**Common Machine Learning Algorithms**

*Linear Regression*

Linear Regression is one of the most basic forms of machine learning, it is supervised learning where we train it to predict future relationships between variables from a data set. As the name suggests linear regression usually depicts a linear correlation between what we know (predictor variable, X -axis) and what we want to know (target variable, Y- axis). A linear regression algorithm could be used to predict the yearly changes in house prices. Using past information of the increase and decrease in house prices over the years the algorithm could predict the future outcomes. Linear regression models are commonly used in the world of social sciences and finance.

*Logistic Regression*

Logistic regression is a supervised learning algorithm with input variables (X-axis) and target variable (Y- axis). Logistic regression is used to estimate the likelihood of an event based on previous data. Uses a variable with 2 class values (Y- axis, event occurring, event not occurring) that are turned into binary data (1,0) to test the presence of other variables (X- axis). Unlike linear regression logistic regression does not seek to find a correlating relationship between the variables. However, it uses the function to find a relationship from the data and then identifies the coefficients that allow it to predict the future outcomes resulting in an ‘s’ shaped data model. Logistic regression will determine the probability of a 1 outcome. Logistic regression is useful when we are assessing variables that are opposites (true/false, present/absent, on/off), such as detecting spam mail or likelihood of developing a disease. For example, logistic regression could be used to predict the likelihood of developing cancer How many years a person has smoked (X- axis) and developing lung cancer (Y- axis).

*Decision Tree*

Decision Trees are another form of supervised learning algorithm, although, unlike other supervised learning algorithms decision trees can be used to solve problems that are both regression (continuous data type) and classification (yes/no data type) based. The goal of the decision tree is to create a model that predicts the class or value of a target variable through learning simple decision rules that are inferred from prior training data. The decision tree would start at the root with a variable and then move to a decision node based on the attributes of the variable (fit or not fit) and the prior training this process will repeat until the variable arrives in a terminal node(leaf). Decision trees can be implemented in a variety of settings from predicting a drink combination in Starbucks, to deciding if someone is eligible for a loan. A person's data would be inputted, and the decision tree algorithm would run through the decision nodes in an if/else scenario:

* Are they employed? Yes (move on) No (not eligible for a loan)
* Income? Below 20k (not eligible) above 20k (move on)
* Monthly Outgoings? Less than 50% of income (move of) more than 50% of income (not eligible).

Although decision trees can be split into more than 2 outcomes for the decision nodes i.e. the income could have been split (< 5k, 5k - 20k, 20k - 30k, 30k+), in turn meaning the outcome of loan eligibility could be varying (not eligible, up to 10k, up to 15k).

*Support Vector Machine (SVM)*

SVM is another supervised learning model which like decision trees can be used to analyse data for both regression and classification. SVM plots each data point on the n- dimension (n- being the number of functions) and then separates the data plots via a hyperplane, simply a line that separates the classes (collections of data points). SVM uses a technique known as the kernel trick which is most useful when there is a non-linear separation problem, allowing the SVM to use hyperplanes to separate the classes over a higher dimensional space to separate the data adequately. SVM’s are used in facial recognition classifications, for apps such as snap chat where filters require a person to do certain facial expressions to activate the features. Where the snapchat dog filter requires you to open your mouth to make the animated tongue appear.  The SVM will run the facial expressions and plot them into the classes based on the prior data set and labelling parameters that have been established.

*Naive Bayes*

Naive Bayes is a supervised learning algorithm that is quite simple and seen as an extremely accurate, fast (usable for real-time output), and reliable method used in classification tasks. Based on a simple maths formula Bayes Theorem that is used to calculate conditional probability. The underpinning assumption of Naive Bayes is that each feature is independent and equal. Naive Bayes are effective with large data sets. Naive Bayes is a probability-based algorithm that takes the input data and based on prior learning and labelling decides the outcome classification. The input data can have many values, the more assigned values the more accurate the outcome would be. For example: if you wanted the algorithm to identify the most likely piece of fruit based on description alone. (banana, orange, other), the more values that the data is passing through the more accurate the outcome would be. Using values such as: sweet, yellow, long, peel, red, etc. Naive Bayes is extremely useful in text classification for spam filters and document classifications, using data filters of common phrases (prize/you’re a winner/ congratulations)

*K-Nearest Neighbours (KNN)*

KNN is a supervised learning algorithm that is used for both classification and regression problems. Although it has a massive drawback as the data set grows the algorithm becomes significantly slower. The KNN algorithm works on the assumption that items that are similar would be close to each other. KNN works with structured table format data and the column all contain numerical data, with the last column being labels that functions can be performed on. If KNN was implemented for movie recommendations a data set containing movie titles and then every genre category would be binary based (1 for yes 0 for no) when the algorithm is presented with a new movie title it would look at the genre classification and then recommend other films with the same genre classification.

KNN would be most useful in recommendation systems (Netflix, Amazon etc).

*K-means*

K-means is an unsupervised learning algorithm, which means the groups have not been labelled K-means separates data into subcategories that are separate from one and other. Data points will only belong to one sub-category (cluster), they will be assigned to the closest cluster. K-means attempts to keep the clusters small to decrease the amount of variation in each cluster, having the same principle as KNN where data points that are near to each other are similar, when running a K-means algorithm you do have to define the number of clusters that you want the algorithm to use which entails prior testing of the data to establish the existing. K-means is commonly used in insurance fraud detection. By using past fraudulent claims K-means can predict the likelihood of new claims being fraudulent based on the data points proximity to fraudulent patterns clusters.

*Random Forest*

The random forest algorithm like the decision tree algorithm is a supervised learning algorithm, which incorporates many decision trees to overcome the shortfalls of a single decision tree. The algorithm randomly selects observations and features to build the decision trees and then averages the results. The theory is that the use of many smaller decision trees that have been built with random subsets of features to decrease the issue of ‘overfitting’ seen with a single decision tree. This is known as the ‘bagging’ method where many smaller trees that are then merged provide more accurate outcomes. They are highly accurate due to the high numbers of variables and the capability to handle a large sample size. Random forest algorithms are commonly used within the stock market and e-commerce due to the strong predictability capabilities.