LETTERKENNY INSTITUTE OF TECHNOLOGY

ASSIGNMENT COVER SHEET

Lecturer's Name: Dr **James Connolly**

Assessment Title: **CA – 2 DATA ANALYTICS .**

Work to be submitted to: Blackboard

Date for submission of work: **29/05/2022**

Place and time for submitting work: _____

To be completed by the Student

Student's Name: **Chakkiyath Jaisal Sabu (L00163251)**

Class: **MSC. BIG DATA ANALYITCS**

Subject/Module: Data Science

Word Count (where applicable): _____

I confirm that the work submitted has been produced solely through my own efforts.

Student's signature: **Jaisal** Date: **29/05/2022**

Notes

Penalties: The total marks available for an assessment is reduced by 15% for work submitted up to one week late. The total marks available are reduced by 30% for work up to two weeks late. Assessment work received more than two weeks late will receive a mark of zero. [Incidents of alleged plagiarism and cheating are dealt with in accordance with the Institute's Assessment Regulations.]

Plagiarism: Presenting the ideas etc. of someone else without proper acknowledgement (see section L1 paragraph 8).

Cheating: The use of unauthorised material in a test, exam etc., unauthorised access to test matter, unauthorised collusion, dishonest behaviour in respect of assessments, and deliberate plagiarism (see section L1 paragraph 8).

Continuous Assessment: For students repeating an examination, marks awarded for continuous assessment, shall normally be carried forward from the original examination to the repeat examination.

ABSTRACT:

Myocardial cardio vascular infraction also known as heart attack is one of the most severe medical condition that can cause death or permanent disability. There are several factors that can be held responsible for heart attacks. In our dataset , we have a set of variables that contribute towards the chances of having heart attack. There are around 300 observations and 14 variables. In our project, we will understand each of these variables and the relationship with the response variable, i.e. the chances of having heart attack. After examining these, we will build a predictive model that will predict patient's chances of having heart attack. Predictive modelling has been in use for a very long time. Weather, agriculture, health are some of the domains where predictive modelling has been successfully implemented.

The two efficient and commonly used predictive models are linear regression and logistic regression. We will use logistic regression in our project to build the predictive model. From eyeballing the dataset, it's found that the output variable, which defines the chances of having heart attack, is binary values. From background reading, it's clear that logistic regression is the model we can use in our model building based on our research question.

After cleaning and transforming the data, we will build two models, one with the p value significance and another one with the random forest estimator. We will be comparing these both models in the model validation section and the accuracy will also be estimated. The later we will forecast values with our model and cross checking.

RESEARCH QUESTION:

Taking into consideration the variables that affect the chances of having a heart attack and understanding the impact of each of the variable on the chances of having heart attack, build a efficient predictive model using regression technique that will predict a patients chance of having heart attack.

BUILDING PREDICTIVE MODEL:

a. Loading the dataset and preliminary analysis

As the initial step, we must load the data from a csv file and create a data frame. We use the **read.csv ()** function and create a data frame *heartattack.* The next Phase of our project is preliminary analysis of the data we have with us. We check whether the data frame is created correctly and count the number of rows and columns in our data frame. There are 303 rows and 14 columns. We also look at the schema of our data frame to get clarity on the data type of the variables present.

```
> heartattack
   age sex cp trtbps chol fbs restecg thalachh exng oldpeak slp caa thall output
1   63   1  3    145  233   1       0      150    0     2.3   0   0     1      1
2   37   1  2    130  250   0       1      187    0     3.5   0   0     2      1
3   41   0  1    130  204   0       0      172    0     1.4   2   0     2      1
4   56   1  1    120  236   0       1      178    0     0.8   2   0     2      1
5   57   0  0    120  354   0       1      163    1     0.6   2   0     2      1
6   57   1  0    140  192   0       1      148    0     0.4   1   0     1      1
7   56   0  1    140  294   0       0      153    0     1.3   1   0     2      1
8   44   1  1    120  263   0       1      173    0     0.0   2   0     3      1
9   52   1  2    172  199   1       1      162    0     0.5   2   0     3      1
10  57   1  2    150  168   0       1      174    0     1.6   2   0     2      1
11  54   1  0    140  239   0       1      160    0     1.2   2   0     2      1
12  48   0  2    130  275   0       1      139    0     0.2   2   0     2      1
13  49   1  1    130  266   0       1      171    0     0.6   2   0     2      1
14  64   1  3    110  211   0       0      144    1     1.8   1   0     2      1
```

Fig 1. heartattack data frame

```
'data.frame':   303 obs. of  14 variables:
 $ age     : int  63 37 41 56 57 57 56 44 52 57 ...
 $ sex     : int  1 1 0 1 0 1 0 1 1 1 ...
 $ cp      : int  3 2 1 1 0 0 1 1 2 2 ...
 $ trtbps  : int  145 130 130 120 120 140 140 120 172 150 ...
 $ chol    : int  233 250 204 236 354 192 294 263 199 168 ...
 $ fbs     : int  1 0 0 0 0 0 0 0 1 0 ...
 $ restecg : int  0 1 0 1 1 1 0 1 1 1 ...
 $ thalachh: int  150 187 172 178 163 148 153 173 162 174 ...
 $ exng    : int  0 0 0 0 1 0 0 0 0 0 ...
 $ oldpeak : num  2.3 3.5 1.4 0.8 0.6 0.4 1.3 0 0.5 1.6 ...
 $ slp     : int  0 0 2 2 2 1 1 2 2 2 ...
 $ caa     : int  0 0 0 0 0 0 0 0 0 0 ...
 $ thall   : int  1 2 2 2 2 1 2 3 3 2 ...
 $ output  : int  1 1 1 1 1 1 1 1 1 1 ...
```

Fig 2. Schema of heartattack data frame

We also identify the type of variables as categorical and numerical. We find out the unique values in each of the variable to determine the categorical and numerical variables. The variables with less unique values will be categorical and the ones with higher unique values will be numerical. From our analysis we find out that ***sex, cp, fbs, rest_ecg, exang, slope, ca, thal,*** and ***target*** are the categorical variables and the ***age, trtbps, chol, thalach*** *and* ***oldpeak*** are the numerical variables.

```
> colnames(heartattack)
 [1] "age"      "sex"      "cp"       "trtbps"   "chol"     "fbs"      "restecg"  "thalachh"
 [9] "exng"     "oldpeak"  "slp"      "caa"      "thall"    "output"
> #understanding the type of values
> sapply(heartattack, class)
      age       sex        cp    trtbps      chol       fbs   restecg  thalachh      exng
"integer" "integer" "integer" "integer" "integer" "integer" "integer" "integer" "integer"
  oldpeak       slp       caa     thall    output
"numeric" "integer" "integer" "integer" "integer"
>
```

Fig 3. Column names and the type of values stored

b. Analysis and Data Transformation

We try find out the Na Values in our data set and take the necessary actions for preparing the data for our analysis. We use ***is.na ()*** function for this purpose. In our Initial analysis we find out that Na values are not present in our dataset. The data types in our dataset is not always correct, so we will split the data set into numerical and categorical values and cross check them to combine them to a new data frame called ***heartattack_prediction***. We will use the ***cbind*** function to combine the categorical and numerical data.

The very next step is to identify the missing values, complete cases and incomplete cases. We use a package called DataExplorer. Using this package will provide us a graphical visualization of the details.

From the plot we can understand that 64% of our columns are discrete, 36% of columns are continuous. We don't have any incomplete cases in our dataset. From the plot we can also infer that there are no missing observations.

We also use the ***summary()*** command to understand our variables better , this will give us the information regarding count of each value in a categorical column and he minimum, 1st quartile, median, mean, 3rd quartile and the maximum for all numeric variables. This is clearly shown in Fig 5.

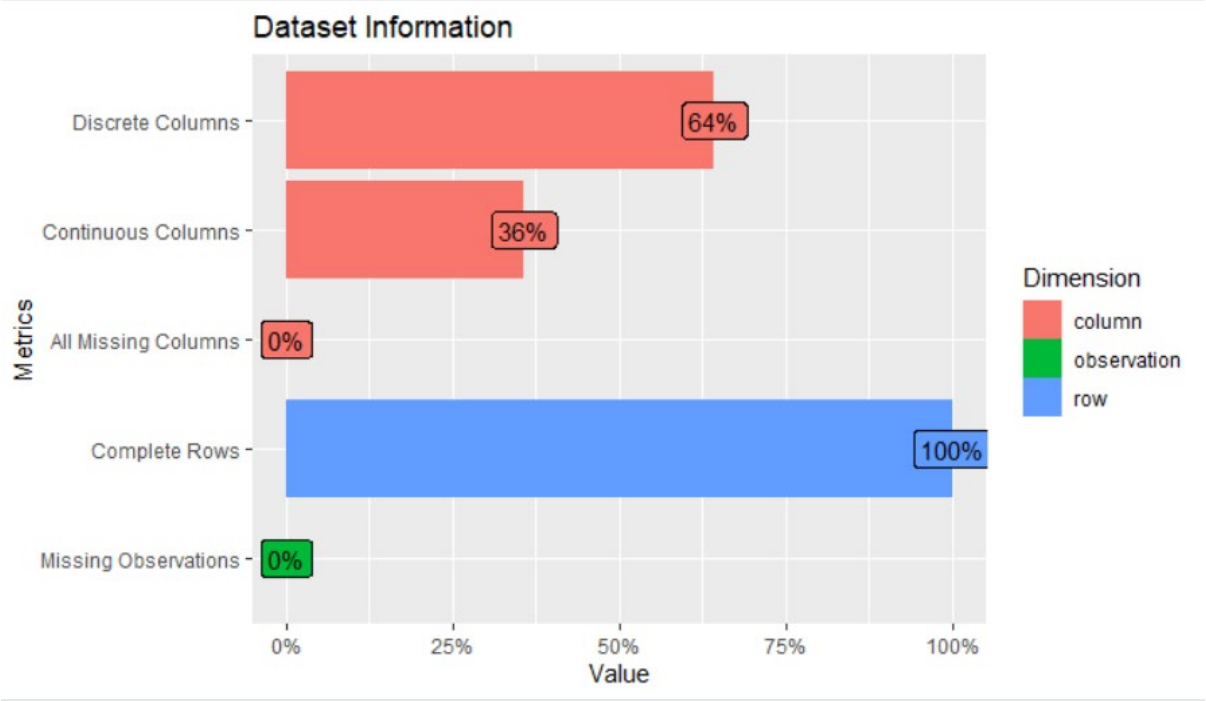

Fig 4. DataExplorer Plot

```
> summary(heartattack_prediction)
 sex     cp       fbs     restecg exng    slp     caa     thall   output       age
 0: 96   0:143   0:258   0:147   0:204   0: 21   0:175   0:  2   0:138   Min.   :29.00
 1:207   1: 50   1: 45   1:152   1: 99   1:140   1: 65   1: 18   1:165   1st Qu.:47.50
         2: 87           2:  4           2:142   2: 38   2:166           Median :55.00
         3: 23                                   3: 20   3:117           Mean   :54.37
                                                 4:  5                   3rd Qu.:61.00
                                                                         Max.   :77.00

    trtbps           chol         thalachh       oldpeak
 Min.   : 94.0   Min.   :126.0   Min.   : 71.0   Min.   :0.00
 1st Qu.:120.0   1st Qu.:211.0   1st Qu.:133.5   1st Qu.:0.00
 Median :130.0   Median :240.0   Median :153.0   Median :0.80
 Mean   :131.6   Mean   :246.3   Mean   :149.6   Mean   :1.04
 3rd Qu.:140.0   3rd Qu.:274.5   3rd Qu.:166.0   3rd Qu.:1.60
 Max.   :200.0   Max.   :564.0   Max.   :202.0   Max.   :6.20
```

Fig 5. Summary of heartattack_prediction dataframe.

c. Descriptive Statistics

Descriptive statistics techniques are very helpful in understanding the data. Here we will be using the ***stat.desc()*** function from the ***pastecs*** package. This will provide us detailed information about the numerical variables in our dataframe. The output of ***stat.desc()*** function is clearly shown in the Fig 6. From the function we will be viewing a wide variety of statistical descriptions from our dataset like kurtosis, skewness, mean, median, standard deviation etc. All these metrics are quite useful for the in-depth understanding of the dataset. By using this descriptive statistical method, we infer that our dataset is highly skewed. The nba.nr value also proves the absence of NA values in our dataset. The ***stat.desc()*** also provides information about the normality of variables in our dataset.

	chol	thalachh	oldpeak
nbr.val	3.030000e+02	3.030000e+02	3.030000e+02
nbr.null	0.000000e+00	0.000000e+00	9.900000e+01
nbr.na	0.000000e+00	0.000000e+00	0.000000e+00
min	1.260000e+02	7.100000e+01	0.000000e+00
max	5.640000e+02	2.020000e+02	6.200000e+00
range	4.380000e+02	1.310000e+02	6.200000e+00
sum	7.461800e+04	4.534300e+04	3.150000e+02
median	2.400000e+02	1.530000e+02	8.000000e-01
mean	2.462640e+02	1.496469e+02	1.039604e+00
SE.mean	2.977599e+00	1.315867e+00	6.670202e-02
CI.mean	5.859469e+00	2.589429e+00	1.312596e-01
var	2.686427e+03	5.246464e+02	1.348095e+00
std.dev	5.183075e+01	2.290516e+01	1.161075e+00
coef.var	2.104682e-01	1.530614e-01	1.116844e+00
skewness	1.132105e+00	-5.321005e-01	1.257176e+00
skew.2SE	4.042379e+00	-1.899958e+00	4.488968e+00
kurtosis	4.362841e+00	-9.992646e-02	1.500340e+00
kurt.2SE	7.814217e+00	-1.789767e-01	2.687235e+00
normtest.W	9.468815e-01	9.763154e-01	8.441834e-01
normtest.p	5.364848e-09	6.620819e-05	8.183378e-17

>

Fig 6. Output of stat.desc() function.

As a next step, we will be implementing a pair plot

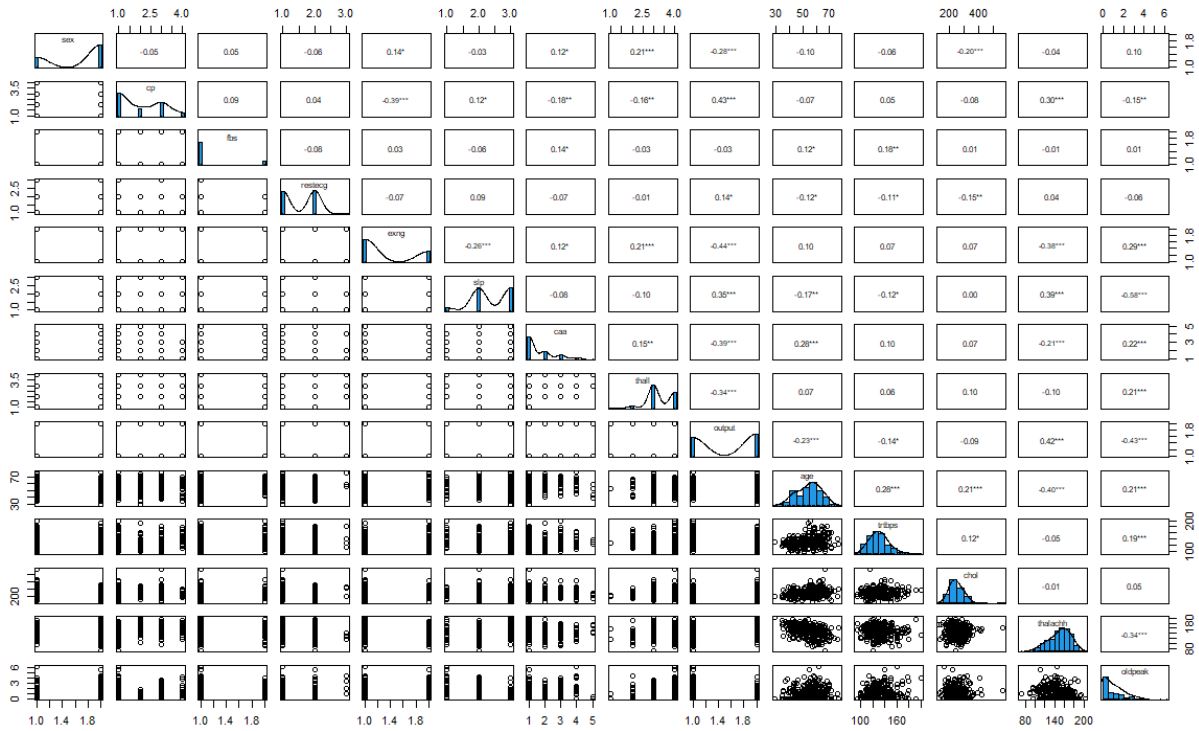

Fig 7. Pair plot for heartattack_prediction.

We will also implement a correlation plot to understand the correlation between each of the variables.

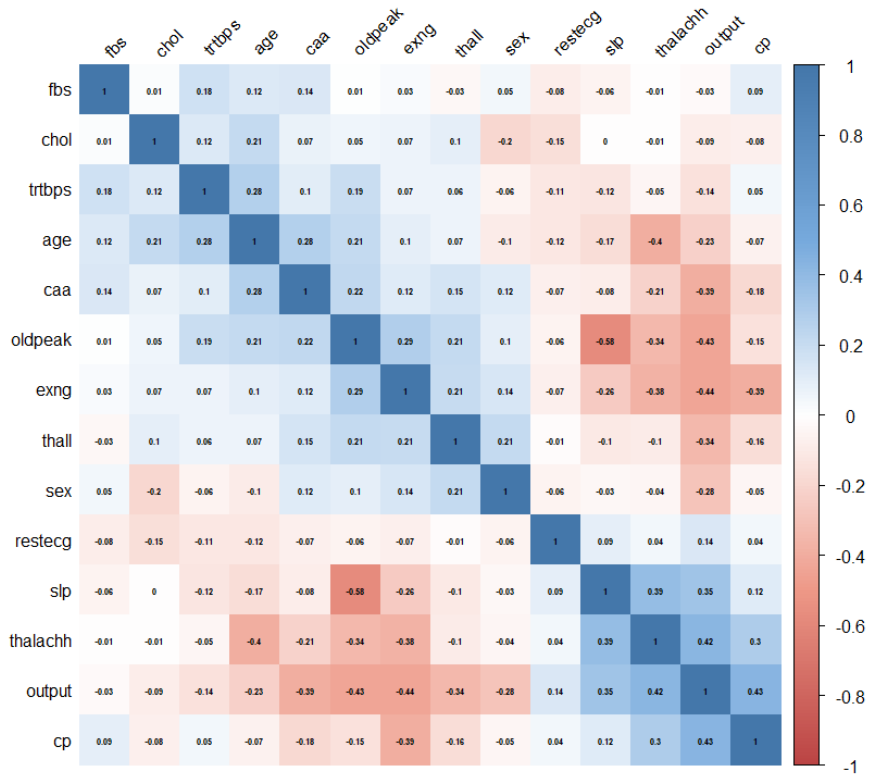

Fig 8. Correlation plot for heartattack_prediction

From our Correlation plot, we draw some inferences about the variable. Taking the age variable into consideration, thalachh variable has the highest correlation with age variable. The severity is -0.4, which indicates a inverse correlation between the variables. Next we take a look into chol variable. The chol variable has the highest correlation is age. There is very low positive correlation between the variables.

Next, we take the trtbps. Similar to the chol variable, trtbps also has a low positive correlation with the age variable. The oldpeak variable has a correlation with slope and target variable. The sex variable has no significant correlation with any of the other variables. The thalachh variable has a moderate positive correlation. We can say that the maximum heart rate achieved can trigger a heart attack. There is a direct correlation between the chest pain variable and the target variable too. The fasting blood pressure and resting echocardiogram variables does not have any significant correlation with the other variables.

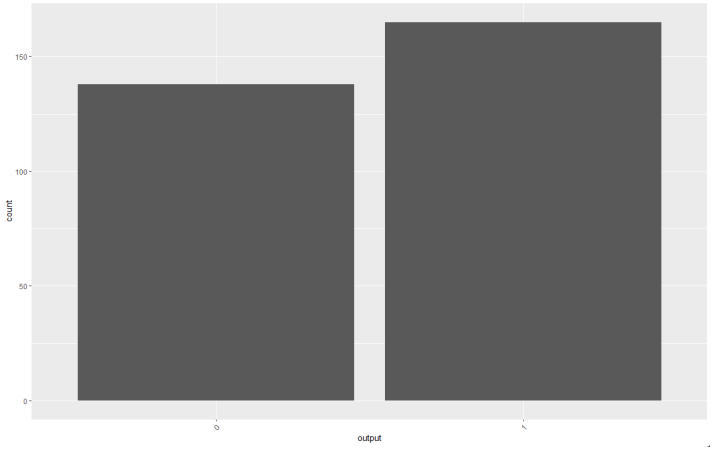

Fig 9. Barplot for output variable.

We will also take a look at the balance of the output variable with the help of a bar plot.

From the bar plot we can infer that around 138 people have the no chances of a heart attack while 165 people from our observation have the risk of heart attack. Around 54% of people have the chances of heart attack in our dataset.

d. Outlier detection and remedial measures.

There are several reasons for the presence of outliers in data. It ranges from human errors to sampling errors. The key to building a exceptional model is the detection and treatment of these outliers. We also need to check our data for outliers before we build our model. There are several methods to detect outliers within the data like Scatter Plot and Box Plot. We will be using Box plot to detect the presence of outliers. After conducting the box plot test for all the numerical variables, we find out that the age variable does not have the presence of outliers.

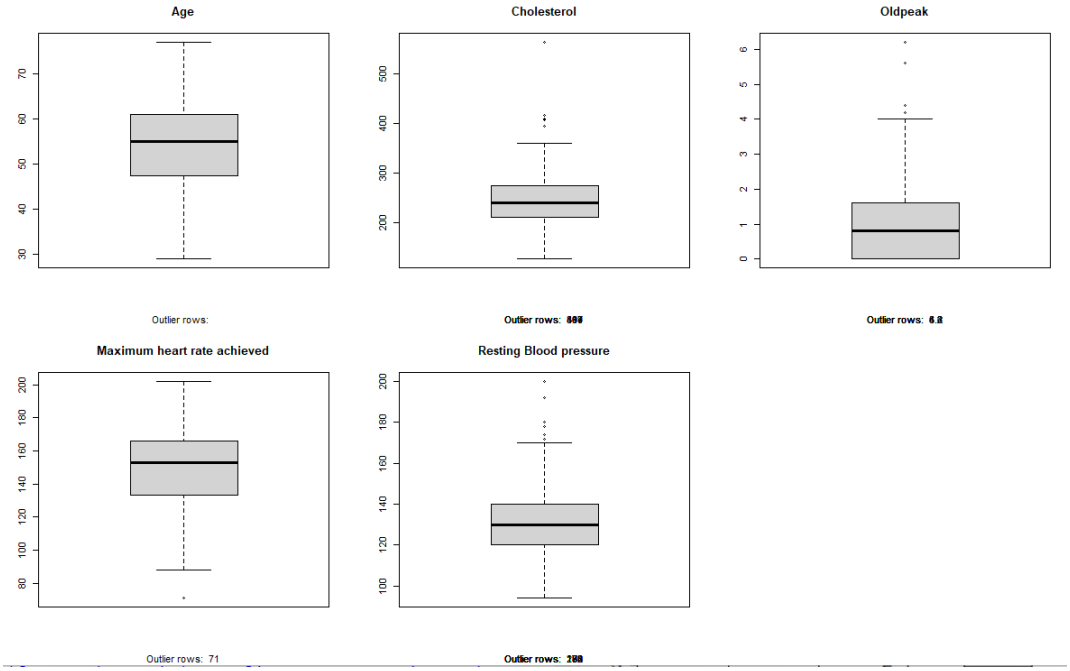

Fig 10. Boxplot for numerical variables to detect the outliers.

The Cholesterol, oldpeak , maximum heart rate and resting blood pressure have the presence of outliers. In our project, we will remove these outliers as a remedial measure and then use the box plot again to confirm the deletion of these outlier values from these variables.

```
> #age
> boxplot.stats(heartattack_prediction$age)$out
integer(0)
> #cholesterol
> boxplot.stats(heartattack_prediction$chol)$out
[1] 417 564 394 407 409
> #oldpeak
> boxplot.stats(heartattack_prediction$oldpeak)$out
[1] 4.2 6.2 5.6 4.2 4.4
> #Maximum heart rate achieved(thalachh)
> boxplot.stats(heartattack_prediction$thalachh)$out
[1] 71
> #Resting blood Pressure(trtbps)
> boxplot.stats(heartattack_prediction$trtbps)$out
[1] 172 178 180 180 200 174 192 178 180
```

Fig 11. Outliers found with the Boxplot.

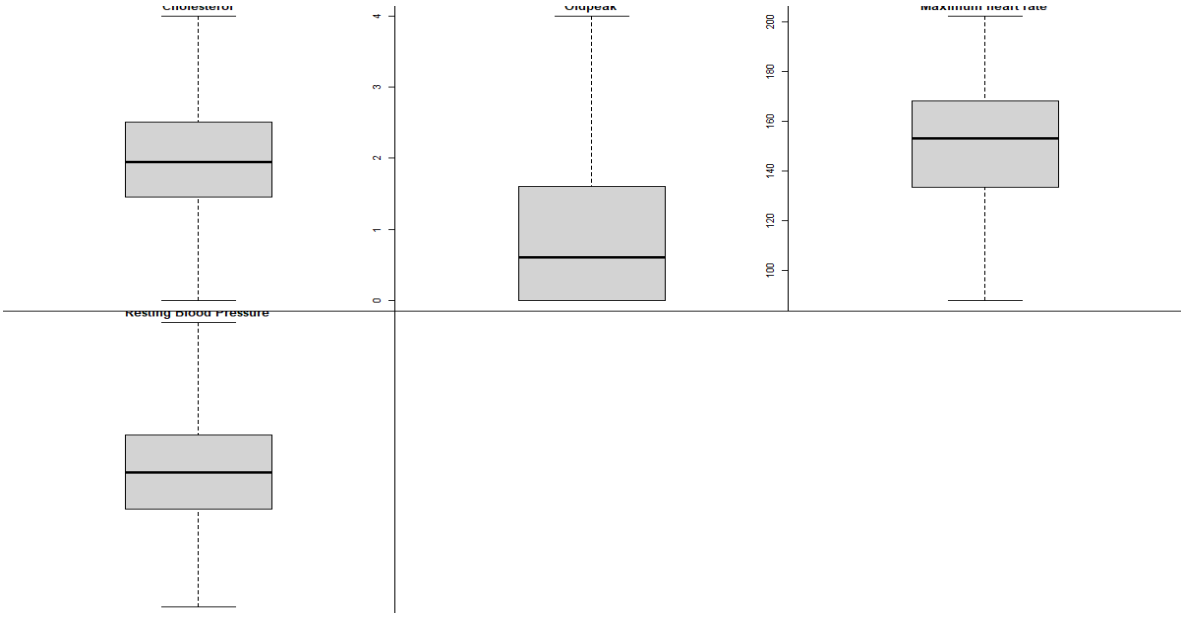

Fig 12. Boxplot after deleting Outliers.

e. Building the model

Predictive modelling is one of the most important advancements of statistics. It helps us to forecast and estimate the metrics that are impractical to measure. For many years, scientists and researchers have been using predictive modelling to predict weather patterns, crop yields, economic growth and many other metrics. Regression models are the functions that describe the relationship between a single independent variable or multiple independent variables and a response/target variable. There are multiple types of regression like linear, logistic and multiple regression models. Logistic regression is mainly used for predictive analysis. The response or target variable is a probability of the occurrence. Since the response variable is a probability, the values will range between 0 and 1. Linear regression and logistic regression are very popular predictive models, but there are significant differences. Linear regression is used to estimate the relationship between a independent variable and individual or multiple dependent variable, where the independent variable is continuous. We apply logistic regression in cases where we have our response variable as binary. Taking our data into consideration, we have selected the output (chances of having heart attack) as our target variable. The values are either 0s or 1s. So we will be using Logistic regression in building our predictive model based on our research question. *Wikipedia (2022)*

To better understand the data , we will be changing the column names before we build our model. The new column names are provided in the Fig13.

```
> names(heartattack_prediction)
 [1] "Gender"                "Chest_pain"            "Fasting_blood_sugar"
 [4] "Resting_ecocardiograph" "agina_exercise"        "slp"
 [7] "number_of_vessels"     "thall"                 "output"
[10] "age"                   "Resting_blood_pressure" "Cholesterol"
[13] "Maximum_heart_rate"    "oldpeak"
>
```

Fig 13. Changed Column names.

Before we build our model, we have to split our dataset to test and train sets. For this we set a seed using the ***set.seed()*** . By using this, we get the same numbers every time we split the dataset. Then we split our dataset into ***train_set*** and ***test_set.*** To use the split function, we will be using the ***caTools*** package with ***sample.split()***

function. We split the data into 80:20 ratio. We will use 80% percent of our data to train our logistic regression model and the rest 20 to forecast and check the values. We build our logistic regression model with all the variables present and the take a look at the results. After we successfully build our model, we obtain the summary of our model.

From the summary of our model, we understand that Gender, Chest Pain, Number of vessels, Cholesterol and Old peak are the significant factors in determining the chances of heart attack. Maximum heart rate is also a factors but it's not as significant as the other variables.

Results of the logistic regression model are given in Fig 14.

```
Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.913  -0.263   0.112   0.444   3.040

Coefficients:
                        Estimate Std. Error z value Pr(>|z|)
(Intercept)             1.29e+01   1.46e+03    0.01  0.99295
Gender1                -1.42e+00   6.88e-01   -2.06  0.03958 *
Chest_pain1             6.77e-01   6.62e-01    1.02  0.30608
Chest_pain2             1.50e+00   6.15e-01    2.44  0.01467 *
Chest_pain3             2.04e+00   8.11e-01    2.51  0.01204 *
Fasting_blood_sugar1    6.34e-01   6.99e-01    0.91  0.36470
Resting_ecocardiograph1 8.46e-01   4.90e-01    1.73  0.08445 .
Resting_ecocardiograph2 6.79e-01   4.09e+00    0.17  0.86817
agina_exercise1        -3.51e-01   5.29e-01   -0.66  0.50678
slp1                    4.35e-01   1.02e+00    0.43  0.66905
slp2                    1.65e+00   1.12e+00    1.47  0.14192
number_of_vessels1     -2.47e+00   6.45e-01   -3.83  0.00013 ***
number_of_vessels2     -3.15e+00   9.43e-01   -3.34  0.00084 ***
number_of_vessels3     -2.18e+00   1.00e+00   -2.17  0.02976 *
number_of_vessels4      7.93e-01   1.72e+00    0.46  0.64480
thall1                 -1.15e+01   1.46e+03   -0.01  0.99371
thall2                 -1.17e+01   1.46e+03   -0.01  0.99360
thall3                 -1.34e+01   1.46e+03   -0.01  0.99264
age                     3.32e-02   2.94e-02    1.13  0.25995
Resting_blood_pressure -2.08e-02   1.60e-02   -1.30  0.19212
Cholesterol            -1.29e-02   6.21e-03   -2.07  0.03841 *
Maximum_heart_rate      2.82e-02   1.49e-02    1.89  0.05942 .
oldpeak                -6.65e-01   3.05e-01   -2.18  0.02938 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 311.47  on 226  degrees of freedom
Residual deviance: 131.49  on 204  degrees of freedom
AIC: 177.5

Number of Fisher Scoring iterations: 14
```

Fig 14. Results of model ran with all factors.

f. Building Models based on Feature selection

We will be building a model first without splitting the dataset into train and test. This model will be used to identify the influential variables in the dataset. Then we will be selecting those variables and then building a model based on those variables. We will take a deeper look into the results of model. Similarly, a model that is based on the variables predicted from the random forest estimator will also be used. We will compare the accuracy and other metrics of these models *Pretorius, Arnu & Bierman, Surette & Steel, Sarel. (2016).*

MODEL BUILT ON BASIS OF VARIABLES SELECTED WITH P-VALUE

We build a model called ***model1*** , we select the significant variables from the previous model we built. We will be using Gender, Chest_pain, Resting_ecocardiograph, number_of_vessels, Cholesterol, Maximum_heart_raate and oldpeak.

MODEL BUILT ON BASIS OF VARIABLES FROM RFE

We will use random forest estimator to predict the variables that will be used for building the model. The random forest estimator provided us with number_of_vessels, Chest_pain, agina_exercise, oldpeak, Gender, thall and Maximum_heart_rate. We build a model with these variables and the results will be discussed in the model validation section of the report *Abbas, Ali. (2012).*

The summary of the models can be seen in Fig.15 and 16

```
glm(formula = output ~ Gender + Chest_pain + Resting_ecocardiograph +
    number_of_vessels + Cholesterol + Maximum_heart_rate + oldpeak,
    family = binomial, data = train_set)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.411  -0.416   0.173   0.526   2.534

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)             0.38334    1.97988    0.19  0.84647
Gender1                -1.90581    0.54408   -3.50  0.00046 ***
Chest_pain1             1.07928    0.61211    1.76  0.07786 .
Chest_pain2             1.76779    0.51442    3.44  0.00059 ***
Chest_pain3             2.05665    0.70376    2.92  0.00347 **
Resting_ecocardiograph1 0.49668    0.42020    1.18  0.23721
Resting_ecocardiograph2 -0.32037   2.30515   -0.14  0.88947
number_of_vessels1      -1.83730   0.54359   -3.38  0.00072 ***
number_of_vessels2      -2.27261   0.71340   -3.19  0.00144 **
number_of_vessels3      -1.74745   0.83233   -2.10  0.03578 *
number_of_vessels4      -0.51384   1.45994   -0.35  0.72487
Cholesterol             -0.01477   0.00541   -2.73  0.00637 **
Maximum_heart_rate       0.03587   0.01131    3.17  0.00152 **
oldpeak                 -0.95010   0.23835   -3.99  6.7e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 311.47  on 226  degrees of freedom
Residual deviance: 155.25  on 213  degrees of freedom
AIC: 183.3

Number of Fisher Scoring iterations: 6
```

```
glm(formula = output ~ number_of_vessels + Chest_pain + agina_exercise +
    oldpeak + Gender + thall + Maximum_heart_rate, family = "binomial",
    data = train_set)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.625  -0.365   0.181   0.491   2.725

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)          11.3258  1455.3982    0.01  0.99379
number_of_vessels1   -1.9716     0.5621   -3.51  0.00045 ***
number_of_vessels2   -2.3718     0.7594   -3.12  0.00179 **
number_of_vessels3   -1.8802     0.8688   -2.16  0.03045 *
number_of_vessels4    0.3371     1.4491    0.23  0.81603
Chest_pain1           0.6952     0.6109    1.14  0.25517
Chest_pain2           1.5042     0.5562    2.70  0.00684 **
Chest_pain3           1.4920     0.7131    2.09  0.03642 *
agina_exercise1      -0.5278     0.4809   -1.10  0.27235
oldpeak              -0.8286     0.2573   -3.22  0.00128 **
Gender1              -1.0350     0.5776   -1.79  0.07316 .
thall1              -12.1347  1455.3980   -0.01  0.99335
thall2              -12.1405  1455.3978   -0.01  0.99334
thall3              -13.8119  1455.3978   -0.01  0.99243
Maximum_heart_rate    0.0238     0.0118    2.01  0.04403 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 311.47  on 226  degrees of freedom
Residual deviance: 149.79  on 212  degrees of freedom
AIC: 179.8

Number of Fisher Scoring iterations: 14
```

Fig 15. Summary of model1 and rfe.model.

MODEL VALIDATION:

As we have successfully built out logistic regression model and produced the summary of model, we will validate our model. For validation, we have to split our data into train and test sets. For splitting, we will be using the ***sample.split()*** function from the ***caTools*** package. We will split our dataset into 8:2. 80 percent of data will be used for training our logistic regression model and 20 percent of our data will be used as test data set. The reason why we split the dataset into 80:20 is because we will have a major share of data to train and this will improve the model accuracy. The two models we had built were ***model1*** and ***rfe.model.*** First we take into consideration the summary of models.

The first part of output reminds us of the choices we made while building the model and the variables we have selected to build our model. The second part of summary is the deviance residuals; it is a measure of fit for the model we have built. In the next part, we get the coefficients with estimate, standard error, z value and the p value. From these values we decide the significance of each variable and then we select the variables to use in our model. We have selected the variables using this method for our ***model1***. In the next part we have the null deviance and the residual deviance. We also get information regarding the AIC value. AIC Score is a single number score that determines which model is a best fit. Usually a lower AIC score indicates a better model. From both our models built using the variables predicted from random forest estimators is a better model than the one predicted from p values.

Now we move on to the pR2 of each model we have built and then we will use the test and train to produce accuracy metric for our models. The accuracy of our model is tested using the test data set that we have split from our original dataset. We will use our model to predict the chances of heart attack for the parameters in the test dataset and then we will compare that against the observed value from the dataset. The mean of predicted and observed value will provide us with the accuracy of our model. The ***model1*** has an accuracy of 75.4% and our ***rfe.model*** has an accuracy of 84.2%. From this we can infer that the model built using the random forest estimator variables has a better accuracy in predicting the chances of having heart attack

MODEL FORECASTING:

As the first step in forecasting, we will predict the chances of having heart attack for test data set. Since we found out that the ***rfe.model*** is better in predicting the chances, we will be using this model. The values we predict are stored in predicted variable. We will find out the optimal probability cutoff using the ***OptimalCutoff*** function. We get a value of 0.9899969 as the cutoff. This means that any value above the optimal cutoff will be predicted to the default and the values below will not be predicted to the cutoff. Using this cutoff we will create a confusion matrix. This confusion matrix will show our predictions against the actual defaults. We also find the misclassification error rate for our model, which turns out to be 1.75%. This error rate proves that our model can predict the output variable with much efficiency.

```
> confusionMatrix(test_set$output, predicted)
   0
0 22
1 35
```

Fig 16. Confusion matrix.

CONCLUSION:

The main aim of our project is to understand the variables that contribute towards predicting the chances of having heart attack for a person. Using these variables , we have to build a predictive model. Taking a look into the data we have, major share of variables are categorical in nature including the output variable. It shows the chances of a patient having heart attack. The values are in 0s and 1s, i.e. Binary values as output. In such cases we have to use logistic regression in building our model. First, we will check the variables and the data present in them to confirm that we have numerical values. If we have any categorical values , we will factorize them. In the next stage , we will find the data about our data. Complete cases, incomplete cases, discerte and continuous columns. A wide variety of descriptive statistical methods can be used to understand the data. We will be using the pair plot, correlation plot and several other techniques to gain insights about our data.

After we finish with the analysis section, the next major step is outlier detection and the measures to tackle outliers. There are several ways to detect outliers, Boxplot and Scatterplot are the ways used to detect outliers. We will be using a boxplot analysis to detect the outliers. Except from the age variable we get outliers. To tackle the outliers we have delete those outliers. After deleting, we will again check the presence of output variables. Next we build a logistic regression model on the dataset and from the results we will find the variables that are significant and using that we will be building a model with those variables. We will also use the random forest estimators to find the variables and then we will also build a model with them. For the model1 we get a accuracy of 75% and the model we built with random forest estimators have a accuracy of 84%. In the model forecasting section , we will split the data into train ad test. We will train our model with the train set and the run the model on our test set to predict values. This is how we predict the accuracy for our models. A wide variety of tests like collinearity test, normality test, variable importance, likelyhood ratio test have been conducted throughout model at various stages in our project. Attached with the report is the Github link to CA2.R file with a detailed explanation each and every line of code.

REFERENCES:

Wikipedia. (2022). *Logistic_regression.* Available: https://en.wikipedia.org/wiki/Logistic_regression. Last accessed 29th May 2022.

Abbas, Ali. (2012). Using Logistic Regression Model to predict the models with economic categorical dependent variables. 2. 234-253.

Pretorius, Arnu & Bierman, Surette & Steel, Sarel. (2016). A meta-analysis of research in random forests for classification. 1-6. 10.1109/RoboMech.2016.7813171.