1 Introduction

In machine learning, we wish to *learn* the mapping between input and output without having to program it explicitly.

> Maching learning is the field of study that gives computers the ability to learn without being explicitly programmed. - Arther Samuel.

> A computer program is said to learn from exerience E with respect to some task T and some performance measure P, if its performance on T, as measured by P, improves with experience E. - Tom Mitchell.

In general, most machine learning boils down to a simple prediction equation.

$$\hat{Y} = f(X, \theta) \tag{1}$$
$$Y = f(X, \theta) + \varepsilon \tag{2}$$

where f is a mathematical model for predicting Y from X. X is a list of input features that are used for prediction. They could be independent variables or some kind of predictor. $\hat{Y}$ is the output of the predicting model which are the predicted values. θ are the model parameters that the model tries to learn and optimise for. Finally, ε is the error, which represents the difference between predicted values and desired values.

1.1 Prediction error

In order to measure the difference between the predicted values and actual values, the error for each data point must be measured or calculated in some way. To do this a **loss function** (cost function) is used to evaluate the quality of the model. The goal is to then try to minimise the loss function by altering the θ parameters to allow the model to make better predictions.

An example of a common loss function is the squared error loss.

$$L(\theta) = (Y - f(X, \theta))^2 \tag{3}$$

The loss function serves as a way to measure the quality of prediction for the set of parameters.

1.2 Machine learning process

1. Gather data

2. Prepare data (data cleaning and preprocessing)

3. Pick an algorithm (or a few different ones)

4. Pick a set of parameters

5. Evaluate performance of model

6. If the performance is not good, re-evaluate by altering parameters

7. Apply model to new/unseen data

This is generally implemented as an optimisation process, where the loss function is minimised on each iteration by changing the θ parameters. This strategy picks the optimal set of parameters. An example of such an optmisation process is gradient descent.

1.3 Supervised machine learning

The idea of supervised machine learning it to be able to predict an outcome based on given input data with tagged correct outputs. This could involve predicting a value or classifying an output class type. This requires a **labelled dataset** so it is known what the output is. There are two main types of supervised learning:

- Regression - The data predicted for regression is usually continuous and if it is decrete, it has to be rounded to the closest discrete value. Regression predicts the *value* of the data.

- Classification - The probability it is of some class. Classification predicts whether the data belongs to a certain class.

Classification uses a **decision boundary**, which everything on one side of the boundary is classified as one class and another on the other side of the boundary. It is possible to have hyper-dimensional decision boundaries for when there are more than 3 features which cannot be easily visualised.
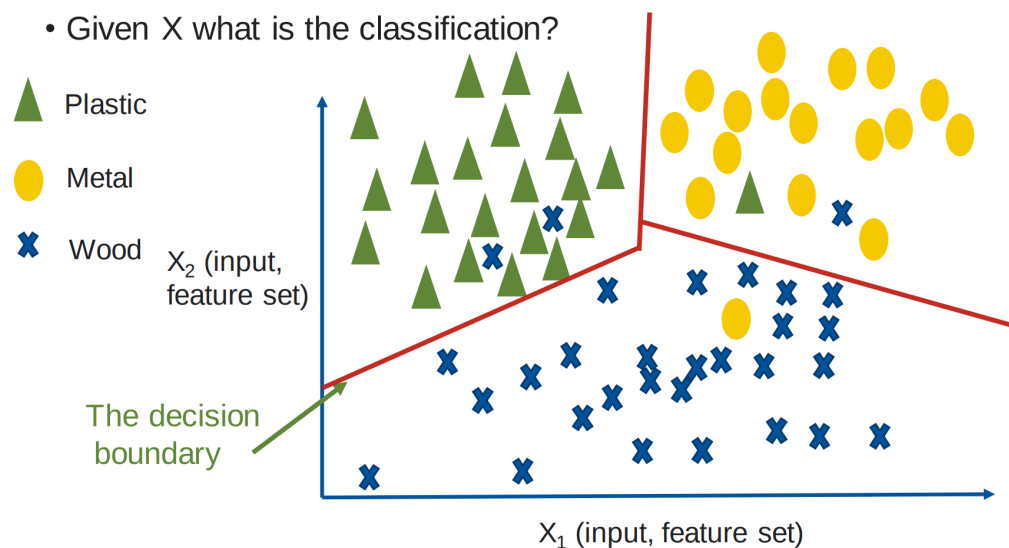

Figure 1: Classification of 3 classes using 2 input features. Having more than 3 input features (3D plot) is difficult to visualise.

One major issue with many machine learning algorithms: **Overfitting**. Often the models trained become optimised too well towards the training data, leading to very low errors, but an inability to generalise to unseen data.
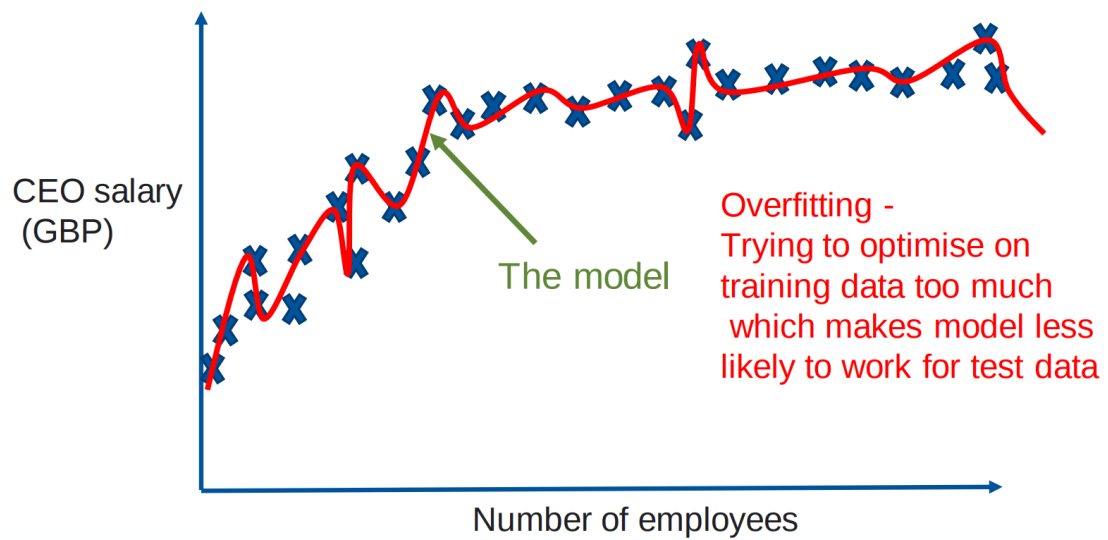
Figure 2: Example of overfitting in linear regression.

1.4   Unsupervised machine learning

In unsupervised learning, the dataset is unlabelled, and the output for predicted values is unknown. To learn, clusters are automatically created to separate the data into distinct groups. This is useful for discovering information where we don't know what we are looking for or to discover things we don't know about.