a. DATA GATHERING AND INTEGRATION

- Used: "https://www.kaggle.com/datasets/samayashar/fraud-detection-transactions-dataset" dataset for this homework problem set.
- It is developed to create robust fraud detection models.
- It contains 50,000 transactions with 21 features
- Includes numerical, categorical, and binary variables.
- The target label for classification is **"Fraud_Label"** (0 = not fraud, 1 = fraud).

b. DATA EXPLORATION

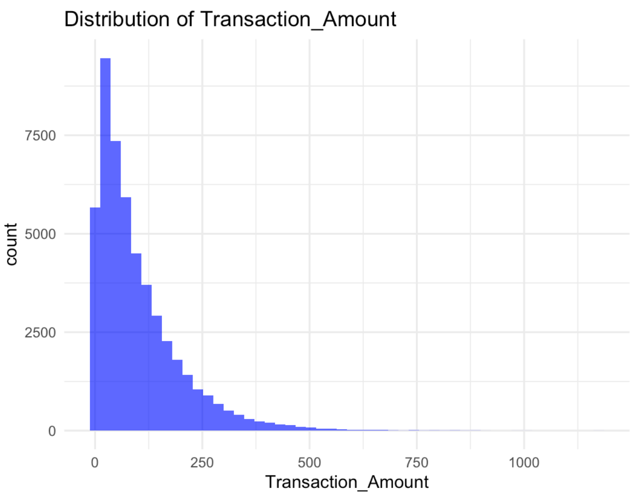
The code includes a summary of the dataset:

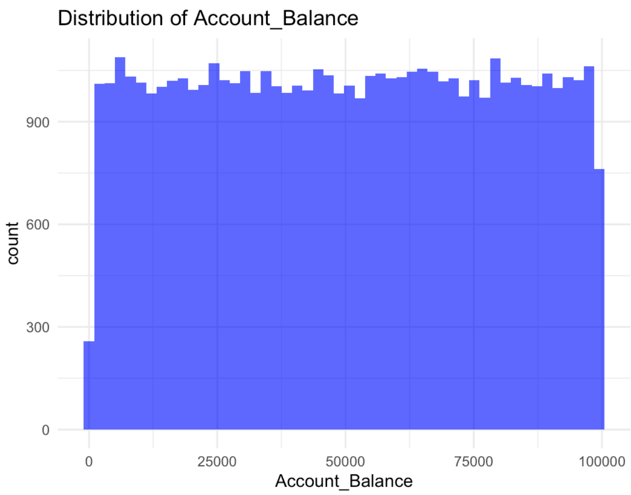
Total missing values in the dataset:

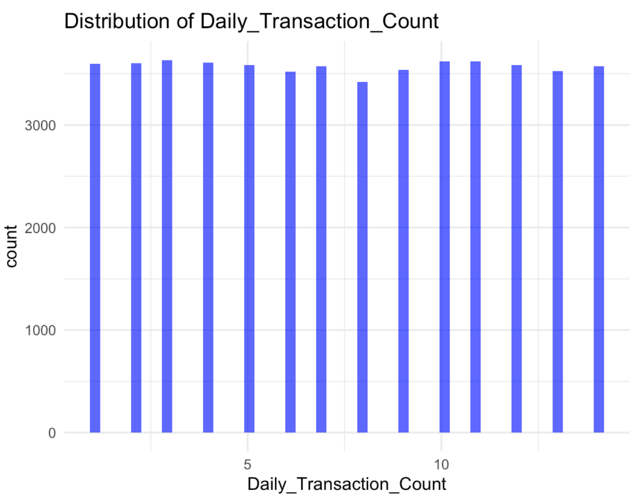
print(paste("Total missing values:", total_missing_values))
[1] "Total missing values: 0"

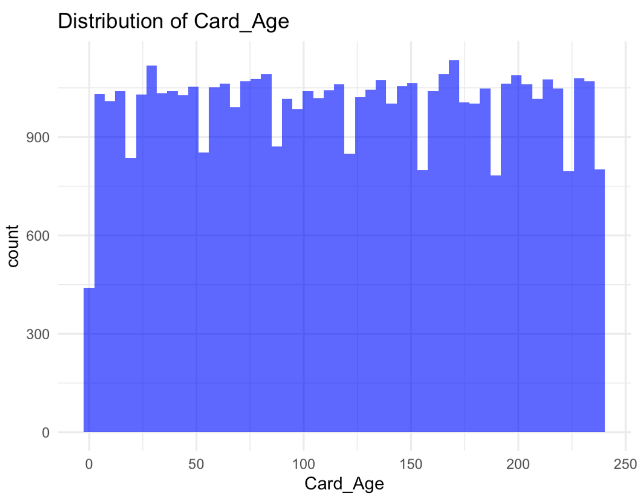
This was great as I did not have to spend time on filling in missing data.

Histograms for some of the numerical variables:
(R-Code file includes graphs for all numrical variables)

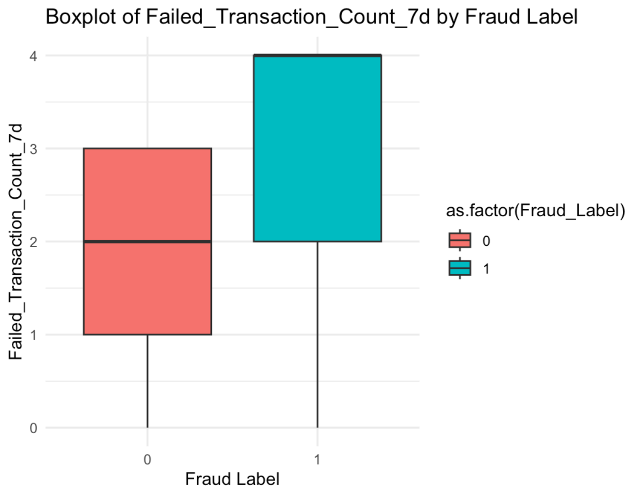


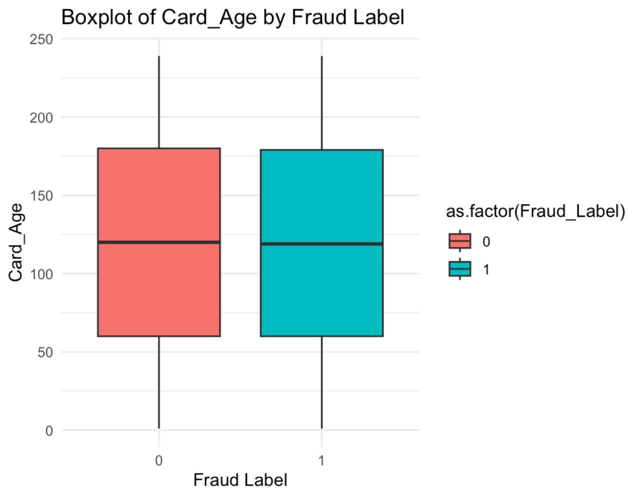


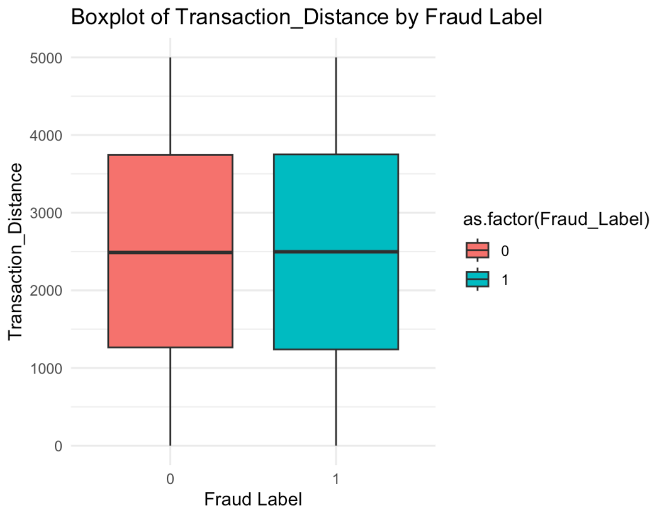




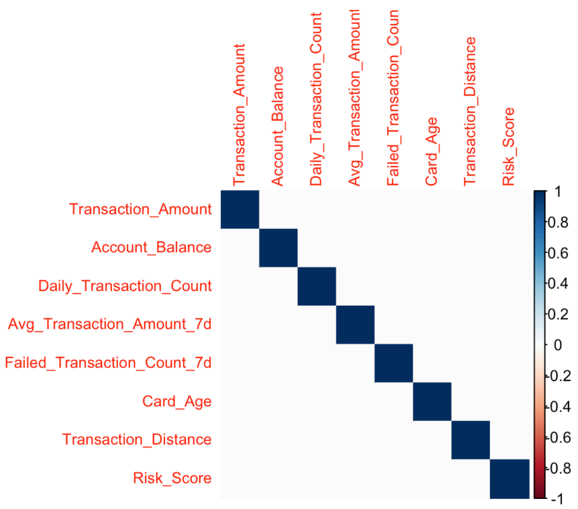
4 Boxplots to examine outliers by fraud label:
(R-Code file includes graphs for all numrical variables)



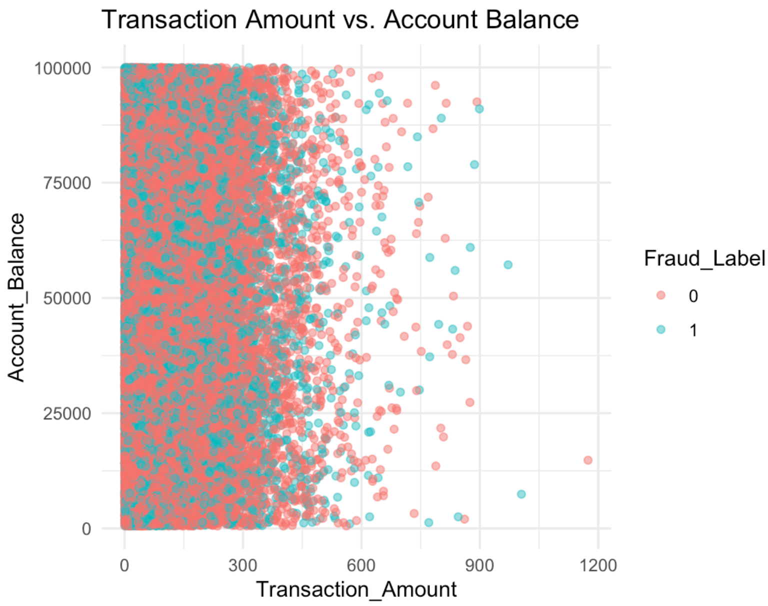




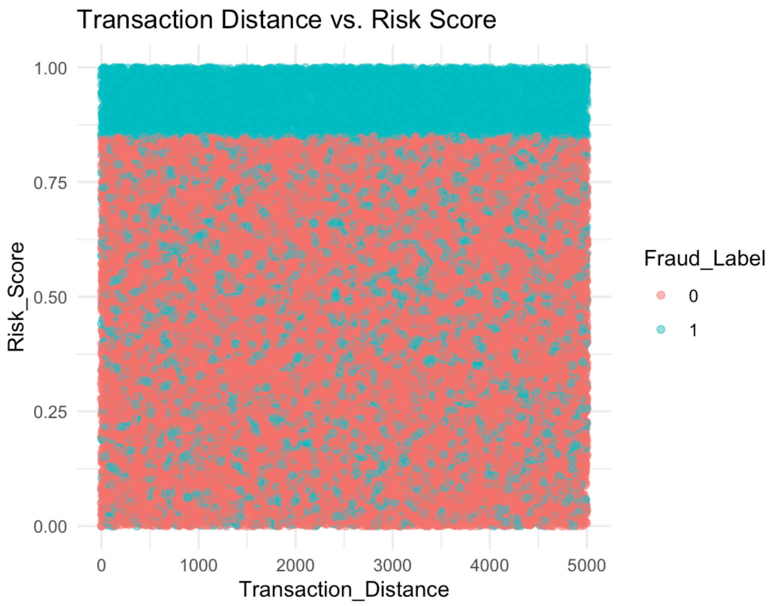
Correlation Matrix (gives me an error I can't fix):



4 Relationships between numerical variables:



Scatterplot to trends in fraudulent transactions.



Scatterplot to determine if high-risk scores correlate with longer distances.

Box plot to detecting fraud patterns in transaction types.

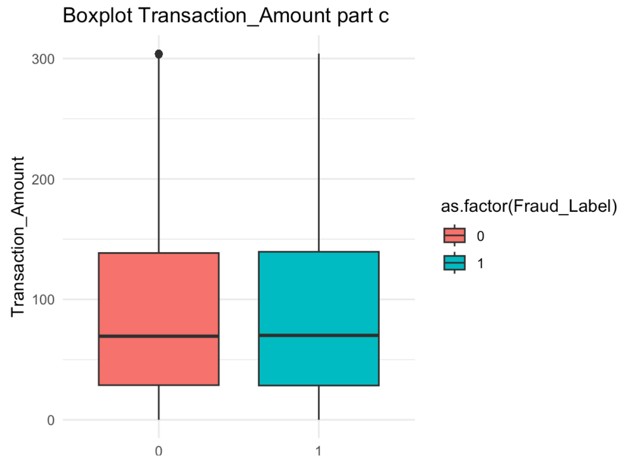
Density plot if higher risk scores correlate with fraud

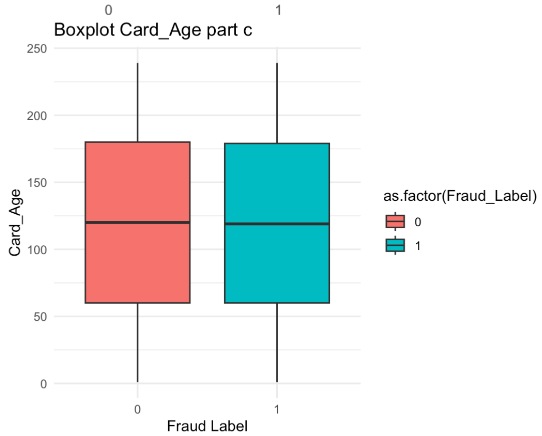
c. DATA CLEANING

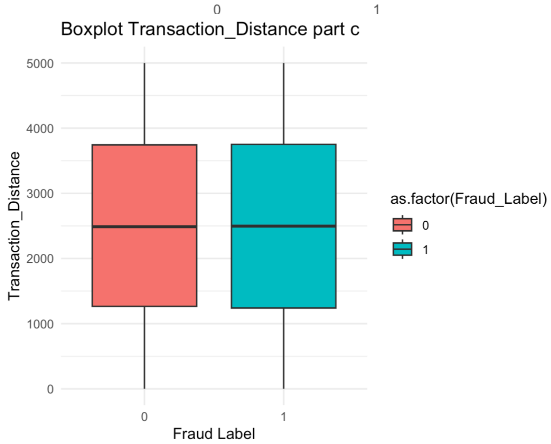
As mentioned in part b, there are no missing values.

- IQR method I used to detect and handle outliers
- Categorical variables are cleaned
- Numerical features are standardised
- Creates dummy variables for categorical variables if needed

4 Boxplot using same variables as part b as evidence for the outliers being handled:
(R-Code file includes graphs for all numrical variables)







Based on the nature of the dataset and the future questions asked from this homework set:

- Transaction_ID and User_ID have been removed as they are unique identifiers with no data prediction value.
- Timestamp can be useful to perform high-level time-based trends; however, it will not be used for classification and hence was dropped.

Final summary of the dataset included in the R-Code

d. DATA PREPROCESSING

Based on the data, preprocessing is neccesary as:

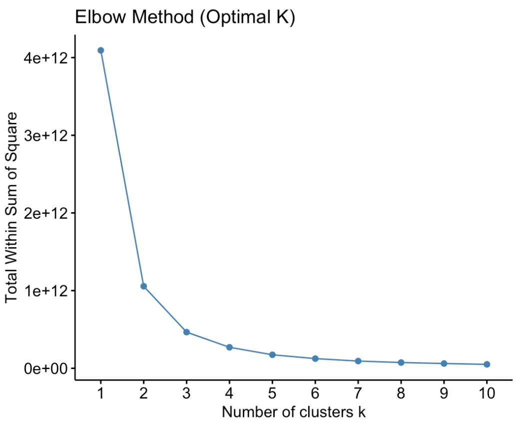
- Categorical variables need to be changed to dummy variables to create ML models, as they require numerical input.
- Numerical features are standardised for clustering as they need to be scaled to perform better
- Removing irrelevant features allows us to work with more precise and relevant datasets which helps not to complicate the process of creating ML models.

Based on the data, preprocessing is unnecessary as:

- The dataset includes no missing values, and there is no point in handling missing values.
- Binning might reduce the importance of numerical features such as Transaction Amount
- Smoothing is not required as the data contains no major random noise.

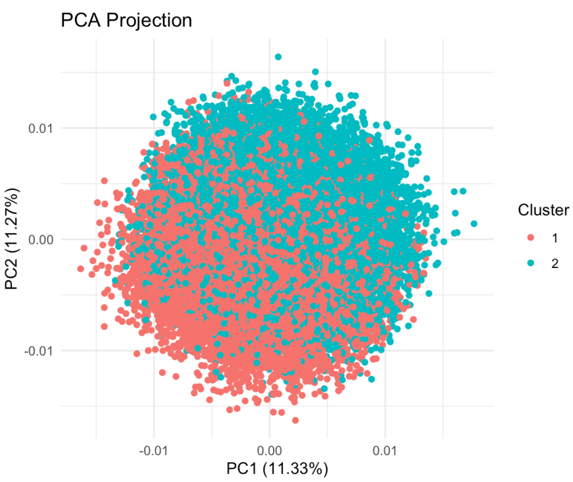
e. CLUSTERING

Elbow Method Graph:

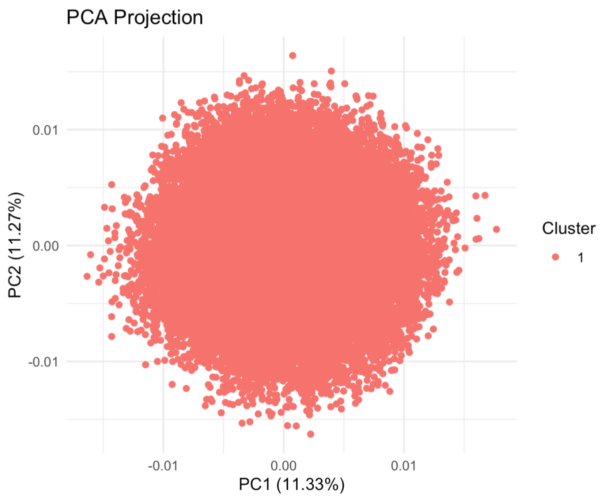


This graph suggests that the best k value = 1. Which is problematic as that means all the data is just one cluster.

For the sake of discussion:



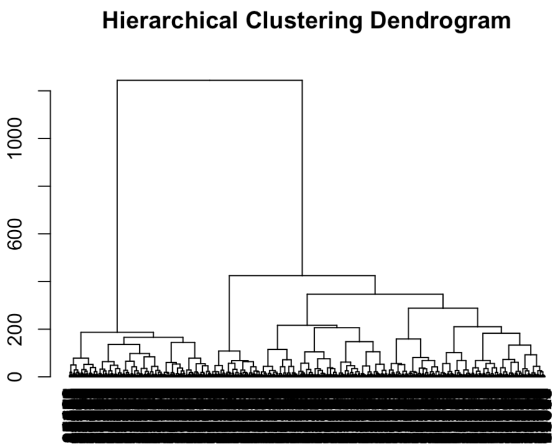
Taking k=2 + PCA Projection further explain that all the data is just one cluster.



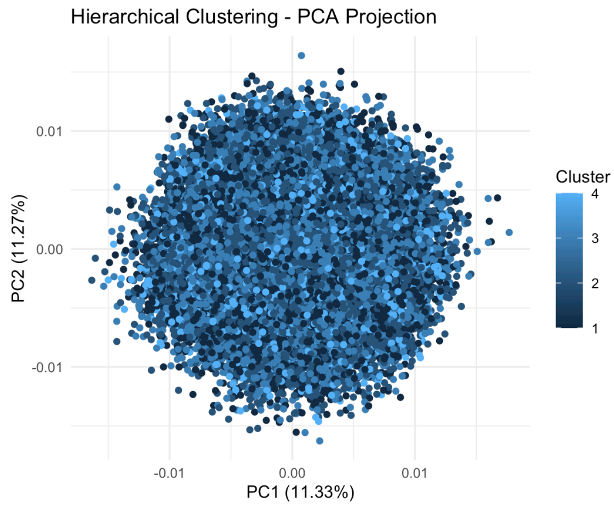
When k = 1, the cluster makes more sense. This suggests that clustering did not find distinct patterns.

Using HAC instead.

Determining k=4 based on:



Also gives a spherical shape as a result



Conclusion on clustering:
- The datasert forms a dense circle cause any clusters mean nothing.
- This suggests that the dataset might not have any useful grouping we are trying to find.
- Alternatively, density-based clustering might be the next best thing to try.

f. CLASSIFICATION

Chose SVM as:
- Works well with high-dimensional data
- Using radial kernel to capture non-linear relationships
- Allows good generalisation

Chose Decision Trees as:
- Fast and Easy to interpret
- Automatically emphasises the important features
- Good for non-linear data

ACCURACY PERCENTAGES FOR BOTH MODELS:

```
print(paste("SVM Accuracy:", round(svm_accuracy * 100, 2), "%"))
[1] "SVM Accuracy: 99.1 %"
> print(paste("Decision Tree Accuracy:", round(dt_accuracy * 100, 2), "%"))
[1] "Decision Tree Accuracy: 100 %"
```

AUC FOR BOTH THE MODELS:

```
print(paste("SVM AUC:", round(svm_auc, 4)))
[1] "SVM AUC: 0.9904"
> print(paste("Decision Tree AUC:", round(dt_auc, 4)))
[1] "Decision Tree AUC: 1"
```

SVM CONFUSION MATRIX:

```
print("SVM Confusion Matrix:")
[1] "SVM Confusion Matrix:"
> print(svm_cm)
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 6732   36
         1   54 3177

               Accuracy : 0.991
                 95% CI : (0.9889, 0.9928)
    No Information Rate : 0.6787
    P-Value [Acc > NIR] : < 2e-16

                  Kappa : 0.9794

 Mcnemar's Test P-Value : 0.07314

            Sensitivity : 0.9920
            Specificity : 0.9888
         Pos Pred Value : 0.9947
         Neg Pred Value : 0.9833
             Prevalence : 0.6787
         Detection Rate : 0.6733
   Detection Prevalence : 0.6769
      Balanced Accuracy : 0.9904

       'Positive' Class : 0
```

DECISION TREES CONFUSION MATRIX:

```
print("Decision Tree Confusion Matrix:")
[1] "Decision Tree Confusion Matrix:"
> print(dt_cm)
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 6786    0
         1    0 3213

               Accuracy : 1
                 95% CI : (0.9996, 1)
```

No Information Rate : 0.6787
P-Value [Acc > NIR] : < 2.2e-16

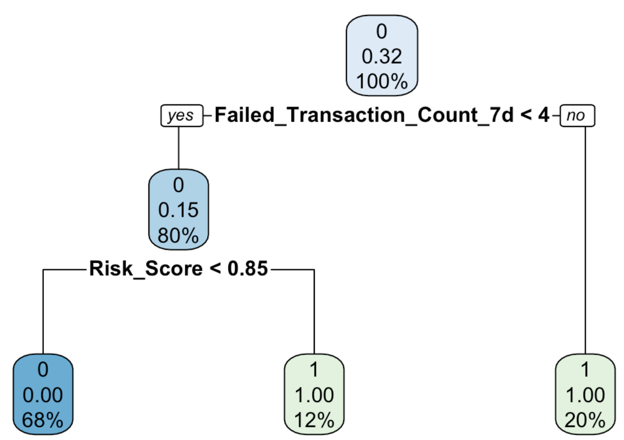
Kappa : 1

Mcnemar's Test P-Value : NA

Sensitivity : 1.0000
Specificity : 1.0000
Pos Pred Value : 1.0000
Neg Pred Value : 1.0000
Prevalence : 0.6787
Detection Rate : 0.6787
Detection Prevalence : 0.6787
Balanced Accuracy : 1.0000

'Positive' Class : 0

Decision Tree for Fraud Prediction



g. EVALUATION

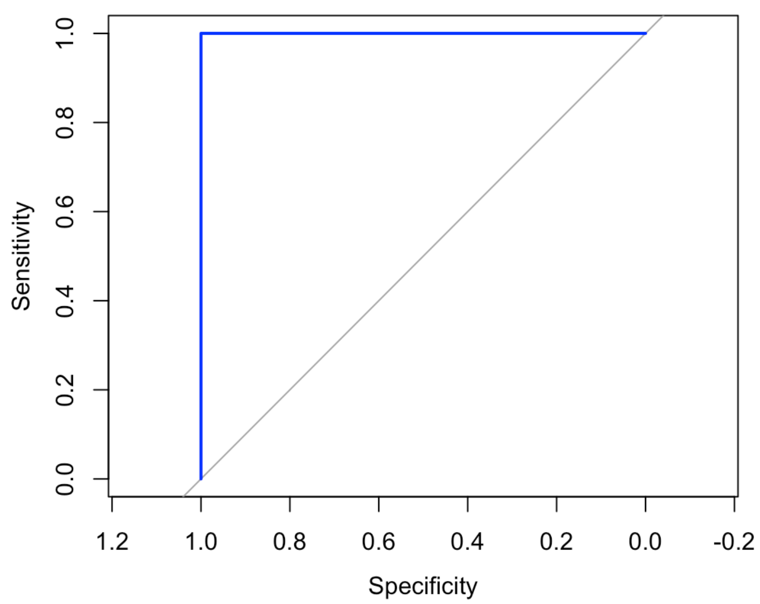
print(paste("Precision:", round(precision, 4)))
[1] "Precision: 1"
> print(paste("Recall:", round(recall, 4)))
[1] "Recall: 1"

Precision, Recall and AUC all are 1

print(paste("AUC:", round(auc_value, 4)))
[1] "AUC: 1"

The evaluation above suggests that the Decision Tree perfectly classifies transactions.
While this sounds great, it is not very realistic and can be a sign of overitting.

ROC Curve



DATA PRUNING:

After data pruning:

AUC:
print(paste("AUC After Fixing Overfitting:", round(auc_value, 4)))
[1] "AUC After Fixing Overfitting: 1"

Confusion Matrix after pruning:

print("Confusion Matrix After Fixing Overfitting:")
[1] "Confusion Matrix After Fixing Overfitting:"
> print(dt_cm)
Confusion Matrix and Statistics

	Reference	
Prediction	0	1
0	6786	0
1	0	3213

Accuracy : 1

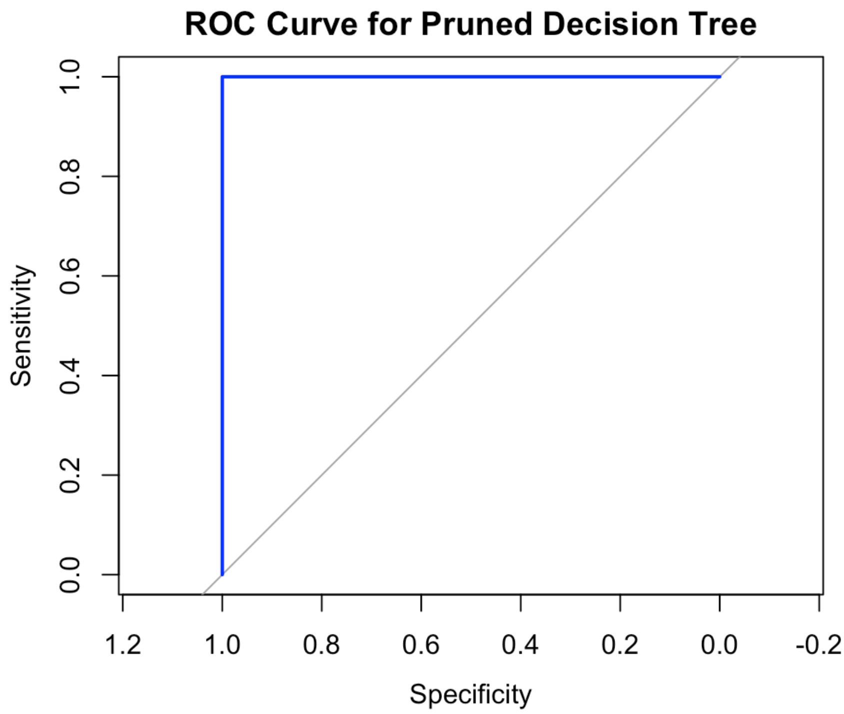
95% CI : (0.9996, 1)
No Information Rate : 0.6787
P-Value [Acc > NIR] : < 2.2e-16

Kappa : 1

Mcnemar's Test P-Value : NA

Sensitivity : 1.0000
Specificity : 1.0000
Pos Pred Value : 1.0000
Neg Pred Value : 1.0000
Prevalence : 0.6787
Detection Rate : 0.6787
Detection Prevalence : 0.6787
Balanced Accuracy : 1.0000

'Positive' Class : 0



Conclusion on classification:
The 100% accuracy AFTER pruning the model suggests that the dataset is too simple.
Frauds are easy to separate.
This is great for the homework set but realistically will not be the case in real-life case studies.

h. REPORT

Based on everything included in this report:

The dataset:
- Is highly structured

Based on the plots from part b fraudulent cases:
- Had a higher risk score
- Had distinct transaction amounts and balances

i. REFLECTION

I enjoyed the coding and the hands-on case studies we tackled for every homework set as a whole and enjoyed learning about the fundamentals of the data pipeline as a whole. I have come to the realization that I personally really enjoy data visualization and take a great interest in ensuring that data is presented clearly so that even a non tech-savvy person can easily interpret what the data is trying to say. However, as I am not great with memorisation, I did have a bit of an issue with the theoretical section of this course and felt stressed about the final exam.