INTRODUCTION

This report includes a diagnosis that shows how to use RStudio to fit, analyse, and assess a regression model.

This assignment module's learning outcomes are as follows:

• Using standard functions and diagnostic procedures, fit, analyze, and assess regression models.

• Address over-fitting, linearity, multicollinearity, and outliers concerns.

• Using automated algorithms, selecting the best fit model from several predictors.

A little history about correlation and regression

According to an analysis of Sir Francis Galton's and Karl Pearson's work, Galton's study on hereditary traits of sweet peas influenced the development of linear regression. Following Galton and Pearson's work, the more general methods of multiple regression and the product-moment correlation coefficient were developed. Before addressing prediction issues and the use of linear regression, most modern textbooks show and explain correlation. This study gives a short overview of how Galton developed and used linear regression to issues of heredity in the first place. Additional techniques that teachers might use to impart basic linear regression to students are shown in this history (Taylor & Francis., 2017).

Multiple regression and its application in different industries.

Multiple regression, also known as multiple linear regression, is a statistical technique that uses two or more explanatory variables to predict the outcome of a response variable. (Indeed., 2020) It can be used to analyze the relationship between a single dependent variable and several independent variables. The objective of multiple regression analysis is to use the independent variable whose values are known to predict the value of the single dependent value (Indeed Career Guide., 2020).

Multiple regression can be used in a variety of industries for instance, gaming industry.

difficulty, performance issue, release data, cost etc. Researchers can conduct a study to predict how a change in these variables will affect the game's whole performance.

Hypothesis testing with respect to regression analysis and its application in a industry.

Hypothesis tests are statistical procedures for putting a claim or assumption about a population's underlying distribution to the test using sample data. Following are the ways to do hypothesis testing in linear regression (Kumar A., 2022):

Formulating the null and alternate hypotheses is the first phase in hypothesis testing. Based on that we get the H0 and Ha values which are used to conduct hypothesis testing. Then the linear regression formula values are calculated which gives the equation y = a + b1x1 + b2x2 + … (Depending on linear or multiple regression model). T test and F test values are then calculated by using their respective formula to get the values for linear or multiple regression. Then these test values are used to run the hypothesis test with a critical value which gives us information if we can reject the null hypothesis or we fail to reject the null hypothesis (Banerji A., 2022).

For instance we can use hypothesis testing in gaming industry as well. Imagine you are a game developer who wants to create a game which is successful and which hold its player base. You'll try to sell that game for maximum profits but multiple factors can affect that price range. These variables could be the game's performance, media coverage, not enough publicity, etc. We can build a prediction model for these independent variables to access the maximum sales you can achieve. We can then use hypothesis testing to find the correlation between these variable and then make changes based on the results we obtain.

ANALYSIS SECTION

Libraries Used

```
library(psych)
library(readxl)
library(tidyverse)
library(dplyr)
library(RColorBrewer)
library(knitr)
library(ggplot2)
library(DT)
library(corrplot)
library(dlookr)
library(car)
library(leaps)
```

First task

Loading the data set.

```
## Rows: 2930 Columns: 82
```

```
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (45): PID, MS SubClass, MS Zoning, Street, Alley, Lot Shape, Land Contou...
## dbl (37): Order, Lot Frontage, Lot Area, Overall Qual, Overall Cond, Year Bu...
```

The data Ames Housing was imported and loaded in this task where it was stored under the object name Proj1_DS. This data set contains 82 fields(columns) for 2,930 properties values(rows) in Ames IA.

Second task

Perform Exploratory and descriptive analysis on the provided data set.

```
## # A tibble: 6 x 82
##   Order PID     `MS SubClass` `MS Zoning` `Lot Frontage` `Lot Area` Street Alley
##   <dbl> <chr>   <chr>         <chr>               <dbl>      <dbl> <chr>  <chr>
## 1     1 052630~ 020           RL                    141      31770 Pave   <NA>
## 2     2 052635~ 020           RH                     80      11622 Pave   <NA>
## 3     3 052635~ 020           RL                     81      14267 Pave   <NA>
## 4     4 052635~ 020           RL                     93      11160 Pave   <NA>
## 5     5 052710~ 060           RL                     74      13830 Pave   <NA>
## 6     6 052710~ 060           RL                     78       9978 Pave   <NA>
## # ... with 74 more variables: Lot Shape <chr>, Land Contour <chr>,
## #   Utilities <chr>, Lot Config <chr>, Land Slope <chr>, Neighborhood <chr>,
## #   Condition 1 <chr>, Condition 2 <chr>, Bldg Type <chr>, House Style <chr>,
## #   Overall Qual <dbl>, Overall Cond <dbl>, Year Built <dbl>,
## #   Year Remod/Add <dbl>, Roof Style <chr>, Roof Matl <chr>,
## #   Exterior 1st <chr>, Exterior 2nd <chr>, Mas Vnr Type <chr>,
## #   Mas Vnr Area <dbl>, Exter Qual <chr>, Exter Cond <chr>, ...
```

```
## # A tibble: 37 x 26
##    described_variables     n    na   mean     sd se_mean   IQR skewness kurtosis
##    <chr>               <int> <int>  <dbl>  <dbl>   <dbl> <dbl>    <dbl>    <dbl>
##  1 Order                2930     0 1.47e3 8.46e2 1.56e+1 1464.    0      -1.2
##  2 Lot Frontage         2440   490 6.92e1 2.34e1 4.73e-1   22     1.50   11.2
##  3 Lot Area             2930     0 1.01e4 7.88e3 1.46e+2 4115    12.8   265.
##  4 Overall Qual         2930     0 6.09e0 1.41e0 2.61e-2    2     0.191   0.0524
##  5 Overall Cond         2930     0 5.56e0 1.11e0 2.05e-2    1     0.574   1.49
##  6 Year Built           2930     0 1.97e3 3.02e1 5.59e-1   47    -0.604  -0.502
##  7 Year Remod/Add       2930     0 1.98e3 2.09e1 3.85e-1   39    -0.452  -1.34
##  8 Mas Vnr Area         2907    23 1.02e2 1.79e2 3.32e+0  164     2.61    9.29
##  9 BsmtFin SF 1         2929     1 4.43e2 4.56e2 8.42e+0  734     1.42    6.86
## 10 BsmtFin SF 2         2929     1 4.97e1 1.69e2 3.13e+0    0     4.14   18.8
## # ... with 27 more rows, and 17 more variables: p00 <dbl>, p01 <dbl>,
## #   p05 <dbl>, p10 <dbl>, p20 <dbl>, p25 <dbl>, p30 <dbl>, p40 <dbl>,
## #   p50 <dbl>, p60 <dbl>, p70 <dbl>, p75 <dbl>, p80 <dbl>, p90 <dbl>,
## #   p95 <dbl>, p99 <dbl>, p100 <dbl>
```

The head function was used to collect the first six values of the complete data set for exploratory analysis to see how the Ames Housing data set looks. The full data set is next subjected to basic descriptive statistics. The describe function from the dlookr package was used for this since it only chooses and presents descriptive statistics for numerical values and ignores category variables. We can see that the values for some of the column fields are different in some of the observations, indicating that the data set includes missing values. Additionally, some of the skewness values are positive, indicating that the normal distribution graph is positively skewed because the mean is greater than the median, while some are negative, indicating that the mean is less than the median and thus the normal distribution graph will be negatively skewed.

Third task

Prepare data set by inputting missing values in the data set.

```
#Inputting missing values with means values of those columns.

Proj1_DS$`Lot Frontage`[is.na(Proj1_DS$`Lot Frontage`)] = mean(Proj1_DS$`Lot Frontage`, na.rm = TRUE)

Proj1_DS$`Garage Cars`[is.na(Proj1_DS$`Garage Cars`)] = mean(Proj1_DS$`Garage Cars`, na.rm = TRUE)

Proj1_DS$`Garage Area`[is.na(Proj1_DS$`Garage Area`)] = mean(Proj1_DS$`Garage Area`, na.rm = TRUE)

Proj1_DS$`Bsmt Half Bath`[is.na(Proj1_DS$`Bsmt Half Bath`)] = mean(Proj1_DS$`Bsmt Half Bath`, na.rm = TRUE)

Proj1_DS$`Bsmt Full Bath`[is.na(Proj1_DS$`Bsmt Full Bath`)] = mean(Proj1_DS$`Bsmt Full Bath`, na.rm = TRUE)

Proj1_DS$`Total Bsmt SF`[is.na(Proj1_DS$`Total Bsmt SF`)] = mean(Proj1_DS$`Total Bsmt SF`, na.rm = TRUE)

Proj1_DS$`Bsmt Unf SF`[is.na(Proj1_DS$`Bsmt Unf SF`)] = mean(Proj1_DS$`Bsmt Unf SF`, na.rm = TRUE)

Proj1_DS$`BsmtFin SF 1`[is.na(Proj1_DS$`BsmtFin SF 1`)] = mean(Proj1_DS$`BsmtFin SF 1`, na.rm = TRUE)

Proj1_DS$`BsmtFin SF 2`[is.na(Proj1_DS$`BsmtFin SF 2`)] = mean(Proj1_DS$`BsmtFin SF 2`, na.rm = TRUE)

Proj1_DS$`Mas Vnr Area`[is.na(Proj1_DS$`Mas Vnr Area`)] = mean(Proj1_DS$`Mas Vnr Area`, na.rm = TRUE)

#Or we can use the following code to update all the missing numeric value from the data set and store it in an object

#Separating integer values from the housing data set first

int_dataProj1_DS = Proj1_DS[, sapply(Proj1_DS, class) == 'integer']

updated_Proj1_DS = int_dataProj1_DS %>%
  mutate_if(is.integer, function(n) ifelse(is.na(n), mean(n, na.rm = TRUE), n))
```

The str function was used to check for NA values in columns before selecting those columns to fill in missing values. The missing fields were replaced with mean values of those respective columns using the is.na function, allowing a thorough regression analysis to be performed on the given data set down the line. The same procedure was done again as the whole process was tedious. Here the integer values were stored in an object then mutate_if function was used on the data set to fill the missing values with mean values.

Fourth task

Exhibit a correlation matrix for the missing values by using the cor function.

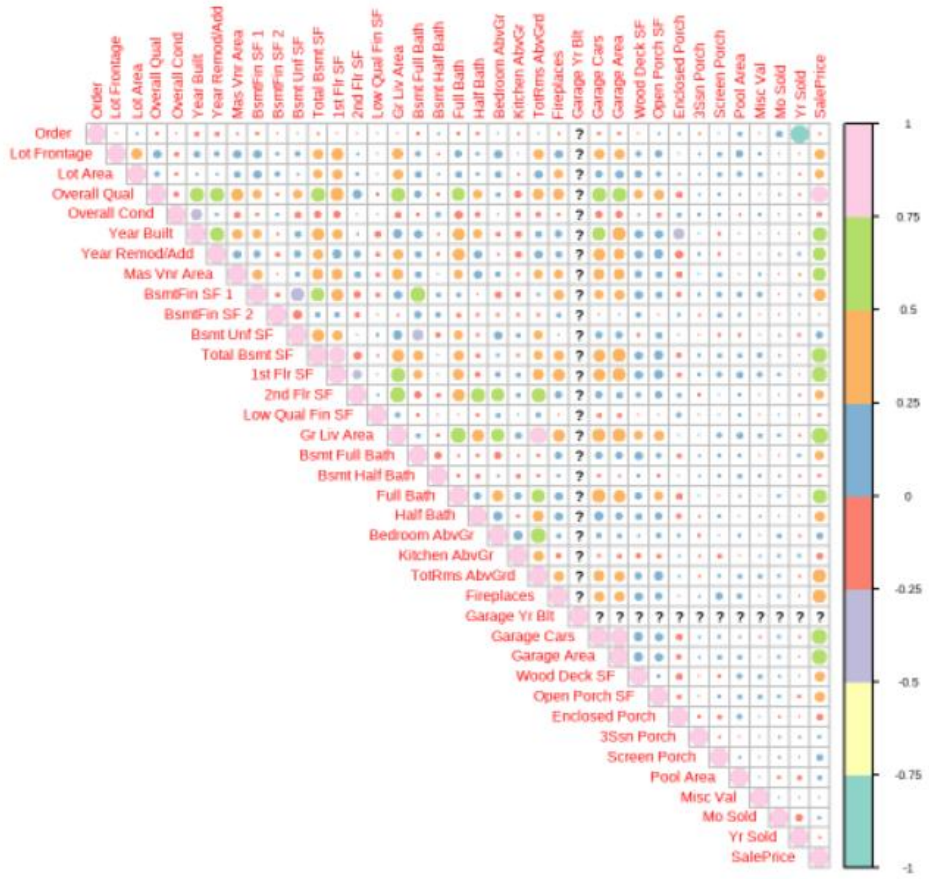
	Order	Lot Frontage	Lot Area	Overall Qual	Overall Cond	Year Built	Year Remod/Add	Mas Vnr Area	BsmtFin SF 1	BsmtFin SF 2	Bsmt Unf SF	Total Bsmt SF
Order	1	-0.006	0.031	-0.049	-0.011	-0.052	-0.076	-0.031	-0.032	-0.003	0.006	-0.029
Lot Frontage	-0.006	1	0.366	0.199	-0.067	0.116	0.086	0.203	0.2	0.04	0.108	0.33
Lot Area	0.031	0.366	1	0.097	-0.035	0.023	0.022	0.127	0.192	0.083	0.024	0.254
Overall Qual	-0.049	0.199	0.097	1	-0.095	0.597	0.57	0.427	0.284	-0.041	0.27	0.547
Overall Cond	-0.011	-0.067	-0.035	-0.095	1	-0.369	0.048	-0.135	-0.051	0.041	-0.137	-0.173
Year Built	-0.052	0.116	0.023	0.597	-0.369	1	0.612	0.312	0.28	-0.027	0.129	0.407
Year Remod/Add	-0.076	0.086	0.022	0.57	0.048	0.612	1	0.196	0.152	-0.062	0.165	0.297
Mas Vnr Area	-0.031	0.203	0.127	0.427	-0.135	0.312	0.196	1	0.3	-0.016	0.091	0.395
BsmtFin SF 1	-0.032	0.2	0.192	0.284	-0.051	0.28	0.152	0.3	1	-0.054	-0.478	0.537
BsmtFin SF 2	-0.003	0.04	0.083	-0.041	0.041	-0.027	-0.062	-0.016	-0.054	1	-0.239	0.09

Showing 1 to 10 of 37 entries Previous 1 2 3 4 Next

The category values were removed from the full data set using the unlist function, and a correlation matrix was generated using the remaining numerical data set using the cor functions on the unlisted numeric data. The correlation values were then presented with the help of the DT table library. We can see that the correlation values for the same columns, such as order - order or lot frontage - lot frontage, are 1, which is perfect correlation, as both values are the same.

Fifth task

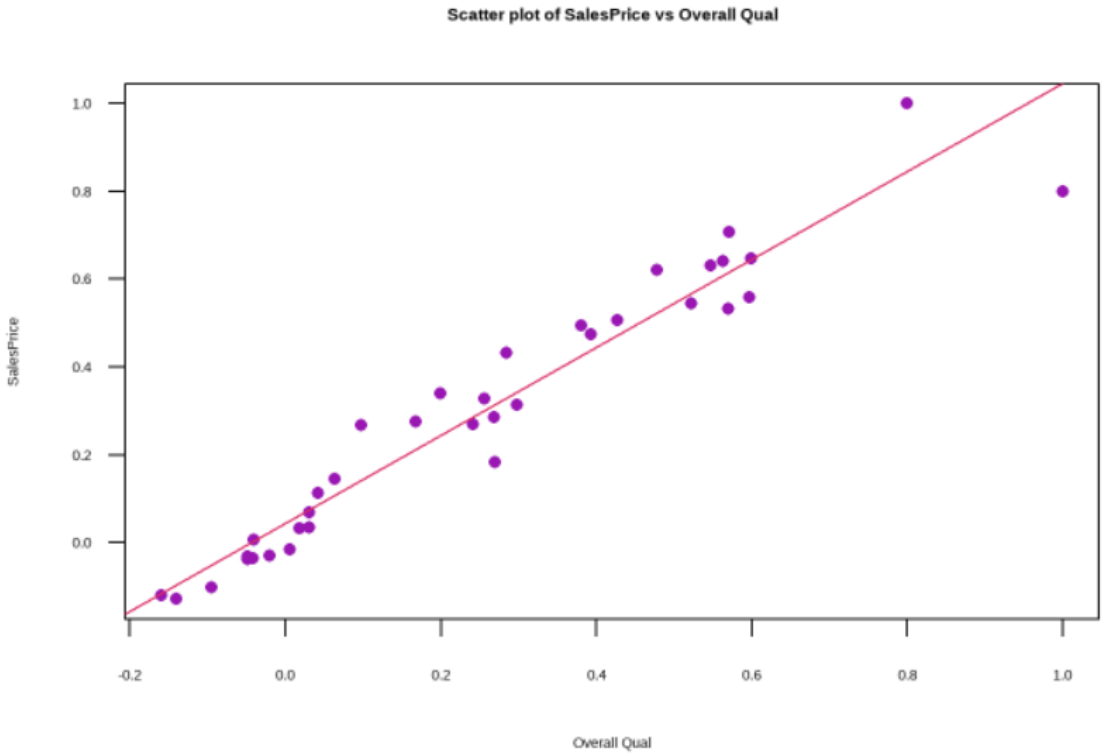
Exhibit a plot of correlation matrix which was obtained in the previous task.

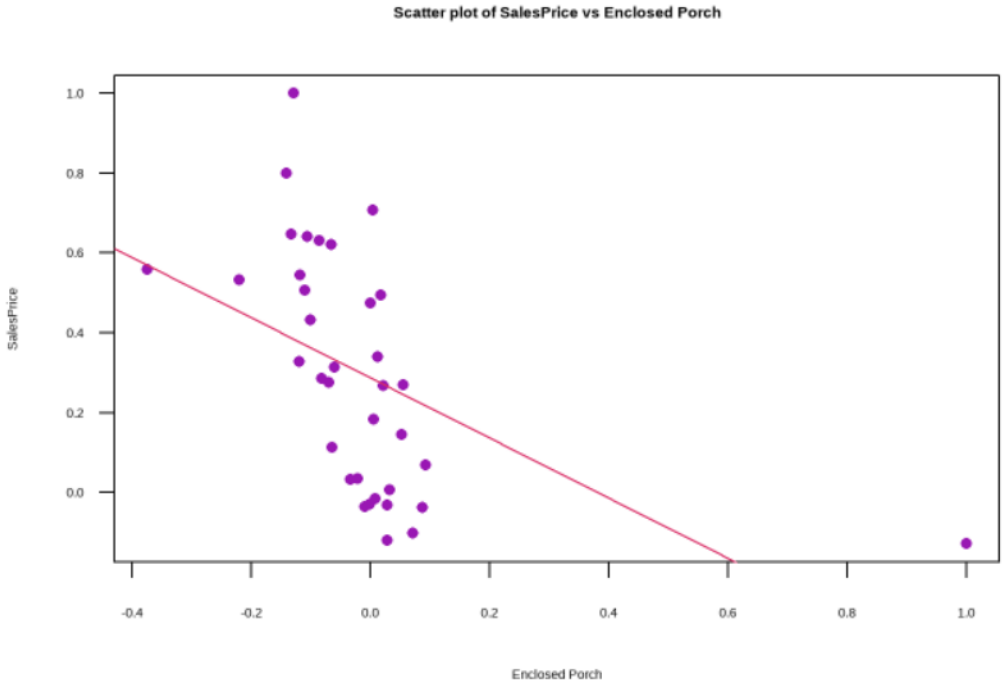


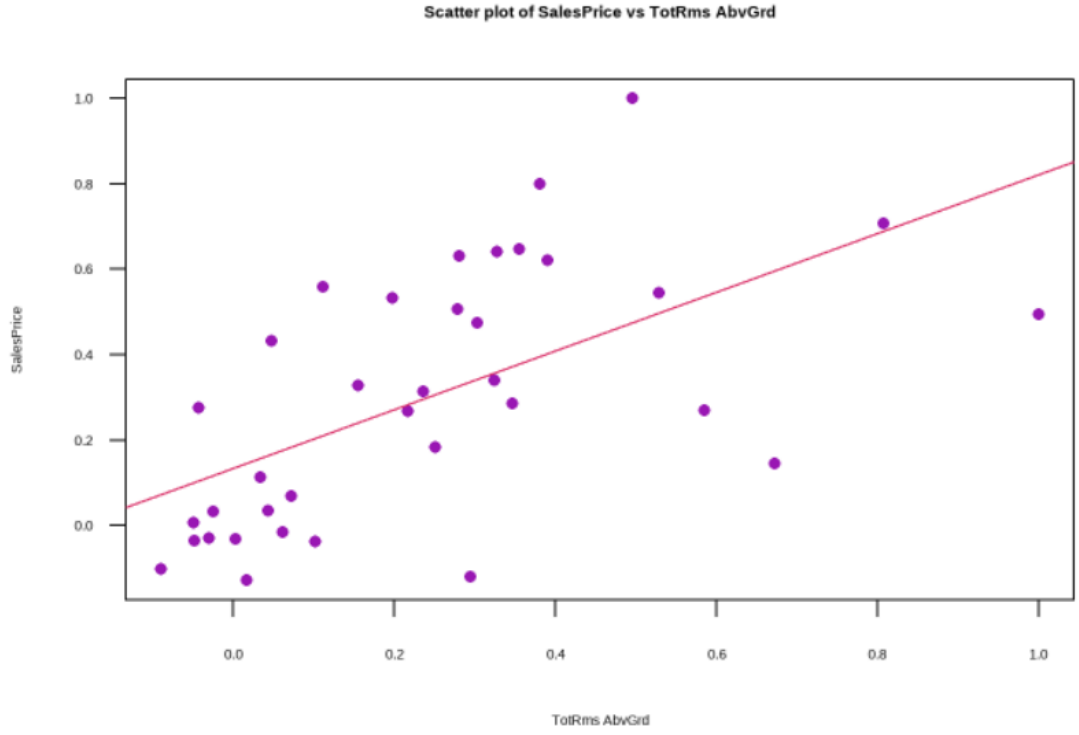
The correlation matrix acquired in the previous task was shown using the corrplot package. We can see that various shades indicate different correlation ranges by looking at the correlation matrix plot. For example, pink and cyan show strong correlation ranges of 0.75 to 1 for positive and negative correlation, respectively.

Sixth task

Exhibit multiple scatterplot for highest and lowest correlation values with respect to dependent variable SalesPrice. Also, present a plot with for correlation value nearest to 0.5 with SalesPrice.







In this task, the plot function was used to create a scatterplot for dependent and independent data. The regression line was calculated using the lm function, and the

regression line on the scatterplot was created using the abline function. By examining the first scatterplot, we can see that it is a strong positively correlative scatterplot, since most of the values are close to the regression line, implying that the residual difference will be minimal. The second scatterplot is negative, but it lacks a significant negative association since the residual difference for some values is enormous. When compared to the first scatterplot, the last scatterplot has a positive correlation but not a strong positive correlation because most of the values are not near the regression line, resulting in a large residual difference and thus the correlation value, though positive, will be closer to 0 than to 1.

Seventh task

Using three continuous variables of your choice fit the regression model.

```
##
## Call:
## lm(formula = Proj1_DS$SalePrice ~ Proj1_DS$`BsmtFin SF 2` + Proj1_DS$`Total Bsmt SF` +
##     Proj1_DS$`Bsmt Unf SF`)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -657596  -39052  -12617   32632  407529
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)              63608.422   2957.041  21.511  < 2e-16 ***
## Proj1_DS$`BsmtFin SF 2`    -40.223      7.048  -5.707 1.27e-08 ***
## Proj1_DS$`Total Bsmt SF`   125.087      2.884  43.378  < 2e-16 ***
## Proj1_DS$`Bsmt Unf SF`     -22.093      2.965  -7.452 1.21e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61220 on 2926 degrees of freedom
## Multiple R-squared:  0.4133, Adjusted R-squared:  0.4127
## F-statistic: 687.1 on 3 and 2926 DF,  p-value: < 2.2e-16
```

The continuous variables BsmtFin SF2, Total Bsmt SF, BsmtUnf SF were used as independent variable while SalesPrice was used as the dependent variable. The multiple regression value for the above variables were calculated using the lm function then summary function was used to present them. By looking at the summary table we can

notice that the muliple R-squared value came out to be 0.4133 or 41.33% of the changes in x(independent) can be explained by variation in y(dependent).

Eighth task

Present the model which was calculated in the previous task in equation form. Also, present the correlation coefficient for each model.

```
## [1] "Multiple regression formula, y = 63608.422 + -40.223 x1 + 125.087 x2 + -22.093 x3"
```

Correlation & Determination Coefficient values for the patient data set

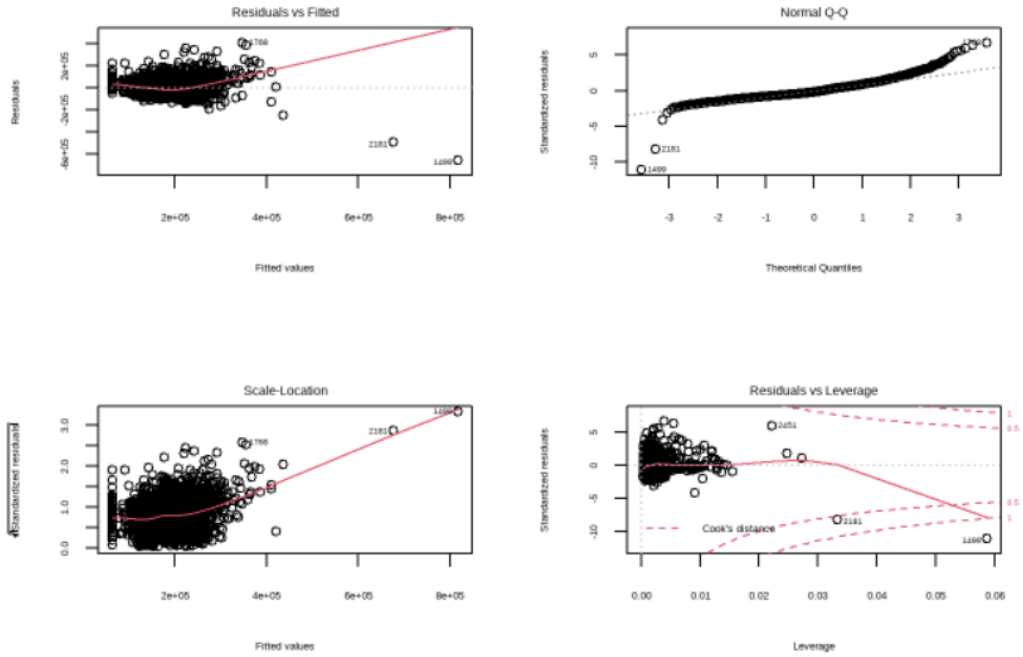
	Correlation Coefficient	Determination Coefficient
SalesPrice & BsmtFin SF 2	0.006	0.000
SalesPrice & Total Bsmt SF	0.632	0.400
SalesPrice & Bsmt Unf SF	0.183	0.033

Showing 1 to 3 of 3 entries Previous 1 Next

The intercept and coefficient values were taken from the multiple regression model summary that was created in the previous job. Then, using the formula y = a + b1x1+ b2x2 + b3x3, where a is the intercept value and b1, b2, and b3 are the coefficient values, these data were expressed in a multiple regression model. Then, using a DT library table, the correlation coefficient and determination for each of the values with the SalesPrice were computed and reported. By analyzing the table we can notice that the correlation coefficient for SalesPrice & BsmtFin SF 2 is very very low 0.006 which means that they have no correlation among themselves while for SalesPrice & Total Bsmt SF it's 0.632 which a positive correlation.

Ninth task

Exhibit all the four graphs by using the plot function on your multiple regression model.



The par function was used to exhibit the four graphs (Residual vs Filled, Normal Q - Q, Scale Location, and Residual vs Leverage) on a single page. When we look at these graphs, we can see how the outliers' positions change depending on the graph type.

Tenth task

Perform a check for multicollinearity for the multiple regression model you created in the previous task.

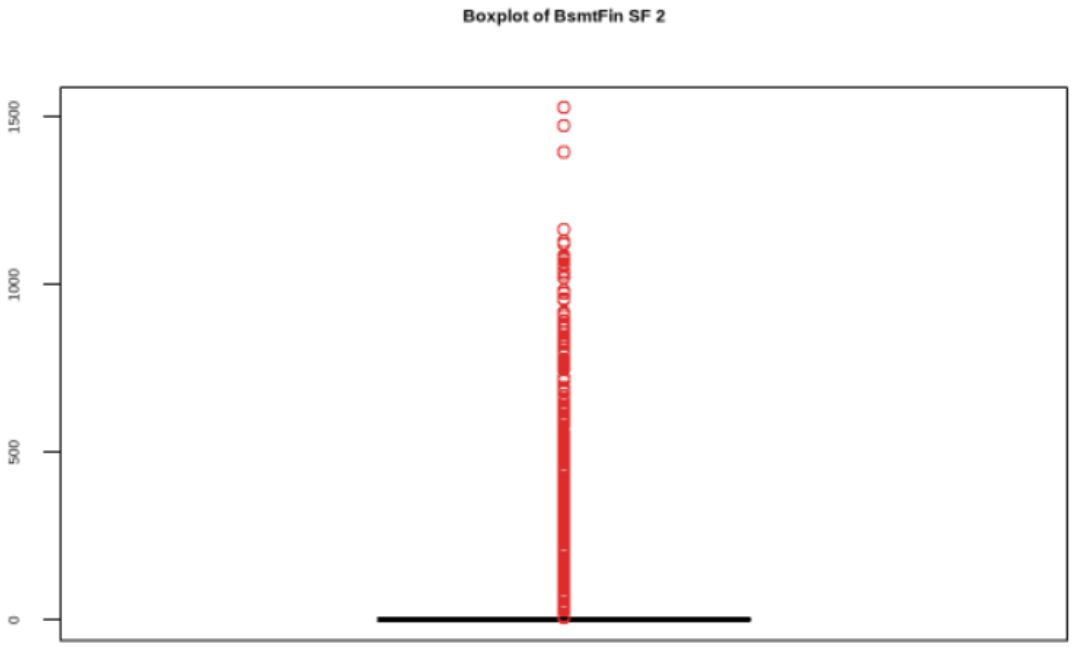
```
##  Proj1_DS$`BsmtFin SF 2` Proj1_DS$`Total Bsmt SF`   Proj1_DS$`Bsmt Unf SF`
##                 1.110603                 1.261156                 1.326477
```

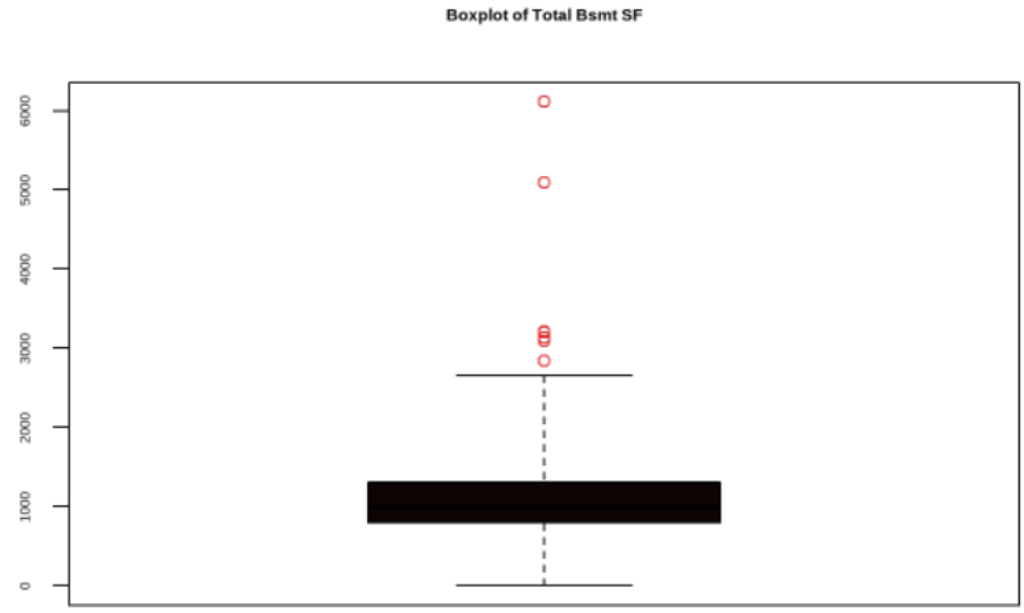
The vif function was used to check for multicollinearity of the selected multiple regression model. Multicollinearity is a high correlation among the predictors essentially where you can use one predictor to predict another predictor. It's not good to have Multicollinearity

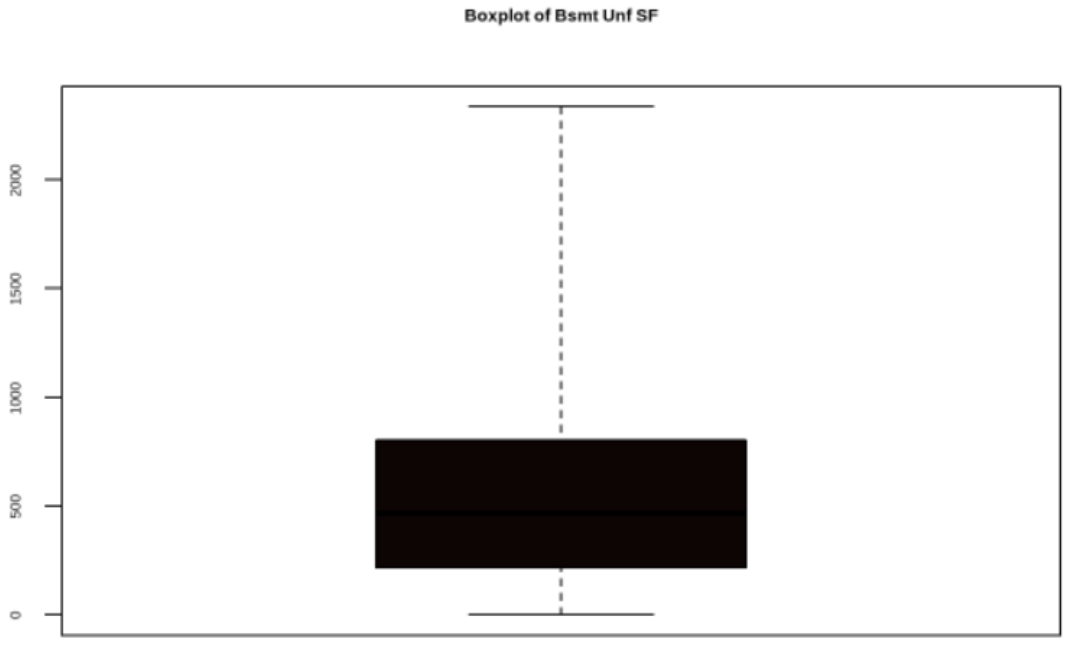
in the model as it will inflate and destabilize the regression coefficients(It makes them harder to trust). Anything less than 5 is not of concern. Anything 10 or greater means that we have high level of multicollinearity. In our case all the values are below 2 so we are safe and it means that we have no multicollinearity. But if multicollinearity existed we would have remove some of the highly correlated independent variables.

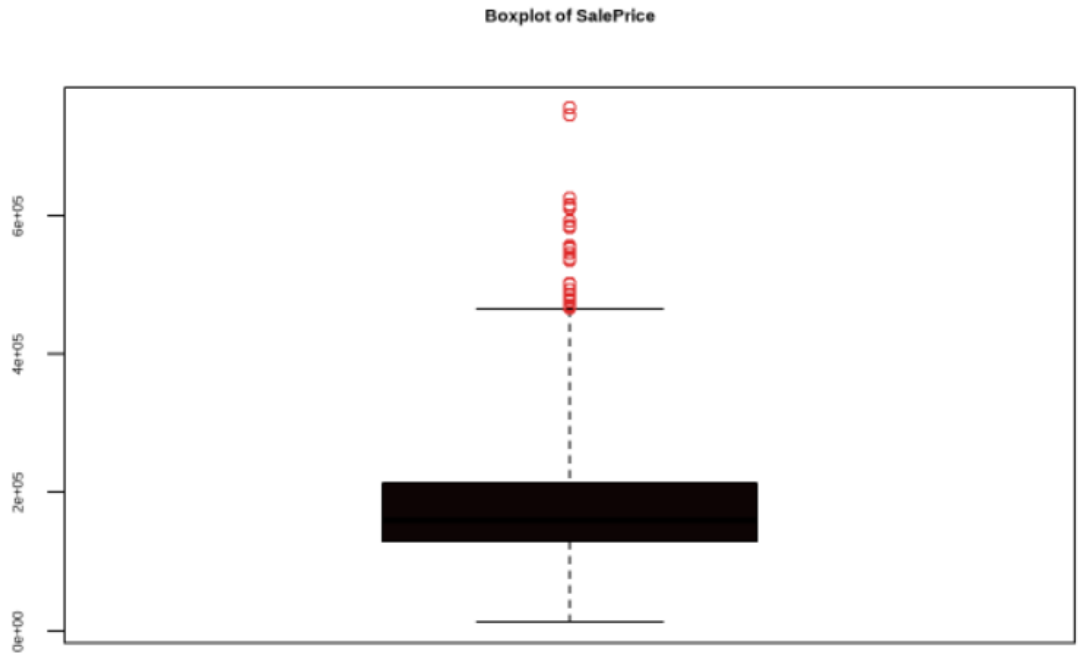
Eleventh task

Perform a check for outliers on the multiple regression model that you created in the previous task.




Boxplot of Total Bsmt SF
0
1000
2000
3000
4000
5000
6000


Boxplot of Bsmt Unf SF
0
500
1000
1500
2000



The calculated values were saved in their appropriate objects using the boxplot$out function for each of the independent and dependent variables that were chosen in the multiple regression model. If eliminating these outliers improves the correlation of determination/correlation, we should do so and see if the removal results in a high determination value.

Twelfth task

Perform a task to remove the outliers which were calculated in the previous task.

```
#The outliers were removed from the multiple regression dependent and independent variables. Outliers were removed from data set then that object(No_outlier_val1) is used in second line of code to remove outliers on second independent variable. Therefore No_outlier_val2 = No_outlier_val1 and not Proj1_DS as we will only be removing outliers from Proj1_DS if we put it there but are removing outliers completely from the entire data set for our multiple regression model.

No_outlier_val1 = Proj1_DS[-which(Proj1_DS$`BsmtFin SF 2` %in% Outlier_val1),]
No_outlier_val2 = No_outlier_val1[-which(Proj1_DS$`Total Bsmt SF` %in% Outlier_val2),]
No_outlier_val3 = No_outlier_val2[-which(Proj1_DS$`Bsmt Unf SF` %in% Outlier_val3),]
No_outlier_val4 = No_outlier_val3[-which(Proj1_DS$SalePrice %in% Outlier_val4),]
```

Outliers were eliminated from the multiple regression dependent and independent variables established in the previous task, and new values were placed in the object without outliers.

Thirteen task

Perform a subset regression method to identify the best model.

```
## Subset selection object
## Call: regsubsets.formula(Proj1_DS$SalePrice ~ Proj1_DS$`BsmtFin SF 2` +
##     Proj1_DS$`Total Bsmt SF` + Proj1_DS$`Bsmt Unf SF`, data = Proj1_DS,
##     nbest = 4)
## 3 Variables  (and intercept)
##                          Forced in Forced out
## Proj1_DS$`BsmtFin SF 2`      FALSE      FALSE
## Proj1_DS$`Total Bsmt SF`     FALSE      FALSE
## Proj1_DS$`Bsmt Unf SF`       FALSE      FALSE
## 4 subsets of each size up to 3
## Selection Algorithm: exhaustive
##          Proj1_DS$`BsmtFin SF 2` Proj1_DS$`Total Bsmt SF`
## 1  ( 1 ) " "                     "*"
## 1  ( 2 ) " "                     " "
## 1  ( 3 ) "*"                     " "
## 2  ( 1 ) " "                     "*"
## 2  ( 2 ) "*"                     "*"
## 2  ( 3 ) "*"                     " "
## 3  ( 1 ) "*"                     "*"
##          Proj1_DS$`Bsmt Unf SF`
## 1  ( 1 ) " "
## 1  ( 2 ) "*"
## 1  ( 3 ) " "
## 2  ( 1 ) "*"
## 2  ( 2 ) " "
## 2  ( 3 ) "*"
## 3  ( 1 ) "*"
```

The regsubsets function from the leaps library was used and then summary function was used on that to get the best predictor model values. Here, the best 1 predictor model is Total Bsmt SF, the best 2 predictor model is Total Bsmt SF and Bsmt Unf SF.

Fourteen task

Comparing the model equation between the multiple regression and subset of that multiple regression.

We received 41.33 percent as our multiple R - Squared value when we ran the summary function on the multiple regression data, which demonstrates correlation but isn't strong

enough because our data contained a lot of outliers. However, when we exclude a section of that data, we can see that we have predictor values for each. For example, in our multiple regression dataset, the best one predictor value model is Total Bsmt SF. As a result, I favor the two model since it gives us the best one predictor value as well as the best two and three predictor model values. There are no outliers in this model, which improves the correlation in our multiple regression model.

CONCLUSION

Finally, I had a better understanding of the multiple regression model. I learnt how to update your data collection and fill in missing numbers using mean, median, and other statistics. I learnt about correlation matrices and the corrplot package, which was used to generate the matrices. Aside from that, I learnt how to use the plot function to create four distinct graphs and how to manipulate them. I also learnt how to use the vif function and handle outliers. Not only do you need to cope with them, but you also need to know how to get rid of them from your data collection. I also learnt about the leaps library and how to use the regsubset function to work with subsets.

BIBLIOGRAPHY

1) Galton, Pearson, and the Peas: A Brief History of Linear Regression for Statistics Instructors. (2017). Taylor & Francis. https://www.tandfonline.com/doi/full/10.1080/10691898.2001.11910537

2) Multiple Regression: Definition, Uses and 5 Examples. (2020). Indeed Career Guide. https://www.indeed.com/career-advice/career-development/multiple-regression

3) Kumar, A. (2022, February 20). Linear regression hypothesis testing: Concepts, Examples. Data Analytics. https://vitalflux.com/linear-regression-hypothesis-testing-

examples/

4) Banerji, A. (2022, January 6). Hypothesis Testing On Linear Regression - Nerd For Tech. Medium. https://medium.com/nerd-for-tech/hypothesis-testing-on-linear-regression-c2a1799ba964