COMP1801 - Machine Learning Coursework Report

Fariha Tasnim Khan – fk7125m – 001249887

Word Count:3925

1. Executive Summary

I am working with an ecommerce company who has collected information of a small group of customers and wants to use to it to find the income of rest of their customers by predicting. This report contains results of different machine learning models I have used to predict.
In Machine Learning, the computers are taught to learn from experience. Therefore, using the existing information, predictions can be made. In part 3, 4 and 5 supervised machine learning is used to predict the salaries of the customer and in part 6 unsupervised learning is applied to the data.

2. Introduction to Machine Learning

Machine learning gives computers ability to become more accurate at predicting without being explicitly programmed to do so, by learning from experience. It can analyse large datasets, find patterns and create models more efficiently than humans.

The two popular types for machine learning are supervised and unsupervised learning. On supervised learning the computer algorithm is trained on input data that has been labelled for a particular output. On the other hand, in unsupervised learning the computer algorithms learn based on the relationships among the data only. Some of the popular approaches to machine learning are: KNN, Decision Trees, etc. (Tagliaferri, 2017)

3. Regression

Regression is a type of supervised learning, which predicts real values from data. (Anon., n.d.)

The dataset contains non-numerical data, but to work on regression models, numerical data is required. Hence, I converted the non-numerical values to numeric values, for which I used Label Encoding for the conversion which assigns numeric value. (Anon., n.d.)

After encoding, I have analyzed the dataset as before working on the model, it is very important to first understand the dataset well and how each variable in the dataset is related to each other. I have used correlation heat maps, which shows how strongly the variables are correlated.

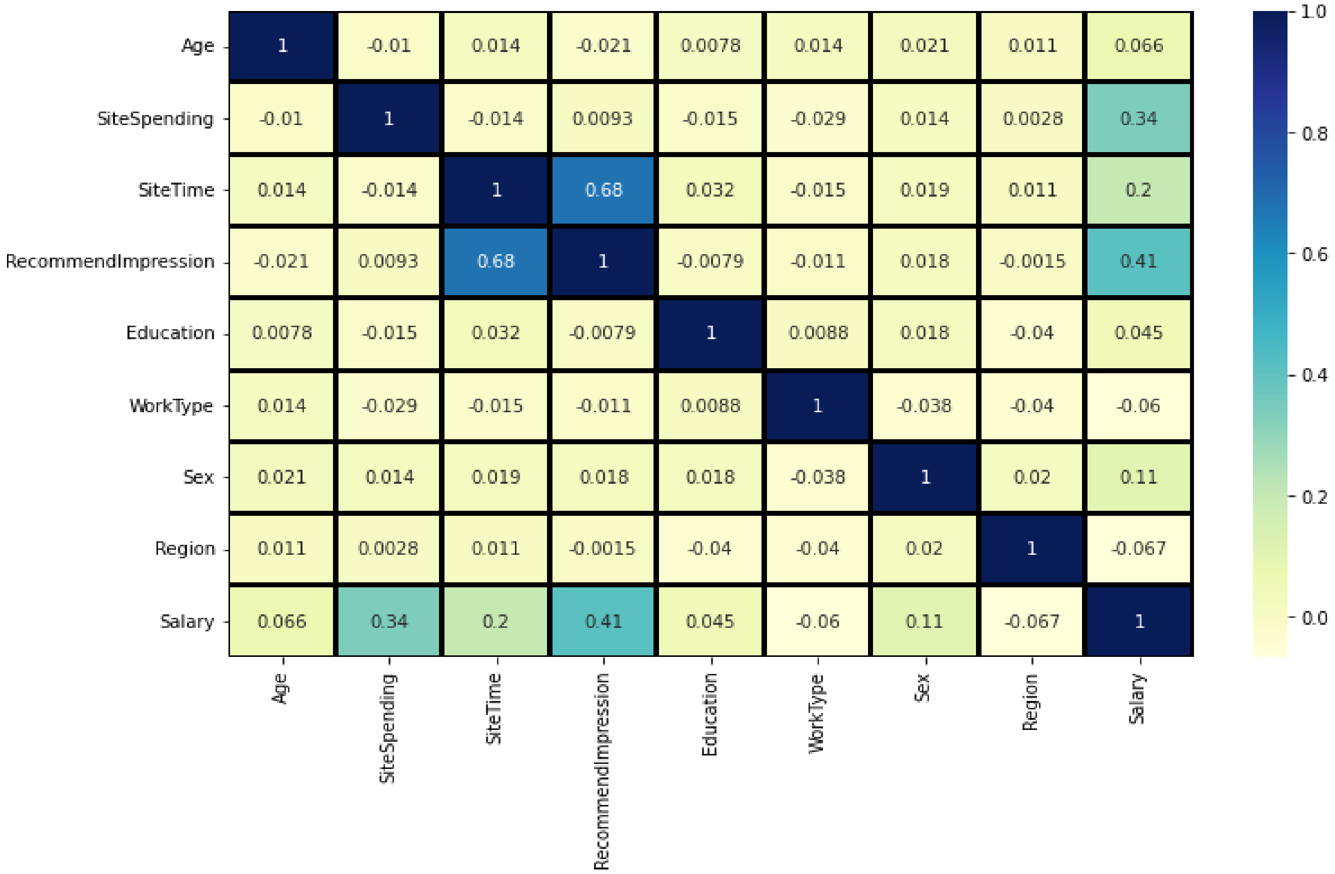


Figure 1: Heat Map showing correlation between all the variables in the dataset "Customer Income".

Variables that are directly connected have positive correlation. It can be seen in Figure 1, that "Region" and "WorkType" have negative correlation with Salary, which indicates they don't have direct relation and hence can be excluded from the training data.

Next, I have scaled the dataset. Scaling is a method for evenly distributing the independent features in the data over a set range. If scaling is not done, machine learning algorithms tend to evaluate larger values more highly, regardless of the value's unit. For example- After label encoding the non-numerical variable “Education”, the values assigned are (0,1,2,3,4,5,6). So if it is not scaled it may be possible that the machine learning algorithm is considering the value 5 greater than 1 and this may lead to wrong predictions. (Gogia, 2019) I have used Min-Max scaling which rescales the features value within the range 0 to 1 which is shown in Table 1.

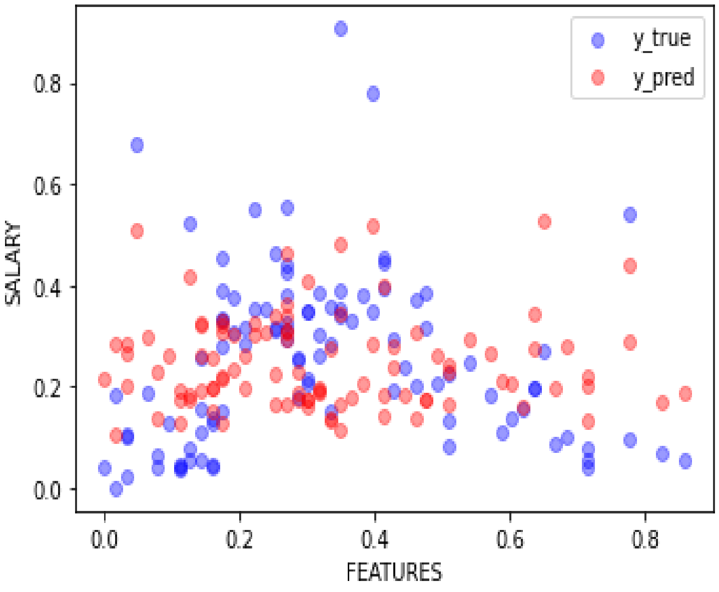
	Age	**Site Spending**	**Site Time**	**Recommend Impression**	**Education**	**Work Type**	**Sex**	**Region**	**Salary**
0	0.206349	0.062878	0.084627	0.000000	0.166667	0.0	1.0	0.181818	0.359470
1	0.015873	0.854345	0.487710	0.235294	0.333333	0.0	0.0	0.636364	0.229875
2	0.269841	0.128979	0.056666	0.000000	0.500000	0.0	1.0	0.090909	0.319939
3	0.031746	0.001025	0.600771	0.529412	0.500000	0.0	0.0	0.454545	0.107643

Table 1: Dataset after Scaling.

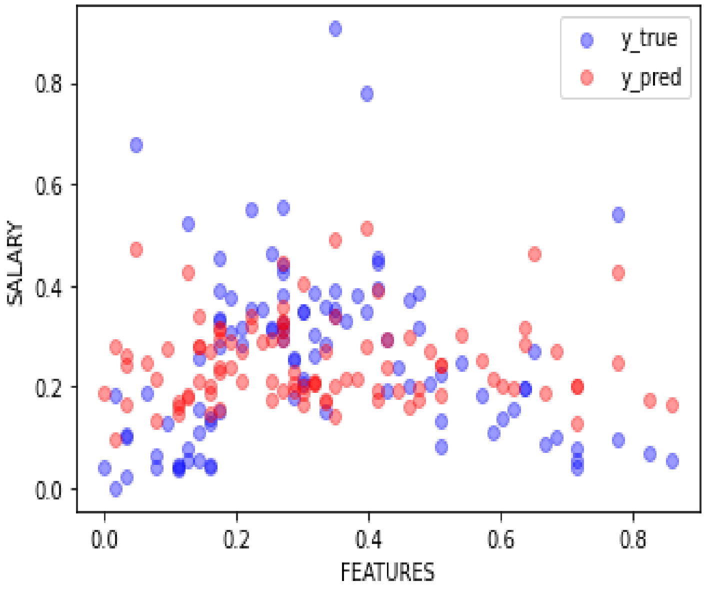
I have then shuffled the dataset for a random sample, which is done to avoid any patterns in the datasets. (Gowda, 2017) The features and target variables are then assigned, after which I have split them into train set and test set. I have chosen a split percentage according to how my model best works (Train: 90%, Test: 10%). The train-test split is a way for assessing the effectiveness of a machine learning algorithm. The train set contains data with which the model will learn and the test set contains data with which the trained model is tested.

Finally, the regression models are fitted and trained using the training set. The types of regression model I have fit are: Linear Regression, Random Forest Regressor and Kernel Rigde with 2 different kernels which are “rbf” and “laplacian”. After completion of training, the testing data set is used to predict and to ensure that the final model operates correctly, the training and test sets of the target data are compared.

LINEAR REGRESSION



KERNEL RIDGE(Kernel=’rbf’)



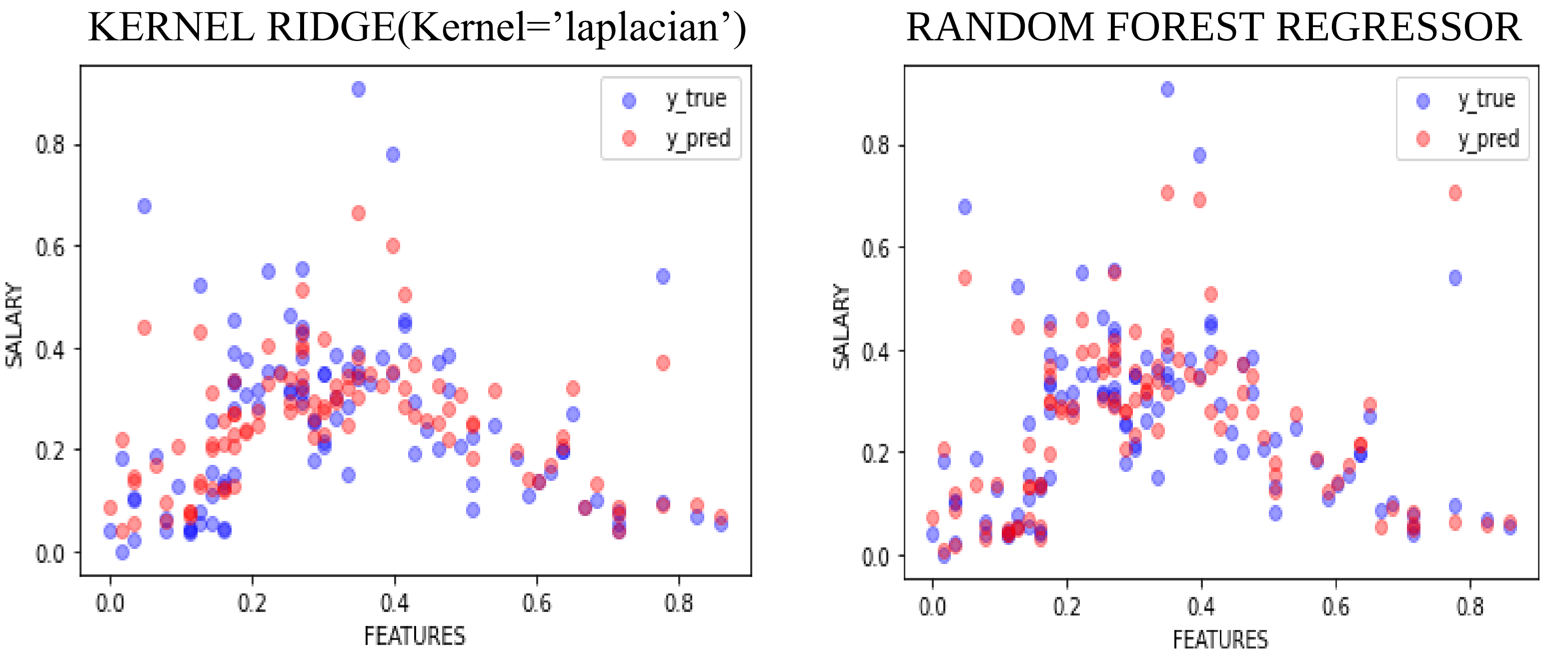


Figure 2: Scatter Plots of the different models I have implemented, where the actual values are shown in blue points and predicted values are shown in red points.

Amongst the 4 plots in Figure 2, Kernel Ridge with "laplacian" kernel and Random Forest Regressor has the highest number of actual points that lies on top of the predicted points, which indicates that these models have high accuracy. On the other hand, in comparison, Linear Regression and Kernel Ridge with "rbf" kernel has a high number of predicted points away from the actuals.

After making predictions I assessed the models using a regression evaluation metric to see if the prediction is accurate or not. I have used R-Squared (R^2) which measures how well predictions match actual data, where 1 is perfect fit. (Agrawal, 2021)

Regression Model	R-Squared (R^2) Scores
1. Linear Regression	0.3807 (38.07%)
2. Kernel Ridge(Kernel='rbf')	0.4519 (45.19%)
3. Kernel Ridge(Kernel='laplacian')	0.8061 (80.61%)
4. Random Forest Regressor	0.9076 (90.76%)

Table 2: R-Squared scores of Regression models.

Here, it can be seen that I have received the highest accuracy with 2 models which are Random Forest Regressor (90.76%) and Kernel Ridge with Kernal "laplacian" (80.61%). As the model Random Forest Regressor has the highest R^2 score and also it can be seen in the Figure 2 that most of the points of actual and predictions match so it can be said that it has the highest accuracy and this is why I have chosen this model.

Observations while experimenting:

1. When testing and trying, I have observed that when I selected split percentages as Train: 80% and Test: 20%. 80% my results were bad compared to Train: 90% and Test: 10%. The table 3 below shows results with Train: 80% and Test: 20%.

Regression Model	**R-Squared (R²) Scores**
1. Linear Regression	0.3047
2. Kernel Ridge(Kernel='rbf')	0.3822
3. Kernel Ridge(Kernel='laplacian')	0.7963
4. Random Forest Regressor	0.8801

Table 3: Result with Train: 80% and Test: 20%.

2. I also tried fitting models without scaling to experiment and ended up with very poor results shown below.

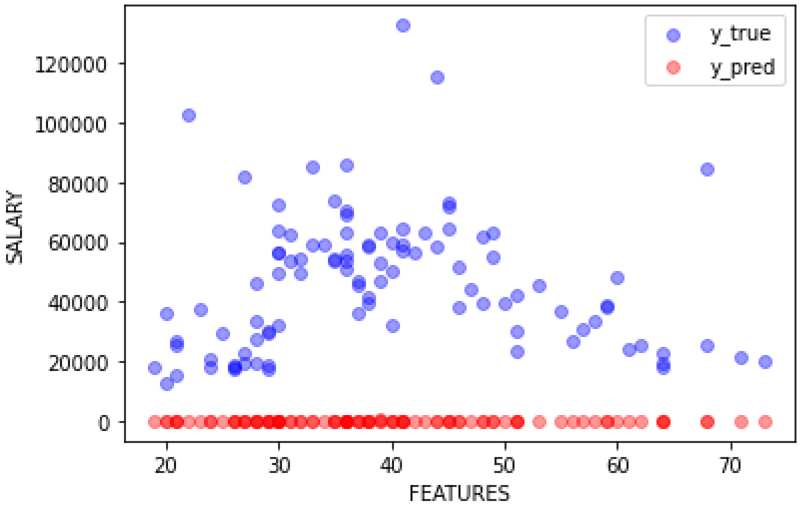


Figure 3: Scatter plot of prediction and actuals without Scaling for the model Kernel Ridge with "rbf" kernel.

3. Binary Classification

Classification is a supervised machine learning approach, which predicts categorical data and in binary classification, the targets are categorized into two classes. (Anon., 2022)

I have categorized the target “Salary” into the following categories:

1. If Customer’s salary greater than £35000→ 1
2. If Customer’s salary less than £35000→0

	Age	Site Spending	Site Time	Recommend Impression	Education	Work Type	Sex	Region	Salary
0	32	314.06	30.14	0	Degree	Private sector	Male	London	60173.49
1	20	3758.36	149.36	4	GCSE	Private sector	Female	South East	42965.45
2	36	601.72	21.87	0	Masters	Private sector	Male	East of England	54924.41
3	21	44.89	182.80	9	Masters	Private sector	Female	Northern Ireland	26734.99

Table 4: Table of the dataset’s first 3 rows

	Age	Site Spending	Site Time	Recommend Impression	Education	Work Type	Sex	Region	Salary
0	32	314.06	30.14	0	Degree	Private sector	Male	London	1
1	20	3758.36	149.36	4	GCSE	Private sector	Female	South East	1
2	36	601.72	21.87	0	Masters	Private sector	Male	East of England	1
3	21	44.89	182.80	9	Masters	Private sector	Female	Northern Ireland	0

Table 5: Table of the dataset’s first 3 rows after categorizing the salary into 2 classes (0,1).

The difference between the Table 4 and 5 can be seen, where all salaries greater than £35000 have been replaced by 1 and those with less than £35000 have been replaced by 0.

Next, similar to the regression models, I have label encoded, scaled the data using Min Max Scaling and also found out the correlations between the target and the features and according to the heatmap plot in Figure 1, I have taken all the columns into consideration except, “Salary”, “WorkType” and “Region” and assigned it as features and assigned the target as “Salary”.

Next, I split the features and target into train and test sets and have assigned size of test subset as 30%. Then finally, I have fit different classification models and trained the models using the training set. The models I have implemented are: Decision Tree Classifier, Random Forest Classifier and Support Vector Machines (SVM) with the kernel “rbf”. After the training is complete, the testing data set is used to predict and to ensure that the final model operates correctly, the training and test sets of the target data are compared.

After implementing different classification models, I assessed it using a classification accuracy metric to see if the prediction is accurate or not. For assessment I have used the accuracy score for the predicted values against the actuals. I chose Accuracy because it produces reliable results when the target class is evenly distributed. When I checked to see if the target "Salary" was balanced, I discovered that there were around 400 rows of 0s and around 600 rows of 1s in the training data. Thus, it is not imbalanced. It would have been imbalanced in a scenario if there were 900 rows of 0s and only 100 rows of 1s. (Agrawal , 2021)

Classification Models	Accuracy
1. Decision Tree Classifier	0.91 (91%)
2. Random Forest Classifier	0.93 (93%)
3. Support Vector Machines (SVM) with the kernel “rbf”.	0.86 (86%)

Table 6: Accuracy of the Classification Models.

According to Table 6, it can be said that I achieved the highest accuracy of above 90% with two models which are Decision Tree Classifier (91%) and Random Forest Classifier (93%).

Below I have provided the confusion matrix of the implemented models:

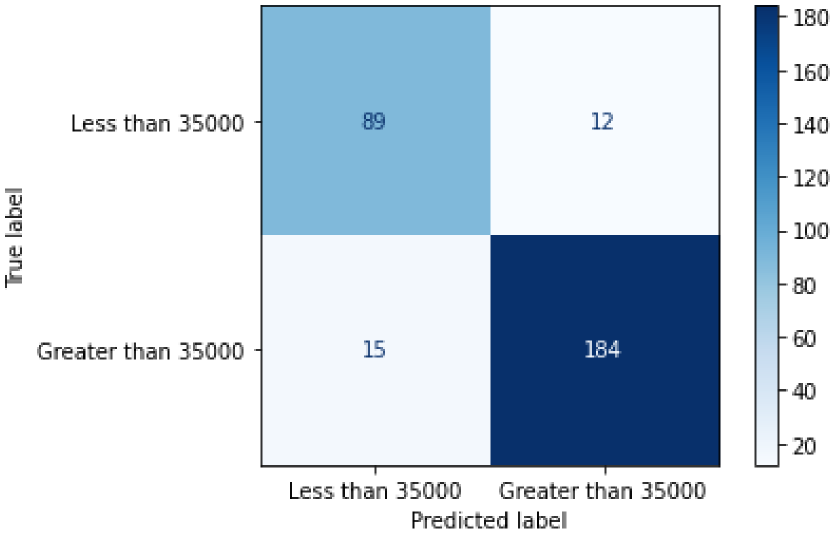


Figure 4: Confusion Matrix of Decision Tree Classifier.

It can be seen in Figure 4, that Decision Tree Classifier predicted 184 salaries as greater than £35000 and 89 salaries as less than £35000 correctly. However, it failed to predict 27 salaries correctly, where the algorithm predicted 15 salaries as less than £35000 but is greater than £35000 and predicted 12 salaries as greater than £35000 but actually is less than £35000.

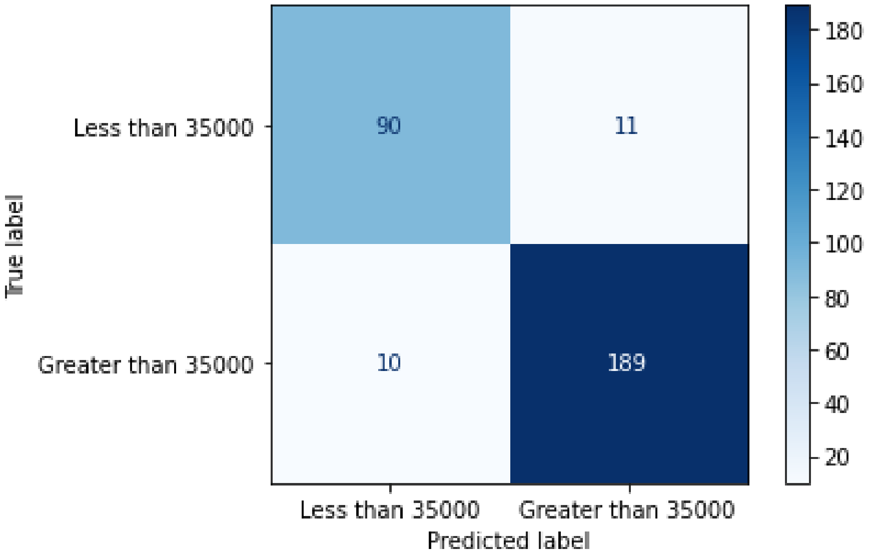


Figure 5: Confusion Matrix of Random Forest Classifier.

It can be seen in Figure 5, that Random Forest Classifier predicted 189 salaries as greater than £35000 and 90 salaries as less than £35000 correctly. However, it failed to predict 21 salaries correctly, where the algorithm predicted 10 salaries as less than £35000 but is greater than £35000 and predicted 11 salaries as greater than £35000 but actually is less than £35000.

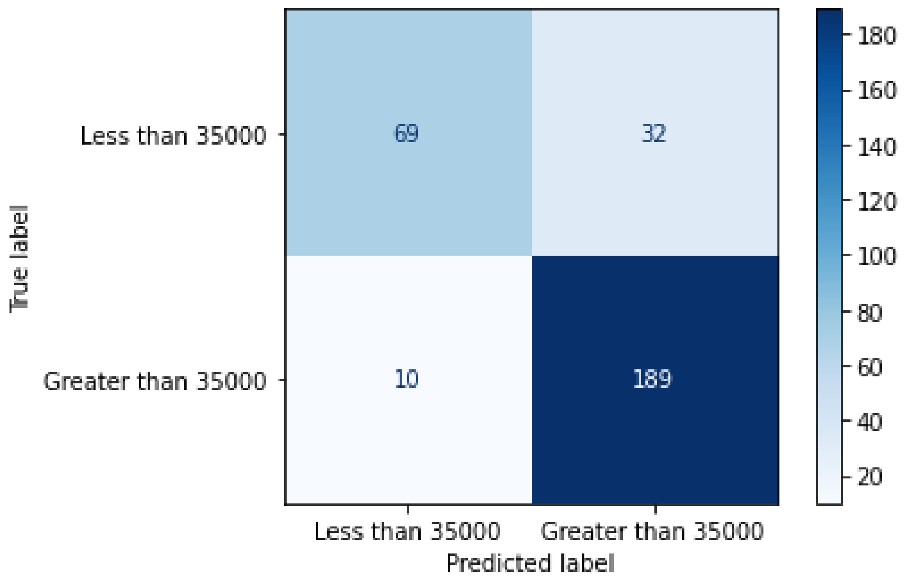


Figure 6: Confusion Matrix of SVM.

It can be seen in Figure 6, that SVM predicted 189 salaries as greater than £35000 correctly and 69 salaries as less than £35000 correctly. However, it failed to predict 42 salaries correctly, where the algorithm predicted 10 salaries as less than £35000 but is greater than £35000 and predicted 32 salaries as greater than £35000 but actually is less than £35000.

4. Neural Networks

A neural network is a technique for instructing computers to process data in a manner that is similar to the way the human brain does. (Anon., 2020)

The Neural Network architecture:

The architecture I used is the Multilayer feed-forward network. In this network, data enters the model through input later, travels through the hidden layers and then goes to output layer.

There can be one or more hidden layers in a network and this layer is essential for a neural network's ability to learn challenging tasks and perform well. The hidden layer takes the sum of weights and bias and passes it through an activation function to get the actual output. The degree to which the input will affect the output is determined by a weight. On the other hand, bias have a constant value of 1. It aids the models in turning the activation function in either a positive or negative direction. The activation functions used depend on the application. (Anon., 2022)

The Hyper parameters and training algorithms used:

1. Regularization Technique: L2. It is used to prevent overfitting or under fitting.
2. The number of hidden layers I used is 1 and have used the Dense layer with the Sigmoid Activation function.
   I did not use more than 1 layer because I got good accuracy with just 1 hidden layer. I have used the Dense layer because it is for general use and is fully connected, feeds forward and also performs the function of finding the output, where output(y)= Activation function (∑ (weights*input + bias))
   As the activation function, I have used "Sigmoid" as I am creating neural network for a binary classification problem and Sigmoid gives a value between 0 and 1.
   By calculating the weighted total and then adding bias to it, the activation function determines whether or not a neuron should be activated. The activation function's objective is to add non-linearity to a neuron's output.
3. I have used Stochastic Gradient Descent(SGD) as optimization algorithm which divides the whole dataset into mini batches (subsets of data) for each epochs (number of iteration), which are randomly chosen. (Anon., 2022) The batch size is the total amount of data points used by a mini-batch.
   For SGD, I have assigned the number of epochs=60, batch size=100 and learning rate=0.05. The learning rate is one of the hyper parameters and is configurable. It controls the model's rate of problem adaptation and is used in the training of neural network.

Building the model and Results:

I have imported all the “tensorflow” libraries needed to construct the neural network model. Then, similar to classification, I have dropped the columns “Region” and “WorkType” from the dataset because they have a negative correlation and then categorized the target “Salary” column into two classes: 1 and 0, where 1 represents those with salary greater than £35000. Next, I have label encoded the non-numerical values to numerical values. After these modifications in the dataset, I have assigned all columns in features except “Salary”, “WorkType” and “Region” and assigned “Salary” as target, after which I have split them into training, validation and testing and have scaled the dataset using Standard Scaler.

I have defined the Regularizer L2 and for the hidden layer, I have defined the Dense layer with the “Sigmoid” activation function. Next I have defined the virtual input, output and the neural network model and have then found the summary of the model to see what the model is comprised of.

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 6)]	0
dense_4 (Dense)	(None, 10)	70
dense_5 (Dense)	(None, 1)	11
Total params: 81 Trainable params: 81 Non-trainable params: 0		

Table 7: Summary of the model

The parameter values in the “param” column in Table 7 is calculated in this way: (Features*Number of nodes in the layer) +bias. So, for the first dense layer, the parameter is (6*10) +10=70.

After defining the accuracy metric, optimization algorithm (Stochastic Gradient Descent(SGD)) and the loss function (Binary cross entropy), I compiled the model. In the next step, I have fitted the model, by assigning 60 epochs and the batch size as 100.

Epochs	loss	binary_accuracy	val_loss	val_binary_accuracy
Epoch 1/60	0.9276	0.4261	0.8838	0.4296
Epoch 2/60	0.8476	0.4627	0.8181	0.4519
Epoch 3/60	0.7989	0.4810	0.7763	0.4963

Table 8: Table showing first 3 out of 60 epochs.

Epochs	loss	binary_accuracy	val_loss	val_binary_accuracy
Epoch 58/60	0.5104	0.8771	0.5278	0.8815
Epoch 59/60	0.5086	0.8771	0.5262	0.8815
Epoch 60/60	0.5067	0.8758	0.5246	0.8815

Table 9: Table showing last 3 out of 60 epochs.

The differences between the values can be seen in Table 8 and 9, that the value of loss is decreasing and the value of accuracy is increasing with each epoch.

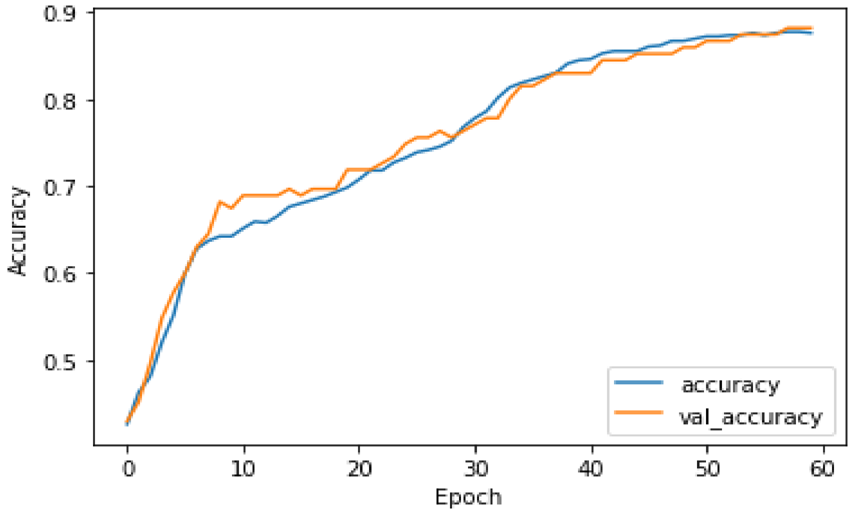
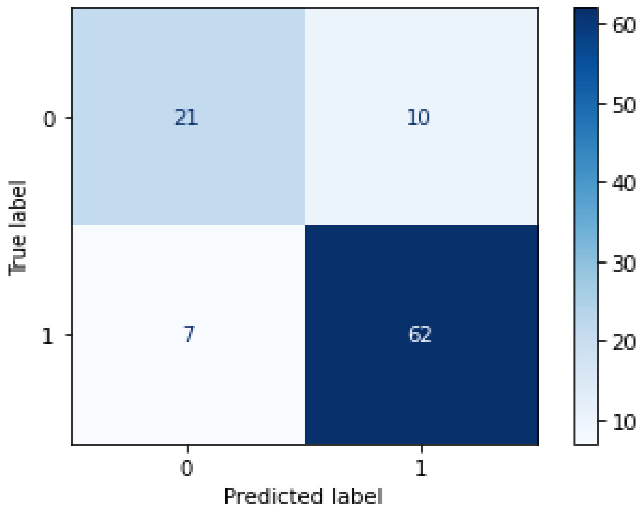


Figure 7: Accuracy.

It can be said from Figure 7, that with each epoch the accuracy is increasing constantly. The validation accuracy and accuracy from trained data is very similar which proves good performance of the model. Also, the model is not over fitting or under fitting, as the accuracy values are not poor and nor is it performing bad compared to training data.

Next, for evaluation I have found out the accuracy on test data with the selected hyper parameters, where I got 80%, which shows that the model is predicting well.

Lastly, I calculated the confusion matrix.



0→Salary < £35000.

1→Salary > £35000.

Figure 8: Confusion Matrix of Neural Network.

It can be seen in Figure 8 that the model identified 62 salaries correctly as greater than £35000 and 21 salaries correctly as less than £35000. However, it has made wrong predictions for 17 salaries where the model predicted 10 salaries as greater than £35000 but actually is less than £35000 and predicted 7 salaries as less than £35000 but is actually greater than £35000.

Observations during test and trial:

When I used a smaller batch size of 90 and 50 epochs, with a lower learning rate of 0.01, this is the result I got.

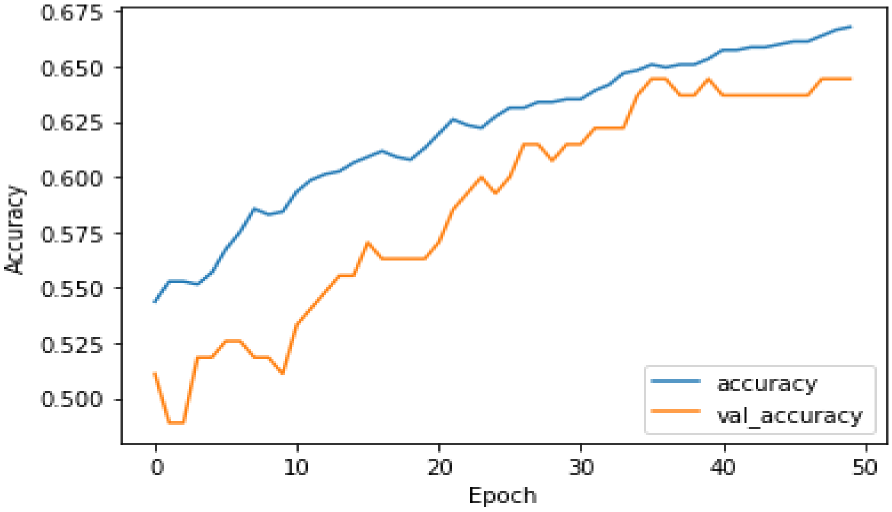


Figure 9: Confusion Matrix of test and trial network.

The accuracy of the model I got is 69%. It can be seen in Figure 9, that the validation accuracy is lower than the trained accuracy which indicates model is not performing well.

Comparison between Neural Network and the final Binary Classification model chosen:

I have evaluated both Neural Network and Binary classification by accuracy. So in comparison, I received 93% accuracy from Binary classification model, whereas I received 80% with Neural Network which clearly shows that Binary Classification model performed better.

5. Clustering

Clustering is known as, grouping a set of items so that they are more similar to one another than to those in other groups. (Anon., 2022)I have performed clustering using the k-means clustering algorithm on the Customers Salary against Age.

I have first imported all the required libraries for K-means clustering. Before starting clustering, I have plotted a scatter plot of Salary against Age to see how it looks like which will aid in choosing the correct number of clusters.

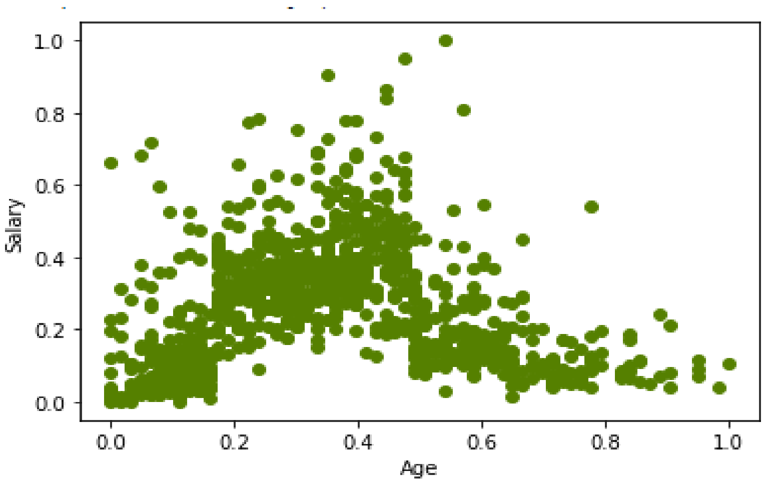


Figure 10: The scatter plot of Salary against Age before clustering.

In Figure 10, it can be said that there are 3 groups, however, the ideal number of clusters (k) cannot be said by just looking at the plot. One of the most popular techniques for finding the value of k is the Elbow Method, where the value of k at the point at which the distortion start reducing in a linear direction, is chosen. A line plot between SSE (Sum of Squared Errors) and the number of clusters(k) is necessary for the elbow method. So, for each iteration from 1 to 10 which are the number of clusters, I have found out the SSE and plotted it.

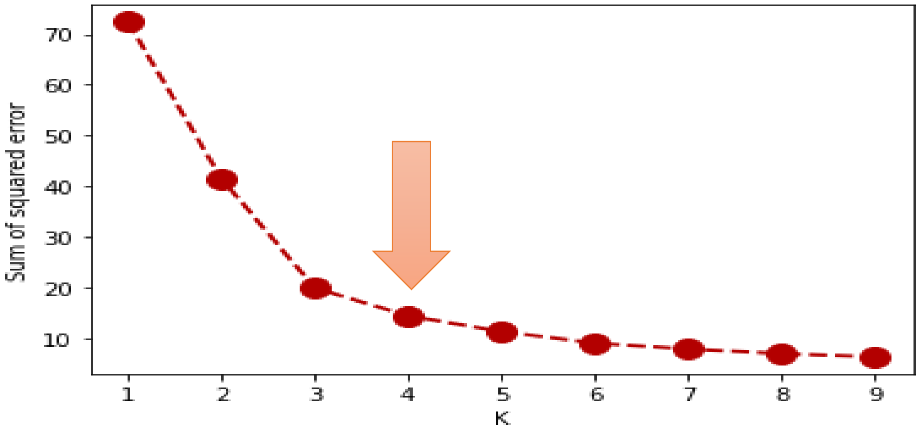


Figure 11: Elbow Method.

Figure 11 shows that k = 4, is the point at which this curve begins to turn. Therefore, it can be said that 4 clusters are the right number.

Next, with 4 clusters I have fitted the K means object, which ran the K-means algorithm on Age and Salary and computed the 4 clusters by assigning different labels for each cluster.

```
array([1, 2, 1, 2, 2, 2, 1, 1, 1, 1, 1, 2, 2, 3, 3, 1, 2, 1, 0, 2, 0, 1, 1, 1, 2, 0, 1, 1, 0, 1, 1, 3, 3, 2,
0, 2, 3, 1, 0, 3, 0, 1, 1, 0, 0, 1, 1, 2, 3, 3, 2, 2, 1, 1, 0, 3, 2, 1, 1, 0, 1, 1, 0, 2, 1, 2, 1, 2, 3, 1,
0, 1, 2, 1, 2, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 3, 1, 3, 0, 1, 0, 0, 1, 1, 1, 2, 2, 1, 1, 0, 0, 1, 0, 1, 1, 2,
0, 2, 2, 1, 0, 0, 0, 1, 3, 3, 0, 1, 0, 2, 2, 0, 0, 3, 3, 0, 0, 1, 2, 0, 2, 1, 1, 2, 2, 0, 2, 0, 1, 1, 3, 1,
1, 3, 2, 0, 0, 2, 0, 0, 0, 2, 2, 2, 0, 2, 1, 1, 3, 1, 0, 3, 1, 0, 2, 2, 2, 1, 2, 1, 1, 0, 2, 0, 2, 3, 1, 2,
1, 0, 1, 0, 1, 1, 0, 0, 1, 2, 0, 2, 0, 3, 1, 1, 0, 3, 1, 0,……………………………………]
```

Table 10: Table showing some of the results of the different labels assigned for salary for each cluster.

The table above shows that the data has been computed to 4 clusters with labels (0,1,2,3). To visualize this better, I have separated the 4 clusters into 4 different data frames and plotted their scatter plots using different colors.

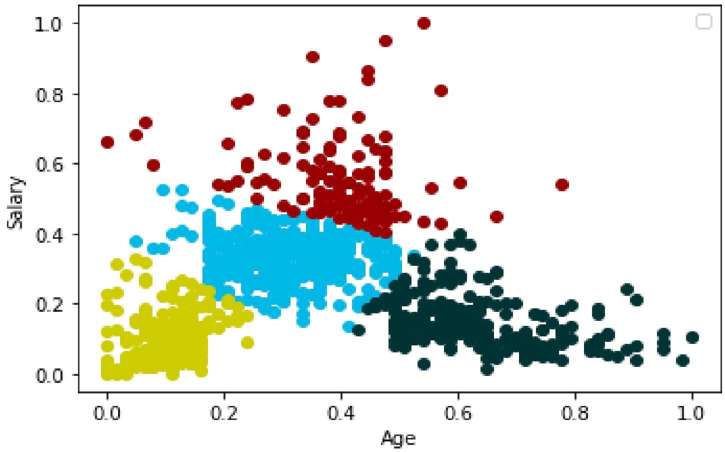


Figure 12: 4 clusters.

Figure 12 shows that it has been separated into 4 clusters.

Finally, the table below shows that values of cluster centres that I have predicted from the closest cluster in each sample.

0.63135593	0.15304672
0.30592758	0.33128716
0.10417448	0.09700514
0.39329806	0.56317918

Table 11: Cluster Centres.

These are the centroids of each cluster. The first row is first centroid where the first column is the x axis and second column is the y axis of the centre points. To visualize the centroids of each cluster, I have plotted a scatter plot.

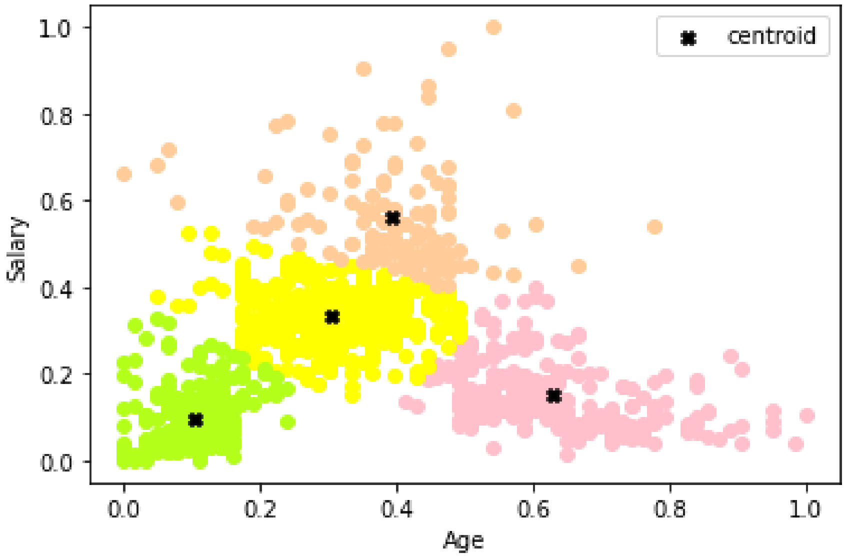


Figure 13: Scatter plot of 4 clusters with its centres.

Figure 13 shows the Centroids of each cluster are marked with a “X”. This is the final result of clustering with 4 clusters according to elbow method. Each data point's cluster label is denoted by a colour.

It can be said that this the best accuracy as it can be seen in Figure 11, that with 4 clusters the error is very low.

6. Conclusion

In conclusion, I have predicted customer’s salary using different machine learning models.

I have implemented 4 regression models, where the models have been trained by existing data to predict salary which are: Linear Regression, Kernel Ridge with kernel “rbf” and “laplacian” and Random Forest Regressor. Amongst them, I got the highest accuracy with Random Forest Regressor of 90% and this is why I chose this regression model.

In order to fit a Binary Classification, I categorized the target Salary into two classes: which are 1 for those with salary greater than £35000 and 0 for those with salary less than £35000. With these categorized target, I have implemented 3 classification models, which are: Decision Tree Classifier, Random Forest Classifier and SVM. Amongst them, I got the highest accuracy with Random Forest Classifier of 93% and this is why I chose this model.

I have designed and trained a neural network to solve the binary classification problem with the categorized target Salary and received an accuracy of 80%. I have worked with a feed forward network which has an input, hidden and an output layer. The hidden layer receives input from input layer and then processes the data received using a set of weights and biases. I used 1 hidden layer as I got good results with it, with activation function “Sigmoid” as it gives a value between 0 and 1. I have used a regularizer to avoid over fitting or under fitting and have used the dense layer which is fully connected and finally, for the optimization algorithm I have used Stochastic Gradient Descent(SGD) with a learning rate of 0.05. In short SGD splits dataset and runs a number of specified iterations(epochs) for optimisation in minibatches.

Lastly I performed clustering, where I used K means clustering and found that 4 clusters best groups the salary through elbow method.

References

Agrawal , S. K., 2021. Metrics to Evaluate your Classification Model to take the right decisions. *Data Science Blogathon,* 20 July.

Agrawal, R., 2021. Know The Best Evaluation Metrics for Your Regression Model !. *Data Science Blogathon* , 19 May.

Tagliaferri, L., 2017. *An Introduction to Machine Learning.* [Online]
Available at: https://www.digitalocean.com/community/tutorials/an-introduction-to-machine-learning
[Accessed 10 December 2022].

Anon., 2020. *What are Neural Networks?.* [Online]
Available at: https://www.ibm.com/cloud/learn/neural-networks
[Accessed 10 December 2022].

Anon., 2022. *Classification in Machine Learning: An Introduction.* [Online]
Available at: https://www.datacamp.com/blog/classification-machine-learning
[Accessed 10 December 2022].

Anon., 2022. *Clustering in Machine Learning.* [Online]
Available at: https://www.geeksforgeeks.org/clustering-in-machine-learning/
[Accessed 16 December 2022].

Anon., 2022. *ML | Stochastic Gradient Descent (SGD).* [Online]
Available at: https://www.geeksforgeeks.org/ml-stochastic-gradient-descent-sgd/
[Accessed 10 December 2022].

Anon., 2022. *Multilayer Feed-Forward Neural Network in Data Mining.* [Online]
Available at: https://www.geeksforgeeks.org/multilayer-feed-forward-neural-network-in-data-mining/
[Accessed 10 December 2022].

Anon., n.d. *Predicting numerical values with regression: Machine Learning in the Elastic Stack [master].* [Online]
Available at: https://www.elastic.co/guide/en/machine-learning/master/ml-dfa-regression.html
[Accessed 10 December 2022].

Anon., n.d. *sklearn.preprocessing.LabelEncoder.* [Online]
Available at: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.LabelEncoder.html
[Accessed 10 December 2022].

Gogia, N., 2019. Why Scaling is Important in Machine Learning?. 8 November.

Gowda, D., 2017. *LinkedIn.* [Online]
Available at: https://www.linkedin.com/pulse/data-shuffling-why-important-machine-learning-how-do-deepak-n-gowda/
[Accessed 10 December 2022].