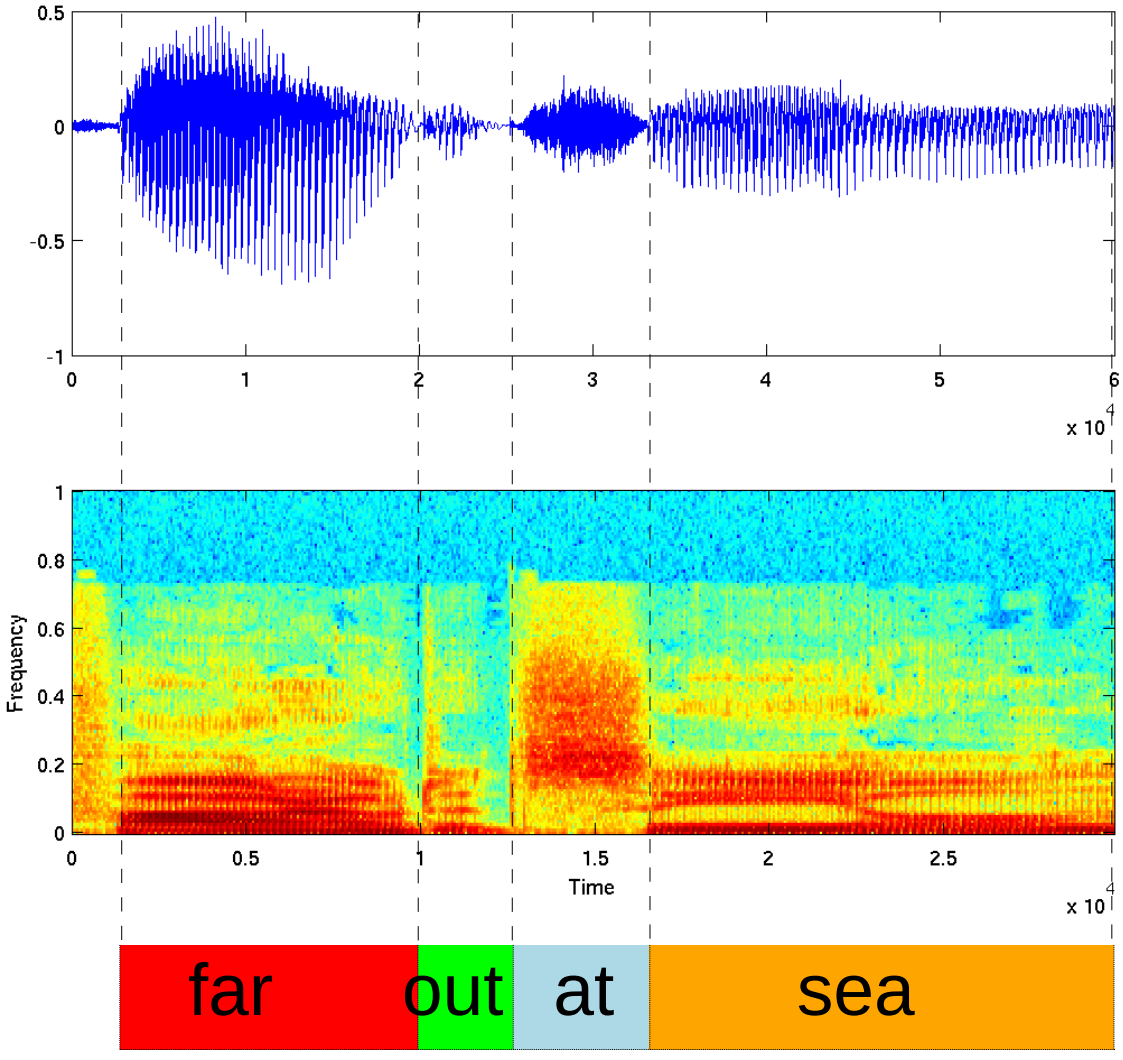
- Imagine you have a database with images of handwritten characters. You need to write software which takes a new handwritten character and decide automatically which one it is (a **classification** tool).



- How would you go about this? How about just distinguishing between 1's and 2's?

Speech recognition

In this type of problem (speech-to-text), we take a recorded speech signal and infer the words it contains.



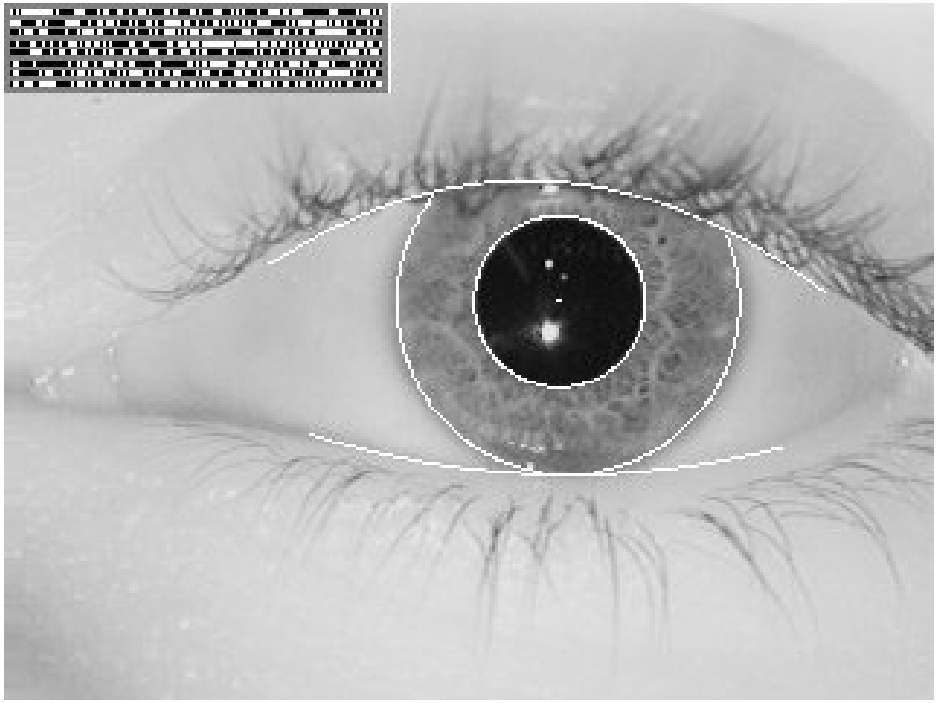
Predicting the stockmarket

- Patterns also exist in other types of sequential data, for example share prices. Here we want to predict what the next value in the sequence will be (a **regression** problem).

- More commonly, we try to predict a range of values with different certainties.

Representations of data

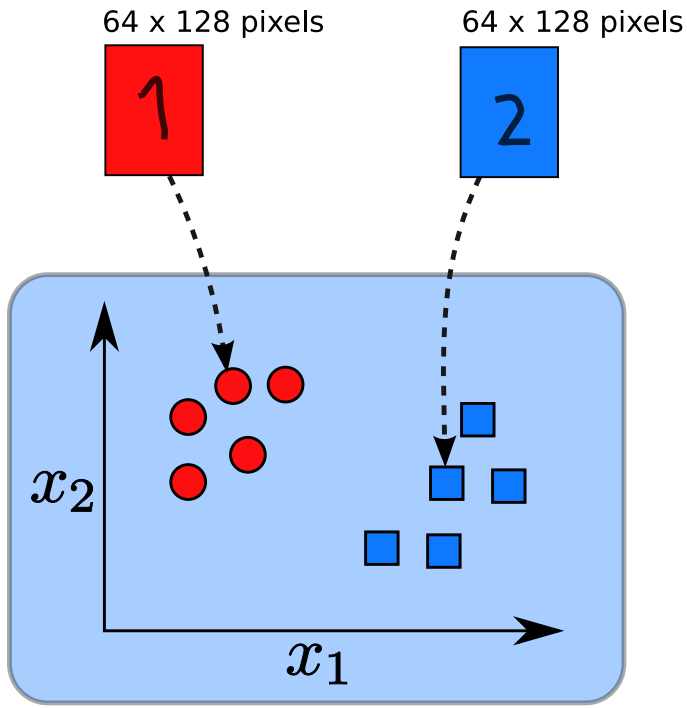
- If you have a hard problem to solve, convert it into an easy one and solve that.
- There might be a simple way to represent a complex dataset, making pattern recognition easy.
- For example, the “iriscode” below contains a compact summary of information about patterns in the iris, which makes it possible to identify individuals:



www.cl.cam.ac.uk/~jgd1000

Dimensionality reduction

- Returning to the handwritten digit identification problem, part of the difficulty is that the raw data is high dimensional (e.g. 64×128 pixels $= 8192$ dimensions).

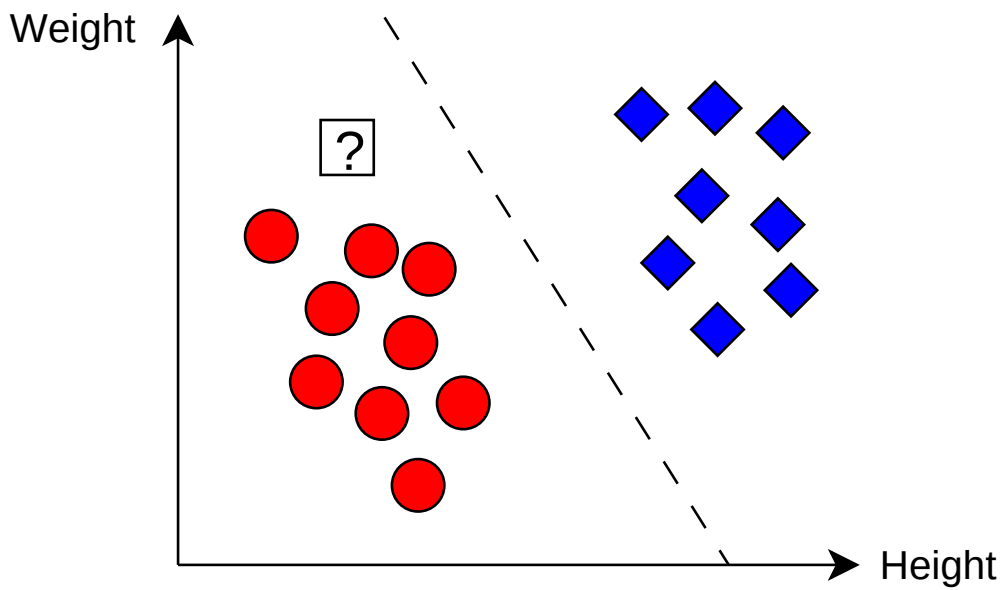


- If there was some way of mapping each image to a point in a lower dimensional space (2D in this example), then the problem might become easier.

Decision boundaries

Imagine you have two dimensional data – the heights and weights of different animals. There are two types of animal in your collection.

The red circles are all your examples of class 1 (snarks). The blue diamonds are all your examples of class 2 (boojums). Given a new datapoint, which type of animal is it?

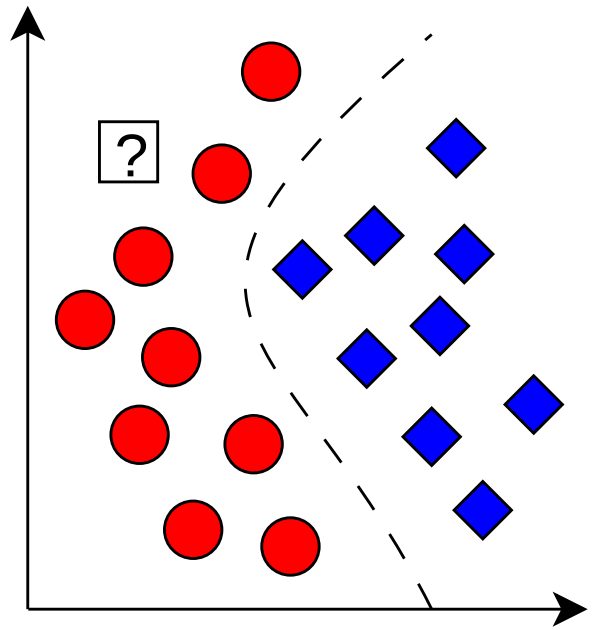


The decision boundary (dashed line) is the set of points at which both classes are equally likely.

Nonlinear decision boundaries

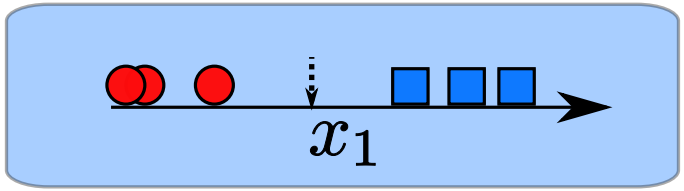
We will look at some methods which draw a straight line (a hyperplane in higher dimensions) between classes – **linear classifiers**.

But sometimes the problem is nonlinear, as in the diagram below. We need different methods to deal with such cases, e.g. neural networks.

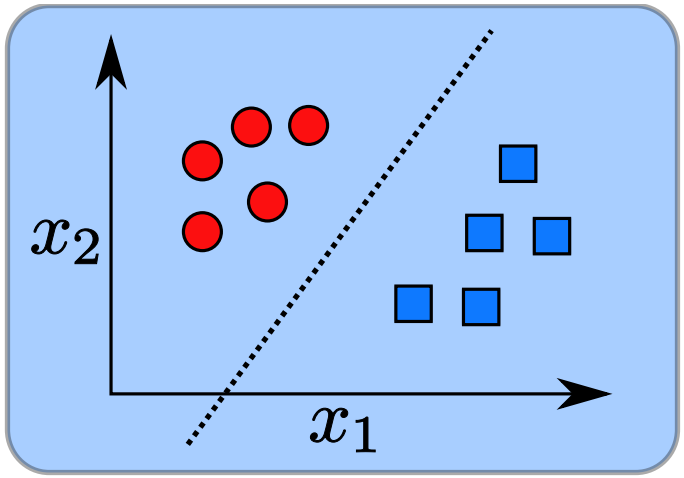


Describing linear decision boundaries

In 1D, the decision boundary can be described with the equation $x_1 w_1 + b = 0$.



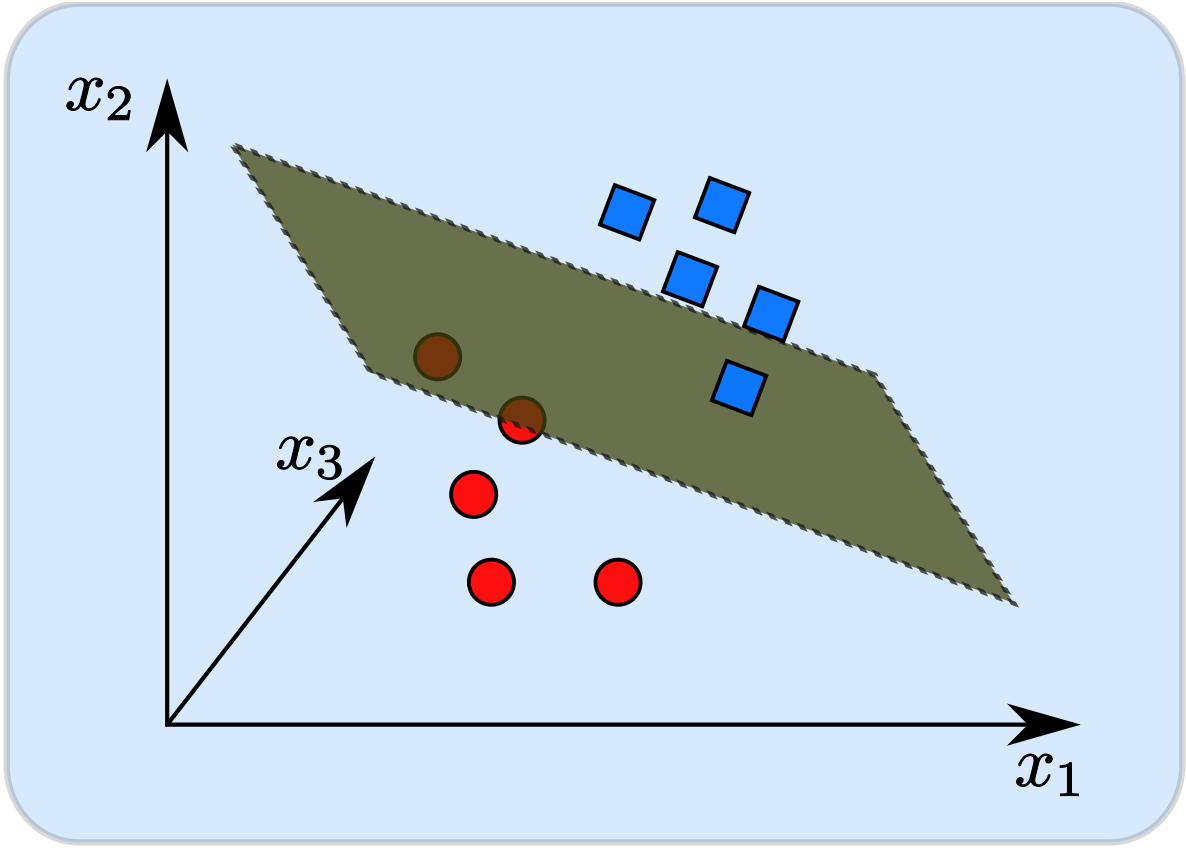
If this quantity is positive, we assign to “blue squares”, if negative we assign to “red circles”.



In 2D, the boundary is at $w_1 x_1 + w_2 x_2 + b = 0$

Decision boundaries in higher dimensions

With three dimensions the decision boundary is a plane (in higher dimensions a hyperplane).



This can be expressed in vector notation as $\mathbf{w}^\top \mathbf{x} + b = 0$.

Practical exercises

- Pattern recognition is a practical subject, and there will be a number of exercises throughout the semester.
- Programming will be done in Python.

Pattern recognition tips

Keep the following in mind while you are working on practical exercises:

1. Good features usually beat fancy techniques: try to find a way of representing the data which has only the information you need.
2. Try to find ways of incorporating background (prior) knowledge about the problem into your method.
3. Visualisation is important: always use plotting tools to try and see the characteristics of the data. If you are trying to classify data, and you can find a way of plotting it so that the classes are clearly separated (as in the snarks and boojums example), then you will definitely be able to find a method to do it automatically.