Regression

Aleezah Athar

Linear regression works by using the x values, called predictors, to find y values, called target values. We aim to find the relationship between x and y. In linear regression, we define the formula using w and b, where w is the slope of the line and b is the intercept. It is important to note that w quantifies the amount of change in y for every unit change in x. Furthermore, linear regression is used when our target variable is quantitative. The strengths of linear regression are that its a relatively simple algorithm, works well when data follow a linear pattern, and has low variance. However, the weakness is that it has a high bias because it tends to assume and look for a linear relationship in the data. Linear models in general have advantages that are that they are easy to interpret, computationally efficient, can handle missing data points, and can be extended to non-linear relationships if we use complex transformation functions. However, the disadvantages are that they assume a linear relationship where there might not be one and tend to overfit.

This is a data set showing the energy use of appliances (in Wh) and the energy use of light fixtures in a hour(in Wh) and the corresponding temperature, humidity and weather conditions at the time of recording. This dataset is from the UCI Machine Learning Repository.

Data Exploration

First we set the seed to 3 to get the same results each time Then we read in the csv file which contains out data

```
set.seed(3)
df<-read.csv("energydata_complete.csv")
```

Divide into 80/20 train/test Then we randomly select 80% of the rows to be the training data and 20% to be the testing data.

```
set.seed(3)
i<-sample(1:nrow(df), nrow(df)*0.8, replace=FALSE)
train<-df[i,]
test<-df[-i,]
```

Use at least 5 R functions for data exploration, using the training data. We can use the names, dimention, summary, structure and head functions to get information about the data. We can also use the colSums and is.na functions together to check for any NA values.

```
names(train)
```

```
##  [1] "date"        "Appliances"  "lights"      "T1"          "RH_1"
##  [6] "T2"          "RH_2"        "T3"          "RH_3"        "T4"
## [11] "RH_4"        "T5"          "RH_5"        "T6"          "RH_6"
## [16] "T7"          "RH_7"        "T8"          "RH_8"        "T9"
## [21] "RH_9"        "T_out"       "Press_mm_hg" "RH_out"      "Windspeed"
## [26] "Visibility"  "Tdewpoint"   "rv1"         "rv2"
```

```
dim(train)
```

```
## [1] 15788    29
```

```
summary(train)
```

```
##      date              Appliances          lights             T1
##  Length:15788       Min.   :  10.00   Min.   : 0.000   Min.   :16.79
##  Class :character   1st Qu.:  50.00   1st Qu.: 0.000   1st Qu.:20.79
##  Mode  :character   Median :  60.00   Median : 0.000   Median :21.60
##                     Mean   :  97.45   Mean   : 3.847   Mean   :21.68
##                     3rd Qu.: 100.00   3rd Qu.: 0.000   3rd Qu.:22.60
##                     Max.   :1080.00   Max.   :50.000   Max.   :26.26
##       RH_1             T2             RH_2             T3
##  Min.   :27.02   Min.   :16.10   Min.   :20.46   Min.   :17.20
##  1st Qu.:37.40   1st Qu.:18.79   1st Qu.:37.93   1st Qu.:20.79
##  Median :39.66   Median :20.00   Median :40.50   Median :22.10
##  Mean   :40.28   Mean   :20.33   Mean   :40.45   Mean   :22.26
##  3rd Qu.:43.06   3rd Qu.:21.50   3rd Qu.:43.29   3rd Qu.:23.29
##  Max.   :63.36   Max.   :29.86   Max.   :56.03   Max.   :29.24
##       RH_3             T4             RH_4             T5
##  Min.   :28.77   Min.   :15.10   Min.   :28.14   Min.   :15.33
##  1st Qu.:36.90   1st Qu.:19.50   1st Qu.:35.56   1st Qu.:18.28
##  Median :38.56   Median :20.60   Median :38.43   Median :19.39
##  Mean   :39.26   Mean   :20.84   Mean   :39.05   Mean   :19.59
##  3rd Qu.:41.76   3rd Qu.:22.10   3rd Qu.:42.13   3rd Qu.:20.60
##  Max.   :50.16   Max.   :26.20   Max.   :51.09   Max.   :25.80
##       RH_5             T6              RH_6             T7
##  Min.   :29.82   Min.   :-6.065   Min.   : 1.00   Min.   :15.39
##  1st Qu.:45.50   1st Qu.: 3.592   1st Qu.:30.40   1st Qu.:18.70
##  Median :49.10   Median : 7.293   Median :55.47   Median :20.02
##  Mean   :51.03   Mean   : 7.882   Mean   :54.81   Mean   :20.26
##  3rd Qu.:53.72   3rd Qu.:11.204   3rd Qu.:83.30   3rd Qu.:21.60
##  Max.   :96.32   Max.   :28.236   Max.   :99.90   Max.   :26.00
##       RH_7             T8             RH_8             T9             RH_9
##  Min.   :23.2   Min.   :16.31   Min.   :29.60   Min.   :14.89   Min.   :29.17
##  1st Qu.:31.5   1st Qu.:20.79   1st Qu.:39.09   1st Qu.:18.00   1st Qu.:38.53
##  Median :34.9   Median :22.12   Median :42.40   Median :19.39   Median :40.90
##  Mean   :35.4   Mean   :22.03   Mean   :42.96   Mean   :19.48   Mean   :41.57
##  3rd Qu.:39.0   3rd Qu.:23.39   3rd Qu.:46.59   3rd Qu.:20.60   3rd Qu.:44.33
##  Max.   :51.4   Max.   :27.23   Max.   :58.78   Max.   :24.50   Max.   :53.33
##      T_out          Press_mm_hg        RH_out          Windspeed
##  Min.   :-5.000   Min.   :729.3   Min.   : 24.00   Min.   : 0.000
##  1st Qu.: 3.646   1st Qu.:750.9   1st Qu.: 70.33   1st Qu.: 2.000
##  Median : 6.900   Median :756.0   Median : 83.67   Median : 3.667
##  Mean   : 7.393   Mean   :755.5   Mean   : 79.80   Mean   : 4.053
##  3rd Qu.:10.400   3rd Qu.:760.9   3rd Qu.: 91.67   3rd Qu.: 5.500
##  Max.   :26.033   Max.   :772.3   Max.   :100.00   Max.   :14.000
##    Visibility      Tdewpoint           rv1                rv2
##  Min.   : 1.00   Min.   :-6.600   Min.   : 0.00603   Min.   : 0.00603
##  1st Qu.:29.00   1st Qu.: 0.900   1st Qu.:12.58015   1st Qu.:12.58015
##  Median :40.00   Median : 3.467   Median :25.00667   Median :25.00667
##  Mean   :38.34   Mean   : 3.755   Mean   :25.02091   Mean   :25.02091
##  3rd Qu.:40.00   3rd Qu.: 6.567   3rd Qu.:37.56472   3rd Qu.:37.56472
##  Max.   :66.00   Max.   :15.317   Max.   :49.99653   Max.   :49.99653
```

```
str(train)
```

```
## 'data.frame':    15788 obs. of  29 variables:
##  $ date       : chr  "2016-03-28 04:20:00" "2016-02-27 04:10:00" "2016-02-28 05:10:0
0" "2016-03-08 10:10:00" ...
##  $ Appliances : int  60 50 70 50 120 60 40 40 40 40 ...
##  $ lights     : int  0 0 0 0 0 30 0 0 0 0 ...
##  $ T1         : num  21.5 20.1 20.2 19.4 24.8 ...
##  $ RH_1       : num  38.2 36.6 35.2 37.6 41.6 ...
##  $ T2         : num  18.8 18.3 18.2 17.6 23.8 ...
##  $ RH_2       : num  41.6 37.3 36 40 39.3 ...
##  $ T3         : num  22.7 20.5 20.6 20.2 25.4 ...
##  $ RH_3       : num  38.2 37.3 36.6 35.8 38.1 ...
##  $ T4         : num  20 19.2 18.9 18.7 24.3 ...
##  $ RH_4       : num  39 35.1 33.5 35.8 39.8 ...
##  $ T5         : num  19.9 18.6 17.6 17.6 23.8 ...
##  $ RH_5       : num  47.5 56.3 50.6 45 44.3 ...
##  $ T6         : num  7.56 0.167 -0.55 2.56 15.033 ...
##  $ RH_6       : num  56.7 68.9 60.2 79.6 1 ...
##  $ T7         : num  21.4 18.8 19.5 17.9 23.5 ...
##  $ RH_7       : num  38.2 35.8 34.3 32.4 33.9 ...
##  $ T8         : num  23.1 20.2 21.2 19.5 24.7 ...
##  $ RH_8       : num  43 44.3 40.8 39.5 40 ...
##  $ T9         : num  20.3 17.7 18 17.4 22.7 ...
##  $ RH_9       : num  42.7 40.6 39.8 36.8 37.8 ...
##  $ T_out      : num  8.53 1.3 -0.1 2.13 13.6 ...
##  $ Press_mm_hg: num  744 750 755 757 758 ...
##  $ RH_out     : num  75.3 82.7 77.7 95.5 65.3 ...
##  $ Windspeed  : num  10 3 5.17 2.83 5.67 ...
##  $ Visibility : num  48.3 20.8 24.3 54.3 32.7 ...
##  $ Tdewpoint  : num  4.27 -1.35 -3.55 1.48 7.2 ...
##  $ rv1        : num  25.44 26.31 24.05 2.03 1.28 ...
##  $ rv2        : num  25.44 26.31 24.05 2.03 1.28 ...
```

```
head(train)
```

	date	Appliances	lights	T1	RH_1	T2	RH_2	T
	<chr>	<int>	<int>	<dbl>	<dbl>	<dbl>	<dbl>	<db
11013	2016-03-28 04:20:00	60	0	21.50000	38.20000	18.82333	41.62667	22.700
6692	2016-02-27 04:10:00	50	0	20.10000	36.59000	18.29000	37.29000	20.463
6842	2016-02-28 05:10:00	70	0	20.16667	35.20000	18.20000	36.00000	20.600
8168	2016-03-08 10:10:00	50	0	19.39000	37.59000	17.63333	40.02667	20.200
19307	2016-05-24 18:40:00	120	0	24.76000	41.59667	23.79000	39.32667	25.356
5087	2016-02-16 00:40:00	60	30	21.29000	39.09000	19.13333	40.20000	21.000

6 rows | 1-10 of 30 columns

```
colSums(is.na(train))
```

```
##        date  Appliances      lights          T1        RH_1          T2
##           0           0           0           0           0           0
##        RH_2          T3        RH_3          T4        RH_4          T5
##           0           0           0           0           0           0
##        RH_5          T6        RH_6          T7        RH_7          T8
##           0           0           0           0           0           0
##        RH_8          T9        RH_9       T_out Press_mm_hg      RH_out
##           0           0           0           0           0           0
##   Windspeed  Visibility   Tdewpoint         rv1         rv2
##           0           0           0           0           0
```

Data Visualization

Create at least 2 informative graphs, using the training data We use a boxplot to compare the temperature in the kitchen vs in the living room area and a violin plot to see the energy use of appliances

```
boxplot(train$T1,train$T2,
        xlab="Kitchen vs Living room area", ylab="Temp in Celsius")
#install.packages("vioplot")
library("vioplot")
```

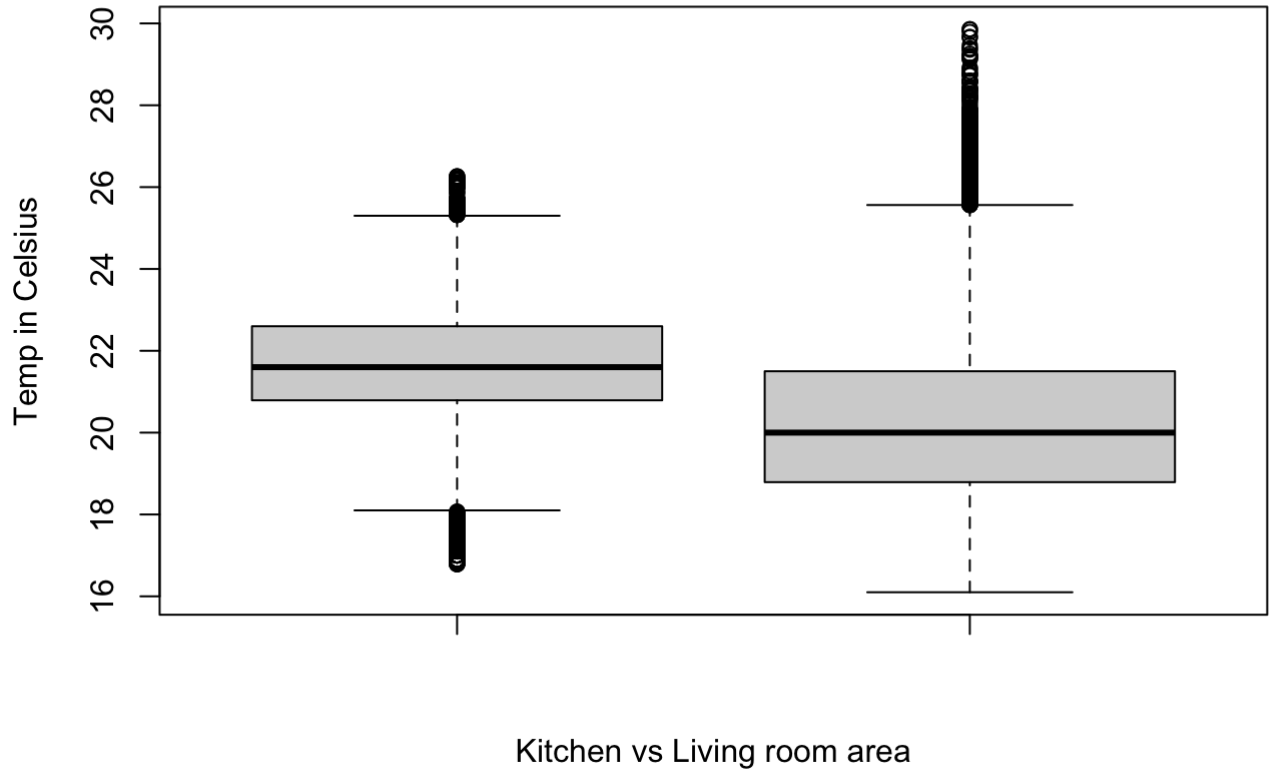
```
## Loading required package: sm
```

```
## Package 'sm', version 2.2-5.7: type help(sm) for summary information
```

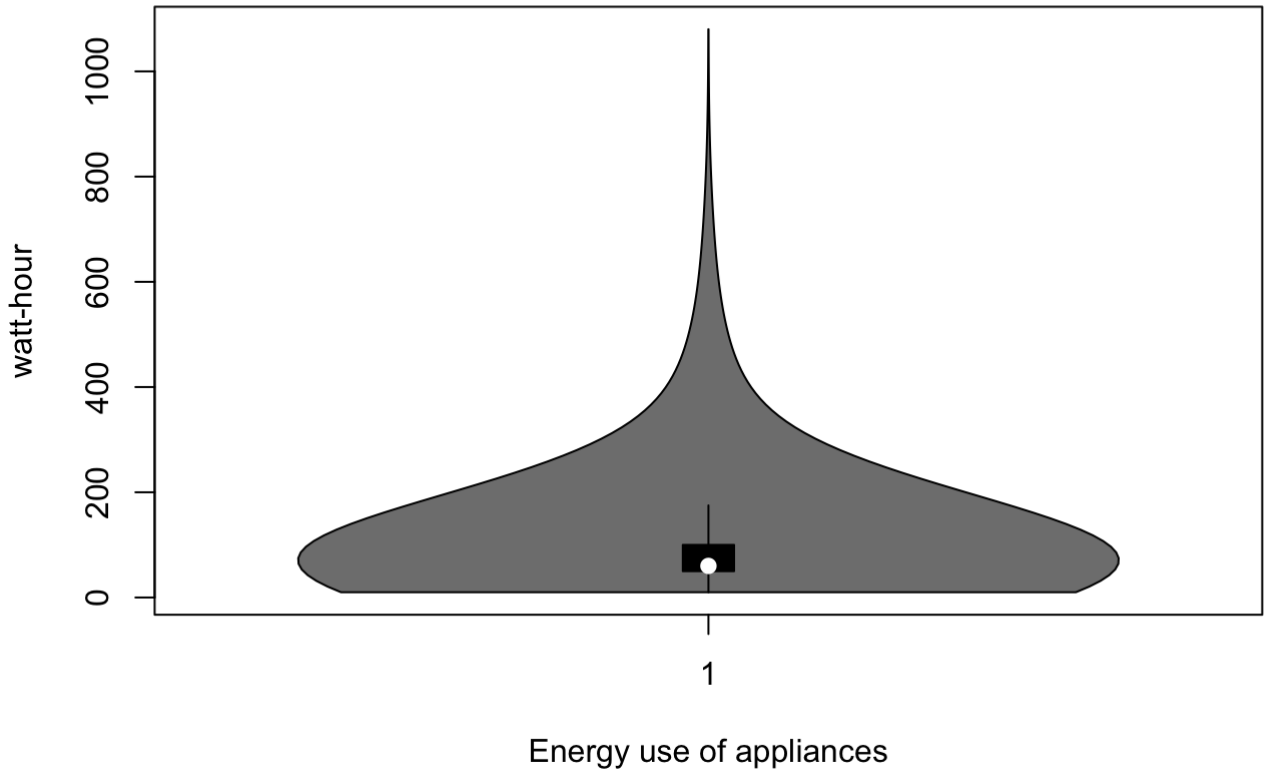
```
## Loading required package: zoo
```

```
##
## Attaching package: 'zoo'
```

```
## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```




```
vioplot(train$Appliances,
        xlab="Energy use of appliances", ylab="watt-hour")
```



##

Linear Regression

Build a simple linear regression model (one predictor) and output the summary. We use the lm and summary function for this

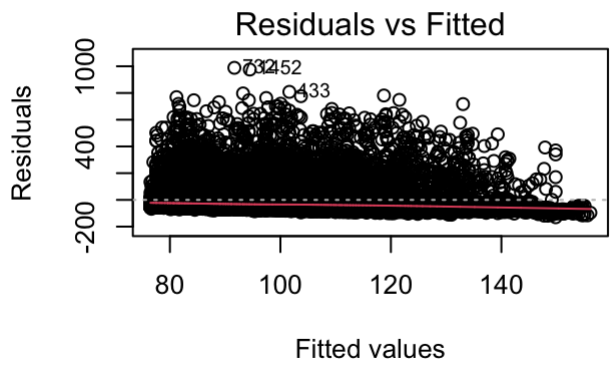
```
lm1<-lm(Appliances~RH_out, data=df)
summary(lm1)
```

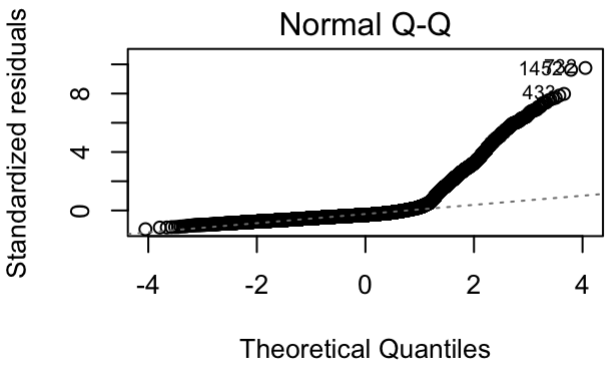
```
##
## Call:
## lm(formula = Appliances ~ RH_out, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -129.82  -46.04  -30.67   -3.22  988.33
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 181.25410    3.92739   46.15   <2e-16 ***
## RH_out       -1.04776    0.04841  -21.64   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 101.3 on 19733 degrees of freedom
## Multiple R-squared:  0.02319,    Adjusted R-squared:  0.02314
## F-statistic: 468.5 on 1 and 19733 DF,  p-value: < 2.2e-16
```

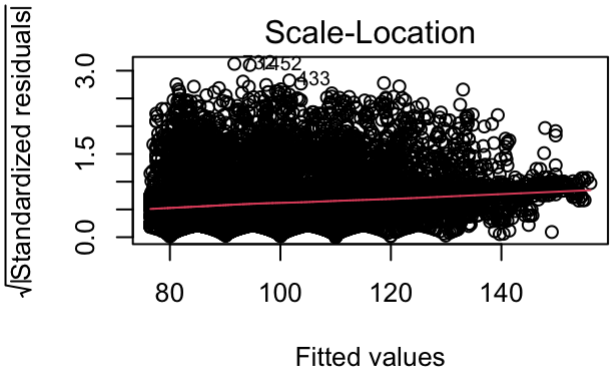
We learn the following from the data: The b is 181 and the w is -1.05. For every percentage increase in humidity, we can expect the Appliance energy use to decrease by 1.05 Wh. The three asterisks tell us that R thought that Rh_out was a good predictor. The R-squared is low, closer to 0, which means that this wasn't a good fit. Our F-statistic is greater than 1 and our p-value is very low. Therefore this is an okay model.

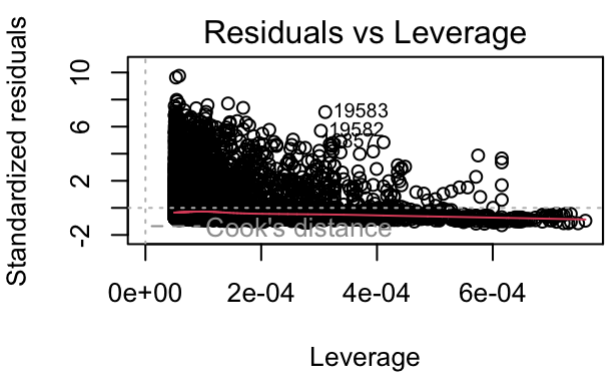
Plot the residuals

```
par(mfrow=c(2,2)) #change panel layout to 2 by 2
plot(lm1)
```









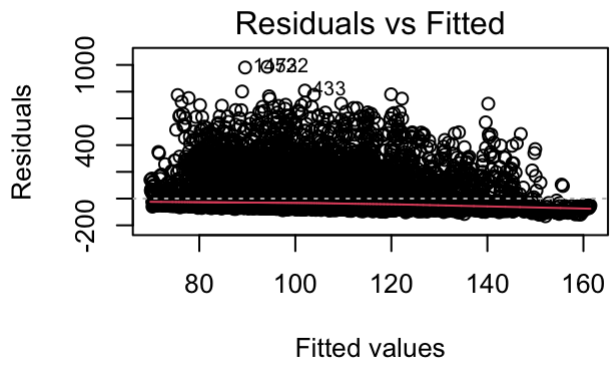
Residuals show how poorly a model represents data. The residuals vs fitted graph show a horizontal line which indicates a linear relationship but the data is not spread evenly around the line. The Normal Q-Q shows that initially, the residuals fit on the straight line but then do not which isn't good. The Scale-Location plot doesn't show us if the points are spaced out equally but they do not seem to be as some points are closer than others. From the residuals vs leverage plot, we can see that most of the points have lower leverage(which means that if we remove these observations the coefficients of the model would not change noticeably). Most points are outside of Cook's distance and are considered to be influential observations.

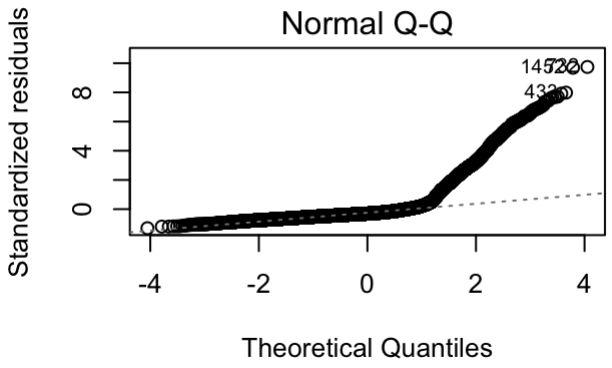
Next, we repeat the same process using different combinations of predictors

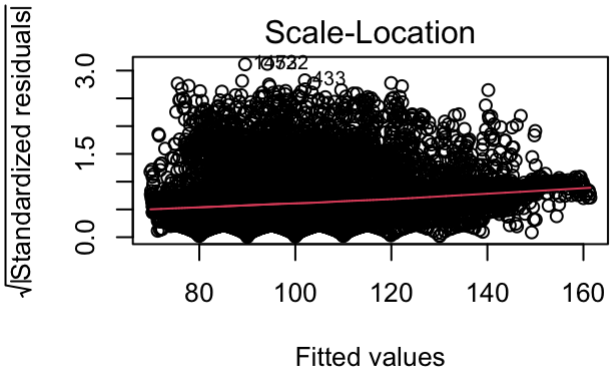
```
lm2<-lm(Appliances~RH_out+T2+T6, data=df)
summary(lm2)
```

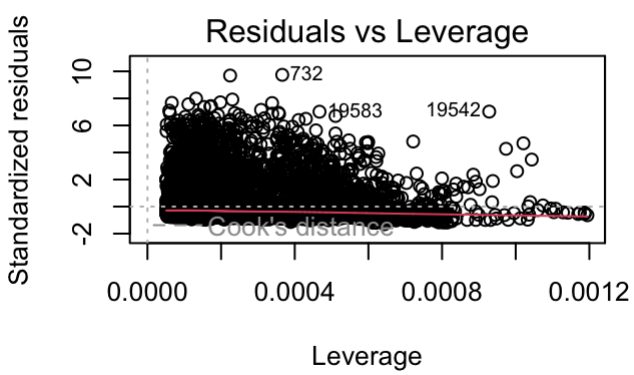
```
##
## Call:
## lm(formula = Appliances ~ RH_out + T2 + T6, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -132.09  -45.66  -30.93   -3.79  985.81
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 111.65363   11.83949   9.431  < 2e-16 ***
## RH_out       -0.84064    0.05910 -14.225  < 2e-16 ***
## T2            2.58466    0.55177   4.684 2.83e-06 ***
## T6            0.06413    0.20851   0.308    0.758
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 101.2 on 19731 degrees of freedom
## Multiple R-squared:  0.02569,    Adjusted R-squared:  0.02554
## F-statistic: 173.4 on 3 and 19731 DF,  p-value: < 2.2e-16
```

```
par(mfrow=c(2,2))
plot(lm2)
```





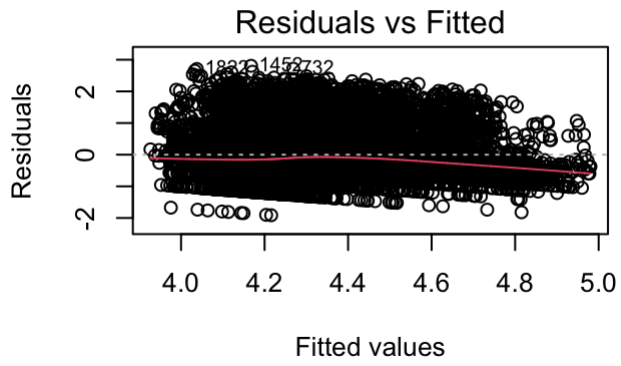


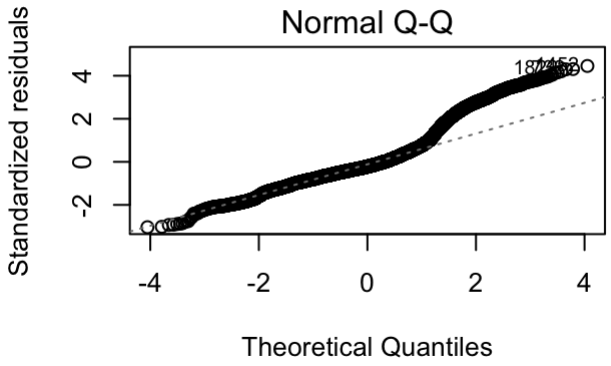


```
lm3<-lm(log(Appliances)~RH_out+T2+T6+Windspeed+T_out, data=df)
summary(lm3)
```

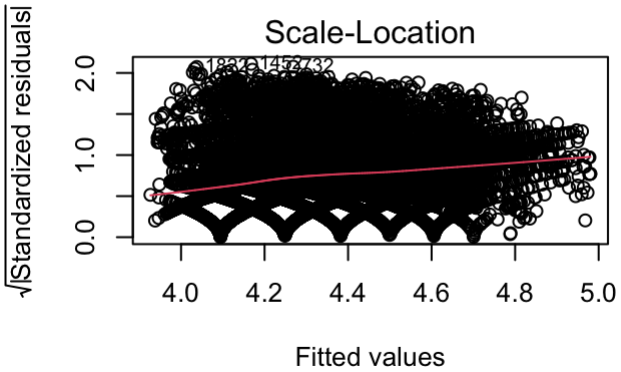
```
##
## Call:
## lm(formula = log(Appliances) ~ RH_out + T2 + T6 + Windspeed +
##     T_out, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.9132 -0.3758 -0.1399  0.2313  2.8077
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.8447162  0.0764368  50.299   <2e-16 ***
## RH_out      -0.0071112  0.0003719 -19.122   <2e-16 ***
## T2           0.0503295  0.0034932  14.408   <2e-16 ***
## T6           0.0427923  0.0033863  12.637   <2e-16 ***
## Windspeed    0.0187737  0.0019053   9.853   <2e-16 ***
## T_out       -0.0555907  0.0038477 -14.448   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6304 on 19729 degrees of freedom
## Multiple R-squared:  0.07749,    Adjusted R-squared:  0.07726
## F-statistic: 331.5 on 5 and 19729 DF,  p-value: < 2.2e-16
```

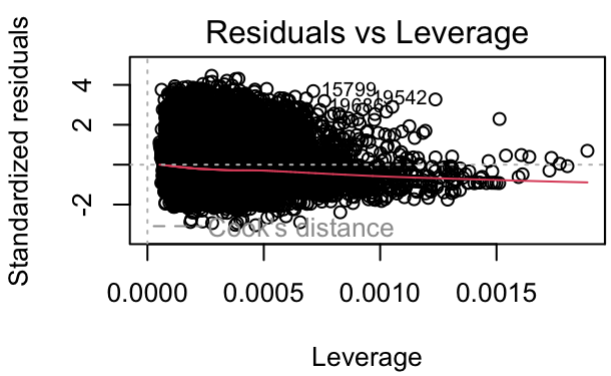
```
par(mfrow=c(2,2))
plot(lm3)
```





We think the third model is the best because the residuals are spread out more evenly around the red lines. The red lines are also more linear. The adjusted R-squared is also 3 times higher than in the other two models. All the predictors in the 3rd model have 3 asterisks which indicate that R thinks they are good predictors. Although, the F statistic is lower than the first but higher than the second





Evaluate on test data

Using the 3 models, we will predict and evaluate on the test data using metrics correlation and mse(mean squared error).

```
pred1 <- predict(lm1, newdata=test)
cor1 <- cor(pred1, test$Appliances)
mse1 <- mean((pred1-test$Appliances)^2)
rmse1 <- sqrt(mse1)
print(paste('correlation:', cor1))
```

```
## [1] "correlation: 0.138709809724305"
```

```
print(paste('mse:', mse1))
```

```
## [1] "mse: 11060.7224216856"
```

```
print(paste('rmse:', rmse1))
```

```
## [1] "rmse: 105.169969200745"
```

```
pred2 <- predict(lm2, newdata=test)
cor2 <- cor(pred2, test$Appliances)
mse2 <- mean((pred2-test$Appliances)^2)
rmse2 <- sqrt(mse2)
print(paste('correlation:', cor2))
```

```
## [1] "correlation: 0.142422230386791"
```

```
print(paste('mse:', mse2))
```

```
## [1] "mse: 11050.5206837333"
```

```
print(paste('rmse:', rmse2))
```

```
## [1] "rmse: 105.121456818926"
```

```
pred3 <- predict(lm3, newdata=test)
pred3<-exp(pred3)
cor3 <- cor(pred3, test$Appliances)
mse3 <- mean((pred3-test$Appliances)^2)
rmse3 <- sqrt(mse3)
print(paste('correlation:', cor3))
```

```
## [1] "correlation: 0.169358695215322"
```

```
print(paste('mse:', mse3))
```

```
## [1] "mse: 11490.9245064184"
```

```
print(paste('rmse:', rmse3))
```

```
## [1] "rmse: 107.195729888921"
```

Results

We can see that as we add more and more predictors, there is a significant improvement in the correlation. We ideally want to see a correlation close to +1 or -1 and we can see that the correlation of the 3rd model is closer to +1 than the other two. Adding more predictors allows the model to account for more variations that are not

accounted for by a lower number of predictors. While the third model did have the best correlation, it suffered from a higher MSE value than the other models indicating a higher presence of errors. Model three, however, is still better able to fit the data and thus produce better results due to its correlation. In general, however, these three models are ineffective at explaining the high variance in the data and produce inaccurate results, indicating that linear regression may not be the best approach for this data set.