COMP1801 - Machine Learning Coursework Report

Chittiboina – Sai Sandeep – 001203994

Word Count:1431

1. Executive Summary

In this report, I will run some machine learning models on the data for a small group of customer's income. The goal of this report is to use regression methods to predict a customer's salary and binary classification and neural networks to classify customers who earn more than or less than £35K. After dividing the customers into two different groups using the k-means clustering technique, I will summarise my findings, compare the methodologies used and suggest a single method for predicting the customers' salaries.

2. Introduction to Machine Learning

Machine learning is a subfield of artificial intelligence that refers to a machine's ability to mimic intelligent human behaviour. Artificial intelligence systems are used to complete complex tasks in the same way that humans solve problems. Machine learning is an essential component of the rapidly expanding field of data science. Algorithms are trained to perform classifications or forecasting using statistical methods, revealing key insights in data mining tasks. These insights are then used to influence decision-making within applications and companies, with the goal of influencing key growth metrics. As big data expands and grows, so will the demand for data scientists to assist in the choice of the best business questions. Although the terms deep learning and machine learning are frequently used interchangeably, it is critical to understand the distinctions between the two. Machine learning, deep learning, and neural networks are all subfields of artificial intelligence. Deep learning, on the other hand, is a subfield of machine learning, and neural networks are a subfield of deep learning.

3. Regression

Here I have used two models to perform regression namely multi-linear regression, and extra Trees regressor. By comparing the R2 scores, I am choosing the better model to predict the salaries of the customers.

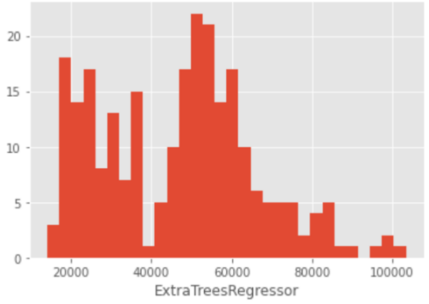
Multi-linear regression: Besides incorporating a linear model into actual observations, multi-linear regression tries to simulate the connection between two or more independent variables and dependent variables. Each independent variable x value corresponds to the value of the dependent variable y.

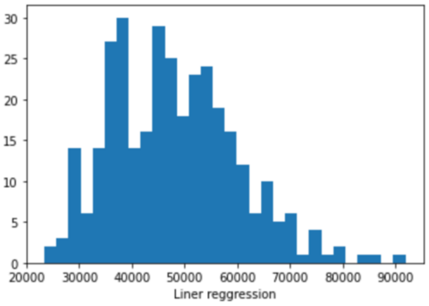
$$\mu_{y\,=}\ \beta_0 X_0 +\ \beta_1 X_1 +\ \beta_2 X_2 + \cdots + \beta_n X_n + \in$$

Extra Trees Regressor: Just like the random forests algorithm, the extra trees algorithm generates a large number of decision trees, however, the sampling to every tree is random and without replacement. This generates a dataset with unique samples for each tree. For each tree, a specific number of features are chosen at random from the total set of features. The random selection of a separating value for a feature is the most important and distinguishing feature of extra trees. Instead of determining a locally optimal value for splitting the data using Gini or entropy, the algorithm chooses a split value at random. As a result, the trees are diverse and unrelated.

Implementation and results:
For implementing the multi-linear regression, I first encoded the education, work type, sex and region. After the encoding I divided the data into test and train splits where I assisted the target as salaries of the customers by using the function linear_model.LinearRegression() from sklearn. Then, I fitted the X_train, y_train and calculated the predicted value of X_test. Post that after fitting the model, I imported r2_score from sklearn.metrics and executed this r2_score(y_test, pred). For the extra tree regressor as we already divided the data into X_train, X_test, y_train, y_test we can just implement the function.

R2 score for Multi-linear regression: 0.29367721139243297
R2 score for Extra Trees Regressor: 0.8279567568412703





By taking the R2 scores into consideration we can say that **extra trees regression** is giving more accurate predictions of the customer salaries with the **R2 score of 8.2**

4. Binary Classification

Here I have used three models to perform binary classification that is classifying customers who earn more than or less than £35K per annum.

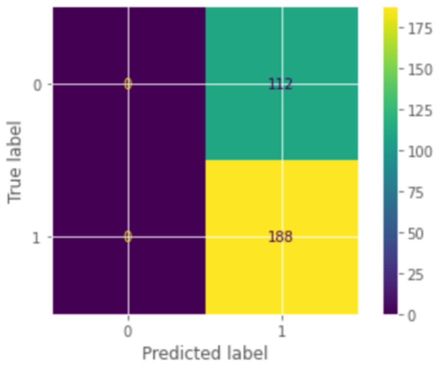
SVM Support Vector Machine: The support vector machine is a type of supervised learning system that is used to solve regression and classification problems. Many people prefer support vector machines because they produce significant correctness with less computing capability. It is typically applied to classification problems. Learning can be classified into three types: supervised, unsupervised, and reinforcement learning. A support vector is a type of classifier that is formalised by separating the hyperplane. Given labelled training data, the algorithm generates the best hyperplane for categorising new examples. This hyperplane is a two-dimensional line that divides a plan into two parts, with each class on either side. The support vector machine algorithm seeks a hyperplane in an N-dimensional space that classifies the data points separately.

GaussianNB: The Bayes Theorem is used to influence Naive Bayes, a simple but effective probability classification model in machine learning.

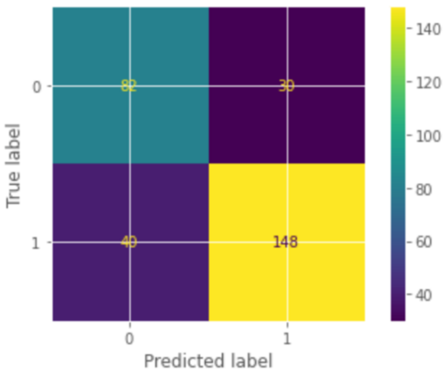
The mathematical formula for it is: $p\left(\frac{a}{b}\right) = \frac{p\left(\frac{b}{a}\right).p(a)}{p(b)}$

Decision Tree: The Decision Tree algorithm is a member of the supervised learning algorithm family. The decision tree algorithm, unlike other supervised learning algorithms, can also be used to solve regression and classification problems.

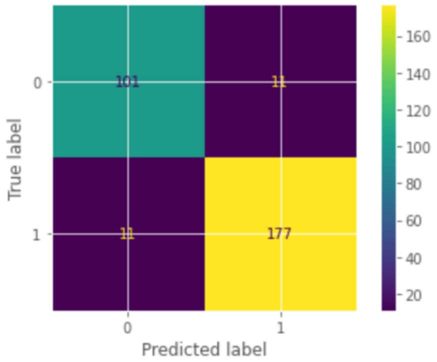
Results:



Confusion Matrix for SVM model



Confusion Matrix for Naïve Bayes model



Confusion Matrix for Decision Tree model

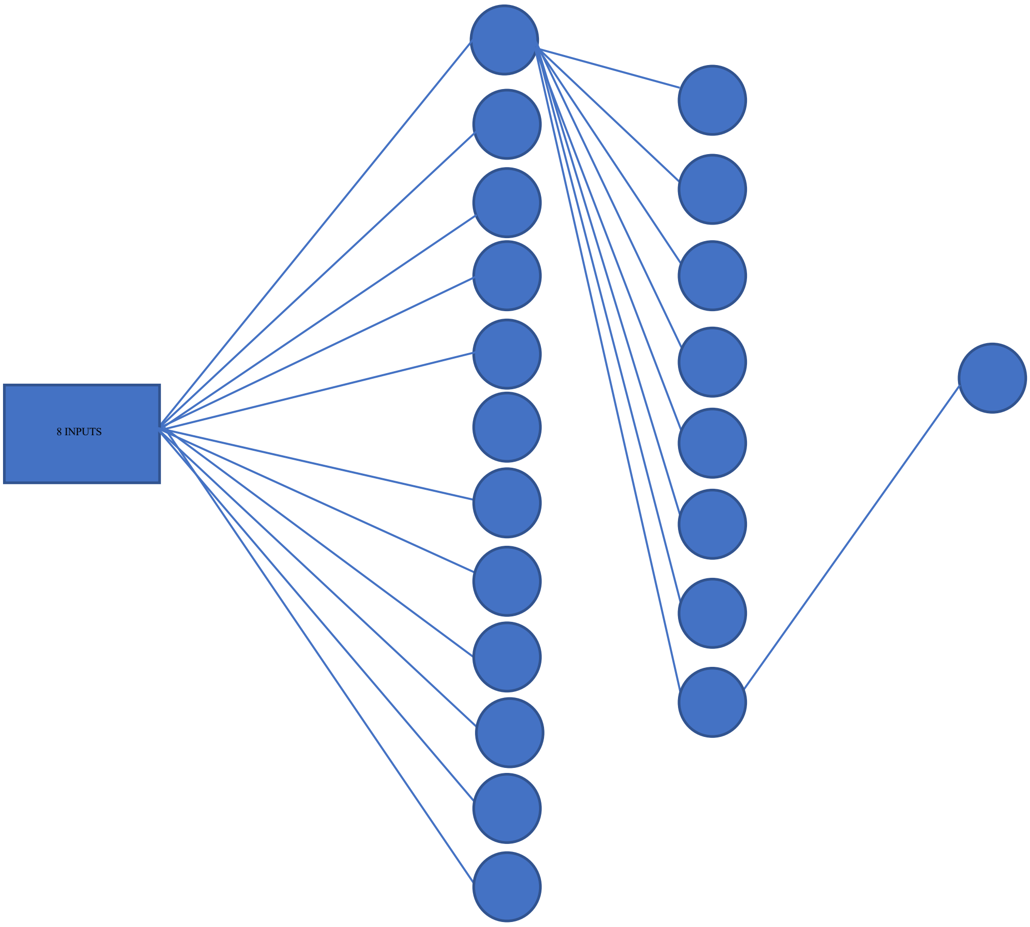
Executing the above said three models for classifying the customers depending on their average salary in the UK into those who are earning more than 35,000 pounds per year and those who are earning less than 35,000 pounds and by understanding the confusion matrix **decision tree,** I have concluded that **the best fit to classify this problem is with the accuracy of 92.6%.**

5. Neural Networks

For the neural network model, I have trained a sequential model where the first step is to initialise it as an empty neural network model until we add more layers sequentially beginning at the top.

The first layer that we add to the model is 12 neurons with 8 inputs (age, site spending, site time, recommended impression, education, work type, sex, religion) Then I add another hidden layer with 8 neurons and a 1-neuron output layer.

The visual of my layers in the model has only one output layer because it is for the same problem statement that we have solved to classify the customers based on their annual income of less or more than 35k pounds.



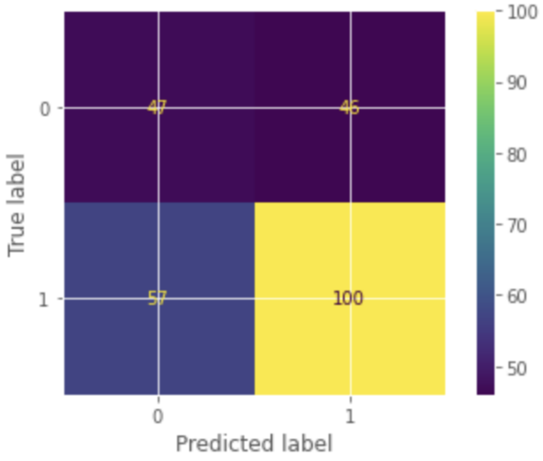
Results:

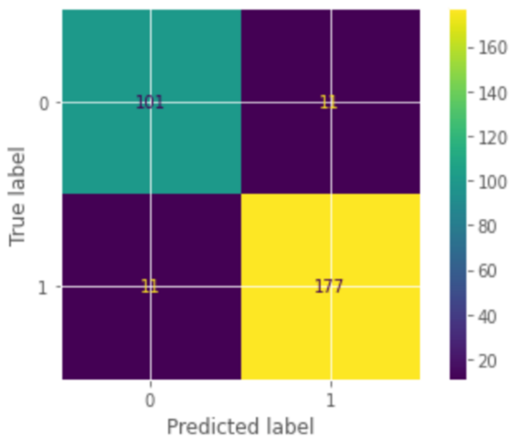
Accuracy score: **58.8%**

Confusion score matrix:

array([[47, 46], [57, 100]])

Comparing the neural network with the binary classification model which we have selected



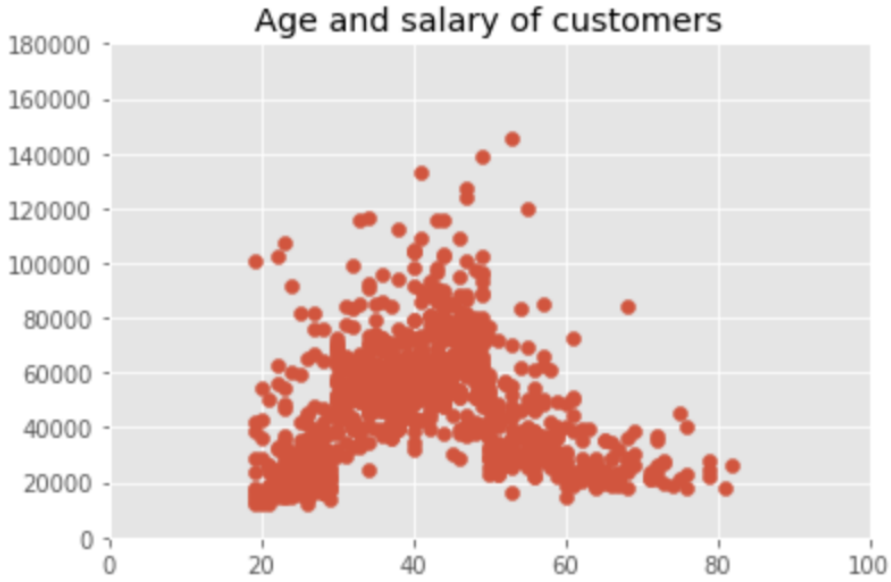


Confusion Matrix for Sequential neural network Confusion Matrix for Decision Tree

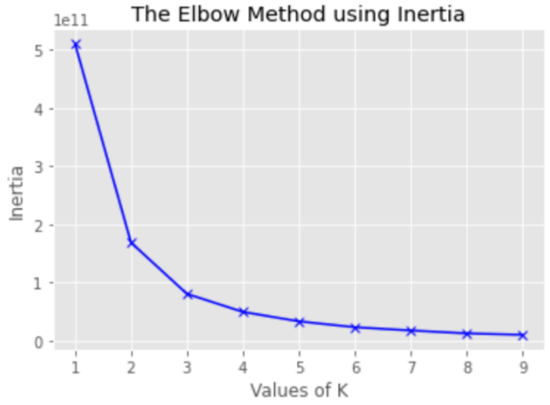
By comparing the two we can clearly say that the decision tree is giving good results with an accuracy of 92.6%, and **the reason why the neural network didn't perform well is because we are using a small data set and the neural network trains well with large data sets**.

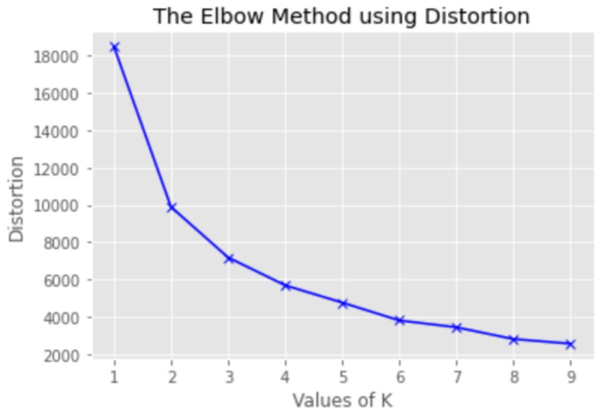
6. Clustering

By performing k means clustering between the age and the salary of the customers, I have decided that age is an appropriate variable to perform clustering.

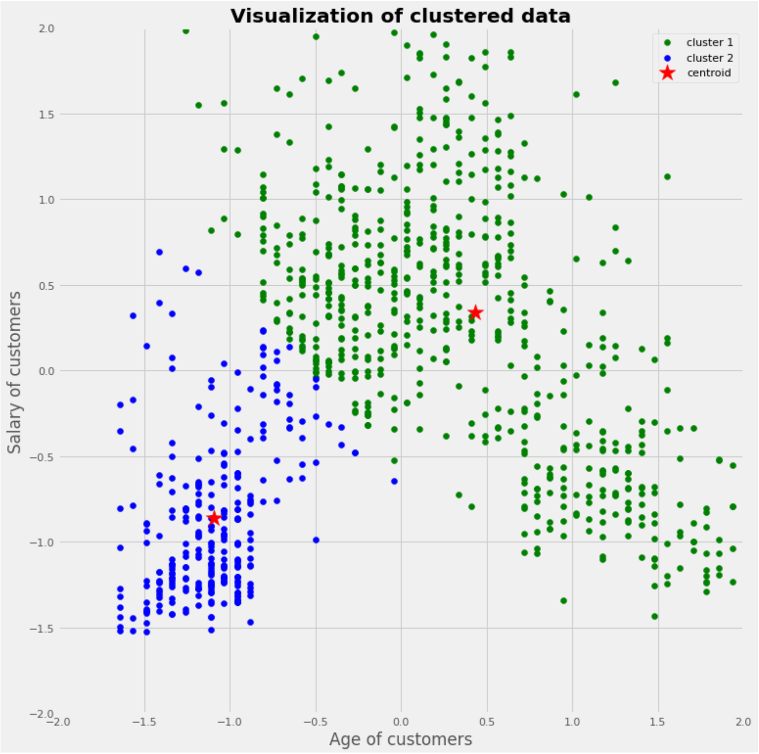


From looking at the output signals obtained from performing the elbow method for the data using inertia and distortion, we can see the bends from the graph and can take **K-value = 2.** Thus, we can say that the manager's decision to split the customers into 2 groups is the right choice.





After selecting the k value as 2 this was the visualization of the clustered data divided into 2 groups with the blue and green colours and the star being as its centroid.



7. Conclusion

- I used two models in the regression method to predict the customers' salaries.

	Multi-linear regression	Extra Trees Regressor
mean square error	334340661.0616987	87998538.74369547
root mean square error	18284.984579203196	9380.753634100805
R2 Score	0.29367721139243297	0.8279567568412703

- After obtaining the results of the models and examining the matrix, it was clear that the extra tree regression provided the predicted values of the customers' salaries.

- And when it comes to classifying customers based on their salaries, the decision tree outperforms all others with an accuracy of 92.6%.

	SVM	Naïve Bayes	Decision tree
Accuracy	62.6%	76.6%	92.6%

- When it comes to piece-by-piece model fitting, neural networks take a probabilistic approach, whereas trees take a deterministic approach. Regardless, both rely on the depth of their models to perform well because their components correspond to different parts of the feature space. The main distinction between tree-based methods and neural networks is the use of deterministic (or 1/1) vs. probabilistic data structures. Deterministic models consistently perform better when modelling structured (tabular) data.

	Decision tree	Neural network
Accuracy	92.6%	58.8%

- For the k means clustering, by performing the elbow method for the data using inertia and distortion it’s very evident that the manager's decision of splitting the group into 2 groups was a good choice.