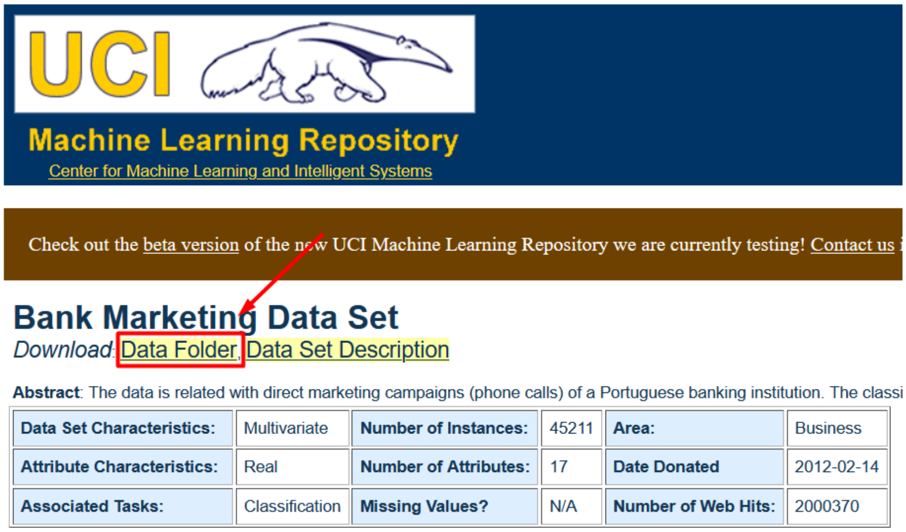
Pepito, Alyssa Mae & Soleño, Keziah Antonette | Group 18 BM2 CS174

Chapter Project (Regression Model)

Previously in M1-FA1, we installed R and R studio to get started with R programming. Using R studio, we were tasked to run an R script using a provided dataset and observe the number of data, and percentage of train and test datasets, list all variables and identify qualitative and quantitative attributes. For this assessment, we are tasked with creating a simple multiple regression model using R Studio.

I. Downloading the bank.zip file

To get started, we had to download our dataset from the provided link, https://archive.ics.uci.edu/ml/datasets/Bank+Marketing. First, we had to click on *Data Folder*, then it directed us to another webpage listing two zipped files and a link to the parent directory.



UCI

Machine Learning Repository

Center for Machine Learning and Intelligent Systems

Check out the beta version of the new UCI Machine Learning Repository we are currently testing! Contact us i

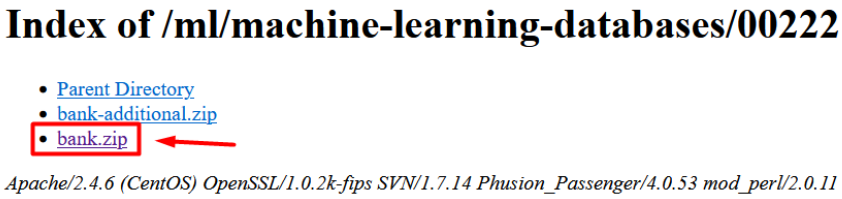
Bank Marketing Data Set

Download: Data Folder, Data Set Description

Abstract: The data is related with direct marketing campaigns (phone calls) of a Portuguese banking institution. The classi

Data Set Characteristics:	Multivariate	**Number of Instances:**	45211	**Area:**	Business
Attribute Characteristics:	Real	**Number of Attributes:**	17	**Date Donated**	2012-02-14
Associated Tasks:	Classification	**Missing Values?**	N/A	**Number of Web Hits:**	2000370

Our dataset should be inside the *bank.zip* file, so we clicked on it to start downloading.

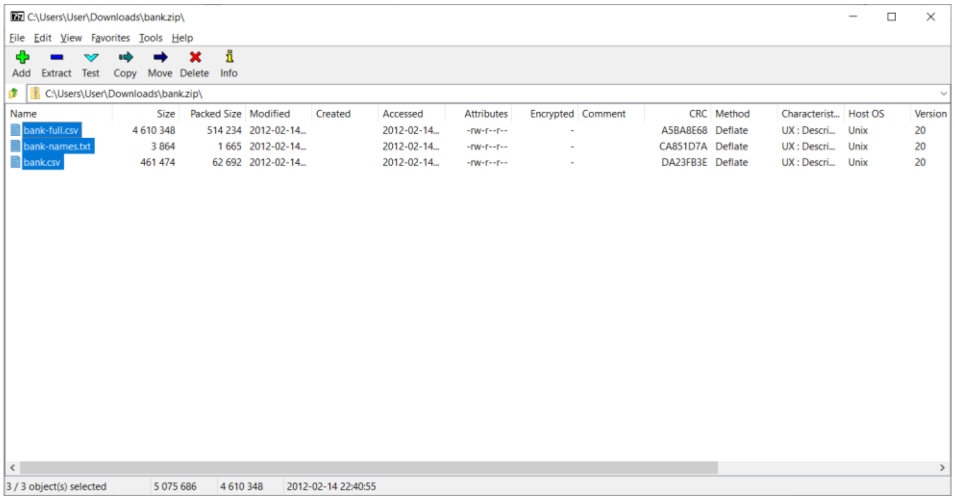


Index of /ml/machine-learning-databases/00222

- Parent Directory
- bank-additional.zip
- bank.zip

Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips SVN/1.7.14 Phusion_Passenger/4.0.53 mod_perl/2.0.11

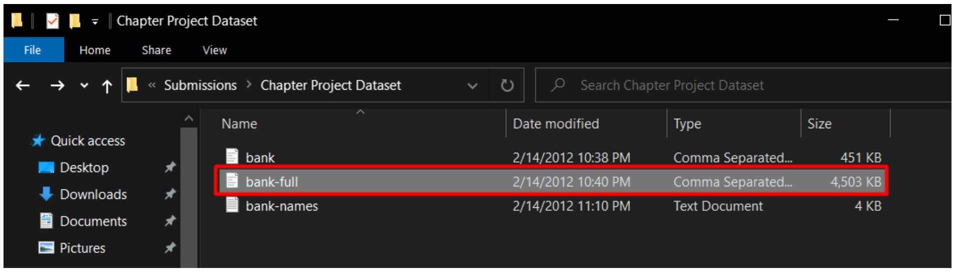
After it finished downloading, we extracted the zipped file using a file archiver, such as 7-zip, to extract the contents to a folder of our choosing.


C:\Users\User\Downloads\bank.zip\
File Edit View Favorites Tools Help
Add Extract Test Copy Move Delete Info
C:\Users\User\Downloads\bank.zip\

Name	Size	Packed Size	Modified	Created	Accessed	Attributes	Encrypted	Comment	CRC	Method	Characterist...	Host OS	Version
bank-full.csv	4 610 348	514 234	2012-02-14...		2012-02-14...	-rw-r--r--	-		A5BA8E68	Deflate	UX : Descri...	Unix	20
bank-names.txt	3 864	1 665	2012-02-14...		2012-02-14...	-rw-r--r--	-		CA851D7A	Deflate	UX : Descri...	Unix	20
bank.csv	461 474	62 692	2012-02-14...		2012-02-14...	-rw-r--r--	-		DA23FB3E	Deflate	UX : Descri...	Unix	20

3 / 3 object(s) selected   5 075 686   4 610 348   2012-02-14 22:40:55

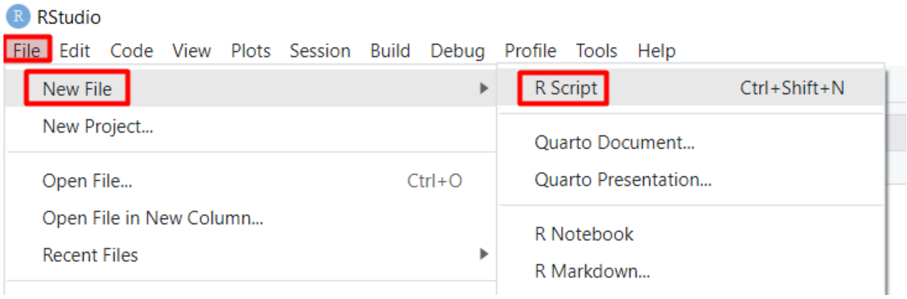
In this assessment, we would be using the *bank-full* csv file as our dataset in creating the multiple regression model using R Studio.

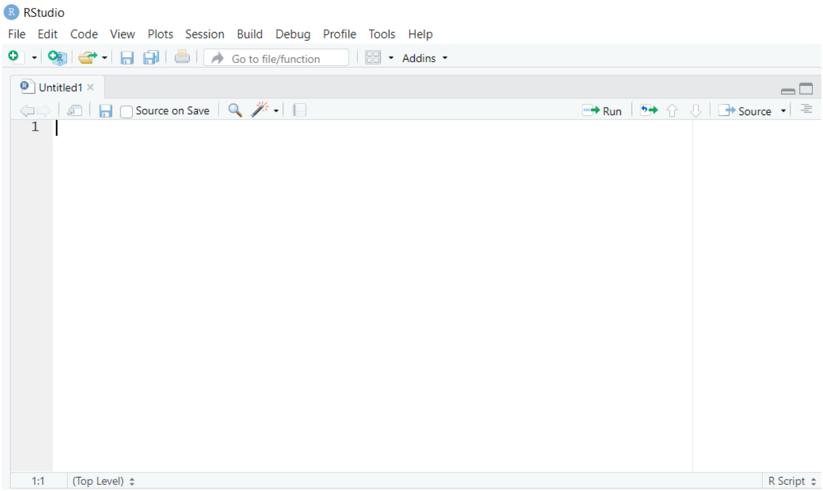


Name	Date modified	Type	Size
bank	2/14/2012 10:38 PM	Comma Separated...	451 KB
bank-full	2/14/2012 10:40 PM	Comma Separated...	4,503 KB
bank-names	2/14/2012 11:10 PM	Text Document	4 KB

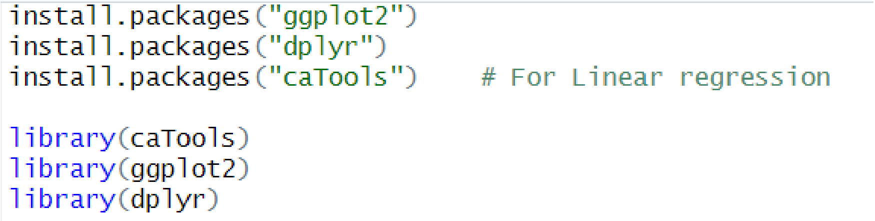
II. Preparing the bank-full dataset

To begin, we first opened R Studio and clicked on File > New File > R Script or simply enter the shortcut Ctrl + Shift + N to open a blank new R Script.





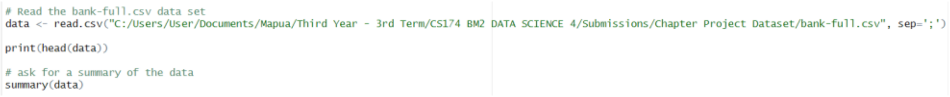
Inside our Rscript, we will first install the necessary libraries. These libraries will be useful in mapping variables, plotting, and creating our linear regression model.



```
install.packages("ggplot2")
install.packages("dplyr")
install.packages("caTools")     # For Linear regression

library(caTools)
library(ggplot2)
library(dplyr)
```

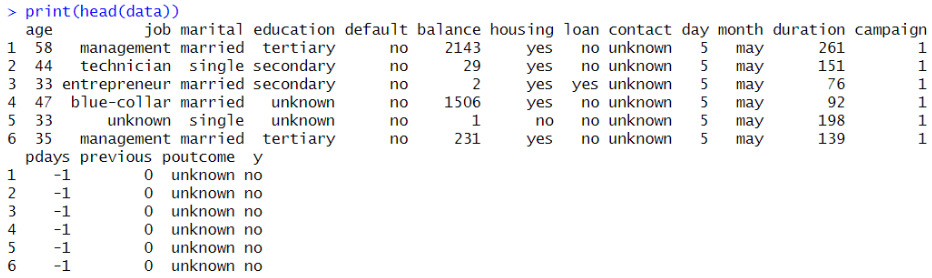
Then, we will read the bank-full.csv data set and print the first six rows of the data frame to confirm if the data was properly attached in R Studio. After, we use the *summary* function to summarize the values in the data frame.



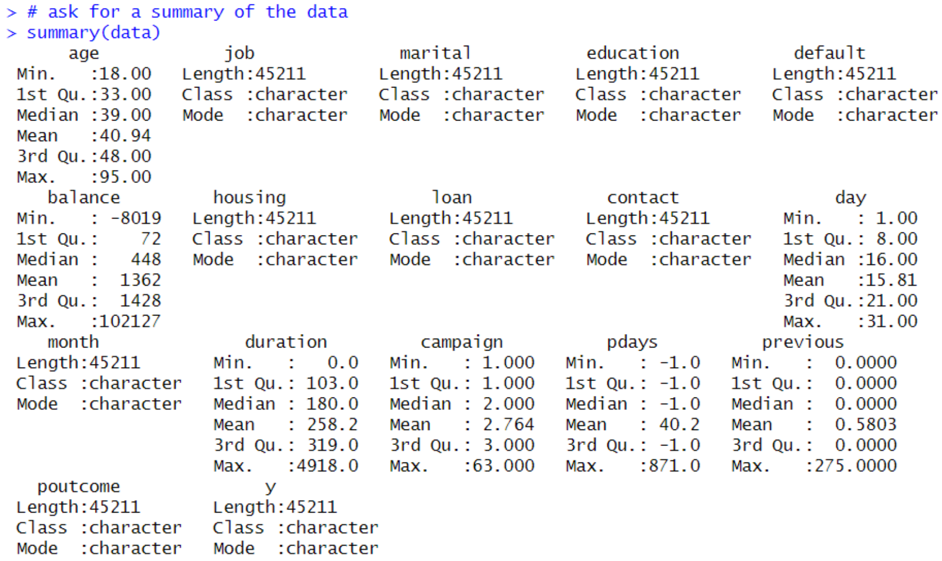
```
# Read the bank-full.csv data set
data <- read.csv("C:/Users/User/Documents/Mapua/Third Year - 3rd Term/CS174 BM2 DATA SCIENCE 4/Submissions/Chapter Project Dataset/bank-full.csv", sep=';')

print(head(data))

# ask for a summary of the data
summary(data)
```



```
> print(head(data))
  age          job marital education default balance housing loan contact day month duration campaign
1  58   management married  tertiary      no    2143     yes   no unknown   5   may      261        1
2  44   technician  single secondary      no      29     yes   no unknown   5   may      151        1
3  33 entrepreneur married secondary      no       2     yes  yes unknown   5   may       76        1
4  47  blue-collar married   unknown      no    1506     yes   no unknown   5   may       92        1
5  33      unknown  single   unknown      no       1      no   no unknown   5   may      198        1
6  35   management married  tertiary      no     231     yes   no unknown   5   may      139        1
  pdays previous poutcome  y
1    -1        0  unknown no
2    -1        0  unknown no
3    -1        0  unknown no
4    -1        0  unknown no
5    -1        0  unknown no
6    -1        0  unknown no
```



```
> # ask for a summary of the data
> summary(data)
      age            job              marital           education           default         
 Min.   :18.00   Length:45211       Length:45211       Length:45211       Length:45211      
 1st Qu.:33.00   Class :character   Class :character   Class :character   Class :character  
 Median :39.00   Mode  :character   Mode  :character   Mode  :character   Mode  :character  
 Mean   :40.94                                                                              
 3rd Qu.:48.00                                                                              
 Max.   :95.00                                                                              
    balance         housing              loan             contact               day       
 Min.   : -8019   Length:45211       Length:45211       Length:45211       Min.   : 1.00  
 1st Qu.:    72   Class :character   Class :character   Class :character   1st Qu.: 8.00  
 Median :   448   Mode  :character   Mode  :character   Mode  :character   Median :16.00  
 Mean   :  1362                                                            Mean   :15.81  
 3rd Qu.:  1428                                                            3rd Qu.:21.00  
 Max.   :102127                                                            Max.   :31.00  
    month              duration         campaign          pdays          previous       
 Length:45211       Min.   :   0.0   Min.   : 1.000   Min.   : -1.0   Min.   :  0.0000  
 Class :character   1st Qu.: 103.0   1st Qu.: 1.000   1st Qu.: -1.0   1st Qu.:  0.0000  
 Mode  :character   Median : 180.0   Median : 2.000   Median : -1.0   Median :  0.0000  
                    Mean   : 258.2   Mean   : 2.764   Mean   : 40.2   Mean   :  0.5803  
                    3rd Qu.: 319.0   3rd Qu.: 3.000   3rd Qu.: -1.0   3rd Qu.:  0.0000  
                    Max.   :4918.0   Max.   :63.000   Max.   :871.0   Max.   :275.0000  
   poutcome              y            
 Length:45211       Length:45211      
 Class :character   Class :character  
 Mode  :character   Mode  :character  
```

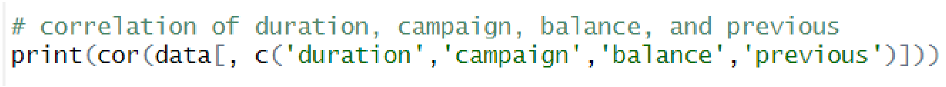
Before creating the multiple regression model, we noted the attribute information on the site https://archive.ics.uci.edu/ml/datasets/Bank+Marketing and considered three input variables as our independent variable and one dependent variable for our analysis.

Our three input variables were the following:

- **Campaign** - number of contacts performed during the direct marketing campaign of a Portuguese banking institute and for this client (numeric, includes last contact)
- **Balance** - the amount of money in a bank account at a given time (numeric)
- **Previous** - number of contacts performed before this campaign and for this client (numeric)

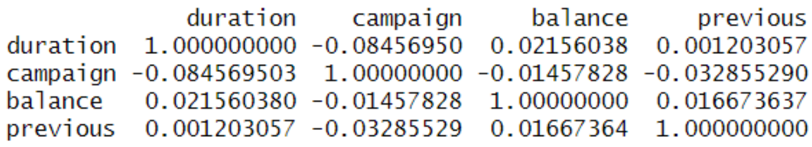
We noticed that the supposed output/dependent variable in the dataset was *y = has the client subscribed to a term deposit? (binary: 'yes', 'no')*. However, regression models require the dependent variable to be numerical. So, instead of y, we used **duration** as it highly affects the output target (e.g., if duration=0, then y='no'). If the duration is more than 0, it would mean the client subscribed to a term deposit.

Moving forward with these variables, we used the *cor* function to determine the correlation between the four variables.



```
# correlation of duration, campaign, balance, and previous
print(cor(data[, c('duration','campaign','balance','previous')]))
```

Which gave the following table:



```
            duration    campaign     balance     previous
duration 1.000000000 -0.08456950  0.02156038  0.001203057
campaign -0.084569503  1.00000000 -0.01457828 -0.032855290
balance   0.021560380 -0.01457828  1.00000000  0.016673637
previous  0.001203057 -0.03285529  0.01667364  1.000000000
```

First, correlation ranges from -1 to 1. It gives us an indication on two things:

1. The direction of the relationship between the 2 variables
2. The strength of the relationship between the 2 variables

Looking at the table, duration and campaign has a negative correlation implying that the two variables vary in opposite directions, that is, if a variable increases the other decreases and vice versa. However, as the correlation is closer to 0 than to 1, it may also indicate that the two variables are independent, that is, as one variable increases, there is no tendency in the other variable to either decrease or increase. The same goes for the balance and previous variables, despite having a positive correlation.

III. Creating the Multiple Regression Model

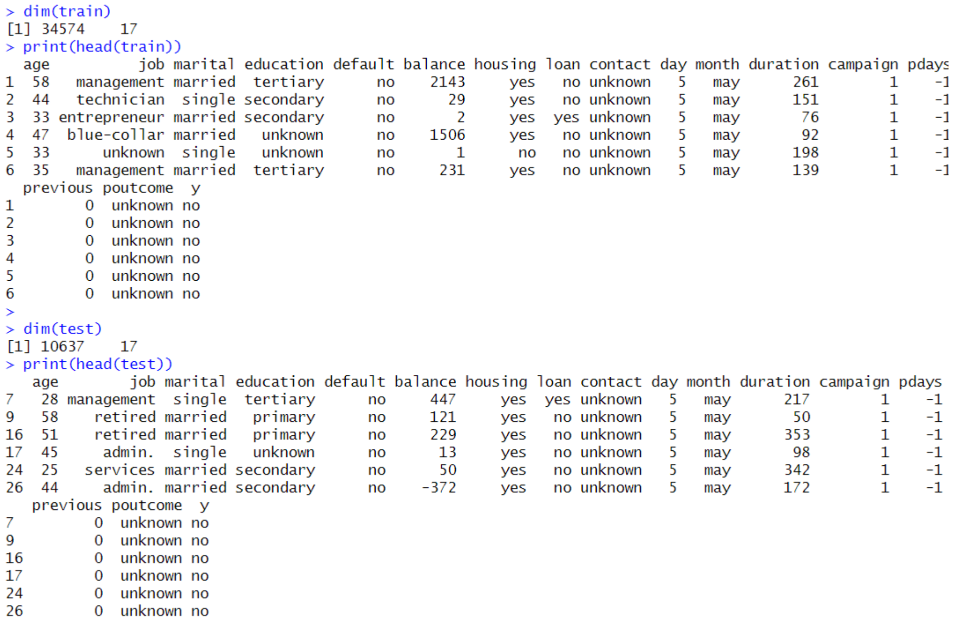
Now that we have attached our dataset, we will now be using the *split* function to split the dataset into 80% for training and 20% for testing.

```
> # splitting of data
> split <- sample.split(data, SplitRatio = 0.8)
> split
 [1]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
TRUE  TRUE  TRUE FALSE FALSE
```

The train dataset gets all the data points that are 'TRUE' and similarly the test dataset gets all the data points which are 'FALSE'.

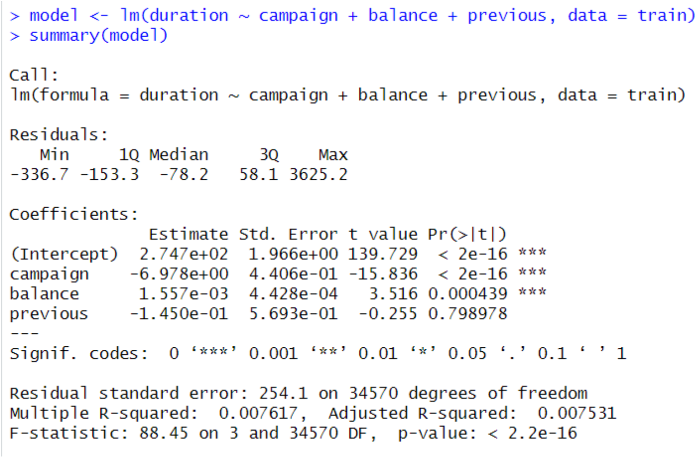
```
> train <- subset(data, split == "TRUE")
> test <- subset(data, split == "FALSE")
```

We then display the training and testing datasets using the *dim* and *print* functions.



```
> dim(train)
[1] 34574    17
> print(head(train))
  age          job marital education default balance housing loan contact day month duration campaign pdays
1  58   management married  tertiary      no    2143     yes   no unknown   5   may      261        1    -1
2  44   technician  single secondary      no      29     yes   no unknown   5   may      151        1    -1
3  33 entrepreneur married secondary      no       2     yes  yes unknown   5   may       76        1    -1
4  47  blue-collar married   unknown      no    1506     yes   no unknown   5   may       92        1    -1
5  33      unknown  single   unknown      no       1      no   no unknown   5   may      198        1    -1
6  35   management married  tertiary      no     231     yes   no unknown   5   may      139        1    -1
  previous poutcome  y
1        0  unknown no
2        0  unknown no
3        0  unknown no
4        0  unknown no
5        0  unknown no
6        0  unknown no
>
> dim(test)
[1] 10637    17
> print(head(test))
   age        job marital education default balance housing loan contact day month duration campaign pdays
7   28 management  single  tertiary      no     447     yes  yes unknown   5   may      217        1    -1
9   58    retired married   primary      no     121     yes   no unknown   5   may       50        1    -1
16  51    retired married   primary      no     229     yes   no unknown   5   may      353        1    -1
17  45     admin.  single   unknown      no      13     yes   no unknown   5   may       98        1    -1
24  25   services married secondary      no      50     yes   no unknown   5   may      342        1    -1
26  44     admin. married secondary      no    -372     yes   no unknown   5   may      172        1    -1
   previous poutcome  y
7         0  unknown no
9         0  unknown no
16        0  unknown no
17        0  unknown no
24        0  unknown no
26        0  unknown no
```

After, we used the *lm* function to fit linear models to data frames in the R Language.



```
> model <- lm(duration ~ campaign + balance + previous, data = train)
> summary(model)

Call:
lm(formula = duration ~ campaign + balance + previous, data = train)

Residuals:
   Min     1Q Median     3Q    Max 
-336.7 -153.3  -78.2   58.1 3625.2 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  2.747e+02  1.966e+00 139.729  < 2e-16 ***
campaign    -6.978e+00  4.406e-01 -15.836  < 2e-16 ***
balance      1.557e-03  4.428e-04   3.516 0.000439 ***
previous    -1.450e-01  5.693e-01  -0.255 0.798978    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 254.1 on 34570 degrees of freedom
Multiple R-squared:  0.007617,	Adjusted R-squared:  0.007531 
F-statistic: 88.45 on 3 and 34570 DF,  p-value: < 2.2e-16
```

The output reveals three sections: residuals, coefficients, and performance measures. The **residuals** section summarizes the residuals, the error between the prediction of the model and the actual results. It is noted that smaller residuals are better. In the **coefficients** section, for each variable and the intercept, a weight is produced, and that weight has other attributes like the standard error, a t-test value and significance. Lastly, under the **performance** section, three sets of measurements are provided: residual standard error, multiple r-square, and f-statistic.

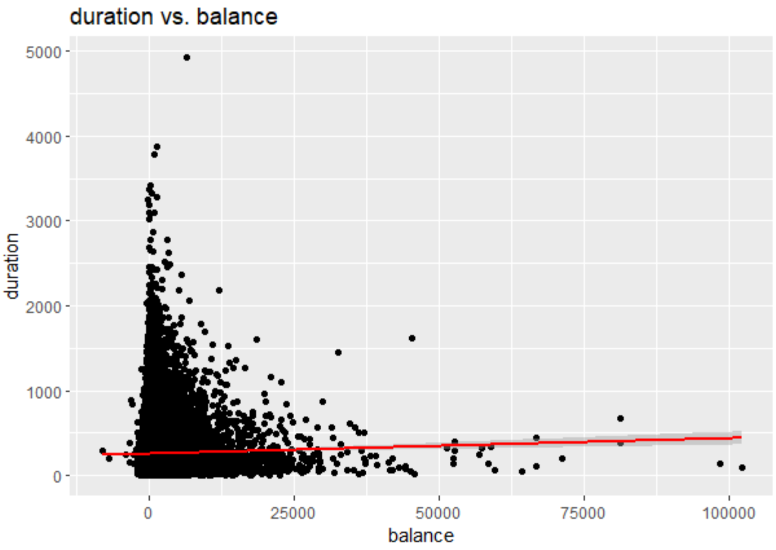
- For the residuals section, we can see that the multiple regression model has a range of -336.7 to 3625.2.
- Coefficients:
    - (Intercept): The intercept is the left over when you average the independent and dependent variable. The intercept of 274.7 is the estimated mean Y value when all Xs are zero. This would be the estimated duration for someone with campaign, balance, and previous of 0.
    - Campaign: This means that for every second the call lasts, you should expect to get a decrease amount of ~7 contacts performed during the marketing campaign.
    - Balance: As the call duration increases, the balance of the person increases by 0.001557.
    - Previous: For every second of the call, the number of contacts decrease by 0.145.
- Performance Measures:
    - Residual Standard Error: This gives us an idea of how far observed duration (y value) are from the predicted or fitted duration (the y-hats). A standard error of 254.1 is not that bad.
    - Multiple / Adjusted R-Square: The R-squared is bad for this model since we could only reach 0.7617%. Which means a variation of duration cannot be explained by our model using campaign, balance, and previous.

- F-Statistic: With a p value of 2.2e-16, our model does not seem to be doing anything.

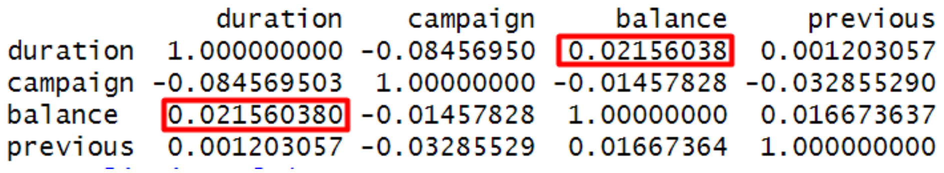
IV. Making the Regression Graph for the Multiple Regression Model

Commented [KACS1]: Looking at the table, duration and campaign has a negative correlation implying that the two variables vary in opposite directions, that is, if a variable increases the other decreases and vice versa. However, as the correlation is closer to 0 than to 1, it may also indicate that the two variables are independent, that is, as one variable increases, there is no tendency in the other variable to either decrease or increase. The same goes for the balance and previous variables, despite having a positive correlation.

This section includes the regression graph for each correlation between each independent variable (*balance*, *campaign*, and previous) and the dependent variable *duration*. The correlation between the variables is described by the graph through a regression line, which also represents the numerical correlation derived from the correlation table from the section **“II. Preparing the bank-full dataset”**.

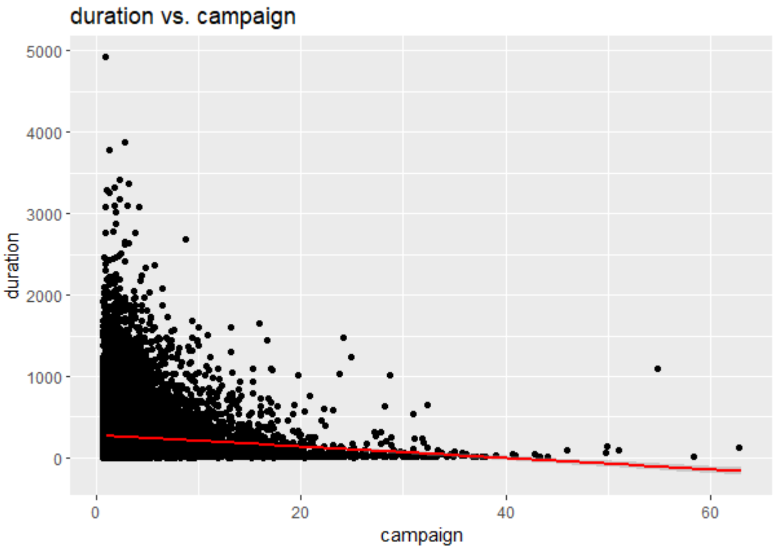


According to the correlation table, the variables *duration* and *balance* have a **correlation rate of 0.02156038**, indicating a positive relationship between the two variables.

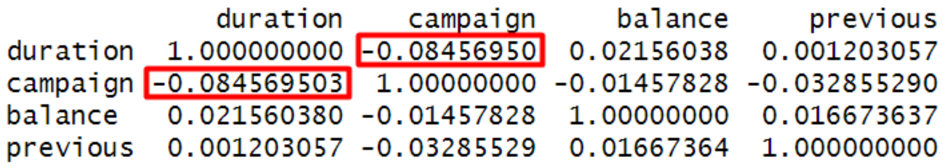


```
            duration    campaign     balance     previous
duration  1.000000000 -0.08456950  0.02156038  0.001203057
campaign -0.084569503  1.00000000 -0.01457828 -0.032855290
balance   0.021560380 -0.01457828  1.00000000  0.016673637
previous  0.001203057 -0.03285529  0.01667364  1.000000000
```

The values on the correlation table are further validated by the *duration vs. balance* graph as the regression line on the scatterplot indicates a **low positive correlation** between the two variables. This relationship is somewhat evident as the regression line slightly increases from its originating to its concluding point. Therefore, the observation that a weak positive correlation nearing 0 or 1 indicates a relationship between independent variables, where neither variable influences each other, is confirmed.

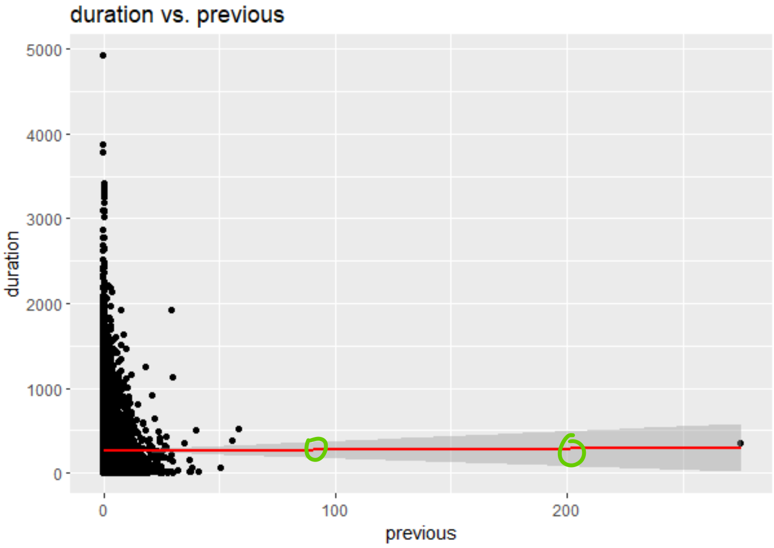


In contrast, the variables *duration* and *campaign* have a **correlation rate of -0.084569503**, indicating a negative relationship between the two variables.

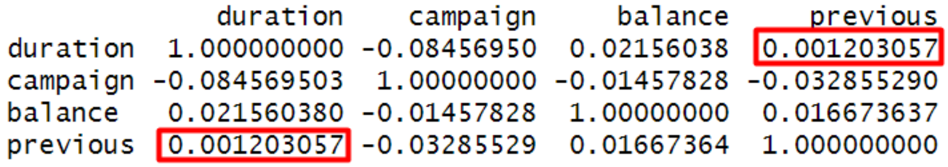


```
            duration    campaign     balance     previous
duration  1.000000000 -0.08456950  0.02156038  0.001203057
campaign -0.084569503  1.00000000 -0.01457828 -0.032855290
balance   0.021560380 -0.01457828  1.00000000  0.016673637
previous  0.001203057 -0.03285529  0.01667364  1.000000000
```

The values on the correlation table are further validated by the *duration vs. campaign* graph as the regression line on the scatterplot indicates a **low negative correlation** between the two variables. This relationship is clear as the regression line decreases from its originating to its concluding point. Furthermore, the weak negative correlation produced by the graph confirms the observation that a directly proportional relationship exists between the independent and dependent variables.



Lastly, the variables *duration* and *previous* have a **correlation rate of 0.001203057**—indicating an extremely weak positive relationship between the two variables, as evidenced by the nearing 0 correlation rate.



```
            duration    campaign     balance     previous
duration  1.000000000 -0.08456950  0.02156038  0.001203057
campaign -0.084569503  1.00000000 -0.01457828 -0.032855290
balance   0.021560380 -0.01457828  1.00000000  0.016673637
previous  0.001203057 -0.03285529  0.01667364  1.000000000
```

The values on the correlation table are further validated by the *duration vs. previous* graph as the regression line on the scatterplot portrays an **extremely low positive correlation** between the two variables. The regression line may not indicate such a correlation given that its slope is not as evident as the first regression line. This relationship is somewhat clear as the regression line increases at the two highlighted points from its originating to its concluding point. The weak positive correlation produced by the graph confirms the observation that two independent variables have no effect on each other with regards to increasing or decreasing data. Another observation as evidenced by the horizontal regression line is that a correlation between the *duration* and *previous* variables does not exist as the data neither increases nor decreases. Furthermore, this indication also shows that no correlation exists between the *duration* and *previous* variable.

REFERENCES

GeeksforGeeks. “How to Calculate Correlation Between Multiple Variables in R.” *GeeksforGeeks*, 19 Dec. 2021, www.geeksforgeeks.org/how-to-calculate-correlation-between-multiple-variables-in-r.

“How to Perform Multiple Linear Regression in R -.” *ProjectPro,* www.projectpro.io/recipes/perform-multiple-linear-regression-r.

Linear Regression Example in R Using Lm() Function – Learn by Marketing. www.learnbymarketing.com/tutorials/linear-regression-in-r.

MarinStatsLectures-R Programming & Statistics. “Multiple Linear Regression in R | R Tutorial 5.3 | MarinStatsLectures.” *YouTube*, 22 Nov. 2013, www.youtube.com/watch?v=q1RD5ECsSB0.

MarinStatsLectures-R Programming & Statistics. “Simple Linear Regression in R | R Tutorial 5.1 | MarinStatsLectures.” *YouTube*, 10 Oct. 2013, www.youtube.com/watch?v=66z_MRwtFJM.

Shipyardsdotter. “Plotting a Scatter Plot With Categorical Data.” *Posit Forum (Formerly RStudio Community)*, 11 June 2020, community.rstudio.com/t/plotting-a-scatter-plot-with-categorical-data/69456/2.

UCI Machine Learning Repository: Bank Marketing Data Set. archive.ics.uci.edu/ml/datasets/Bank+Marketing.

“What Do Correlation Coefficients Positive, Negative, and Zero Mean?” *Investopedia*, 1 June 2021, www.investopedia.com/ask/answers/032515/what-does-it-mean-if-correlation-coefficient-positive-negative-or-zero.asp.