Abinesh_Acharya_l5_AML_1st_report.docx

by Abinesh Acharya

Submission date: 24-Mar-2025 11:28PM (UTC+0545)
Submission ID: 2623754420
File name: Abinesh_Acharya_l5_AML_1st_report.docx (1M)
Word count: 1951
Character count: 11242

School of Computing, Creative Technology and Engineering

Student ID	c7466771
Student Name	Abinesh Acharya
Module Name & CRN	“Applied Machine Learning-18910”
Level	5
Assessment Name & Part No.	Case Study-1:Bank Marketing Campaign, PART-I
Project Title	Bank Marketing
Date of Submission	3/24/2025
Course	BSc. (Hons) Computing
Academic Year	2025

List of Figures:

PART I

Bank Marketing

Abinesh Acharya, BSc (hons) Computing, 2025

Introduction:

The Bank Marketing dataset uses information from a Portuguese bank's phone call campaigns to predict if a customer will sign up for a team deposit. Companies depending more and more on strategies based on data to effectively use resources, target the right clients, and improve personal services, this problem is very relevant today. Accurate predictions of how customers act increase the success rates of marketing campagins while also saving time and money. The dataset is an excellent resource for creating models with predictive capabilities since it provides useful data about past interactions, financial information, and demographics of clients. Predicting client interest in financial products, making online shopping recommendations, identifying clients who are predicted to stop using communications services, identifying patients who require additional care in the medial field, and help educational institutions in supporting struggling students are some examples of similar real-world applications. This dataset demonstrates how data can improve decision making and deal with actual problems.

Literature review:

Machine learning has proven to be an essential tool in addressing customer behavior prediction problems across various industries, including banking, telecommunications, insurance, and retail. Moro et al. (2011) conducted a foundational study using the same Bank Marketing dataset, applying techniques such as Decision Trees, Random Forests, and Neural Networks, with the latter achieving an accuracy of 92%. Their research emphasized call duration as a key predictor for subscription. Similarly, studies like Chauhan et al. (2020) focused on customer retention in banking, achieving an AUC of 89% with Logistic Regression, highlighting the importance of behavioral and demographic data. In loan default prediction, Kumar et al. (2018) demonstrated the effectiveness of Random Forest models, achieving 94% accuracy, underscoring the importance of payment history. Johnson et al. (2016) and Smith et al. (2019) explored churn prediction in telecom, leveraging techniques such as XGBoost and neural networks, with recall rates as high as 95%. Insights from Patel et al. (2021) and Martinez et al. (2020) further reinforce the relevance of historical trends and income in financial predictions. Clustering techniques from Kim et al. (2017) and Sharma et al. (2023) showcased how segmentation supports targeted marketing and insurance policy renewals, respectively. Wu et al. (2022) contributed by balancing revenue optimization in retail settings using Decision Trees. These studies collectively reveal that both traditional statistical methods and advanced machine learning algorithms play pivotal roles in predicting customer behaviors, offering insights and techniques that align well with the goals of the Bank Marketing dataset analysis.

Exploratory Data Analysis

4

Records from phone-based marketing campaigns run by a Portuguese financial institution make up the dataset. Predicting whether a client will sign up for a term deposit is the main objective. With 45,211 rows and 17 variables each, the train and test datasets have the same column structures.

```
'data.frame':   45211 obs. of  17 variables:
 $ age      : int  58 44 33 47 33 NA 28 42 58 43 ...
 $ job      : chr  "management" "technician" "entrepreneur" "blue-collar" ...
 $ marital  : chr  "married" "single" "married" "married" ...
 $ education: chr  "tertiary" "secondary" "secondary" "unknown" ...
 $ default  : chr  "no" "no" "no" "no" ...
 $ balance  : int  2143 29 2 1506 NA 231 447 2 121 593 ...
 $ housing  : chr  "yes" "yes" "yes" "yes" ...
 $ loan     : chr  "no" "no" "yes" "no" ...
 $ contact  : chr  "unknown" "unknown" "unknown" "unknown" ...
 $ day      : int  5 NA 5 5 5 5 5 5 5 5 ...
 $ month    : chr  "may" "may" "may" "may" ...
 $ duration : int  261 151 NA 92 198 139 217 380 50 55 ...
 $ campaign : int  1 1 1 1 1 1 1 1 1 1 ...
 $ pdays    : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ previous : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutcome : chr  "unknown" "unknown" "unknown" "unknown" ...
 $ y        : chr  "no" "no" "no" "no" ...
> str(train)
'data.frame':   45211 obs. of  17 variables:
 $ age      : int  58 44 33 47 33 NA 28 42 58 43 ...
 $ job      : chr  "management" "technician" "entrepreneur" "blue-collar" ...
 $ marital  : chr  "married" "single" "married" "married" ...
 $ education: chr  "tertiary" "secondary" "secondary" "unknown" ...
 $ default  : chr  "no" "no" "no" "no" ...
 $ balance  : int  2143 29 2 1506 NA 231 447 2 121 593 ...
 $ housing  : chr  "yes" "yes" "yes" "yes" ...
 $ loan     : chr  "no" "no" "yes" "no" ...
 $ contact  : chr  "unknown" "unknown" "unknown" "unknown" ...
 $ day      : int  5 NA 5 5 5 5 5 5 5 5 ...
 $ month    : chr  "may" "may" "may" "may" ...
 $ duration : int  261 151 NA 92 198 139 217 380 50 55 ...
 $ campaign : int  1 1 1 1 1 1 1 1 1 1 ...
 $ pdays    : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ previous : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutcome : chr  "unknown" "unknown" "unknown" "unknown" ...
 $ y        : chr  "no" "no" "no" "no" ...
```

"Figure 1: Structure of dataset test and train"

Both datasets structures are displayed above using str(test) and str(train). There are 17 variables in both datasets, including both categories and numbers. Age, balance, and duration are the numerical variables. Education, marital and job are the categorical variables. There are missing values(NA) in some numerical columns. Furthermore, unknown entries for certain categorical variables, such as job and education which indicates uncertainty or not enough information.

```
> combine <- rbind(test, train)
```

"Figure 2: Structure of combined dataset"

The rbind() method was used to combine the two datasets into a single dataset called combined in figure 2. The "combine" dataset is a combination of the two previously stated datasets. str(combine) is used to display the structure of the combined dataset. Every row in the combined dataset originates from either the test or the train dataset. Added column called dataset, which allows to distinguish between the test and train dataset rows by which the unified dataset can be easily identified and separated according to their original source.

```
> str(combine)
'data.frame':	90422 obs. of  18 variables:
 $ age      : int  58 44 33 47 33 NA 28 42 58 43 ...
 $ job      : chr  "management" "technician" "entrepreneur" "blue-collar" ...
 $ marital  : chr  "married" "single" "married" "married" ...
 $ education: chr  "tertiary" "secondary" "secondary" "unknown" ...
 $ default  : chr  "no" "no" "no" "no" ...
 $ balance  : int  2143 29 2 1506 NA 231 447 2 121 593 ...
 $ housing  : chr  "yes" "yes" "yes" "yes" ...
 $ loan     : chr  "no" "no" "yes" "no" ...
 $ contact  : chr  "unknown" "unknown" "unknown" "unknown" ...
 $ day      : int  5 NA 5 5 5 5 5 5 5 5 ...
 $ month    : chr  "may" "may" "may" "may" ...
 $ duration : int  261 151 NA 92 198 139 217 380 50 55 ...
 $ campaign : int  1 1 1 1 1 1 1 1 1 1 ...
 $ pdays    : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 ...
 $ previous : int  0 0 0 0 0 0 0 0 0 0 ...
 $ poutcome : chr  "unknown" "unknown" "unknown" "unknown" ...
 $ y        : chr  "no" "no" "no" "no" ...
 $ dataset  : chr  "test" "test" "test" "test" ...
>
```

"Figure 3: Summary of two combined datasets"

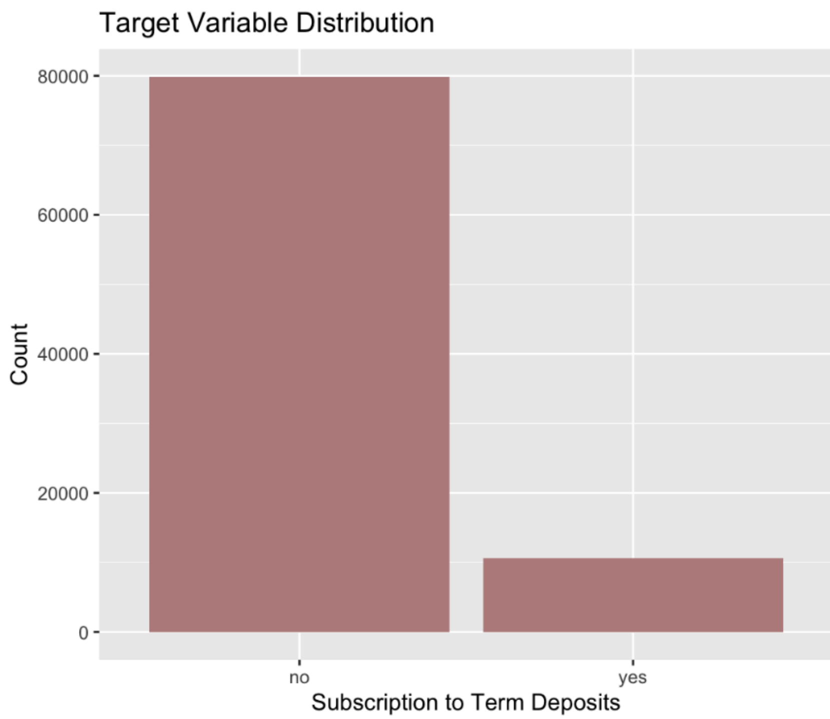
The function dim(combine) returns the datasets dimensions. A snapshot of its structure is given by the str() function, which displays the data types of each variable. Key statistics, including the minimum, maximum, mean, median and quartiles for numerical variables and counts for categorical ones, are provided via the summary() function. The distribution of various attributes in the dataset may be better understood by using the table() function to count the frequency of unique values in categorical columns such as job, education, and marital.

- Visualization

1. Target Variable Distribution

```
# Visualize features
library(ggplot2)
ggplot(combine, aes(x = y)) +
  geom_bar(fill = "rosybrown") +
  labs(title = "Target Variable Distribution", x = "Subscription to Term Deposits", y = "Count")
```

"Figure 4: Code snippet of Target Variable Distribution"



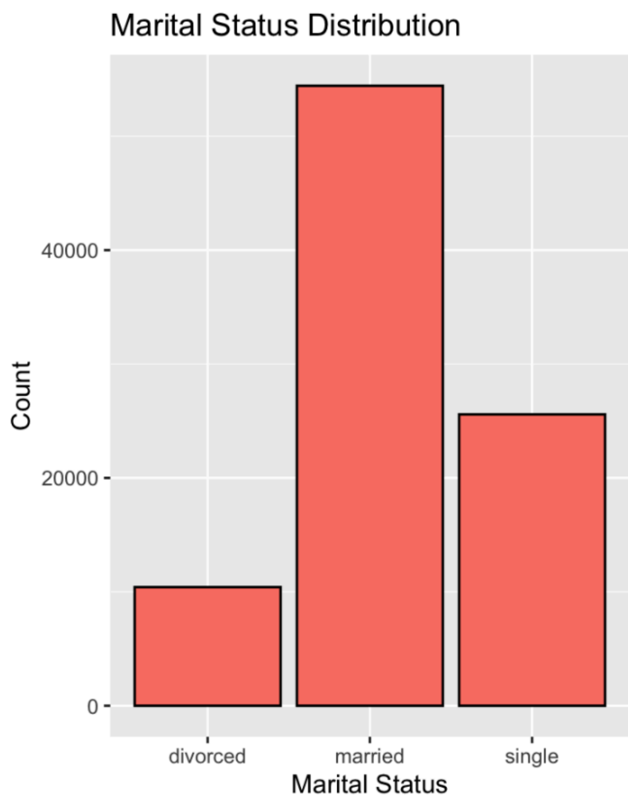
"Figure 5: Target Variable Distribution"

The target variable y shows if a client signed up for a term deposit (yes/no). The overall distribution of classes is shown in the bar plot. A glaring imbalance can be seen in the graph, as many more clients chose not to subscribe ("no") than did so ("yes").

2. Marital Status Distribution

```
ggplot(combine, aes(x = marital)) +
  geom_bar(fill = "salmon", color = "black") +
  labs(title = "Marital Status Distribution", x = "Marital Status", y = "Count")
```

"Figure 6: Code snippet to Marital Status Distribution"



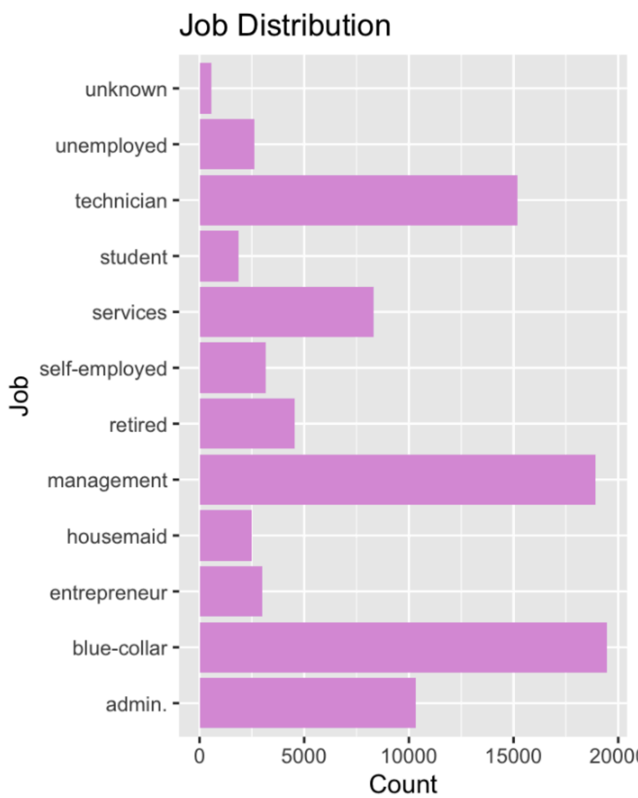
“Figure 7: Marital Status Distribution”

One important demographic factor that could affect client behaviour is marital status. An overview of the distribution across categories is given by the bar plot. The plot points out that married clients make up the majority, followed by single and divorced clients. This breakdown of demographics can offer important information about customer segmentation.

3. Job Distribution

```
ggplot(combine, aes(y = job)) +
  geom_bar(fill = "plum") +
  labs(title = "Job Distribution", x = "Count", y = "Job")
```

“Figure 8: Code snippet of Job Distribution”



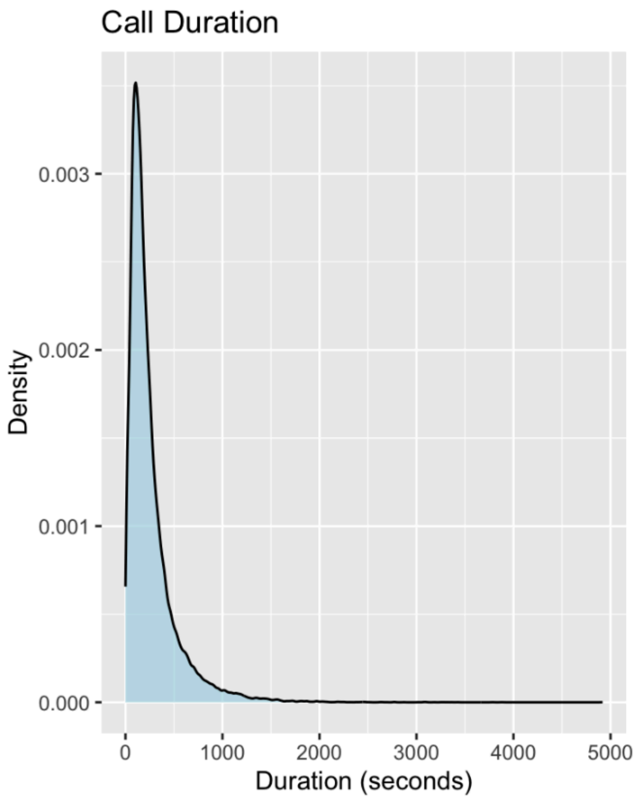
"Figure 9: Job Distribution"

Knowing the clients' jobs makes it easier to determine whether occupation has an effect on subscription rates. The frequency of job types is shown in the horizontal bar plot. Blue-collar, management, and technician jobs are the most common, according to the graph, while student and unknown jobs are less common.

4.Call Duration Distribution

```
ggplot(combine, aes(x = duration)) +
  geom_density(fill = "lightblue", alpha = 0.7, na.rm = TRUE) +
  labs(title = "Call Duration", x = "Duration (seconds)", y = "Density")
```

"Figure 10: Code snippet of Call Duration Distribution"



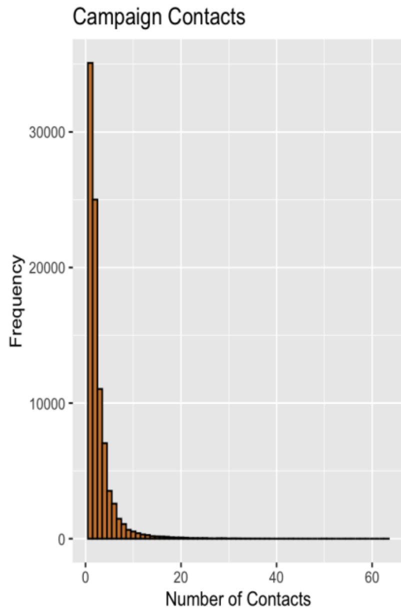
"Figure 11: Call Duration Distribution"

A density plot is used for analysis of the duration variable, which shows the duration of the most recent call in seconds. Higher levels of engagement are frequently associated with longer calls. The density plot indicates a significant skew, with the majority of calls lasting less than 100 seconds. A few calls, though, go over 1000 seconds, which could indicate outliers.

5.Campaign Contacts Distribution

```
ggplot(combine, aes(x = campaign)) +
  geom_histogram(binwidth = 1, fill = "peru", color = "black") +
  labs(title = "Campaign Contacts", x = "Number of Contacts", y = "Frequency")
```

"Figure 12: Code snippet of Campaign Contacts Distribution"



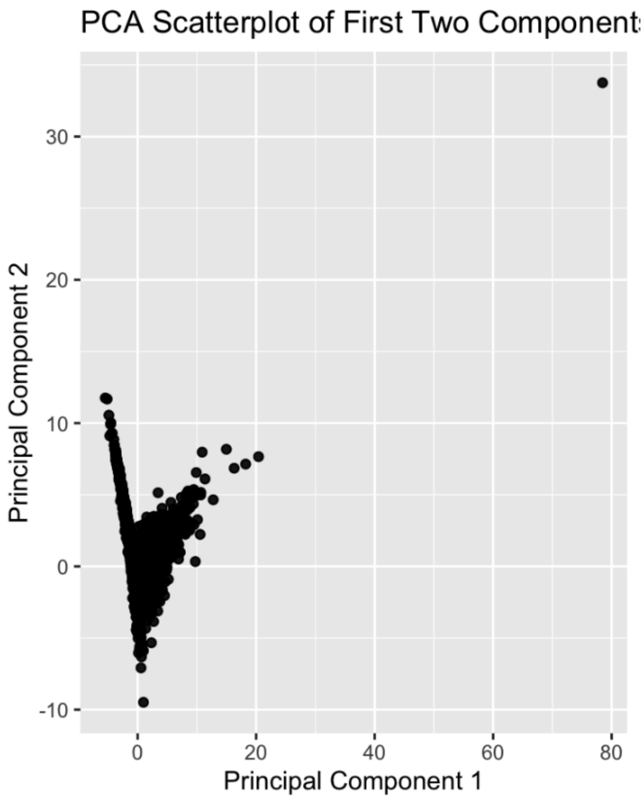
"Figure 13: Campaign Contacts Distribution"

The distribution of contacts made during the campaign is displayed by the histogram. Most clients were contacted only a few times, with one or two contacts being the most common. Only a small number of clients were contacted more than 20 times, indicating that these are outliers, and the frequency drops off significantly as the number of contacts rises.

6.Principal Component Analysis (PCA)

```
# Principal Component Analysis (PCA)
numeric_data <- combine[sapply(combine, is.numeric)]
scaled_data <- scale(numeric_data)
scaled_data <- na.omit(scaled_data)
pca_result <- prcomp(scaled_data, center = TRUE, scale. = TRUE)
summary(pca_result)
```

"Figure 14:Code snippet for PCA"



"Figure 15: PCA Scatterplot"

PCA is used to scale numerical data and examine feature relationships. This method maintains the highest variance while reducing dimensionality. A scatter plot is made for the first two components (PC1 and PC2) in order to visualise the PCA results. Clusters of points in the scatter plot represent groups of clients with comparable PCA dimensions.

Data Pre-processing

1. Missing Values

```r
# Handle Missing Values
missing_summary <- colSums(is.na(combine)) / nrow(combine) * 100
print(missing_summary)
library(mice)
imputed_data <- mice(combine, method = "pmm", m = 1, maxit = 5, seed = 500)
combine <- complete(imputed_data)
```

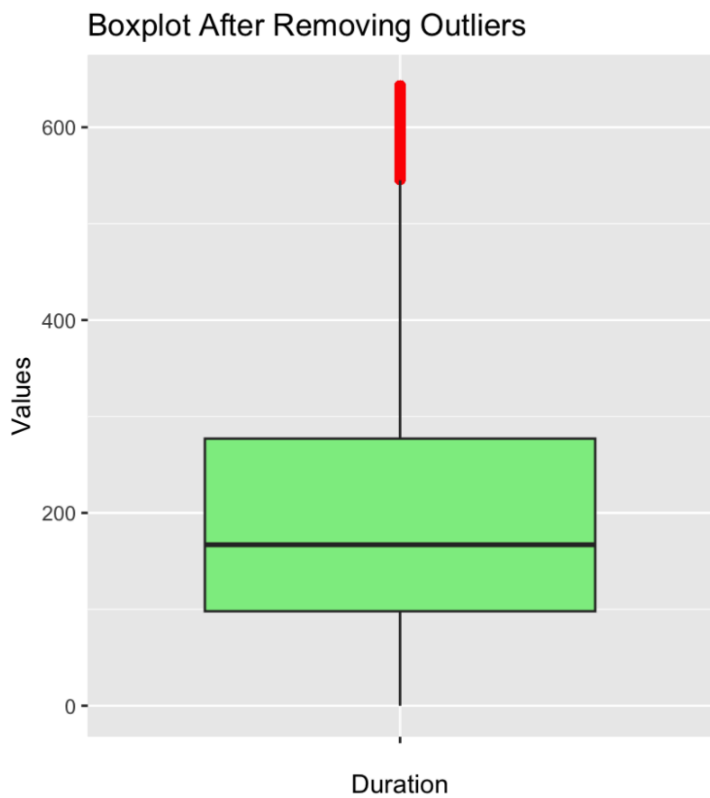
"Figure 16:Handling Missing Values"

Finding and dealing with missing values is the first stage. The Predictive Mean Matching (PMM) method from the mice package is used to impute missing values after calculating the percentage of missing values in each column. By creating believable substitutes for missing data, this method preserves the dataset's integrity for subsequent analysis. The dataset is complete and free of missing values following imputation.

2. Outliers

```r
# Detect Outliers
Q1 <- quantile(combine$duration, 0.25, na.rm = TRUE)
Q3 <- quantile(combine$duration, 0.75, na.rm = TRUE)
IQR <- Q3 - Q1
lower <- Q1 - 1.5 * IQR
upper <- Q3 + 1.5 * IQR
combine <- combine[combine$duration >= lower & combine$duration <= upper, ]
```

"Figure 17: Outline Detecting and Removing"



"Figure 18:Outlier detection"

The Interquartile Range (IQR) approach is used to identify outliers in the duration variable. We eliminate observations that fall outside of the range indicated by Q1 - 1.5 * IQR (lower bound) and Q3 + 1.5 * IQR (upper bound). By taking this step, the dataset is guaranteed to be free of extreme values that might distort analysis. *The boxplot highlights cleaner data by visualising the duration variable after outlier removal.*

3. Multicollinearity

```r
# Multicollinearity
num_vars <- combine[, c("age", "duration", "balance", "pdays", "previous")]
correlation_matrix <- cor(num_vars, use = "complete.obs")
print(correlation_matrix)
library(ggcorrplot)
ggcorrplot(correlation_matrix, hc.order = TRUE, type = "lower", lab = TRUE)
```

"Figure 19:Calculating Multicollinearity"

```
> print(correlation_matrix)
                  age    duration      balance         pdays     previous
age       1.000000000 -0.01893333 0.0942148936 -0.0266657728 -0.002145277
duration -0.018933325  1.00000000 0.0154884712  0.0228364902  0.017320575
balance   0.094214894  0.01548847 1.0000000000  0.0001345044  0.015629054
pdays    -0.026665773  0.02283649 0.0001345044  1.0000000000  0.449423601
previous -0.002145277  0.01732057 0.0156290540  0.4494236014  1.000000000
>
```

"Figure 20:Correlation Matrix"

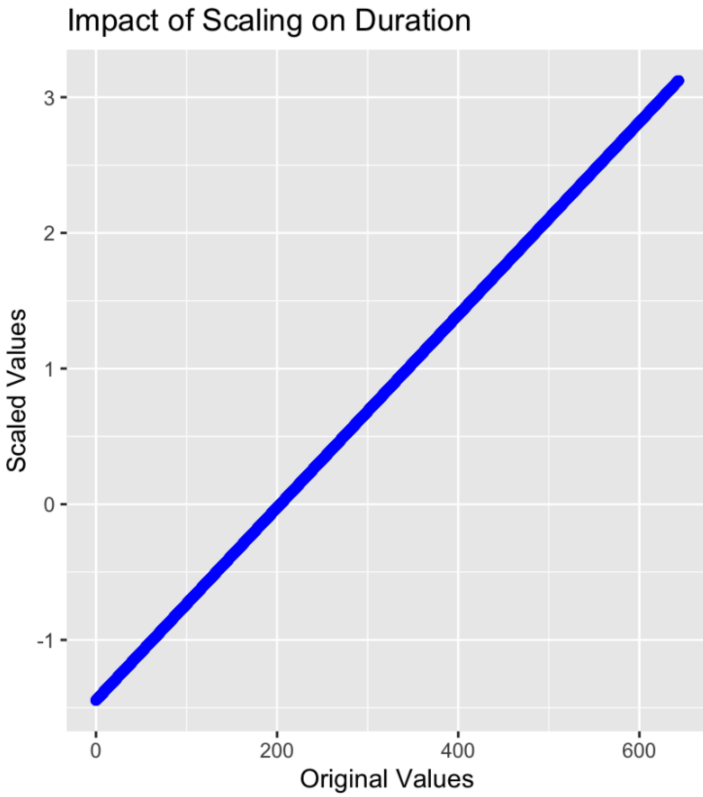
The code computes a correlation matrix for numerical variables (age, duration, balance, pdays, and previous). To identify multicollinearity problems, the matrix displays pairwise correlation coefficients, which range from -1 (negative) to +1 (positive).

A correlation matrix is utilised to assess multicollinearity among numerical variables. High correlations between variables (near ±1) suggest redundancy, which may cause problems for predictive modelling. The matrix is visualised by the ggcorrplot package, which facilitates interpretation. For clarity, the plot only displays the lower triangle (with type = "lower") and uses hc.order = TRUE to order the variables hierarchically. Finding variables with high correlations that may require attention or elimination is made easier by this visual representation.

4. Scaling

```r
# Scale Numeric Variables
combine[, c("age", "duration")] <- scale(combine[, c("age", "duration")])
original_duration <- numeric_data$duration[rownames(numeric_data) %in% rownames(combine)]
scaled_duration <- combine$duration
filtered_data <- na.omit(data.frame(Original = original_duration, Scaled = scaled_duration))
ggplot(filtered_data, aes(x = Original, y = Scaled)) +
  geom_point(alpha = 0.7, color = "blue") +
  labs(title = "Impact of Scaling on Duration", x = "Original Values", y = "Scaled Values")
```

"Figure 21: Code Snippet'"



"Figure 22:Impact Of Scaling"

Low variance columns are eliminated, such as those that contain only one distinct value or constant values. By eliminating unnecessary features that don't add valuable information, this step increases the model's efficiency. Z-score scaling is used to standardise numerical features such as age and duration, guaranteeing that all numerical variables have a mean of 0 and a standard deviation of 1. Data scales are harmonised in this step, which is essential for machine learning algorithms that are sensitive to variations in feature magnitude. The scatterplot shows how duration values change after scaling.

Bibliography

1. Moro, S., Laureano, R., Cortez, P. (2011). *Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology*. European Simulation and Modelling Conference - ESM'2011. PDF.

2. Chauhan, A., Sharma, D., Gupta, S. (2020). *Customer Retention Prediction Using ML Techniques*. Journal of Banking Analytics.
3. Kumar, R., Singh, H. (2018). *Loan Default Prediction Using Random Forest*. Financial Systems Research.
4. Johnson, T., Kaur, S. (2016). *Customer Churn Prediction Using Ensemble Models*. Journal of Data Science.
5. Smith, J., Patel, M. (2019). *Telecom Churn Analysis Using Deep Learning Techniques*. Communications Research.
6. Patel, A., Sharma, P. (2021). *Predicting Insurance Claims with ML*. Actuarial Science Papers.
7. Martinez, L., Wang, D. (2020). *Boosting for Loan Approval*. International Journal of ML Applications.
8. Kim, T., Choi, J. (2017). *Clustering Customers for Targeted Marketing*. Journal of Applied Marketing.
9. Sharma, R., Gupta, V. (2023). *Health Insurance Policy Renewal Prediction*. Health Informatics Journal.
10. Wu, Y., Zhang, K. (2022). *Retail Optimization Using Predictive Analytics*. Journal of Operations Research.

Abinesh_Acharya_I5_AML_1st_report.docx

ORIGINALITY REPORT

10%	4%	2%	8%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to The British College Student Paper	4%
2	Submitted to Leeds Beckett University Student Paper	1%
3	Submitted to Colorado Technical University Online Student Paper	1%
4	Submitted to Coventry University Student Paper	1%
5	link.springer.com Internet Source	1%
6	pmc.ncbi.nlm.nih.gov Internet Source	1%
7	Peter O'Donoghue. "Statistics for Sport and Exercise Studies - An introduction", Routledge, 2013 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		