Predicting Software Reselling Profits

Carrington Body

Necessary Packages: I need to load these packages and libraries to conduct my analysis.

```
#install.packages("tidyverse")
#install.packages("ggplot2")
library(ggplot2)
#install.packages("rlang")
#install.packages("ggpubr")
library(ggpubr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

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

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

```
#install.packages("caret")
library(MASS)
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.0
## v lubridate 1.9.2     v tibble    3.2.1
## v purrr     1.0.1     v tidyr     1.3.0
## v readr     2.1.4
```

```
## -- Conflicts ---------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x MASS::select()  masks dplyr::select()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(leaps)
#install.packages("rpart")
#install.packages("rpart.plot")
#install.packages("randomForest")
#install.packages("stargazer")
#install.packages("forecast")
#getwd()
```

Synopsis: Tayko Software is a software catalog firm that sells games and educational software. It started out as a software manufacturer and then added third-party titles to its offerings. It recently revised its collection of items in a new catalog, which it mailed out to its customers. This mailing yielded 2000 purchases. Based on these data, Tayko wants to devise a model for predicting the spending amount that a purchasing customer will yield. The file Tayko.csv contains information on 2000 purchases.

Table 1 describes the predictors (variables) that I will use in this problem (the Excel file contains additional predictors – that are insignificant).

Table 1: Description of Variables for Tayko Software

FREQ: Number of transactions in the preceding year *LAST_UPDATE*: Number of days since last update to customer record *WEB*: Whether customer purchased by Web order at least once *GENDER*: Male or female *ADDRESS_RES*: Whether it is a residential address *ADDRESS_US*: Whether it is a US address *SPENDING (response/target variable)*: Amount spent by customer in test mailing (in dollars)

Reading in the file Here, I will read in Tayko.csv and convert it to a data frame. Then, I will make a smaller data set by extracting the necessary predictors to be used in analysis (listed in Table 1).

```
Taykoo <- read.csv('Tayko.csv', header = TRUE)
Taykoo.df <- as.data.frame(Taykoo)
attach(Taykoo.df)
View(Taykoo.df)
sub_taykoo <- dplyr::select(Taykoo.df, Freq, last_update_days_ago, Web.order, Gender.male, Address_is_r
View(sub_taykoo)
head(sub_taykoo)
```

```
##   Freq last_update_days_ago Web.order Gender.male Address_is_res US Spending
## 1    2                3662         1           0              1  1      128
## 2    0                2900         1           1              0  1        0
## 3    2                3883         0           0              0  1      127
## 4    1                 829         0           1              0  1        0
## 5    1                 869         0           0              0  1        0
## 6    1                1995         0           0              1  1        0
```

Any missing data? I have to make sure that there is no missing data so that I can conduct my analysis. Many times, if there is missing data, you can proceed with analysis only if there is specific instruction on handling it.

```r
sum(is.na(sub_taykoo))
```

```
## [1] 0
```

```r
#This command checks if there are any NAs (missing values) in the data frame. Since it returned 0,
#this means that there are no missing values. Thus, we have no missing data and can continue
#with the procedure.
```

Checking the Structure of the Data

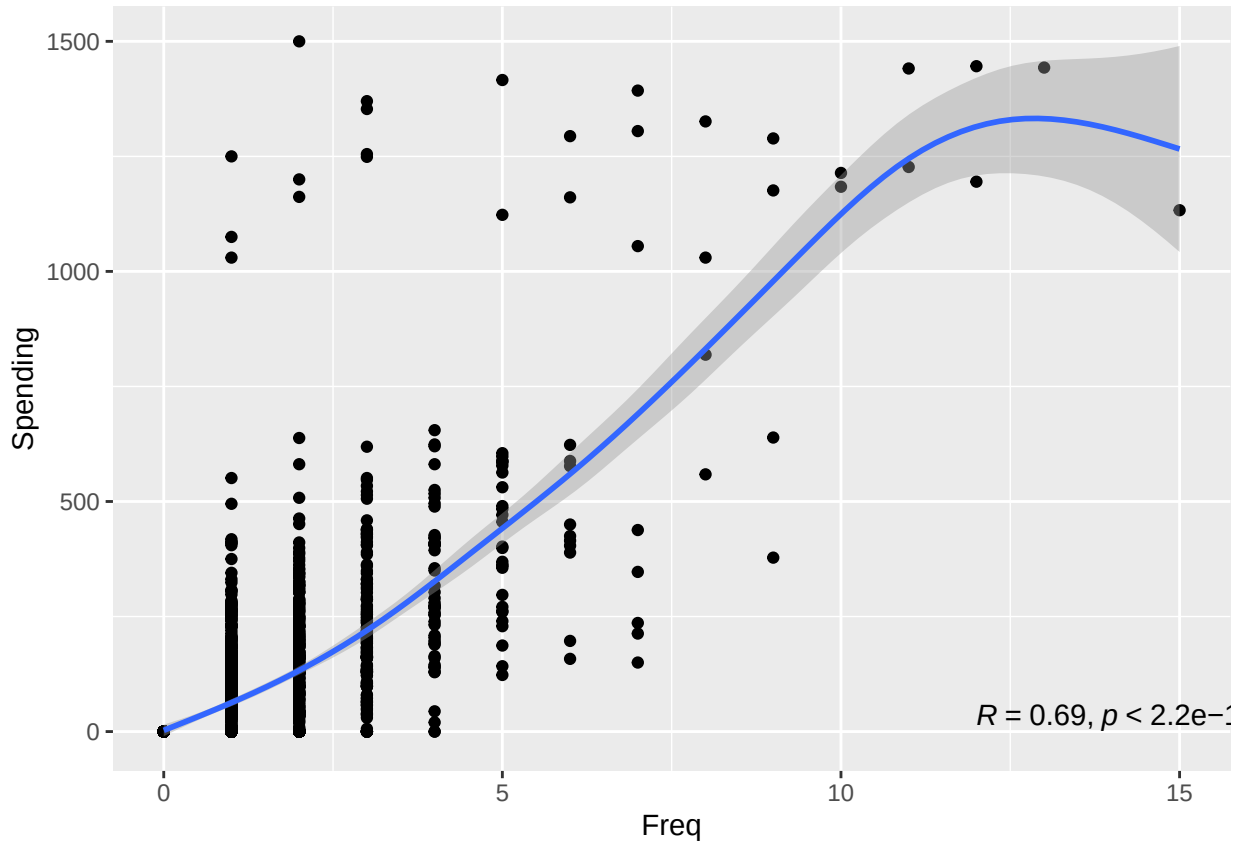
```r
str(sub_taykoo)
```

```
## 'data.frame':    2000 obs. of  7 variables:
##  $ Freq               : int  2 0 2 1 1 1 2 1 4 1 ...
##  $ last_update_days_ago: int  3662 2900 3883 829 869 1995 1498 3397 525 3215 ...
##  $ Web.order          : int  1 1 0 0 0 0 0 0 1 0 ...
##  $ Gender.male        : int  0 1 0 1 0 0 0 1 1 0 ...
##  $ Address_is_res     : int  1 0 0 0 0 1 1 0 0 0 ...
##  $ US                 : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Spending           : int  128 0 127 0 0 0 0 0 489 174 ...
```

Exploratory Data Analysis Here, I will explore the relationship between spending and each of the two continuous predictors by using ggplot to create two scatterplots (Spending vs. Freq, and Spending vs. last_update_days_ago).

```r
FreqSpend <- ggplot(sub_taykoo, aes(x = Freq, y = Spending)) + geom_point() + geom_smooth() + stat_cor(
LastSpend <- ggplot(sub_taykoo, aes(x = last_update_days_ago, y = Spending)) + geom_point() + geom_smoo
FreqSpend
```

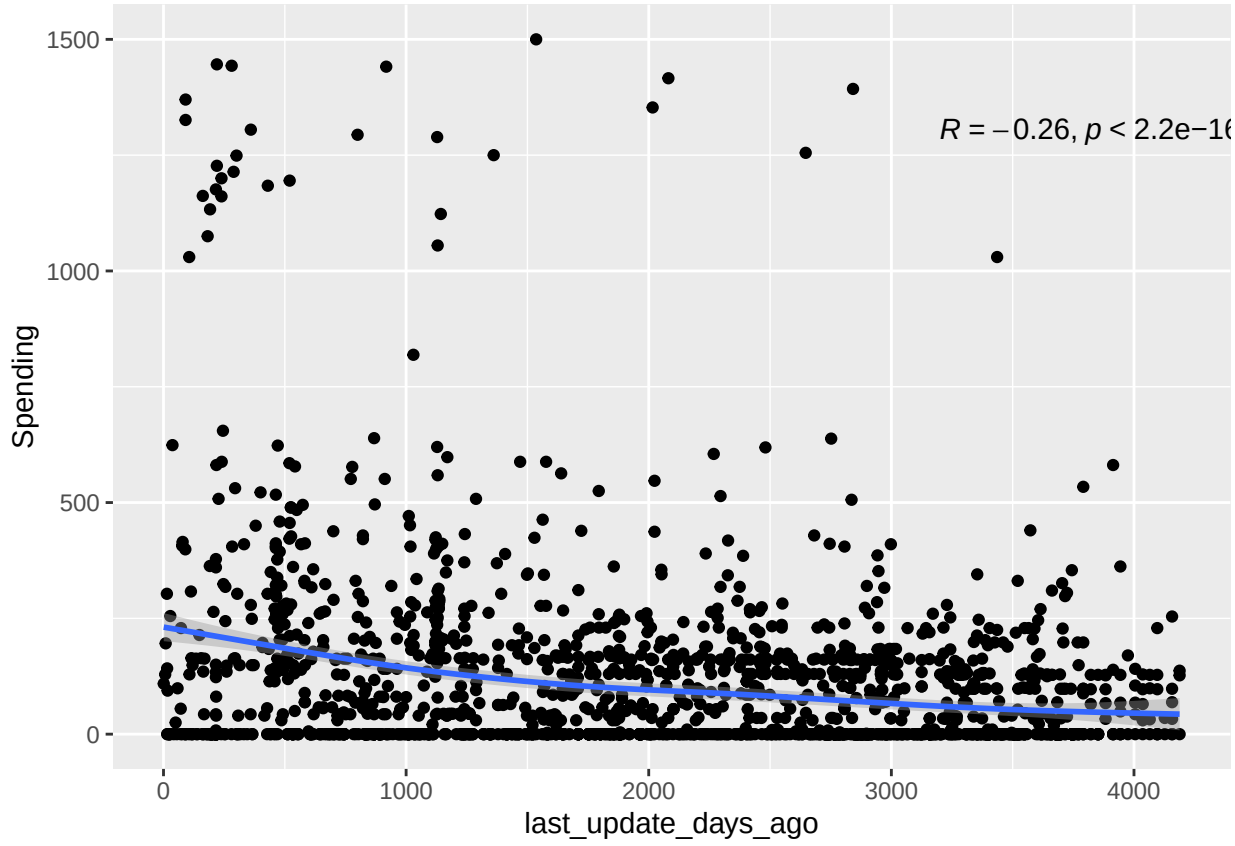
```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```




```
#There seems to be a positive, linear relationship between Spending and Freq. The correlation value
#between these two variables is 0.69, which is moderately strong. The p-value is nearly 0,
#so we can say that the correlation value is statistically significant. We can see that,
#once Freq hit 12-13 transactions, there seems to be a slight drop-off in the spending amount.
```

```
LastSpend
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```




```
#On the other hand, the relationship between Spending and Last Update Days Ago has a negative,
#but very low correlation with a value of -0.26 and p-value of nearly 0.
```

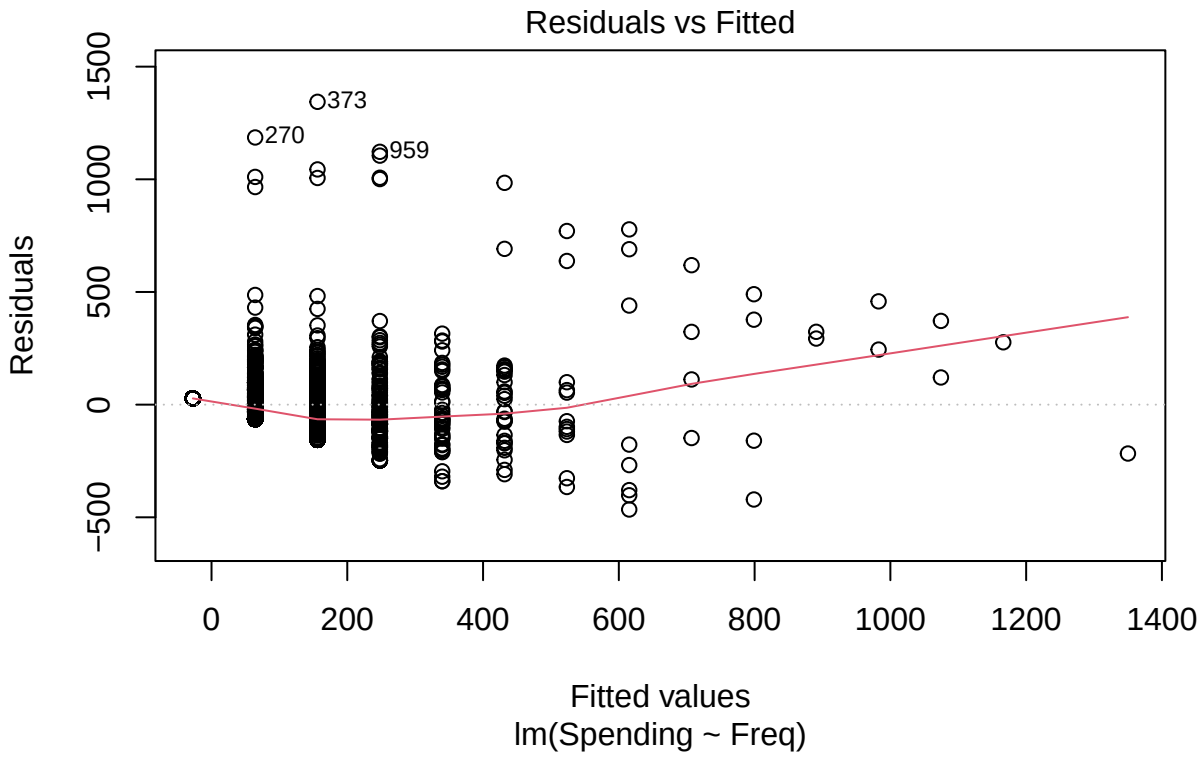
More Exploration & Evaluation on the Explanatory Variables Here, we can furthermore examine the linear relationship between Spending and the continuous variables Freq & Last_update_days_ago by running linear regression.

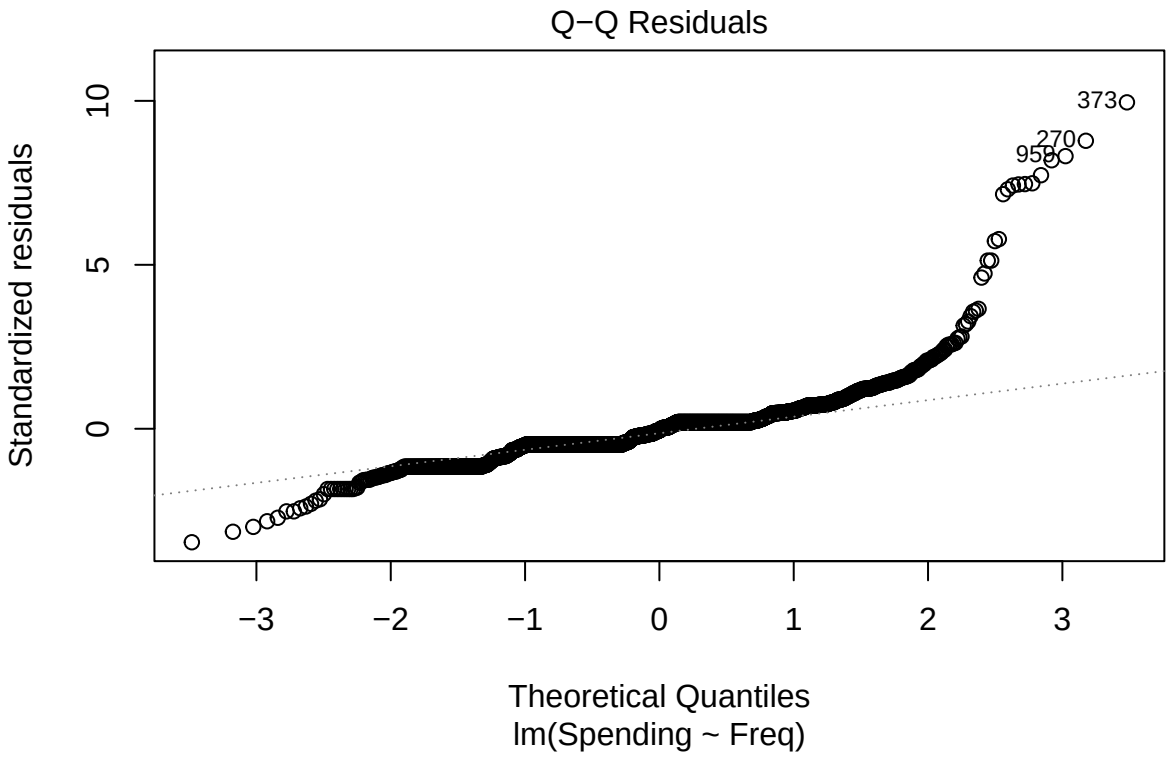
```
#Spending vs Freq
fit.freq <- lm(Spending~Freq, data = sub_taykoo)
summary(fit.freq)
```

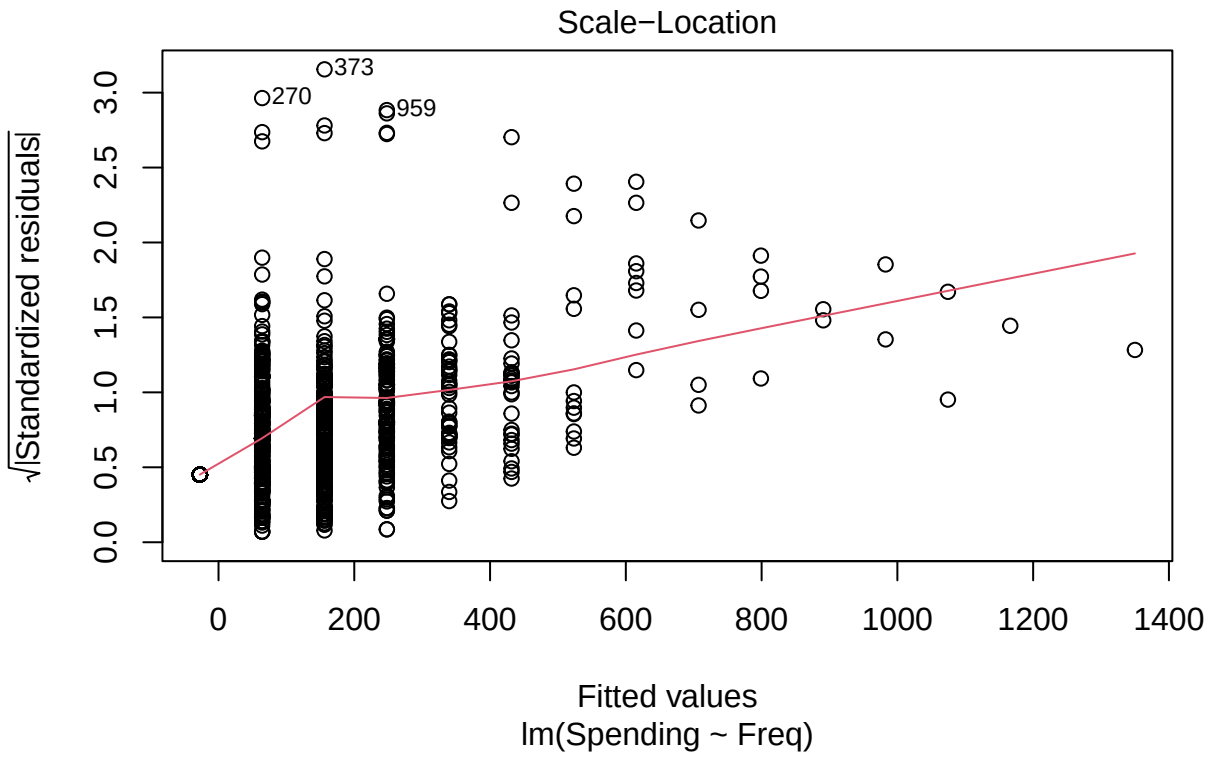
```
##
## Call:
## lm(formula = Spending ~ Freq, data = sub_taykoo)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -465.32  -64.33   -6.33   27.50 1343.84
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -27.500      4.288  -6.414 1.77e-10 ***
## Freq          91.831      2.148  42.744  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

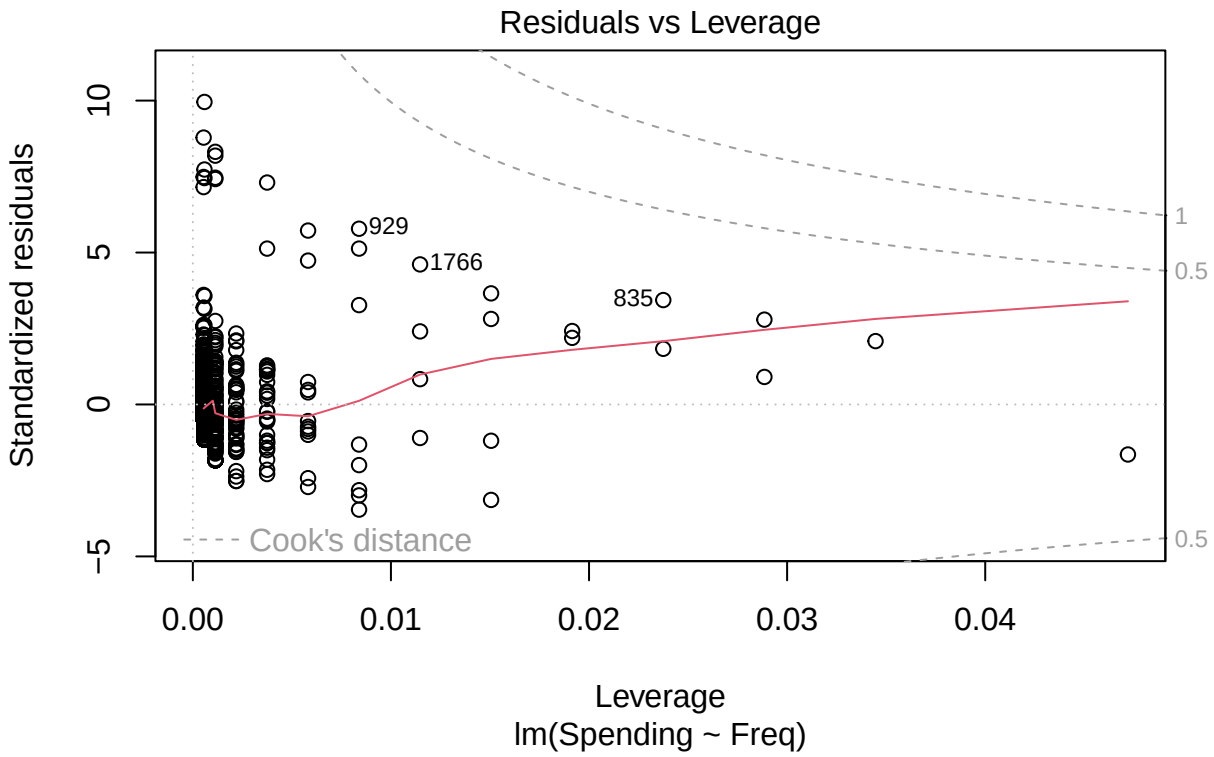
```
##
## Residual standard error: 135 on 1998 degrees of freedom
## Multiple R-squared:  0.4777, Adjusted R-squared:  0.4774
## F-statistic:  1827 on 1 and 1998 DF,  p-value: < 2.2e-16
```

```
plot(fit.freq)
```




Q−Q Residuals
Standardized residuals
10
5
0
373
270
95
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(Spending ~ Freq)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
3.0
373
270
959
0
200
400
600
800
1000
1200
1400
Fitted values
lm(Spending ~ Freq)



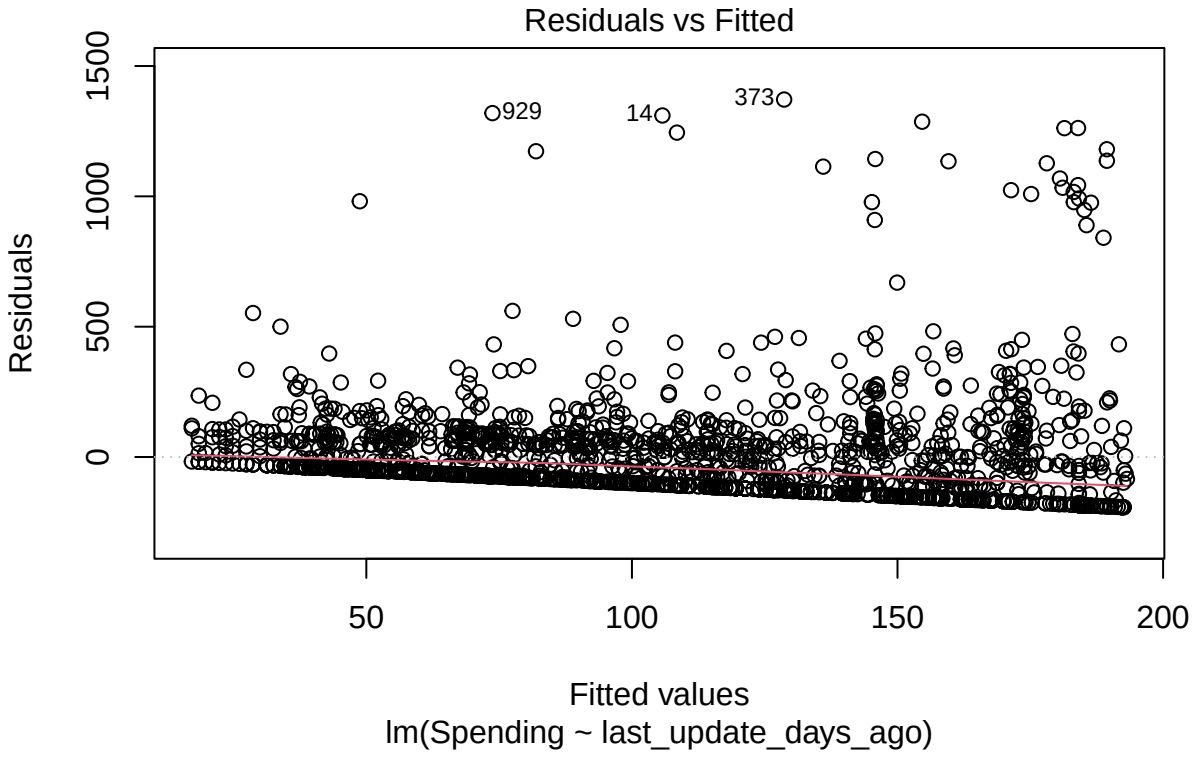

```
#Equation for this model: Spending(hat) = -27.50 + 91.83*Freq
#Adjusted R-squared is 0.4774, which means almost than half of the variability in Spending
#can be explained by Freq--because of this, we can see why Freq is a such a significant predictor
#in this model. In our residual plots, we see that we have nonconstant variance and a pattern
#in the scattering of residuals, so we may have dependency problem (whereas Spending is
#too dependent on Freq out of the six predictors we have to account for.) Since the p-value
#is low and less than alpha (0.05), we can say that this model somewhat fits data well and that
#there is a significant linear relationship between Spending and Freq.
```

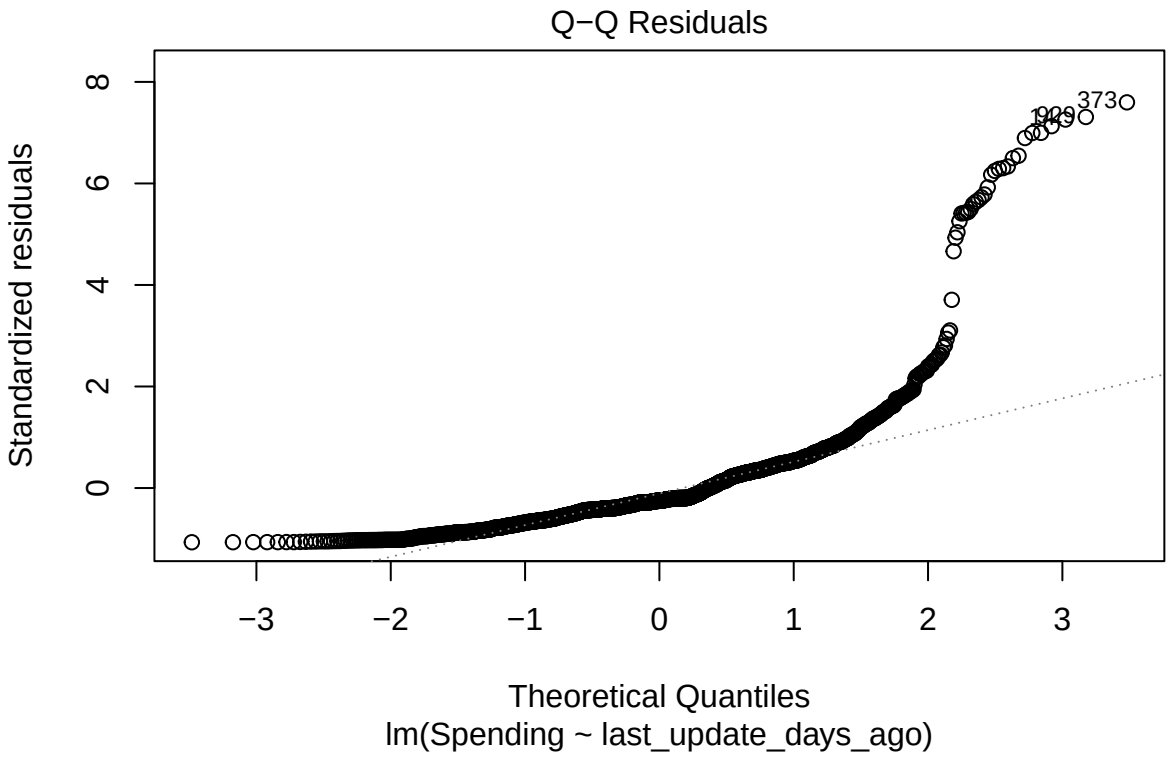
```
fit.last <- lm(Spending~last_update_days_ago, data = sub_taykoo)
summary(fit.last)
```

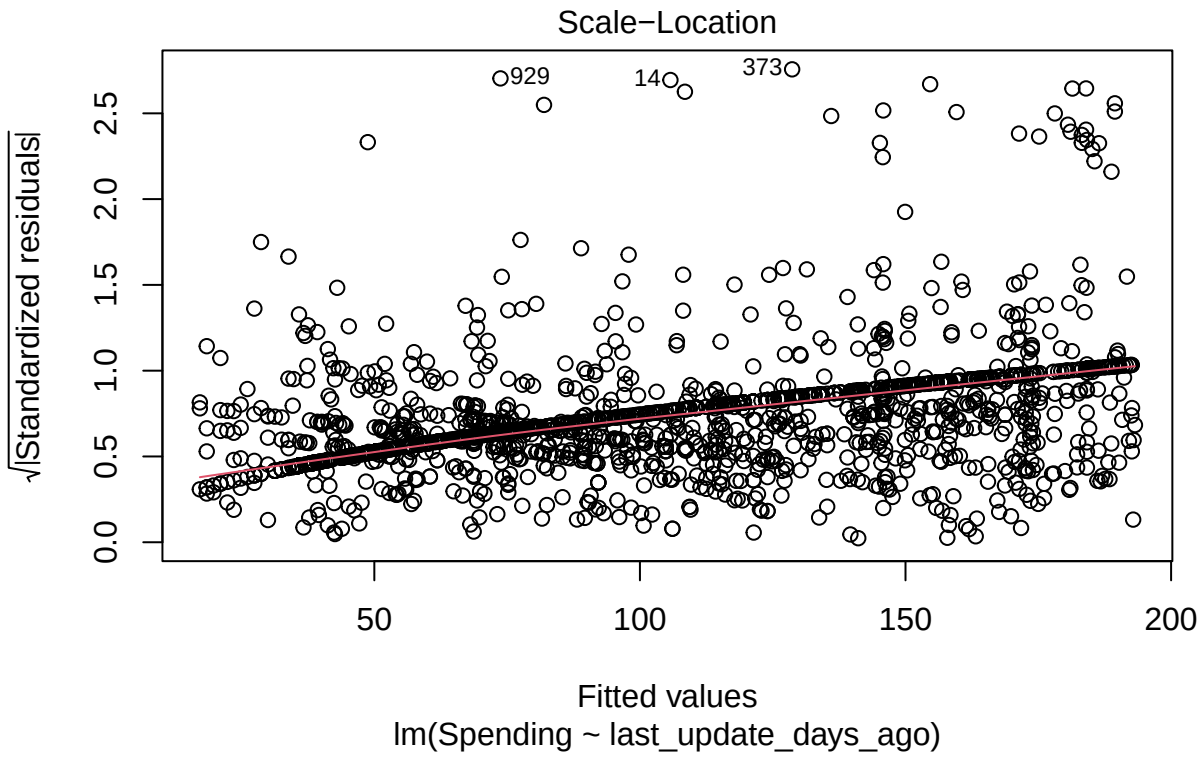
```
##
## Call:
## lm(formula = Spending ~ last_update_days_ago, data = sub_taykoo)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -192.61  -95.29  -45.15   56.68 1371.34
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           193.237411   8.628528   22.39   <2e-16 ***
## last_update_days_ago  -0.042046   0.003538  -11.88   <2e-16 ***
## ---
```

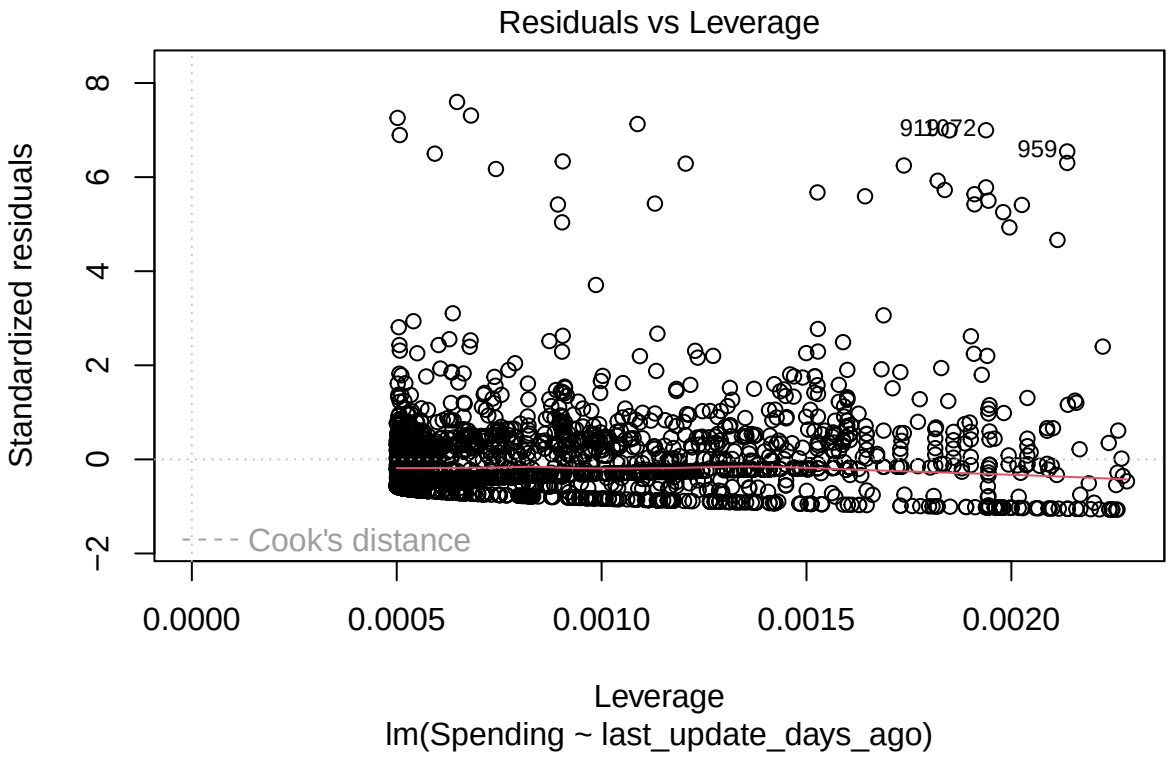
```
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 180.6 on 1998 degrees of freedom
## Multiple R-squared:  0.066,  Adjusted R-squared:  0.06554
## F-statistic: 141.2 on 1 and 1998 DF,  p-value: < 2.2e-16
```

```
plot(fit.last)
```




Q−Q Residuals
Standardized residuals
0
2
4
6
8
373
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(Spending ~ last_update_days_ago)


Scale−Location
929
14
373
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
50
100
150
200
Fitted values
lm(Spending ~ last_update_days_ago)




```
#Equation: 193.24 - 0.04 * last_update_days_ago
#The Adjusted R^2 for this model is 0.0655, which is very low. Only 6.5% of the variability in Spending
#can be explained by last_update_days_ago. This makes sense because of the poor linear relationship
#that is shown between Spending and last_update_days_ago. This model looks like it has a significant
#lack of fit, but it actually fits the data well considering what our analysis is based on.
#This model was bound to have a low R^2 value because of the existing poor linear relationship,
#compiled with the fact that we are trying to predict how much people would spend in test mailing.
#There is a significant linear relationship existing between Spending and last_update_days_ago because
#p-value of the model is low and less than alpha(0.05). CLARIFICATION: Because an independent variable
#has a poor relationship with the corresponding dependent variable, it does not mean the relationship
#between the two are insignificant.
```

Multicollinearity Issues? Here, we have to determine whether mulitcollinearity will be a problem since there are more binary predictors (4) than continuous predictors (2). We shall be mindful of this as it can potentially skew our results in predicting resell profits and disrupt the reliability of our statistical inferences.

```
#install.packages("Hmisc")
library(Hmisc)
```

```
##
## Attaching package: 'Hmisc'
```

```
## The following objects are masked from 'package:dplyr':
##
##     src, summarize
```

```
## The following objects are masked from 'package:base':
##
##     format.pval, units
```

```
tayko.corr.matrix <- rcorr(as.matrix(sub_taykoo))
tayko.corr.matrix #shows the correlation coefficient and respective p-values between all variables
```

```
##                       Freq last_update_days_ago Web.order Gender.male
## Freq                  1.00                -0.35      0.10       -0.04
## last_update_days_ago -0.35                 1.00     -0.03        0.02
## Web.order             0.10                -0.03      1.00       -0.01
## Gender.male          -0.04                 0.02     -0.01        1.00
## Address_is_res        0.22                -0.21     -0.04       -0.05
## US                    0.03                 0.04      0.00        0.03
## Spending              0.69                -0.26      0.12       -0.02
##                      Address_is_res   US Spending
## Freq                           0.22 0.03     0.69
## last_update_days_ago          -0.21 0.04    -0.26
## Web.order                     -0.04 0.00     0.12
## Gender.male                   -0.05 0.03    -0.02
## Address_is_res                 1.00 0.02    -0.03
## US                             0.02 1.00     0.00
## Spending                      -0.03 0.00     1.00
##
## n= 2000
##
##
## P
##                      Freq   last_update_days_ago Web.order Gender.male
## Freq                        0.0000               0.0000    0.0888
## last_update_days_ago 0.0000                      0.1188    0.4675
## Web.order            0.0000 0.1188                        0.7948
## Gender.male          0.0888 0.4675               0.7948
## Address_is_res       0.0000 0.0000               0.0759    0.0324
## US                   0.1392 0.0880               0.8561    0.2347
## Spending             0.0000 0.0000               0.0000    0.2826
##                      Address_is_res US     Spending
## Freq                 0.0000         0.1392 0.0000
## last_update_days_ago 0.0000         0.0880 0.0000
## Web.order            0.0759         0.8561 0.0000
## Gender.male          0.0324         0.2347 0.2826
## Address_is_res                      0.3521 0.2282
## US                   0.3521                0.8764
## Spending             0.2282         0.8764
```

```
#in the data set
```

```
#We can see that we do not have any high intercorrelations among the independent variables
#in our data set, so we are good to go. No multicollinearity issues.
```

Multiple Linear Regression Let's fit a full model for Spending:

Partition the Data Before we begin, we want to split our data into a training set and validation set. I will partition the 2000 records into non-overlapping training and validation set with a 60:40 ratio. We want to

train our multiple regression model on the training data and, then, assess its performance on the validation data.

```
set.seed(96) #does not matter what you set the seed to (unless you are checking for results
#from another machine and want to test and get the same data as me). I will be using different
#seeds to test multiple models to optimize the predictive accuracy.

train.index <- sample(c(1:dim(sub_taykoo)[1]), 0.6*dim(sub_taykoo)[1])
valid.index <- setdiff(c(1:dim(sub_taykoo)[1]), train.index)
sub_taykoo_train.df <- sub_taykoo[train.index, ]
sub_taykoo_valid.df <- sub_taykoo[valid.index, ]
dim(sub_taykoo_train.df) #checking number of observations and variables in training set
```

```
## [1] 1200    7
```

```
dim(sub_taykoo_valid.df) # ^ same thing for validation set
```

```
## [1] 800   7
```

Fitting a model Here, we are running the full model (vs. all six predictors) and will see which variables are statistically significant to predict Spending and evaluate the diagnostics on the model.

```
tayko_flm <- lm(Spending ~ ., data = sub_taykoo_train.df)
summary(tayko_flm)
```

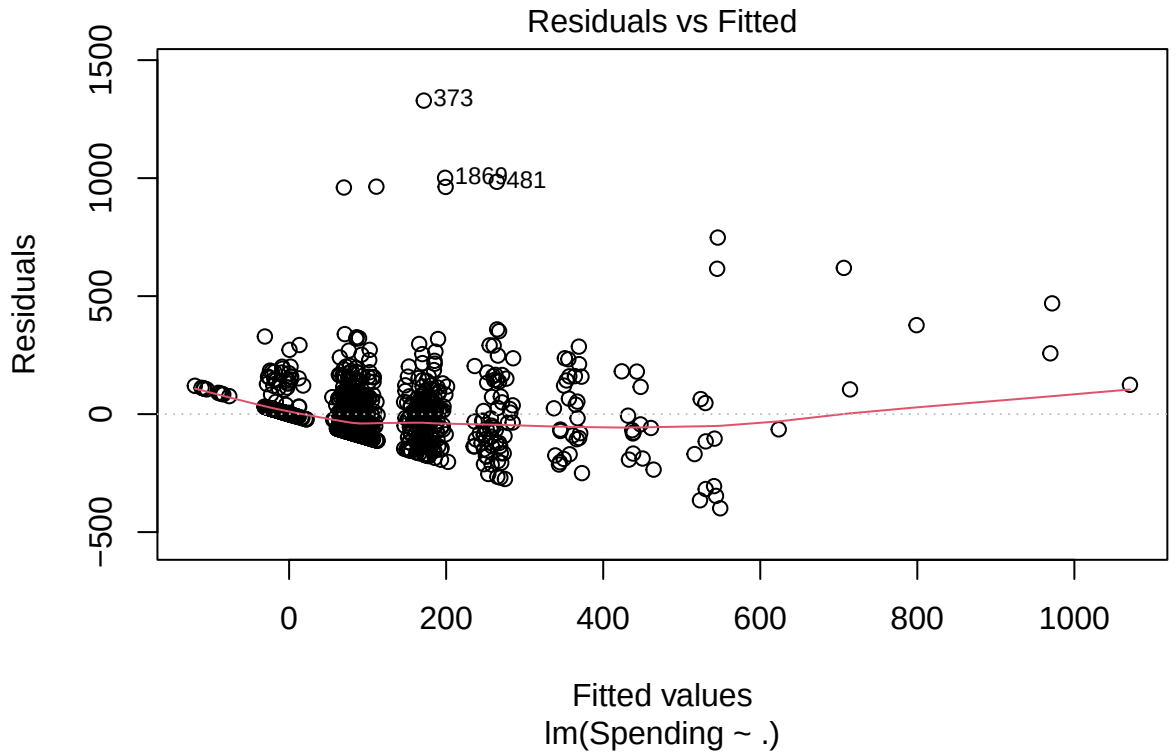
```
##
## Call:
## lm(formula = Spending ~ ., data = sub_taykoo_train.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -399.01  -76.42   -0.74   30.64 1328.46
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              5.741330  13.946148   0.412   0.6806
## Freq                    88.017118   2.966768  29.668   <2e-16 ***
## last_update_days_ago    -0.008619   0.003506  -2.458   0.0141 *
## Web.order               15.950869   7.361859   2.167   0.0305 *
## Gender.male              5.639355   7.273332   0.775   0.4383
## Address_is_res         -89.940506   9.178633  -9.799   <2e-16 ***
## US                      -2.640464   9.527497  -0.277   0.7817
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 125.3 on 1193 degrees of freedom
## Multiple R-squared:  0.4737, Adjusted R-squared:  0.4711
## F-statistic:   179 on 6 and 1193 DF,  p-value: < 2.2e-16
```

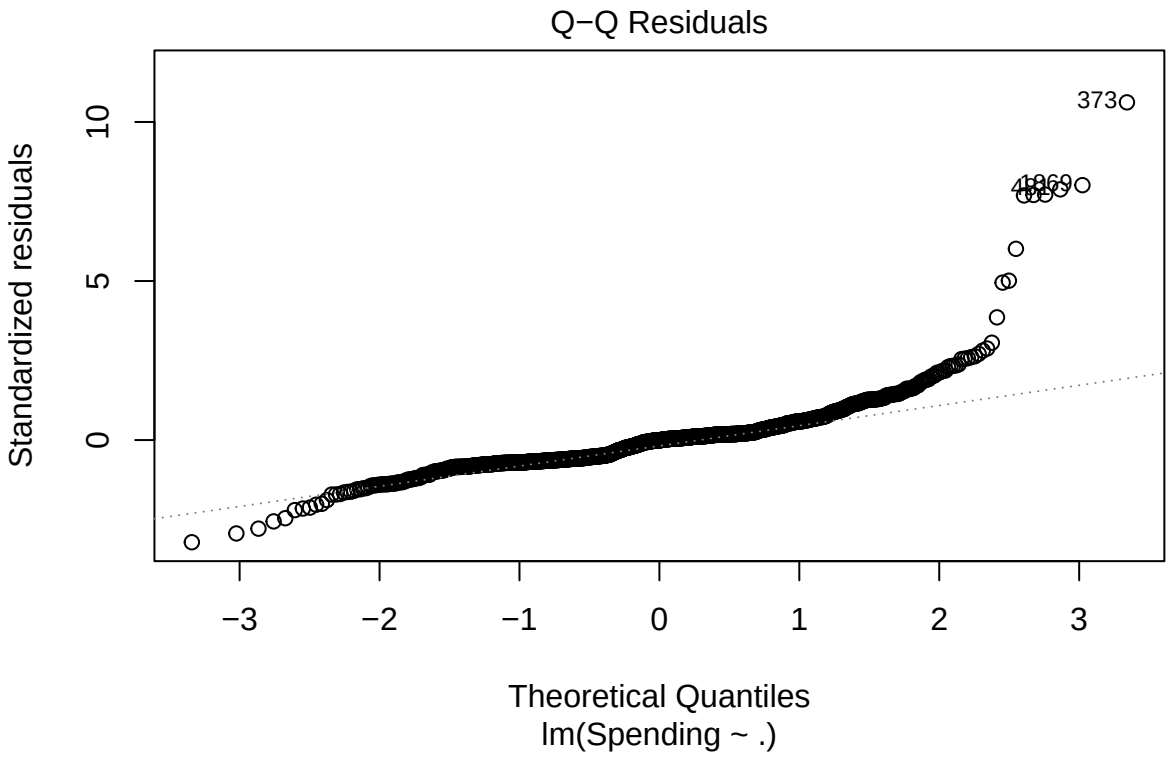
```
#After testing different seeds, it seems that our statistically significant predictors are Freq,
#last_update_days_ago, Web.order, and Address_is_res. Web.order should be closely monitored,
```

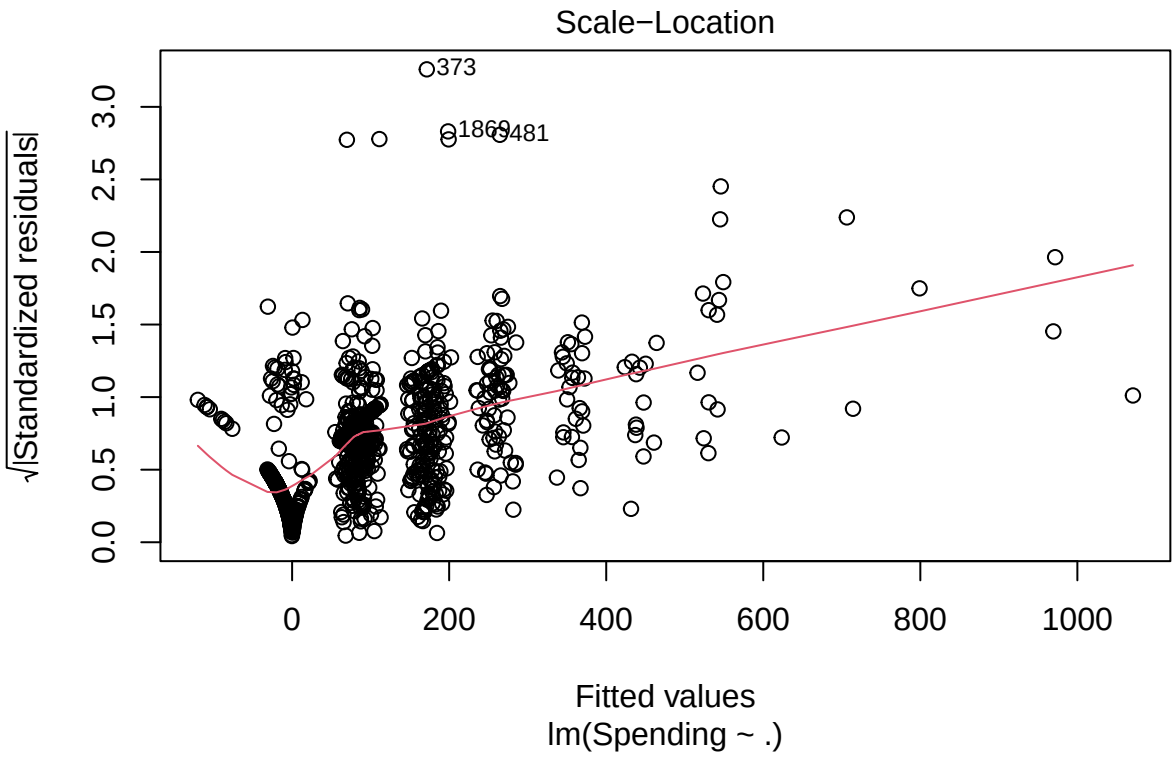
```
#because in other seeds, this predictor was insignificant (p-value > 0.05).

#Estimated Predictive Linear Equation:
#Spending(hat) = 5.74 + 88.02*Freq - 0.01*last_update + 15.95*Web.order + 5.64*Gender.male -
#89.94*Address_res - 2.64*US

#Keep in mind, the adjusted R^2 is 0.4711, which is close to the fit.freq where we only measured
#against one predictor. This model may be useful with respect to the trends in this data, but low
#in precision for accurately predicting profits for Tayko.

#Based on this model, a purchaser with a higher Freq (amount of transactions in the previous year),
#who also purchased by web order at least once, is most likely to spend a large amount of money.
#This idea is logical in the sense that a customer who is spending a larger amount of money is
#probably more active in dealing with Tayko, as they are more likely to be more familiar with
#(and fond of) the company's products.
```

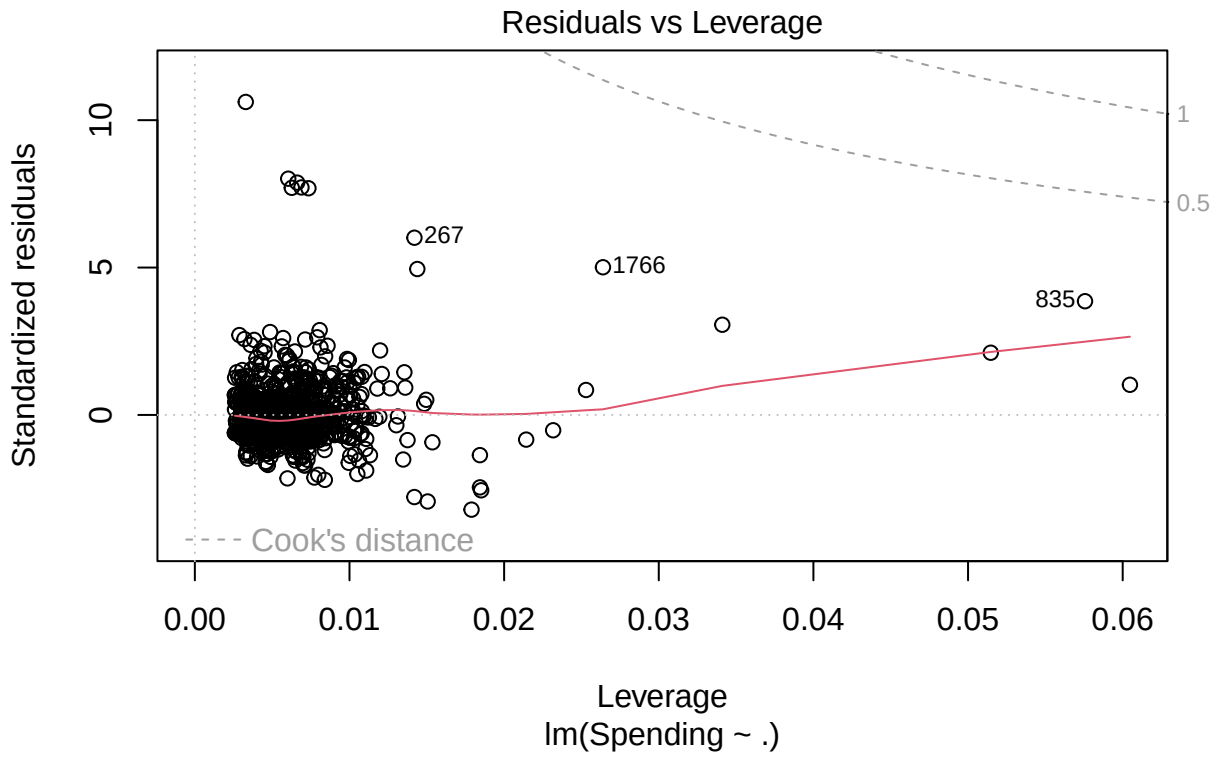
Evaluating Diagnostics & Validity Conditions

```
plot(tayko_flm)
```



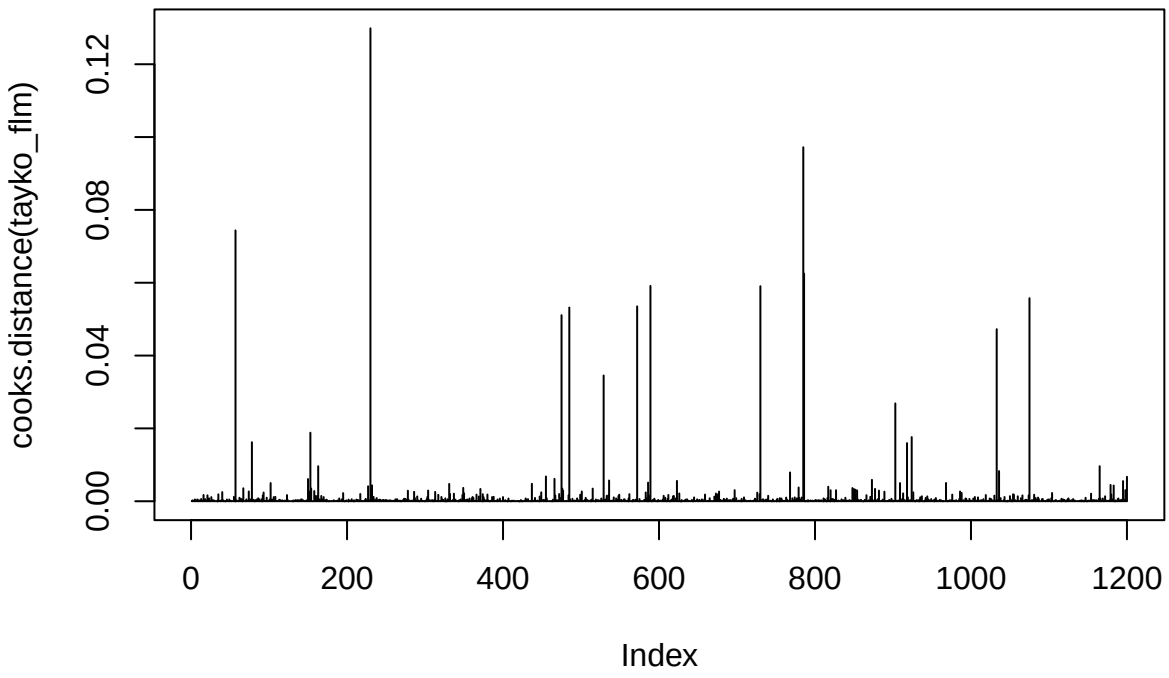

Q−Q Residuals
Standardized residuals
10
5
0
373
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(Spending ~ .)


Scale−Location
373
1869
481
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0
200
400
600
800
1000
Fitted values
lm(Spending ~ .)




```
#With the Residuals vs. Fitted plot, we can see that we have a slight parabolic pattern with a few
#evident outliers. The slight pattern could be a cause for concern, but not disruptive enough to presen
#us with any issues with the randomness in residuals. This occurrence is also due to many data entries
#in the predictors having the same value, especially in the binary predictors, so we are okay here.
#Our QQ-plot presents us with a minor problem because we have a break at the right end tail where those
#outliers seem to take over, but it's not enough to say our data isn't normal. We are okay here as well
#The scale-location plot is a warning for nonconstant variance/dependency but this plot is acceptable
#since majority of fitted values stay within the same range of standardized residuals.
#With the residuals vs. leverage plot, it is difficult to tell if we have any influential points.
#Let's try computing it.
```

```
plot(cooks.distance(tayko_flm), type = "h")
```




```
max(cooks.distance(tayko_flm))
```

```
## [1] 0.1298504
```

```
qf(.5,7,1193) #baseline for Cook's Distance
```

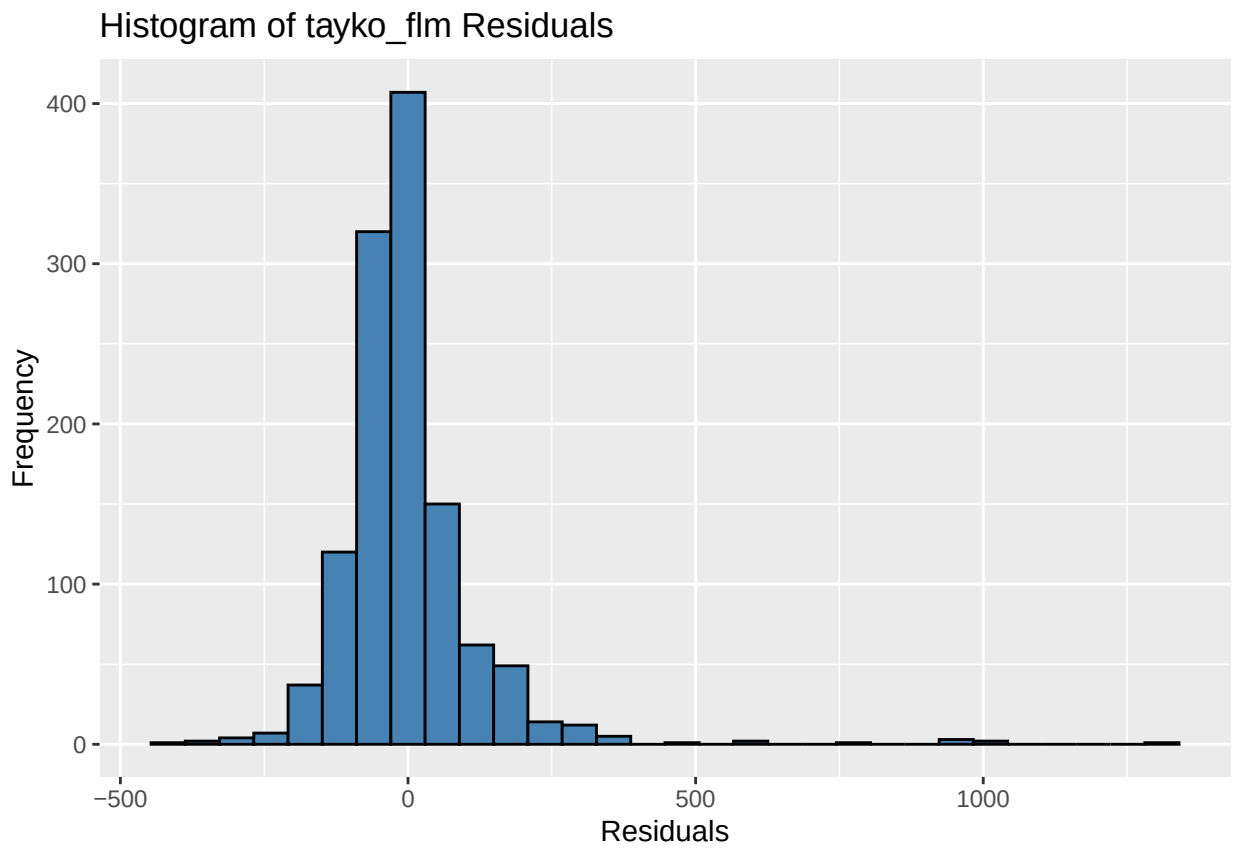
```
## [1] 0.907056
```

```
#With the baseline being 0.907 and our maximum Cook's Distance being 0.130, we do not have to
#worry about any points having too much influence.
```

Distribution of Residuals? Let's see what a histogram of the residuals look like now that we have updated our model.

```
residual.hist.train <- ggplot(data = sub_taykoo_train.df, aes(x = tayko_flm$residuals)) + geom_histogra
residual.hist.train
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
#The residuals follow a Normal distribution with a few outliers. Roughly satisfies
#normality assumptions and randomness is not lost. We can trust the predictions of Spending
#since they won't be biased.
```

Predictive Accuracy Here, I will show the predictive accuracy of the full model on the training set and the validation set.

```
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

##
## Attaching package: 'forecast'

## The following object is masked from 'package:ggpubr':
##
##     gghistogram
```

```
tayko_flm.train_prediction <- predict(tayko_flm, sub_taykoo_train.df)
accuracy(tayko_flm.train_prediction, sub_taykoo_train.df$Spending)
```

```
##                      ME     RMSE      MAE MPE MAPE
## Test set -5.397234e-13 124.9607 77.30365 NaN  Inf
```

```
tayko_flm.valid_prediction <- predict(tayko_flm, sub_taykoo_valid.df)
accuracy(tayko_flm.valid_prediction, sub_taykoo_valid.df$Spending)
```

```
##                 ME     RMSE      MAE MPE MAPE
## Test set 7.599623 138.7133 83.36178 NaN  Inf
```

```
#The two sets of predictive accuracy indicate that we have an issue with overfitting since the RMSE
#is lower for the training data than the validation data. 138.71 is a high value for RMSE,
#which indicates the wide scattering around the line of fit we obtained.
```

More predictions w/ validation data & Showing Residuals Here, we can get a more in-depth look at the precision of our model with the actual values of Spending vs. the predicted values of Spending and the residuals. We'll use the first 20 instances of the validation data for this case.

```
some.residuals <- sub_taykoo_valid.df$Spending[1:20] - tayko_flm.valid_prediction[1:20]
data.frame("Predicted" = tayko_flm.valid_prediction[1:20], "Actual" = sub_taykoo_valid.df$Spending[1:20]
```

```
##      Predicted Actual     Residual
## 2   -0.3048398      0    0.3048398
## 3  145.6664118    127  -18.6664118
## 4   89.6119509      0  -89.6119509
## 5   83.6278240      0  -83.6278240
## 6  -16.0179991      0   16.0179991
## 9  372.2344362    489  116.7655638
## 10  63.4069777    174  110.5930223
## 13  88.5569678      0  -88.5569678
## 14 430.8890744   1416  985.1109256
## 20 -18.1519523      0   18.1519523
## 23  -1.4945012      0    1.4945012
## 24  76.5845790    174   97.4154210
## 25 257.6009616    131 -126.6009616
## 26  85.8654085    189  103.1345915
## 29  62.6829576     90   27.3170424
## 31 -23.5327291      0   23.5327291
## 32 172.3253975    352  179.6746025
## 33  81.0990580      0  -81.0990580
## 34  -0.3479363      0    0.3479363
## 35 -95.8414902      0   95.8414902
```

Stepwise Regression Here, I will use stepwise regression to see if we can reduce the variables used in the model to ultimately improve it so by increasing robustness and increasing the predictive accuracy. Previously, I tested the "backwards elimination" method and the "both" (forwards and backwards) method, and they both work the same, so I will only demonstrate the backwards elimination method.

```
step(tayko_flm, direction = 'backward')
```

```
## Start:  AIC=11601.2
## Spending ~ Freq + last_update_days_ago + Web.order + Gender.male +
##     Address_is_res + US
##
##                       Df Sum of Sq      RSS   AIC
```

```
## - US                     1      1206 18739421 11599
## - Gender.male            1      9442 18747657 11600
## <none>                               18738214 11601
## - Web.order              1     73736 18811951 11604
## - last_update_days_ago   1     94920 18833134 11605
## - Address_is_res         1   1508143 20246357 11692
## - Freq                   1  13824671 32562885 12262
##
## Step:  AIC=11599.28
## Spending ~ Freq + last_update_days_ago + Web.order + Gender.male +
##     Address_is_res
##
##                         Df Sum of Sq      RSS   AIC
## - Gender.male            1      9228 18748649 11598
## <none>                               18739421 11599
## - Web.order              1     73439 18812859 11602
## - last_update_days_ago   1     95994 18835415 11603
## - Address_is_res         1   1515655 20255075 11691
## - Freq                   1  13825722 32565143 12260
##
## Step:  AIC=11597.87
## Spending ~ Freq + last_update_days_ago + Web.order + Address_is_res
##
##                         Df Sum of Sq      RSS   AIC
## <none>                               18748649 11598
## - Web.order              1     72569 18821218 11600
## - last_update_days_ago   1     97569 18846218 11602
## - Address_is_res         1   1532154 20280803 11690
## - Freq                   1  13824704 32573353 12259

##
## Call:
## lm(formula = Spending ~ Freq + last_update_days_ago + Web.order +
##     Address_is_res, data = sub_taykoo_train.df)
##
## Coefficients:
##          (Intercept)                  Freq  last_update_days_ago
##             7.108782             87.884865             -0.008728
##           Web.order        Address_is_res
##            15.819557            -90.427748
```

```
step.tayko_lm <- lm(Spending ~ Freq + last_update_days_ago + Web.order +
    Address_is_res, data = sub_taykoo_train.df)
summary(step.tayko_lm)
```

```
##
## Call:
## lm(formula = Spending ~ Freq + last_update_days_ago + Web.order +
##     Address_is_res, data = sub_taykoo_train.df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -395.81  -76.87    0.20   28.99 1330.53
```

```
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)              7.108782  11.134667   0.638   0.5233
## Freq                    87.884865   2.960651  29.684   <2e-16 ***
## last_update_days_ago    -0.008728   0.003500  -2.494   0.0128 *
## Web.order               15.819557   7.355608   2.151   0.0317 *
## Address_is_res         -90.427748   9.150640  -9.882   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 125.3 on 1195 degrees of freedom
## Multiple R-squared:  0.4734, Adjusted R-squared:  0.4716
## F-statistic: 268.6 on 4 and 1195 DF,  p-value: < 2.2e-16
```

```
#We can see that using stepwise regression eliminated the statistically insignificant predictors
#and increased adjusted R^2 by 0.0005, which is not much of a difference.

#Estimated Predictive Linear Equation:
#Spending(hat) = 7.11 + 87.88*Freq - 0.01*last_update + 15.82*Web.order - 90.43*Address_is_res
```

```
#Measuring predictive accuracy
step.tayko_lm.train_pred <- predict(step.tayko_lm, sub_taykoo_train.df)
accuracy(step.tayko_lm.train_pred, sub_taykoo_train.df$Spending)
```

```
##                       ME     RMSE      MAE MPE MAPE
## Test set -5.605532e-13 124.9955 77.26586 NaN  Inf
```

```
step.tayko_lm.valid_pred <- predict(step.tayko_lm, sub_taykoo_valid.df)
accuracy(step.tayko_lm.valid_pred, sub_taykoo_valid.df$Spending)
```

```
##                 ME     RMSE     MAE MPE MAPE
## Test set 7.597272 138.5999 83.2111 NaN  Inf
```

```
#Here, we can see that the RMSE for the validation set decreased with the new model created via
#stepwise regression in comparison to the original full model by 10%, which is worth noting.
#This new model shows an increased predictive accuracy and we can do so by looking at the same set
#of residuals from the predictions of the original model.
```

```
some.residuals2 <- sub_taykoo_valid.df$Spending[1:20] - step.tayko_lm.valid_pred[1:20]
data.frame("Predicted" = step.tayko_lm.valid_pred[1:20], "Actual" = sub_taykoo_valid.df$Spending[1:20],
```

```
##      Predicted Actual    Residual
## 2    -2.383429      0    2.383429
## 3   148.986928    127  -21.986928
## 4    87.757973      0  -87.757973
## 5    87.408845      0  -87.408845
## 6   -12.846851      0   12.846851
## 9   369.885497    489  119.114503
## 10   66.932497    174  107.067503
## 13   86.356799      0  -86.356799
```

```
## 14  428.369733   1416  987.630267
## 20  -20.123189      0   20.123189
## 23    1.860159      0   -1.860159
## 24   79.681414    174   94.318586
## 25  255.047963    131 -124.047963
## 26   89.341842    189   99.658158
## 29   66.199329     90   23.800671
## 31  -19.861343      0   19.861343
## 32  172.976076    352  179.023924
## 33   76.463687      0  -76.463687
## 34   -2.427070      0    2.427070
## 35 -100.818999      0  100.818999
```

```
#After comparing the numbers, we can see that most of the instances in the validation set have slightly
#better residuals, meaning we had more accurate predictions in the newer model. As mentioned previously
#it is worth noting that in fields of study such as predicting human behavior, and in this case, how
#much money people are spending on something, you can never expect to get highly accurate predictions.
#You will most likely get a wide scattering of residuals because people are just harder to predict than
#things like physical processes. You should expected to get a R^2 of less than 0.50, or 50%, in these
#case because of the level of unpredictability and the heightened amount of unexplainable variation.
#The fact that our models' adjusted R^2 were extremely close to 50% is telling. The unexplainable
#variation, in this case, could be due to multiple different factors that are not accounted for
#in our dataset, such as people's level of income, the extent to which they may budget, how much they
#value certain products, etc. All in all, it is safe to say that our model is unbiased, demonstrated
#an increased in predictive performance, and is ultimately a good linear model for Tayko to use to
#increase their profits based on the available data.
```

Comparing the Regression Results from the Two Models

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
stargazer(tayko_flm, step.tayko_lm, type="text", dep.var.labels = c("Amount that customers spend in tes
```

```
##
## Regression Results
## ================================================================================
##                                          Dependent variable:
##                      ---------------------------------------------------------
##                      Amount that customers spend in test mailing (in dollars)
##                                  (1)                          (2)
## --------------------------------------------------------------------------------
## Freq                            88.02***                     87.88***
##                                  (2.97)                       (2.96)
##
```

```
## last_update_days_ago          -0.01**                    -0.01**
##                               (0.004)                    (0.004)
##
## Web.order                     15.95**                    15.82**
##                                (7.36)                     (7.36)
##
## Gender.male                    5.64
##                               (7.27)
##
## Address_is_res               -89.94***                  -90.43***
##                                (9.18)                     (9.15)
##
## US                            -2.64
##                               (9.53)
##
## Constant                       5.74                       7.11
##                               (13.95)                    (11.13)
##
## ------------------------------------------------------------------------------
## Observations                  1,200                      1,200
## R2                             0.47                       0.47
## Adjusted R2                    0.47                       0.47
## Residual Std. Error       125.33 (df = 1193)         125.26 (df = 1195)
## F Statistic            178.96*** (df = 6; 1193)   268.58*** (df = 4; 1195)
## ==============================================================================
## Note:                                          *p<0.1; **p<0.05; ***p<0.01
```