Project 2 - Regression

Siri Bafna

Weather History Data Analysis

Source: Kaggle.com

Link: https://www.kaggle.com/budincsevity/szeged-weather

Number of Observations: 98.5K # ## Data Cleaning

Factoring character variables, cleaning NA's, reading in data

Data cleaning for this data set included making the character variables, which was Summary and Daily Summary into factors, as well as not including 'preciptype', 'loud_cover', 'date', as these did not directly affect the the Humidity which is the target.

```
weatherHistory <- read.csv("/Users/siri/Downloads/weatherHistory.csv", header=TRUE)
weatherHistory <- weatherHistory[,c(2,4,5,6,7, 8, 9, 11, 12)]
summary(weatherHistory) # data function # 1
```

```
   Summary           Temperature      Apparent_Temperature
 Length:96453       Min.   :-21.822   Min.   :-27.717
 Class :character   1st Qu.:  4.689   1st Qu.:  2.311
 Mode  :character   Median : 12.000   Median : 12.000
                    Mean   : 11.933   Mean   : 10.855
                    3rd Qu.: 18.839   3rd Qu.: 18.839
                    Max.   : 39.906   Max.   : 39.344
    Humidity        Wind_Speed       Wind_Bearing     Visibility
 Min.   :0.0000   Min.   : 0.000   Min.   :  0.0   Min.   : 0.00
 1st Qu.:0.6000   1st Qu.: 5.828   1st Qu.:116.0   1st Qu.: 8.34
 Median :0.7800   Median : 9.966   Median :180.0   Median :10.05
 Mean   :0.7349   Mean   :10.811   Mean   :187.5   Mean   :10.35
 3rd Qu.:0.8900   3rd Qu.:14.136   3rd Qu.:290.0   3rd Qu.:14.81
 Max.   :1.0000   Max.   :63.853   Max.   :359.0   Max.   :16.10
    Pressure     Daily_Summary
 Min.   :   0   Length:96453
 1st Qu.:1012   Class :character
 Median :1016   Mode  :character
```

```
 Mean   :1003
 3rd Qu.:1021
 Max.   :1046
weatherHistory$Summary <- as.factor(weatherHistory$Summary)
weatherHistory$Daily.Summary <- as.factor(weatherHistory$Daily_Summary)
```

Divide into train and test

```
set.seed(1234)
i <- sample(1:nrow(weatherHistory), .75*nrow(weatherHistory), replace=FALSE)
train <- weatherHistory[i,]
test <- weatherHistory[-i,]
```

The feature selection of the following algorithms include all columns except Summary and Daily summary, as they resulted in error prone results and more than 32 levels of results.

Linear Regression - Algorithm 1

```
library(ISLR)
lm1 <- lm(Humidity~Temperature+Wind_Speed+Visibility+Apparent_Temperature+Wind_Bearing
+Pressure, data=train)
summary(lm1) # data exploration # 2

Call:
lm(formula = Humidity ~ Temperature + Wind_Speed + Visibility +
    Apparent_Temperature + Wind_Bearing + Pressure, data = train)

Residuals:
     Min       1Q   Median       3Q      Max
-1.25533 -0.09365  0.01189  0.10138  0.37051

Coefficients:
                       Estimate Std. Error t value Pr(>|t|)
(Intercept)           1.014e+00  4.899e-03 206.922   <2e-16 ***
Temperature          -3.229e-02  5.438e-04 -59.390   <2e-16 ***
Wind_Speed           -4.086e-03  9.190e-05 -44.459   <2e-16 ***
Visibility           -5.522e-03  1.383e-04 -39.931   <2e-16 ***
Apparent_Temperature  1.825e-02  4.857e-04  37.574   <2e-16 ***
```

```
Wind_Bearing          7.284e-05  4.960e-06  14.684   <2e-16 ***
Pressure             -4.076e-06  4.519e-06  -0.902    0.367
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.1422 on 72332 degrees of freedom
Multiple R-squared:  0.4725,    Adjusted R-squared:  0.4725
F-statistic: 1.08e+04 on 6 and 72332 DF,  p-value: < 2.2e-16
```

Accuracy and Predictions for Linear Regression

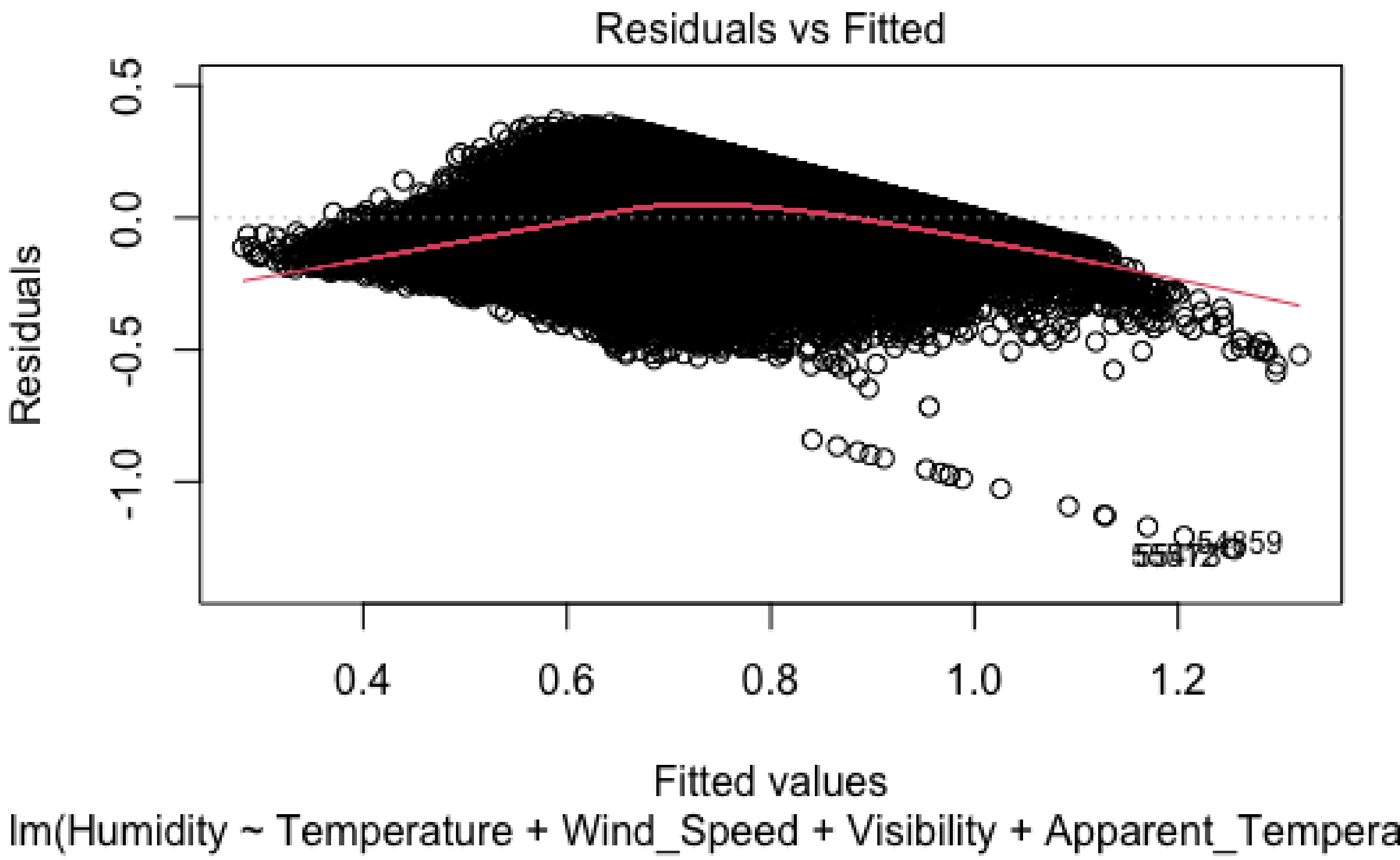
```
pred <- predict(lm1, newdata=test)
acc <- cor(pred, test$Humidity)
mse <- mean((pred - test$Humidity) ^2)
print("Correlation:")
[1] "Correlation:"
print(acc)
[1] 0.6850601
print("MSE:")
[1] "MSE:"
print(mse)
[1] 0.02010298
```

Commentary on Linear Regression

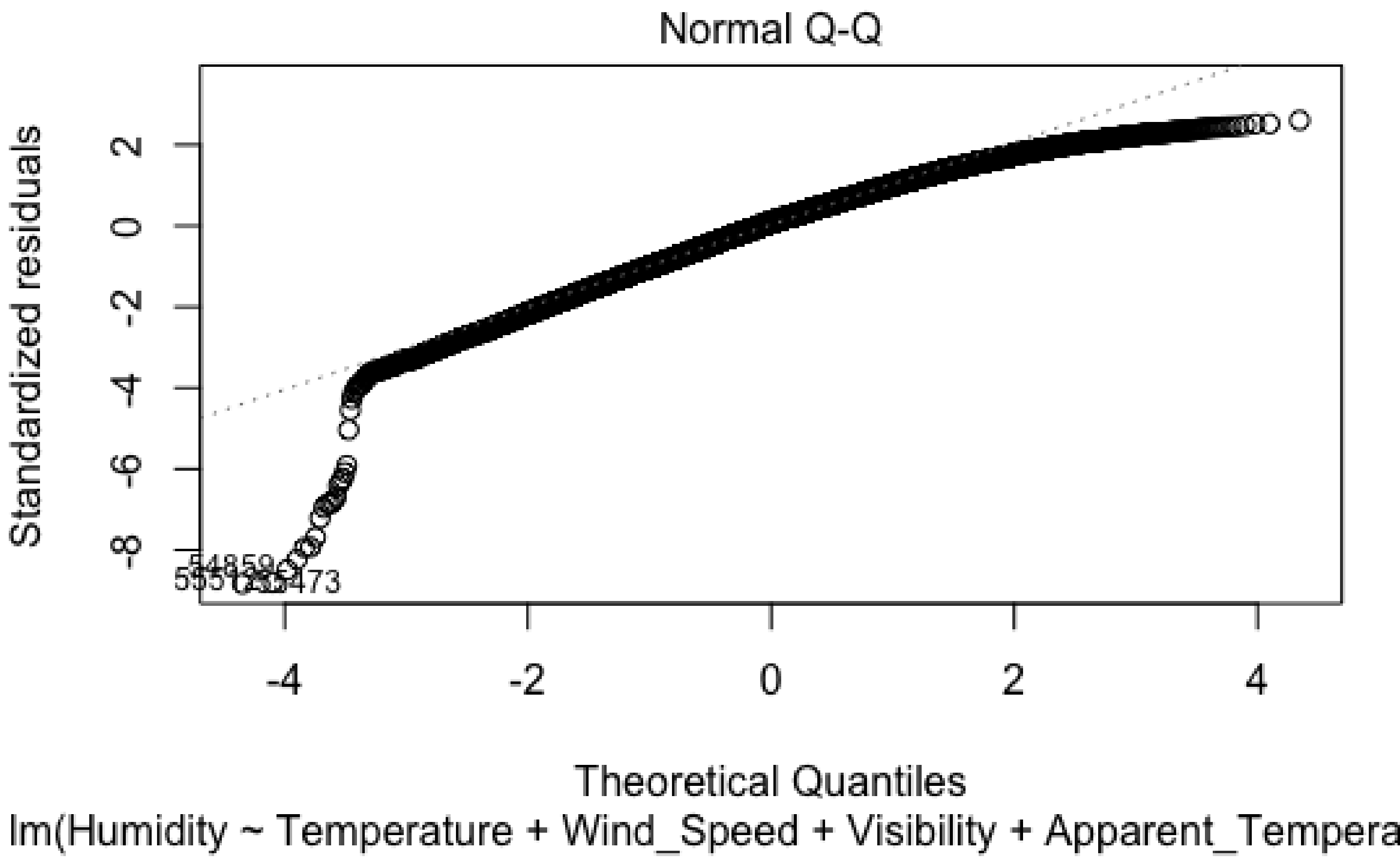
The linear model for this data set was a pretty average performance, with an accuracy percentage of 68%. This is lower than how efficient linear regression usually performs, and it seems like it predicted all but Pressure as an efficient predictor. Additionally, the p-value is relatively low which is a pro, whereas R-squared value is .4 which is average! Therefore, with it's good and bad aspects, it was a pretty average performance.

Data Exploration Plots

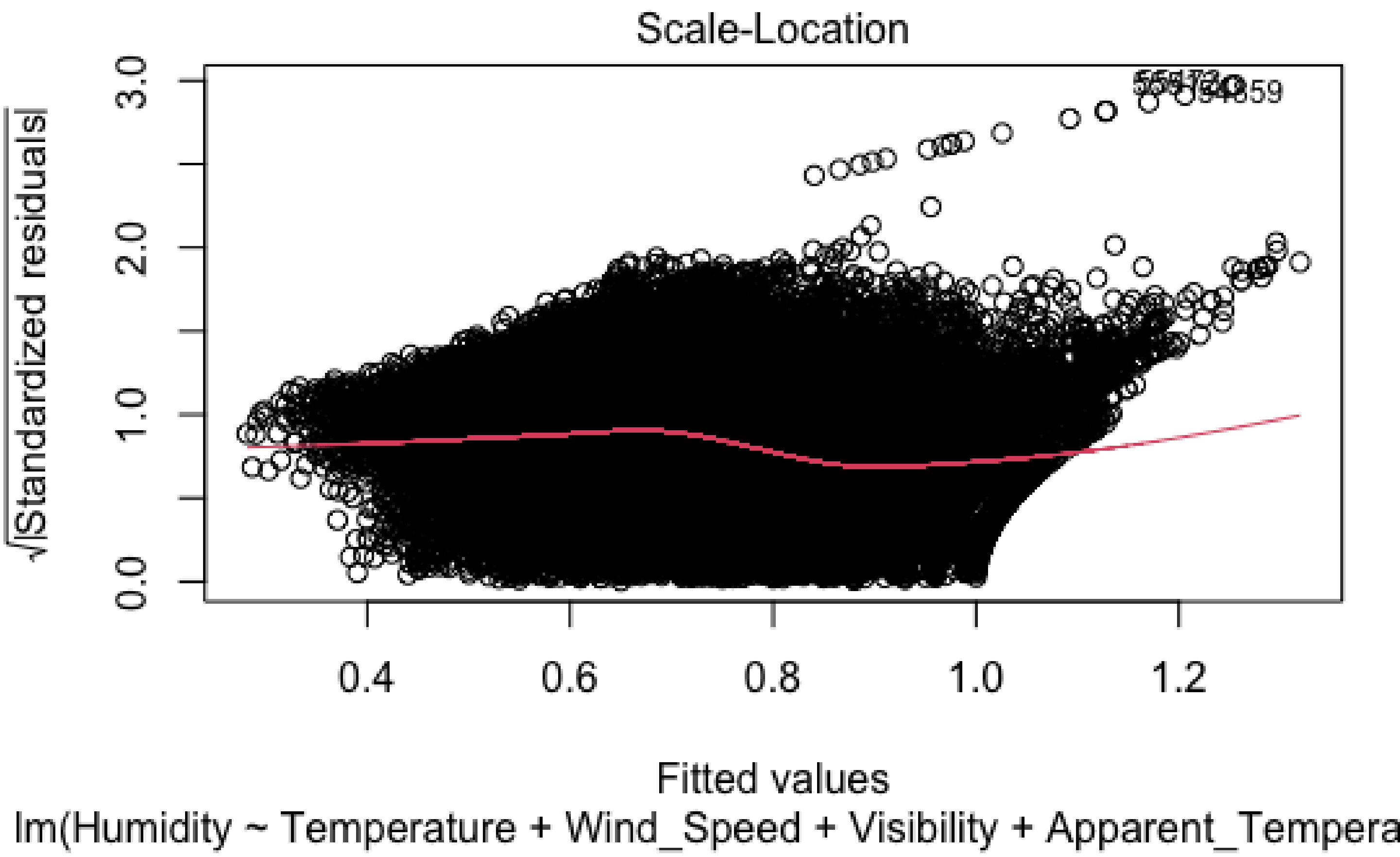
```
plot(lm1) # data exploration plot # 1
```

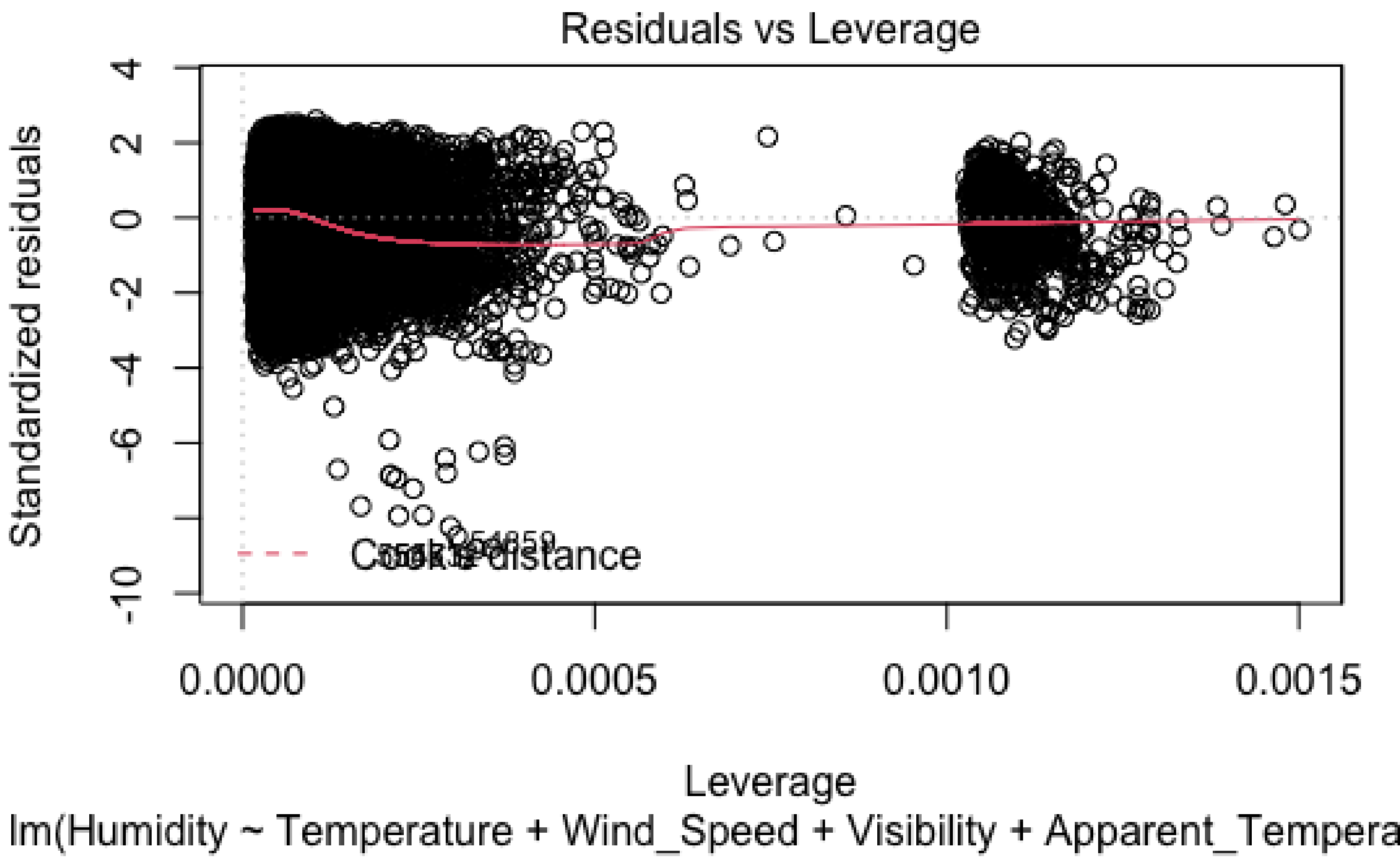


Residuals vs Fitted
Residuals
0.5
0.0
-0.5
-1.0
0.4
0.6
0.8
1.0
1.2
Fitted values
lm(Humidity ~ Temperature + Wind_Speed + Visibility + Apparent_Tempera
54859

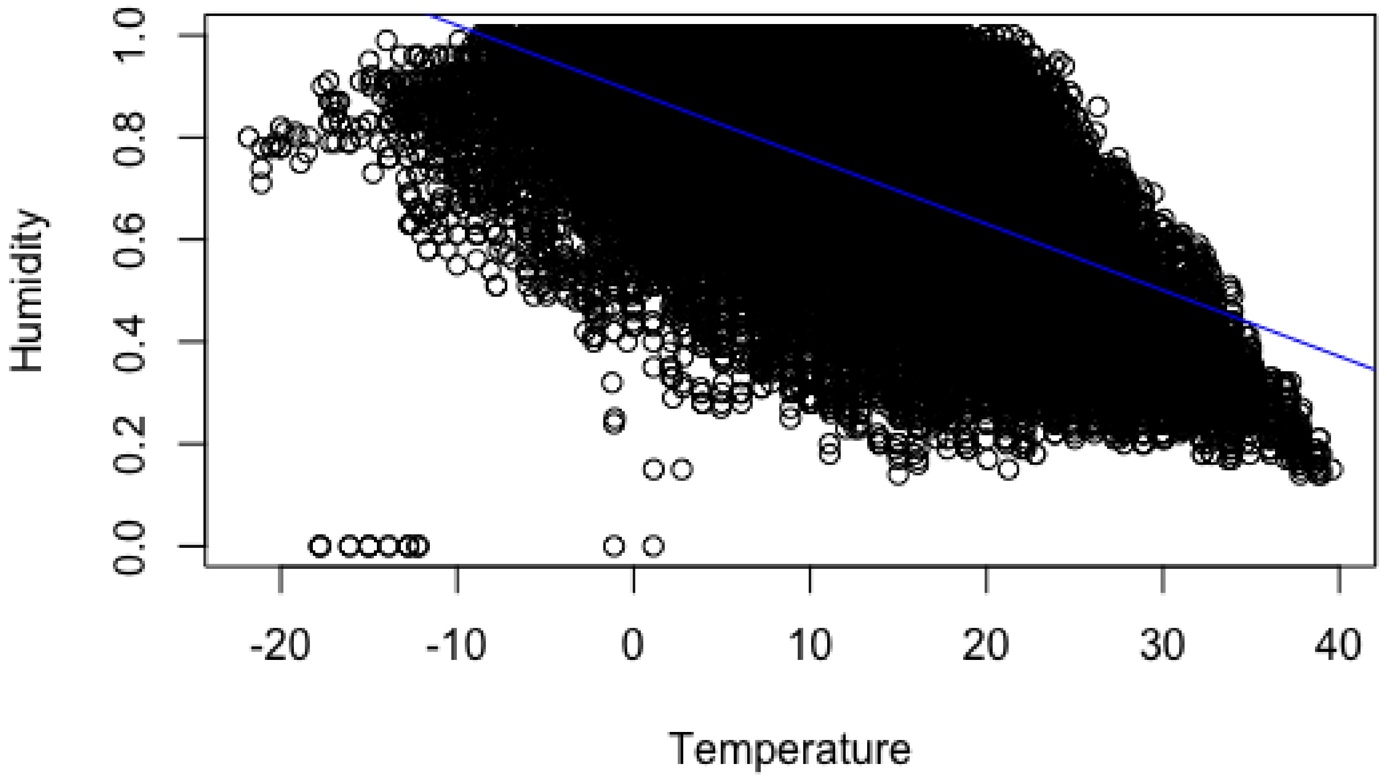


Normal Q-Q
Standardized residuals
Theoretical Quantiles
lm(Humidity ~ Temperature + Wind_Speed + Visibility + Apparent_Tempera


Scale-Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
2.5
3.0
0.4
0.6
0.8
1.0
1.2
Fitted values
lm(Humidity ~ Temperature + Wind_Speed + Visibility + Apparent_Tempera




```
plot(train$Humidity~train$Temperature, xlab="Temperature", ylab="Humidity")
abline(lm(train$Humidity~train$Temperature), col="blue")
```



kNN - Algorithm # 2

```
library(caret)
fit <- knnreg(train[,c(2,3,5,6,7,8)],train[,4])
```

Accuracy and Predictions for kNN

```
testpred <- predict(fit, test[,c(2,3,5,6,7,8)])
correlation_knn <- cor(testpred, test$Humidity)
mse_knn <- mean((testpred - test$Humidity) ^2)
print(paste("correlation: ", correlation_knn))
[1] "correlation:  0.783242531333751"
print(paste("mse: ", mse_knn))
[1] "mse:  0.0147434327039157"
```

Commentary on kNN

The performance on kNN was optimal at a rate of 78% along with a low mse at .014. The use of nearest neighbors was efficient in this case as all the predictors do effectively impact the algorithm.

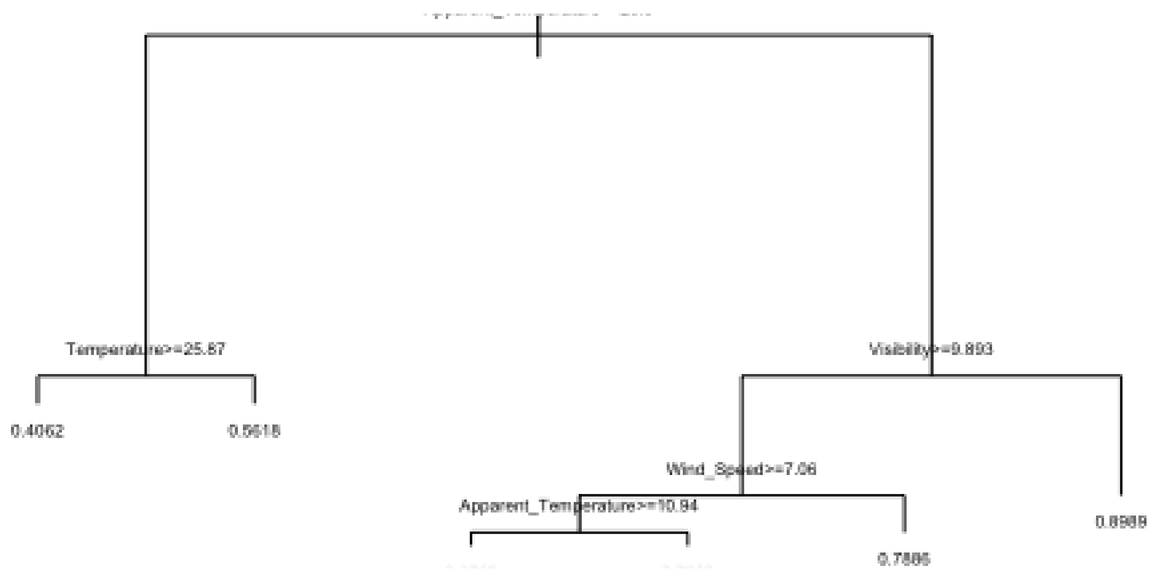
Decision Trees - Algorithm # 3

```
library(rpart)

dtree <- rpart(Humidity~Temperature+Wind_Speed+Visibility+Apparent_Temperature+Wind_Be
aring+Pressure, data=train)

dtree

n= 72339


node), split, n, deviance, yval

      * denotes terminal node


 1) root 72339 2771.96600 0.7345190

   2) Apparent_Temperature>=20.89722 14172  348.26560 0.4930469

     4) Temperature>=25.86944 6264   71.83489 0.4061925 *

     5) Temperature< 25.86944 7908  191.74720 0.5618450 *

   3) Apparent_Temperature< 20.89722 58167 1396.01300 0.7933521

     6) Visibility>=9.89345 37341  882.23560 0.7345117

      12) Wind_Speed>=7.05985 25622  609.59430 0.7097752

        24) Apparent_Temperature>=10.93611 13330  371.17820 0.6715484 *

        25) Apparent_Temperature< 10.93611 12292  197.81320 0.7512301 *

      13) Wind_Speed< 7.05985 11719  222.68580 0.7885946 *

     7) Visibility< 9.89345 20826  152.69350 0.8988529 *

summary(dtree)

Call:

rpart(formula = Humidity ~ Temperature + Wind_Speed + Visibility +

    Apparent_Temperature + Wind_Bearing + Pressure, data = train)

  n= 72339


          CP nsplit rel error    xerror        xstd

1 0.37074316      0 1.0000000 1.0000102 0.004610020

2 0.13026274      1 0.6292568 0.6298421 0.003725065

3 0.03055000      2 0.4989941 0.4994272 0.003141537

4 0.01802171      3 0.4684441 0.4690782 0.003012789

5 0.01464768      4 0.4504224 0.4511654 0.002931187
```

```
6 0.01000000      5 0.4357747 0.4372307 0.002853882

Variable importance
         Temperature Apparent_Temperature           Visibility
                  43                   42                   13
          Wind_Speed             Pressure
                   2                    1

Node number 1: 72339 observations,    complexity param=0.3707432
  mean=0.734519, MSE=0.03831911
  left son=2 (14172 obs) right son=3 (58167 obs)
  Primary splits:
      Apparent_Temperature < 20.89722  to the right, improve=0.37074320, (0 missing)
      Temperature          < 20.89722  to the right, improve=0.37074320, (0 missing)
      Visibility           < 9.95785   to the right, improve=0.28530590, (0 missing)
      Wind_Speed           < 6.89885   to the right, improve=0.05095032, (0 missing)
      Pressure             < 1026.355  to the left,  improve=0.01397541, (0 missing)
  Surrogate splits:
      Temperature < 20.89722  to the right, agree=1, adj=1, (0 split)

Node number 2: 14172 observations,    complexity param=0.03055
  mean=0.4930469, MSE=0.02457421
  left son=4 (6264 obs) right son=5 (7908 obs)
  Primary splits:
      Temperature          < 25.86944  to the right, improve=0.24315800, (0 missing)
      Apparent_Temperature < 25.86944  to the right, improve=0.24315800, (0 missing)
      Visibility           < 10.58575  to the left,  improve=0.05796188, (0 missing)
      Wind_Speed           < 6.67345   to the right, improve=0.04019841, (0 missing)
      Pressure             < 1018.135  to the right, improve=0.02504725, (0 missing)
  Surrogate splits:
      Apparent_Temperature < 25.86944  to the right, agree=1.000, adj=1.000, (0 split)
      Visibility           < 10.4489   to the left,  agree=0.566, adj=0.019, (0 split)
      Wind_Bearing         < 353.5     to the right, agree=0.558, adj=0.001, (0 split)

Node number 3: 58167 observations,    complexity param=0.1302627
  mean=0.7933521, MSE=0.02400009
  left son=6 (37341 obs) right son=7 (20826 obs)
```

```
  Primary splits:
      Visibility           < 9.89345   to the right, improve=0.258653700, (0 missing)
      Apparent_Temperature < 10.91389  to the right, improve=0.098566830, (0 missing)
      Temperature          < 10.91389  to the right, improve=0.098566830, (0 missing)
      Wind_Speed           < 11.26195  to the right, improve=0.065810690, (0 missing)
      Pressure             < 1009.915  to the right, improve=0.005898743, (0 missing)
  Surrogate splits:
      Temperature          < 4.819444  to the right, agree=0.722, adj=0.222, (0 split)
      Apparent_Temperature < 2.975     to the right, agree=0.704, adj=0.174, (0 split)
      Pressure             < 1028.015  to the left,  agree=0.666, adj=0.067, (0 split)

Node number 4: 6264 observations
  mean=0.4061925, MSE=0.01146789

Node number 5: 7908 observations
  mean=0.561845, MSE=0.02424724

Node number 6: 37341 observations,    complexity param=0.01802171
  mean=0.7345117, MSE=0.02362646
  left son=12 (25622 obs) right son=13 (11719 obs)
  Primary splits:
      Wind_Speed           < 7.05985   to the right, improve=0.05662383, (0 missing)
      Visibility           < 10.58575  to the left,  improve=0.04213674, (0 missing)
      Temperature          < 16.86944  to the right, improve=0.03795920, (0 missing)
      Apparent_Temperature < 16.86944  to the right, improve=0.03795920, (0 missing)
      Pressure             < 1019.655  to the right, improve=0.01827203, (0 missing)
  Surrogate splits:
      Wind_Bearing < 0.5       to the right, agree=0.695, adj=0.027, (0 split)
      Temperature  < -12.79167 to the right, agree=0.686, adj=0.000, (0 split)
      Pressure     < 1040.46   to the left,  agree=0.686, adj=0.000, (0 split)

Node number 7: 20826 observations
  mean=0.8988529, MSE=0.007331869

Node number 12: 25622 observations,    complexity param=0.01464768
  mean=0.7097752, MSE=0.02379183
  left son=24 (13330 obs) right son=25 (12292 obs)
```

```
  Primary splits:
      Apparent_Temperature < 10.93611  to the right, improve=0.06660637, (0 missing)
      Temperature          < 10.93611  to the right, improve=0.06660637, (0 missing)
      Visibility           < 10.58575  to the left,  improve=0.03076945, (0 missing)
      Pressure             < 1015.085  to the right, improve=0.02505892, (0 missing)
      Wind_Speed           < 20.21355  to the right, improve=0.02231901, (0 missing)
  Surrogate splits:
      Temperature  < 10.93611  to the right, agree=1.000, adj=1.000, (0 split)
      Pressure     < 1021.705  to the left,  agree=0.592, adj=0.149, (0 split)
      Wind_Speed   < 16.80035  to the left,  agree=0.559, adj=0.080, (0 split)
      Visibility   < 11.45515  to the left,  agree=0.530, adj=0.021, (0 split)
      Wind_Bearing < 11.5      to the right, agree=0.523, adj=0.005, (0 split)

Node number 13: 11719 observations
  mean=0.7885946, MSE=0.01900211

Node number 24: 13330 observations
  mean=0.6715484, MSE=0.02784533

Node number 25: 12292 observations
  mean=0.7512301, MSE=0.01609284
plot(dtree) # data exploration plot # 3
text(dtree, cex=0.4, pretty=0)
```



Accuracy and Predictions on Decision Trees

```
pred <- predict(dtree, newdata=test)

table(pred, test$Humidity) # data exploration function


pred                   0 0.12 0.13 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22

  0.406192528735634   0    0    1    1    1    2    2    4  10    8   14

  0.561844967121901   0    0    0    1    0    0    1    2   2    1    1

  0.671548387096781   0    1    0    1    1    1    2    1   1    2    1

  0.751230068337141   0    0    0    0    0    0    0    0   0    0    0

  0.788594589982088   1    0    0    0    0    0    1    0   0    1    0

  0.898852876212437   3    0    0    0    0    0    0    0   0    0    0


pred                 0.23 0.24 0.25 0.26 0.27 0.28 0.29 0.3 0.31 0.32 0.33

  0.406192528735634    15   16   27   31   40   55   70  64   59   66   73

  0.561844967121901     4    4    7    7   15    9   28  22    9   50   34

  0.671548387096781     3    2    1    7    8    5    8  13   20   20   27
```

```
  0.751230068337141    0    1    1    1    0    1    0   2    3    1    2
  0.788594589982088    1    1    0    2    3    1    2   3    2    3    4
  0.898852876212437    0    0    0    1    0    0    0   0    0    0    0

pred                 0.34 0.35 0.36 0.37 0.38 0.39 0.4 0.41 0.42 0.43 0.44
  0.406192528735634   76   74   73   82   54   81  80   69   66   61   72
  0.561844967121901   34   30   32   31   65   45  30   56   50   21   35
  0.671548387096781   23   23   32   39   33   35  40   26   28   41   59
  0.751230068337141    9    3    4    3    5    7   8   19    8    9    9
  0.788594589982088    2    7    5    7    1   10  11    7    6   11   10
  0.898852876212437    1    1    0    0    0    0   2    0    0    1    2

pred                 0.45 0.46 0.47 0.48 0.49 0.5 0.51 0.52 0.53 0.54 0.55
  0.406192528735634   40   67   66   46   53  41   35   31   32   45   36
  0.561844967121901   41   45   49   64   62  82   50   33   84   80   62
  0.671548387096781   41   40   57   56   49  53   59   83   47   51   70
  0.751230068337141   17   15   11   15   31  23   25   21   26   42   30
  0.788594589982088   12   13    9   11   13  11   24   25   27   26   22
  0.898852876212437    0    1    2    2    1   8    1    5    4    2    7

pred                 0.56 0.57 0.58 0.59 0.6 0.61 0.62 0.63 0.64 0.65 0.66
  0.406192528735634   18   20   25   28  23    7   10   11   11    9    5
  0.561844967121901   43   71   56   81  36   47   69   41   57   44   63
  0.671548387096781   96   59   58   71  84   99   84   86   80   79   93
  0.751230068337141   21   45   49   40  40   40   44   58   54   64   75
  0.788594589982088   33   21   32   24  37   40   31   32   40   37   37
  0.898852876212437    2   13    6    8   6   10   10   11   10   20   16

pred                 0.67 0.68 0.69 0.7 0.71 0.72 0.73 0.74 0.75 0.76 0.77
  0.406192528735634    5    8    6   1    2    0    2    5    2    0    0
  0.561844967121901   44   63   57  43   35   29   67   53   12   34   17
  0.671548387096781  104  117   80  61   94  165  118   89   82   70  102
  0.751230068337141   70   70  111 119  143   93  101   94  137  174  145
  0.788594589982088   51   67   50  68   59   91   78   69   89   92   96
  0.898852876212437   19   23   31  19   32   34   35   36   41   48   56

pred                 0.78 0.79 0.8 0.81 0.82 0.83 0.84 0.85 0.86 0.87 0.88
```

```
  0.406192528735634    3    0    0    0    0    0    0    0    0    0    0
  0.561844967121901   26   52   28   28   24   13   18    7    7   19    9
  0.671548387096781  135   75  106  101   50  102  118   46   91   84   42
  0.751230068337141  136  160  126  104  197  167  101  125  160   95   96
  0.788594589982088  104   74   98  111  104  147  168  111  174  169   96
  0.898852876212437   76   83   95  100  130  157  168  215  309  224  219

pred                0.89  0.9 0.91 0.92 0.93 0.94 0.95 0.96 0.97 0.98 0.99
  0.406192528735634    0    0    0    0    0    0    0    0    0    0    0
  0.561844967121901    6   10    4    6    9    0    1    3    4    1    1
  0.671548387096781   50  113   27   25  115    7    9   34   11    4   15
  0.751230068337141  181   44   30   88   90   18   19   45    0    2    8
  0.788594589982088  160  190   51  130  257   21   47  126   45    2   33
  0.898852876212437  319  237  141  720  919  192  160  799   94   56  338

pred                   1
  0.406192528735634    0
  0.561844967121901    0
  0.671548387096781   12
  0.751230068337141   12
  0.788594589982088   31
  0.898852876212437  663
acc <- cor(pred, test$Humidity)
print("Accuracy for Decision Trees")
[1] "Accuracy for Decision Trees"
print(acc)
[1] 0.747848
```

The results of the Decision Trees are slightly above average with the accuracy of 74%. With decision trees, it also creates levels up to 663 for the decision tree.

Random Forest - Ensemble Method

I had to trim the amount of observations we use for Random Forest as it does not efficiently run with big sets, and additionally had the predictors just set at Temperature to test the strongest predictor out of them as a wayof curiosity.

```
weatherHist <- weatherHistory[1:50000,]
```

```
i <- sample(1:nrow(weatherHist), .75*nrow(weatherHist), replace=FALSE)

train_em <- weatherHist[i,]

test_em <- weatherHist[-i,]

str(train_em)

'data.frame':   37500 obs. of  10 variables:

 $ Summary             : Factor w/ 27 levels "Breezy","Breezy and Dry",..: 18 7 18 18
19 18 20 20 13 20 ...

 $ Temperature         : num  11.14 6.04 7.78 20.99 -5.02 ...

 $ Apparent_Temperature: num  11.14 3.71 5.76 20.99 -5.02 ...

 $ Humidity            : num  0.8 0.86 0.58 0.63 0.99 0.89 0.81 0.72 0.99 0.45 ...

 $ Wind_Speed          : num  11.21 11.04 11.27 14.15 4.49 ...

 $ Wind_Bearing        : int  271 159 220 300 92 273 319 42 280 37 ...

 $ Visibility          : num  15.83 6.99 16.1 9.98 3.98 ...

 $ Pressure            : num  1023 1015 1011 1014 1030 ...

 $ Daily_Summary       : chr  "Mostly cloudy throughout the day." "Mostly cloudy start
ing overnight continuing until night." "Partly cloudy starting in the afternoon." "Mos
tly cloudy throughout the day." ...

 $ Daily.Summary       : Factor w/ 214 levels "Breezy and foggy starting in the evenin
g.",..: 112 95 157 112 36 179 170 149 58 170 ...

library(randomForest)

rf <- randomForest(Humidity~Temperature, data=train_em, importance=TRUE)

rf


Call:

 randomForest(formula = Humidity ~ Temperature, data = train_em,      importance = TRU
E)

               Type of random forest: regression

                     Number of trees: 500

No. of variables tried at each split: 1


          Mean of squared residuals: 0.02032372

                    % Var explained: 45.09

pred2 <- predict(rf, newdata=test_em, type="response")

acc_rf <- cor(pred2, test_em$Humidity)

print("Accuracy for Random Forest")

[1] "Accuracy for Random Forest"

print(acc_rf)

[1] 0.6699508
```

The performance of Random Forest was average, similar to linear regression. With a variance of 45.09 and an accuracy of 64%, it had higher than normal variance levels, all which result it to be an average performance.

Results Analysis

kNN Accuracy: 78.324 | Decision Trees Accuracy: 74.01 | Linear Regression Accuracy: 68.50601 |

In this data set, kNN performed the best with an accuracy of 78%, Decision Trees were close after and the lowest quality of performance happened to be linear regression. These results surprised me as according to the plots, target to predictors, the pattern was obviously linear. Therefore, linear regression should have been able to outperform more than it actually did. According to the model, most attributes were strong predictors for the data. However, kNN did outperform all the other algorithms because it uses an efficient similarity measure when comparing algorithms. A good pattern recognition technique that resulted in good performance because all the predictors were related to the target and therefore resulted in obvious patterns in data. Decision Trees were averagely performing with an accuracy of 74% and an MSE of 3.831911, which is incredibly low. The MSE should have resulted in a higher accuracy rate but didn’t as decision trees tend to be unstable and highly bias.