5303 – STATISTICAL AND SCIENTIFIC COMPUTING I

Predicting In-Vehicle Coupon Acceptance

GROUP – 11

University of Texas at Arlington
Applied Statistics and Data Science

Presented By:

Rushika Badri Prasad (1002235923)
Trupti Shailendra Gandhi (1002202261)
Girish Sunkadakatte Chandrappa (1002212727)

Index

I - Introduction

II - Problem Statement

III - Overview of Data set

IV - Feature Description

V - Exploratory Data Analysis

VI - Data Preprocessing

VII - Feature Engineering

VIII - Machine Learning Model Selection, Training and Evaluation

IX - Variance Inflation Factor (VIF)

X - Challenges Faced

XI – Conclusions

XII - Reference

I. Introduction

In today's competitive marketplace, companies are increasingly using focused marketing strategies to increase customer engagement and optimize sales outcomes. One such approach is the use of in-vehicle coupons, which are offered to customers based on their real-time context, demographic profiles, and behavioral patterns. However, customers do not respond to such promotion strategies uniformly; therefore, understanding the factors that influence their acceptance can significantly enhance the effectiveness of marketing campaigns.

The goal of this study is to predict the probability that a customer will accept an in-vehicle coupon, given several demographic, contextual, and behavioral variables with the help of multiple machine learning models. The features include age, income level, marital status, time of the day, type of vehicle, and past behavior regarding coupon usage, among others. Identification of trends and determinants of coupon acceptance.
As the target variable is a binary category of Accepted or Not Accepted, we can predict coupon acceptance using classification models such as Logistic Regression, Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA) implemented in R.

This predictive analysis would benefit customer:

- Optimize the timing and targeting of in-vehicle coupons to increase acceptance rates.
- Provide actionable insights into customer preferences and behaviors for data-driven decision-making.
- Reduce marketing costs by focusing efforts on high-probability customers.

Basically, this effort aims at relating consumer preferences to corporate marketing strategies to enhance customer satisfaction and generate more revenue.

II. Problem Statement

Understanding the factors that influence customer acceptance of in-vehicle coupons is very important for refining marketing strategies to increase engagement and enhance the overall efficacy of promotional campaigns by using machine learning models.

III. Overview of Dataset

For analyzing the redemption behavior, the chosen dataset contains 12,684 observations with 26 attributes, including categorical and numerical variables. These features are categorized into various domains to analyze factors influencing coupon redemption behavior. The key categories and their respective features include:

- Demographic: User profiles are outlined using attributes like Gender, Age, Has Children, Education, Income, Marital Status, and Occupation.
- Behavioral: Features like Bar, CoffeHouse, CarryAway, RestaurantLessThan20, and Restaurant20to50 indicate past behavior on how the coupons were consumed.

- Coupon Features: Variables such as Coupon and Expiration indicate the type of coupon and for how long they are valid.
- Environmental Factors: Information such as Destination, Passenger, Time, Weather, and Temperature to understand the situational context.
- Proximity Information: Features such as toCouponGEF_5min, toCoupon_GEQ15min, and toCoupon_GEQ25min represent the distance of the customer from the location of the coupon redemption.
- Navigational Features: Variables such as Direction Same and Direction Opposite capture navigational alignment with coupon destinations.
- Target Variable: The target variable Y indicates whether the customer accepted the coupon or not.
- Evaluating whether the target variable is balanced or imbalanced.

IV. Features description

- **destination**: The type of destination Customer intended to visit (No Urgent Place, Home, Work).
- **passenger**: The person accompanying the individual (Alone, Friends, Kids, Partner).
- **weather**: The weather conditions (Sunny, Rainy, Snow).
- **temperature**: The temperature in Fahrenheit.
- **time**: The time of the day (10AM, 2PM, 10PM, 6PM, 7AM).
- **coupon**: The type of coupon offered (Coffee House, Restaurant (<20), Bar, Restaurant (20-50), Carry out & take away).
- **expiration**: Duration before the coupon expires (Example: 2 hours, 1 day).
- **gender**: The gender of the individual (Male, Female).
- **age**: The age of an individual
- **maritalStatus**: The marital status of the individual (Example Single, Married).
- **has_children**: Whether the individual has children.
- **education**: The education level of the individual (Example: High School, Graduate).
- **occupation**: The job category of the individual (Example Student, Professional).
- **income**: The income range of the individual.
- **Bar**: The frequency of an individual visiting bars.
- **CoffeeHouse**: The frequency of an individual visiting coffee houses.
- **CarryAway**: The frequency of an individual ordering carryout meals.
- **RestaurantLessThan20**: The frequency of an individual visiting restaurants with meals priced under $20.
- **Restaurant20To50**: The frequency of an individual visiting restaurants with meals priced $20-$50.
- **toCoupon_GEQ5min**: Whether the driving distance to the destination for using the coupon is greater than or equal to 5 minutes.
- **toCoupon_GEQ15min**: Whether the driving distance to the destination for using the coupon is greater than or equal to 15 minutes.
- **toCoupon_GEQ25min**: Whether the driving distance to the destination for using the coupon is greater than or equal to 25 minutes.

- **direction_same**: Whether the coupon destination is in the same direction as the individual destination.
- **direction_opp**: Whether the coupon destination is in the opposite direction of the individual destination.
- **Y**: The target variable indicating whether the coupon was accepted (1) or not (0).

V. Exploratory Data Analysis

- Importing required libraries and data set

```
##Importing libraries
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(MASS)
library(glmnet)

## Loading required package: Matrix

## Loaded glmnet 4.1-8

library(janitor)

## Warning: package 'janitor' was built under R version 4.4.2

##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test

library(ggplot2)
library(corrplot)

## Warning: package 'corrplot' was built under R version 4.4.2

## corrplot 0.95 loaded
```

```
library(tidyr)

##
## Attaching package: 'tidyr'

## The following objects are masked from 'package:Matrix':
##
##     expand, pack, unpack

library(reshape2)

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##     smiths

library(dplyr)

## Warning: package 'dplyr' was built under R version 4.4.2

##
## Attaching package: 'dplyr'

## The following object is masked from 'package:MASS':
##
##     select

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(car)

## Loading required package: carData
```

```
##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode

library(gridExtra)

## Warning: package 'gridExtra' was built under R version 4.4.2

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine

library(pROC)

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

options(max.print = 10000000)

##########################################################################
#Importing Dataset

df_data_main <- read.csv('/Users/rushi/OneDrive/Desktop/in-vehicle-coupon-recommendation.csv')
```

- Create a copy of data set “df_data” from the imported data.

```r
df_data = df_data_main
```

- Printing the summary of the data set with the 5 number summary displaying numerical features and Class for the categorical features.

```r
summary(df_data)

##  destination         passanger           weather           temperature
##  Length:12684       Length:12684       Length:12684       Min.   :30.0
##  Class :character   Class :character   Class :character   1st Qu.:55.0
##  Mode  :character   Mode  :character   Mode  :character   Median :80.0
##                                                           Mean   :63.3
##                                                           3rd Qu.:80.0
##                                                           Max.   :80.0
##      time              coupon           expiration           gender
##  Length:12684       Length:12684       Length:12684       Length:12684
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      age            maritalStatus       has_children     education
##  Length:12684       Length:12684       Min.   :0.0000   Length:12684
##  Class :character   Class :character   1st Qu.:0.0000   Class :character
##  Mode  :character   Mode  :character   Median :0.0000   Mode  :character
##                                        Mean   :0.4141
##                                        3rd Qu.:1.0000
##                                        Max.   :1.0000
##   occupation           income              car                Bar
##  Length:12684       Length:12684       Length:12684       Length:12684
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
```

```
##  CoffeeHouse         CarryAway          RestaurantLessThan20 Restaurant20To50
##  Length:12684       Length:12684       Length:12684         Length:12684
##  Class :character   Class :character   Class :character     Class :character
##  Mode  :character   Mode  :character   Mode  :character     Mode  :character
##
##
##
##  toCoupon_GEQ5min toCoupon_GEQ15min toCoupon_GEQ25min direction_same
##  Min.   :1        Min.   :0.0000    Min.   :0.0000    Min.   :0.0000
##  1st Qu.:1        1st Qu.:0.0000    1st Qu.:0.0000    1st Qu.:0.0000
##  Median :1        Median :1.0000    Median :0.0000    Median :0.0000
##  Mean   :1        Mean   :0.5615    Mean   :0.1191    Mean   :0.2148
##  3rd Qu.:1        3rd Qu.:1.0000    3rd Qu.:0.0000    3rd Qu.:0.0000
##  Max.   :1        Max.   :1.0000    Max.   :1.0000    Max.   :1.0000
##  direction_opp          Y
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:1.0000   1st Qu.:0.0000
##  Median :1.0000   Median :1.0000
##  Mean   :0.7852   Mean   :0.5684
##  3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000
```

- Displaying first 6 entries of the data set “df_data”.

```
head(df_data)
```

```
##       destination passanger weather temperature time                coupon
## 1 No Urgent Place     Alone   Sunny          55  2PM       Restaurant(<20)
## 2 No Urgent Place Friend(s)   Sunny          80 10AM          Coffee House
## 3 No Urgent Place Friend(s)   Sunny          80 10AM Carry out & Take away
## 4 No Urgent Place Friend(s)   Sunny          80  2PM          Coffee House
## 5 No Urgent Place Friend(s)   Sunny          80  2PM          Coffee House
## 6 No Urgent Place Friend(s)   Sunny          80  6PM       Restaurant(<20)
##   expiration gender age      maritalStatus has_children                education
## 1         1d Female  21 Unmarried partner            1 Some college - no degree
```

```
## 2        2h Female  21 Unmarried partner          1 Some college - no degree
## 3        2h Female  21 Unmarried partner          1 Some college - no degree
## 4        2h Female  21 Unmarried partner          1 Some college - no degree
## 5        1d Female  21 Unmarried partner          1 Some college - no degree
## 6        2h Female  21 Unmarried partner          1 Some college - no degree
##   occupation          income car   Bar CoffeeHouse CarryAway
## 1 Unemployed $37500 - $49999     never       never
## 2 Unemployed $37500 - $49999     never       never
## 3 Unemployed $37500 - $49999     never       never
## 4 Unemployed $37500 - $49999     never       never
## 5 Unemployed $37500 - $49999     never       never
## 6 Unemployed $37500 - $49999     never       never
##   RestaurantLessThan20 Restaurant20To50 toCoupon_GEQ5min toCoupon_GEQ15min
## 1                  4~8              1~3                1                 0
## 2                  4~8              1~3                1                 0
## 3                  4~8              1~3                1                 1
## 4                  4~8              1~3                1                 1
## 5                  4~8              1~3                1                 1
## 6                  4~8              1~3                1                 1
##   toCoupon_GEQ25min direction_same direction_opp Y
## 1                 0              0             1 1
## 2                 0              0             1 0
## 3                 0              0             1 1
## 4                 0              0             1 0
## 5                 0              0             1 0
## 6                 0              0             1 1
```

```
#Description of the dataset
str(df_data)
```

```
## 'data.frame':    12684 obs. of  26 variables:
##  $ destination         : chr  "No Urgent Place" "No Urgent Place" "No Urgent Place" "No Urgent Place" ...
##  $ passanger           : chr  "Alone" "Friend(s)" "Friend(s)" "Friend(s)" ...
```

```
## $ weather             : chr "Sunny" "Sunny" "Sunny" "Sunny" ...
## $ temperature         : int 55 80 80 80 80 80 55 80 80 80 ...
## $ time                : chr "2PM" "10AM" "10AM" "2PM" ...
## $ coupon              : chr "Restaurant(<20)" "Coffee House" "Carry out & Take away" "Coffee House" ...
## $ expiration          : chr "1d" "2h" "2h" "2h" ...
## $ gender              : chr "Female" "Female" "Female" "Female" ...
## $ age                 : chr "21" "21" "21" "21" ...
## $ maritalStatus       : chr "Unmarried partner" "Unmarried partner" "Unmarried partner" "Unmarried partner" ...
## $ has_children        : int 1 1 1 1 1 1 1 1 1 1 ...
## $ education           : chr "Some college - no degree" "Some college - no degree" "Some college - no degree" "Some college - no degree" ...
## $ occupation          : chr "Unemployed" "Unemployed" "Unemployed" "Unemployed" ...
## $ income              : chr "$37500 - $49999" "$37500 - $49999" "$37500 - $49999" "$37500 - $49999" ...
## $ car                 : chr "" "" "" "" ...
## $ Bar                 : chr "never" "never" "never" "never" ...
## $ CoffeeHouse         : chr "never" "never" "never" "never" ...
## $ CarryAway           : chr "" "" "" "" ...
## $ RestaurantLessThan20: chr "4~8" "4~8" "4~8" "4~8" ...
## $ Restaurant20To50    : chr "1~3" "1~3" "1~3" "1~3" ...
## $ toCoupon_GEQ5min    : int 1 1 1 1 1 1 1 1 1 1 ...
## $ toCoupon_GEQ15min   : int 0 0 1 1 1 1 1 1 1 1 ...
## $ toCoupon_GEQ25min   : int 0 0 0 0 0 0 0 0 0 0 ...
## $ direction_same      : int 0 0 0 0 0 0 0 0 0 0 ...
## $ direction_opp       : int 1 1 1 1 1 1 1 1 1 1 ...
## $ Y                   : int 1 0 1 0 0 1 1 1 1 0 ...
```

- Renaming the feature “passenger” to “passenger”

```
# Rename the column passanger to passenger
df_data <- df_data %>% rename(passenger = passanger)
```

```
##############################################################################
#Replace the space in the column names with '_' and column names with lower case

df_data <- df_data %>%
  clean_names()
names(df_data)

##  [1] "destination"          "passenger"            "weather"
##  [4] "temperature"          "time"                 "coupon"
##  [7] "expiration"           "gender"               "age"
## [10] "marital_status"       "has_children"         "education"
## [13] "occupation"           "income"               "car"
## [16] "bar"                  "coffee_house"         "carry_away"
## [19] "restaurant_less_than20" "restaurant20to50"     "to_coupon_geq5min"
## [22] "to_coupon_geq15min"   "to_coupon_geq25min"   "direction_same"
## [25] "direction_opp"        "y"

##############################################################################
#Listing different classes in features
sapply(df_data, function(x) table(x))

## $destination
## x
##            Home No Urgent Place            Work
##            3237            6283            3164
##
## $passenger
## x
##     Alone Friend(s)    Kid(s)   Partner
##      7305      3298      1006      1075
##
## $weather
## x
## Rainy Snowy Sunny
##  1210  1405 10069
```

```
##
## $temperature
## x
##   30   55   80
## 2316 3840 6528
##
## $time
## x
## 10AM 10PM  2PM  6PM  7AM
## 2275 2006 2009 3230 3164
##
## $coupon
## x
##                   Bar Carry out & Take away          Coffee House
##                  2017                  2393                  3996
##       Restaurant(<20)     Restaurant(20-50)
##                  2786                  1492
##
## $expiration
## x
##   1d   2h
## 7091 5593
##
## $gender
## x
## Female   Male
##   6511   6173
##
## $age
## x
##      21      26      31      36      41      46  50plus below21
##    2653    2559    2039    1319    1093     686    1788     547
##
```

```
## $marital_status
## x
##         Divorced   Married partner          Single Unmarried partner
##              516             5100             4752             2186
##          Widowed
##              130
##
## $has_children
## x
##    0    1
## 7431 5253
##
## $education
## x
##                     Associates degree                     Bachelors degree
##                                  1153                                 4335
## Graduate degree (Masters or Doctorate)                 High School Graduate
##                                  1852                                  905
##               Some college - no degree                     Some High School
##                                  4351                                   88
##
## $occupation
## x
##                 Architecture & Engineering
##                                        175
##  Arts Design Entertainment Sports & Media
##                                        629
## Building & Grounds Cleaning & Maintenance
##                                         44
##                      Business & Financial
##                                        544
##               Community & Social Services
##                                        241
```

```
##                     Computer & Mathematical
##                                        1408
##                    Construction & Extraction
##                                         154
##                   Education&Training&Library
##                                         943
##                   Farming Fishing & Forestry
##                                          43
##           Food Preparation & Serving Related
##                                         298
##         Healthcare Practitioners & Technical
##                                         244
##                           Healthcare Support
##                                         242
##            Installation Maintenance & Repair
##                                         133
##                                        Legal
##                                         219
##                  Life Physical Social Science
##                                         170
##                                   Management
##                                         838
##              Office & Administrative Support
##                                         639
##                     Personal Care & Service
##                                         175
##                       Production Occupations
##                                         110
##                           Protective Service
##                                         175
##                                      Retired
##                                         495
##                              Sales & Related
```

```
##                                    1093
##                                 Student
##                                    1584
##        Transportation & Material Moving
##                                     218
##                              Unemployed
##                                    1870
##
## $income
## x
##  $100000 or More  $12500 - $24999  $25000 - $37499  $37500 - $49999
##             1736             1831             2013             1805
##  $50000 - $62499  $62500 - $74999  $75000 - $87499  $87500 - $99999
##             1659              846              857              895
## Less than $12500
##             1042
##
## $car
## x
##
##                                   12576
## Car that is too old to install Onstar :D
##                                      21
##                               crossover
##                                      21
##                            do not drive
##                                      22
##                                  Mazda5
##                                      22
##                  Scooter and motorcycle
##                                      22
##
## $bar
```

```
## x
##        1~3   4~8   gt8 less1 never
##   107  2473  1076   349  3482  5197
##
## $coffee_house
## x
##        1~3   4~8   gt8 less1 never
##   217  3225  1784  1111  3385  2962
##
## $carry_away
## x
##        1~3   4~8   gt8 less1 never
##   151  4672  4258  1594  1856   153
##
## $restaurant_less_than20
## x
##        1~3   4~8   gt8 less1 never
##   130  5376  3580  1285  2093   220
##
## $restaurant20to50
## x
##        1~3   4~8   gt8 less1 never
##   189  3290   728   264  6077  2136
##
## $to_coupon_geq5min
## x
##     1
## 12684
##
## $to_coupon_geq15min
## x
##    0    1
## 5562 7122
```

```
##
## $to_coupon_geq25min
## x
##     0     1
## 11173  1511
##
## $direction_same
## x
##    0    1
## 9960 2724
##
## $direction_opp
## x
##    0    1
## 2724 9960
##
## $y
## x
##    0    1
## 5474 7210
```

Evaluating and Handling Null values in the data set

- We could observe features "car" have null values around 99.14% which we are assuming that users were asked to enter a value only if they have a different vehicle other than car.
- Since the count of null values in car features is significantly higher, we are dropping the feature assuming that it has unique value.
- Also, the features coffee_house, carry_away, restaurant_less_than20, restaurant20t050 have null values which are less than 1.5% of the total count.
- Also, we are handling these null values using mode imputation method during the data preprocessing steps because of the significantly lesser count of null values and no relationships have been found between other features to handle these null values.

```r
#checking null , na , empty cells in the data set
sapply(df_data, function(x) sum(x == ""))

##          destination            passenger              weather
##                    0                    0                    0
##          temperature                 time               coupon
##                    0                    0                    0
##           expiration               gender                  age
##                    0                    0                    0
##       marital_status         has_children            education
##                    0                    0                    0
##           occupation               income                  car
##                    0                    0                12576
##                  bar         coffee_house            carry_away
##                  107                  217                  151
## restaurant_less_than20      restaurant20to50     to_coupon_geq5min
##                  130                  189                    0
##    to_coupon_geq15min   to_coupon_geq25min        direction_same
##                    0                    0                    0
##        direction_opp                    y
##                    0                    0

sapply(df_data, function(x) sum(is.na(x)))

##          destination            passenger              weather
##                    0                    0                    0
##          temperature                 time               coupon
##                    0                    0                    0
##           expiration               gender                  age
##                    0                    0                    0
##       marital_status         has_children            education
##                    0                    0                    0
##           occupation               income                  car
##                    0                    0                    0
##                  bar         coffee_house            carry_away
```

```
##                    0                    0                    0
## restaurant_less_than20       restaurant20to50      to_coupon_geq5min
##                    0                    0                    0
##     to_coupon_geq15min     to_coupon_geq25min         direction_same
##                    0                    0                    0
##          direction_opp                    y
##                    0                    0
```

```r
sapply(df_data, function(x) sum(is.null(x)))
```

```
##            destination            passenger              weather
##                    0                    0                    0
##            temperature                 time               coupon
##                    0                    0                    0
##             expiration               gender                  age
##                    0                    0                    0
##         marital_status         has_children            education
##                    0                    0                    0
##             occupation               income                  car
##                    0                    0                    0
##                    bar         coffee_house            carry_away
##                    0                    0                    0
## restaurant_less_than20       restaurant20to50      to_coupon_geq5min
##                    0                    0                    0
##     to_coupon_geq15min     to_coupon_geq25min         direction_same
##                    0                    0                    0
##          direction_opp                    y
##                    0                    0
```

```r
###########################################################################
```

- Segregate the categorical and numerical columns to analyze 5 number summaries for each numerical column.

```r
#summary of categorical and numerical columns
```

```r
categorical_col <- sapply(df_data, is.factor) | sapply(df_data, is.character)
categorical_col <- names(df_data)[categorical_col]
numerical_col <- sapply(df_data, is.numeric)
numerical_col <- names(df_data)[numerical_col]
print(categorical_col)
```

```
##  [1] "destination"          "passenger"            "weather"
##  [4] "time"                 "coupon"               "expiration"
##  [7] "gender"               "age"                  "marital_status"
## [10] "education"            "occupation"           "income"
## [13] "car"                  "bar"                  "coffee_house"
## [16] "carry_away"           "restaurant_less_than20" "restaurant20to50"
```

```r
print(numerical_col)
```

```
## [1] "temperature"       "has_children"      "to_coupon_geq5min"
## [4] "to_coupon_geq15min" "to_coupon_geq25min" "direction_same"
## [7] "direction_opp"     "y"
```

```r
summary(df_data[,categorical_col])
```

```
##  destination         passenger           weather              time
##  Length:12684       Length:12684       Length:12684       Length:12684
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##     coupon          expiration          gender              age
##  Length:12684       Length:12684       Length:12684       Length:12684
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##  marital_status      education          occupation           income
##  Length:12684       Length:12684       Length:12684       Length:12684
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##      car                bar             coffee_house        carry_away
##  Length:12684       Length:12684       Length:12684       Length:12684
##  Class :character   Class :character   Class :character   Class :character
```

```
## Mode :character Mode :character Mode :character Mode :character
## restaurant_less_than20 restaurant20to50
## Length:12684 Length:12684
## Class :character Class :character
## Mode :character Mode :character
```

```
summary(df_data[,numerical_col])
```

```
## temperature has_children to_coupon_geq5min to_coupon_geq15min
## Min. :30.0 Min. :0.0000 Min. :1 Min. :0.0000
## 1st Qu.:55.0 1st Qu.:0.0000 1st Qu.:1 1st Qu.:0.0000
## Median :80.0 Median :0.0000 Median :1 Median :1.0000
## Mean :63.3 Mean :0.4141 Mean :1 Mean :0.5615
## 3rd Qu.:80.0 3rd Qu.:1.0000 3rd Qu.:1 3rd Qu.:1.0000
## Max. :80.0 Max. :1.0000 Max. :1 Max. :1.0000
## to_coupon_geq25min direction_same direction_opp y
## Min. :0.0000 Min. :0.0000 Min. :0.0000 Min. :0.0000
## 1st Qu.:0.0000 1st Qu.:0.0000 1st Qu.:1.0000 1st Qu.:0.0000
## Median :0.0000 Median :0.0000 Median :1.0000 Median :1.0000
## Mean :0.1191 Mean :0.2148 Mean :0.7852 Mean :0.5684
## 3rd Qu.:0.0000 3rd Qu.:0.0000 3rd Qu.:1.0000 3rd Qu.:1.0000
## Max. :1.0000 Max. :1.0000 Max. :1.0000 Max. :1.0000
```

- Listing out the unique classes and counts in each feature.
- Post analyzing, we got to know that most of the features are either nominal or binary in nature.

```
##Listing out the classes and its respective counts for categorical and numerical column

cat_value_counts<- lapply(df_data[,categorical_col], table)
cat_value_counts

## $destination
##
## Home No Urgent Place Work
## 3237 6283 3164
```

```
##
## $passenger
##
##     Alone Friend(s)    Kid(s)   Partner
##      7305      3298      1006      1075
##
## $weather
##
## Rainy Snowy Sunny
##  1210  1405 10069
##
## $time
##
## 10AM 10PM  2PM  6PM  7AM
## 2275 2006 2009 3230 3164
##
## $coupon
##
##                   Bar Carry out & Take away          Coffee House
##                  2017                  2393                  3996
##       Restaurant(<20)     Restaurant(20-50)
##                  2786                  1492
##
## $expiration
##
##   1d   2h
## 7091 5593
##
## $gender
##
## Female   Male
##   6511   6173
##
```

```
## $age
##
##     21      26      31      36      41      46  50plus below21
##    2653    2559    2039    1319    1093     686    1788     547
##
## $marital_status
##
##          Divorced   Married partner           Single Unmarried partner
##               516              5100              4752              2186
##           Widowed
##               130
##
## $education
##
##                      Associates degree                      Bachelors degree
##                                   1153                                  4335
## Graduate degree (Masters or Doctorate)                  High School Graduate
##                                   1852                                   905
##               Some college - no degree                      Some High School
##                                   4351                                    88
##
## $occupation
##
##                 Architecture & Engineering
##                                        175
##  Arts Design Entertainment Sports & Media
##                                        629
## Building & Grounds Cleaning & Maintenance
##                                         44
##                      Business & Financial
##                                        544
##                Community & Social Services
##                                        241
```

```
##                     Computer & Mathematical
##                              1408
##                   Construction & Extraction
##                               154
##                  Education&Training&Library
##                               943
##                  Farming Fishing & Forestry
##                                43
##          Food Preparation & Serving Related
##                               298
##        Healthcare Practitioners & Technical
##                               244
##                          Healthcare Support
##                               242
##            Installation Maintenance & Repair
##                               133
##                                 Legal
##                               219
##                 Life Physical Social Science
##                               170
##                            Management
##                               838
##              Office & Administrative Support
##                               639
##                     Personal Care & Service
##                               175
##                      Production Occupations
##                               110
##                          Protective Service
##                               175
##                               Retired
##                               495
##                            Sales & Related
```

```
##                                     1093
##                                  Student
##                                     1584
##          Transportation & Material Moving
##                                      218
##                               Unemployed
##                                     1870
##
## $income
##
##  $100000 or More  $12500 - $24999  $25000 - $37499  $37500 - $49999
##             1736             1831             2013             1805
##  $50000 - $62499  $62500 - $74999  $75000 - $87499  $87500 - $99999
##             1659              846              857              895
## Less than $12500
##             1042
##
## $car
##
##
##                                    12576
## Car that is too old to install Onstar :D
##                                       21
##                                crossover
##                                       21
##                             do not drive
##                                       22
##                                   Mazda5
##                                       22
##                   Scooter and motorcycle
##                                       22
##
## $bar
```

```
##
##       1~3   4~8   gt8 less1 never
##   107  2473  1076   349  3482  5197
##
## $coffee_house
##
##       1~3   4~8   gt8 less1 never
##   217  3225  1784  1111  3385  2962
##
## $carry_away
##
##       1~3   4~8   gt8 less1 never
##   151  4672  4258  1594  1856   153
##
## $restaurant_less_than20
##
##       1~3   4~8   gt8 less1 never
##   130  5376  3580  1285  2093   220
##
## $restaurant20to50
##
##       1~3   4~8   gt8 less1 never
##   189  3290   728   264  6077  2136
```

```
num_value_counts<- lapply(df_data[,numerical_col], table)
num_value_counts
```

```
## $temperature
##
##   30   55   80
## 2316 3840 6528
##
## $has_children
##
```

```
##    0    1
## 7431 5253
##
## $to_coupon_geq5min
##
##     1
## 12684
##
## $to_coupon_geq15min
##
##    0    1
## 5562 7122
##
## $to_coupon_geq25min
##
##     0     1
## 11173  1511
##
## $direction_same
##
##    0    1
## 9960 2724
##
## $direction_opp
##
##    0    1
## 2724 9960
##
## $y
##
##    0    1
## 5474 7210
```

➢ We are creating a customized function (bivariate_analysis) to visualize and analyze how each feature is affecting the coupon acceptance rate based on the percentage values.

```
######################Exploratory Data Analysis (EDA)######################

#Creating function for bivariate analysis
percent_value_counts <- function(df, feature, target) {
  df_summary <- df %>%
    group_by_at(vars(feature)) %>% ##Grouping based on the feature
    summarise(
      Total_Count = n(),
      Accepted = sum(get(target) == 1, na.rm = TRUE),
      Rejected = sum(get(target) == 0, na.rm = TRUE)
    ) %>%
    mutate(
      Total_Percent = round((Total_Count / sum(Total_Count)) * 100, 3),
      Percent_Accepted = round((Accepted / Total_Count) * 100, 3),
      Percent_Rejected = round((Rejected / Total_Count) * 100, 3)
    )
  return(df_summary)
}

bivariate_analysis <- function(df, feature, target) {
  df_EDA <- percent_value_counts(df, feature, target)
  df_EDA <- df_EDA %>%
    mutate(
      Total_Label = paste0("(", Total_Percent, "%)"),
      Accepted_Label = paste0("(", Percent_Accepted, "%)")
    )

  #Creating bar plots
  plot <- ggplot(data = df_EDA) +
    geom_bar(aes_string(x = feature, y = "Total_Count"), stat = "identity", fill = "grey", alpha =
0.7) +
```

```
  geom_bar(aes_string(x = feature, y = "Accepted"), stat = "identity", fill = "blue", alpha = 0.7)
+
  geom_text(aes_string(x = feature, y = "Total_Count", label = "Total_Label"),
          vjust = -0.5, size = 3, color = "black") +
  geom_text(aes_string(x = feature, y = "Accepted", label = "Accepted_Label"),
          vjust = -0.5, size = 3, color = "black") +
  labs(
    title = paste("Accepted Coupons with respect to", feature),
    x = feature,
    y = "Coupon Counts"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

 print(plot)
 return(df_EDA)
}
```

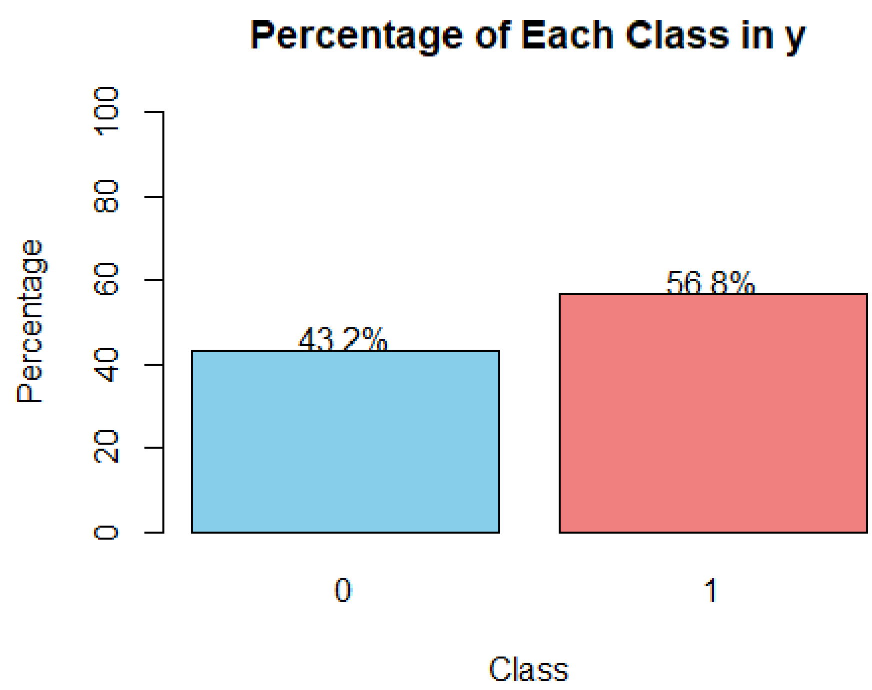
Evaluating whether the target variable is balanced or imbalanced

Code:

```
################################################################################
#

#Calculating the percentages for classes in target variable (Y)
y_table <- table(df_data$y)
y_percentage <- prop.table(y_table) * 100

#Creating bar plots
barplot_heights <- barplot(y_percentage,
                    main = "Percentage of Each Class in y",
                    col = c("skyblue", "lightcoral"),
```

```
                    names.arg = c("0", "1"),
                    ylim = c(0, 100),
                    ylab = "Percentage",
                    xlab = "Class")


#Adding the percentage values for the respective bar plots
text(barplot_heights,
    y = y_percentage + 3,
    labels = paste0(round(y_percentage, 1), "%"),
    col = "black",
    cex = 1)
```

Output:



- This graph gives an overview of the target variable distribution (Y), which shows the customers accepted or rejected the in-vehicle coupon.
- This is an approximately balanced target feature where 56.8% are accepted and 43.2% rejected.

- Balance in a target value and data set is always critical for developing any reliable predictive models, because this makes the model learn properly on both outcomes without being biased to one class. The balanced distribution reduces the chances of overfitting and underperforming, hence allowing for accurate predictions.
- As the target variable is approximately balanced, we are good enough to proceed with this.

Performing analyses on key features in terms of coupon acceptance rate

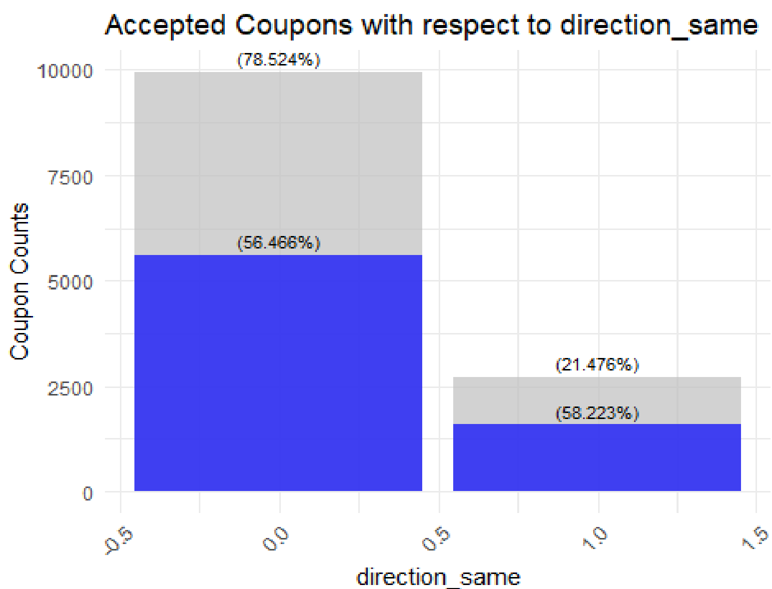
- **Feature distance_same:**

  Code:

```
#Bivariate analysis for direction same
#The no. of people who accepted the coupon with respect to direction
feature_column_dir <- "direction_same"  #categorical feature to analyze
target_column <- "y"                    #target column
df_analysis_direction_same <- bivariate_analysis(df_data, feature_column_dir, target_column)

print(df_analysis_direction_same) # 78% of them are direction opposite in that 56% percent are accepted.
```

  Output:

```
## # A tibble: 2 × 9
##   direction_same Total_Count Accepted Rejected Total_Percent Percent_Accepted
##            <int>       <int>    <int>    <int>         <dbl>            <dbl>
## 1              0        9960     5624     4336          78.5             56.5
## 2              1        2724     1586     1138          21.5             58.2
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>

# 21% of them are direction same in that 58% percent are accepted
```



- The above plot shows that a majority of 78.52% of coupons are issued to users who travel in the opposite direction, compared to those traveling in the same direction.
- Whereas more coupons were distributed in the opposite direction, the rate at which these coupons are taken up is relatively lower compared to those in the same direction. In contrast, the rate of acceptance for coupons offered to users in the same direction stands remarkably higher, with 58.22% acceptance.
- This indicates that, although more coupons may be given out in the opposite direction, users who are traveling in the same direction are more likely to accept and use the offers.

- **Feature coupon:**

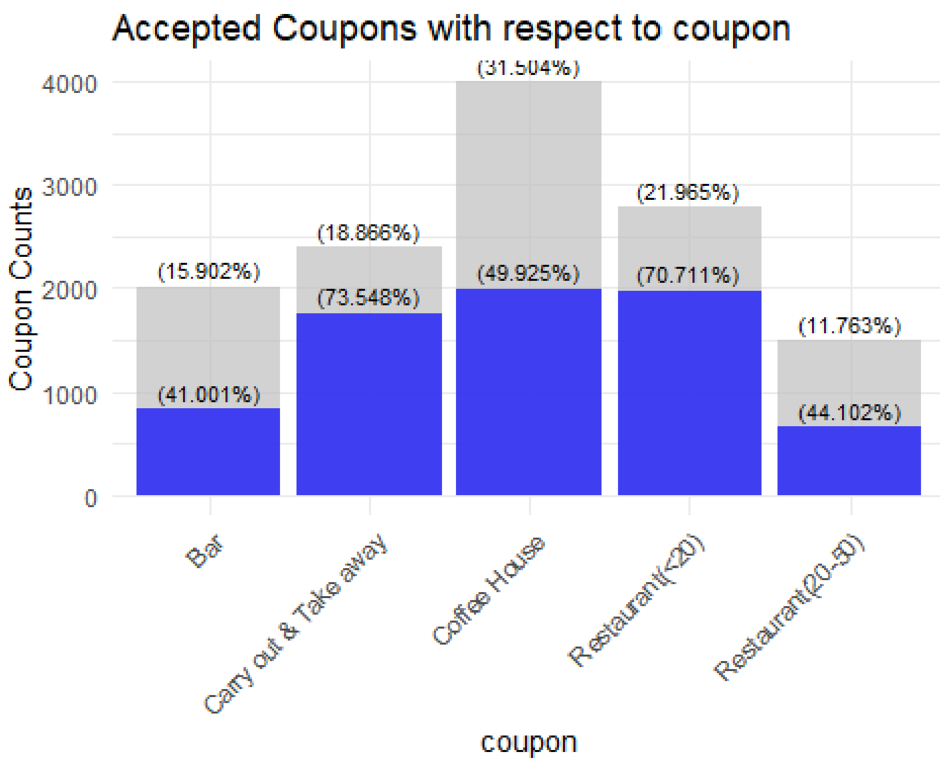
  Code:

```
#Bivariate analysis for coupons
#Different types of coupons accepted and the no. of people in each coupon
feature_column_coupon <- "coupon"
df_analysis_coupon <- bivariate_analysis(df_data, feature_column_coupon, target_column)

print(df_analysis_coupon)
```

  Output:

```
## # A tibble: 5 × 9
##   coupon          Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>                 <int>    <int>    <int>         <dbl>            <dbl>
```

```
## 1 Bar                    2017      827     1190         15.9            41.0
## 2 Carry out & Take…        2393      1760      633          18.9             73.5
## 3 Coffee House           3996      1995      2001          31.5            49.9
## 4 Restaurant(20-50)        1492      658       834          11.8             44.1
## 5 Restaurant(<20)         2786      1970      816          22.0             70.7
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```



- The above plot highlights that the coupons issued for Coffee House form the lion's share at 31.50%, while the acceptance of such coupons is quite low at 49.92%.
- Contrasting this with other types of businesses, such as Carry Out & Take Away and Restaurants (<20), even though their offered frequency is very low, their acceptance rate stands higher at 73.54% and 70.11%, respectively. Thus, these categories will help engage customers more, unlike those of Coffee Houses.

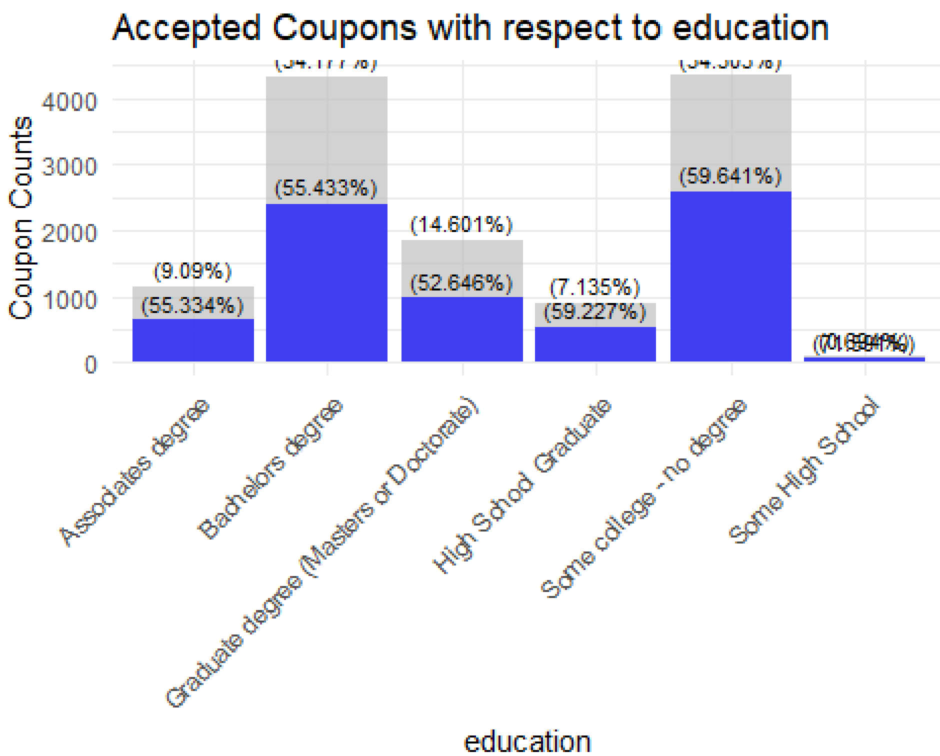
- **Feature education:**

  Code:

```r
#Bivariate analysis for education
#The no. of coupons accepted with respect to education
feature_column_edu <- "education"
df_analysis_education <- bivariate_analysis(df_data, feature_column_edu, target_column)

print(df_analysis_education)
```

Output:

```
## # A tibble: 6 × 9
##   education         Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>                   <int>    <int>    <int>         <dbl>            <dbl>
## 1 Associates degree       1153      638      515          9.09             55.3
## 2 Bachelors degree        4335     2403     1932         34.2              55.4
## 3 Graduate degree …       1852      975      877         14.6              52.6
## 4 High School Grad…        905      536      369          7.14             59.2
## 5 Some High School          88       63       25          0.694            71.6
## 6 Some college - n…       4351     2595     1756         34.3              59.6
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```



- The above graph shows that most of the coupons are offered to users with a bachelor's degree and those with some college education but with no degree.
- However, in terms of usage, users with some college education but no degree and high school graduates show the largest acceptance. This means that while bachelor's degree holders receive the most coupons, other educational groups tend to redeem them more.

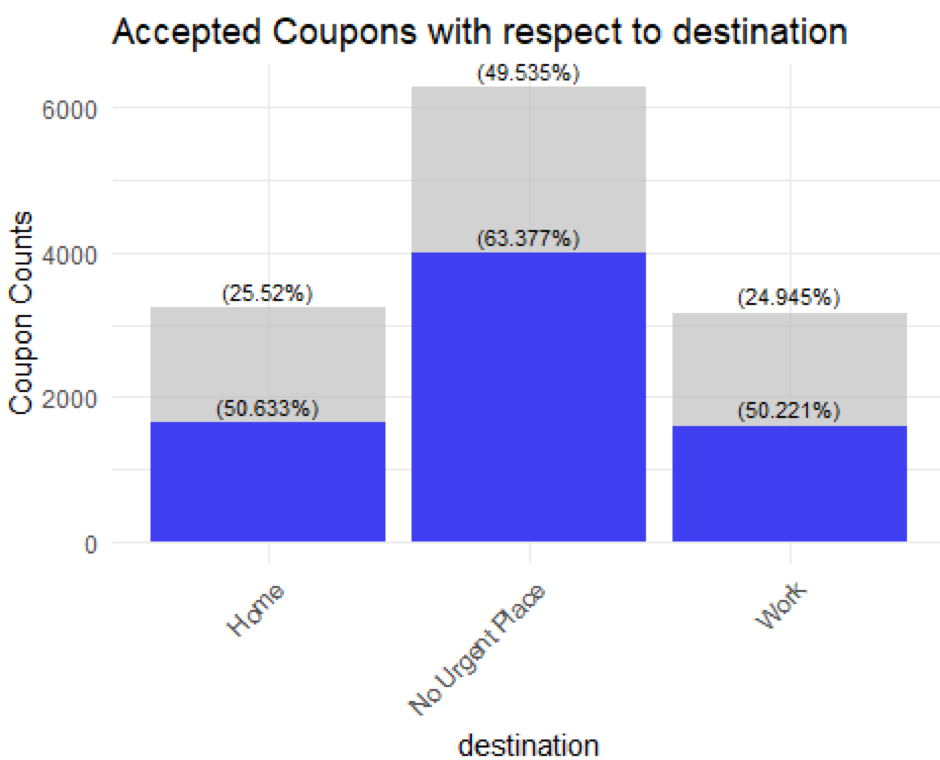
- **Feature destination:**

  Code:

```r
#Bivariate analysis for destination
#The no. of coupons accepted with respect to destination
feature_column_destination <- "destination"
df_analysis_destination <- bivariate_analysis(df_data, feature_column_destination,
target_column)

print(df_analysis_destination)
```

  Output:

```
## # A tibble: 3 × 9
##   destination     Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>                 <int>    <int>    <int>         <dbl>            <dbl>
## 1 Home                   3237     1639     1598          25.5             50.6
## 2 No Urgent Place        6283     3982     2301          49.5             63.4
## 3 Work                   3164     1589     1575          24.9             50.2
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```



- The above graph shows that most coupons were distributed to users whose destination was "No Urgent Place," while this group had a higher rate of acceptance, too, at 63.37%, compared with people whose destination was defined as "Home" or "Work."
- This implies that customers with no pressing destination may also be more accepting of promotional offers because of lesser rigidity in their schedules.
- In contrast, users traveling to Home or Work may have lower acceptance rates, likely because of time constraints or less relevance of the offers. Businesses can use this insight to target users with non-urgent destinations, where the potential for coupon engagement is higher.

- **Feature Passenger:**

Code:

```
#Bivariate analysis for passenger
#The no. of coupons accepted with respect to passenger
feature_column_passenger <- "passenger"
df_analysis_passenger <- bivariate_analysis(df_data, feature_column_passenger,
target_column)

print(df_analysis_passenger)
```

Output:

```
## # A tibble: 4 × 9
##   passenger Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>           <int>    <int>    <int>         <dbl>            <dbl>
## 1 Alone            7305     3841     3464         57.6              52.6
## 2 Friend(s)        3298     2221     1077         26.0              67.3
## 3 Kid(s)           1006      508      498          7.93             50.5
## 4 Partner          1075      640      435          8.48             59.5
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```

- The above graph shows that the majority of passengers are solo travelers, 57.59%, which is the highest among all travel groups.
- However, despite the high solo traveler proportion, the highest coupon acceptance rate is among people traveling with friends, at an acceptance ratio of 67.34%.
- This could indicate that social interaction during journeys might positively affect the probability of accepting coupons, because passengers who travel with friends are more open to shared activities or conversations about offers.
- Also, solo travelers, despite being more frequent, showed lower acceptance rates, thus hinting at different priorities or tendencies in decision-making.

- **Feature weather:**

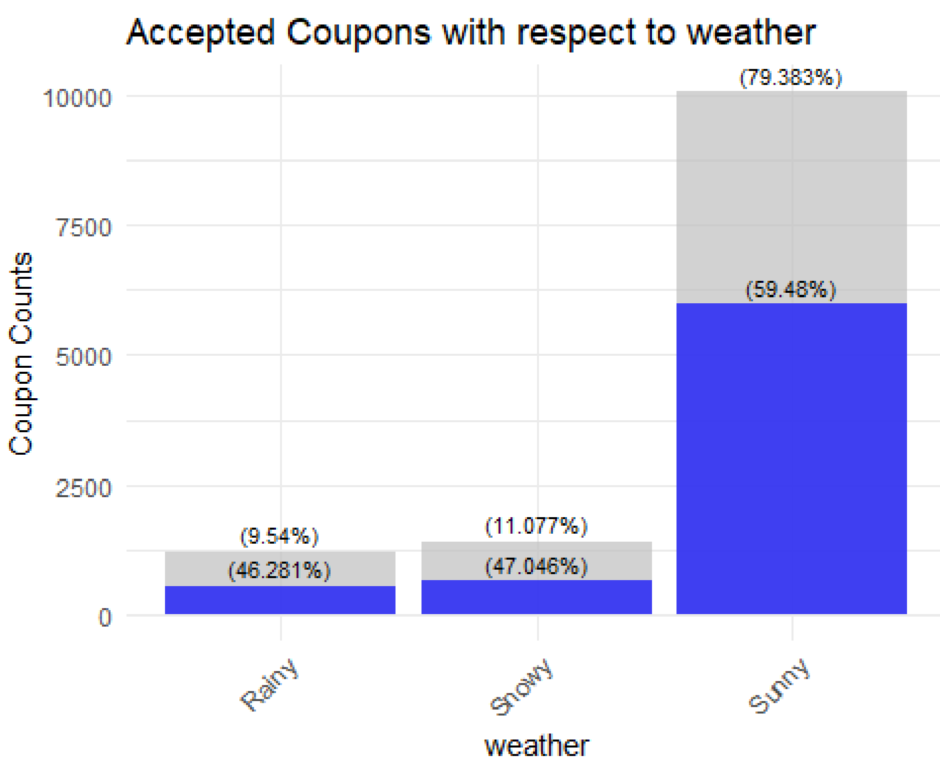
  Code:

```
#Bivariate analysis for weather
#The no. of coupons accepted with respect to weather
feature_column_weather <- "weather"
df_analysis_weather <- bivariate_analysis(df_data, feature_column_weather, target_column)
```

```
print(df_analysis_weather)
```

Output:

```
## # A tibble: 3 × 9
##   weather Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>         <int>    <int>    <int>         <dbl>            <dbl>
## 1 Rainy          1210      560      650          9.54             46.3
## 2 Snowy          1405      661      744         11.1              47.0
## 3 Sunny         10069     5989     4080         79.4              59.5
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```



- ➢ The above graph shows that coupon distribution for users in the low-income and medium-income groups is the highest.
- ➢ Similarly, the acceptance rate is also higher in these groups compared to high-income users. This indicates that low- and medium-income users are more responsive to coupons, likely due to greater cost sensitivity, making them key targets for promotional campaigns.

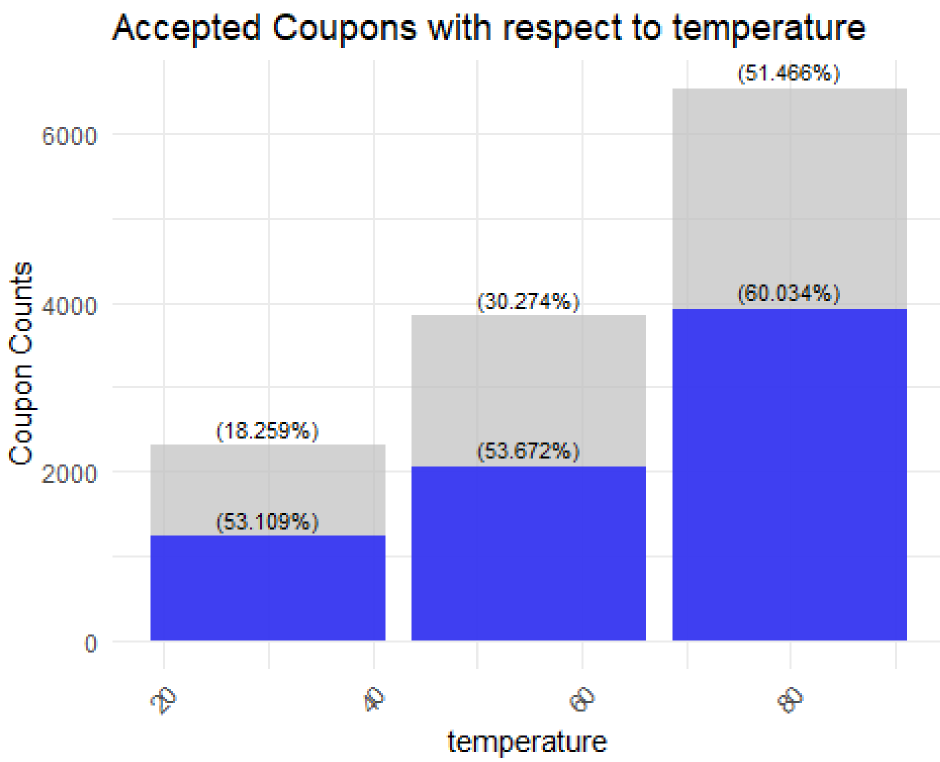
- **Feature temperature:**

  Code:

```
#Bivariate analysis for temperature
#The no. of coupons accepted with respect to temperature
feature_column_temperature <- "temperature"
df_analysis_temperature <- bivariate_analysis(df_data, feature_column_temperature,
target_column)

print(df_analysis_temperature)
```

  Output:

```
## # A tibble: 3 × 9
##   temperature Total_Count Accepted Rejected Total_Percent Percent_Accepted
##         <int>       <int>    <int>    <int>         <dbl>            <dbl>
## 1          30        2316     1230     1086          18.3             53.1
## 2          55        3840     2061     1779          30.3             53.7
## 3          80        6528     3919     2609          51.5             60.0
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```



- The above graph shows that during sunny weather, coupons are offered and accepted at the highest levels compared to rainy or snowy conditions.
- This may simply indicate that when the weather is nice, customers would be more willing to try promotional offers since they spend more time outdoors and have a happier mood.
- In contrast, customers may be less likely to accept coupons on rainy or snowy days because they may be spending less time outdoors or concentrating on indoor activities. This insight can help businesses optimize their coupon distribution strategies by focusing on offering promotions during favorable weather conditions for higher engagement.

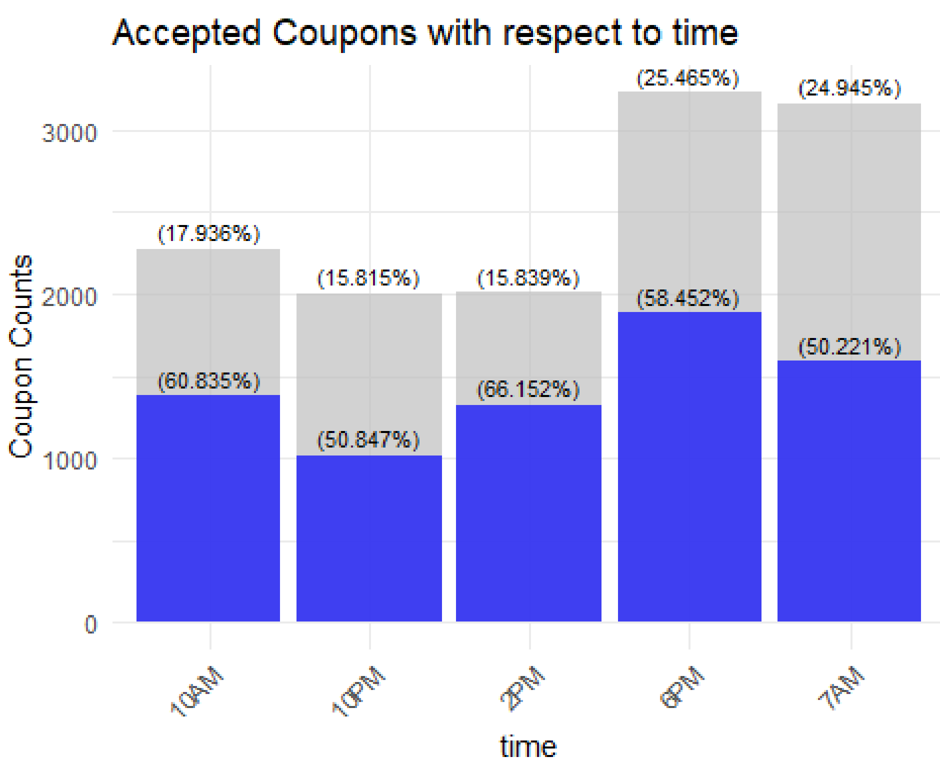
- **Feature time:**

  Code:

```r
#Bivariate analysis for time
#The no. of coupons accepted with respect to time
feature_column_time <- "time"
df_analysis_time <- bivariate_analysis(df_data, feature_column_time, target_column)

print(df_analysis_time)
```

Output:

```
## # A tibble: 5 × 9
##   time  Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>       <int>    <int>    <int>         <dbl>            <dbl>
## 1 10AM         2275     1384      891          17.9             60.8
## 2 10PM         2006     1020      986          15.8             50.8
## 3 2PM          2009     1329      680          15.8             66.2
## 4 6PM          3230     1888     1342          25.5             58.5
## 5 7AM          3164     1589     1575          24.9             50.2
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```



- ➢ This graph shows that most coupons are provided to moving users at 6 PM and 7 AM, but the highest pull rates for the user occur at 10 AM and 2 PM.
- ➢ It means that though more coupons are distributed in peak travel hours, moving users during mid-morning and early afternoon show more engagement in pulling offers.

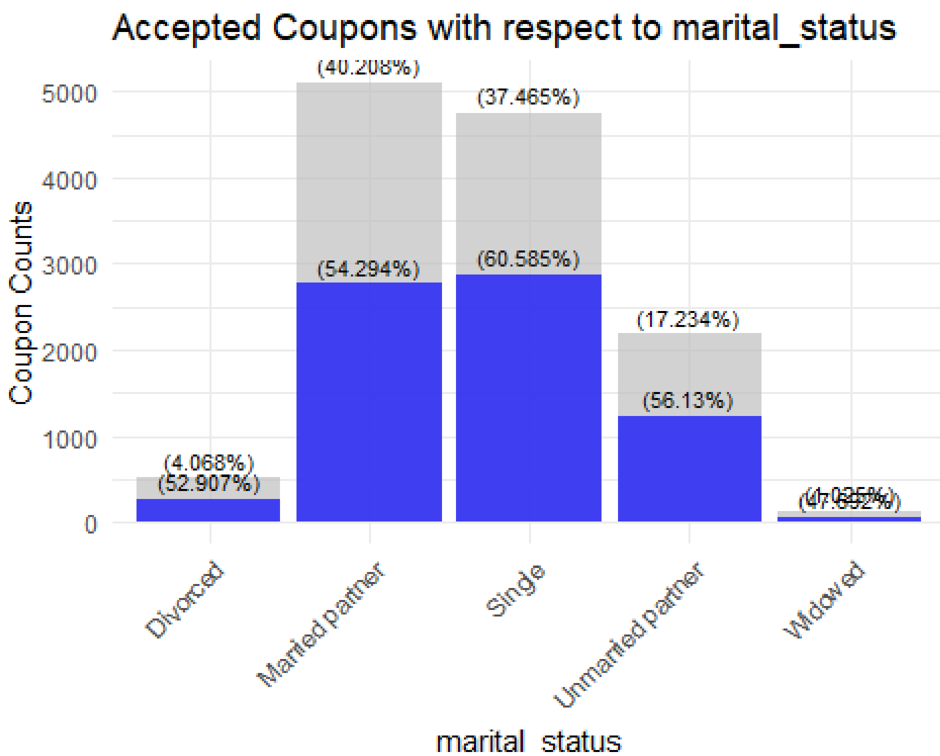
- **Feature time:**

Code:

```
#Bivariate analysis for maritalStatus
#The no. of coupons accepted with respect to maritalStatus
feature_column_maritalStatus <- "marital_status"
df_analysis_maritalstatus <- bivariate_analysis(df_data, feature_column_maritalStatus,
target_column)

print(df_analysis_maritalstatus)
```

Output:

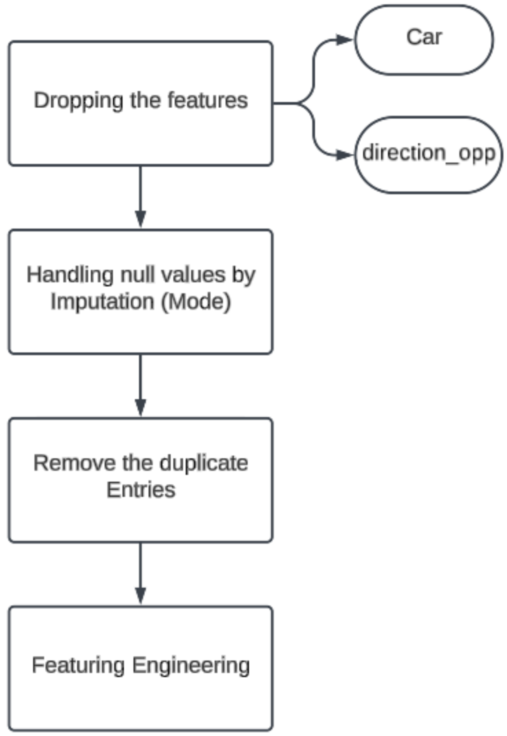
```
## # A tibble: 5 × 9
##   marital_status    Total_Count Accepted Rejected Total_Percent Percent_Accepted
##   <chr>                   <int>    <int>    <int>         <dbl>            <dbl>
## 1 Divorced                  516      273      243          4.07             52.9
## 2 Married partner          5100     2769     2331         40.2              54.3
## 3 Single                   4752     2879     1873         37.5              60.6
## 4 Unmarried partner        2186     1227      959         17.2              56.1
## 5 Widowed                   130       62       68          1.02             47.7
## # ℹ 3 more variables: Percent_Rejected <dbl>, Total_Label <chr>,
## #   Accepted_Label <chr>
```

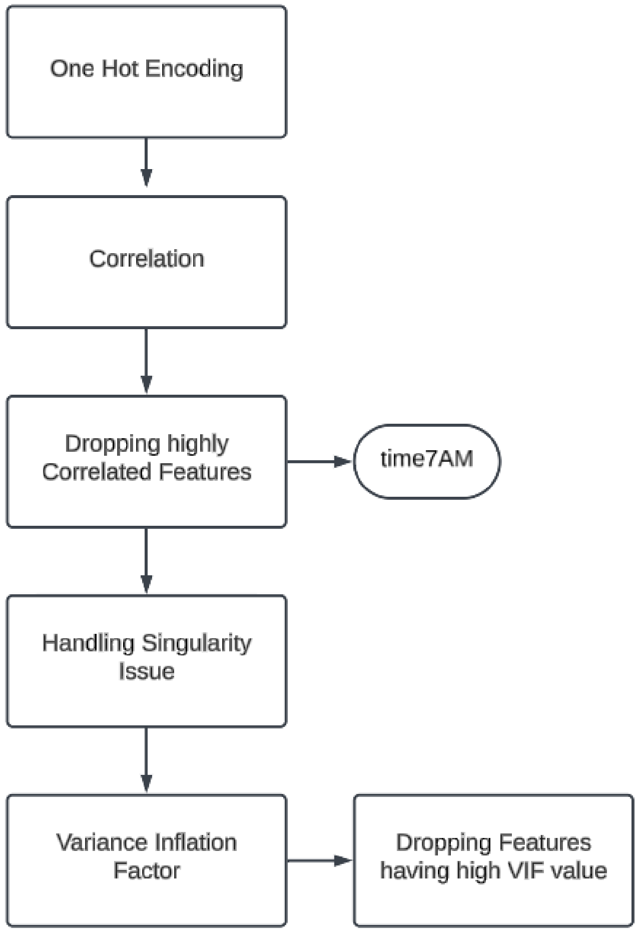


- The above graph emphasizes that the majority of coupons were distributed to people categorized as "Married Partner" and "Single" in the marital status column.
- On the other hand, the usage rate is much higher in "Single" users, 60.58%, and "Unmarried Partner" users, 56.13%, compared with other categories.
- This may be indicative of single people or those living in unmarried partnerships being more open and receptive to engaging with coupon offers.

VI. Data Preprocessing

Workflow:


Dropping the features
Car
direction_opp
Handling null values by Imputation (Mode)
Remove the duplicate Entries
Featuring Engineering



- During our EDA we observed that feature “car” has 99.14% of the missing values as informed during EDA analysis users might have been asked in the survey to enter a value if they are owning a different vehicle other than car.
- Since feature “car” has significant null values and it is not impacting our analysis and model, we are dropping this feature.
- Also, for feature “direction_opp” we have redundancy column “direction_same” because of redundancy in nature we are dropping this feature.

Code for dropping features:

```
############################################## Data Pre-processing ###############

#Dropping columns "car" and "direction opp"
## Dropping 'car' column as it has significant missing values
## Dropping 'direction opposite' column due to redundancy

df_data_dummy <- df_data[ , !(names(df_data) %in% c("car","direction_opp"))]
dim(df_data_dummy)
```

```
## [1] 12684    24
```

Output:

- We could from the below output features “car” and “direction_opp” has been dropped.

```
head(df_data_dummy)

##       destination passenger weather temperature time                coupon
## 1 No Urgent Place     Alone   Sunny          55  2PM       Restaurant(<20)
## 2 No Urgent Place Friend(s)   Sunny          80 10AM          Coffee House
## 3 No Urgent Place Friend(s)   Sunny          80 10AM Carry out & Take away
## 4 No Urgent Place Friend(s)   Sunny          80  2PM          Coffee House
## 5 No Urgent Place Friend(s)   Sunny          80  2PM          Coffee House
## 6 No Urgent Place Friend(s)   Sunny          80  6PM       Restaurant(<20)
##   expiration gender age    marital_status has_children                education
## 1         1d Female  21 Unmarried partner            1 Some college - no degree
## 2         2h Female  21 Unmarried partner            1 Some college - no degree
## 3         2h Female  21 Unmarried partner            1 Some college - no degree
## 4         2h Female  21 Unmarried partner            1 Some college - no degree
## 5         1d Female  21 Unmarried partner            1 Some college - no degree
## 6         2h Female  21 Unmarried partner            1 Some college - no degree
##   occupation          income   bar coffee_house carry_away
## 1 Unemployed $37500 - $49999 never        never
## 2 Unemployed $37500 - $49999 never        never
## 3 Unemployed $37500 - $49999 never        never
## 4 Unemployed $37500 - $49999 never        never
## 5 Unemployed $37500 - $49999 never        never
## 6 Unemployed $37500 - $49999 never        never
##   restaurant_less_than20 restaurant20to50 to_coupon_geq5min to_coupon_geq15min
## 1                    4~8              1~3                 1                  0
## 2                    4~8              1~3                 1                  0
## 3                    4~8              1~3                 1                  1
## 4                    4~8              1~3                 1                  1
## 5                    4~8              1~3                 1                  1
```

```
## 6                4~8          1~3             1               1
##   to_coupon_geq25min direction_same y
## 1              0            0 1
## 2              0            0 0
## 3              0            0 1
## 4              0            0 0
## 5              0            0 0
## 6              0            0 1
```

Handling missing values using imputation methods:

- Features coffee_house, carry_away, restaurant_less_than20, restaurant20t050 have null values which are less than 1.5% of the total count.
- Also, we are handling these null values using mode imputation methods during the data preprocessing steps because of the significantly lesser count of null values and no relationships have been found between other features to handle these null values.
- We are creating a customized function called "get_mode" to fetch the most repeated value from each feature which is mentioned in the variable "columns_to_imput" and performing mode imputation.

  Code:

```
##########################Handling the missing values.#####################
#Mode imputation to handle missing values

get_mode <- function(x) {
  uniq_x <- unique(x)
  uniq_x[which.max(tabulate(match(x, uniq_x)))]
}

df_data_dummy[df_data_dummy == ""] <- NA

columns_to_impute <- c("bar",
"coffee_house","carry_away","restaurant_less_than20","restaurant20to50")

for (col in columns_to_impute) {
  mode_value <- get_mode(df_data_dummy[[col]])
  df_data_dummy[[col]][is.na(df_data_dummy[[col]])] <- mode_value
```

```
}
```

- Post handling null values we are dropping duplicate entries from the data set.

  Code:

```
#dropping duplicate entries after mode imputation

df_data_dummy <- df_data_dummy[!duplicated(df_data_dummy), ]
dim(df_data_dummy)

## [1] 12610    24
```

- Evaluating whether null values still exist after mode imputation.

  Code:

```
sapply(df_data_dummy, function(x) sum(x == ""))

##          destination            passenger              weather
##                    0                    0                    0
##          temperature                 time               coupon
##                    0                    0                    0
##           expiration               gender                  age
##                    0                    0                    0
##       marital_status         has_children            education
##                    0                    0                    0
##           occupation               income                  bar
##                    0                    0                    0
##         coffee_house           carry_away restaurant_less_than20
##                    0                    0                    0
##      restaurant20to50     to_coupon_geq5min    to_coupon_geq15min
##                    0                    0                    0
##    to_coupon_geq25min        direction_same                    y
##                    0                    0                    0

sapply(df_data_dummy, function(x) sum(is.na(x)))
```

```
##          destination            passenger            weather
##                    0                    0                  0
##          temperature                 time             coupon
##                    0                    0                  0
##           expiration               gender                age
##                    0                    0                  0
##       marital_status         has_children          education
##                    0                    0                  0
##           occupation               income                bar
##                    0                    0                  0
##         coffee_house           carry_away restaurant_less_than20
##                    0                    0                  0
##      restaurant20to50     to_coupon_geq5min   to_coupon_geq15min
##                    0                    0                  0
##    to_coupon_geq25min       direction_same                   y
##                    0                    0                  0
```

```
sapply(df_data_dummy, function(x) sum(is.null(x)))
```

```
##          destination            passenger            weather
##                    0                    0                  0
##          temperature                 time             coupon
##                    0                    0                  0
##           expiration               gender                age
##                    0                    0                  0
##       marital_status         has_children          education
##                    0                    0                  0
##           occupation               income                bar
##                    0                    0                  0
##         coffee_house           carry_away restaurant_less_than20
##                    0                    0                  0
##      restaurant20to50     to_coupon_geq5min   to_coupon_geq15min
##                    0                    0                  0
```

```
##    to_coupon_geq25min      direction_same             y
##                  0                 0                 0
```

VII. Feature Engineering

- Feature engineering is one of the crucial steps in machine learning. It transforms raw data into useful features so that models perform better.
- It enables prediction of outcomes more accurately through complicated patterns. It eases the working of datasets by removing unimportant or repeated information.
- Prepares the data for modeling by fixing problems such as missing values and adjusting scales.
- Helps in building interpretable features and preprocessing the data to be molded into a specific model, ensuring the optimization of predictive model effectiveness and reliability.
- We perform feature engineering by analyzing feature correlations and interpreting their relevance in real-world scenarios.

- **Feature engineering for features destination and passenger**

Code:

```
#######################Feature engineering##############################

#For columns destination and passenger

df_data_dummy$destination_passenger <- paste(df_data_dummy$destination,
df_data_dummy$passenger, sep = "_")
head(df_data_dummy$destination_passenger)
```

Output:

```
## [1] "No Urgent Place_Alone"     "No Urgent Place_Friend(s)"
## [3] "No Urgent Place_Friend(s)" "No Urgent Place_Friend(s)"
## [5] "No Urgent Place_Friend(s)" "No Urgent Place_Friend(s)"
```

```
length(df_data_dummy$destination_passenger)

## [1] 12610
```

- **Feature engineering for features Temperature and Weather**

Code:

```
#For columns Temperature and Weather

df_data_dummy$weather_temperature <- paste(df_data_dummy$weather,
df_data_dummy$temperature, sep = "_")
head(df_data_dummy$weather_temperature)
```

Output:

```
## [1] "Sunny_55" "Sunny_80" "Sunny_80" "Sunny_80" "Sunny_80" "Sunny_80"

length(df_data_dummy$weather_temperature)

## [1] 12610
```

- **Feature engineering for features Marital_Status and Children**

Code:

```
#For columns Marital Status and Children

df_data_dummy$maritalstatus_children <- paste(df_data_dummy$marital_status,
df_data_dummy$has_children, sep = "_")
head(df_data_dummy$maritalstatus_children)
```

Output:

```
## [1] "Unmarried partner_1" "Unmarried partner_1" "Unmarried partner_1"
## [4] "Unmarried partner_1" "Unmarried partner_1" "Unmarried partner_1"

length(df_data_dummy$maritalstatus_children)

## [1] 12610
```

- **Feature engineering for features “to_coupon_geq5min to_coupon_geq15min to_coupon_geq25min” into “to_coupon”**

  - Here we are combining three features “to_coupon_geq5min to_coupon_geq15min to_coupon_geq25min” and creating a new column “to_coupon” based on certain conditions using label encoding method.

Code:

```
#For columns to_coupon_geq5min to_coupon_geq15min to_coupon_geq25min

df_data_dummy <- df_data_dummy %>%
 mutate(
  to_coupon = case_when(
    to_coupon_geq25min == 1 ~ 2,                    # Condition 3: Greater than 25 minutes -> 2
    to_coupon_geq15min == 1 & to_coupon_geq25min == 0 ~ 1,    # Condition 2: Between 15 and 25 minutes -> 1
    to_coupon_geq5min == 1 & to_coupon_geq15min == 0 ~ 0,     # Condition 1: Less than 15 minutes -> 0
    TRUE ~ NA_real_                        # In case of missing or unexpected values
  )
 )
table(df_data_dummy$to_coupon)
```

Output:

```
##
##    0    1    2
## 5551 5596 1463
```

- An essential step in data preprocessing is dropping irrelevant features to enhance model performance and support feature engineering.

  Code:

```
#Dropping the columns which are used for feature engineering

df_data_dummy <- df_data_dummy[ , !(names(df_data_dummy) %in% c("marital_status",
```

```
"has_children",
                                             "destination", "passenger","weather",
"temperature","to_coupon_geq5min","to_coupon_geq15min","to_coupon_geq25min"))]
```

- **Performing Feature engineering on column “age”**
  - The feature age is being grouped into categories based on the following conditions for better interpretation and understanding.

Code:

```
#Lisitng unique values in age columns

sapply(df_data_dummy['age'], unique)

##      age
## [1,] "21"
## [2,] "46"
## [3,] "26"
## [4,] "31"
## [5,] "41"
## [6,] "50plus"
## [7,] "36"
## [8,] "below21"

####Categorize the age into age groups

age_group <- character(length(df_data_dummy$age))
print(length(age_group))

## [1] 12610

length(age_group) == length(df_data_dummy$age)

## [1] TRUE

for (i in 1:length(df_data_dummy$age)) {
  if (df_data_dummy$age[i] < 21 | df_data_dummy$age[i] == 'below21') {
```

```
    age_group[i] <- "Teenagers"
  } else if (df_data_dummy$age[i] >= 21 && df_data_dummy$age[i] <= 35) {
    age_group[i] <- "Young Adults"
  } else if (df_data_dummy$age[i] >= 36 && df_data_dummy$age[i] <= 50) {
    age_group[i] <- "Middle-Aged Adults"
  } else if (df_data_dummy$age[i] == '50plus') {
    age_group[i] <- "Seniors"
  }
}

df_data_dummy$age <- age_group
head(df_data_dummy$age)
```

Output:

```
## [1] "Young Adults" "Young Adults" "Young Adults" "Young Adults" "Young Adults"
## [6] "Young Adults"

#listing out the unique value counts in the column age
table(df_data_dummy$age)

##
## Middle-Aged Adults            Seniors          Teenagers       Young Adults
##               3076               1781                544               7209
```

- **Performing Feature engineering on column "income"**
  - The feature "income" is being grouped into categories based on the following conditions for better interpretation and understanding.

Code:

```
##### Categorize the income into groups

#Listing out the unique values in the income feature.
table(df_data_dummy$income)
```

```
##
## $100000 or More  $12500 - $24999  $25000 - $37499  $37500 - $49999
##          1717            1825            2006            1795
## $50000 - $62499  $62500 - $74999  $75000 - $87499  $87500 - $99999
##          1655             843             856             879
## Less than $12500
##          1034
```

```
income_group <- character(length(df_data_dummy$income))
print(length(income_group))
```

```
## [1] 12610
```

```
length(income_group) == length(df_data_dummy$income)
```

```
## [1] TRUE
```

```
for (i in 1:length(df_data_dummy$income)) {
  if (df_data_dummy$income[i] == 'Less than $12500' |
      df_data_dummy$income[i] == '$12500 - $24999' |
      df_data_dummy$income[i] == '$25000 - $37499') {
    income_group[i] <- "Low_income"
  } else if (df_data_dummy$income[i] == '$37500 - $49999' |
             df_data_dummy$income[i] == '$50000 - $62499' |
             df_data_dummy$income[i] == '$62500 - $74999') {
    income_group[i] <- "Medium_income"
  } else if (df_data_dummy$income[i] == '$75000 - $87499' |
             df_data_dummy$income[i] == '$87500 - $99999' |
             df_data_dummy$income[i] == '$100000 or More') {
    income_group[i] <- "High_income"
  }
}


df_data_dummy$income <- income_group
```

```
#Listing out the classes in the income after feature engineering.
table(df_data_dummy$income)
```

Output:

```
##
##   High_income    Low_income Medium_income
##          3452          4865          4293
```

- **Performing Feature engineering on column “occupation”**
    - The feature “occupation“ is being grouped into categories based on the following conditions for better interpretation and understanding.

Code:

```
###Listing out the unique values in occupation feature.
table(df_data_dummy$occupation)

##
##               Architecture & Engineering
##                                      175
##  Arts Design Entertainment Sports & Media
##                                      627
## Building & Grounds Cleaning & Maintenance
##                                       44
##                     Business & Financial
##                                      543
##              Community & Social Services
##                                      239
##                   Computer & Mathematical
##                                     1390
##                Construction & Extraction
##                                      154
```

```
##              Education&Training&Library
##                                     939
##              Farming Fishing & Forestry
##                                      43
##        Food Preparation & Serving Related
##                                     298
##     Healthcare Practitioners & Technical
##                                     244
##                       Healthcare Support
##                                     242
##        Installation Maintenance & Repair
##                                     133
##                                   Legal
##                                     219
##            Life Physical Social Science
##                                     169
##                              Management
##                                     821
##          Office & Administrative Support
##                                     638
##                  Personal Care & Service
##                                     175
##                   Production Occupations
##                                     108
##                      Protective Service
##                                     174
##                                 Retired
##                                     493
##                        Sales & Related
##                                    1088
##                                 Student
##                                    1575
##          Transportation & Material Moving
```

```
##                          218
##                      Unemployed
##                         1861
```

```
####Categorize the occupation_list into groups

occupation_group <- character(length(df_data_dummy$occupation))
print(length(occupation_group))
```

```
## [1] 12610
```

```
length(occupation_group) == length(df_data_dummy$occupation)
```

```
## [1] TRUE
```

```
for (i in 1:length(df_data_dummy$occupation)) {
  if (df_data_dummy$occupation[i] == 'Installation Maintenance & Repair' |
      df_data_dummy$occupation[i] == 'Transportation & Material Moving' |
      df_data_dummy$occupation[i] == 'Food Preparation & Serving Related' |
      df_data_dummy$occupation[i] == 'Building & Grounds Cleaning & Maintenance') {
    occupation_group[i] <- "Labour"
  } else if (df_data_dummy$occupation[i] == 'Architecture & Engineering' |
             df_data_dummy$occupation[i] == 'Education & Training & Library' |
             df_data_dummy$occupation[i] == 'Healthcare Practitioners & Technical' |
             df_data_dummy$occupation[i] == 'Management' |
             df_data_dummy$occupation[i] == 'Arts Design Entertainment Sports & Media' |
             df_data_dummy$occupation[i] == 'Computer & Mathematical' |
             df_data_dummy$occupation[i] == 'Legal' |
             df_data_dummy$occupation[i] == 'Business & Financial' |
             df_data_dummy$occupation[i] == 'Farming Fishing & Forestry') {
    occupation_group[i] <- "Professionals"
  } else if (df_data_dummy$occupation[i] == 'Retired') {
    occupation_group[i] <- "Retired"
  } else if (df_data_dummy$occupation[i] == 'Sales & Related' |
             df_data_dummy$occupation[i] == 'Personal Care & Service' |
             df_data_dummy$occupation[i] == 'Protective Service') {
```

```
    occupation_group[i] <- "Service and sales"
  } else if (df_data_dummy$occupation[i] == 'Student') {
    occupation_group[i] <- "Student"
  } else if (df_data_dummy$occupation[i] == 'Healthcare Support' |
             df_data_dummy$occupation[i] == 'Life Physical Social Science' |
             df_data_dummy$occupation[i] == 'Community & Social Services' |
             df_data_dummy$occupation[i] == 'Construction & Extraction' |
             df_data_dummy$occupation[i] == 'Office & Administrative Support' |
             df_data_dummy$occupation[i] == 'Production Occupations') {
    occupation_group[i] <- "Technicians"
  } else if (df_data_dummy$occupation[i] == 'Unemployed') {
    occupation_group[i] <- "Unemployed"
  } else occupation_group[i] <- "Others"
}


df_data_dummy$occupation <- occupation_group
head(df_data_dummy$occupation)
```

Output:

```
## [1] "Unemployed" "Unemployed" "Unemployed" "Unemployed" "Unemployed"
## [6] "Unemployed"
```

Code:

```
#Listing out the classes in the occupation_list after feature engineering.
table(df_data_dummy$occupation)
```

Output:

```
##
##           Labour           Others    Professionals          Retired
##              693              939             4062              493
## Service and sales          Student      Technicians       Unemployed
##             1437             1575             1550             1861
```

➢ Listing out the column names after feature engineering.

Code:

```
names(df_data_dummy)
```

Output:

```
##  [1] "time"                 "coupon"               "expiration"
##  [4] "gender"               "age"                  "education"
##  [7] "occupation"           "income"               "bar"
## [10] "coffee_house"         "carry_away"           "restaurant_less_than20"
## [13] "restaurant20to50"     "direction_same"       "y"
## [16] "destination_passenger"  "weather_temperature"    "maritalstatus_children"
## [19] "to_coupon"
```

➢ Printing the dimension of our data set after feature engineering.

Code:

```
dim(df_data_dummy)

## [1] 12610    19
```

➢ Analyzing features after feature engineering to evaluate their impact and relevance to the model.

➢ In order to achieve this, we are performing multi variate analysis on key features.

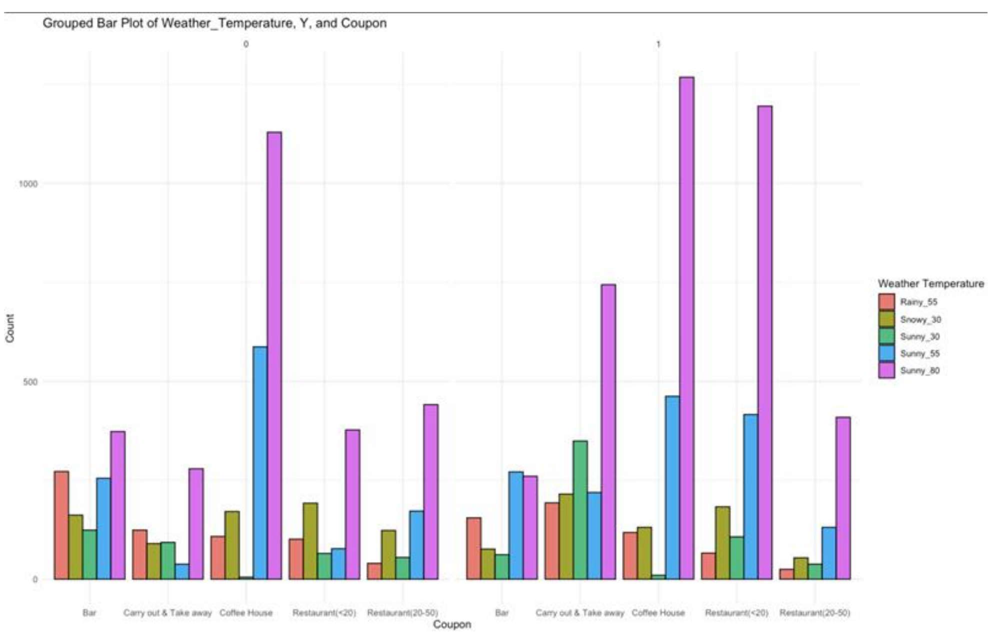
Multi variate analysis for the feature "coupon, weather_temperature" with target (y).

Code:

```
############### Analyzing the features after feature engineering ############

#Multivariate analysis after feature engineering.

# Calculate counts group by coupon,weather_temperature
df_data_dummy_summary <- df_data_dummy %>%
```

```r
  group_by(coupon, weather_temperature, y) %>%
  summarise(count = n(), .groups = "drop")

#Bar plot of Weather_Temperature, Y, and Coupon
ggplot(df_data_dummy_summary, aes(x = factor(coupon), y = count, fill =
factor(weather_temperature))) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9), color = "black") +
  labs(title = "Grouped Bar Plot of Weather_Temperature, Y, and Coupon",
       x = "Coupon",
       y = "Count",
       fill = "Weather Temperature") +
  theme_minimal() +
  facet_wrap(vars(y))
```

Output:



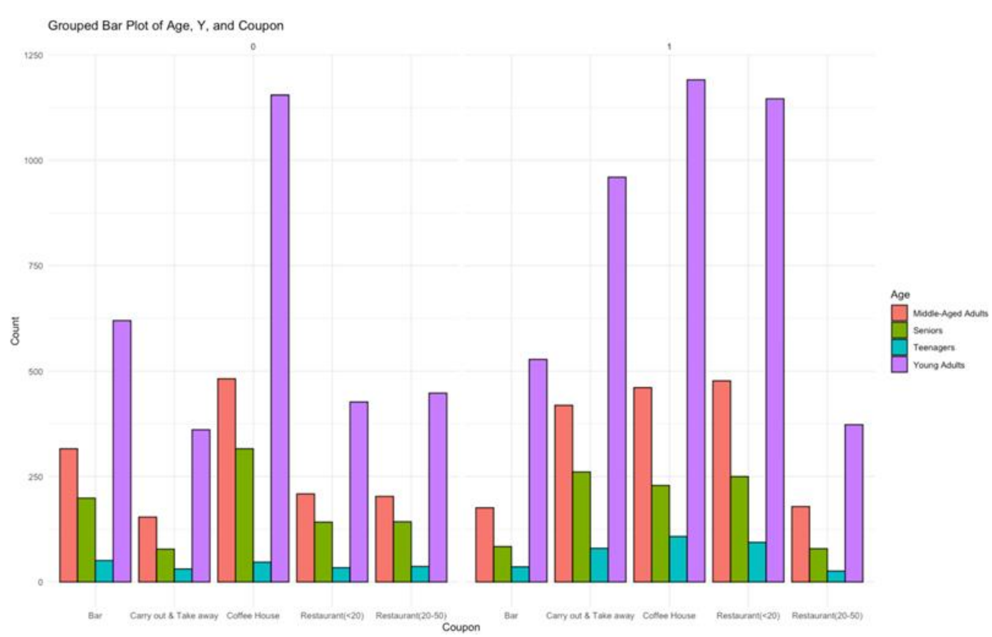
- ➢ The above graph clearly shows that coupons are most accepted during sunny weather compared to other weather conditions, such as rainy or snowy.
- ➢ More precisely, "Restaurant (<20)" and "Carry out & Take away" coupons have a higher acceptance rate in sunny weather, meaning that customers are more likely to engage with these offers when the weather is favorable.
- ➢ This insight will help businesses implement the best strategy for distributing coupons by targeting the days when there is a lot of sunlight.

Multi variate analysis for the feature “coupon, age” with target (y).

Code:

```
# Calculate counts group by age and coupon
df_data_dummy_summary <- df_data_dummy %>%
  group_by(coupon, age, y) %>%
  summarise(count = n(), .groups = "drop")

#Grouped Bar Plot of Age, Y, and Coupon
ggplot(df_data_dummy_summary, aes(x = factor(coupon), y = count, fill = factor(age))) +
  geom_bar(stat = "identity", position = position_dodge(width = 0.9), color = "black") +
  labs(title = "Grouped Bar Plot of Age, Y, and Coupon",
       x = "Coupon",
       y = "Count",
       fill = "Age") +
  theme_minimal() +
  facet_wrap(vars(y))
```

Output:



- ➢ The above graph further shows that young adults lead in the acceptance of coupons than any other age group classes, with a great variation from the rejected coupons, while seniors and teenagers trailed behind, indicating low usability of the coupons among this age group.

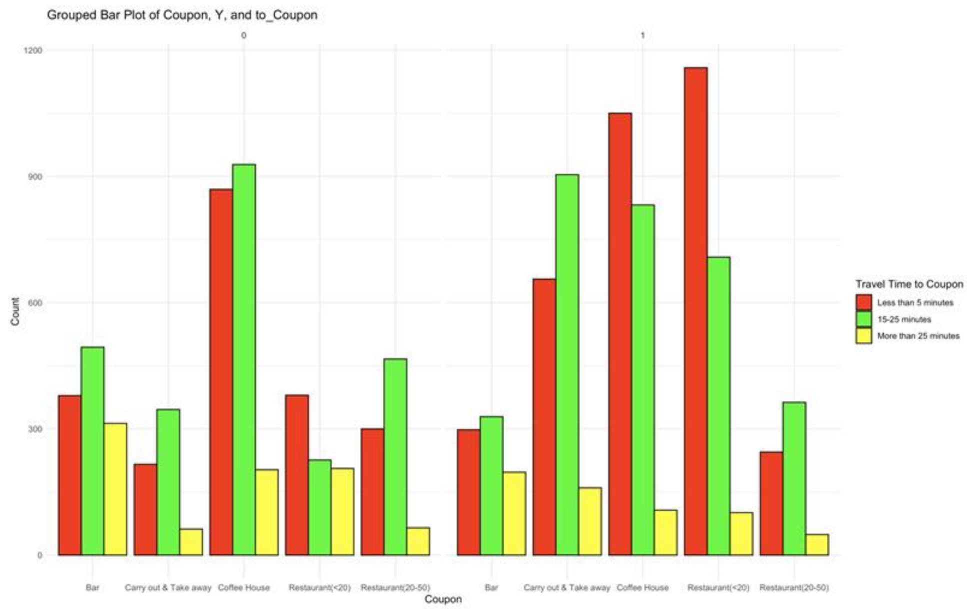
- This can provide valuable insight for businesses to effectively target young adults with promotional efforts while re-evaluating and possibly reworking their approach to better engage seniors and teenagers.

Multi variate analysis for the feature "coupon, to_coupon" with target (y).

Code:

```
# Calculate counts group by coupon, to_coupon
df_data_dummy_summary <- df_data_dummy %>%
  group_by(coupon, to_coupon, y) %>%
  summarise(count = n(), .groups = "drop")

ggplot(df_data_dummy_summary, aes(x = factor(coupon), y = count, fill = factor(to_coupon)))
+
  geom_bar(stat = "identity", position = position_dodge(width = 0.9), color = "black") +
  scale_fill_manual(
    values = c("red", "green", "yellow"),
    labels = c("Less than 5 minutes", "15-25 minutes", "More than 25 minutes"),
    name = "Travel Time to Coupon"
  ) +
  labs(
    title = "Grouped Bar Plot of Coupon, Y, and to_Coupon",
    x = "Coupon",
    y = "Count"
  ) +
  theme_minimal() +
  facet_wrap(vars(y))
```

Output:



- The above graph shows that coupons are mostly accepted when the time spent travelling is less than 25 minutes, with an increased frequency of acceptance rather than refusal.
- On the other hand, coupons are not taken much when the travelling time exceeds 25 minutes, reflecting that longer durations spent traveling are less capable of soliciting promotional engagement.
- This insight can help businesses optimize their coupon distribution strategies by focusing on customers with shorter travel times to increase the rate of engagement and acceptance.

Correlation between features.

Correlation analysis is important in understanding the relationship between variables and identifying patterns that can affect model performance. Applied to nominal and binary data encoded through one-hot encoding, correlation helps to:

- Find extra or similar features that can be taken out to make the model simpler and reduce multicollinearity.
- Highlight meaningful relationships between encoded features and the target variable, improving feature selection and model interpretability.
- Enhance the model effectiveness by focusing on the most pertinent features to generalize better and not overfit. This will ensure the dataset is organized, which helps in developing powerful and efficient predictive models by performing correlation analysis on one-hot encoded data.

Code:

- Here initially we are splitting categorical and quantitative columns to perform one hot encoding.

```
##############################################################################
#Correlation via one hot encoding method

#Separating the categorical and numerical columns

categorical_cols <- names(df_data_dummy)[sapply(df_data_dummy, function(x) is.factor(x) |
is.character(x))]
numeric_cols <- names(df_data_dummy)[sapply(df_data_dummy, function(x) is.numeric(x))]
```

- Total count of categorical and numerical columns in the data set.

```
length(categorical_cols)

## [1] 16

length(numeric_cols)

## [1] 3
```

- Performing encoding operation on categorical columns into binary values for each class.
  Code:

```
#encoding categorical column
df_categorical_encoded <- df_data_dummy %>%
  select(all_of(categorical_cols)) %>%
  mutate_if(is.character, as.factor)

#converting the encoded data
df_categorical_encoded <- model.matrix(~ . - 1, data = df_categorical_encoded) %>%
  as.data.frame()

head(df_categorical_encoded)
```

  Output:

```
##   time10AM time10PM time2PM time6PM time7AM couponCarry out & Take away
## 1        0        0       1       0       0                           0
## 2        1        0       0       0       0                           0
```

```
## 3      1      0      0      0      0                    1
## 4      0      0      1      0      0                    0
## 5      0      0      1      0      0                    0
## 6      0      0      0      1      0                    0
##   couponCoffee House couponRestaurant(<20) couponRestaurant(20-50) expiration2h
## 1                 0                     1                       0            0
## 2                 1                     0                       0            1
## 3                 0                     0                       0            1
## 4                 1                     0                       0            1
## 5                 1                     0                       0            0
## 6                 0                     1                       0            1
##   genderMale ageSeniors ageTeenagers ageYoung Adults educationBachelors degree
## 1          0          0            0               1                         0
## 2          0          0            0               1                         0
## 3          0          0            0               1                         0
## 4          0          0            0               1                         0
## 5          0          0            0               1                         0
## 6          0          0            0               1                         0
##   educationGraduate degree (Masters or Doctorate) educationHigh School Graduate
## 1                                               0                             0
## 2                                               0                             0
## 3                                               0                             0
## 4                                               0                             0
## 5                                               0                             0
## 6                                               0                             0
##   educationSome college - no degree educationSome High School occupationOthers
## 1                                 1                         0                0
## 2                                 1                         0                0
## 3                                 1                         0                0
## 4                                 1                         0                0
## 5                                 1                         0                0
## 6                                 1                         0                0
##   occupationProfessionals occupationRetired occupationService and sales
```

```
## 1                   0             0                  0
## 2                   0             0                  0
## 3                   0             0                  0
## 4                   0             0                  0
## 5                   0             0                  0
## 6                   0             0                  0
##   occupationStudent occupationTechnicians occupationUnemployed incomeLow_income
## 1               0                 0               1            0
## 2               0                 0               1            0
## 3               0                 0               1            0
## 4               0                 0               1            0
## 5               0                 0               1            0
## 6               0                 0               1            0
##   incomeMedium_income bar4~8 bargt8 barless1 barnever coffee_house4~8
## 1                 1      0      0       0       1            0
## 2                 1      0      0       0       1            0
## 3                 1      0      0       0       1            0
## 4                 1      0      0       0       1            0
## 5                 1      0      0       0       1            0
## 6                 1      0      0       0       1            0
##   coffee_housegt8 coffee_houseless1 coffee_housenever carry_away4~8
## 1             0              0              1           0
## 2             0              0              1           0
## 3             0              0              1           0
## 4             0              0              1           0
## 5             0              0              1           0
## 6             0              0              1           0
##   carry_awaygt8 carry_awayless1 carry_awaynever restaurant_less_than204~8
## 1           0             0             0                    1
## 2           0             0             0                    1
## 3           0             0             0                    1
## 4           0             0             0                    1
## 5           0             0             0                    1
```

```
## 6            0             0             0                    1
##   restaurant_less_than20gt8 restaurant_less_than20less1
## 1                         0                           0
## 2                         0                           0
## 3                         0                           0
## 4                         0                           0
## 5                         0                           0
## 6                         0                           0
##   restaurant_less_than20never restaurant20to504~8 restaurant20to50gt8
## 1                           0                   0                   0
## 2                           0                   0                   0
## 3                           0                   0                   0
## 4                           0                   0                   0
## 5                           0                   0                   0
## 6                           0                   0                   0
##   restaurant20to50less1 restaurant20to50never destination_passengerHome_Kid(s)
## 1                     0                     0                                0
## 2                     0                     0                                0
## 3                     0                     0                                0
## 4                     0                     0                                0
## 5                     0                     0                                0
## 6                     0                     0                                0
##   destination_passengerHome_Partner destination_passengerNo Urgent Place_Alone
## 1                                 0                                          1
## 2                                 0                                          0
## 3                                 0                                          0
## 4                                 0                                          0
## 5                                 0                                          0
## 6                                 0                                          0
##   destination_passengerNo Urgent Place_Friend(s)
## 1                                              0
## 2                                              1
## 3                                              1
```

```
## 4                                   1
## 5                                   1
## 6                                   1
##   destination_passengerNo Urgent Place_Kid(s)
## 1                                 0
## 2                                 0
## 3                                 0
## 4                                 0
## 5                                 0
## 6                                 0
##   destination_passengerNo Urgent Place_Partner destination_passengerWork_Alone
## 1                                  0                          0
## 2                                  0                          0
## 3                                  0                          0
## 4                                  0                          0
## 5                                  0                          0
## 6                                  0                          0
##   weather_temperatureSnowy_30 weather_temperatureSunny_30
## 1                      0                     0
## 2                      0                     0
## 3                      0                     0
## 4                      0                     0
## 5                      0                     0
## 6                      0                     0
##   weather_temperatureSunny_55 weather_temperatureSunny_80
## 1                      1                     0
## 2                      0                     1
## 3                      0                     1
## 4                      0                     1
## 5                      0                     1
## 6                      0                     1
##   maritalstatus_childrenDivorced_1 maritalstatus_childrenMarried partner_0
## 1                           0                               0
```

```
## 2                    0                        0
## 3                    0                        0
## 4                    0                        0
## 5                    0                        0
## 6                    0                        0
##   maritalstatus_childrenMarried partner_1 maritalstatus_childrenSingle_0
## 1                        0                  0
## 2                        0                  0
## 3                        0                  0
## 4                        0                  0
## 5                        0                  0
## 6                        0                  0
##   maritalstatus_childrenSingle_1 maritalstatus_childrenUnmarried partner_0
## 1                    0                        0
## 2                    0                        0
## 3                    0                        0
## 4                    0                        0
## 5                    0                        0
## 6                    0                        0
##   maritalstatus_childrenUnmarried partner_1 maritalstatus_childrenWidowed_0
## 1                          1                    0
## 2                          1                    0
## 3                          1                    0
## 4                          1                    0
## 5                          1                    0
## 6                          1                    0
##   maritalstatus_childrenWidowed_1
## 1                    0
## 2                    0
## 3                    0
## 4                    0
## 5                    0
## 6                    0
```

```
dim(df_categorical_encoded)

## [1] 12610    68
```

- Post one hot encoding we are combining categorical and numerical columns to evaluate the correlation between features.
  Code:

```
#combining encoded categorical and numerical columns
df_data_dummy_encoded <- cbind(df_data_dummy[numeric_cols], df_categorical_encoded)
```

- Dimension of the data set after one hot encoding. It is a known behavior count of features will increase after one hot encoding due to conversion of each class in each feature to a new column with binary values (0 & 1).

Code:

```
dim(df_data_dummy_encoded)
```

Output:

```
## [1] 12610    71
```

- Post one hot encoding it is ideal we need to evaluate for duplicate rows and drop them.

  Code:

```
#Dropping duplicates after one hot encoding
df_data_dummy_encoded <- df_data_dummy_encoded[!duplicated(df_data_dummy_encoded),
]
dim(df_data_dummy_encoded)

## [1] 12564    71
```

- Creating correlation matrix

```
#generating correlation matrix
cor_matrix <- cor(df_data_dummy_encoded, use = "complete.obs")

cor_matrix_melted <- melt(cor_matrix)
head(cor_matrix_melted)
```

```
##             Var1           Var2       value
## 1 direction_same direction_same  1.00000000
## 2              y direction_same  0.01434834
## 3      to_coupon direction_same -0.31392392
## 4       time10AM direction_same -0.24582455
## 5       time10PM direction_same  0.02299210
## 6        time2PM direction_same -0.22808759
```

- Finding the pairs of features which are highly correlated to each other. Here we have set a threshold of 0.5.

  Code:

```
# Find pairs of highly correlated variables with threshold above 0.5
threshold <- 0.5
highly_correlated <- which(abs(cor_matrix) > threshold, arr.ind = TRUE)

# printing the indices of the highly correlated pairs
print(highly_correlated)
```

```
##                                           row col
## direction_same                              1   1
## y                                           2   2
## to_coupon                                   3   3
## time10AM                                    4   4
## time10PM                                    5   5
## time2PM                                     6   6
## time6PM                                     7   7
## time7AM                                     8   8
## destination_passengerWork_Alone            58   8
## couponCarry out & Take away                 9   9
## couponCoffee House                         10  10
## couponRestaurant(<20)                      11  11
## couponRestaurant(20-50)                    12  12
## expiration2h                               13  13
## genderMale                                 14  14
```

```
## ageSeniors                                    15  15
## ageTeenagers                                  16  16
## ageYoung Adults                               17  17
## educationBachelors degree                     18  18
## educationSome college - no degree             21  18
## educationGraduate degree (Masters or Doctorate)  19  19
## educationHigh School Graduate                 20  20
## educationBachelors degree                     18  21
## educationSome college - no degree             21  21
## educationSome High School                     22  22
## occupationOthers                              23  23
## occupationProfessionals                       24  24
## occupationRetired                             25  25
## occupationService and sales                   26  26
## occupationStudent                             27  27
## occupationTechnicians                         28  28
## occupationUnemployed                          29  29
## incomeLow_income                              30  30
## incomeMedium_income                           31  30
## incomeLow_income                              30  31
## incomeMedium_income                           31  31
## bar4~8                                        32  32
## bargt8                                        33  33
## barless1                                      34  34
## barnever                                      35  34
## barless1                                      34  35
## barnever                                      35  35
## coffee_house4~8                               36  36
## coffee_housegt8                               37  37
## coffee_houseless1                             38  38
## coffee_housenever                             39  39
## carry_away4~8                                 40  40
## carry_awaygt8                                 41  41
```

```
## carry_awayless1                          42  42
## carry_awaynever                          43  43
## restaurant_less_than204~8                44  44
## restaurant_less_than20gt8                45  45
## restaurant_less_than20less1              46  46
## restaurant_less_than20never              47  47
## restaurant20to504~8                      48  48
## restaurant20to50gt8                      49  49
## restaurant20to50less1                    50  50
## restaurant20to50never                    51  51
## destination_passengerHome_Kid(s)         52  52
## destination_passengerHome_Partner        53  53
## destination_passengerNo Urgent Place_Alone       54  54
## destination_passengerNo Urgent Place_Friend(s)   55  55
## destination_passengerNo Urgent Place_Kid(s)      56  56
## destination_passengerNo Urgent Place_Partner     57  57
## time7AM                                   8  58
## destination_passengerWork_Alone          58  58
## weather_temperatureSnowy_30              59  59
## weather_temperatureSunny_30              60  60
## weather_temperatureSunny_55              61  61
## weather_temperatureSunny_80              62  61
## weather_temperatureSunny_55              61  62
## weather_temperatureSunny_80              62  62
## maritalstatus_childrenDivorced_1         63  63
## maritalstatus_childrenMarried partner_0  64  64
## maritalstatus_childrenMarried partner_1  65  65
## maritalstatus_childrenSingle_0           66  66
## maritalstatus_childrenSingle_1           67  67
## maritalstatus_childrenUnmarried partner_0  68  68
## maritalstatus_childrenUnmarried partner_1  69  69
## maritalstatus_childrenWidowed_0          70  70
## maritalstatus_childrenWidowed_1          71  71
```

```r
# Extracting the variables
correlated_var_names <- data.frame(
  Var1 = rownames(cor_matrix)[highly_correlated[, 1]],
  Var2 = colnames(cor_matrix)[highly_correlated[, 2]],
  Correlation = cor_matrix[highly_correlated]
)

# Remove duplicate pairs (present in the upper triangle of the matrix)
correlated_var_names <- correlated_var_names[correlated_var_names$Var1 <
correlated_var_names$Var2, ]
```

- Below are the feature pairs which have correlation value greater than 0.5 (either positive or negative)
- Positive value indicates positive association between one another and vice versa for negative value.
- Also, we observed that feature pair (destination_passengerWork_Alone and time7AM) are highly correlated to each other with the correlation value “1”.
- Based on the correlation output we are dropping feature “time7AM”.

```r
# Getting only high correlated pairs.
print(correlated_var_names)

##                             Var1                          Var2
## 9  destination_passengerWork_Alone                        time7AM
## 23       educationBachelors degree educationSome college - no degree
## 35                incomeLow_income             incomeMedium_income
## 41                        barless1                        barnever
## 71     weather_temperatureSunny_55      weather_temperatureSunny_80
##    Correlation
## 9    1.0000000
## 23  -0.5205121
## 35  -0.5714826
```

```
## 41  -0.5181823
## 71  -0.5289461

#Dropping highly correlated variables

df_data_dummy_encoded <- df_data_dummy_encoded[ ,!(names(df_data_dummy_encoded)
%in% c("time7AM"))]
```

- Below are the set of features which will be used in our model obtained after correlation analysis.

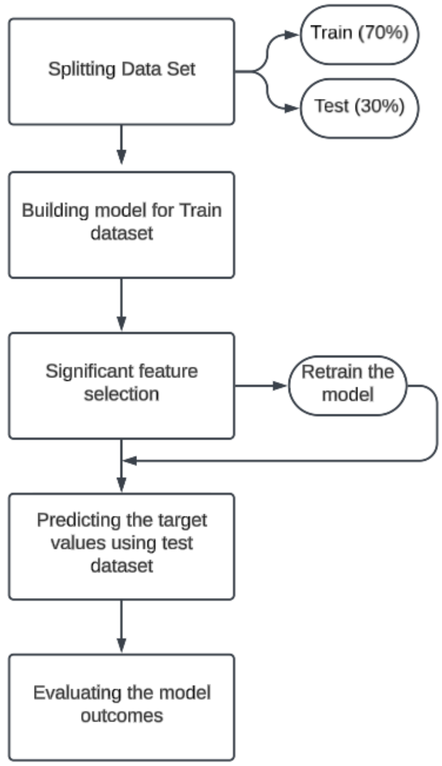
```
names(df_data_dummy_encoded)

##  [1] "direction_same"
##  [2] "y"
##  [3] "to_coupon"
##  [4] "time10AM"
##  [5] "time10PM"
##  [6] "time2PM"
##  [7] "time6PM"
##  [8] "couponCarry out & Take away"
##  [9] "couponCoffee House"
## [10] "couponRestaurant(<20)"
## [11] "couponRestaurant(20-50)"
## [12] "expiration2h"
## [13] "genderMale"
## [14] "ageSeniors"
## [15] "ageTeenagers"
## [16] "ageYoung Adults"
## [17] "educationBachelors degree"
## [18] "educationGraduate degree (Masters or Doctorate)"
## [19] "educationHigh School Graduate"
## [20] "educationSome college - no degree"
## [21] "educationSome High School"
## [22] "occupationOthers"
```

```
## [23] "occupationProfessionals"
## [24] "occupationRetired"
## [25] "occupationService and sales"
## [26] "occupationStudent"
## [27] "occupationTechnicians"
## [28] "occupationUnemployed"
## [29] "incomeLow_income"
## [30] "incomeMedium_income"
## [31] "bar4~8"
## [32] "bargt8"
## [33] "barless1"
## [34] "barnever"
## [35] "coffee_house4~8"
## [36] "coffee_housegt8"
## [37] "coffee_houseless1"
## [38] "coffee_housenever"
## [39] "carry_away4~8"
## [40] "carry_awaygt8"
## [41] "carry_awayless1"
## [42] "carry_awaynever"
## [43] "restaurant_less_than204~8"
## [44] "restaurant_less_than20gt8"
## [45] "restaurant_less_than20less1"
## [46] "restaurant_less_than20never"
## [47] "restaurant20to504~8"
## [48] "restaurant20to50gt8"
## [49] "restaurant20to50less1"
## [50] "restaurant20to50never"
## [51] "destination_passengerHome_Kid(s)"
## [52] "destination_passengerHome_Partner"
## [53] "destination_passengerNo Urgent Place_Alone"
## [54] "destination_passengerNo Urgent Place_Friend(s)"
## [55] "destination_passengerNo Urgent Place_Kid(s)"
```

```
## [56] "destination_passengerNo Urgent Place_Partner"
## [57] "destination_passengerWork_Alone"
## [58] "weather_temperatureSnowy_30"
## [59] "weather_temperatureSunny_30"
## [60] "weather_temperatureSunny_55"
## [61] "weather_temperatureSunny_80"
## [62] "maritalstatus_childrenDivorced_1"
## [63] "maritalstatus_childrenMarried partner_0"
## [64] "maritalstatus_childrenMarried partner_1"
## [65] "maritalstatus_childrenSingle_0"
## [66] "maritalstatus_childrenSingle_1"
## [67] "maritalstatus_childrenUnmarried partner_0"
## [68] "maritalstatus_childrenUnmarried partner_1"
## [69] "maritalstatus_childrenWidowed_0"
## [70] "maritalstatus_childrenWidowed_1"
```

VIII. Machine Learning Model Selection, Training and Evaluation

Workflow:



- Considering the binary nature of the target variable, classification models such as Logistic Regression, LDA (Linear Discriminant Analysis), and QDA (Quadratic Discriminant Analysis) are well-suited for this task. These models align with the dataset structure and the problem objective.
- Feature engineering, data preprocessing, and correlation analysis guarantee that the data is ready for model training and hence the results are more reliable.
- Metrics like accuracy, sensitivity, specificity, and precision will give a full assessment of how well the model works, making sure predictions are balanced and easy to understand.
- Comparing the results of Logistic Regression, LDA, and QDA will help determine the most effective model based on its ability to generalize and meet the project's objectives.

Logistic regression model

- With target variable as ‘y’ we are building a logistic regression model using the function “glm” because of binary nature classification model like logistic regression is used here.

```
############################## Building Model part #######################

##Building the logistic regression model for the population data

model_coupon_dummy <- glm(y ~ ., data=df_data_dummy_encoded, family = binomial)
summary(model_coupon_dummy)

##
## Call:
## glm(formula = y ~ ., family = binomial, data = df_data_dummy_encoded)
##
## Coefficients: (1 not defined because of singularities)
##                                                   Estimate Std. Error z value
## (Intercept)                                      -0.578525   0.400595  -1.444
## direction_same                                    0.486029   0.067165   7.236
## to_coupon                                        -0.013171   0.036701  -0.359
## time10AM                                         -0.020178   0.096223  -0.210
## time10PM                                         -0.211416   0.075439  -2.802
## time2PM                                          -0.118457   0.095887  -1.235
## time6PM                                           0.199374   0.062572   3.186
## `couponCarry out & Take away`                     1.687107   0.072721  23.200
## `couponCoffee House`                              0.512493   0.064799   7.909
## `couponRestaurant(<20)`                           1.536562   0.071881  21.376
## `couponRestaurant(20-50)`                         0.391459   0.079009   4.955
## expiration2h                                     -0.830374   0.043721 -18.993
## genderMale                                        0.210063   0.043523   4.826
## ageSeniors                                       -0.162976   0.073764  -2.209
## ageTeenagers                                     -0.044711   0.129216  -0.346
## `ageYoung Adults`                                -0.029894   0.054910  -0.544
## `educationBachelors degree`                      -0.135354   0.077321  -1.751
## `educationGraduate degree (Masters or Doctorate)` -0.333696   0.089795  -3.716
## `educationHigh School Graduate`                   0.163347   0.104977   1.556
## `educationSome college - no degree`               0.066594   0.077784   0.856
## `educationSome High School`                       0.674238   0.283079   2.382
```

```
## occupationOthers                              0.005538   0.120062   0.046
## occupationProfessionals                       0.093505   0.100025   0.935
## occupationRetired                            -0.081707   0.146365  -0.558
## `occupationService and sales`                 0.170444   0.109457   1.557
## occupationStudent                             0.078661   0.113028   0.696
## occupationTechnicians                         0.311779   0.107800   2.892
## occupationUnemployed                          0.019333   0.105036   0.184
## incomeLow_income                              0.144915   0.058591   2.473
## incomeMedium_income                           0.121530   0.054638   2.224
## `bar4~8`                                     -0.112636   0.086086  -1.308
## bargt8                                       -0.436622   0.143992  -3.032
## barless1                                     -0.167041   0.064020  -2.609
## barnever                                     -0.199325   0.061426  -3.245
## `coffee_house4~8`                            -0.043729   0.070391  -0.621
## coffee_housegt8                              -0.341712   0.084922  -4.024
## coffee_houseless1                            -0.457484   0.057539  -7.951
## coffee_housenever                            -0.916468   0.062727 -14.610
## `carry_away4~8`                              -0.067000   0.050184  -1.335
## carry_awaygt8                                -0.142246   0.074094  -1.920
## carry_awayless1                              -0.185444   0.063820  -2.906
## carry_awaynever                               0.054834   0.189269   0.290
## `restaurant_less_than204~8`                   0.035645   0.052241   0.682
## restaurant_less_than20gt8                     0.155126   0.085385   1.817
## restaurant_less_than20less1                   0.038621   0.062216   0.621
## restaurant_less_than20never                   0.269920   0.164281   1.643
## `restaurant20to504~8`                         0.099122   0.099722   0.994
## restaurant20to50gt8                           0.072770   0.177326   0.410
## restaurant20to50less1                        -0.145566   0.051007  -2.854
## restaurant20to50never                        -0.292129   0.068372  -4.273
## `destination_passengerHome_Kid(s)`            0.159670   0.198813   0.803
## destination_passengerHome_Partner             0.243619   0.155790   1.564
## `destination_passengerNo Urgent Place_Alone`  0.813013   0.103326   7.868
## `destination_passengerNo Urgent Place_Friend(s)`   1.011952   0.080708  12.538
```

```
## `destination_passengerNo Urgent Place_Kid(s)`      0.287833   0.105099   2.739
## `destination_passengerNo Urgent Place_Partner`     1.084042   0.113823   9.524
## destination_passengerWork_Alone                          NA         NA      NA
## weather_temperatureSnowy_30                     -0.161592   0.090503  -1.785
## weather_temperatureSunny_30                      0.187386   0.102628   1.826
## weather_temperatureSunny_55                      0.537859   0.080883   6.650
## weather_temperatureSunny_80                      0.366852   0.073876   4.966
## maritalstatus_childrenDivorced_1                -0.167426   0.377962  -0.443
## `maritalstatus_childrenMarried partner_0`        -0.056340   0.371295  -0.152
## `maritalstatus_childrenMarried partner_1`         0.089733   0.366354   0.245
## maritalstatus_childrenSingle_0                   0.168138   0.368393   0.456
## maritalstatus_childrenSingle_1                   0.013036   0.375553   0.035
## `maritalstatus_childrenUnmarried partner_0`      -0.084496   0.370963  -0.228
## `maritalstatus_childrenUnmarried partner_1`       0.026264   0.376444   0.070
## maritalstatus_childrenWidowed_0                 -0.537199   0.512092  -1.049
## maritalstatus_childrenWidowed_1                  0.334898   0.437174   0.766
##                                              Pr(>|z|)
## (Intercept)                                  0.148692
## direction_same                                4.61e-13 ***
## to_coupon                                    0.719696
## time10AM                                     0.833902
## time10PM                                     0.005071 **
## time2PM                                      0.216690
## time6PM                                      0.001441 **
## `couponCarry out & Take away`                 < 2e-16 ***
## `couponCoffee House`                         2.60e-15 ***
## `couponRestaurant(<20)`                       < 2e-16 ***
## `couponRestaurant(20-50)`                    7.25e-07 ***
## expiration2h                                  < 2e-16 ***
## genderMale                                   1.39e-06 ***
## ageSeniors                                   0.027146 *
## ageTeenagers                                 0.729328
## `ageYoung Adults`                            0.586150
```

```
## `educationBachelors degree`                     0.080023 .
## `educationGraduate degree (Masters or Doctorate)` 0.000202 ***
## `educationHigh School Graduate`                 0.119701
## `educationSome college - no degree`             0.391921
## `educationSome High School`                     0.017228 *
## occupationOthers                                0.963207
## occupationProfessionals                         0.349883
## occupationRetired                               0.576681
## `occupationService and sales`                   0.119427
## occupationStudent                               0.486465
## occupationTechnicians                           0.003825 **
## occupationUnemployed                            0.853966
## incomeLow_income                                0.013387 *
## incomeMedium_income                             0.026130 *
## `bar4~8`                                        0.190736
## bargt8                                          0.002427 **
## barless1                                        0.009075 **
## barnever                                        0.001175 **
## `coffee_house4~8`                               0.534445
## coffee_housegt8                                 5.73e-05 ***
## coffee_houseless1                               1.85e-15 ***
## coffee_housenever                                < 2e-16 ***
## `carry_away4~8`                                 0.181851
## carry_awaygt8                                   0.054883 .
## carry_awayless1                                 0.003664 **
## carry_awaynever                                 0.772033
## `restaurant_less_than204~8`                     0.495033
## restaurant_less_than20gt8                       0.069251 .
## restaurant_less_than20less1                     0.534764
## restaurant_less_than20never                     0.100374
## `restaurant20to504~8`                           0.320234
## restaurant20to50gt8                             0.681531
## restaurant20to50less1                           0.004320 **
```

```
## restaurant20to50never                              1.93e-05 ***
## `destination_passengerHome_Kid(s)`               0.421907
## destination_passengerHome_Partner                0.117873
## `destination_passengerNo Urgent Place_Alone`      3.59e-15 ***
## `destination_passengerNo Urgent Place_Friend(s)`   < 2e-16 ***
## `destination_passengerNo Urgent Place_Kid(s)`     0.006168 **
## `destination_passengerNo Urgent Place_Partner`     < 2e-16 ***
## destination_passengerWork_Alone                         NA
## weather_temperatureSnowy_30                     0.074183 .
## weather_temperatureSunny_30                     0.067871 .
## weather_temperatureSunny_55                     2.93e-11 ***
## weather_temperatureSunny_80                     6.84e-07 ***
## maritalstatus_childrenDivorced_1                0.657787
## `maritalstatus_childrenMarried partner_0`        0.879394
## `maritalstatus_childrenMarried partner_1`        0.806506
## maritalstatus_childrenSingle_0                  0.648095
## maritalstatus_childrenSingle_1                  0.972310
## `maritalstatus_childrenUnmarried partner_0`       0.819822
## `maritalstatus_childrenUnmarried partner_1`       0.944377
## maritalstatus_childrenWidowed_0                 0.294165
## maritalstatus_childrenWidowed_1                 0.443646
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 17185  on 12563  degrees of freedom
## Residual deviance: 14871  on 12495  degrees of freedom
## AIC: 15009
##
## Number of Fisher Scoring iterations: 4
```

- The feature **destination_passengerWork_Alone** was observed to have coefficients as NA in the model output, indicating singularity.

- Singularity occurs when a feature is perfectly correlated with other variables or is a linear combination of them, making it redundant for the model.
- In this case, the feature does not provide unique information and leads to instability in parameter estimation.
- To address this issue, the feature was dropped to ensure the model's robustness and eliminate multicollinearity. This step helps prevent overfitting, simplifies the model, and improves computational efficiency without compromising predictive accuracy.
- To analyze which are the features can expressed as a linear combination to get feature **destination_passengerWork_Alone** can be done using alias command output.

Code:

```
#post modeling we found singularity issue with the variable "destination_passengerWork_Alone"

alias(model_coupon_dummy)
```

Output:

```
## Model :
## y ~ direction_same + to_coupon + time10AM + time10PM + time2PM +
##     time6PM + `couponCarry out & Take away` + `couponCoffee House` +
##     `couponRestaurant(<20)` + `couponRestaurant(20-50)` + expiration2h +
##     genderMale + ageSeniors + ageTeenagers + `ageYoung Adults` +
##     `educationBachelors degree` + `educationGraduate degree (Masters or Doctorate)` +
##     `educationHigh School Graduate` + `educationSome college - no degree` +
##     `educationSome High School` + occupationOthers + occupationProfessionals +
##     occupationRetired + `occupationService and sales` + occupationStudent +
##     occupationTechnicians + occupationUnemployed + incomeLow_income +
##     incomeMedium_income + `bar4~8` + bargt8 + barless1 + barnever +
##     `coffee_house4~8` + coffee_housegt8 + coffee_houseless1 +
##     coffee_housenever + `carry_away4~8` + carry_awaygt8 + carry_awayless1 +
##     carry_awaynever + `restaurant_less_than204~8` + restaurant_less_than20gt8 +
##     restaurant_less_than20less1 + restaurant_less_than20never +
##     `restaurant20to504~8` + restaurant20to50gt8 + restaurant20to50less1 +
##     restaurant20to50never + `destination_passengerHome_Kid(s)` +
##     destination_passengerHome_Partner + `destination_passengerNo Urgent Place_Alone` +
##     `destination_passengerNo Urgent Place_Friend(s)` + `destination_passengerNo Urgent
Place_Kid(s)` +
```

```
##    `destination_passengerNo Urgent Place_Partner` + destination_passengerWork_Alone +
##    weather_temperatureSnowy_30 + weather_temperatureSunny_30 +
##    weather_temperatureSunny_55 + weather_temperatureSunny_80 +
##    maritalstatus_childrenDivorced_1 + `maritalstatus_childrenMarried partner_0` +
##    `maritalstatus_childrenMarried partner_1` + maritalstatus_childrenSingle_0 +
##    maritalstatus_childrenSingle_1 + `maritalstatus_childrenUnmarried partner_0` +
##    `maritalstatus_childrenUnmarried partner_1` + maritalstatus_childrenWidowed_0 +
##    maritalstatus_childrenWidowed_1
##
## Complete :
##                              (Intercept) direction_same to_coupon time10AM
## destination_passengerWork_Alone  1           0              0        -1
##                              time10PM time2PM time6PM
## destination_passengerWork_Alone -1       -1      -1
##                              `couponCarry out & Take away`
## destination_passengerWork_Alone  0
##                              `couponCoffee House` `couponRestaurant(<20)`
## destination_passengerWork_Alone  0                    0
##                              `couponRestaurant(20-50)` expiration2h
## destination_passengerWork_Alone  0                         0
##                              genderMale ageSeniors ageTeenagers
## destination_passengerWork_Alone  0          0          0
##                              `ageYoung Adults` `educationBachelors degree`
## destination_passengerWork_Alone  0                 0
##                              `educationGraduate degree (Masters or Doctorate)`
## destination_passengerWork_Alone  0
##                              `educationHigh School Graduate`
## destination_passengerWork_Alone  0
##                              `educationSome college - no degree`
## destination_passengerWork_Alone  0
##                              `educationSome High School` occupationOthers
## destination_passengerWork_Alone  0                           0
##                              occupationProfessionals occupationRetired
```

```
## destination_passengerWork_Alone  0                       0
##                                `occupationService and sales` occupationStudent
## destination_passengerWork_Alone  0                             0
##                                occupationTechnicians occupationUnemployed
## destination_passengerWork_Alone  0                     0
##                                incomeLow_income incomeMedium_income `bar4~8`
## destination_passengerWork_Alone  0                0                    0
##                                bargt8 barless1 barnever `coffee_house4~8`
## destination_passengerWork_Alone  0      0        0        0
##                                coffee_housegt8 coffee_houseless1
## destination_passengerWork_Alone  0               0
##                                coffee_housenever `carry_away4~8` carry_awaygt8
## destination_passengerWork_Alone  0                 0               0
##                                carry_awayless1 carry_awaynever
## destination_passengerWork_Alone  0               0
##                                `restaurant_less_than204~8`
## destination_passengerWork_Alone  0
##                                restaurant_less_than20gt8
## destination_passengerWork_Alone  0
##                                restaurant_less_than20less1
## destination_passengerWork_Alone  0
##                                restaurant_less_than20never
## destination_passengerWork_Alone  0
##                                `restaurant20to504~8` restaurant20to50gt8
## destination_passengerWork_Alone  0                     0
##                                restaurant20to50less1 restaurant20to50never
## destination_passengerWork_Alone  0                     0
##                                `destination_passengerHome_Kid(s)`
## destination_passengerWork_Alone  0
##                                destination_passengerHome_Partner
## destination_passengerWork_Alone  0
##                                `destination_passengerNo Urgent Place_Alone`
## destination_passengerWork_Alone  0
```

```
##                               `destination_passengerNo Urgent Place_Friend(s)`
## destination_passengerWork_Alone  0
##                               `destination_passengerNo Urgent Place_Kid(s)`
## destination_passengerWork_Alone  0
##                               `destination_passengerNo Urgent Place_Partner`
## destination_passengerWork_Alone  0
##                               weather_temperatureSnowy_30
## destination_passengerWork_Alone  0
##                               weather_temperatureSunny_30
## destination_passengerWork_Alone  0
##                               weather_temperatureSunny_55
## destination_passengerWork_Alone  0
##                               weather_temperatureSunny_80
## destination_passengerWork_Alone  0
##                               maritalstatus_childrenDivorced_1
## destination_passengerWork_Alone  0
##                               `maritalstatus_childrenMarried partner_0`
## destination_passengerWork_Alone  0
##                               `maritalstatus_childrenMarried partner_1`
## destination_passengerWork_Alone  0
##                               maritalstatus_childrenSingle_0
## destination_passengerWork_Alone  0
##                               maritalstatus_childrenSingle_1
## destination_passengerWork_Alone  0
##                               `maritalstatus_childrenUnmarried partner_0`
## destination_passengerWork_Alone  0
##                               `maritalstatus_childrenUnmarried partner_1`
## destination_passengerWork_Alone  0
##                               maritalstatus_childrenWidowed_0
## destination_passengerWork_Alone  0
##                               maritalstatus_childrenWidowed_1
## destination_passengerWork_Alone  0
```

```
alias(model_coupon_dummy)$Complete
```

```
##                              (Intercept) direction_same to_coupon time10AM
## destination_passengerWork_Alone  1           0              0       -1
##                              time10PM time2PM time6PM
## destination_passengerWork_Alone -1       -1      -1
##                              `couponCarry out & Take away`
## destination_passengerWork_Alone  0
##                              `couponCoffee House` `couponRestaurant(<20)`
## destination_passengerWork_Alone  0                    0
##                              `couponRestaurant(20-50)` expiration2h
## destination_passengerWork_Alone  0                         0
##                              genderMale ageSeniors ageTeenagers
## destination_passengerWork_Alone  0          0          0
##                              `ageYoung Adults` `educationBachelors degree`
## destination_passengerWork_Alone  0                 0
##                              `educationGraduate degree (Masters or Doctorate)`
## destination_passengerWork_Alone  0
##                              `educationHigh School Graduate`
## destination_passengerWork_Alone  0
##                              `educationSome college - no degree`
## destination_passengerWork_Alone  0
##                              `educationSome High School` occupationOthers
## destination_passengerWork_Alone  0                          0
##                              occupationProfessionals occupationRetired
## destination_passengerWork_Alone  0                       0
##                              `occupationService and sales` occupationStudent
## destination_passengerWork_Alone  0                             0
##                              occupationTechnicians occupationUnemployed
## destination_passengerWork_Alone  0                     0
##                              incomeLow_income incomeMedium_income `bar4~8`
## destination_passengerWork_Alone  0                0                   0
##                              bargt8 barless1 barnever `coffee_house4~8`
## destination_passengerWork_Alone  0      0        0        0
##                              coffee_housegt8 coffee_houseless1
```

```
## destination_passengerWork_Alone  0              0
##                              coffee_housenever `carry_away4~8` carry_awaygt8
## destination_passengerWork_Alone  0                0              0
##                              carry_awayless1 carry_awaynever
## destination_passengerWork_Alone  0              0
##                              `restaurant_less_than204~8`
## destination_passengerWork_Alone  0
##                              restaurant_less_than20gt8
## destination_passengerWork_Alone  0
##                              restaurant_less_than20less1
## destination_passengerWork_Alone  0
##                              restaurant_less_than20never
## destination_passengerWork_Alone  0
##                              `restaurant20to504~8` restaurant20to50gt8
## destination_passengerWork_Alone  0                   0
##                              restaurant20to50less1 restaurant20to50never
## destination_passengerWork_Alone  0                   0
##                              `destination_passengerHome_Kid(s)`
## destination_passengerWork_Alone  0
##                              destination_passengerHome_Partner
## destination_passengerWork_Alone  0
##                              `destination_passengerNo Urgent Place_Alone`
## destination_passengerWork_Alone  0
##                              `destination_passengerNo Urgent Place_Friend(s)`
## destination_passengerWork_Alone  0
##                              `destination_passengerNo Urgent Place_Kid(s)`
## destination_passengerWork_Alone  0
##                              `destination_passengerNo Urgent Place_Partner`
## destination_passengerWork_Alone  0
##                              weather_temperatureSnowy_30
## destination_passengerWork_Alone  0
##                              weather_temperatureSunny_30
## destination_passengerWork_Alone  0
```

```
##                             weather_temperatureSunny_55
## destination_passengerWork_Alone  0
##                             weather_temperatureSunny_80
## destination_passengerWork_Alone  0
##                             maritalstatus_childrenDivorced_1
## destination_passengerWork_Alone  0
##                             `maritalstatus_childrenMarried partner_0`
## destination_passengerWork_Alone  0
##                             `maritalstatus_childrenMarried partner_1`
## destination_passengerWork_Alone  0
##                             maritalstatus_childrenSingle_0
## destination_passengerWork_Alone  0
##                             maritalstatus_childrenSingle_1
## destination_passengerWork_Alone  0
##                             `maritalstatus_childrenUnmarried partner_0`
## destination_passengerWork_Alone  0
##                             `maritalstatus_childrenUnmarried partner_1`
## destination_passengerWork_Alone  0
##                             maritalstatus_childrenWidowed_0
## destination_passengerWork_Alone  0
##                             maritalstatus_childrenWidowed_1
## destination_passengerWork_Alone  0
```

```
#Handling singularity issue by dropping column destination_passengerWork_Alone as it can expressed linearly by other variables.
df_data_dummy_encoded <- df_data_dummy_encoded[ ,!(names(df_data_dummy_encoded)
%in% c("destination_passengerWork_Alone"))]

model_coupon_dummy <- glm(y ~ ., data=df_data_dummy_encoded, family = binomial)
summary(model_coupon_dummy)
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial, data = df_data_dummy_encoded)
##
```

```
## Coefficients:
##                                                   Estimate Std. Error z value
## (Intercept)                                      -0.578525   0.400595  -1.444
## direction_same                                    0.486029   0.067165   7.236
## to_coupon                                        -0.013171   0.036701  -0.359
## time10AM                                         -0.020178   0.096223  -0.210
## time10PM                                         -0.211416   0.075439  -2.802
## time2PM                                          -0.118457   0.095887  -1.235
## time6PM                                           0.199374   0.062572   3.186
## `couponCarry out & Take away`                     1.687107   0.072721  23.200
## `couponCoffee House`                              0.512493   0.064799   7.909
## `couponRestaurant(<20)`                           1.536562   0.071881  21.376
## `couponRestaurant(20-50)`                         0.391459   0.079009   4.955
## expiration2h                                     -0.830374   0.043721 -18.993
## genderMale                                        0.210063   0.043523   4.826
## ageSeniors                                       -0.162976   0.073764  -2.209
## ageTeenagers                                     -0.044711   0.129216  -0.346
## `ageYoung Adults`                                -0.029894   0.054910  -0.544
## `educationBachelors degree`                      -0.135354   0.077321  -1.751
## `educationGraduate degree (Masters or Doctorate)` -0.333696   0.089795  -3.716
## `educationHigh School Graduate`                   0.163347   0.104977   1.556
## `educationSome college - no degree`               0.066594   0.077784   0.856
## `educationSome High School`                       0.674238   0.283079   2.382
## occupationOthers                                  0.005538   0.120062   0.046
## occupationProfessionals                           0.093505   0.100025   0.935
## occupationRetired                                -0.081707   0.146365  -0.558
## `occupationService and sales`                     0.170444   0.109457   1.557
## occupationStudent                                 0.078661   0.113028   0.696
## occupationTechnicians                             0.311779   0.107800   2.892
## occupationUnemployed                              0.019333   0.105036   0.184
## incomeLow_income                                  0.144915   0.058591   2.473
## incomeMedium_income                               0.121530   0.054638   2.224
## `bar4~8`                                         -0.112636   0.086086  -1.308
```

```
## bargt8                                     -0.436622   0.143992  -3.032
## barless1                                   -0.167041   0.064020  -2.609
## barnever                                   -0.199325   0.061426  -3.245
## `coffee_house4~8`                          -0.043729   0.070391  -0.621
## coffee_housegt8                            -0.341712   0.084922  -4.024
## coffee_houseless1                          -0.457484   0.057539  -7.951
## coffee_housenever                          -0.916468   0.062727 -14.610
## `carry_away4~8`                            -0.067000   0.050184  -1.335
## carry_awaygt8                              -0.142246   0.074094  -1.920
## carry_awayless1                            -0.185444   0.063820  -2.906
## carry_awaynever                             0.054834   0.189269   0.290
## `restaurant_less_than204~8`                 0.035645   0.052241   0.682
## restaurant_less_than20gt8                   0.155126   0.085385   1.817
## restaurant_less_than20less1                 0.038621   0.062216   0.621
## restaurant_less_than20never                 0.269920   0.164281   1.643
## `restaurant20to504~8`                       0.099122   0.099722   0.994
## restaurant20to50gt8                         0.072770   0.177326   0.410
## restaurant20to50less1                      -0.145566   0.051007  -2.854
## restaurant20to50never                      -0.292129   0.068372  -4.273
## `destination_passengerHome_Kid(s)`          0.159670   0.198813   0.803
## destination_passengerHome_Partner           0.243619   0.155790   1.564
## `destination_passengerNo Urgent Place_Alone`       0.813013   0.103326   7.868
## `destination_passengerNo Urgent Place_Friend(s)`   1.011952   0.080708  12.538
## `destination_passengerNo Urgent Place_Kid(s)`      0.287833   0.105099   2.739
## `destination_passengerNo Urgent Place_Partner`     1.084042   0.113823   9.524
## weather_temperatureSnowy_30                -0.161592   0.090503  -1.785
## weather_temperatureSunny_30                 0.187386   0.102628   1.826
## weather_temperatureSunny_55                 0.537859   0.080883   6.650
## weather_temperatureSunny_80                 0.366852   0.073876   4.966
## maritalstatus_childrenDivorced_1           -0.167426   0.377962  -0.443
## `maritalstatus_childrenMarried partner_0`  -0.056340   0.371295  -0.152
## `maritalstatus_childrenMarried partner_1`   0.089733   0.366354   0.245
## maritalstatus_childrenSingle_0              0.168138   0.368393   0.456
```

```
## maritalstatus_childrenSingle_1                    0.013036   0.375553   0.035
## `maritalstatus_childrenUnmarried partner_0`       -0.084496   0.370963  -0.228
## `maritalstatus_childrenUnmarried partner_1`        0.026264   0.376444   0.070
## maritalstatus_childrenWidowed_0                   -0.537199   0.512092  -1.049
## maritalstatus_childrenWidowed_1                    0.334898   0.437174   0.766
##                                                   Pr(>|z|)
## (Intercept)                                       0.148692
## direction_same                                    4.61e-13 ***
## to_coupon                                         0.719696
## time10AM                                          0.833902
## time10PM                                          0.005071 **
## time2PM                                           0.216690
## time6PM                                           0.001441 **
## `couponCarry out & Take away`                      < 2e-16 ***
## `couponCoffee House`                              2.60e-15 ***
## `couponRestaurant(<20)`                            < 2e-16 ***
## `couponRestaurant(20-50)`                         7.25e-07 ***
## expiration2h                                       < 2e-16 ***
## genderMale                                        1.39e-06 ***
## ageSeniors                                        0.027146 *
## ageTeenagers                                      0.729328
## `ageYoung Adults`                                 0.586150
## `educationBachelors degree`                       0.080023 .
## `educationGraduate degree (Masters or Doctorate)` 0.000202 ***
## `educationHigh School Graduate`                   0.119701
## `educationSome college - no degree`               0.391921
## `educationSome High School`                       0.017228 *
## occupationOthers                                  0.963207
## occupationProfessionals                           0.349883
## occupationRetired                                 0.576681
## `occupationService and sales`                     0.119427
## occupationStudent                                 0.486465
## occupationTechnicians                             0.003825 **
```

```
## occupationUnemployed                          0.853966
## incomeLow_income                              0.013387 *
## incomeMedium_income                           0.026130 *
## `bar4~8`                                      0.190736
## bargt8                                        0.002427 **
## barless1                                      0.009075 **
## barnever                                      0.001175 **
## `coffee_house4~8`                             0.534445
## coffee_housegt8                               5.73e-05 ***
## coffee_houseless1                             1.85e-15 ***
## coffee_housenever                              < 2e-16 ***
## `carry_away4~8`                               0.181851
## carry_awaygt8                                 0.054883 .
## carry_awayless1                               0.003664 **
## carry_awaynever                               0.772033
## `restaurant_less_than204~8`                   0.495033
## restaurant_less_than20gt8                     0.069251 .
## restaurant_less_than20less1                   0.534764
## restaurant_less_than20never                   0.100374
## `restaurant20to504~8`                         0.320234
## restaurant20to50gt8                           0.681531
## restaurant20to50less1                         0.004320 **
## restaurant20to50never                         1.93e-05 ***
## `destination_passengerHome_Kid(s)`            0.421907
## destination_passengerHome_Partner             0.117873
## `destination_passengerNo Urgent Place_Alone`      3.59e-15 ***
## `destination_passengerNo Urgent Place_Friend(s)`   < 2e-16 ***
## `destination_passengerNo Urgent Place_Kid(s)`     0.006168 **
## `destination_passengerNo Urgent Place_Partner`     < 2e-16 ***
## weather_temperatureSnowy_30                   0.074183 .
## weather_temperatureSunny_30                   0.067871 .
## weather_temperatureSunny_55                   2.93e-11 ***
## weather_temperatureSunny_80                   6.84e-07 ***
```

```
## maritalstatus_childrenDivorced_1                0.657787
## `maritalstatus_childrenMarried partner_0`        0.879394
## `maritalstatus_childrenMarried partner_1`        0.806506
## maritalstatus_childrenSingle_0                  0.648095
## maritalstatus_childrenSingle_1                  0.972310
## `maritalstatus_childrenUnmarried partner_0`      0.819822
## `maritalstatus_childrenUnmarried partner_1`      0.944377
## maritalstatus_childrenWidowed_0                 0.294165
## maritalstatus_childrenWidowed_1                 0.443646
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 17185  on 12563  degrees of freedom
## Residual deviance: 14871  on 12495  degrees of freedom
## AIC: 15009
##
## Number of Fisher Scoring iterations: 4
```

IX. Variation Inflation Factor (VIF)

Key characteristics and importance of VIF

- VIF is an important measure to find and fix multicollinearity. Multicollinearity arises when two or more independent variables in a dataset are highly related to each other. High multicollinearity can make it difficult to understand the coefficients, increase standard errors, and lower the trustworthiness of the model.
- VIF helps in finding features that cause multicollinearity issues, it helps remove or change problem variables. The selection of features with lower VIF values is one of the approaches to making a model stable, easy to interpret, and good at making predictions.
- In this work, the VIF was computed in order to evaluate multicollinearity between predictors, so the most independent features contributing the most to the model could be selected. This will enhance model robustness and eliminate any redundancy-related issues.

Process involved in calculating VIF:

- Prepare the Dataset for calculating the VIF:

Ensure the dataset is cleaned and preprocessed, with all features ready for analysis (e.g., handle missing values and encode categorical variables if necessary).

- Fit the Regression Models:
  For each predictor variable in the dataset, fit a regression model where the variable is treated as the dependent variable, and all other predictors are treated as independent variables.
- Obtaining R-Squared Values from the regression models:
  For each regression model, calculate the R-squared value, which indicates how well the remaining predictors explain the variability of the target variable.
- Calculate VIF for each feature:
  Use the R-squared value to compute the VIF for each predictor. Higher VIF values indicate stronger multicollinearity.

Below are the key factors considered during VIF analysis.

- A VIF value of 1 indicates no multicollinearity.
- VIF values between 1 and 5 suggest moderate multicollinearity.
- VIF values greater than 10 indicate significant multicollinearity, requiring corrective action.
- Drop or combine features with high VIF values.

Displaying VIF value for each feature.

Code:

```
#Selecting the best features and to tackle multi colinearity issue with VIF
#calculating the vif values

vif_values <- vif(model_coupon_dummy)
```

Output:

```
vif_values

##                              direction_same
##                                  1.995175
##                                  to_coupon
##                                  1.567131
##                                  time10AM
##                                  3.597992
```

```
##                                          time10PM
##                                          1.872990
##                                           time2PM
##                                          2.984564
##                                           time6PM
##                                          1.896156
##                       `couponCarry out & Take away`
##                                          1.855566
##                               `couponCoffee House`
##                                          2.416839
##                             `couponRestaurant(<20)`
##                                          2.066552
##                            `couponRestaurant(20-50)`
##                                          1.756157
##                                     expiration2h
##                                          1.211678
##                                       genderMale
##                                          1.207098
##                                       ageSeniors
##                                          1.705809
##                                     ageTeenagers
##                                          1.699654
##                                 `ageYoung Adults`
##                                          1.886583
##                        `educationBachelors degree`
##                                          3.449653
## `educationGraduate degree (Masters or Doctorate)`
##                                          2.604596
##                     `educationHigh School Graduate`
##                                          1.867868
##                 `educationSome college - no degree`
##                                          3.451725
##                        `educationSome High School`
```

```
##                                 1.143719
##                          occupationOthers
##                                 2.595574
##                   occupationProfessionals
##                                 5.573740
##                         occupationRetired
##            `occupationService and sales`
##                                 3.126661
##                         occupationStudent
##                                 3.532986
##                     occupationTechnicians
##                                 3.091144
##                      occupationUnemployed
##                                 3.589794
##                          incomeLow_income
##                                 2.073564
##                       incomeMedium_income
##                                 1.709161
##                                 `bar4~8`
##                                 1.421366
##                                   bargt8
##                                 1.420840
##                                 barless1
##                                 2.073418
##                                 barnever
##                                 2.356368
##                        `coffee_house4~8`
##                                 1.493218
##                          coffee_housegt8
##                                 1.491405
##                        coffee_houseless1
##                                 1.745615
```

```
##                        coffee_housenever
##                              1.833642
##                          `carry_away4~8`
##                              1.433589
##                           carry_awaygt8
##                              1.504363
##                         carry_awayless1
##                              1.332439
##                         carry_awaynever
##                              1.131370
##               `restaurant_less_than204~8`
##                              1.399302
##                restaurant_less_than20gt8
##                              1.668635
##              restaurant_less_than20less1
##                              1.361371
##              restaurant_less_than20never
##                              1.162061
##                   `restaurant20to504~8`
##                              1.296489
##                    restaurant20to50gt8
##                              1.558924
##                  restaurant20to50less1
##                              1.659975
##                  restaurant20to50never
##                              1.696100
##        `destination_passengerHome_Kid(s)`
##                              1.129598
##         destination_passengerHome_Partner
##                              1.209229
##  `destination_passengerNo Urgent Place_Alone`
##                              2.496990
## `destination_passengerNo Urgent Place_Friend(s)`
```

```
##                                     3.054879
##   `destination_passengerNo Urgent Place_Kid(s)`
##                                     1.898592
##   `destination_passengerNo Urgent Place_Partner`
##                                     2.039466
##                      weather_temperatureSnowy_30
##                                     1.955129
##                      weather_temperatureSunny_30
##                                     1.720991
##                      weather_temperatureSunny_55
##                                     2.742280
##                      weather_temperatureSunny_80
##                                     3.476657
##                  maritalstatus_childrenDivorced_1
##                                    13.378621
##         `maritalstatus_childrenMarried partner_0`
##                                    37.364715
##         `maritalstatus_childrenMarried partner_1`
##                                    69.877954
##                    maritalstatus_childrenSingle_0
##                                    75.467502
##                    maritalstatus_childrenSingle_1
##                                    15.809786
##       `maritalstatus_childrenUnmarried partner_0`
##                                    39.980304
##       `maritalstatus_childrenUnmarried partner_1`
##                                    15.294316
##                   maritalstatus_childrenWidowed_0
##                                     2.230093
##                   maritalstatus_childrenWidowed_1
##                                     3.392867
```

- Filtering the features which is having VIF value with greater than 5

Code:

```r
#Printing features with has VIF Value greater than 5
vif_df <- data.frame(Variable = names(vif_values), VIF = vif_values)
high_vif_vars <- vif_df[vif_values > 5,]
```

Output:

```r
print(high_vif_vars)

##                                                                        Variable
## occupationProfessionals                                 occupationProfessionals
## maritalstatus_childrenDivorced_1                 maritalstatus_childrenDivorced_1
## `maritalstatus_childrenMarried partner_0`     `maritalstatus_childrenMarried partner_0`
## `maritalstatus_childrenMarried partner_1`     `maritalstatus_childrenMarried partner_1`
## maritalstatus_childrenSingle_0                     maritalstatus_childrenSingle_0
## maritalstatus_childrenSingle_1                     maritalstatus_childrenSingle_1
## `maritalstatus_childrenUnmarried partner_0` `maritalstatus_childrenUnmarried partner_0`
## `maritalstatus_childrenUnmarried partner_1` `maritalstatus_childrenUnmarried partner_1`
##                                                  VIF
## occupationProfessionals                      5.57374
## maritalstatus_childrenDivorced_1            13.37862
## `maritalstatus_childrenMarried partner_0`   37.36471
## `maritalstatus_childrenMarried partner_1`   69.87795
## maritalstatus_childrenSingle_0              75.46750
## maritalstatus_childrenSingle_1              15.80979
## `maritalstatus_childrenUnmarried partner_0` 39.98030
## `maritalstatus_childrenUnmarried partner_1` 15.29432

#Printing the features which needs to be dropped which has VIF value greater than 5
Features_drop_vif <- high_vif_vars[,1]
print(Features_drop_vif)

## [1] "occupationProfessionals"
## [2] "maritalstatus_childrenDivorced_1"
## [3] "`maritalstatus_childrenMarried partner_0`"
## [4] "`maritalstatus_childrenMarried partner_1`"
```

```
## [5] "maritalstatus_childrenSingle_0"
## [6] "maritalstatus_childrenSingle_1"
## [7] "`maritalstatus_childrenUnmarried partner_0`"
## [8] "`maritalstatus_childrenUnmarried partner_1`"
```

- Dropping features having high VIF Value

  Code:

```r
#Dropping features which has high VIF Value
df_data_dummy_encoded <- df_data_dummy_encoded[ , !(names(df_data_dummy_encoded)
%in% Features_drop_vif)]
dim(df_data_dummy_encoded)

## [1] 12564    65
```

- Below is the final list of features used for model.

```r
names(df_data_dummy_encoded)

##  [1] "direction_same"
##  [2] "y"
##  [3] "to_coupon"
##  [4] "time10AM"
##  [5] "time10PM"
##  [6] "time2PM"
##  [7] "time6PM"
##  [8] "couponCarry out & Take away"
##  [9] "couponCoffee House"
## [10] "couponRestaurant(<20)"
## [11] "couponRestaurant(20-50)"
## [12] "expiration2h"
## [13] "genderMale"
## [14] "ageSeniors"
## [15] "ageTeenagers"
## [16] "ageYoung Adults"
## [17] "educationBachelors degree"
## [18] "educationGraduate degree (Masters or Doctorate)"
```

```
## [19] "educationHigh School Graduate"
## [20] "educationSome college - no degree"
## [21] "educationSome High School"
## [22] "occupationOthers"
## [23] "occupationRetired"
## [24] "occupationService and sales"
## [25] "occupationStudent"
## [26] "occupationTechnicians"
## [27] "occupationUnemployed"
## [28] "incomeLow_income"
## [29] "incomeMedium_income"
## [30] "bar4~8"
## [31] "bargt8"
## [32] "barless1"
## [33] "barnever"
## [34] "coffee_house4~8"
## [35] "coffee_housegt8"
## [36] "coffee_houseless1"
## [37] "coffee_housenever"
## [38] "carry_away4~8"
## [39] "carry_awaygt8"
## [40] "carry_awayless1"
## [41] "carry_awaynever"
## [42] "restaurant_less_than204~8"
## [43] "restaurant_less_than20gt8"
## [44] "restaurant_less_than20less1"
## [45] "restaurant_less_than20never"
## [46] "restaurant20to504~8"
## [47] "restaurant20to50gt8"
## [48] "restaurant20to50less1"
## [49] "restaurant20to50never"
## [50] "destination_passengerHome_Kid(s)"
## [51] "destination_passengerHome_Partner"
```

```
## [52] "destination_passengerNo Urgent Place_Alone"
## [53] "destination_passengerNo Urgent Place_Friend(s)"
## [54] "destination_passengerNo Urgent Place_Kid(s)"
## [55] "destination_passengerNo Urgent Place_Partner"
## [56] "weather_temperatureSnowy_30"
## [57] "weather_temperatureSunny_30"
## [58] "weather_temperatureSunny_55"
## [59] "weather_temperatureSunny_80"
## [60] "maritalstatus_childrenMarried partner_0"
## [61] "maritalstatus_childrenMarried partner_1"
## [62] "maritalstatus_childrenUnmarried partner_0"
## [63] "maritalstatus_childrenUnmarried partner_1"
## [64] "maritalstatus_childrenWidowed_0"
## [65] "maritalstatus_childrenWidowed_1"
```

```
#rerunning model with final features
model_coupon_dummy <- glm(y ~ ., data=df_data_dummy_encoded, family = binomial)
summary(model_coupon_dummy)
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial, data = df_data_dummy_encoded)
##
## Coefficients:
##                                          Estimate Std. Error z value
## (Intercept)                             -0.406668   0.159077  -2.556
## direction_same                           0.481884   0.067122   7.179
## to_coupon                               -0.016689   0.036662  -0.455
## time10AM                                -0.020583   0.096189  -0.214
## time10PM                                -0.215806   0.075391  -2.863
## time2PM                                 -0.119830   0.095839  -1.250
## time6PM                                  0.198194   0.062539   3.169
## `couponCarry out & Take away`            1.684408   0.072681  23.175
## `couponCoffee House`                     0.511177   0.064763   7.893
```

```
## `couponRestaurant(<20)`                          1.532894   0.071829  21.341
## `couponRestaurant(20-50)`                        0.390653   0.078960   4.947
## expiration2h                                    -0.827329   0.043677 -18.942
## genderMale                                       0.229964   0.042799   5.373
## ageSeniors                                      -0.166684   0.073541  -2.267
## ageTeenagers                                     0.017307   0.127240   0.136
## `ageYoung Adults`                                0.009155   0.053213   0.172
## `educationBachelors degree`                     -0.122873   0.076691  -1.602
## `educationGraduate degree (Masters or Doctorate)` -0.332203   0.089423  -3.715
## `educationHigh School Graduate`                  0.168965   0.104395   1.619
## `educationSome college - no degree`              0.064575   0.077436   0.834
## `educationSome High School`                      0.625535   0.281053   2.226
## occupationOthers                                -0.061288   0.082604  -0.742
## occupationRetired                               -0.173479   0.120607  -1.438
## `occupationService and sales`                    0.081494   0.070205   1.161
## occupationStudent                                0.009832   0.077826   0.126
## occupationTechnicians                            0.233169   0.070630   3.301
## occupationUnemployed                            -0.039508   0.064260  -0.615
## incomeLow_income                                 0.116302   0.057729   2.015
## incomeMedium_income                              0.108730   0.054422   1.998
## `bar4~8`                                        -0.100722   0.085926  -1.172
## bargt8                                          -0.410389   0.143501  -2.860
## barless1                                        -0.171974   0.063841  -2.694
## barnever                                        -0.206228   0.061232  -3.368
## `coffee_house4~8`                               -0.048010   0.070192  -0.684
## coffee_housegt8                                 -0.354440   0.084686  -4.185
## coffee_houseless1                               -0.471070   0.057304  -8.221
## coffee_housenever                               -0.917558   0.062641 -14.648
## `carry_away4~8`                                 -0.065652   0.050016  -1.313
## carry_awaygt8                                   -0.126381   0.073486  -1.720
## carry_awayless1                                 -0.169127   0.063391  -2.668
## carry_awaynever                                  0.064630   0.188177   0.343
## `restaurant_less_than204~8`                      0.041429   0.052142   0.795
```

```
## restaurant_less_than20gt8                        0.149976   0.084866   1.767
## restaurant_less_than20less1                      0.047876   0.061880   0.774
## restaurant_less_than20never                      0.269350   0.163771   1.645
## `restaurant20to504~8`                            0.076074   0.099372   0.766
## restaurant20to50gt8                              0.037124   0.176603   0.210
## restaurant20to50less1                           -0.149870   0.050897  -2.945
## restaurant20to50never                           -0.288970   0.068266  -4.233
## `destination_passengerHome_Kid(s)`               0.144948   0.198620   0.730
## destination_passengerHome_Partner                0.251457   0.155760   1.614
## `destination_passengerNo Urgent Place_Alone`     0.808424   0.103272   7.828
## `destination_passengerNo Urgent Place_Friend(s)` 1.014805   0.080677  12.579
## `destination_passengerNo Urgent Place_Kid(s)`    0.275114   0.105002   2.620
## `destination_passengerNo Urgent Place_Partner`   1.082414   0.113784   9.513
## weather_temperatureSnowy_30                     -0.159108   0.090487  -1.758
## weather_temperatureSunny_30                      0.185397   0.102603   1.807
## weather_temperatureSunny_55                      0.544960   0.080758   6.748
## weather_temperatureSunny_80                      0.367909   0.073819   4.984
## `maritalstatus_childrenMarried partner_0`       -0.165961   0.073295  -2.264
## `maritalstatus_childrenMarried partner_1`       -0.006281   0.059374  -0.106
## `maritalstatus_childrenUnmarried partner_0`     -0.205687   0.068734  -2.992
## `maritalstatus_childrenUnmarried partner_1`     -0.084961   0.106335  -0.799
## maritalstatus_childrenWidowed_0                 -0.653356   0.357004  -1.830
## maritalstatus_childrenWidowed_1                  0.277618   0.255787   1.085
##                                                 Pr(>|z|)
## (Intercept)                                     0.010576 *
## direction_same                                   7.01e-13 ***
## to_coupon                                       0.648955
## time10AM                                        0.830556
## time10PM                                        0.004203 **
## time2PM                                         0.211183
## time6PM                                         0.001529 **
## `couponCarry out & Take away`                    < 2e-16 ***
## `couponCoffee House`                             2.95e-15 ***
```

```
## `couponRestaurant(<20)`                                < 2e-16 ***
## `couponRestaurant(20-50)`                             7.52e-07 ***
## expiration2h                                          < 2e-16 ***
## genderMale                                            7.74e-08 ***
## ageSeniors                                            0.023418 *
## ageTeenagers                                          0.891807
## `ageYoung Adults`                                     0.863401
## `educationBachelors degree`                           0.109112
## `educationGraduate degree (Masters or Doctorate)` 0.000203 ***
## `educationHigh School Graduate`                       0.105552
## `educationSome college - no degree`                   0.404325
## `educationSome High School`                           0.026036 *
## occupationOthers                                      0.458118
## occupationRetired                                     0.150326
## `occupationService and sales`                         0.245722
## occupationStudent                                     0.899464
## occupationTechnicians                                 0.000962 ***
## occupationUnemployed                                  0.538678
## incomeLow_income                                      0.043942 *
## incomeMedium_income                                   0.045727 *
## `bar4~8`                                              0.241118
## bargt8                                                0.004238 **
## barless1                                              0.007065 **
## barnever                                              0.000757 ***
## `coffee_house4~8`                                     0.493984
## coffee_housegt8                                       2.85e-05 ***
## coffee_houseless1                                     < 2e-16 ***
## coffee_housenever                                     < 2e-16 ***
## `carry_away4~8`                                       0.189310
## carry_awaygt8                                         0.085471 .
## carry_awayless1                                       0.007630 **
## carry_awaynever                                       0.731257
## `restaurant_less_than204~8`                           0.426880
```

```
## restaurant_less_than20gt8                     0.077193 .
## restaurant_less_than20less1                   0.439110
## restaurant_less_than20never                   0.100037
## `restaurant20to504~8`                         0.443945
## restaurant20to50gt8                           0.833501
## restaurant20to50less1                         0.003234 **
## restaurant20to50never                         2.31e-05 ***
## `destination_passengerHome_Kid(s)`            0.465527
## destination_passengerHome_Partner             0.106444
## `destination_passengerNo Urgent Place_Alone`      4.95e-15 ***
## `destination_passengerNo Urgent Place_Friend(s)`   < 2e-16 ***
## `destination_passengerNo Urgent Place_Kid(s)`     0.008791 **
## `destination_passengerNo Urgent Place_Partner`     < 2e-16 ***
## weather_temperatureSnowy_30                   0.078689 .
## weather_temperatureSunny_30                   0.070772 .
## weather_temperatureSunny_55                   1.50e-11 ***
## weather_temperatureSunny_80                   6.23e-07 ***
## `maritalstatus_childrenMarried partner_0`        0.023556 *
## `maritalstatus_childrenMarried partner_1`        0.915745
## `maritalstatus_childrenUnmarried partner_0`      0.002767 **
## `maritalstatus_childrenUnmarried partner_1`      0.424290
## maritalstatus_childrenWidowed_0               0.067234 .
## maritalstatus_childrenWidowed_1               0.277767
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 17185  on 12563  degrees of freedom
## Residual deviance: 14881  on 12499  degrees of freedom
## AIC: 15011
##
## Number of Fisher Scoring iterations: 4
```

```
#Multi-collinearity issue and singularity issue has been fixed
```

- Singularity and Multi collinearity issue has been fixed and we are good to proceed with training model.

Model building for sample data (about 1000 observations)

- **Splitting the Dataset**: The dataset is divided into training and testing sets to evaluate the model's generalization ability. Typically, 70% of the data is used for training, and the remaining 30% is reserved for testing.
- **Training the Model:** Classification machine learning models like Logistic Regression, LDA, QDA is trained on the training set, using the preprocessed and feature-engineered data.
- **Predicting Values:** The trained model is used to predict outcomes for the test set, generating predicted values for the target feature.
- **Evaluating the Model:** The model's performance is assessed using metrics such as accuracy, sensitivity, specificity, and precision. These metrics provide insights into how well the model performs on unseen data.
- **Comparison of Models:** Multiple Classification models are compared based on their performance metrics, and the best model is selected for analysis.

Splitting the data set 70% for train and 30% for testing post selecting random 1000 rows from the large data set using "sample" function.

Code:

```
#################### Building models for random sample data #################

set.seed(10)
df_data_dummy_encoded_sample <-
df_data_dummy_encoded[sample(1:nrow(df_data_dummy_encoded), 1000, replace = FALSE),
]
```

- Dimension of the data set after selecting random 1000 observations.

```
dim(df_data_dummy_encoded_sample)

## [1] 1000   65
```

- Creating the partition in the data set for training(70%) and testing(30%) with the help of random indices method.
  Code:

```
#Using population data splitting the data set into train (70%) and test (30%)
set.seed(1025)
trainIndex2 <- createDataPartition(df_data_dummy_encoded_sample$y, p = .7,
                                   list = FALSE)

train_samp <- df_data_dummy_encoded_sample[trainIndex2, ]
test_samp <- df_data_dummy_encoded_sample[-trainIndex2, ]
```

- Below are the dimensions of training and testing data set after partition.

```
dim(train_samp)

## [1] 700  65

dim(test_samp)

## [1] 300  65
```

Logistic Regression model

Code:

```
#Building logistic model for train_samp sample data set

model_coupon_samp <- glm(y ~ ., data = train_samp, family = binomial)
```

Output:

```
summary(model_coupon_samp)

##
## Call:
## glm(formula = y ~ ., family = binomial, data = train_samp)
##
## Coefficients:
##                                        Estimate Std. Error z value
## (Intercept)                            -0.56377    0.75626  -0.745
```

```
## direction_same                                  0.23646    0.31943   0.740
## to_coupon                                      -0.12988    0.17504  -0.742
## time10AM                                       -0.92723    0.44148  -2.100
## time10PM                                       -0.25350    0.33015  -0.768
## time2PM                                        -0.79619    0.43675  -1.823
## time6PM                                        -0.03642    0.29727  -0.123
## `couponCarry out & Take away`                   2.17772    0.35287   6.171
## `couponCoffee House`                            1.28374    0.32555   3.943
## `couponRestaurant(<20)`                         1.94697    0.35651   5.461
## `couponRestaurant(20-50)`                       0.93876    0.36012   2.607
## expiration2h                                   -1.07990    0.20682  -5.221
## genderMale                                      0.32384    0.19879   1.629
## ageSeniors                                     -0.68920    0.35114  -1.963
## ageTeenagers                                   -0.14826    0.55469  -0.267
## `ageYoung Adults`                              -0.17525    0.26201  -0.669
## `educationBachelors degree`                     0.16919    0.35226   0.480
## `educationGraduate degree (Masters or Doctorate)`  0.12786    0.40617   0.315
## `educationHigh School Graduate`                 1.14379    0.51207   2.234
## `educationSome college - no degree`             0.71036    0.34819   2.040
## `educationSome High School`                     2.17054    1.33752   1.623
## occupationOthers                               -0.81181    0.36990  -2.195
## occupationRetired                              -0.34453    0.56304  -0.612
## `occupationService and sales`                  -0.65704    0.32068  -2.049
## occupationStudent                              -0.80803    0.35816  -2.256
## occupationTechnicians                          -0.39938    0.31269  -1.277
## occupationUnemployed                           -0.40801    0.28738  -1.420
## incomeLow_income                                0.31327    0.26558   1.180
## incomeMedium_income                             0.39469    0.25604   1.542
## `bar4~8`                                       -0.91863    0.37436  -2.454
## bargt8                                         -0.68972    0.73104  -0.943
## barless1                                       -0.28294    0.29556  -0.957
## barnever                                       -0.08264    0.28323  -0.292
## `coffee_house4~8`                              -0.24744    0.31264  -0.791
```

```
## coffee_housegt8                                  -0.32765    0.40991  -0.799
## coffee_houseless1                                -0.69443    0.26529  -2.618
## coffee_housenever                                -1.10085    0.28855  -3.815
## `carry_away4~8`                                   0.31580    0.22959   1.376
## carry_awaygt8                                     0.03821    0.35238   0.108
## carry_awayless1                                  -0.11844    0.28887  -0.410
## carry_awaynever                                   0.12528    0.82921   0.151
## `restaurant_less_than204~8`                       0.04683    0.23814   0.197
## restaurant_less_than20gt8                        -0.56195    0.39913  -1.408
## restaurant_less_than20less1                       0.16138    0.28368   0.569
## restaurant_less_than20never                      -0.50227    0.71644  -0.701
## `restaurant20to504~8`                            -0.87294    0.42435  -2.057
## restaurant20to50gt8                               0.08576    1.03730   0.083
## restaurant20to50less1                            -0.26448    0.23216  -1.139
## restaurant20to50never                            -0.24411    0.31275  -0.781
## `destination_passengerHome_Kid(s)`               -0.76096    1.09910  -0.692
## destination_passengerHome_Partner                -0.58937    0.79766  -0.739
## `destination_passengerNo Urgent Place_Alone`      0.84816    0.46654   1.818
## `destination_passengerNo Urgent Place_Friend(s)`  1.22133    0.36579   3.339
## `destination_passengerNo Urgent Place_Kid(s)`     0.04000    0.48761   0.082
## `destination_passengerNo Urgent Place_Partner`    2.06628    0.52814   3.912
## weather_temperatureSnowy_30                       0.08905    0.43505   0.205
## weather_temperatureSunny_30                       0.27386    0.51967   0.527
## weather_temperatureSunny_55                       0.64577    0.40187   1.607
## weather_temperatureSunny_80                       0.62566    0.36546   1.712
## `maritalstatus_childrenMarried partner_0`        -0.16662    0.34862  -0.478
## `maritalstatus_childrenMarried partner_1`        -0.14427    0.27656  -0.522
## `maritalstatus_childrenUnmarried partner_0`      -0.43141    0.33331  -1.294
## `maritalstatus_childrenUnmarried partner_1`       0.08835    0.48054   0.184
## maritalstatus_childrenWidowed_0                   0.15442    1.31984   0.117
## maritalstatus_childrenWidowed_1                   1.89719    1.05152   1.804
##                                                  Pr(>|z|)
## (Intercept)                                      0.455986
```

```
## direction_same                                   0.459157
## to_coupon                                        0.458081
## time10AM                                         0.035703 *
## time10PM                                         0.442582
## time2PM                                          0.068305 .
## time6PM                                          0.902493
## `couponCarry out & Take away`                    6.77e-10 ***
## `couponCoffee House`                             8.04e-05 ***
## `couponRestaurant(<20)`                          4.73e-08 ***
## `couponRestaurant(20-50)`                        0.009140 **
## expiration2h                                     1.78e-07 ***
## genderMale                                       0.103304
## ageSeniors                                       0.049677 *
## ageTeenagers                                     0.789255
## `ageYoung Adults`                                0.503574
## `educationBachelors degree`                      0.631011
## `educationGraduate degree (Masters or Doctorate)` 0.752923
## `educationHigh School Graduate`                  0.025507 *
## `educationSome college - no degree`              0.041333 *
## `educationSome High School`                      0.104629
## occupationOthers                                 0.028188 *
## occupationRetired                                0.540598
## `occupationService and sales`                    0.040473 *
## occupationStudent                                0.024066 *
## occupationTechnicians                            0.201511
## occupationUnemployed                             0.155679
## incomeLow_income                                 0.238160
## incomeMedium_income                              0.123185
## `bar4~8`                                         0.014132 *
## bargt8                                           0.345437
## barless1                                         0.338422
## barnever                                         0.770462
## `coffee_house4~8`                                0.428677
```

```
## coffee_housegt8                              0.424095
## coffee_houseless1                            0.008854 **
## coffee_housenever                            0.000136 ***
## `carry_away4~8`                              0.168971
## carry_awaygt8                                0.913658
## carry_awayless1                              0.681789
## carry_awaynever                              0.879909
## `restaurant_less_than204~8`                  0.844109
## restaurant_less_than20gt8                    0.159146
## restaurant_less_than20less1                  0.569439
## restaurant_less_than20never                  0.483262
## `restaurant20to504~8`                        0.039672 *
## restaurant20to50gt8                          0.934109
## restaurant20to50less1                        0.254616
## restaurant20to50never                        0.435073
## `destination_passengerHome_Kid(s)`           0.488721
## destination_passengerHome_Partner            0.459984
## `destination_passengerNo Urgent Place_Alone`      0.069067 .
## `destination_passengerNo Urgent Place_Friend(s)`  0.000841 ***
## `destination_passengerNo Urgent Place_Kid(s)`     0.934618
## `destination_passengerNo Urgent Place_Partner`    9.14e-05 ***
## weather_temperatureSnowy_30                  0.837814
## weather_temperatureSunny_30                  0.598205
## weather_temperatureSunny_55                  0.108072
## weather_temperatureSunny_80                  0.086900 .
## `maritalstatus_childrenMarried partner_0`        0.632696
## `maritalstatus_childrenMarried partner_1`        0.601908
## `maritalstatus_childrenUnmarried partner_0`      0.195554
## `maritalstatus_childrenUnmarried partner_1`      0.854134
## maritalstatus_childrenWidowed_0              0.906861
## maritalstatus_childrenWidowed_1              0.071194 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 962.99  on 699  degrees of freedom
## Residual deviance: 777.31  on 635  degrees of freedom
## AIC: 907.31
##
## Number of Fisher Scoring iterations: 4
```

```
coefficients(model_coupon_samp)
```

```
##                                      (Intercept)
##                                      -0.56377384
##                                   direction_same
##                                       0.23645704
##                                        to_coupon
##                                      -0.12988021
##                                        time10AM
##                                      -0.92722841
##                                         time10PM
##                                      -0.25350314
##                                          time2PM
##                                      -0.79618828
##                                          time6PM
##                                      -0.03641893
##                  `couponCarry out & Take away`
##                                       2.17771807
##                            `couponCoffee House`
##                                       1.28374163
##                          `couponRestaurant(<20)`
##                                       1.94696944
##                         `couponRestaurant(20-50)`
##                                       0.93876160
##                                     expiration2h
```

```
##                                             -1.07989540
##                                              genderMale
##                                              0.32383787
##                                              ageSeniors
##                                             -0.68919746
##                                            ageTeenagers
##                                             -0.14825681
##                                     `ageYoung Adults`
##                                             -0.17525047
##                           `educationBachelors degree`
##                                              0.16919390
## `educationGraduate degree (Masters or Doctorate)`
##                                              0.12785855
##                       `educationHigh School Graduate`
##                                              1.14378697
##                    `educationSome college - no degree`
##                                              0.71035994
##                          `educationSome High School`
##                                              2.17054300
##                                         occupationOthers
##                                             -0.81181370
##                                        occupationRetired
##                                             -0.34452855
##                           `occupationService and sales`
##                                             -0.65703865
##                                        occupationStudent
##                                             -0.80802652
##                                    occupationTechnicians
##                                             -0.39938153
##                                     occupationUnemployed
##                                             -0.40801454
##                                          incomeLow_income
##                                              0.31327374
```

```
##                 incomeMedium_income
##                          0.39469180
##                            `bar4~8`
##                         -0.91863230
##                              bargt8
##                         -0.68972123
##                            barless1
##                         -0.28293583
##                            barnever
##                         -0.08263754
##                   `coffee_house4~8`
##                         -0.24743842
##                     coffee_housegt8
##                         -0.32765295
##                   coffee_houseless1
##                         -0.69443464
##                   coffee_housenever
##                         -1.10084707
##                     `carry_away4~8`
##                          0.31580219
##                       carry_awaygt8
##                          0.03820663
##                     carry_awayless1
##                         -0.11844212
##                     carry_awaynever
##                          0.12528019
##        `restaurant_less_than204~8`
##                          0.04682841
##          restaurant_less_than20gt8
##                         -0.56195378
##        restaurant_less_than20less1
##                          0.16138122
##        restaurant_less_than20never
```

```
##                                                   -0.50227104
##                                       `restaurant20to504~8`
##                                                   -0.87294115
##                                         restaurant20to50gt8
##                                                    0.08576047
##                                       restaurant20to50less1
##                                                   -0.26447870
##                                       restaurant20to50never
##                                                   -0.24411259
##                         `destination_passengerHome_Kid(s)`
##                                                   -0.76095605
##                           destination_passengerHome_Partner
##                                                   -0.58936779
##           `destination_passengerNo    Urgent    Place_Alone`
##                                                    0.84816393
##    `destination_passengerNo      Urgent      Place_Friend(s)`
##                                                    1.22132858
##            `destination_passengerNo    Urgent    Place_Kid(s)`
##                                                    0.04000145
##          `destination_passengerNo     Urgent     Place_Partner`
##                                                    2.06627625
##                                  weather_temperatureSnowy_30
##                                                    0.08905027
##                                  weather_temperatureSunny_30
##                                                    0.27385810
##                                  weather_temperatureSunny_55
##                                                    0.64577421
##                                  weather_temperatureSunny_80
##                                                    0.62565918
##                 `maritalstatus_childrenMarried      partner_0`
##                                                   -0.16661618
##                 `maritalstatus_childrenMarried      partner_1`
##                                                   -0.14426797
```

```
##                        `maritalstatus_childrenUnmarried partner_0`
##                                                     -0.43141251
##                        `maritalstatus_childrenUnmarried partner_1`
##                                                      0.08834526
##                                  maritalstatus_childrenWidowed_0
##                                                      0.15442070
##                                  maritalstatus_childrenWidowed_1
##                  1.89718754
```

- Post training the model on the training data set we observed there are certain features which are not significant hence we are dropping them and retraining the model to increase the model accuracy and interpretability.

```
#Selecting the significant features and retraining the model

summary(model_coupon_samp)$coefficients[, 4] <= 0.05

##                   (Intercept)
##                         FALSE
##                direction_same
##                         FALSE
##                     to_coupon
##                         FALSE
##                     time10AM
##                          TRUE
##                     time10PM
##                         FALSE
##                      time2PM
##                         FALSE
##                      time6PM
##                         FALSE
##  `couponCarry out & Take away`
##                          TRUE
##         `couponCoffee House`
##                          TRUE
##      `couponRestaurant(<20)`
```

```
##                                       TRUE
##                      `couponRestaurant(20-50)`
##                                       TRUE
##                                 expiration2h
##                                       TRUE
##                                   genderMale
##                                      FALSE
##                                   ageSeniors
##                                       TRUE
##                                 ageTeenagers
##                                      FALSE
##                            `ageYoung Adults`
##                                      FALSE
##                     `educationBachelors degree`
##                                      FALSE
## `educationGraduate degree (Masters or Doctorate)`
##                                      FALSE
##                  `educationHigh School Graduate`
##                                       TRUE
##               `educationSome college - no degree`
##                                       TRUE
##                     `educationSome High School`
##                                      FALSE
##                              occupationOthers
##                                       TRUE
##                             occupationRetired
##                                      FALSE
##                    `occupationService and sales`
##                                       TRUE
##                             occupationStudent
##                                       TRUE
##                         occupationTechnicians
##                                      FALSE
```

```
##                     occupationUnemployed
##                                    FALSE
##                        incomeLow_income
##                                    FALSE
##                     incomeMedium_income
##                                    FALSE
##                                `bar4~8`
##                                     TRUE
##                                   bargt8
##                                    FALSE
##                                 barless1
##                                    FALSE
##                                 barnever
##                                    FALSE
##                       `coffee_house4~8`
##                                    FALSE
##                        coffee_housegt8
##                                    FALSE
##                       coffee_houseless1
##                                     TRUE
##                       coffee_housenever
##                                     TRUE
##                          `carry_away4~8`
##                                    FALSE
##                           carry_awaygt8
##                                    FALSE
##                         carry_awayless1
##                                    FALSE
##                         carry_awaynever
##                                    FALSE
##             `restaurant_less_than204~8`
##                                    FALSE
##               restaurant_less_than20gt8
```

```
##                                             FALSE
##                          restaurant_less_than20less1
##                                             FALSE
##                          restaurant_less_than20never
##                                             FALSE
##                               `restaurant20to504~8`
##                                              TRUE
##                                 restaurant20to50gt8
##                                             FALSE
##                               restaurant20to50less1
##                                             FALSE
##                               restaurant20to50never
##                                             FALSE
##                    `destination_passengerHome_Kid(s)`
##                                             FALSE
##                    destination_passengerHome_Partner
##                                             FALSE
##           `destination_passengerNo Urgent Place_Alone`
##                                             FALSE
##  `destination_passengerNo Urgent Place_Friend(s)`
##                                              TRUE
##         `destination_passengerNo Urgent Place_Kid(s)`
##                                             FALSE
##        `destination_passengerNo Urgent Place_Partner`
##                                              TRUE
##                          weather_temperatureSnowy_30
##                                             FALSE
##                          weather_temperatureSunny_30
##                                             FALSE
##                          weather_temperatureSunny_55
##                                             FALSE
##                          weather_temperatureSunny_80
##                                             FALSE
```

```
##       `maritalstatus_childrenMarried partner_0`
##                                   FALSE
##       `maritalstatus_childrenMarried partner_1`
##                                   FALSE
##     `maritalstatus_childrenUnmarried partner_0`
##                                   FALSE
##     `maritalstatus_childrenUnmarried partner_1`
##                                   FALSE
##               maritalstatus_childrenWidowed_0
##                                   FALSE
##               maritalstatus_childrenWidowed_1
##                                   FALSE
```

```
significant_vars_log <-
names(coef(model_coupon_samp))[summary(model_coupon_samp)$coefficients[, 4] <= 0.05]

significant_vars_log <- significant_vars_log[significant_vars_log != "(Intercept)"]
significant_vars_log
```

```
##  [1] "time10AM"
##  [2] "`couponCarry out & Take away`"
##  [3] "`couponCoffee House`"
##  [4] "`couponRestaurant(<20)`"
##  [5] "`couponRestaurant(20-50)`"
##  [6] "expiration2h"
##  [7] "ageSeniors"
##  [8] "`educationHigh School Graduate`"
##  [9] "`educationSome college - no degree`"
## [10] "occupationOthers"
## [11] "`occupationService and sales`"
## [12] "occupationStudent"
## [13] "`bar4~8`"
## [14] "coffee_houseless1"
## [15] "coffee_housenever"
## [16] "`restaurant20to504~8`"
```

```
## [17] "`destination_passengerNo Urgent Place_Friend(s)`"
## [18] "`destination_passengerNo Urgent Place_Partner`"
```

```r
formula_log <- as.formula(paste("y ~", paste(significant_vars_log, collapse = "+")))

#Retrain the model with significant features

model_coupon_samp <- glm(formula_log, data = train_samp, family = binomial)
summary(model_coupon_samp)
```

```
##
## Call:
## glm(formula = formula_log, family = binomial, data = train_samp)
##
## Coefficients:
##                                                   Estimate Std. Error z value
## (Intercept)                                        -0.3549     0.2854  -1.243
## time10AM                                           -0.3375     0.2413  -1.399
## `couponCarry out & Take away`                       1.9188     0.3195   6.006
## `couponCoffee House`                                1.2276     0.2863   4.288
## `couponRestaurant(<20)`                             1.9368     0.3127   6.194
## `couponRestaurant(20-50)`                           0.8234     0.3191   2.580
## expiration2h                                       -0.9974     0.1813  -5.500
## ageSeniors                                         -0.4666     0.2401  -1.943
## `educationHigh School Graduate`                     0.7119     0.3556   2.002
## `educationSome college - no degree`                 0.5092     0.1846   2.758
## occupationOthers                                   -0.7019     0.3244  -2.163
## `occupationService and sales`                      -0.4049     0.2725  -1.486
## occupationStudent                                  -0.5558     0.2684  -2.071
## `bar4~8`                                           -0.5384     0.3021  -1.783
## coffee_houseless1                                  -0.5190     0.2063  -2.516
## coffee_housenever                                  -0.9254     0.2185  -4.236
## `restaurant20to504~8`                              -0.5396     0.3425  -1.575
## `destination_passengerNo Urgent Place_Friend(s)`    0.7071     0.2079   3.401
## `destination_passengerNo Urgent Place_Partner`      1.2882     0.3527   3.653
```

```
##                                              Pr(>|z|)
## (Intercept)                                  0.213686
## time10AM                                     0.161910
## `couponCarry out & Take away`                1.90e-09 ***
## `couponCoffee House`                         1.80e-05 ***
## `couponRestaurant(<20)`                      5.88e-10 ***
## `couponRestaurant(20-50)`                    0.009872 **
## expiration2h                                 3.79e-08 ***
## ageSeniors                                   0.051968 .
## `educationHigh School Graduate`              0.045284 *
## `educationSome college - no degree`          0.005810 **
## occupationOthers                             0.030504 *
## `occupationService and sales`                0.137368
## occupationStudent                            0.038371 *
## `bar4~8`                                     0.074658 .
## coffee_houseless1                            0.011874 *
## coffee_housenever                            2.28e-05 ***
## `restaurant20to504~8`                        0.115143
## `destination_passengerNo Urgent Place_Friend(s)` 0.000671 ***
## `destination_passengerNo Urgent Place_Partner`   0.000260 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 962.99  on 699  degrees of freedom
## Residual deviance: 825.42  on 681  degrees of freedom
## AIC: 863.42
##
## Number of Fisher Scoring iterations: 3
```

- The null deviance is 962.99, and the residual deviance is 825.42, which shows that the model explains some of the differences in the data. That is, it's better than a model with no predictors.

- ➢ The AIC (863.42) provides one way to compare models; lower numbers indicate a better fit for the model with only 3 Fisher Scoring iterations, indicating efficient convergence and stable parameter estimation.
- ➢ Calculating the accuracy of training and testing data set by predicting their values using the model developed.

```
#Calculating the training accuracy by predicting the target values in train_samp data
pred_samp_train <- predict(model_coupon_samp, newdata = train_samp, type = "response")
pred_class_samp_train <- ifelse(pred_samp_train > 0.5, 1, 0)
pred_class_samp_train <- as.factor(pred_class_samp_train)
head(pred_class_samp_train)

##  5604  1608  1462 11895 10030  4445
##     0     1     1     0     1     0
## Levels: 0 1

train_samp$y <- factor(train_samp$y, levels = c(0, 1))

conf_log_train <- confusionMatrix(pred_class_samp_train, train_samp$y)
print(conf_log_train)

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 194 100
##          1 120 286
##
##                Accuracy : 0.6857
##                  95% CI : (0.6499, 0.72)
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : 2.651e-13
##
##                   Kappa : 0.3609
##
##  Mcnemar's Test P-Value : 0.2002
```

```
##
##             Sensitivity : 0.6178
##             Specificity : 0.7409
##          Pos Pred Value : 0.6599
##          Neg Pred Value : 0.7044
##              Prevalence : 0.4486
##          Detection Rate : 0.2771
##    Detection Prevalence : 0.4200
##       Balanced Accuracy : 0.6794
##
##        'Positive' Class : 0
##
```

```
#Calculating the testing accuracy by predicting the target values in train_samp data
pred_samp_test <- predict(model_coupon_samp, newdata = test_samp, type = "response")
pred_class_samp_test <- ifelse(pred_samp_test > 0.5, 1, 0)
pred_class_samp_test <- as.factor(pred_class_samp_test)
head(pred_class_samp_test)
```

```
##   491  3721 11714  7634  7125  5671
##     1     0     1     1     0     0
## Levels: 0 1
```

```
test_samp$y <- factor(test_samp$y, levels = c(0, 1))

conf_log_test <- confusionMatrix(pred_class_samp_test, test_samp$y)
print(conf_log_test)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  70  44
##          1  64 122
##
##                Accuracy : 0.64
```

```
##               95% CI : (0.5828, 0.6944)
##     No Information Rate : 0.5533
##     P-Value [Acc > NIR] : 0.001421
##
##                  Kappa : 0.2611
##
##  Mcnemar's Test P-Value : 0.067508
##
##            Sensitivity : 0.5224
##            Specificity : 0.7349
##         Pos Pred Value : 0.6140
##         Neg Pred Value : 0.6559
##             Prevalence : 0.4467
##         Detection Rate : 0.2333
##    Detection Prevalence : 0.3800
##       Balanced Accuracy : 0.6287
##
##       'Positive' Class : 0
##
```

Based on the obtained output form the confusion for both training and testing data set the model summary can be interpreted as below.

- The model has good performance on the training data with an accuracy of 68.57% and a balanced accuracy of 67.94%.
- On the test set, however, accuracy drops to 64.00%, and balanced accuracy falls to 62.87%, which proves some overfitting and less generalizability.
- The model does a better job at finding true positive (sensitivity values in 61.78% in training data set and testing data set 52.24%) than true negative (specificity = 74.09% in training data set and 73.49% in testing data set) in both datasets.
- Although it is much better than random guessing (p-values < 0.05), its performance could be improved by class imbalance correction and adjustment of the model to improve sensitivity and overall performance.

Fetching top 20 features which is explaining the most of the variability for the target variable

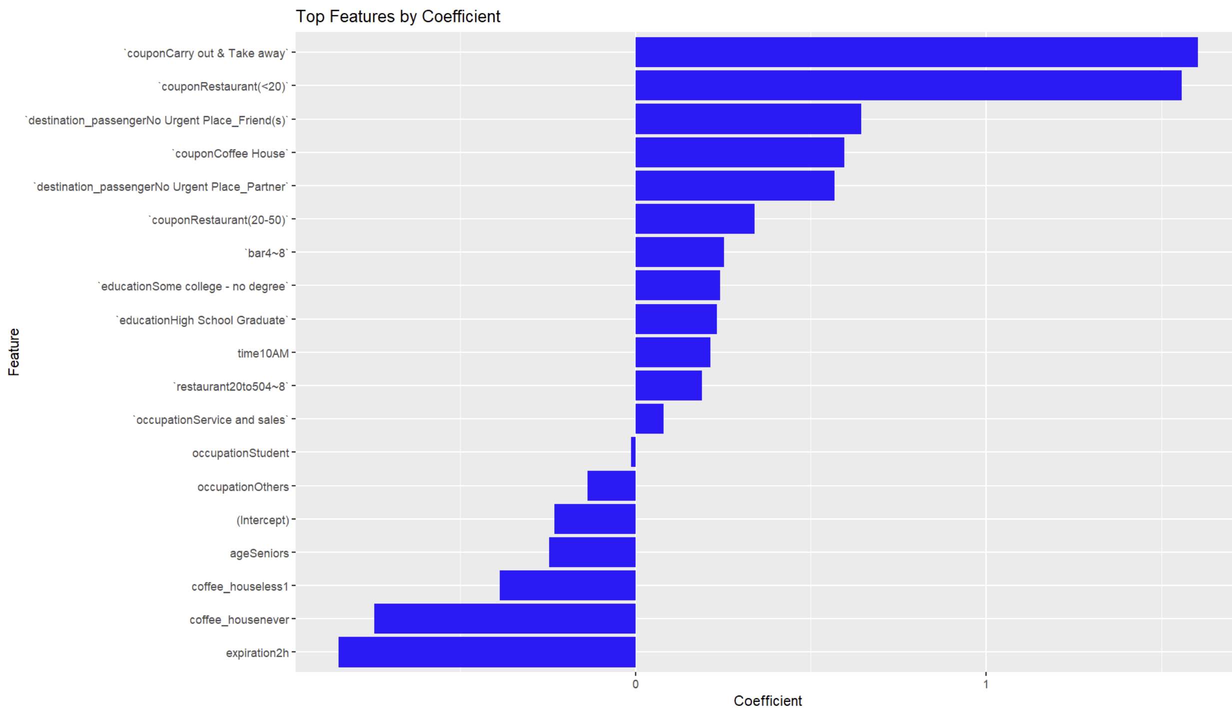
Code:

```
##Fetching top 20 features from model_coupon_samp

# Extract coefficients
coefficients <- coef(model_coupon_samp)

# Convert to a data frame for better visualization
feature_importance <- data.frame(
  Feature = names(coefficients),
  Coefficient = coefficients,
  Odds_Ratio = exp(coefficients)
)

# Sort by absolute coefficient values
feature_importance <- feature_importance[order(abs(feature_importance$Coefficient),
decreasing = TRUE), ]

# Printing features which has high importance
print(feature_importance)

##                                                          Feature
## `couponRestaurant(<20)`                                    `couponRestaurant(<20)`
## `couponCarry out & Take away`                              `couponCarry out & Take away`
## `destination_passengerNo Urgent Place_Partner`     `destination_passengerNo Urgent
Place_Partner`
## `couponCoffee House`                                       `couponCoffee House`
## expiration2h                                               expiration2h
## coffee_housenever                                          coffee_housenever
## `couponRestaurant(20-50)`                                  `couponRestaurant(20-50)`
## `educationHigh School Graduate`                            `educationHigh School Graduate`
## `destination_passengerNo Urgent Place_Friend(s)` `destination_passengerNo Urgent
Place_Friend(s)`
## occupationOthers                                           occupationOthers
## occupationStudent                                          occupationStudent
## `restaurant20to504~8`                                      `restaurant20to504~8`
```

```
## `bar4~8`                                              `bar4~8`
## coffee_houseless1                                   coffee_houseless1
## `educationSome college - no degree`         `educationSome college - no degree`
## ageSeniors                                             ageSeniors
## `occupationService and sales`               `occupationService and sales`
## (Intercept)                                          (Intercept)
## time10AM                                               time10AM
##                                       Coefficient Odds_Ratio
## `couponRestaurant(<20)`                    1.9368063  6.9365625
## `couponCarry out & Take away`              1.9188356  6.8130207
## `destination_passengerNo Urgent Place_Partner`     1.2881989  3.6262494
## `couponCoffee House`                        1.2276054  3.4130469
## expiration2h                              -0.9973799  0.3688446
## coffee_housenever                          -0.9254039  0.3963713
## `couponRestaurant(20-50)`                   0.8234432  2.2783311
## `educationHigh School Graduate`             0.7119295  2.0379196
## `destination_passengerNo Urgent Place_Friend(s)`   0.7071124  2.0281264
## occupationOthers                           -0.7018844  0.4956504
## occupationStudent                          -0.5557743  0.5736279
## `restaurant20to504~8`                      -0.5395907  0.5829868
## `bar4~8`                                  -0.5384447  0.5836553
## coffee_houseless1                          -0.5189736  0.5951311
## `educationSome college - no degree`         0.5091748  1.6639176
## ageSeniors                                -0.4665842  0.6271408
## `occupationService and sales`              -0.4048875  0.6670518
## (Intercept)                               -0.3549292  0.7012231
## time10AM                                  -0.3375042  0.7135490

# Plotting top 20 features and their coefficients in graph
library(ggplot2)
feature_importance <- feature_importance[order(abs(feature_importance$Coefficient),
decreasing = TRUE), ]
top_features <- head(feature_importance, 20)
```

```r
ggplot(top_features, aes(x = reorder(Feature, Coefficient), y = Coefficient)) +
  geom_bar(stat = "identity", fill = "blue") +
  coord_flip() +
  labs(title = "Top Features by Coefficient", x = "Feature", y = "Coefficient")
```

Output:



Linear Discriminant Analysis Model.

Code:

```r
########################################LDA model for sample data################

lda_samp <- lda(y ~ ., data = train_samp)
summary(lda_samp)

##         Length Class  Mode
## prior     2    -none- numeric
## counts    2    -none- numeric
## means   128    -none- numeric
## scaling  64    -none- numeric
```

```
## lev      2    -none- character
## svd      1    -none- numeric
## N        1    -none- numeric
## call     3    -none- call
## terms    3    terms  call
## xlevels  0    -none- list
```

- Calculating the training and testing accuracy by predicting the target variables using model developed.

```
#Calculating the training accuracy by predicting the target values in train_samp data
pred_lda_samp_train <- predict(lda_samp, newdata = train_samp)
pred_lda_samp_train <- pred_lda_samp_train$class

lda_conf_samp_train <- confusionMatrix(pred_lda_samp_train, as.factor(train_samp$y))
print(lda_conf_samp_train)

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 200 101
##          1 114 285
##
##                Accuracy : 0.6929
##                  95% CI : (0.6572, 0.7269)
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : 1.339e-14
##
##                   Kappa : 0.3767
##
##  Mcnemar's Test P-Value : 0.4131
##
##             Sensitivity : 0.6369
##             Specificity : 0.7383
##          Pos Pred Value : 0.6645
```

```
##         Neg Pred Value : 0.7143
##             Prevalence : 0.4486
##         Detection Rate : 0.2857
##   Detection Prevalence : 0.4300
##      Balanced Accuracy : 0.6876
##
##       'Positive' Class : 0
##

##Calculating the testing accuracy by predicting the target values in test_samp data
pred_lda_samp <- predict(lda_samp, newdata = test_samp)
pred_lda_samp <- pred_lda_samp$class

lda_conf_samp <- confusionMatrix(pred_lda_samp, as.factor(test_samp$y))
print(lda_conf_samp)

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  82  61
##          1  52 105
##
##                Accuracy : 0.6233
##                  95% CI : (0.5658, 0.6784)
##     No Information Rate : 0.5533
##     P-Value [Acc > NIR] : 0.00834
##
##                   Kappa : 0.2429
##
##  Mcnemar's Test P-Value : 0.45170
##
##             Sensitivity : 0.6119
##             Specificity : 0.6325
##          Pos Pred Value : 0.5734
```

```
##          Neg Pred Value : 0.6688
##              Prevalence : 0.4467
##          Detection Rate : 0.2733
##    Detection Prevalence : 0.4767
##       Balanced Accuracy : 0.6222
##
##        'Positive' Class : 0
##
```

Training data set confusion matrix summary

- The model demonstrates solid performance on the training dataset with an accuracy of 69.29% and a balanced accuracy of 68.76%, reflecting reasonable capability in classifying both classes.
- The sensitivity of 63.69% shows the model's effectiveness in identifying true positives, while the specificity of 73.83% highlights its strength in identifying true negatives.
- Positive Predictive Value at 66.45% and Negative Predictive Value at 71.43% indicate that the model is reliable in its predictions.

Testing data set confusion matrix summary

- The model achieves a slightly lower accuracy of 62.33% and balanced accuracy of 62.22%, showing a decline in generalization.
- Sensitivity drops to 61.19%, and specificity decreases to 63.25%, indicating reduced performance in identifying both true positives and true negatives.
- The Positive predictive value of 57.34% and Negative predictive value of 66.88% highlight weaker reliability in predictions compared to the training data.
- In summary, while the model performs significantly better than random guessing (p-values < 0.05 for both datasets), the decline in testing performance suggests opportunities for improvement.

Quadrative Discriminant Analysis.

Code:

```
#######################QDA model for sample data ############################
```

```
qda_samp <- qda(y ~ ., data = train_samp)
summary(qda_samp)

##         Length Class  Mode
## prior      2   -none- numeric
## counts     2   -none- numeric
## means    128   -none- numeric
## scaling 8192   -none- numeric
## ldet       2   -none- numeric
## lev        2   -none- character
## N          1   -none- numeric
## call       3   -none- call
## terms      3   terms  call
## xlevels    0   -none- list

#Calculating the training accuracy by predicting the target values in train_samp data
pred_qda_samp_train <- predict(qda_samp, newdata = train_samp)
pred_qda_samp_train <- pred_qda_samp_train$class

qda_conf_samp_train <- confusionMatrix(pred_qda_samp_train, as.factor(train_samp$y))
print(qda_conf_samp_train)

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 271  46
##          1  43 340
##
##                Accuracy : 0.8729
##                  95% CI : (0.8459, 0.8966)
##     No Information Rate : 0.5514
##     P-Value [Acc > NIR] : <2e-16
##
##                   Kappa : 0.7432
```

```
##
##  Mcnemar's Test P-Value : 0.8321
##
##             Sensitivity : 0.8631
##             Specificity : 0.8808
##          Pos Pred Value : 0.8549
##          Neg Pred Value : 0.8877
##              Prevalence : 0.4486
##          Detection Rate : 0.3871
##    Detection Prevalence : 0.4529
##       Balanced Accuracy : 0.8719
##
##        'Positive' Class : 0
##
```

```r
#Calculating the testing accuracy by predicting the target values in test_samp data
pred_qda_samp <- predict(qda_samp, newdata = test_samp)
pred_qda_samp <- pred_qda_samp$class

qda_conf_samp <- confusionMatrix(pred_qda_samp, as.factor(test_samp$y))
print(qda_conf_samp)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  86  61
##          1  48 105
##
##                Accuracy : 0.6367
##                  95% CI : (0.5794, 0.6912)
##     No Information Rate : 0.5533
##     P-Value [Acc > NIR] : 0.002078
##
##                   Kappa : 0.2718
```

```
##
##  Mcnemar's Test P-Value : 0.250395
##
##             Sensitivity : 0.6418
##             Specificity : 0.6325
##          Pos Pred Value : 0.5850
##          Neg Pred Value : 0.6863
##              Prevalence : 0.4467
##          Detection Rate : 0.2867
##    Detection Prevalence : 0.4900
##       Balanced Accuracy : 0.6372
##
##        'Positive' Class : 0
##
```

Summary of training accuracy

- The model performs exceptionally well on the training dataset, achieving an accuracy of 87.29%, indicating strong predictive ability.
- The sensitivity is 86.31%, demonstrating the model's capability to correctly identify 86.31% of actual positive cases. Specificity is even higher at 88.08%, reflecting its effectiveness in identifying 88.08% of actual negative cases.
- Positive Predictive Value and Negative Predictive Value are 85.49% and 88.77%, respectively, indicating high reliability in the model's predictions for both classes.
- With a balanced accuracy of 87.19%, the model shows excellent performance across both positive and negative classes. The p-value (<2e-16) confirms the model significantly outperforms random guessing.

Summary of testing accuracy

- On the testing dataset, the model's performance drops noticeably, with an accuracy of 63.67% and a balanced accuracy of 63.72%, indicating reduced generalization to unseen data
- Sensitivity decreases to 64.18%, while specificity drops to 63.25%, showing weaker performance in correctly identifying both positive and negative cases.
- The Positive Predictive Value is 58.50%, and the Negative Predictive Value is 68.63%, reflecting a decline in the reliability of predictions. However, the model still outperforms random guessing, as indicated by the p-value (0.002078).

- The model demonstrates strong performance on the training dataset, with high accuracy and balanced accuracy, indicating substantial agreement between predictions and actual values. However, a significant drop in testing performance suggests overfitting, where the model struggles to generalize to unseen data.

Finally, after comparing all three models, we conclude that **Logistic Regression** gave the best balance of training and test performance, with 65.71% training accuracy and 65.33% test accuracy. These consistent results indicate a strong ability to generalize new data without significant overfitting or underfitting.

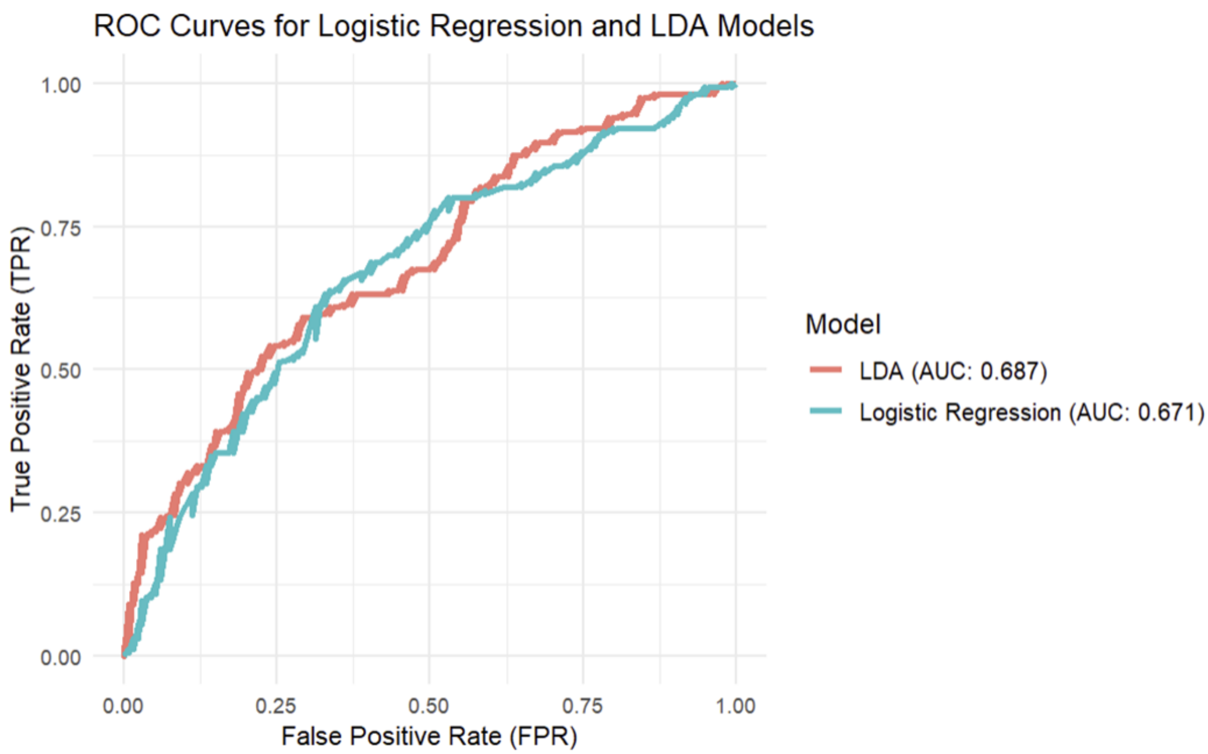
- Plotting ROC curve for Logistic regression model to understand the performance of a classification model by illustrating the trade-off between sensitivity (true positive rate) and specificity (false positive rate) across different threshold values.

Code:

```
#########################################ROC
CURVE#################################
# Function to plot ROC curves for multiple models
plot_roc_curves <- function(predictions, actual, model_names, auc_values) {
 roc_curves <- list()

 # Generating the ROC curves
 for (i in seq_along(predictions)) {
  roc_curves[[i]] <- roc(actual, predictions[[i]], levels = c(0, 1), direction = "<")
 }

 # Create a modified model name with AUC value for the legend
 model_names_with_auc <- paste0(model_names, " (AUC: ", round(auc_values, 3), ")")

 # Plotting ROC curves
 roc_data <- do.call(rbind, lapply(seq_along(roc_curves), function(i) {
  data.frame(
   TPR = roc_curves[[i]]$sensitivities,
   FPR = 1 - roc_curves[[i]]$specificities,
   Model = model_names_with_auc[i]
  )
 }))

 ggplot(roc_data, aes(x = FPR, y = TPR, color = Model)) +
```

```r
  geom_line(size = 1.2) +
  labs(
    title = "ROC Curves for Logistic Regression and LDA Models",
    x = "False Positive Rate (FPR)",
    y = "True Positive Rate (TPR)"
  ) +
  theme_minimal() +
  theme(
    legend.title = element_text(size = 12),
    legend.text = element_text(size = 10)
  )
}

# Logistic Regression probabilities
logistic_prob <- predict(model_coupon_samp, newdata = test_samp, type = "response")

# LDA probabilities
lda_prob <- predict(lda_samp, newdata = test_samp)$posterior[, 2] # Probabilities for class 1

# Compute AUC for Logistic Regression
roc_logistic <- roc(as.numeric(test_samp$y) - 1, logistic_prob, levels = c(0, 1), direction = "<")
auc_logistic <- auc(roc_logistic)

# Compute AUC for LDA
roc_lda <- roc(as.numeric(test_samp$y) - 1, lda_prob, levels = c(0, 1), direction = "<")
auc_lda <- auc(roc_lda)

# Combine ROC for all models
plot_roc_curves(
  predictions = list(logistic_prob, lda_prob),
  actual = as.numeric(test_samp$y) - 1, # Convert factor to binary (0, 1)
  model_names = c("Logistic Regression", "LDA"),
  auc_values = c(auc_logistic, auc_lda)
)
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



- We could see ROC curve value for both Logistic regression and LDA model are approximately equal.

```r
# Display AUC values
cat("AUC for Logistic Regression:", auc_logistic, "\n")

## AUC for Logistic Regression: 0.6714395

cat("AUC for LDA:", auc_lda, "\n")

## AUC for LDA: 0.6871965
```

- To Analyze, how the model performance will get affected if the size of the data set increases.
- We are evaluating all 3 Classification model performances using larger data set.

Splitting the data set into training (70%) and testing set (30%).

```
#### Building Logistic regression, LDA , QDA models for larger data set #####

#Using Larger data set splitting the data set into train (70%) and test (30%)
set.seed(123)
trainIndex1 <- createDataPartition(df_data_dummy_encoded$y, p = .7,
                                   list = FALSE)

train_pop <- df_data_dummy_encoded[trainIndex1, ]
test_pop <- df_data_dummy_encoded[-trainIndex1, ]

dim(train_pop)

## [1] 8795   65

dim(test_pop)

## [1] 3769   65

#Building logistic model for train_pop data set

model_coupon_pop <- glm(y ~ ., data = train_pop, family = binomial)
summary(model_coupon_pop)

##
## Call:
## glm(formula = y ~ ., family = binomial, data = train_pop)
##
## Coefficients:
##                                              Estimate Std. Error z value
## (Intercept)                                 -0.449457   0.190735  -2.356
## direction_same                               0.546348   0.080782   6.763
## to_coupon                                   -0.005761   0.043863  -0.131
## time10AM                                     0.009546   0.114852   0.083
## time10PM                                    -0.178749   0.090127  -1.983
## time2PM                                     -0.093579   0.115277  -0.812
```

```
## time6PM                                         0.247440   0.074967   3.301
## `couponCarry out & Take away`                   1.706757   0.086301  19.777
## `couponCoffee House`                            0.539244   0.077511   6.957
## `couponRestaurant(<20)`                         1.567721   0.085964  18.237
## `couponRestaurant(20-50)`                       0.442713   0.093868   4.716
## expiration2h                                   -0.895574   0.052809 -16.959
## genderMale                                      0.251958   0.051452   4.897
## ageSeniors                                     -0.178627   0.088636  -2.015
## ageTeenagers                                   -0.077375   0.154663  -0.500
## `ageYoung Adults`                              -0.031320   0.063905  -0.490
## `educationBachelors degree`                    -0.130463   0.091143  -1.431
## `educationGraduate degree (Masters or Doctorate)` -0.332881   0.106795  -3.117
## `educationHigh School Graduate`                 0.172426   0.124761   1.382
## `educationSome college - no degree`             0.090944   0.092008   0.988
## `educationSome High School`                     0.478981   0.307572   1.557
## occupationOthers                               -0.007960   0.099328  -0.080
## occupationRetired                              -0.193771   0.144428  -1.342
## `occupationService and sales`                   0.113164   0.084509   1.339
## occupationStudent                               0.043801   0.093517   0.468
## occupationTechnicians                           0.272327   0.085093   3.200
## occupationUnemployed                            0.027656   0.077213   0.358
## incomeLow_income                                0.106273   0.069337   1.533
## incomeMedium_income                             0.074968   0.065324   1.148
## `bar4~8`                                       -0.089579   0.103042  -0.869
## bargt8                                         -0.604374   0.172793  -3.498
## barless1                                       -0.239440   0.077168  -3.103
## barnever                                       -0.252012   0.074338  -3.390
## `coffee_house4~8`                              -0.077583   0.084202  -0.921
## coffee_housegt8                                -0.376788   0.102138  -3.689
## coffee_houseless1                              -0.436648   0.068882  -6.339
## coffee_housenever                              -0.882987   0.075468 -11.700
## `carry_away4~8`                                -0.093659   0.060104  -1.558
## carry_awaygt8                                  -0.045364   0.088733  -0.511
```

```
## carry_awayless1                              -0.197825   0.076001  -2.603
## carry_awaynever                               0.086507   0.222510   0.389
## `restaurant_less_than204~8`                   0.021260   0.062305   0.341
## restaurant_less_than20gt8                     0.124907   0.101870   1.226
## restaurant_less_than20less1                   0.027955   0.074655   0.374
## restaurant_less_than20never                   0.288160   0.196317   1.468
## `restaurant20to504~8`                         0.085629   0.118254   0.724
## restaurant20to50gt8                           0.147536   0.216497   0.681
## restaurant20to50less1                        -0.132626   0.060994  -2.174
## restaurant20to50never                        -0.222535   0.081885  -2.718
## `destination_passengerHome_Kid(s)`            0.090908   0.240947   0.377
## destination_passengerHome_Partner             0.271918   0.194806   1.396
## `destination_passengerNo Urgent Place_Alone`       0.858815   0.124298   6.909
## `destination_passengerNo Urgent Place_Friend(s)`   1.059323   0.097650  10.848
## `destination_passengerNo Urgent Place_Kid(s)`      0.272484   0.124814   2.183
## `destination_passengerNo Urgent Place_Partner`     1.099356   0.137487   7.996
## weather_temperatureSnowy_30                  -0.190845   0.108554  -1.758
## weather_temperatureSunny_30                   0.138594   0.122742   1.129
## weather_temperatureSunny_55                   0.480002   0.097288   4.934
## weather_temperatureSunny_80                   0.365287   0.088706   4.118
## `maritalstatus_childrenMarried partner_0`    -0.198070   0.088448  -2.239
## `maritalstatus_childrenMarried partner_1`     0.077688   0.070925   1.095
## `maritalstatus_childrenUnmarried partner_0`  -0.183550   0.083110  -2.209
## `maritalstatus_childrenUnmarried partner_1`  -0.086286   0.125647  -0.687
## maritalstatus_childrenWidowed_0              -0.732081   0.404057  -1.812
## maritalstatus_childrenWidowed_1               0.206293   0.306722   0.673
##                                                Pr(>|z|)
## (Intercept)                                    0.018451 *
## direction_same                                 1.35e-11 ***
## to_coupon                                      0.895511
## time10AM                                       0.933762
## time10PM                                       0.047335 *
## time2PM                                        0.416923
```

```
## time6PM                                    0.000965 ***
## `couponCarry out & Take away`              < 2e-16 ***
## `couponCoffee House`                       3.48e-12 ***
## `couponRestaurant(<20)`                    < 2e-16 ***
## `couponRestaurant(20-50)`                  2.40e-06 ***
## expiration2h                               < 2e-16 ***
## genderMale                                 9.73e-07 ***
## ageSeniors                                 0.043875 *
## ageTeenagers                               0.616878
## `ageYoung Adults`                          0.624059
## `educationBachelors degree`                0.152311
## `educationGraduate degree (Masters or Doctorate)` 0.001827 **
## `educationHigh School Graduate`            0.166956
## `educationSome college - no degree`        0.322940
## `educationSome High School`                0.119400
## occupationOthers                           0.936126
## occupationRetired                          0.179713
## `occupationService and sales`              0.180547
## occupationStudent                          0.639519
## occupationTechnicians                      0.001373 **
## occupationUnemployed                       0.720213
## incomeLow_income                           0.125353
## incomeMedium_income                        0.251124
## `bar4~8`                                   0.384661
## bargt8                                     0.000469 ***
## barless1                                   0.001917 **
## barnever                                   0.000699 ***
## `coffee_house4~8`                          0.356848
## coffee_housegt8                            0.000225 ***
## coffee_houseless1                          2.31e-10 ***
## coffee_housenever                          < 2e-16 ***
## `carry_away4~8`                            0.119169
## carry_awaygt8                              0.609180
```

```
## carry_awayless1                              0.009243 **
## carry_awaynever                              0.697440
## `restaurant_less_than204~8`                  0.732932
## restaurant_less_than20gt8                    0.220146
## restaurant_less_than20less1                  0.708063
## restaurant_less_than20never                  0.142151
## `restaurant20to504~8`                        0.468994
## restaurant20to50gt8                          0.495575
## restaurant20to50less1                        0.029675 *
## restaurant20to50never                        0.006575 **
## `destination_passengerHome_Kid(s)`           0.705955
## destination_passengerHome_Partner            0.162763
## `destination_passengerNo Urgent Place_Alone`      4.87e-12 ***
## `destination_passengerNo Urgent Place_Friend(s)`   < 2e-16 ***
## `destination_passengerNo Urgent Place_Kid(s)`     0.029027 *
## `destination_passengerNo Urgent Place_Partner`    1.28e-15 ***
## weather_temperatureSnowy_30                  0.078737 .
## weather_temperatureSunny_30                  0.258833
## weather_temperatureSunny_55                  8.06e-07 ***
## weather_temperatureSunny_80                  3.82e-05 ***
## `maritalstatus_childrenMarried partner_0`         0.025130 *
## `maritalstatus_childrenMarried partner_1`         0.273366
## `maritalstatus_childrenUnmarried partner_0`       0.027209 *
## `maritalstatus_childrenUnmarried partner_1`       0.492252
## maritalstatus_childrenWidowed_0              0.070013 .
## maritalstatus_childrenWidowed_1              0.501220
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12031  on 8794  degrees of freedom
## Residual deviance: 10369  on 8730  degrees of freedom
```

```
## AIC: 10499
##
## Number of Fisher Scoring iterations: 4
```

```r
coefficients(model_coupon_pop)
```

```
##                         (Intercept)
##                        -0.449456710
##                      direction_same
##                         0.546348165
##                           to_coupon
##                        -0.005760643
##                           time10AM
##                         0.009545626
##                           time10PM
##                        -0.178749065
##                            time2PM
##                        -0.093578779
##                            time6PM
##                         0.247440461
##     `couponCarry out & Take away`
##                         1.706757416
##              `couponCoffee House`
##                         0.539244374
##           `couponRestaurant(<20)`
##                         1.567720952
##         `couponRestaurant(20-50)`
##                         0.442712925
##                      expiration2h
##                        -0.895574204
##                        genderMale
##                         0.251958274
##                        ageSeniors
##                        -0.178626823
```

```
##                                    ageTeenagers
##                                    -0.077374660
##                                  `ageYoung Adults`
##                                    -0.031320349
##                          `educationBachelors degree`
##                                    -0.130463273
## `educationGraduate degree (Masters or Doctorate)`
##                                    -0.332881373
##                       `educationHigh School Graduate`
##                                     0.172426428
##                   `educationSome college - no degree`
##                                     0.090943635
##                          `educationSome High School`
##                                     0.478981195
##                                 occupationOthers
##                                    -0.007960169
##                                 occupationRetired
##                                    -0.193771122
##                        `occupationService and sales`
##                                     0.113163519
##                                 occupationStudent
##                                     0.043800669
##                              occupationTechnicians
##                                     0.272327302
##                              occupationUnemployed
##                                     0.027655766
##                                 incomeLow_income
##                                     0.106272510
##                               incomeMedium_income
##                                     0.074967836
##                                        `bar4~8`
##                                    -0.089578579
##                                          bargt8
```

```
## -0.604373772
## barless1
## -0.239440156
## barnever
## -0.252011515
## `coffee_house4~8`
## -0.077582643
## coffee_housegt8
## -0.376788490
## coffee_houseless1
## -0.436648056
## coffee_housenever
## -0.882986752
## `carry_away4~8`
## -0.093659082
## carry_awaygt8
## -0.045364493
## carry_awayless1
## -0.197825399
## carry_awaynever
## 0.086507125
## `restaurant_less_than204~8`
## 0.021260155
## restaurant_less_than20gt8
## 0.124907445
## restaurant_less_than20less1
## 0.027955416
## restaurant_less_than20never
## 0.288159682
## `restaurant20to504~8`
## 0.085629425
## restaurant20to50gt8
## 0.147535952
```

```
##                          restaurant20to50less1
##                                   -0.132626360
##                          restaurant20to50never
##                                   -0.222534833
##                 `destination_passengerHome_Kid(s)`
##                                    0.090907801
##                  destination_passengerHome_Partner
##                                    0.271917634
##     `destination_passengerNo Urgent Place_Alone`
##                                    0.858814810
##  `destination_passengerNo Urgent Place_Friend(s)`
##                                    1.059322736
##    `destination_passengerNo Urgent Place_Kid(s)`
##                                    0.272484125
##   `destination_passengerNo Urgent Place_Partner`
##                                    1.099356275
##                      weather_temperatureSnowy_30
##                                   -0.190844995
##                      weather_temperatureSunny_30
##                                    0.138594168
##                      weather_temperatureSunny_55
##                                    0.480001832
##                      weather_temperatureSunny_80
##                                    0.365287167
##         `maritalstatus_childrenMarried partner_0`
##                                   -0.198069629
##         `maritalstatus_childrenMarried partner_1`
##                                    0.077687719
##       `maritalstatus_childrenUnmarried partner_0`
##                                   -0.183549627
##       `maritalstatus_childrenUnmarried partner_1`
##                                   -0.086285705
##                    maritalstatus_childrenWidowed_0
```

```
##                                  -0.732080627
##                   maritalstatus_childrenWidowed_1
##                                   0.206292612
```

#Selecting the significant features and rerunning the model

```
summary(model_coupon_pop)$coefficients[, 4] <= 0.05
```

```
##                                   (Intercept)
##                                          TRUE
##                                direction_same
##                                          TRUE
##                                     to_coupon
##                                         FALSE
##                                      time10AM
##                                         FALSE
##                                      time10PM
##                                          TRUE
##                                       time2PM
##                                         FALSE
##                                       time6PM
##                                          TRUE
##                  `couponCarry out & Take away`
##                                          TRUE
##                        `couponCoffee House`
##                                          TRUE
##                      `couponRestaurant(<20)`
##                                          TRUE
##                    `couponRestaurant(20-50)`
##                                          TRUE
##                                 expiration2h
##                                          TRUE
##                                   genderMale
##                                          TRUE
##                                   ageSeniors
```

```
##                                            TRUE
##                                    ageTeenagers
##                                           FALSE
##                                 `ageYoung Adults`
##                                           FALSE
##                        `educationBachelors degree`
##                                           FALSE
## `educationGraduate degree (Masters or Doctorate)`
##                                            TRUE
##                    `educationHigh School Graduate`
##                                           FALSE
##                 `educationSome college - no degree`
##                                           FALSE
##                        `educationSome High School`
##                                           FALSE
##                                 occupationOthers
##                                           FALSE
##                                occupationRetired
##                                           FALSE
##                       `occupationService and sales`
##                                           FALSE
##                                occupationStudent
##                                           FALSE
##                            occupationTechnicians
##                                            TRUE
##                             occupationUnemployed
##                                           FALSE
##                                 incomeLow_income
##                                           FALSE
##                              incomeMedium_income
##                                           FALSE
##                                         `bar4~8`
##                                           FALSE
```

```
##                                    bargt8
##                                      TRUE
##                                  barless1
##                                      TRUE
##                                  barnever
##                                      TRUE
##                       `coffee_house4~8`
##                                     FALSE
##                         coffee_housegt8
##                                      TRUE
##                       coffee_houseless1
##                                      TRUE
##                       coffee_housenever
##                                      TRUE
##                         `carry_away4~8`
##                                     FALSE
##                           carry_awaygt8
##                                     FALSE
##                         carry_awayless1
##                                      TRUE
##                         carry_awaynever
##                                     FALSE
##           `restaurant_less_than204~8`
##                                     FALSE
##             restaurant_less_than20gt8
##                                     FALSE
##           restaurant_less_than20less1
##                                     FALSE
##           restaurant_less_than20never
##                                     FALSE
##                  `restaurant20to504~8`
##                                     FALSE
##                    restaurant20to50gt8
```

```
##                                        FALSE
##                          restaurant20to50less1
##                                         TRUE
##                          restaurant20to50never
##                                         TRUE
##               `destination_passengerHome_Kid(s)`
##                                        FALSE
##               destination_passengerHome_Partner
##                                        FALSE
##     `destination_passengerNo Urgent Place_Alone`
##                                         TRUE
## `destination_passengerNo Urgent Place_Friend(s)`
##                                         TRUE
##    `destination_passengerNo Urgent Place_Kid(s)`
##                                         TRUE
##   `destination_passengerNo Urgent Place_Partner`
##                                         TRUE
##                      weather_temperatureSnowy_30
##                                        FALSE
##                      weather_temperatureSunny_30
##                                        FALSE
##                      weather_temperatureSunny_55
##                                         TRUE
##                      weather_temperatureSunny_80
##                                         TRUE
##        `maritalstatus_childrenMarried partner_0`
##                                         TRUE
##        `maritalstatus_childrenMarried partner_1`
##                                        FALSE
##      `maritalstatus_childrenUnmarried partner_0`
##                                         TRUE
##      `maritalstatus_childrenUnmarried partner_1`
##                                        FALSE
```

```
##                 maritalstatus_childrenWidowed_0
##                                      FALSE
##                 maritalstatus_childrenWidowed_1
##                                      FALSE
```

```r
significant_vars_log <-
names(coef(model_coupon_pop))[summary(model_coupon_pop)$coefficients[, 4] <= 0.05]

significant_vars_log <- significant_vars_log[significant_vars_log != "(Intercept)"]
significant_vars_log
```

```
##  [1] "direction_same"
##  [2] "time10PM"
##  [3] "time6PM"
##  [4] "`couponCarry out & Take away`"
##  [5] "`couponCoffee House`"
##  [6] "`couponRestaurant(<20)`"
##  [7] "`couponRestaurant(20-50)`"
##  [8] "expiration2h"
##  [9] "genderMale"
## [10] "ageSeniors"
## [11] "`educationGraduate degree (Masters or Doctorate)`"
## [12] "occupationTechnicians"
## [13] "bargt8"
## [14] "barless1"
## [15] "barnever"
## [16] "coffee_housegt8"
## [17] "coffee_houseless1"
## [18] "coffee_housenever"
## [19] "carry_awayless1"
## [20] "restaurant20to50less1"
## [21] "restaurant20to50never"
## [22] "`destination_passengerNo Urgent Place_Alone`"
## [23] "`destination_passengerNo Urgent Place_Friend(s)`"
## [24] "`destination_passengerNo Urgent Place_Kid(s)`"
```

```
## [25] "`destination_passengerNo Urgent Place_Partner`"
## [26] "weather_temperatureSunny_55"
## [27] "weather_temperatureSunny_80"
## [28] "`maritalstatus_childrenMarried partner_0`"
## [29] "`maritalstatus_childrenUnmarried partner_0`"
```

```
formula_log <- as.formula(paste("y ~", paste(significant_vars_log, collapse = "+")))

model_coupon_pop <- glm(formula_log, data = train_pop, family = binomial)
summary(model_coupon_pop)
```

```
##
## Call:
## glm(formula = formula_log, family = binomial, data = train_pop)
##
## Coefficients:
##                                              Estimate Std. Error z value
## (Intercept)                                  -0.50064    0.10441  -4.795
## direction_same                                0.56557    0.07067   8.003
## time10PM                                     -0.12634    0.07164  -1.764
## time6PM                                       0.28368    0.05825   4.870
## `couponCarry out & Take away`                 1.72073    0.08491  20.265
## `couponCoffee House`                          0.51461    0.07472   6.887
## `couponRestaurant(<20)`                       1.53755    0.08173  18.813
## `couponRestaurant(20-50)`                     0.42570    0.09070   4.693
## expiration2h                                 -0.87043    0.05072 -17.161
## genderMale                                    0.22964    0.04911   4.677
## ageSeniors                                   -0.21716    0.07011  -3.097
## `educationGraduate degree (Masters or Doctorate)` -0.35105    0.06945  -5.055
## occupationTechnicians                         0.26979    0.07514   3.590
## bargt8                                       -0.48663    0.14873  -3.272
## barless1                                     -0.16821    0.06627  -2.538
## barnever                                     -0.16937    0.06188  -2.737
## coffee_housegt8                              -0.27214    0.09042  -3.010
## coffee_houseless1                            -0.40631    0.06050  -6.716
```

```
## coffee_housenever                              -0.85332    0.06578 -12.972
## carry_awayless1                                -0.17493    0.06816  -2.566
## restaurant20to50less1                          -0.16286    0.05424  -3.003
## restaurant20to50never                          -0.19296    0.07418  -2.601
## `destination_passengerNo Urgent Place_Alone`       0.86197    0.09009   9.568
## `destination_passengerNo Urgent Place_Friend(s)`   1.03988    0.07052  14.745
## `destination_passengerNo Urgent Place_Kid(s)`      0.27467    0.10024   2.740
## `destination_passengerNo Urgent Place_Partner`     1.08937    0.11281   9.657
## weather_temperatureSunny_55                     0.50760    0.07314   6.940
## weather_temperatureSunny_80                     0.38815    0.06047   6.419
## `maritalstatus_childrenMarried partner_0`        -0.21159    0.07975  -2.653
## `maritalstatus_childrenUnmarried partner_0`      -0.15535    0.07576  -2.051
##                                            Pr(>|z|)
## (Intercept)                                1.63e-06 ***
## direction_same                             1.21e-15 ***
## time10PM                                   0.07779 .
## time6PM                                    1.12e-06 ***
## `couponCarry out & Take away`              < 2e-16 ***
## `couponCoffee House`                       5.68e-12 ***
## `couponRestaurant(<20)`                    < 2e-16 ***
## `couponRestaurant(20-50)`                  2.69e-06 ***
## expiration2h                               < 2e-16 ***
## genderMale                                 2.92e-06 ***
## ageSeniors                                 0.00195 **
## `educationGraduate degree (Masters or Doctorate)` 4.30e-07 ***
## occupationTechnicians                      0.00033 ***
## bargt8                                     0.00107 **
## barless1                                   0.01115 *
## barnever                                   0.00620 **
## coffee_housegt8                            0.00261 **
## coffee_houseless1                          1.86e-11 ***
## coffee_housenever                          < 2e-16 ***
## carry_awayless1                            0.01028 *
```

```
## restaurant20to50less1                          0.00268 **
## restaurant20to50never                          0.00929 **
## `destination_passengerNo Urgent Place_Alone`       < 2e-16 ***
## `destination_passengerNo Urgent Place_Friend(s)`   < 2e-16 ***
## `destination_passengerNo Urgent Place_Kid(s)`      0.00614 **
## `destination_passengerNo Urgent Place_Partner`     < 2e-16 ***
## weather_temperatureSunny_55                    3.92e-12 ***
## weather_temperatureSunny_80                    1.38e-10 ***
## `maritalstatus_childrenMarried partner_0`         0.00798 **
## `maritalstatus_childrenUnmarried partner_0`       0.04030 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 12031  on 8794  degrees of freedom
## Residual deviance: 10430  on 8765  degrees of freedom
## AIC: 10490
##
## Number of Fisher Scoring iterations: 3
```

```
#Predicting the target variable using model_coupon_pop with training data set.

pred_pop_train1 <- predict(model_coupon_pop, newdata = train_pop, type = "response")
pred_class_pop_train1 <- ifelse(pred_pop_train1 > 0.5, 1, 0)
pred_class_pop_train1 <- as.factor(pred_class_pop_train1)
head(pred_class_pop_train1)
```

```
## 1 2 5 6 7 9
## 1 0 1 1 1 0
## Levels: 0 1
```

```
#Predict the target variable using model_coupon_pop with testing data set.

pred_pop_train <- predict(model_coupon_pop, newdata = test_pop, type = "response")
```

```r
pred_class_pop_train <- ifelse(pred_pop_train > 0.5, 1, 0)
pred_class_pop_train <- as.factor(pred_class_pop_train)
head(pred_class_pop_train)
```

```
##  3  4  8 10 19 22
##  1  0  0  0  0  0
## Levels: 0 1
```

```r
test_pop$y <- factor(test_pop$y, levels = c(0, 1))
train_pop$y <- factor(train_pop$y, levels = c(0, 1))

# Generating the confusion for both testing and training dataset.

conf_log_pop <- confusionMatrix(pred_class_pop_train, test_pop$y)
conf_log_train_pop <- confusionMatrix(pred_class_pop_train1, train_pop$y)
print(conf_log_pop)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0  921  521
##          1  705 1622
##
##                Accuracy : 0.6747
##                  95% CI : (0.6595, 0.6897)
##     No Information Rate : 0.5686
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3278
##
##  Mcnemar's Test P-Value : 1.728e-07
##
##             Sensitivity : 0.5664
##             Specificity : 0.7569
##          Pos Pred Value : 0.6387
```

```
##         Neg Pred Value : 0.6970
##             Prevalence : 0.4314
##         Detection Rate : 0.2444
##   Detection Prevalence : 0.3826
##      Balanced Accuracy : 0.6617
##
##       'Positive' Class : 0
##
```

```r
print(conf_log_train_pop)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 2193 1161
##          1 1610 3831
##
##                Accuracy : 0.6849
##                  95% CI : (0.6751, 0.6946)
##     No Information Rate : 0.5676
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.349
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.5767
##             Specificity : 0.7674
##          Pos Pred Value : 0.6538
##          Neg Pred Value : 0.7041
##              Prevalence : 0.4324
##          Detection Rate : 0.2493
##    Detection Prevalence : 0.3814
```

```
##      Balanced Accuracy : 0.6720
##
##       'Positive' Class : 0
##
```

- In the training set, the model achieved an accuracy of 68.49% with a balanced accuracy of 67.20%, sensitivity of 57.67%, and specificity of 76.74%. The positive predictive value (PPV) was 65.38%, and the negative predictive value (NPV) was 70.41%, with a Kappa of 0.349, indicating moderate agreement.
- In the testing set, the accuracy was slightly lower at 67.47%, with a balanced accuracy of 66.17%, sensitivity of 56.64%, and specificity of 75.69%. The PPV and NPV were 63.87% and 69.70%, respectively, and the Kappa was 0.3278, slightly lower than the training set.
- The training set consistently outperformed the testing set across all metrics, though the differences are minor, reflecting good generalizability. However, the slight drop in performance on the testing set suggests the model may still benefit from further tuning or additional data.
- Extracting top 20 features which explains most of the variability in target variable form the model

```
# Extracting coefficients
coefficients <- coef(model_coupon_dummy)

# Convert to a data frame for better visualization
feature_importance <- data.frame(
  Feature = names(coefficients),
  Coefficient = coefficients,
  Odds_Ratio = exp(coefficients)  # Calculate Odds Ratios
)

# Sort by absolute coefficient values
feature_importance <- feature_importance[order(abs(feature_importance$Coefficient),
decreasing = TRUE), ]

# Printing feature importance
print(feature_importance)
```

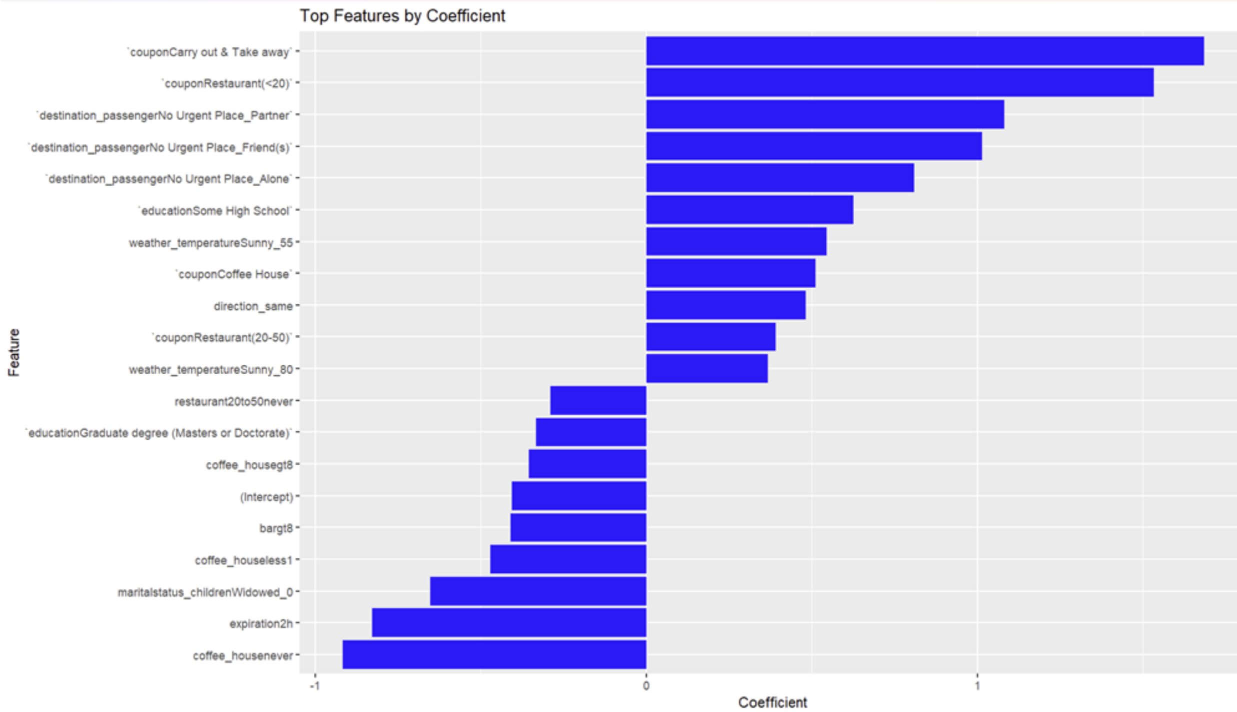
```
##                                                                              Feature
## `couponCarry out & Take away`                              `couponCarry out & Take away`
## `couponRestaurant(<20)`                                      `couponRestaurant(<20)`
## `destination_passengerNo Urgent Place_Partner`       `destination_passengerNo Urgent Place_Partner`
## `destination_passengerNo Urgent Place_Friend(s)`   `destination_passengerNo Urgent Place_Friend(s)`
## coffee_housenever                                                coffee_housenever
## expiration2h                                                        expiration2h
## `destination_passengerNo Urgent Place_Alone`          `destination_passengerNo Urgent Place_Alone`
## maritalstatus_childrenWidowed_0                            maritalstatus_childrenWidowed_0
## `educationSome High School`                                  `educationSome High School`
## weather_temperatureSunny_55                                 weather_temperatureSunny_55
## `couponCoffee House`                                            `couponCoffee House`
## direction_same                                                    direction_same
## coffee_houseless1                                                coffee_houseless1
## bargt8                                                               bargt8
## (Intercept)                                                         (Intercept)
## `couponRestaurant(20-50)`                                  `couponRestaurant(20-50)`
## weather_temperatureSunny_80                                 weather_temperatureSunny_80
## coffee_housegt8                                                  coffee_housegt8
## `educationGraduate degree (Masters or Doctorate)` `educationGraduate degree (Masters or Doctorate)`
## restaurant20to50never                                        restaurant20to50never
## maritalstatus_childrenWidowed_1                            maritalstatus_childrenWidowed_1
## `destination_passengerNo Urgent Place_Kid(s)`        `destination_passengerNo Urgent Place_Kid(s)`
## restaurant_less_than20never                               restaurant_less_than20never
## destination_passengerHome_Partner destination_passengerHome_Partner
## occupationTechnicians                                         occupationTechnicians
## genderMale                                                          genderMale
```

```
## time10PM                                                   time10PM
## barnever                                                   barnever
## `maritalstatus_childrenUnmarried partner_0`           `maritalstatus_childrenUnmarried
partner_0`
## time6PM                                                    time6PM
## weather_temperatureSunny_30                               weather_temperatureSunny_30
## occupationRetired                                          occupationRetired
## barless1                                                   barless1
## carry_awayless1                                            carry_awayless1
## `educationHigh School Graduate`                            `educationHigh School Graduate`
## ageSeniors                                                 ageSeniors
## `maritalstatus_childrenMarried partner_0`             `maritalstatus_childrenMarried
partner_0`
## weather_temperatureSnowy_30                               weather_temperatureSnowy_30
## restaurant_less_than20gt8                                 restaurant_less_than20gt8
## restaurant20to50less1                                     restaurant20to50less1
## `destination_passengerHome_Kid(s)`
`destination_passengerHome_Kid(s)`
## carry_awaygt8                                              carry_awaygt8
## `educationBachelors degree`                               `educationBachelors degree`
## time2PM                                                    time2PM
## incomeLow_income                                           incomeLow_income
## incomeMedium_income                                        incomeMedium_income
## `bar4~8`                                                   `bar4~8`
## `maritalstatus_childrenUnmarried partner_1`           `maritalstatus_childrenUnmarried
partner_1`
## `occupationService and sales`                             `occupationService and sales`
## `restaurant20to504~8`                                     `restaurant20to504~8`
## `carry_away4~8`                                            `carry_away4~8`
## carry_awaynever                                            carry_awaynever
## `educationSome college - no degree`                       `educationSome college - no degree`
## occupationOthers                                           occupationOthers
## `coffee_house4~8`                                          `coffee_house4~8`
```

```
## restaurant_less_than20less1                                   restaurant_less_than20less1
## `restaurant_less_than204~8`                                   `restaurant_less_than204~8`
## occupationUnemployed                                              occupationUnemployed
## restaurant20to50gt8                                             restaurant20to50gt8
## time10AM                                                            time10AM
## ageTeenagers                                                        ageTeenagers
## to_coupon                                                          to_coupon
## occupationStudent                                                occupationStudent
## `ageYoung Adults`                                                `ageYoung Adults`
## `maritalstatus_childrenMarried partner_1`                `maritalstatus_childrenMarried
partner_1`
##                                        Coefficient Odds_Ratio
## `couponCarry out & Take away`                     1.684407791  5.3892584
## `couponRestaurant(<20)`                           1.532894244  4.6315623
## `destination_passengerNo Urgent Place_Partner`     1.082414048  2.9517967
## `destination_passengerNo Urgent Place_Friend(s)`   1.014805019  2.7588254
## coffee_housenever                                -0.917558164  0.3994933
## expiration2h                                     -0.827329257  0.4372154
## `destination_passengerNo Urgent Place_Alone`       0.808423668  2.2443673
## maritalstatus_childrenWidowed_0                  -0.653356297  0.5202966
## `educationSome High School`                        0.625535333  1.8692464
## weather_temperatureSunny_55                        0.544959721  1.7245389
## `couponCoffee House`                               0.511177042  1.6672525
## direction_same                                     0.481884109  1.6191221
## coffee_houseless1                                -0.471069566  0.6243341
## bargt8                                           -0.410389284  0.6633920
## (Intercept)                                      -0.406667776  0.6658654
## `couponRestaurant(20-50)`                          0.390653222  1.4779459
## weather_temperatureSunny_80                        0.367909434  1.4447112
## coffee_housegt8                                  -0.354440391  0.7015659
## `educationGraduate degree (Masters or Doctorate)` -0.332203467  0.7173414
## restaurant20to50never                            -0.288969793  0.7490348
## maritalstatus_childrenWidowed_1                    0.277618322  1.3199823
```

```
## `destination_passengerNo Urgent Place_Kid(s)`      0.275114005  1.3166808
## restaurant_less_than20never                        0.269349832  1.3091130
## destination_passengerHome_Partner                  0.251456836  1.2858974
## occupationTechnicians                              0.233169457  1.2625954
## genderMale                                         0.229963715  1.2585543
## time10PM                                          -0.215805930  0.8058917
## barnever                                          -0.206228005  0.8136475
## `maritalstatus_childrenUnmarried partner_0`       -0.205686853  0.8140880
## time6PM                                            0.198193636  1.2191985
## weather_temperatureSunny_30                        0.185397486  1.2036968
## occupationRetired                                 -0.173479058  0.8407348
## barless1                                          -0.171973655  0.8420014
## carry_awayless1                                   -0.169126776  0.8444018
## `educationHigh School Graduate`                    0.168964673  1.1840783
## ageSeniors                                        -0.166684404  0.8464667
## `maritalstatus_childrenMarried partner_0`         -0.165961301  0.8470790
## weather_temperatureSnowy_30                       -0.159107723  0.8529045
## restaurant_less_than20gt8                          0.149975661  1.1618060
## restaurant20to50less1                             -0.149870427  0.8608195
## `destination_passengerHome_Kid(s)`                 0.144947943  1.1559794
## carry_awaygt8                                     -0.126380955  0.8812791
## `educationBachelors degree`                       -0.122873497  0.8843755
## time2PM                                           -0.119829530  0.8870716
## incomeLow_income                                   0.116302316  1.1233354
## incomeMedium_income                                0.108729577  1.1148608
## `bar4~8`                                          -0.100722194  0.9041842
## `maritalstatus_childrenUnmarried partner_1`       -0.084961493  0.9185477
## `occupationService and sales`                      0.081493891  1.0849066
## `restaurant20to504~8`                              0.076074205  1.0790426
## `carry_away4~8`                                   -0.065651626  0.9364570
## carry_awaynever                                    0.064629834  1.0667641
## `educationSome college - no degree`                0.064575396  1.0667060
## occupationOthers                                  -0.061288090  0.9405522
```

```
## `coffee_house4~8`                          -0.048010007  0.9531242
## restaurant_less_than20less1                  0.047876194  1.0490408
## `restaurant_less_than204~8`                  0.041428647  1.0422988
## occupationUnemployed                        -0.039507622  0.9612626
## restaurant20to50gt8                          0.037124407  1.0378221
## time10AM                                    -0.020583336  0.9796271
## ageTeenagers                                 0.017307052  1.0174577
## to_coupon                                   -0.016689088  0.9834494
## occupationStudent                            0.009832333  1.0098808
## `ageYoung Adults`                            0.009155182  1.0091972
## `maritalstatus_childrenMarried partner_1`   -0.006281472  0.9937382

# Plot top features
library(ggplot2)
feature_importance <- feature_importance[order(abs(feature_importance$Coefficient),
decreasing = TRUE), ]
top_features <- head(feature_importance, 20)

ggplot(top_features, aes(x = reorder(Feature, Coefficient), y = Coefficient)) +
  geom_bar(stat = "identity", fill = "blue") +
  coord_flip() +
  labs(title = "Top Features by Coefficient", x = "Feature", y = "Coefficient")
```



Linear Discriminant Analysis.

Code:

```
################################# LDA for larger data set #####################
lda_pop <- lda(y ~ ., data = train_pop)
coefficients(lda_pop)

##                                              LD1
## direction_same                          0.598727546
## to_coupon                              -0.037174179
## time10AM                                0.053044431
## time10PM                               -0.180590068
## time2PM                                -0.105851133
## time6PM                                 0.270983539
## `couponCarry out & Take away`                1.912997070
## `couponCoffee House`                     0.629513318
## `couponRestaurant(<20)`                   1.747857616
```

```
## `couponRestaurant(20-50)`                        0.512392820
## expiration2h                                    -0.987182451
## genderMale                                       0.271953272
## ageSeniors                                      -0.197723552
## ageTeenagers                                    -0.090332876
## `ageYoung Adults`                               -0.031113310
## `educationBachelors degree`                     -0.147279710
## `educationGraduate degree (Masters or Doctorate)` -0.363626893
## `educationHigh School Graduate`                  0.190998786
## `educationSome college - no degree`              0.090245024
## `educationSome High School`                      0.429322500
## occupationOthers                                -0.009089473
## occupationRetired                               -0.215608536
## `occupationService and sales`                    0.132757269
## occupationStudent                                0.052836653
## occupationTechnicians                            0.294296619
## occupationUnemployed                             0.035769730
## incomeLow_income                                 0.111265262
## incomeMedium_income                              0.075278409
## `bar4~8`                                        -0.101436764
## bargt8                                          -0.640426220
## barless1                                        -0.256641835
## barnever                                        -0.272267527
## `coffee_house4~8`                               -0.087092212
## coffee_housegt8                                 -0.401488849
## coffee_houseless1                               -0.472202689
## coffee_housenever                               -0.966546674
## `carry_away4~8`                                 -0.095305610
## carry_awaygt8                                   -0.046994097
## carry_awayless1                                 -0.210042561
## carry_awaynever                                  0.109213813
## `restaurant_less_than204~8`                      0.018433382
## restaurant_less_than20gt8                        0.127978683
```

```
## restaurant_less_than20less1                     0.023099504
## restaurant_less_than20never                     0.292477262
## `restaurant20to504~8`                           0.071396270
## restaurant20to50gt8                             0.134529430
## restaurant20to50less1                          -0.141770677
## restaurant20to50never                          -0.243083514
## `destination_passengerHome_Kid(s)`              0.090874588
## destination_passengerHome_Partner               0.309789381
## `destination_passengerNo Urgent Place_Alone`      0.910393242
## `destination_passengerNo Urgent Place_Friend(s)`  1.141067067
## `destination_passengerNo Urgent Place_Kid(s)`     0.301238215
## `destination_passengerNo Urgent Place_Partner`    1.184402655
## weather_temperatureSnowy_30                    -0.203016605
## weather_temperatureSunny_30                     0.177753015
## weather_temperatureSunny_55                     0.509120438
## weather_temperatureSunny_80                     0.413031950
## `maritalstatus_childrenMarried partner_0`      -0.215345235
## `maritalstatus_childrenMarried partner_1`       0.083558557
## `maritalstatus_childrenUnmarried partner_0`    -0.194494477
## `maritalstatus_childrenUnmarried partner_1`    -0.079220440
## maritalstatus_childrenWidowed_0                -0.770606651
## maritalstatus_childrenWidowed_1                 0.238772819

##Predicting values for training data set using lda_pop model
pred_lda_pop_train <- predict(lda_pop, newdata = train_pop)
pred_lda_pop_train <- pred_lda_pop_train$class

##Predicting values for testing data set using lda_pop model

pred_lda_pop <- predict(lda_pop, newdata = test_pop)
pred_lda_pop <- pred_lda_pop$class

#Generating confusion matrix for testing and training data set
```

```
lda_conf_pop <- confusionMatrix(pred_lda_pop, as.factor(test_pop$y))
print(lda_conf_pop)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0  924  530
##          1  702 1613
##
##                Accuracy : 0.6731
##                  95% CI : (0.6579, 0.6881)
##     No Information Rate : 0.5686
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3251
##
##  Mcnemar's Test P-Value : 1.106e-06
##
##             Sensitivity : 0.5683
##             Specificity : 0.7527
##          Pos Pred Value : 0.6355
##          Neg Pred Value : 0.6968
##              Prevalence : 0.4314
##          Detection Rate : 0.2452
##    Detection Prevalence : 0.3858
##       Balanced Accuracy : 0.6605
##
##        'Positive' Class : 0
##
```

```
lda_conf_pop_train <- confusionMatrix(pred_lda_pop_train, as.factor(train_pop$y))
print(lda_conf_pop_train)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 2225 1141
##          1 1578 3851
##
##                Accuracy : 0.6908
##                  95% CI : (0.6811, 0.7005)
##     No Information Rate : 0.5676
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3614
##
##  Mcnemar's Test P-Value : < 2.2e-16
##
##             Sensitivity : 0.5851
##             Specificity : 0.7714
##          Pos Pred Value : 0.6610
##          Neg Pred Value : 0.7093
##              Prevalence : 0.4324
##          Detection Rate : 0.2530
##    Detection Prevalence : 0.3827
##       Balanced Accuracy : 0.6782
##
##        'Positive' Class : 0
##
```

- The model achieved an accuracy of 69.08% on the training set, with a balanced accuracy of 67.82%, sensitivity of 58.51%, and specificity of 77.14%. The predictive values were also strong, with a positive predictive value of 66.10% and a negative predictive value of 70.93%. The Kappa value of 0.3614 indicates moderate agreement between predictions and actual values, demonstrating solid performance on the training data.
- For the testing set, the model achieved an accuracy of 67.31%, slightly lower than the training set. The balanced accuracy was 66.05%, with sensitivity of 56.83%

and specificity of 75.27%. The positive predictive value was 63.55%, and the negative predictive value was 69.68%.

- The training set outperforms the testing set across all metrics, showing slightly better detection rates and predictive reliability. However, the difference is minimal, indicating that the model generalizes well but could benefit from further tuning to improve performance on unseen data.

Quadratic Discriminant Analysis

Code:

```
######################################QDA for Larger data set ##################

qda_pop <- qda(y ~ ., data = train_pop)
summary(qda_pop)

##         Length Class  Mode
## prior      2   -none- numeric
## counts     2   -none- numeric
## means    128   -none- numeric
## scaling 8192   -none- numeric
## ldet       2   -none- numeric
## lev        2   -none- character
## N          1   -none- numeric
## call       3   -none- call
## terms      3   terms  call
## xlevels    0   -none- list

coefficients(qda_pop)

## NULL

#Predicting values for training data set

pred_qda_pop1 <- predict(qda_pop, newdata = train_pop)
pred_qda_pop1 <- pred_qda_pop1$class

#Predicting values for testing data set
```

```
pred_qda_pop <- predict(qda_pop, newdata = test_pop)
pred_qda_pop <- pred_qda_pop$class

qda_conf_pop <- confusionMatrix(pred_qda_pop, as.factor(test_pop$y))
print(qda_conf_pop)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 1014  516
##          1  612 1627
##
##                Accuracy : 0.7007
##                  95% CI : (0.6858, 0.7153)
##     No Information Rate : 0.5686
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.3856
##
##  Mcnemar's Test P-Value : 0.004675
##
##             Sensitivity : 0.6236
##             Specificity : 0.7592
##          Pos Pred Value : 0.6627
##          Neg Pred Value : 0.7267
##              Prevalence : 0.4314
##          Detection Rate : 0.2690
##    Detection Prevalence : 0.4059
##       Balanced Accuracy : 0.6914
##
##        'Positive' Class : 0
##
```

```r
qda_conf_pop1 <- confusionMatrix(pred_qda_pop1, as.factor(train_pop$y))
print(qda_conf_pop1)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    0    1
##          0 2509 1065
##          1 1294 3927
##
##                Accuracy : 0.7318
##                  95% CI : (0.7224, 0.741)
##     No Information Rate : 0.5676
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.4496
##
##  Mcnemar's Test P-Value : 2.675e-06
##
##             Sensitivity : 0.6597
##             Specificity : 0.7867
##          Pos Pred Value : 0.7020
##          Neg Pred Value : 0.7522
##              Prevalence : 0.4324
##          Detection Rate : 0.2853
##    Detection Prevalence : 0.4064
##       Balanced Accuracy : 0.7232
##
##        'Positive' Class : 0
##
```

- The model achieved an accuracy of 73.18% on the training set, with a balanced accuracy of 72.32%, sensitivity of 65.97%, and specificity of 78.67%. The positive predictive value was 70.20%, and the negative predictive value was 75.22%.

- On the testing set, the model achieved an accuracy of 70.07%, slightly lower than the training set, with a balanced accuracy of 69.14%. The sensitivity was 62.36%, and specificity was 75.92%, showing a slight decrease in performance compared to the training set. The positive predicted value was 66.27%, and the Negative predicted value was 72.67%.

- The model performs better on the training set than on the testing set, as evidenced by higher accuracy, balanced accuracy, sensitivity, and Kappa values. However, the testing set performance remains competitive, indicating that the model generalizes well with minimal overfitting. Further tuning could help close the performance gap between the datasets.

X. Challenges Faced

- Dimensionality reduction for the features.
- Singularity issue observed during model building.
- Tackling multi-collinearity issues.

XI. CONCLUSION

- Logistic Regression, LDA, and QDA were compared in terms of their performance based on the accuracy of training, accuracy of test, and sensitivity. Each model had different strengths and weaknesses, revealing a trade-off between accuracy, sensitivity and generalizability
- Logistic Regression gave the best balance of training and test performance, with 68.57% training accuracy and 64% test accuracy. These consistent results indicate a strong ability to generalize new data without significant overfitting or underfitting.
- LDA was relatively performing well in terms of sensitivity, with a sensitivity of 61.19%. However, its test accuracy was lower at 62.33%, which may suggest that it is slightly overfitting or has lower generalization ability.
- QDA had the best training accuracy of 87.29%, showing that it was able to model the most complex patterns in the training dataset. However, this was at the expense of test accuracy, which fell to 63.67%, indicating significant overfitting. While QDA can capture non-linear relationships effectively, it struggles to maintain performance on new data.
- Therefore, based on the given dataset and problem, Logistic Regression is the best model among the three models, showing almost uniform performance on both training and test datasets. It is the best model since it is a good compromise between simplicity, interpretability, and predictive accuracy. This model could be further improved with fine-tuning or considering different regularization techniques.

XII. FUTURE SCOPE:

- To enhance the performance of this model and derive deeper insights, it is highly recommended to explore advanced machine learning techniques such as Random Forest and XGBoost. These ensemble methods are well-suited for handling nonlinear relationships, feature interactions, and imbalanced datasets, potentially yielding better accuracy and generalizability. Additionally, it would transform this binary classification task into a multi-class problem for predicting specific coupon names, such as "Coffee House" or "Carry Out & Take Away" or “Bar” or “Restaurants (<20)” or “Restaurants (20-50)”, thus providing more actionable insights into which coupons really resonate most with which segments of customers.

- More sophisticated feature engineering and feature selection could also make substantial improvements. For example, interaction term creation, aggregation of customer behavior metrics, and analysis of temporal patterns might bring out latent relationships in the data. Feature selection techniques such as Recursive Feature Elimination or SHAP values can be used to identify the most important features. Access to larger and more diverse data would also lead to greater robustness in the models and lower overfitting risks. By implementing these strategies, the model will be more accurate and provide more impactful insights to help in optimizing coupon marketing strategies.
- It is not necessary to perform outlier detection for nominal data, since traditional statistical methods do not apply directly, but such detection can also add some value in identifying unusual patterns or rare categories. Outlier analysis on nominal data is not pursued in this work because of time, but it is a potential future direction to increase the robustness and accuracy of the model.

XIII. References:

- **Dataset:** https://archive.ics.uci.edu/static/public/603/in+vehicle+coupon+recommendation.zip
- **Research Paper:** https://jmlr.org/papers/volume18/16-003/16-003.pdf
- https://github.com/dikaaka/In-Vehicle-Coupon-Recommendation-Project/blob/main/STAGE%200/FINAL%20PROJECT%20-%20STAGE%200.pdf
- https://www.kaggle.com/code/maherabdelllatif/invehicle-coupon-recommendation