Farhana Afroze
Ub Person: 50158114

Problem#1

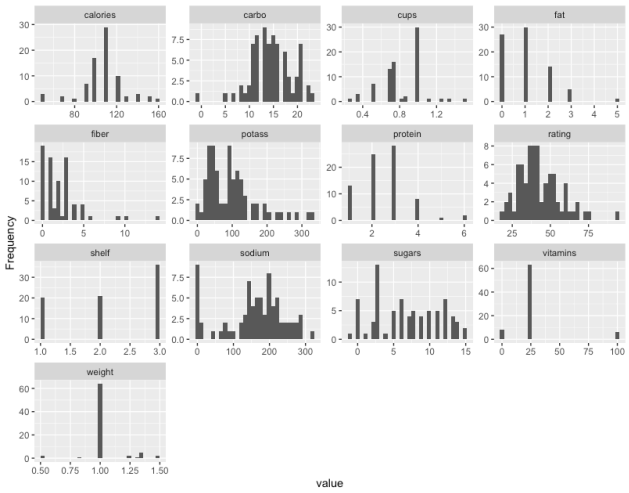
Exploring the cereal dataset:

First, we would like to explore the dataset. Cereal dataset has features called name, mfr, type, calories, protein, fat, sodium, fiber, carbo, sugars, potass, vitamins, shelf, weight, cups and rating. So this data set has total 77 rows and 16 columns. So, first I would like to see summary of my data.

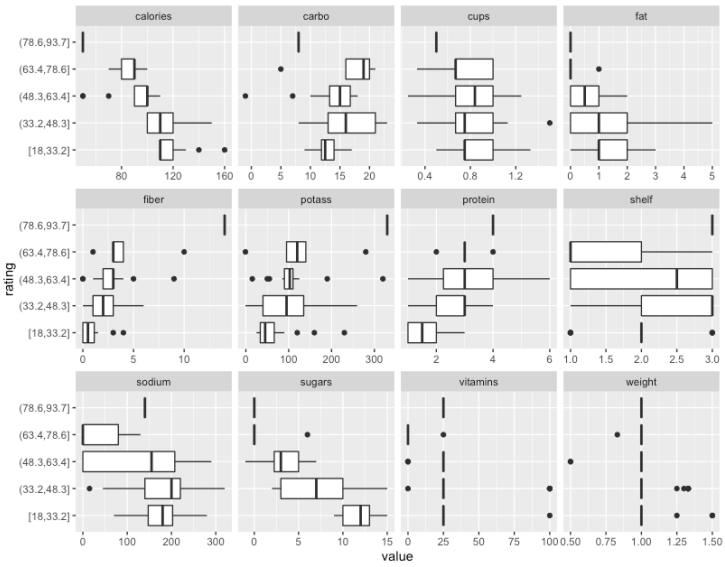
```
> summary(data)
     name               mfr                type              calories        protein
 Length:77          Length:77          Length:77          Min.   : 50.0   Min.   :1.000
 Class :character   Class :character   Class :character   1st Qu.:100.0   1st Qu.:2.000
 Mode  :character   Mode  :character   Mode  :character   Median :110.0   Median :3.000
                                                          Mean   :106.9   Mean   :2.545
                                                          3rd Qu.:110.0   3rd Qu.:3.000
                                                          Max.   :160.0   Max.   :6.000
      fat            sodium          fiber            carbo          sugars
 Min.   :0.000   Min.   :  0.0   Min.   : 0.000   Min.   :-1.0   Min.   :-1.000
 1st Qu.:0.000   1st Qu.:130.0   1st Qu.: 1.000   1st Qu.:12.0   1st Qu.: 3.000
 Median :1.000   Median :180.0   Median : 2.000   Median :14.0   Median : 7.000
 Mean   :1.013   Mean   :159.7   Mean   : 2.152   Mean   :14.6   Mean   : 6.922
 3rd Qu.:2.000   3rd Qu.:210.0   3rd Qu.: 3.000   3rd Qu.:17.0   3rd Qu.:11.000
 Max.   :5.000   Max.   :320.0   Max.   :14.000   Max.   :23.0   Max.   :15.000
     potass          vitamins          shelf           weight          cups
 Min.   : -1.00   Min.   :  0.00   Min.   :1.000   Min.   :0.50   Min.   :0.250
 1st Qu.: 40.00   1st Qu.: 25.00   1st Qu.:1.000   1st Qu.:1.00   1st Qu.:0.670
 Median : 90.00   Median : 25.00   Median :2.000   Median :1.00   Median :0.750
 Mean   : 96.08   Mean   : 28.25   Mean   :2.208   Mean   :1.03   Mean   :0.821
 3rd Qu.:120.00   3rd Qu.: 25.00   3rd Qu.:3.000   3rd Qu.:1.00   3rd Qu.:1.000
 Max.   :330.00   Max.   :100.00   Max.   :3.000   Max.   :1.50   Max.   :1.500
     rating
 Min.   :18.04
 1st Qu.:33.17
 Median :40.40
 Mean   :42.67
 3rd Qu.:50.83
 Max.   :93.70
>
```

Here you can information about each of the features. As we can see the name feature and mfr features doesn't have any numerical interpretation as those features are in character class. Next I would like to see if there is any mission value in dataset. We found 0 missing value. Next We would like to visualize our data set to get information from our dataset.

Plotting histogram:



As you can see from histogram rating features has most value between 30 to 50. You can see rating features is rightly skewed. Potass features is strongly right skewed and also you can see from carbo features it's left skewed data.
Next we would like to see boxplot to get more idea about our dataset:

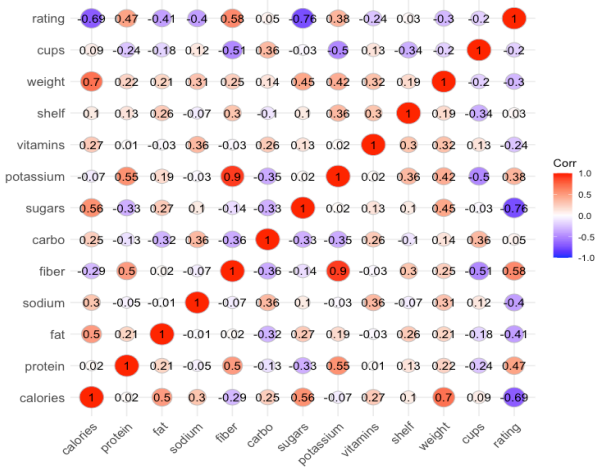


This displays distribution of data for rating variable. The rating is divided into 5 bins. We can see some correlations between calories and rating. For example, we can see calories more than 110 doesn't have that much rating. And there are two outliers after 140 and 160 calories. Those two have rating compared to others calorie group.

Data cleaning and visualization of Cereal dataset:
So first we would like to rename our 'potass' column into potassium. Next we would like to drop 'name', 'mfr' and 'type' features as they are in character class. Now the dataset looks like this:

```
> head(data)
  calories protein fat sodium fiber carbo sugars potassium vitamins shelf weight cups
1       70       4   1    130  10.0   5.0      6       280       25     3      1 0.33
2      120       3   5     15   2.0   8.0      8       135        0     3      1 1.00
3       70       4   1    260   9.0   7.0      5       320       25     3      1 0.33
4       50       4   0    140  14.0   8.0      0       330       25     3      1 0.50
5      110       2   2    200   1.0  14.0      8        -1       25     3      1 0.75
6      110       2   2    180   1.5  10.5     10        70       25     1      1 0.75
    rating
1 68.40297
2 33.98368
3 59.42551
4 93.70491
5 34.38484
6 29.50954
```

So now we have 13 rows. Next we would like to see correlations of our modified dataset.



Here we see most strong significant correlations is between potassium and fiber. Next we would like to see correlations of rating variable. Here, rating variable is mostly correlated with fiber. And we see rating variable having most significant inverse relationship with sugars.

Transforming the cereal dataset:

We would like to normalize our dataset between 0 and 1 so we can have common scales of all the numeric columns values. So, suppose if we have different ranges of the column values, for example large values, won't influence on the dataset. After normalizing the dataset looks like this:

```
> head(data)
   calories protein fat   sodium fiber     carbo sugars potassium vitamins shelf weight
1 0.1818182     0.6 0.2 0.406250  10.0 0.2500000 0.4375 0.8489426     0.25     1    0.5
2 0.6363636     0.4 1.0 0.046875   2.0 0.3750000 0.5625 0.4108761     0.00     1    0.5
3 0.1818182     0.6 0.2 0.812500   9.0 0.3333333 0.3750 0.9697885     0.25     1    0.5
4 0.0000000     0.6 0.0 0.437500  14.0 0.3750000 0.0625 1.0000000     0.25     1    0.5
5 0.5454545     0.2 0.4 0.625000   1.0 0.6250000 0.5625 0.0000000     0.25     1    0.5
6 0.5454545     0.2 0.4 0.562500   1.5 0.4791667 0.6875 0.2145015     0.25     0    0.5
   cups    rating
1 0.064 0.6655928
2 0.600 0.2106846
3 0.064 0.5469406
4 0.200 1.0000000
5 0.400 0.2159866
6 0.400 0.1515514
>
```

1(a). At first, we would like to split our dataset into training and testing. We will keep 80% for our training data set and 20% for our testing set. And our dependent variable would be 'rating'. After dividing the data training set has 13 columns with 61 rows and testing set has 13 columns with 16 rows. Now, we will fit the linear model for training set. After fitting the summary looks like this:

```
Call:
lm(formula = rating ~ ., data = training_data)

Residuals:
      Min        1Q    Median        3Q       Max
-0.177829 -0.029407 -0.003459  0.035561  0.127912

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  2.760211   0.102436  26.946  < 2e-16 ***
calories    -1.171534   0.215351  -5.440 1.77e-06 ***
protein      0.187030   0.039645   4.718 2.09e-05 ***
fat         -0.066896   0.033844  -1.977 0.053851 .
sodium      -0.164368   0.046889  -3.506 0.000999 ***
fiber        0.060850   0.022104   2.753 0.008314 **
carbo        0.435446   0.084813   5.134 5.10e-06 ***
sugars      -0.462719   0.082491  -5.609 9.85e-07 ***
potassium    0.007595   0.075879   0.100 0.920686
vitamins     0.059322   0.066173   0.896 0.374474
shelf       -0.032858   0.026640  -1.233 0.223429
weight       0.951287   0.230423   4.128 0.000145 ***
cups        -0.141334   0.068871  -2.052 0.045629 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.06319 on 48 degrees of freedom
Multiple R-squared:  0.9249,    Adjusted R-squared:  0.9062
F-statistic: 49.29 on 12 and 48 DF,  p-value: < 2.2e-16
```

So, here we see for calories increase, nutritional rating decreased by 1.17. For protein increase, nutritional rating increase by 0.18. For fat increase, nutritional rating decrease by 0.066. For sugar increase, rating decrease by 0.46. Here we see very high p values for potassium, vitamins and shelf. Which means there is almost no relationship with dependent variable rating. So, we can't reject the null hypothesis. Where we see sugar has very strong negative relationship with rating. So, in this case we can reject null hypothesis. The residual standard error is – 0.06319. Then we predicted our model using training data. After predicting we calculated MSE(Mean Sqared Error) and our MSE error for training is- 0.19 Which is pretty good.

Next we fit the linear model for testing set. And the summary looks like this:

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.401041   0.354416   9.596  0.00240 **
calories    -0.038695   0.122845  -0.315  0.77341
protein     -0.031079   0.040980  -0.758  0.50335
fat         -0.239347   0.040044  -5.977  0.00937 **
sodium       0.014052   0.048908   0.287  0.79258
fiber        0.091017   0.024875   3.659  0.03527 *
carbo        0.865961   0.361765   2.394  0.09642 .
sugars      -0.944656   0.155406  -6.079  0.00894 **
potassium    0.360125   0.098789   3.645  0.03561 *
vitamins    -0.110676   0.060347  -1.834  0.16401
shelf        0.006442   0.016297   0.395  0.71905
weight      -0.954372   0.311573  -3.063  0.05486 .
cups        -0.211589   0.097817  -2.163  0.11922
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.02097 on 3 degrees of freedom
Multiple R-squared:  0.9979,    Adjusted R-squared:  0.9897
F-statistic: 121.2 on 12 and 3 DF,  p-value: 0.001088
```

Here we see residual standard error is- 0.02097 and we see little large p values for calories, sodium and shelf which is different than the fitting of training set. Next we predicted values and would like to see how much difference is from actual values.

```
> difference = data.frame(pred = predi_test, actual = test_set$rating)
> difference
       pred   actual
4  3.110778 3.111291
6  2.325868 2.330723
7  2.385405 2.399937
18 2.448106 2.445787
21 2.837172 2.834307
29 2.534061 2.530679
35 2.598059 2.601622
37 2.384126 2.360984
41 2.506323 2.502853
46 2.416741 2.417214
52 2.344809 2.349091
61 2.730857 2.727384
70 2.492617 2.496428
72 2.624296 2.613566
76 2.661704 2.680069
77 2.453687 2.452674
```

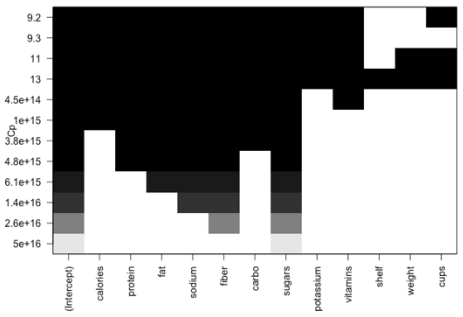
As we can see predicted value is just slightly different than actual values which is good because our error will be less. Next we calculated MSE which is 0.0013 which represents the linear regression model on the test set predicted with very high accuracy.

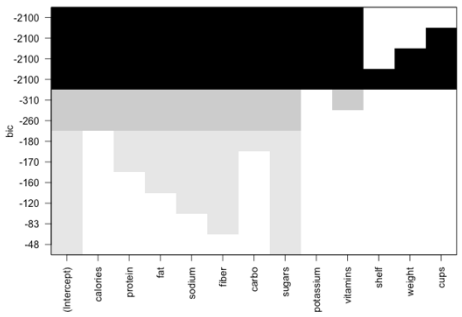
1(b). Here we will perform Forward subset selection which gives good performing model and lowers prediction error. At first, we will on training data and next we will perform on testing data.

Training data forward subset selection: We are using regsubsets package for selection. Here we select nvmax = 13 as we have 13 predictors. After train the model it returns multiple model with different size upto nvmax. Here is the summary:

```
Selection Algorithm: forward
          calories protein fat sodium fiber carbo sugars potassium vitamins shelf weight cups
1  ( 1 )  " "      " "     " " " "    " "   " "   "*"    " "       " "      " "   " "    " "
2  ( 1 )  " "      " "     " " " "    "*"   " "   "*"    " "       " "      " "   " "    " "
3  ( 1 )  " "      " "     " " "*"    "*"   " "   "*"    " "       " "      " "   " "    " "
4  ( 1 )  " "      " "     "*" "*"    "*"   " "   "*"    " "       " "      " "   " "    " "
5  ( 1 )  " "      "*"     "*" "*"    "*"   " "   "*"    " "       " "      " "   " "    " "
6  ( 1 )  " "      "*"     "*" "*"    "*"   "*"   "*"    " "       " "      " "   " "    " "
7  ( 1 )  "*"      "*"     "*" "*"    "*"   "*"   "*"    " "       " "      " "   " "    " "
8  ( 1 )  "*"      "*"     "*" "*"    "*"   "*"   "*"    " "       "*"      " "   " "    " "
9  ( 1 )  "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      " "   " "    " "
10  ( 1 ) "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      " "   " "    "*"
11  ( 1 ) "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      " "   "*"    "*"
12  ( 1 ) "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      "*"   "*"    "*"
>
```

Here we see the best 1 variable model contains only sugars. Next 2 best variable model contains sugars and fiber. Next 3 best variable model would be sugars, fiber and sodium and models go on. Now we would like to see some metrics such as ‘cp’ and ‘bic’.



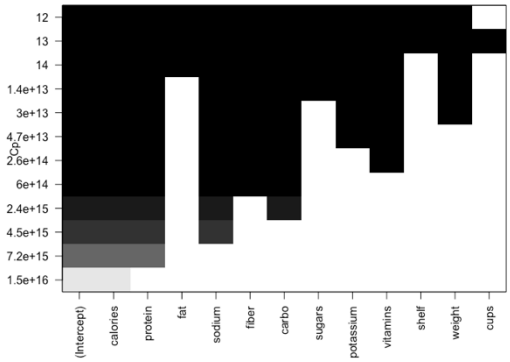


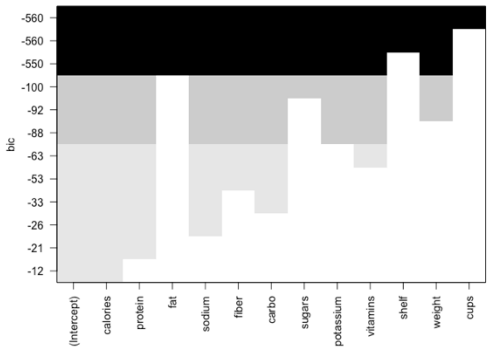
What cp and bic do is when you add additional variable it penalizes. Here you can see in cp the variable which is best has high cp value and in bic variable which is the best high bic value. And in both case, it’s sugar variable in case of 1 variable model.

Test data Forward subset selection: Now we are doing forward subset selection on test data. Here is the summary:

```
Selection Algorithm: forward
          calories protein fat sodium fiber carbo sugars potassium vitamins shelf weight cups
1  ( 1 )  "*"      " "     " " " "    " "   " "   " "    " "       " "      " "   " "    " "
2  ( 1 )  "*"      "*"     " " " "    " "   " "   " "    " "       " "      " "   " "    " "
3  ( 1 )  "*"      "*"     " " "*"    " "   " "   " "    " "       " "      " "   " "    " "
4  ( 1 )  "*"      "*"     " " "*"    " "   "*"   " "    " "       " "      " "   " "    " "
5  ( 1 )  "*"      "*"     " " "*"    "*"   "*"   " "    " "       " "      " "   " "    " "
6  ( 1 )  "*"      "*"     " " "*"    "*"   "*"   " "    " "       "*"      " "   " "    " "
7  ( 1 )  "*"      "*"     " " "*"    "*"   "*"   " "    "*"       "*"      " "   " "    " "
8  ( 1 )  "*"      "*"     " " "*"    "*"   "*"   " "    "*"       "*"      " "   "*"    " "
9  ( 1 )  "*"      "*"     " " "*"    "*"   "*"   "*"    "*"       "*"      " "   "*"    " "
10  ( 1 ) "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      " "   "*"    " "
11  ( 1 ) "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      "*"   "*"    " "
12  ( 1 ) "*"      "*"     "*" "*"    "*"   "*"   "*"    "*"       "*"      "*"   "*"    "*"
>
```

Here as we see best 1 variable model is calories. Best two variable model is calories and protein. Next we would like to see ‘cp’ and ‘bic’ value.





Here we can see in both cases calories has high cp and high bic values.
We found the minimum bic is 12 and cp is 11. And maximum bic and cp is 1. Here are the coefficients for 1, 11 and 12. Here is the plot:

```
>
>
>
> coef(forward_subset_test, 1)
(Intercept)    calories
  0.8448997  -0.9449406
> coef(forward_subset_test, 11)
  (Intercept)      calories       protein           fat        sodium         fiber         carbo
 3.791561e-01 -3.238037e-01  2.163022e-01 -1.117738e-01 -2.304678e-01  4.551132e-02  3.465254e-01
      sugars     potassium      vitamins         shelf        weight
-1.532911e-01 -1.487112e-01 -6.768512e-02 -1.228313e-08  5.764132e-09
> coef(forward_subset_test, 12)
  (Intercept)      calories       protein           fat        sodium         fiber         carbo
 3.791561e-01 -3.238037e-01  2.163022e-01 -1.117738e-01 -2.304678e-01  4.551132e-02  3.465254e-01
      sugars     potassium      vitamins         shelf        weight          cups
-1.532911e-01 -1.487113e-01 -6.768513e-02 -9.895464e-09 -9.747697e-09 -9.912193e-09
>
```

1(c). Here we will perform exhaustive subset selection. We Performed on the training data first then we performed on test data. Here is information on training set:

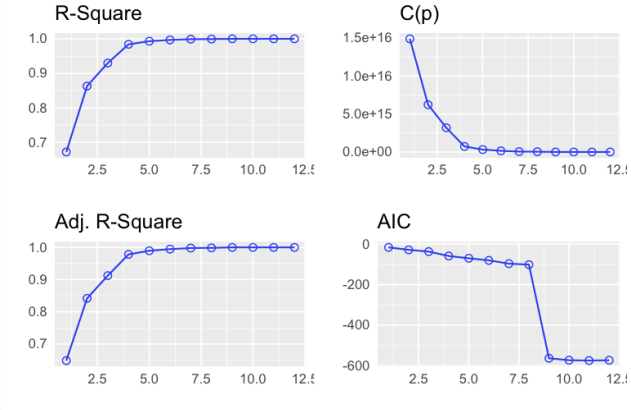
Subsets Regression Summary

Model	R-Square	Adj. R-Square	Pred R-Square	C(p)	AIC	SBIC	SBC	MSEP	FPE	HSP	APC
1	0.6019	0.5951	0.5736	5.039573e+16	-88.5526	-267.6631	-82.2200	0.7837	0.0133	2e-04	0.4251
2	0.7909	0.7837	0.7657	2.646569e+16	-125.8401	-306.9506	-117.3966	0.4188	0.0072	1e-04	0.2307
3	0.8895	0.8837	0.8694	1.398825e+16	-162.7356	-345.8461	-152.1812	0.2253	0.0039	1e-04	0.1260
4	0.9517	0.9483	0.9393	6.108798e+15	-211.2733	-396.3838	-198.6081	0.1002	0.0018	0.0000	0.0569
5	0.9696	0.9669	0.9525	3.843921e+15	-237.5308	-424.6413	-222.7547	0.0642	0.0012	0.0000	0.0370
6	0.9847	0.9830	0.9732	1.941257e+15	-277.2034	-466.3139	-260.3164	0.0330	6e-04	0.0000	0.0193
7	0.9918	0.9907	0.9887	1.034366e+15	-313.6058	-504.7163	-294.6079	0.0179	3e-04	0.0000	0.0106
8	0.9965	0.9959	0.9949	4.469543e+14	-362.7901	-555.9006	-341.6814	0.0079	1e-04	0.0000	0.0048
9	1.0000	1.0000	1	9.3149	-2179.5158	-2348.4692	-2156.2962	0.0000	0.0000	0.0000	0.0000
10	1.0000	1.0000	1	10.9704	-2177.9350	-2346.2870	-2152.6045	0.0000	0.0000	0.0000	0.0000
11	1.0000	1.0000	1	12.9079	-2176.0113	-2343.8866	-2148.5700	0.0000	0.0000	0.0000	0.0000
12	1.0000	1.0000	1	13.0000	-2176.3890	-2342.6045	-2146.8367	0.0000	0.0000	0.0000	0.0000

As we can see MSEP is 0 for model with 12, 11,10 and 9 variable model. Next we performed exhaustive subset selection on the test data.

Model	R-Square	R-Square	R-Square	C(p)	AIC	SBIC	SBC	MSEP
FPE	HSP	APC						
1	0.6722	0.6487	0.3818	1.489041e+16	-16.3367	-67.7428	-14.0190	0.2654
0.0186	0.0013	0.4215						
2	0.8630	0.8419	0.7653	6.223998e+15	-28.2936	-81.6996	-25.2033	0.1202
0.0089	6e-04	0.2003						
3	0.9298	0.9123	0.8554	3.186711e+15	-37.0044	-92.4104	-33.1414	0.0671
0.0052	4e-04	0.1169						
4	0.9841	0.9783	0.9641	7.236907e+14	-58.7225	-116.1285	-54.0869	0.0168
0.0013	1e-04	0.0304						
5	0.9931	0.9896	0.979	3.145848e+14	-70.0522	-129.4583	-64.6441	0.0081
7e-04	1e-04	0.0152						
6	0.9968	0.9947	0.9836	1.456553e+14	-80.3724	-141.7784	-74.1917	0.0042
4e-04	0.0000	0.0082						
7	0.9990	0.9980	0.9939	4.724538e+13	-96.3866	-159.7926	-89.4333	0.0016
1e-04	0.0000	0.0031						
8	0.9993	0.9986	0.9937	3.034512e+13	-101.4701	-166.8762	-93.7442	0.0012
1e-04	0.0000	0.0024						
9	1.0000	1.0000	1	12.0517	-562.7945	-590.4064	-554.2960	0.0000
0.0000	0.0000	0.0000						
10	1.0000	1.0000	1	9.9004	-572.3912	-568.7970	-563.1202	0.0000
0.0000	0.0000	0.0000						
11	1.0000	1.0000	1	11.1190	-573.9683	-554.3696	-563.9246	0.0000
0.0000	0.0000	0.0000						

As we see when there is less variable model the MSEP is high and r square is low as there is low variance. More variable model has low MSEP and high variance.



Here we see R-sqare get higher after 5 variable model and then it gets steady. Next we see for cp get low after 5 variable model and get steady. In the case AIC it's get low after 9.

1(d).

Linear Regression Model: We found MSE for linear regression model = 0.0013 and RMSE = 0.036 with all 13 variables. Our linear model in test set gave very good accuracy.

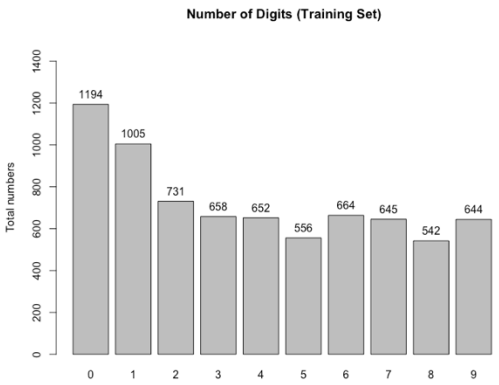
Forward subset Selection Model: In forward subset selection the best model is with 1 variable model and we found the intercept is 0.8449 and we found when calories decrease rating increase by 0.94. So, as we see end result for forward subset selection model is single regression model so for multiple regression model is not good fit.

Exhaustive subset selection model: In exhaustive subset selection we found RMSE is 0 for 9, 10, 12, 11 model. Here with 8 variable model it gives 0.079 RMSE. So, we see best subset gives accuracy higher by choosing important variable models while ignoring non- significant variables.

I would say exhaustive subset selection model is the best. Because it gives high accuracy by selecting important variables models while ignoring non-significant variables. Where Forward subset is good for single regression model.

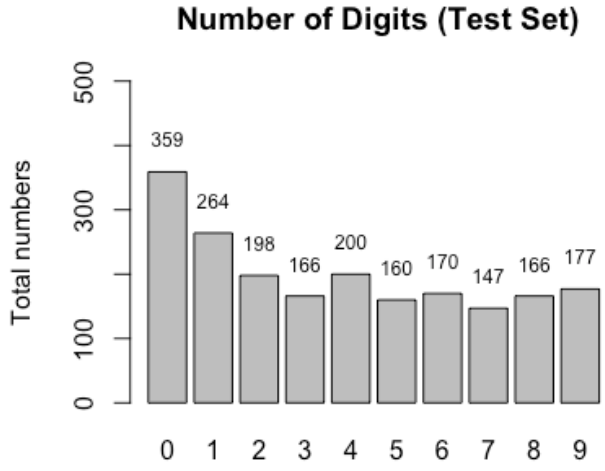
Problem#2

Dataset Exploration, Transformation and Visualization: This zipcode dataset is already splitted into train and test. So, in the train data set has 7219 rows with 257 columns. There are no missing values in training set. Next I would like to transform V1 which is digits into factor. Next we would like to see frequency of the digits from 0-9. Here is the visualization:



As we can there are 731 on digits 2 and 658 on digits 3. Next I would select only 2 and 3 digits data as this assignments requires.

Next we will explore our test data set which has 2007 rows with 257 columns. There are no missing values in the test data. Next, we transformed digit column V1 into factor. Next we would like to see frequency of the digits from 0-9. Here is the visualization:



Here we see there are 198 two and 166 three in the zipcode test data set. Next again we selected 2 and 3 digits for our model. Next we will implement K nearest neighbors and Linear regression for classification.

K Nearest Neighbors: In K nearest Neighbors we predict the category of the test points from available class label by finding distance between the test points and k features values. Choosing right K has effect on model performance. In our assignment chose k = 1,3,5,7,9,11,3,15 and we ran a for loop to get the train and test set error. Here is the train error:

```
>
> list_train_error
[[1]]
[1] 0

[[2]]
[1] 0.005039597

[[3]]
[1] 0.005759539

[[4]]
[1] 0.006479482

[[5]]
[1] 0.009359251

[[6]]
[1] 0.008639309

[[7]]
[1] 0.008639309

[[8]]
[1] 0.009359251
```

```
> list_test_error
[[1]]
[1] 0.02472527

[[2]]
[1] 0.03021978

[[3]]
[1] 0.03021978

[[4]]
[1] 0.03296703

[[5]]
[1] 0.03571429

[[6]]
[1] 0.03571429

[[7]]
[1] 0.03846154

[[8]]
[1] 0.03846154
```

As we when k= 1 it's 0 error but when k value gets increase the error gets slightly increase too. IN test data set it's same case when k values gets increase the error gets slightly increase every time too. So this means when k has low value it's predicting correctly and classifying handwritten digits correctly on the label digit 2 and 3.

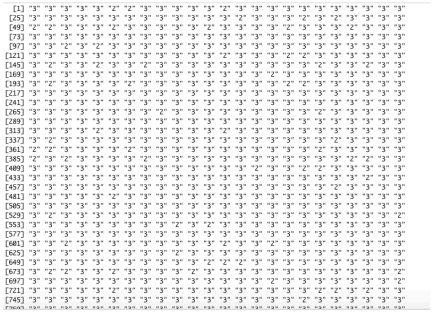
Next we would like to see accuracy rate for train data and test data.

```
> list_train_accuracy
[[1]]
[1] 100

[[2]]
[1] 99.49604

[[3]]
[1] 99.42405

[[4]]
[1] 99.35205

[[5]]
[1] 99.06407

[[6]]
[1] 99.13607

[[7]]
[1] 99.13607

[[8]]
[1] 99.06407
```

```
> list_test_accuracy
[[1]]
[1] 97.52747

[[2]]
[1] 96.97802

[[3]]
[1] 96.97802

[[4]]
[1] 96.7033

[[5]]
[1] 96.42857

[[6]]
[1] 96.42857

[[7]]
[1] 96.15385

[[8]]
[1] 96.15385
```

Here we see training accuracy and testing accuracy for multiple k values. In test data set when k value is 1 the accuracy is 97.5 which is really good. So, it means at k=1 it capturing all the test data correctly and putting in the 2 and 3 digit labels with accuracy 97.5.

In linear regression we tried to classify data. The classifying predicting test data set looks like this:



Problem#3:

3(a). In the college data set we have 777 rows with 18 features. Here we will “apps” as our response variable. We first look at data. There is no empty values. Summary of the data looks like this:

```
> summary(College)
 Private        Apps           Accept          Enroll       Top10perc       Top25perc
 No :212   Min.   :   81   Min.   :   72   Min.   :  35   Min.   : 1.00   Min.   :  9.0
 Yes:565   1st Qu.:  776   1st Qu.:  604   1st Qu.: 242   1st Qu.:15.00   1st Qu.: 41.0
           Median : 1558   Median : 1110   Median : 434   Median :23.00   Median : 54.0
           Mean   : 3002   Mean   : 2019   Mean   : 780   Mean   :27.56   Mean   : 55.8
           3rd Qu.: 3624   3rd Qu.: 2424   3rd Qu.: 902   3rd Qu.:35.00   3rd Qu.: 69.0
           Max.   :48094   Max.   :26330   Max.   :6392   Max.   :96.00   Max.   :100.0
  F.Undergrad     P.Undergrad         Outstate       Room.Board       Books           Personal
 Min.   :  139   Min.   :    1.0   Min.   : 2340   Min.   :1780   Min.   :  96.0   Min.   : 250
 1st Qu.:  992   1st Qu.:   95.0   1st Qu.: 7320   1st Qu.:3597   1st Qu.: 470.0   1st Qu.: 850
 Median : 1707   Median :  353.0   Median : 9990   Median :4200   Median : 500.0   Median :1200
 Mean   : 3700   Mean   :  855.3   Mean   :10441   Mean   :4358   Mean   : 549.4   Mean   :1341
 3rd Qu.: 4005   3rd Qu.:  967.0   3rd Qu.:12925   3rd Qu.:5050   3rd Qu.: 600.0   3rd Qu.:1700
 Max.   :31643   Max.   :21836.0   Max.   :21700   Max.   :8124   Max.   :2340.0   Max.   :6800
      PhD            Terminal       S.F.Ratio      perc.alumni        Expend         Grad.Rate
 Min.   :  8.00   Min.   : 24.0   Min.   : 2.50   Min.   : 0.00   Min.   : 3186   Min.   : 10.00
 1st Qu.: 62.00   1st Qu.: 71.0   1st Qu.:11.50   1st Qu.:13.00   1st Qu.: 6751   1st Qu.: 53.00
 Median : 75.00   Median : 82.0   Median :13.60   Median :21.00   Median : 8377   Median : 65.00
 Mean   : 72.66   Mean   : 79.7   Mean   :14.09   Mean   :22.74   Mean   : 9660   Mean   : 65.46
 3rd Qu.: 85.00   3rd Qu.: 92.0   3rd Qu.:16.50   3rd Qu.:31.00   3rd Qu.:10830   3rd Qu.: 78.00
 Max.   :103.00   Max.   :100.0   Max.   :39.80   Max.   :64.00   Max.   :56233   Max.   :118.00
> 
```

Here we see in “apps” which is our response variable mean is 3002 and median is 1558. We would like to normalize our data set as we see there are some larger value which will have more influence on the prediction. Next we did split our data set into .80 into training and 0.20 into testing. Now training data had 621 with 19 columns and testing has 159 with 19 columns. Next We fitted linear model in the training set and obtained the test error. The test error we got is 0.14.

3(b).