University of Cape Town

Department of Statistical Sciences

STA5076Z Supervised Learning

Assignment 1

EVRCHA001 - Charl Everts

08 June 2020

Department of Statistical Sciences

Plagiarism Declaration Form

A copy of this form completed and signed, to be attached to all coursework submissions to the Statistical Sciences Department. Submissions without this form will not be marked.

COURSE CODE:	STA5076Z
COURSE NAME:	Supervised Learning
STUDENT NAME:	Charl Everts
STUDENT NUM:	EVRCHA001

Plagiarism Declaration

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used a generally accepted citation and referencing style. Each contribution to, and quotation in, this tutorial/report/project from the work(s) of other people has been attributed and has been cited and referenced.
3. This tutorial/report/project is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as his or her own work.
5. I acknowledge that copying someone else's assignment or essay, or part of it, is wrong, and declare that this is my own work.

Note that agreement to this statement does not exonerate you from the University's plagiarism rules (http://www.uct.ac.za/uct/policies/plagiarism_students.pdf).

Signature: Charl E Date: 08 June 2020

Table of Contents

List of Figures

List of Tables

Abbreviations/Symbols

Q_1	25th percentile or 1st quartile
Q_3	75th percentile or 3rd quartile
IQR	interquartile range
SD	standard deviation
MSE	mean squared error
cor	correlation coefficient ‘r’
H_0	null hypothesis
H_A	alternate hypothesis
RSS	residual sum of squares
R^2	proportion of the variance of a response variable explained by a predictor variable
C_p	Mallows’ information criteria
BIC	Bayesian information criteria
X_j	j^{th} predictor variable
β_j	coefficient of X_j
I	for loop index
k	training sets or folds used in cross validation
LASSO	least absolute shrinkage and selection operator

Chapter 1

Introduction

0.8 The Tips Dataset

The “tips” dataset was created by a Californian waiter in 1995. The dataset is comprised of seven variables and 244 observations. The variables are tip in dollars, total bill in dollars, sex of the bill payer, whether there were smokers in the party, day of the week, time of day, and size of the party. Each of the 244 rows in the dataset corresponds to a different dining party. A sample of 200 observations are randomly selected, which make up the final data set used in this report.

The dataset has three numerical and four categorical variables. Numerical variables include total bill, tip, and size of the party. Categorical variables include sex of the bill payer, whether a party had smokers or not, day of the week, and time of the day. Each of these categorical variables have multiple levels. Sex has two levels (male and female), smoker status has two levels (Yes and No), day of the week has four levels (Thursday, Friday, Saturday, Sunday), and time of day has two levels (lunch and dinner).

Generally speaking, a waiter’s tip depends on the quality of service, size of the bill and the mood of the paying customer. According to an article in USA Today, general dining etiquette prescribes that American citizens should tip their servers 15% on average. It is interesting to note that tip as a percentage of the total bill is not a feature of the tips dataset.

1.2 Data Quality Assessment

The dataset is well organised and easily interpretable. There are no missing values or any other erroneous entries. The first column named ‘X’ serves as a reference column for the records. Since the R output automatically generates its own, reference column ‘X’ was dropped from the dataset, so as not to cause confusion when referencing data entries. The remaining seven features were renamed as follows: “Total_Bill”, “Tip”, “Sex_of_Tipper”, “Smoker”, “Day”, “Meal” and “Size_of_Table”.

The variable “Tip” on its own cannot provide further evidence regarding the generosity of the patron’s tip. At face value, a $5 tip may seem like a better pay day than a $4 tip. However, if the $4 tip relates to 50% of the table’s bill, and the $5 tip relates to 10% of a different table’s bill, a waiter would much rather have served the former table. It was therefore determined that “Tip_Percent” would be added to the dataset in order to identify potential outliers.

1.3 Outlier Detection

Upon creating multiple boxplots of the variables, it is evident that two “Tip Percent” data points are blatant outliers.

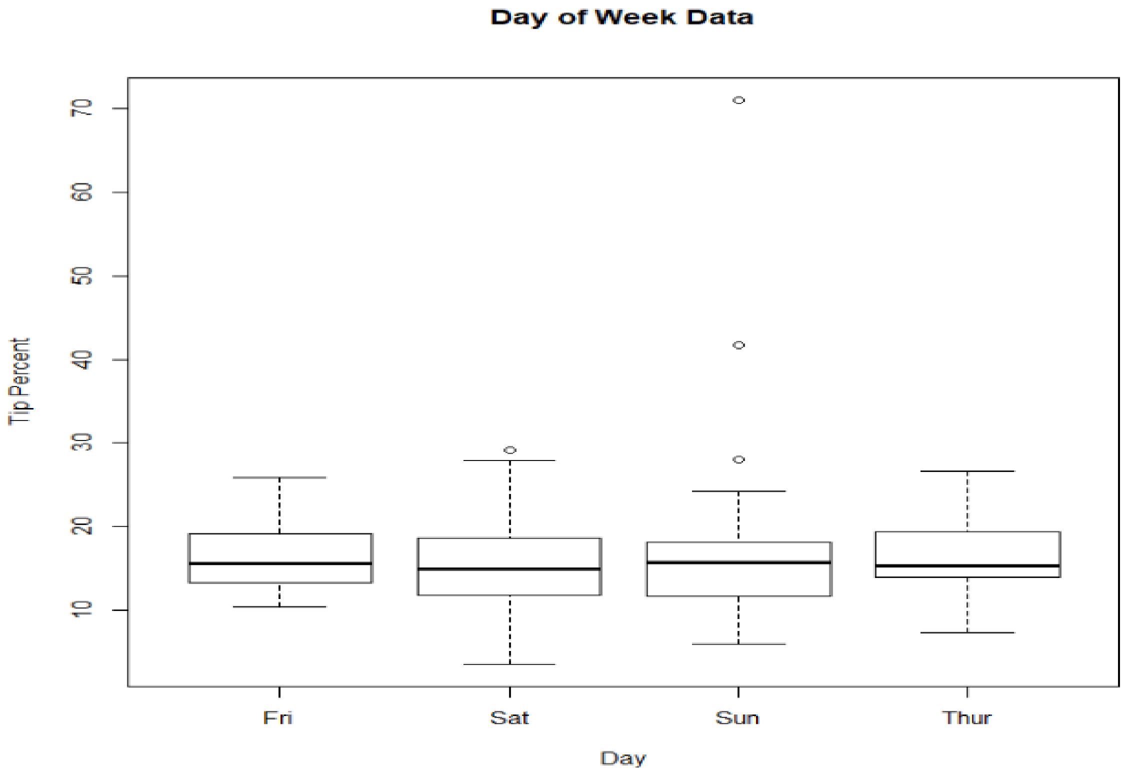


Figure 1.1: *Visible outliers depicted in Day vs Tip Percent boxplot.*

In order to verify the severity of the outliers, Tukey’s fences and the standard deviation test methods are conducted.

Tukey’s fences:

In this method, a data point x is deemed an outlier if it meets the following criteria:

$x < Q_1 – 1.5*IQR$ or $x > Q_3 + 1.5*IQR$

Additionally, x is deemed a “far out” outlier if it meets the following criteria:

$x < Q_1 – 3*IQR$ or $x > Q_3 + 3*IQR$

Standard deviation method:

In this method a data point x is deemed an outlier if it meets the following criteria:

$x < mean – 3*SD$ or $x > mean + 3*SD$

Both approaches confirm the suspicions that two problematic outliers exist, namely entries [134] and [48], with a tip percent of 71.0% and 41.7% respectively, compared to a sample mean of 15.8%. Interestingly, both of these outliers were a Sunday evening dinner at a two-seater table in the smoker’s section.

These data points are removed in order to increase the statistical significance of the subsequent models. The dataset used is now comprised of 198 observations.

1.4 Descriptive Statistics

The frequency distribution of Tip Percent (Figure 1.2) shows that American citizens do indeed follow the prescribed average tip of 15% in a restaurant. The mean and median tip percentages (Table 1.1) are virtually identical, 15.435% and 15.256%. This is good indication of the symmetry in the Tip Percent variable which follows the standard normal distribution. The mode of Tip Percent is tie between 15% and 20%. There is a remarkable spike at the 20% tip rate level. This may be due to the fact that diners are willing to round off their tips at 20% for above average service.

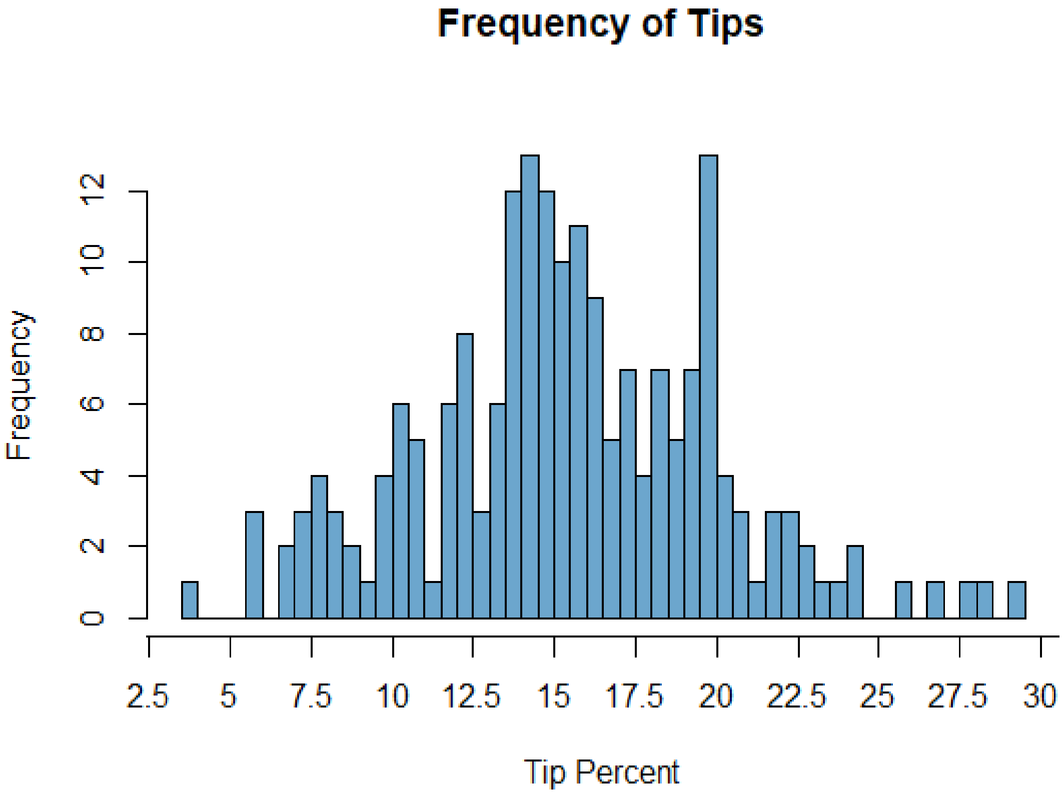


Figure 1.2: *Frequency distribution of Tip Percent.*

It is evident that patrons prefer to tip in whole dollar amounts. A $2 tip is recorded 28 times and a $3 dollar tip the next most prevalent, being tipped 18 times. Examining the Size of Table data, it would appear that the restaurant is designed to fit predominantly smaller tables, or that the waiter in question was designated to work in an area with smaller tables. The waiter served two-seater tables the most, being recorded 123 times. The next most is a three-seater being served 34 times.

Variable	Min	Q1	Median	Mean	Q3	IQR	Max	SD	Mode
Total Bill	5.75	13.42	18.27	20.23	24.58	11.16	50.81	8.8018	-
Tip	1.000	2.000	3.000	3.009	3.587	1.59	10.000	1.4217	2.000
Tip Percent	3.564	12.565	15.256	15.435	18.510	5.95	29.199	4.5654	-
Size of Table	1.000	2.000	2.000	2.621	3.000	-	6.000	0.9680	2.000

Table 1.1: *Summary statistics of numerical variables.*

Figure 1.3 indicates a weak negative linear relationship between Tip Percent and Total Bill (cor = -0.282). This could indicate that patrons are reluctant to tip a higher percentage of the total bill as Total Bill increases. This might be in order to save money on the total bill, or tippers feel that a waiter should be satisfied with a larger tip, even though the tip is a smaller percent of Total Bill.

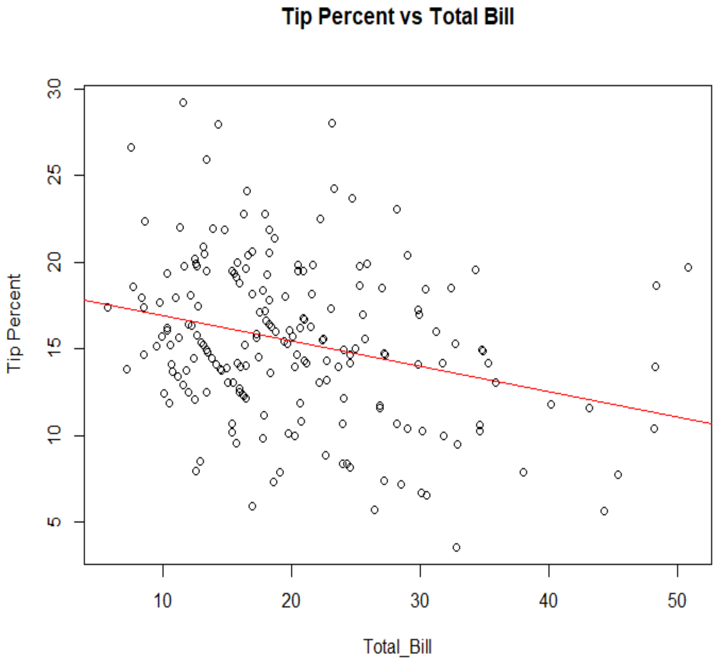


Figure 1.3: *Tip Percent vs Total Bill.*

The trend in figure 1.4 shows the effect of Party Size on Tip Percent. As party size increases, Tip Percent suffers a slight decrease, which agrees with findings in Figure 1.3.

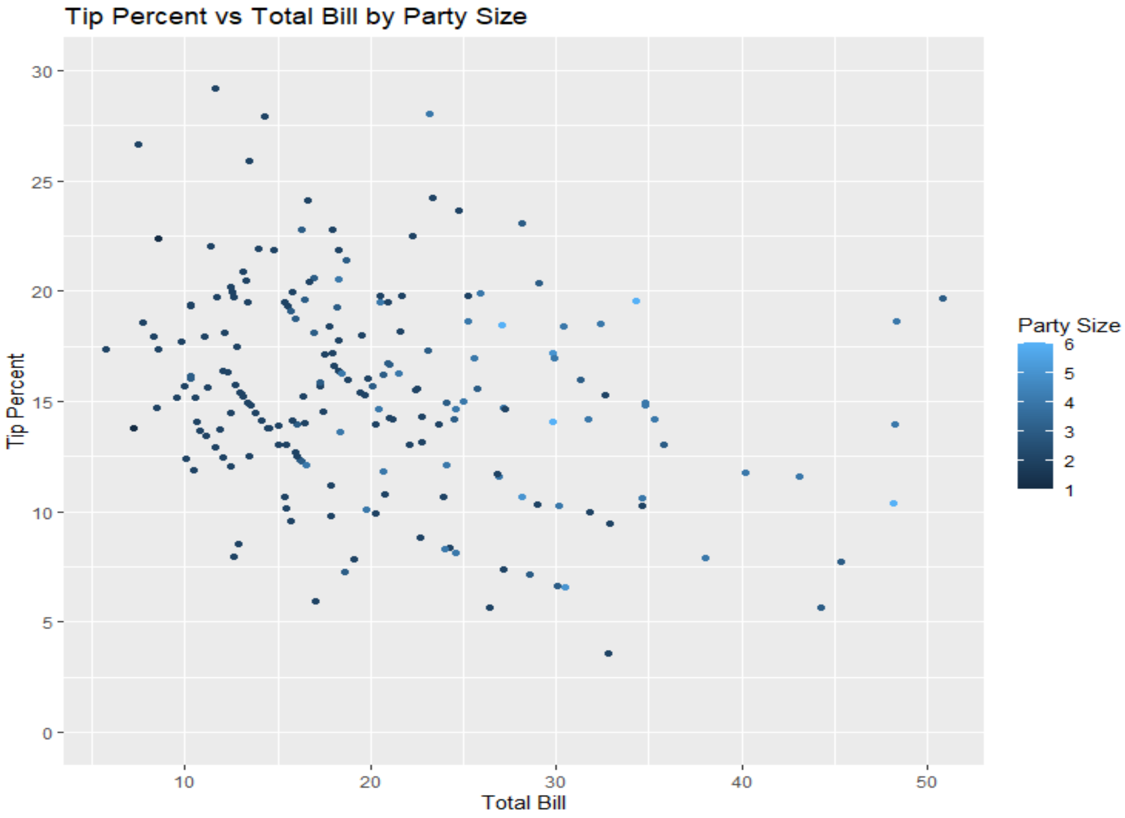


Figure 1.4: *Tip Percent vs Total Bill by Party Size.*

The aforementioned trend continues in Figure 1.5. There exists a weak negative linear relationship between Tip Percent and Total Bill when split into the Dinner and Lunch meals respectively.

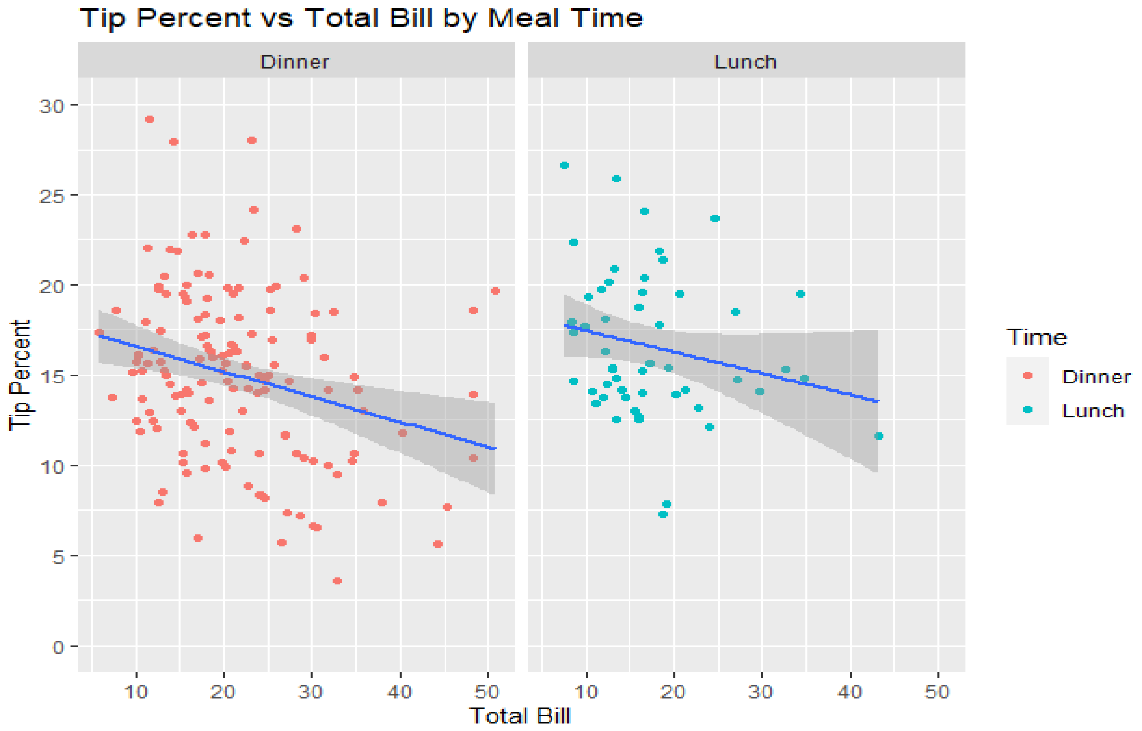


Figure 1.5: *Tip Percent vs Total Bill by Meal Time.*

From figure 1.6 it is possible to draw the following conclusions:

- Male tippers have a greater dispersion of Tip Percent, hence more willing to drastically over- or under-tip than females.
- Smokers display the same trend as Male tippers, that they are more willing to pay a Tip Percent either far above or below the expected average.
- When comparing Figure 1.1 and 1.6, it is evident that the removal of the farout outliers substantially tighened the dispersion of the Day of the Week Data.

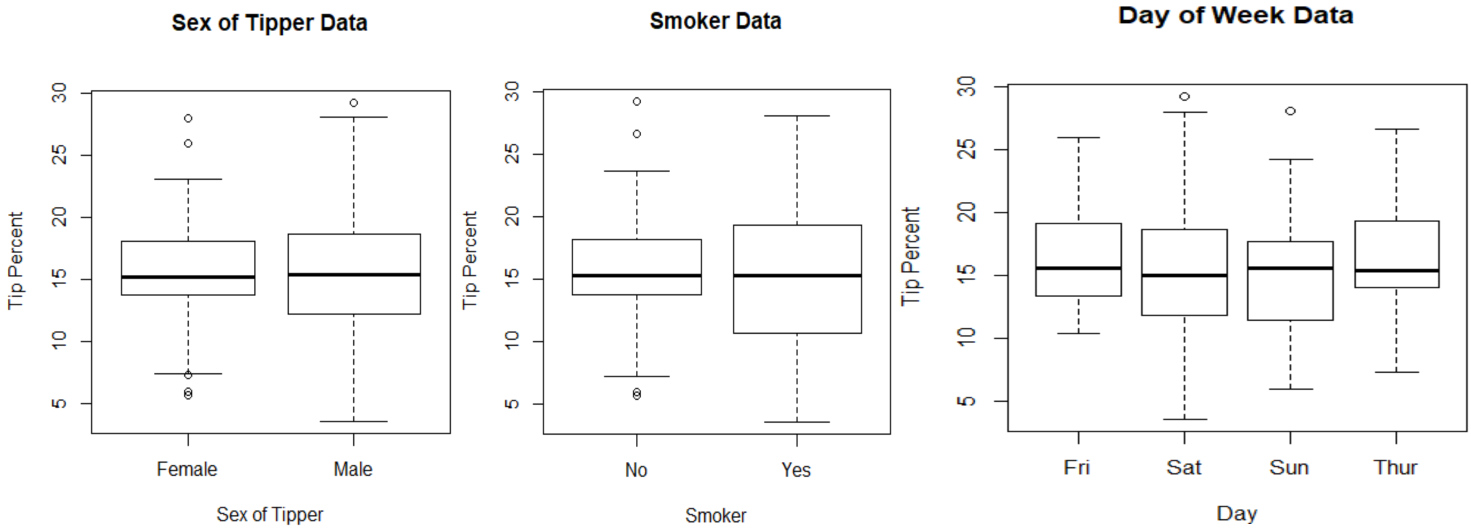


Figure 1.6: *Boxplots of Sex of Tipper, Smoker, and Day data vs Tip Percent*

1.5 Assignment Objectives

The objective of this report is to build a linear model to predict the average amount of tip in dollars a waiter can expect to earn from a table, given the predictor variables. The following objectives are addressed:

i. Fit a multiple linear regression model to a training data set, regressing the variable “tip” on all explanatory variables. Discuss the model in terms of fit and variable significance. Use the model to predict tip for the testing set, and calculate the corresponding MSE.

ii. Improve the model by conducting various variable selection techniques. The final model should include an interaction term. The improvement of the model should be measured in terms of the testing set MSE.

iii. Discuss the residual diagnostics of the final model.

Chapter 2

Methodology

2.1 Multiple Linear Regression Model

In multiple linear regression, multiple distinct predictor variables are used to predict a specific response variable. The prediction of a response variable using predictor variables is expressed as follows:

$$f(X) = E[Y|X] = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p$$

Where *p* denotes the number of predictors used in the model, and β_j is the coefficient corresponding to the predictor variable X_j. Each β_j is interpreted as the change in average value of Y for one-unit change in X_j, while all other variables are held constant. It is therefore possible to retain the interpretive simplicity of a Simple Linear Regression Model, whilst being able to account for multiple predictors. In order to assess the significance of predictor variables in the model, one must investigate the F-statistic and p-values corresponding to the respective coefficients.

2.1.1 The F – Statistic

The F – Statistic makes a judgement on multiple coefficients at the same time. The F-test for overall significance compares the proposed model to an intercept-only regression. The corresponding significance hypothesis test is as follows:

H_0: The intercept-only regression is equivalent to the current model; hence additional variables provide no material improvement.

H_A: The fit of intercept model is significantly less powerful than the current model; hence additional variables will provide drastic improvement.

An F-statistic with a p-value < 0.05 is deemed significant, hence H_0 is rejected. Ideally an F-statistic p-value < 2.2e-16 indicates the highest level of significance.

2.1.2 Pr(>|t|): p-values

The so-called p-values for different coefficients in the multiple linear regression, tests the null hypothesis that its coefficient is equal to zero; meaning the variable does not add value to the model.

H_0: There is no relationship between the variables being investigated.

H_A: The response variable is affected by the predictor variable.

For p-values < 0.05, the null hypothesis is rejected. A coefficient has the highest level of significance if the p-value < 2.2e-16. A predictor with an extremely low p-value is likely to be a meaningful addition to the model since the predictor's value is related to changes in the response.

2.2 Model Improvement by Variable Selection

2.2.1 Simple Linear Regression

One of the methods at our disposal for variable selection is running a simple linear regression on all the predictor variables individually. Evaluating their individual p-values as stated in 2.1.2 gives insight into the predictive power of each variable when considered on their own. Simple liner regression is a first iteration at identifying which variables may be significant or not.

$$f(X) = E[Y|X] = \beta_0 + \beta_1 X.$$

This regression is a simplified version of the multiple linear regression in 2.1. This case is limited to a single predictor variable X and the corresponding coefficient β_1. The intercept of the line is given as β_0.

2.2.2 Best Subset Selection

Best subset selection conducts a comparison of all possible models using a specified set of predictor variables. It displays the best selection of variables for a one-predictor model, two-predictor model, and so on. In this case “best” is quantified using RSS. Additionally, plotting RSS, adjusted R^2, C_p and BIC against number of variables indicates the most appropriate number of variables the model should use.

2.2.3 Forward and Backward Stepwise Selection

Stepwise selection aims to select and remove individual predictors, in an iterative manner, based on their statistical significance. The process can be adjusted to the user’s needs. Significance level can be adjusted, as well as how many predictors are fitted.

2.2.4 Validation Set and Cross-Validation

In the validation set approach, only the training set observations are used. This allows the accurate estimation of the test error. The validation set error for the best model of each size is computed. A model matrix is created from the test data set and run through a loop for each size ‘i’ corresponding to the number of predictors in the model. The best model is selected based on the test set MSE. The final step is to run a best subset selection on the full data set in order to obtain accurate coefficient estimates.

In cross-validation, bust subset selection is performed on each of the ‘k’ training sets, known as “folds”. Each observation is allocated to one of the k folds. A for loop is used to perform the operation and test errors for a subset are stored in a matrix. The (I,j) entry corresponds to the test MSE for the i^{th} fold and best j-variable model. A plot of cv errors against number of variables indicates the best number of variables to use relative to the lowest cv error.

2.2.5 Ridge Regression

Ridge regression uses a regularisation technique that penalises residual parameters of a model being learned. In essence, a degree of bias is added to the regression estimates. It is commonly used to address the problem of multicollinearity. Cross-validation is used to choose the tuning parameter λ and find the associated test MSE. Finally, the ridge regression model is refitted on the full data set, using the value of λ chosen by cross-validation. The final output yields coefficients relating to each predictor variable. Ridge regression attempts to use all variables in a dataset to predict the response, hence non are removed.

2.2.6 LASSO

Ridge regression and LASSO are closely related, the main difference being that ridge regression uses all predictor variables, regardless of the significance, assigning lower coefficients to less useful variables. Whereas LASSO has the ability to completely remove predictors if necessary. The LASSO method performs both variable selection and regularization such that prediction accuracy and interpretability is maximised.

2.2.7 Interaction Terms

A multiple linear regression with interaction terms takes the following form:

$$f(X) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_1 \times X_2$$

Interaction effects represent the combined effects of variables on the response. An interaction effect indicates that an additional variable has influence on the relationship between the predictor and response variables. The addition of interaction terms in a model allows for extra flexibility such that previously unnoticed nuances in the data can now be accounted for.

2.3 Residual Diagnostics

2.3.1 Residuals vs. Fitted Values

Residual plots allow a modeller to evaluate and improve a regression model. A residual difference is defined as the difference between the value of the response variable and its predicted value. The residual vs. fitted values plot tests variable linearity and checks if equal variance is displayed along the regression line.

2.3.2 Normal Q-Q

The quantile-quantile or q-q plot is an exploratory graphical mechanism which checks the validity of a data set’s distributional assumption. If the data follows the assumed distribution, then the points on the Normal Q-Q plot will fall on an approximate straight line.

Chapter 3

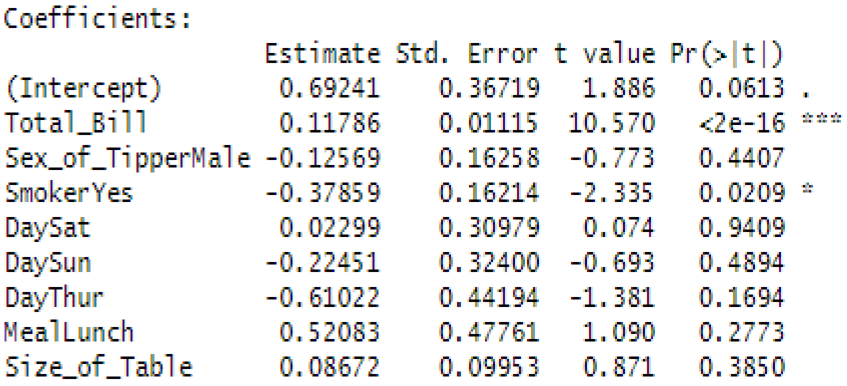
Results

3.1 Multiple Linear Regression Model

In order to fit the model on tips dataset, the 244 observations were read in. The existing ”X” column which serves as a reference column was dropped from the dataset. Using a sample seed of 23, 200 observations were sampled from the full dataset. Upon conducting the Tukey’s fence and SD method for outlier detection, it was determined that entries 134 and 48 be removed as they both exceeded a Tip Percent of 36.5%.

A 20/80 split on the test/train data was implemented. After removing the outliers, 198 observations remained. The test data had 40 observations, with 158 observations in the training data set.

Upon running the multiple linear regression with Tip as the response (model1), the following summary table is constructed (Table 3.1). Total Bill and SmokerYes are the only significant variables since both p-values < 0.05.



```
Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)       0.69241    0.36719   1.886   0.0613 .
Total_Bill        0.11786    0.01115  10.570   <2e-16 ***
Sex_of_TipperMale -0.12569   0.16258  -0.773   0.4407
SmokerYes        -0.37859    0.16214  -2.335   0.0209 *
DaySat            0.02299    0.30979   0.074   0.9409
DaySun           -0.22451    0.32400  -0.693   0.4894
DayThur          -0.61022    0.44194  -1.381   0.1694
MealLunch         0.52083    0.47761   1.090   0.2773
Size_of_Table     0.08672    0.09953   0.871   0.3850
```

Table 3.1: *Coefficients and p-values corresponding to the multiple linear regression.*

The accuracy of this model is based on the MSE of the predicted values and the test set. This would act as the base MSE, with subsequent models’ improvement based thereon. The base MSE (MSE_1) associated with this model is 1.823752.

3.2 Model Improvement by Variable Selection

3.2.1 Simple Linear Regression

Several simple linear regressions were run with Tip as the response against all other variables. Table 3.2 indicates the top three variables in order of significance.

	Estimate	Std. Error	t value	Pr(>|t|)
Total_Bill	0.119073	0.008205	14.512	< 2e-16
Size_of_Table	0.74362	0.09542	7.793	8.62e-13
MealLunch	-0.4312	0.2521	-1.711	0.0891

Table 3.2: *Coefficients and p-values corresponding to simple linear regressions.*

The only variable deemed significant in both the simple and multiple linear regression models was Total Bill. It was therefore decided that model2 would be a simple linear regression with Tip as response and Total Bill as predictor. MSE_2 for model2 was 1.647222, an improvement on the base MSE.

The next model used Size of Table as the only predictor for Tip. This model3 had an associated MSE_3 of 1.890773, which is worse than the base MSE. Regressing Tip on Meal achieved an MSE_4 of 2.118516, again worse than the base MSE.

3.2.2 Best Subset Selection

Conducting a best subset selection method yields a table with asterisks corresponding to the best variables to use in a model, depending on the number of variables required.

# Variables	Total_Bill	SexMale	SmokerYes	DaySat	DaySun	DayThurs	MealLunch	Size
1	*							
2	*							*
3	*		*					*
4	*		*				*	*
5	*		*			*	*	*
6	*	*	*			*	*	*
7	*	*	*		*	*	*	*
8	*	*	*	*	*	*	*	*

Table 3.3: *Best subset selection based on number of variables.*

Figure 3.1 is an indication of the number of variables to be used in the model, based on four different metrics. The residual sum of squares decreases as number of variables increases, however fitting a large number of variables will lead to overfitting and high variance. The adjusted R^2 and C_p plots suggest that two variables are used, while BIC only recommends one predictor variable.

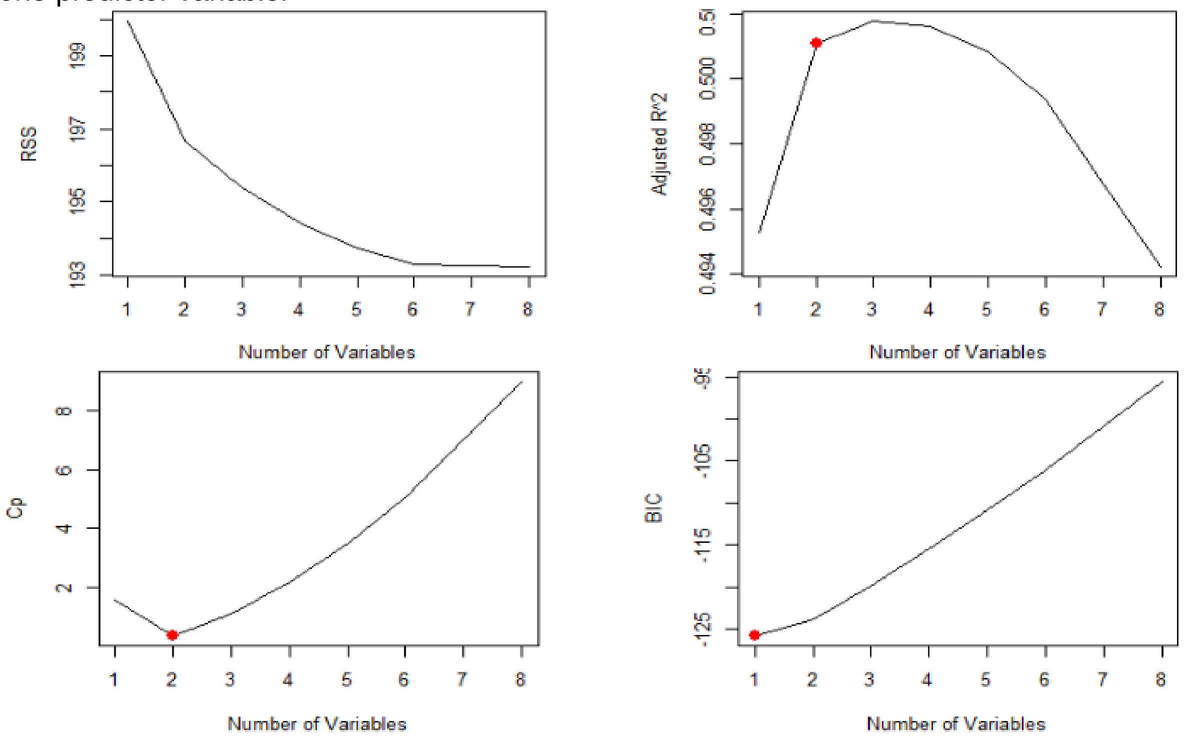


Figure 3.1: *RSS, adjusted R^2, C_p and BIC plotted against number of variables.*

Figure 3.2 indicates the variables of importance for each metric plotted in Figure 3.1. The plots tend to favour Total Bill, Size of Table and SmokerYes as the most significant variables in predicting Tip.

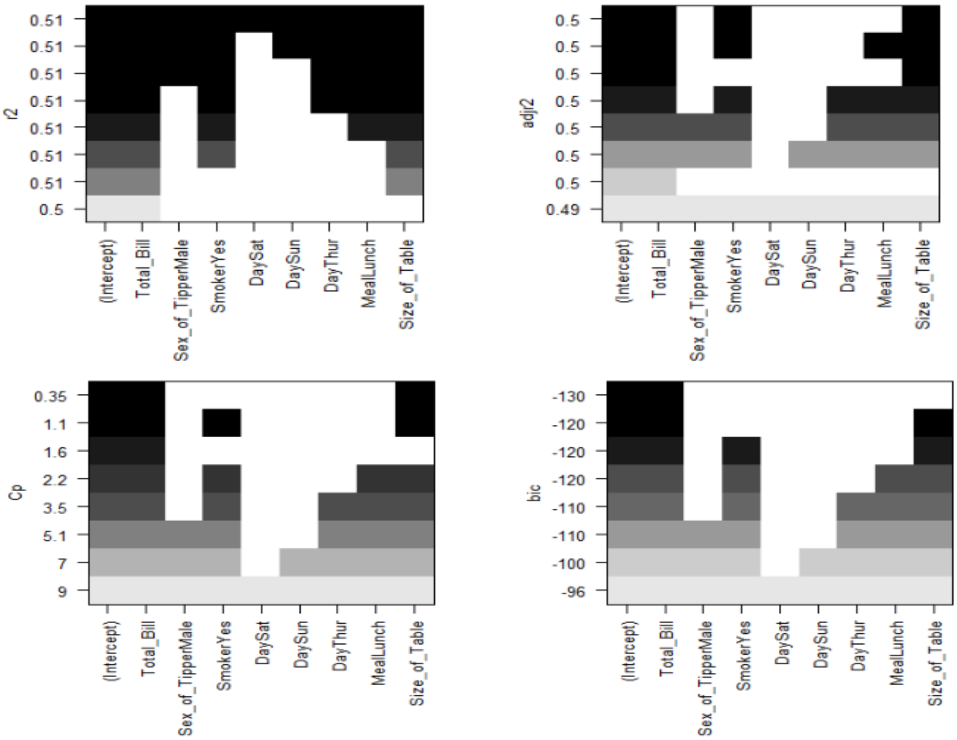


Figure 3.2: *RSS, adjusted R^2, C_p and BIC best variable plot.*

Subsequently, model5 is created with Tip regressed against Total Bill and Size of Table. It is an improvement on the base MSE, with MSE_5 = 1.604321

3.2.3 Forward and Backward Stepwise Selection

As in 3.2.2, forward and backward stepwise selection yields a table with asterisks corresponding to the best variables to use in a model, depending on the number of variables required. The tables for both the forward and backward method yielded the exact same table as Table 3.3, as well as the exact coefficients for each predictor variable calculated in best subset selection.

According to this method, Total Bill, SmokerYes and Size of Table are the most influential variables. Subsequently, model6 was created with Tip regressed against all three predictors. An MSE_6 with 1.66994 was achieved, an improvement on the base MSE.

3.2.4 Validation Set and Cross-Validation

Plotting mean cross-validation errors against the number of variables (Figure 3.3) shows that this min error is reached when one predictor is used. This method suggests that only Total Bill is used to regress Tip, the same as model2.

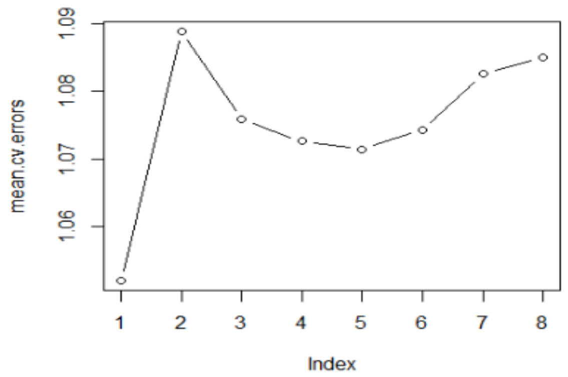


Figure 3.3: *Mean cross validation error against number of variables.*

3.2.5 Ridge Regression

As stated in 2.2.5, ridge regression uses all variables to predict the response variable. A tuning parameter is selected in Figure 3.4 using only the training set. The regression is the refitted on the full set, using the same λ chosen by cross validation.

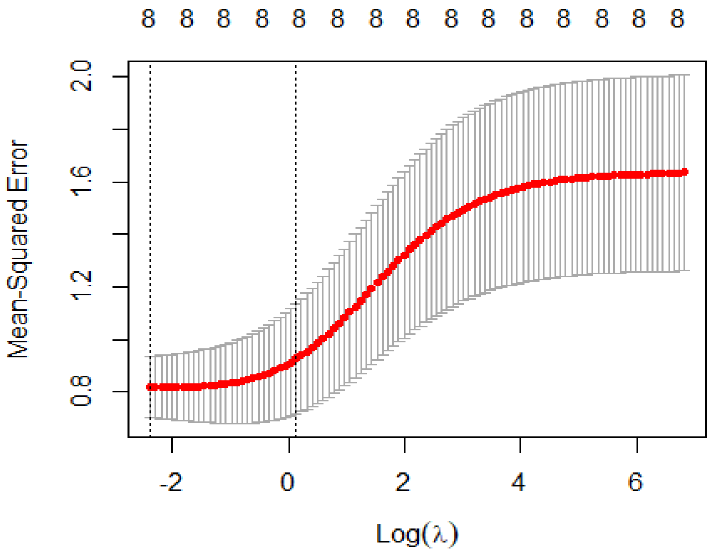


Figure 3.4: *Cross-validation used to predict the tuning parameter λ.*

The resulting ridge regression model has an improved MSE, however fitting all eight predictor variables would lead to overfitting and high variance.

3.2.6 LASSO

The LASSO method performs both variable selection and regularization such that prediction accuracy and interpretability is maximised. In Figure 3.5, coefficients are plotted against different values of *λ*. Cross-validation is conducted to find the lambda corresponding to the lowest test MSE (Figure 3.6).

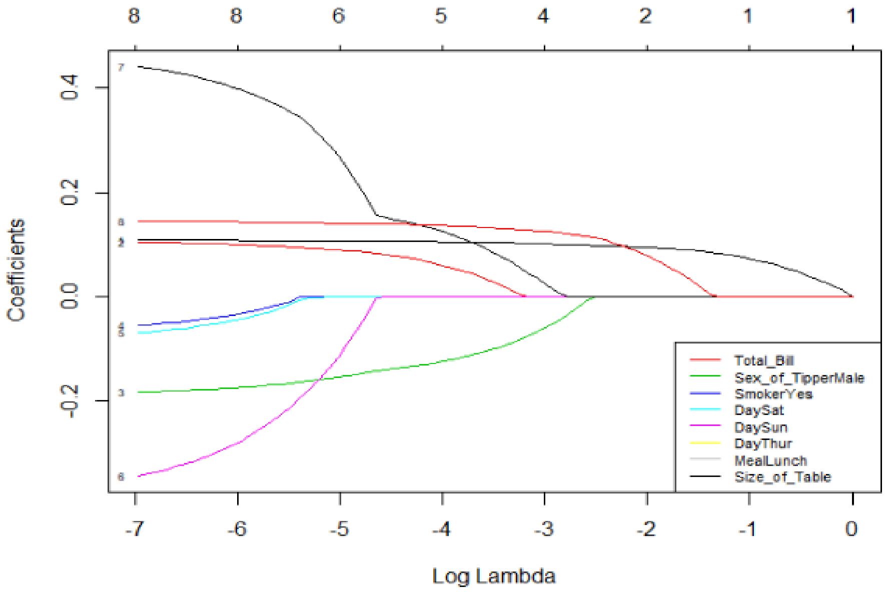


Figure 3.5: *Coefficient plot against the tuning parameter λ.*

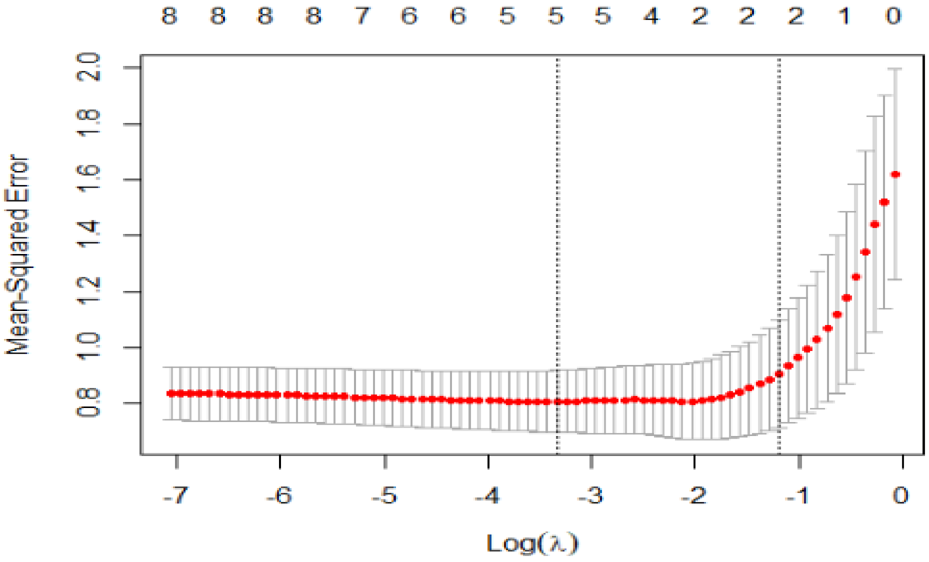


Figure 3.6: *Cross-validation used to predict the tuning parameter λ.*

The resulting model7 contains five predictor variables: Total Bill, Sex_of_TipperMale, SmokerYes, MealLunch and Size_of_Table. Table 3.4 expresses the coefficients of the variables proposed by the LASSO method.

(Intercept)	Total_Bill	SexMale	SmokerYes	MealLunch	Size
0.5996	0.1027	0.0128	-0.0904	0.0662	0.1295

Table 3.4: *Predictor variables and their coefficients determined by LASSO.*

The associated MSE_7 = 1.28043 is the best test MSE out of all the fitted models. The issue may arise that the model is overfitting the data and introducing high variance, since more variables are used than suggested by the subset selection methods.

3.2.7 Interaction Terms

As discussed in 2.2.7, interaction effect represents the combined effects of predictors on the response variable. Their inclusion in a model often exposes previously hidden nuances in the dataset.

In model8, Tip was regressed on three variables: The Size of Table and the interaction between Total Bill and Meal. The resulting test MSE_8 = 1.578557.

In model9, Tip was regressed on four variables: The Size of Table, Meal and the interaction between Total Bill and Smoker. The resulting test MSE_9 = 1.575977. The model's accuracy is only marginally better than model8's. Since model8 uses three variables, its interpretability is favoured over the marginal decrease in test MSE in model 9.

3.3 Residual Diagnostics

Ideally, a residual vs fitted plot should be reasonably shapeless, have a visible symmetric frequency distribution around the 0 position. There are no obvious outliers or clear patterns, hence the model is deemed acceptable.

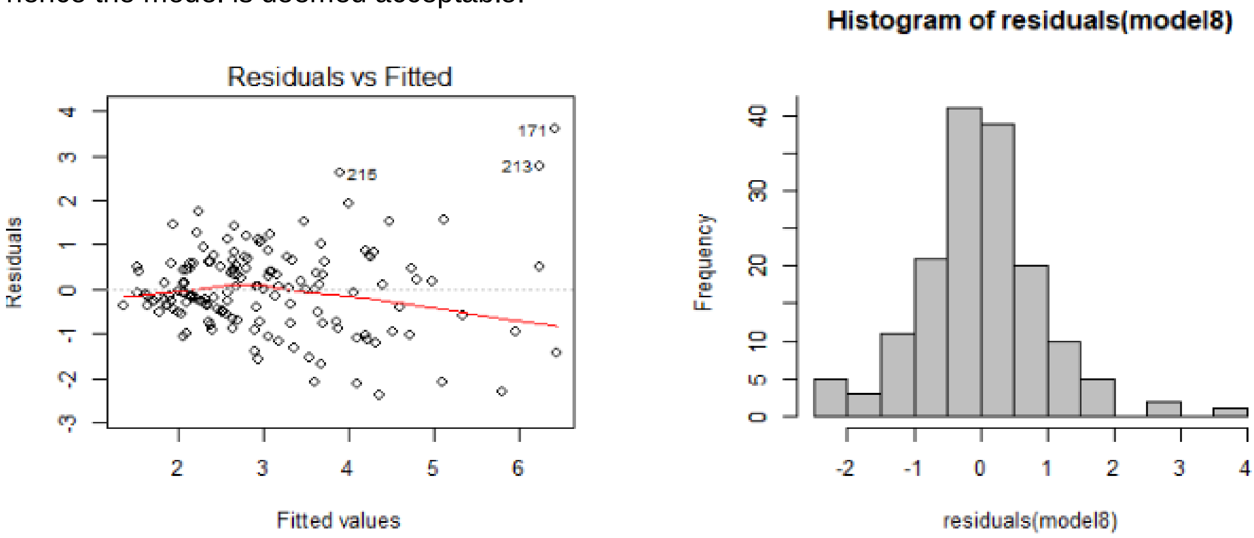


Figure 3.7: *Residuals vs Fitted values and its frequency distribution for model8*

Figure 3.8 shows the Q-Q plot for model8. The shape of the points on the graph seem to be a combination of a normal and light-tailed distribution. No extreme outliers are visible and the relationship seems reasonably straight and acceptable

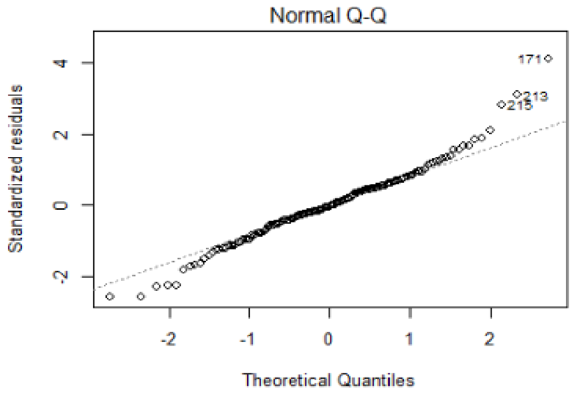


Figure 3.8: QQ plot for model8

Chapter 4

Conclusion

Initially, a multiple linear regression model was used to regress the variable Tip on the remaining seven predictor variables in the “tips” dataset. The remaining variables are Total Bill, Sex of Tipper, Smoker, Day of the Week, Meal and Size of the Table. The model fitted was named “model1”, and its associated test MSE was used as a baseline to judge all subsequent models. MSE improvement is based on the new model’s test MSE compared to the base MSE in model1.

In order to improve on the initial model, multiple variable selection techniques were performed. These techniques included Best Subset Selection, Forward and Backward Subset Selection, Validation Set and Cross-Validation, Ridge Regression, LASSO, as well as including Interaction terms in order to improve the predictive power of the model. In total, nine models were developed.

Model	Form	MSE	MSE Improvement
1	Tip ~ .	1.823752	-
2	Tip ~ Total_Bill	1.647222	9.67946
3	Tip ~ Size_of_Table	1.890773	-3.674934
4	Tip ~ Meal	2.118516	-16.16253
5	Tip ~ Total_Bill + Size_of_Table	1.604321	12.03185
6	Tip ~ Total_Bill + Size_of_Table + Smoker	1.66994	8.433794
7	See Table 3.4	1.280429	29.7915
8	Tip ~ Total_Bill*Meal + Size_of_Table	1.578557	13.44454
9	Tip ~ Total_Bill*Smoker + Meal + Size_of_Table	1.575977	13.58599

Table 4.1: *Comparison of model forms and test MSE values.*

The best model in terms of MSE reduction was the LASSO model with five variables. It was however deemed that model7 made use of too many variables and risked introducing high variance due to overfitting. The next best performing model is model9 comprised of four variables and an interaction term. Model9 and model8 have virtually the same test MSE values. Since model8 only uses three variables, it was deemed more appropriate than model9 due to its simplicity and interpretability. Therefore, model8 is used as the best model to predict average tip (Table 4.2).

	X_1	X_2	X_3	X_1*X_2	Intercept
	Total Bill	MealLunch	Size of Table	Total Bill* MealLunch	-
β_j	0.1111	-0.0419	0.1055	0.0079	0.4476

Table 4.2: *Coefficient and intercept for final model.*

A-0: Descriptive stats code

```
rm(list=ls())
setwd("C:/Users/user/Desktop/FinTech 2020/
Supervised Learning/Assignment 1")

#Read in all 244 observations
tips <- read.csv("tipdata.csv", header = TRUE)
tips$X <- NULL #drop X column
head(tips)

#rename columns
colnames(tips) <- c("Total_Bill", "Tip", "Sex of Tipper", "Smoker",
                    "Day", "Meal", "Party Size")
head(tips)

#create "Tip Percent"
tip_percent <- (tips$'Tip' / tips$'Total_Bill')*100
tips[, "Tip Percent"] <- tip_percent
head(tips)

#create 200 observation sample
set.seed(23)
tips <- tips[sample(1:nrow(tips),200, replace=FALSE), ]
head(tips)
dim(tips)

#remove outliers [134] & [48] as per Tukey's and SD method
tips <- tips[-c(134, 48),]
head(tips)
dim(tips) #confirm correct # observations after row removals (198)

#set graphical parameters
par(mfrow=c(1,1))

#frequency distribution of Tip Percent
hist(tips$`Tip Percent`, main = "Frequency of Tips",
     xlab = "Tip Percent",breaks = 40, col = 'skyblue3', xaxt='n')
axis(side=1, at=seq(0,100,2.5), labels=seq(0,100,2.5))

#table 1.1
summary(tips)
summary(tips)
summary(tips$Total_Bill)
summary(tips$Tip)
summary(tips$`Tip Percent`)
summary(tips$`Party Size`)

#find the mode of Party Size
sort(table(tips$`Party Size`),decreasing=TRUE)[1:6]
```

```
#find the mode of Tip
sort(table(tips$Tip),decreasing=TRUE)[1:6]

#scatterplot of "Tip Percent" vs "Total Bill" with trend line
plot(tips$`Tip Percent` ~ Total_Bill,
     data = tips, main="Tip Percent vs Total Bill",
     xlab = "Total_Bill", ylab = "Tip Percent")
abline(lm(tips$`Tip Percent`~tips$Total_Bill), col="red")
cor(tips$Total_Bill, tips$`Tip Percent`)

#scatterplot of "Total Bill" vs "Party Size" with trend line
plot(tips$Total_Bill ~ tips$`Party Size`,
     data = tips, main="Total Bill vs Party Size",
     xlab = "Party Size", ylab = "Total Bill")
abline(lm(tips$Total_Bill~tips$`Party Size`), col="red")

#scatterplot of "Tip Percent vs Total Bill by Party Size"
require(ggplot2)
ggplot(tips, aes(x = `Total_Bill`, y =  `Tip Percent`)) +
  geom_point(aes(color = `Party Size`)) +
  xlab("Total Bill") + ylab("Tip Percent") +
  scale_y_continuous(name="Tip Percent",
                     limits=c(0, 30), breaks=seq(0,40,5)) +
  ggtitle("Tip Percent vs Total Bill by Party Size")

#ggplot2 scatterplot
ggScatPlot <- ggplot(tips,aes(x = Total_Bill,y = `Tip Percent`)) +
  geom_point()  #basic scatterplot
ggScatPlot <- ggScatPlot + geom_point(aes(color = Meal)) +
  scale_y_continuous(name="Tip Percent",
                     limits=c(0, 30), breaks=seq(0,40,5))
ggScatPlot <- ggScatPlot + facet_grid(~ Meal)
ggScatPlot <- ggScatPlot + geom_smooth(method = 'lm',formula = y~x)
ggScatPlot <- ggScatPlot +
  ggtitle("Tip Percent vs Total Bill by Meal Time") +
  xlab("Total Bill") #add title
ggScatPlot #output graphic

#boxplots of predictor variables
boxplot(`Tip Percent`~`Sex of Tipper`,data=tips,
        main="Sex of Tipper Data",
        xlab="Sex of Tipper", ylab="Tip Percent")
boxplot(`Tip Percent`~`Smoker`,data=tips, main="Smoker Data",
        xlab="Smoker", ylab="Tip Percent")
boxplot(`Tip Percent`~`Day`,data=tips, main="Day of Week Data",
        xlab="Day", ylab="Tip Percent")
boxplot(`Tip Percent`~`Meal`,data=tips, main="Mealtime Data",
        xlab="Meal", ylab="Tip Percent")
boxplot(`Tip Percent`~`Party Size`, data=tips,
        main="Size of Table Data",
        xlab="Size of Table", ylab="Tip Percent")
```

A-1: Multiple Linear Regression

```
rm(list=ls())
setwd("C:/Users/user/Desktop/FinTech 2020/Supervised
Learning/Assignment 1")

### Q1 ###
#Multiple linear regression

#Read in all 244 observations
fulldata <- read.csv("tipdata.csv", header = TRUE)
head(fulldata)

#Drop "X" column
drops <- c("X")
fulldata <- fulldata[ , !(names(fulldata) %in% drops)]
head(fulldata)

#Rename columns
colnames(fulldata) <- c("Total_Bill", "Tip", "Sex_of_Tipper",
"Smoker",
                        "Day", "Meal", "Size_of_Table")
head(fulldata)

#Create "Tip_Percent" column
Tip_Percent <- (fulldata$Tip / fulldata$Total_Bill)*100
fulldata[, "Tip_Percent"] <- Tip_Percent
head(fulldata)
fulldata <- fulldata[, c(1, 2, 8, 3, 4, 5, 6, 7)]
head(fulldata)

#Create 200 observation sample
set.seed(23)
tip_data <- fulldata[sample(1:nrow(fulldata),200, replace=FALSE), ]
head(tip_data)
dim(tip_data)

#stet graphical parameter
par(mfrow=c(2,2))

#Tukeys test for outliers
summary(tip_data$Tip_Percent)
tukeys_IQR <- IQR(tip_data$Tip_Percent)
Q_1 <- quantile(tip_data$Tip_Percent, 0.25)
Q_3 <- quantile(tip_data$Tip_Percent, 0.75)
tukeys_outlier <- Q_3 + 1.5*tukeys_IQR
tukeys_farout <- Q_3 + 3*tukeys_IQR
tukeys_outlier
tukeys_farout

#Standard deviation check for outliers
```

```
SD <- sd(tip_data$Tip_Percent)
x_bar <- mean(tip_data$Tip_Percent)
SD_Max <- x_bar + 3*SD
SD_Max
tukeys_farout

#reset the dataset: without outliers (entries [134],[48])
tip_data <- tip_data[-c(134, 48),]
head(tip_data)
dim(tip_data) #confirm correct dimenions after row removals

#Create train and test sets excluding outliers and Tip Percent
tip_data <- tip_data[,-c(3)]
head(tip_data)
set.seed(23)
ind <- sample(1:nrow(tip_data), 0.8*nrow(tip_data))
train <- tip_data[ind,]
test <- tip_data[-ind,]

#conduct multiple linear regression
model1 <- lm(Tip ~ .,data = train)
model1
summary(model1)

#use MSE to measure accuracy of the model on the test set
yhat1 <- predict(model1, newdata = test)
tip.test <- test$Tip
plot(tip.test, yhat1)
abline(0,1)
MSE_1 <- mean((yhat1-tip.test)^2) #MSE
MSE_1
```

A-2: Model Improvement by Variable Selection

```
#Model improvement by variable selection
#Simple linear regression

SLR_coeff1 <- lm(Tip ~ Total_Bill,data = train)
summary(SLR_coeff1)

SLR_coeff2 <- lm(Tip ~ Sex_of_Tipper,data = train)
summary(SLR_coeff2)

SLR_coeff3 <- lm(Tip ~ Smoker,data = train)
summary(SLR_coeff3)

SLR_coeff4 <- lm(Tip ~ Day,data = train)
summary(SLR_coeff4)

SLR_coeff5 <- lm(Tip ~ Meal,data = train)
summary(SLR_coeff5)

SLR_coeff6 <- lm(Tip ~ Size_of_Table,data = train)
summary(SLR_coeff6)

model2 <- lm(Tip ~ Total_Bill, data = train)
summary(model2)

yhat2 <- predict(model2, newdata = test)
plot(tip.test, yhat2)
abline(0,1)
MSE_2 <- mean((yhat2-tip.test)^2) #MSE
MSE_2

model3 <- lm(Tip ~ Size_of_Table, data = train)
summary(model3)

yhat3 <- predict(model3, newdata = test)
plot(tip.test, yhat3)
abline(0,1)
MSE_3 <- mean((yhat3-tip.test)^2) #MSE
MSE_3

model4 <- lm(Tip ~ Meal, data = train)
summary(model4)

yhat4 <- predict(model4, newdata = test)
plot(tip.test, yhat4)
abline(0,1)
MSE_4 <- mean((yhat4-tip.test)^2) #MSE
```

```
MSE_4

#Best Subset Selection

#indicates best predictive variables
library(leaps)
#gives data for 8 best variables in the models
regfit.full = regsubsets(Tip ~ ., data = tip_data)
summary(regfit.full)

#for max 8 variables
reg.summary = summary(regfit.full)
regfit.full = regsubsets(Tip ~., data = tip_data, nvmax = 8)

#plotting RSS and adjusted R^2 vs # of variables
par(mfrow =c(2,2))
plot(reg.summary$rss, xlab = "Number of Variables",
     ylab = "RSS", type = "l")
plot(reg.summary$adjr2, xlab = "Number of Variables",
     ylab = "Adjusted R^2", type = "l")

#plot max point on adjusted R^2
which.max(reg.summary$adjr2)
points(2, reg.summary$adjr2[2], col = "red", cex = 2, pch = 20)

#plot Cp vs # variables, red dot at min point
plot(reg.summary$cp, xlab = "Number of Variables", ylab = "Cp",
     type = "l")
which.min(reg.summary$cp)
points(2, reg.summary$cp[2], col = "red", cex = 2, pch = 20)

#plot BIC vs # variables, red dot at min point
plot(reg.summary$bic, xlab = "Number of Variables", ylab = "BIC",
     type="l")
which.min(reg.summary$bic)
points(1, reg.summary$bic[1], col = "red", cex = 2, pch = 20)

par(mfrow=c(2,2))
#regfit legoblock display of variables and stats
plot(regfit.full, scale = "r2", col=gray(seq(0, 0.9,
                                             length = 10)))
plot(regfit.full, scale = "adjr2")
plot(regfit.full, scale = "Cp")
plot(regfit.full, scale = "bic")

#gives coefficients for best number variables used (1-8)
coef(regfit.full, 1)
coef(regfit.full, 2)
coef(regfit.full, 3)
coef(regfit.full, 4) #max number of coefficients considered
```

```
model5 <- lm(Tip ~ Total_Bill + Size_of_Table, data = train)
summary(model5)

yhat5 <- predict(model5, newdata = test)
plot(tip.test, yhat5)
abline(0,1)
MSE_5 <- mean((yhat5-tip.test)^2) #MSE
MSE_5

#Forward and Backward Stepwise Selection

#indicates best predictive variables
regfit.fwd = regsubsets(Tip ~., data = tip_data, nvmax = 8,
                        method = "forward")
summary(regfit.fwd)
regfit.bwd = regsubsets(Tip ~., data = tip_data, nvmax = 8,
                        method = "backward")
summary(regfit.bwd)

#full/fwd/bwd yields same variables and coefficients
#1 variable used
coef(regfit.full, 1)
coef(regfit.fwd, 1)
coef(regfit.bwd, 1)

#2 variables used
coef(regfit.full, 2)
coef(regfit.fwd, 2)
coef(regfit.bwd, 2)

#3 variables used
coef(regfit.full, 3)
coef(regfit.fwd, 3)
coef(regfit.bwd, 3)

model6 <- lm(Tip ~ Total_Bill + Size_of_Table + Smoker, data =
train)
summary(model6)

yhat6 <- predict(model6, newdata = test)
plot(tip.test, yhat6)
abline(0,1)
MSE_6 <- mean((yhat6-tip.test)^2) #MSE
MSE_6

#Validation Set Approach and Cross-Validation

#Set seed to split train and test data sets
set.seed(23)
cv.train = sample(c(TRUE,FALSE), nrow(tip_data), rep = TRUE)
cv.test = (!cv.train)
```

```
#perform best subset selection
regfit.best = regsubsets(Tip ~., data = tip_data[cv.train ,], nvmax
= 8)

#make a model matrix from the test data
test.mat = model.matrix(Tip ~., data = tip_data[cv.test ,])

# Run a loop and for each i, extract the coefficients from
# regfit.best for the best model of that size,
# multiply into the appropriate columns of the test matrix
# to form the predictions, and compute the test MSE.

val.errors = rep(NA ,8)

for(i in 1:8) {
  coefi = coef(regfit.best, id = i)
  pred = test.mat[,names(coefi)] %*% coefi
  val.errors[i] = mean((tip_data$Tip[cv.test] - pred)^2)
}
val.errors #test MSE

#best model is the one that contains 1 variable
which.min(val.errors)
coef(regfit.best, 1)

#now run best subset selection on full data set (previously train)
regfit.best = regsubsets(Tip ~ ., data = tip_data, nvmax = 8)
coef(regfit.best ,1)

#Choose among the models of different sizes using cross-validation

#k = 10 folds,create a matrix to store the results
k = 10
set.seed(23)
folds = sample(1:k, nrow(tip_data), replace = TRUE)
cv.errors = matrix(NA, k, 8, dimnames = list(NULL, paste(1:8)))

predict.regsubsets <- function(object, newdata, id) {
  form <- as.formula(object$call[[2]])
  mat <- model.matrix(form, newdata)
  coefi <- coef(object, id = id)
  xvars <- names(coefi)
  mat[,xvars]%*%coefi
}

for(j in 1:k){
  best.fit = regsubsets(Tip ~ ., data = tip_data[folds != j,], nvmax
= 8)
  for(i in 1:8) {
    pred = predict.regsubsets(best.fit, tip_data[folds == j,], id =
i)
    cv.errors[j,i] = mean((tip_data$Tip[folds == j] - pred)^2)
  }
}

#mean cross-validation error
mean.cv.errors = apply(cv.errors, 2, mean)
mean.cv.errors

#plot of mean cv errors vs number of variables
par(mfrow = c(1,1))
plot(mean.cv.errors, type = "b")

#We now perform best subset selection on the full data set
reg.best = regsubsets(Tip ~., data = tip_data, nvmax = 8)
coef(reg.best, 1)
coef(reg.best, 2)
coef(reg.best, 3)
coef(reg.best, 4)
coef(reg.best, 5)

#Ridge Regression

#Pass in an x matrix and y vector
x = model.matrix(Tip ~ .,tip_data )[,-1]
y = tip_data$Tip

#build a Ridge Regression model
library(glmnet)
grid = 10^seq(10, -2, length = 100)
ridge.mod = glmnet(x, y, alpha = 0, lambda = grid)
dim(coef(ridge.mod))

#split the samples into a training set and a test set
set.seed(23)
rr_train = sample(1:nrow(x), nrow(x)/2)
rr_test = (-rr_train)
y.test = y[rr_test]

#use cross-validation to choose the tuning parameter λ
set.seed(23)
cv.out = cv.glmnet(x[rr_train,], y[rr_train], alpha = 0)
plot(cv.out)
extralam = cv.out$lambda.min
bestlam_rr = cv.out$lambda.1se #use this one
bestlam_rr

#test MSE associated with this value of λ
ridge.pred = predict(ridge.mod, s = bestlam_rr , newx = x[rr_test,])
mean((ridge.pred - y.test)^2)
```

```
#Refit the Ridge Regression model on the full data set,
#using the value of λ chosen by cross-validation,
#examine the coefficient estimates for # of variables used

out = glmnet(x, y, alpha = 0)
predict(out, type = "coefficients", s = bestlam_rr)[1:9,]

#LASSO

#build a LASSO model
lasso.mod = glmnet(x, y, alpha = 1, standardize = TRUE)
plot(lasso.mod, label = TRUE, xvar = "lambda")
legend("bottomright", lwd = 1, col = 2:11, legend = colnames(x), cex
= .7)

#perform cross-validation and compute test error
set.seed (23)
cv.out = cv.glmnet(x[rr_train,], y[rr_train], alpha = 1)
plot(cv.out)
bestlam_lasso = cv.out$lambda.min
lasso.pred = predict(lasso.mod, s = bestlam_lasso, newx =
x[rr_test,])
MSE_7 <- mean((lasso.pred - y.test)^2)
MSE_7

out = glmnet(x, y, alpha = 1, lambda = grid)
lasso.coef = predict(out, type = "coefficients",s =
bestlam_lasso)[1:9,]
lasso.coef
model7 <- lasso.coef[lasso.coef != 0]


#introducing interaction terms

model8 <- lm(Tip ~ Total_Bill*Meal + Size_of_Table, data = train)
summary(model8)

yhat8 <- predict(model8, newdata = test)
plot(tip.test, yhat8)
abline(0,1)
MSE_8 <- mean((yhat8-tip.test)^2) #MSE
MSE_8

model9 <- lm(Tip ~ Total_Bill*Smoker + Meal + Size_of_Table, data =
train)
summary(model9)

yhat9 <- predict(model9, newdata = test)
plot(tip.test, yhat9)
abline(0,1)
```

```
MSE_9 <- mean((yhat9-tip.test)^2) #MSE
MSE_9

MSE_reduction_percent <- function(MSE_n) {
  percent_reduction <- (MSE_1 - MSE_n)/MSE_1*100
  return(percent_reduction)
}

MSE_reduction_percent(MSE_2)
MSE_reduction_percent(MSE_3)
MSE_reduction_percent(MSE_4)
MSE_reduction_percent(MSE_5)
MSE_reduction_percent(MSE_6)
MSE_reduction_percent(MSE_7)
MSE_reduction_percent(MSE_8)
MSE_reduction_percent(MSE_9)
```

A-3: Residual Diagnostics

```
#Residual diagnostic analysis

par(mfrow=c(2,2))

#residuals vs fitted values
plot(model8,1)
hist(residuals(model8), breaks = 20, col = "grey75")
ytick<-seq(0, 100, by=5)
axis(side=2, at=ytick, labels = TRUE)

#normal Q-Q plot
plot(model8,2)
```

References

Wiles, R. (2015, June) https://www.usatoday.com/story/money/2015/06/14/much-tip-depends/71137254/ Retrieved from URL

McLeod, S. A. (2019, May 20). What a p-value tells you about statistical significance. Simply Psychology. https://www.simplypsychology.org/p-value.html Retrieved from URL

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. An Introduction to Statistical Learning : with Applications in R. New York :Springer, 2017.

Jackson, S (2017, April) https://drsimonj.svbtle.com/ridge-regression-with-glmnet Retrieved from URL