Linear Models: Regression

Isabelle Kirby, Bridgette Bryant

Linear Regression:

In simple terms, linear regression tries to predict a quantitative target by fitting a linear model to the input of the predictors to best fit the target in the training data. A simple linear regression model is often in the form of y = wX + b where y values as the targets, X values as the predictors, the linear model creates the weight matrix w and the fitting parameter b. Linear models don't always have to be straight lines, they can be complex polynomials with multiple predictor variables. Once a linear regression model is created/trained, you can judge it's performance by the standard error, t-value, and most of all the p-value. A p-value very close to 0 is the most ideal, R will show you with '***' to ' ' next to the value with '***' showing an excellent p-value. Residual standard error is also another way to measure your linear model, it calculated from residual sum of squared errors, and measures the lack of fit the model has for the data. However, it is in the units of the data and can be difficult to interpret at times, therefore it is better to use r-squared which lies between 0 and 1. The closer to 1 the better the model. You can also evaluate the model with correlation, which shows the correlation, which shows the degree in terms of 1 and -1, a close to 1 shows it is strongly positively correlation, a -1 shows it is strongly negatively correlation, and lastly a 0 shows no correlation. Linear regression has a few weaknesses including iteration effects, where there is too much synergy between predictors, causing them to have too much impact as a whole on the model. Along with confounding variables, which correlate with both the target and predictors which can give too high of a correlation in the training data. Also, hidden variables in the model, this can lead to false conclusions and assumptions such as a correlation being a causation in the data. Linear regression is also very likely to underfit data, causing it to fail at capturing some of the data and have a low accuracy. However, linear regression is very simple to implement and has a very high performance on linearly separable data sets. Also, overfitting data is unlikely to happen and can be very easy to fix in linear models using regularization.

We chose the "Air_Pollution.csv" for this assignment. It contains data of the overall air quality of different cities overtime.

It has 32191 observations initially.

```
df <- read.csv("Air_Pollution.csv")
str(df)
```

```
## 'data.frame':    32191 obs. of  10 variables:
##  $ Country.Name          : chr  "Afghanistan" "Albania" "Albania" "Albania" ...
##  $ City                  : chr  "Kabul" "Durres" "Durres" "Elbasan" ...
##  $ Year                  : int  2019 2015 2016 2015 2016 2017 2015 2016 2014 2015 ...
##  $ PM2.5                 : num  119.8 NA 14.3 NA NA ...
##  $ PM10                  : num  NA 17.6 24.6 NA NA ...
##  $ NO2                   : num  NA 26.6 24.8 24 26.3 ...
##  $ PM25.temporal.coverage: num  18 NA NA NA NA NA NA NA NA NA ...
##  $ PM10.temporal.coverage: num  NA NA NA NA NA NA NA NA NA NA ...
##  $ NO2.temporal.coverage : num  NA 84 87.9 97.9 96 ...
##  $ Updated.Year          : int  2022 2022 2022 2022 2022 2022 2022 2022 2022 2022 ...
```

We then fill incomplete rows cells with the medians from their rows

```
df$PM2.5[is.na(df$PM2.5)] <- mean(df$PM2.5, na.rm=TRUE)
df$PM10[is.na(df$PM10)] <- mean(df$PM10, na.rm=TRUE)
df$NO2[is.na(df$NO2)] <- mean(df$NO2, na.rm=TRUE)
df$PM25.temporal.coverage[is.na(df$PM25.temporal.coverage)] <-
  mean(df$PM25.temporal.coverage, na.rm=TRUE)
df$PM10.temporal.coverage[is.na(df$PM10.temporal.coverage)] <-
  mean(df$PM10.temporal.coverage, na.rm=TRUE)
df$NO2.temporal.coverage[is.na(df$NO2.temporal.coverage)] <-
  mean(df$NO2.temporal.coverage, na.rm=TRUE)

str(df)
```

```
## 'data.frame':    32191 obs. of  10 variables:
##  $ Country.Name          : chr  "Afghanistan" "Albania" "Albania" "Albania" ...
##  $ City                  : chr  "Kabul" "Durres" "Durres" "Elbasan" ...
##  $ Year                  : int  2019 2015 2016 2015 2016 2017 2015 2016 2014 2015 ...
##  $ PM2.5                 : num  119.8 22.9 14.3 22.9 22.9 ...
##  $ PM10                  : num  30.5 17.6 24.6 30.5 30.5 ...
##  $ NO2                   : num  20.6 26.6 24.8 24 26.3 ...
##  $ PM25.temporal.coverage: num  18 90.8 90.8 90.8 90.8 ...
##  $ PM10.temporal.coverage: num  90.6 90.6 90.6 90.6 90.6 ...
##  $ NO2.temporal.coverage : num  93.7 84 87.9 97.9 96 ...
##  $ Updated.Year          : int  2022 2022 2022 2022 2022 2022 2022 2022 2022 2022 ...
```

Now we are creating the training and testing sets (80% train, 20% test).

```
i <-sample(1:nrow(df), nrow(df)*0.80, replace=FALSE)
train <- df[i,]
test <- df[-i,]
```

```
head(train)
```

```
##       Country.Name      City Year    PM2.5     PM10      NO2
## 3927         China   Bayi Qu 2019  7.41000 30.53325 20.61934
## 14980        India   Gwalior 2012 22.92032 30.53325 20.61934
## 21976       Poland Bydgoszcz 2019 18.14000 30.34000 22.67000
## 6186         China  Quanzhou 2017 29.25000 30.53325 20.61934
## 21430  New Zealand   Waimate 2012 22.92032 17.90000 20.61934
## 18886        Italy Ostellato 2014 15.83000 30.53325 14.87000
##       PM25.temporal.coverage PM10.temporal.coverage NO2.temporal.coverage
## 3927                91.78082                90.5835              93.69680
## 14980               90.79410                90.5835              16.34615
## 21976               90.79410                90.5835              99.09247
## 6186                76.16438                90.5835              93.69680
## 21430               90.79410                99.0000              93.69680
## 18886               90.79410                90.5835              98.64155
##       Updated.Year
## 3927          2022
## 14980         2022
## 21976         2022
## 6186          2022
## 21430         2016
## 18886         2022
```

```
tail(train)
```

```
##       Country.Name                City Year    PM2.5      PM10   NO2
## 22051       Poland              Elblag 2010 22.68000  26.48000 13.72
## 26203        Spain                Leon 2017 22.92032  18.30000 18.61
## 3692         Chile         Puente Alto 2011 23.00000  59.82000 18.91
## 14855        India Dombivali/Ambernath 2016 22.92032 127.50000 75.50
## 2751        Canada            Cornwall 2015  6.90000  30.53325  9.96
## 28752  Switzerland              Thônex 2012 22.92032  19.80000 21.40
##       PM25.temporal.coverage PM10.temporal.coverage NO2.temporal.coverage
## 22051               88.49300               93.15100              82.03200
## 26203               90.79410               90.58350              98.52169
## 3692                90.79410               90.58350              93.69680
## 14855               90.79410               76.92308              76.92308
## 2751                97.80822               90.58350             100.00000
## 28752               90.79410               90.58350              93.69680
##       Updated.Year
## 22051         2022
## 26203         2022
## 3692          2018
## 14855         2022
## 2751          2018
## 28752         2022
```

```
range(train$PM2.5, na.rm=FALSE)
```

```
## [1]   0.01 191.90
```

```
range(train$PM10, na.rm=FALSE)
```

```
## [1]   1.04 540.00
```

```
mean(train$PM2.5)
```

```
## [1] 22.91425
```

```
mean(train$PM10)
```

```
## [1] 30.51815
```

```
median(train$PM2.5)
```

```
## [1] 22.92032
```

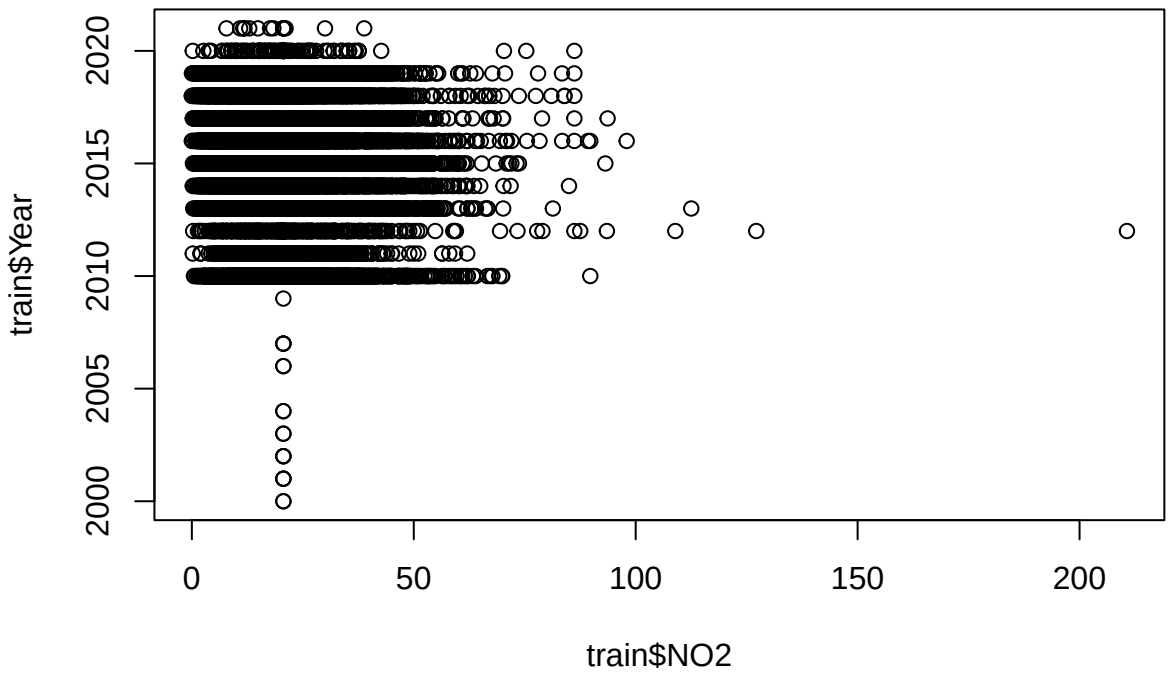
```
median(train$PM10)
```

```
## [1] 30.53325
```

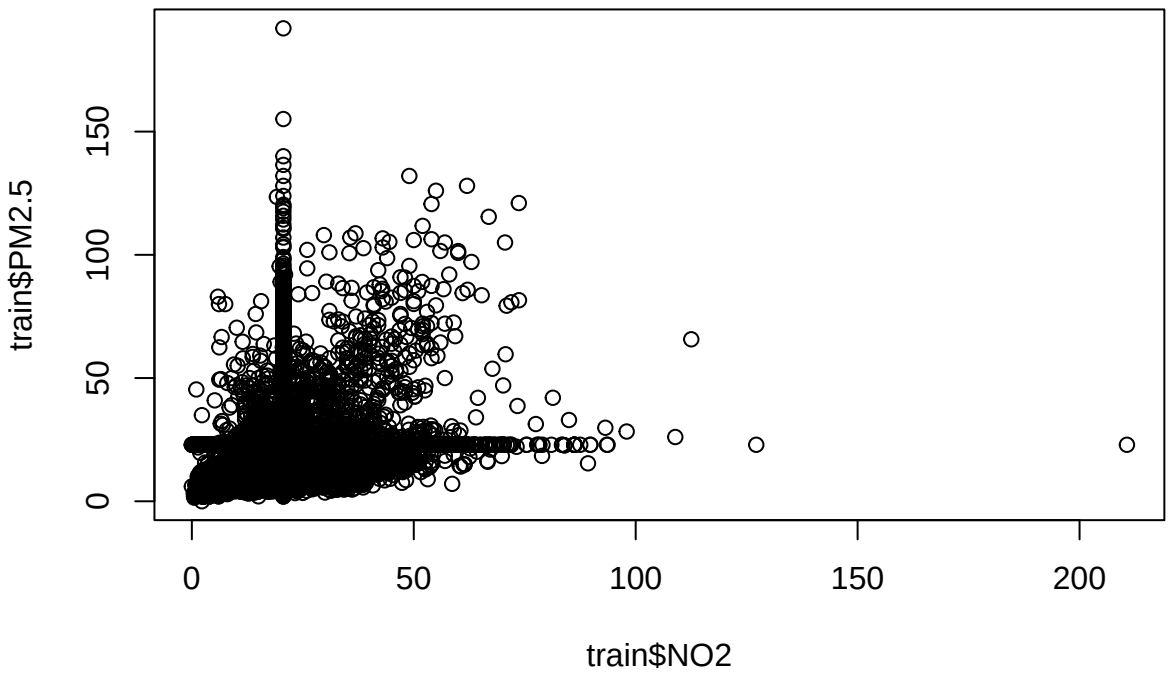
These are our plots

```
hist(train$PM2.5)
```

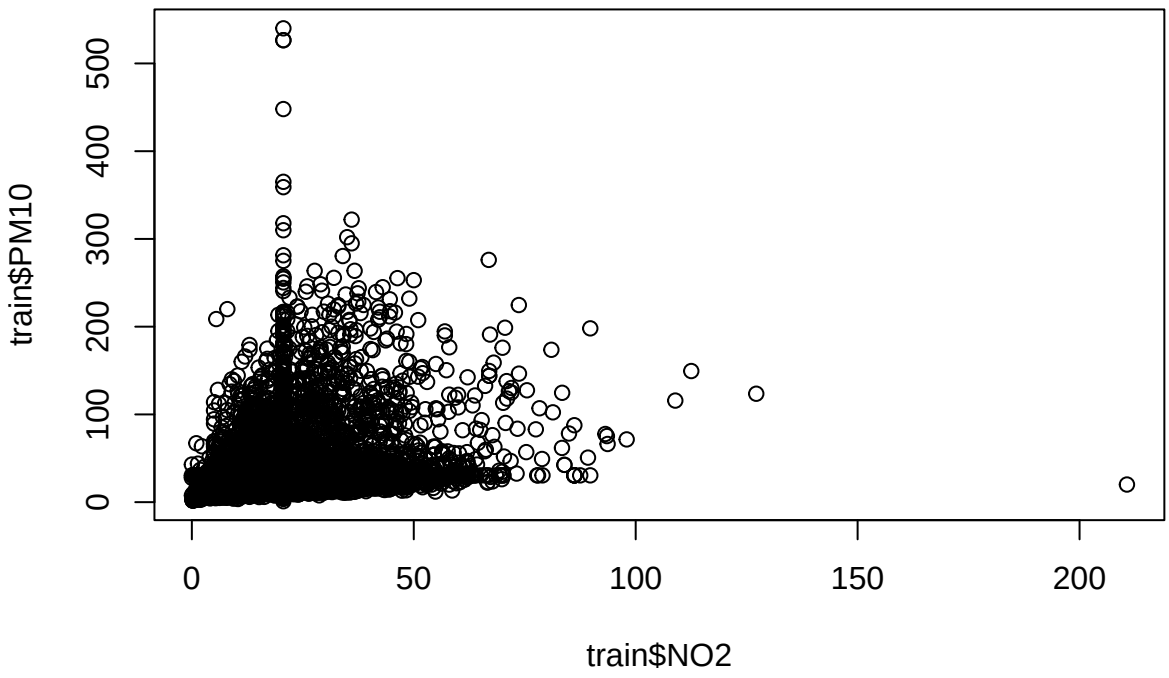

```
plot(train$NO2, train$Year)
```




```
plot(train$NO2, train$PM2.5)
```




```
plot(train$NO2, train$PM10)
```



Plotting the linear regression

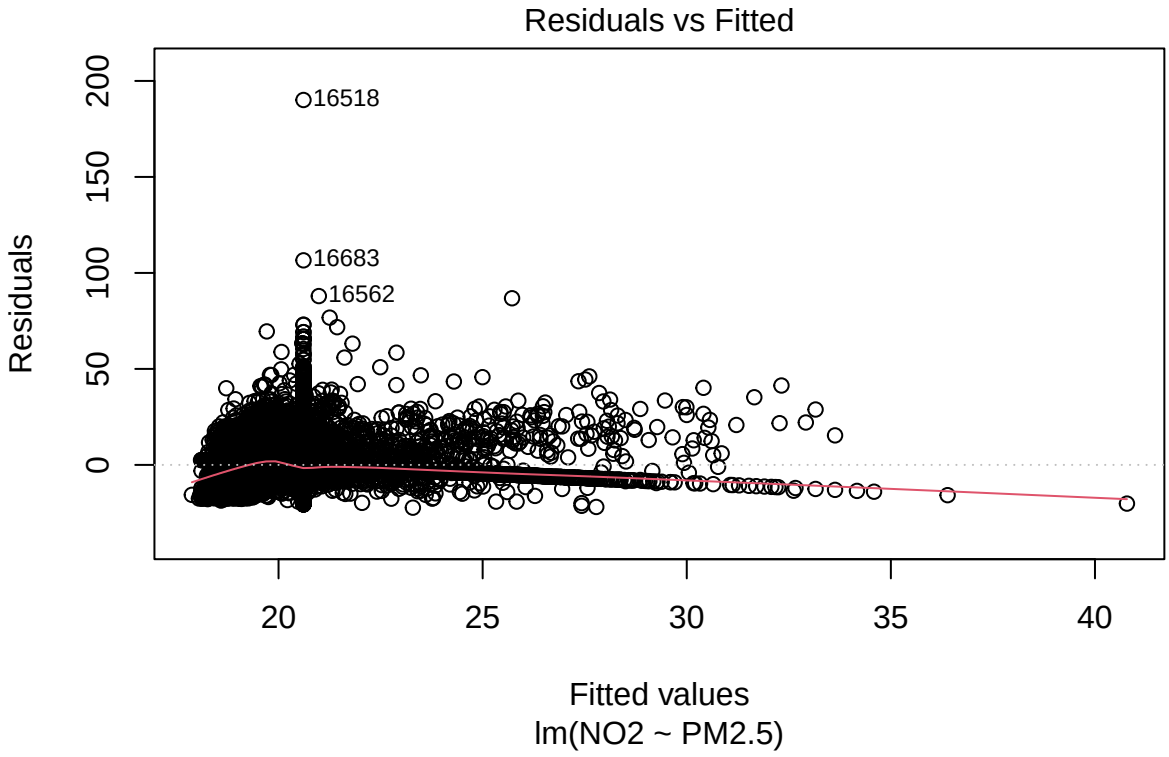
```
lm1 <- lm(NO2~PM2.5, data=train)
summary(lm1)
```

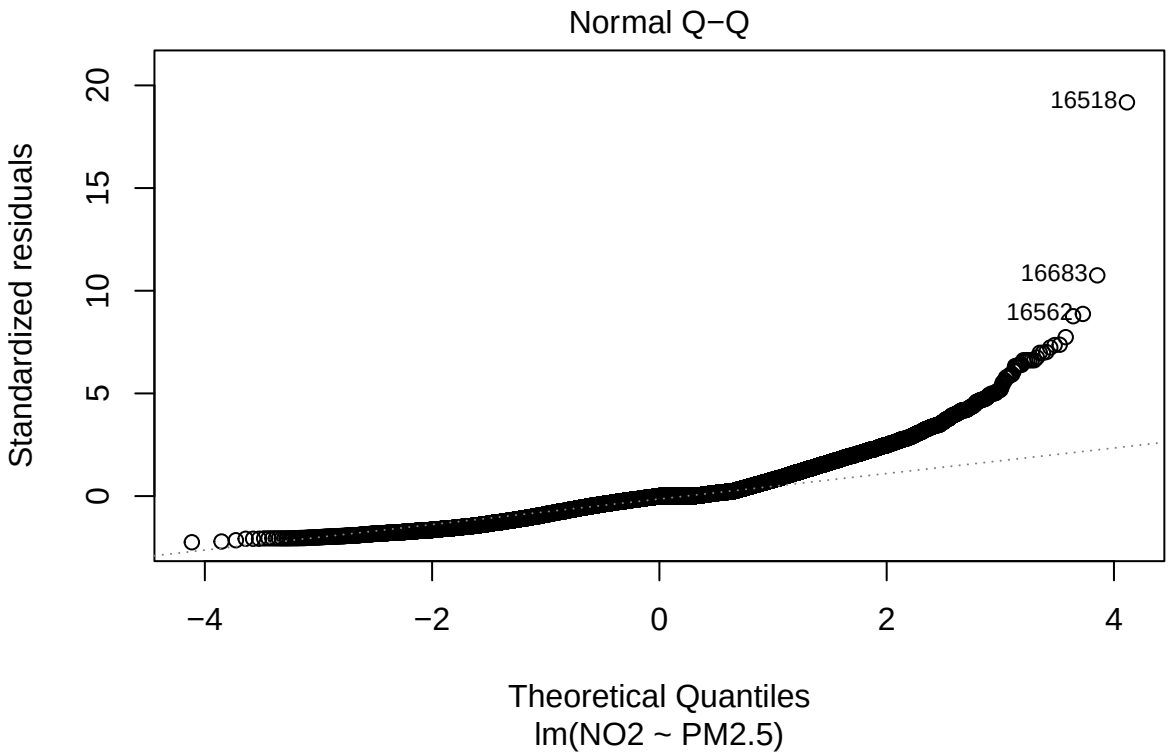
```
##
## Call:
## lm(formula = NO2 ~ PM2.5, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -22.281  -5.581  -0.039   2.731 190.069
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 17.87454    0.13057  136.90   <2e-16 ***
## PM2.5        0.11938    0.00502   23.78   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.913 on 25750 degrees of freedom
## Multiple R-squared:  0.02149,    Adjusted R-squared:  0.02145
## F-statistic: 565.5 on 1 and 25750 DF,  p-value: < 2.2e-16
```

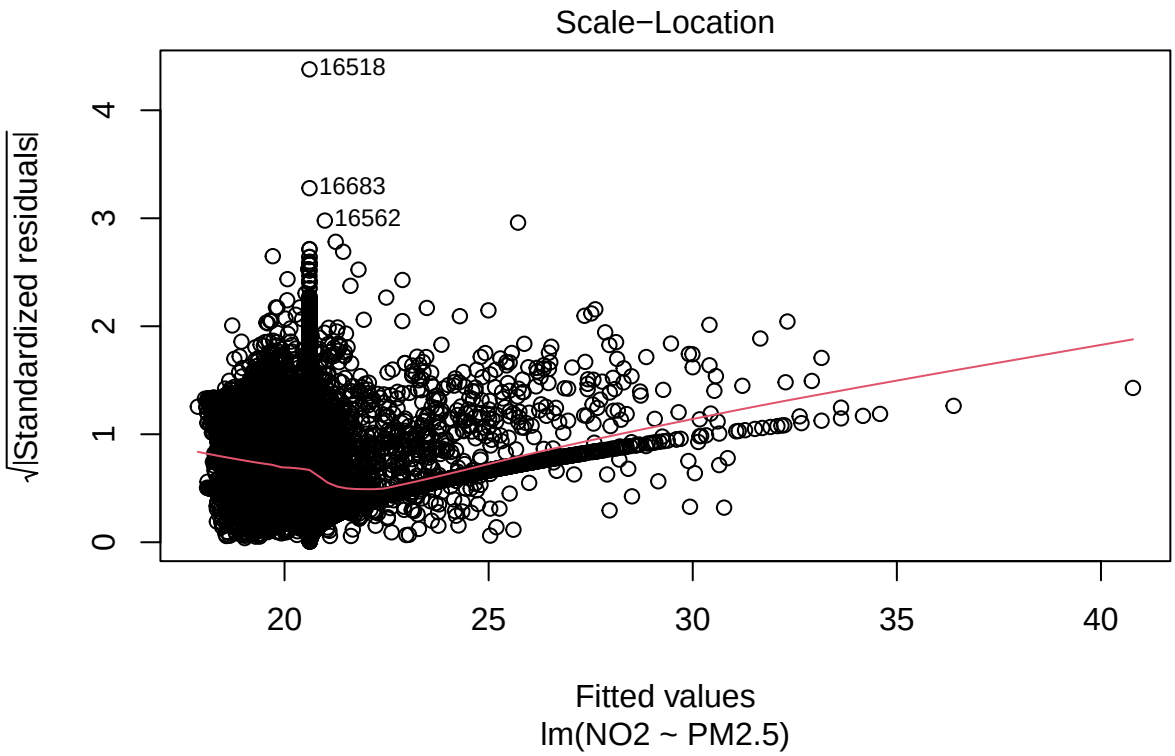
The summary above is used to help determine if our linear model is good or bad, if it accurately predicts our target or not. We can start with the min, max, and median. As you can see it has a very wide range, showing that we may have some outliers or very divided data. The residual standard error is 9.974 on 25650 degrees

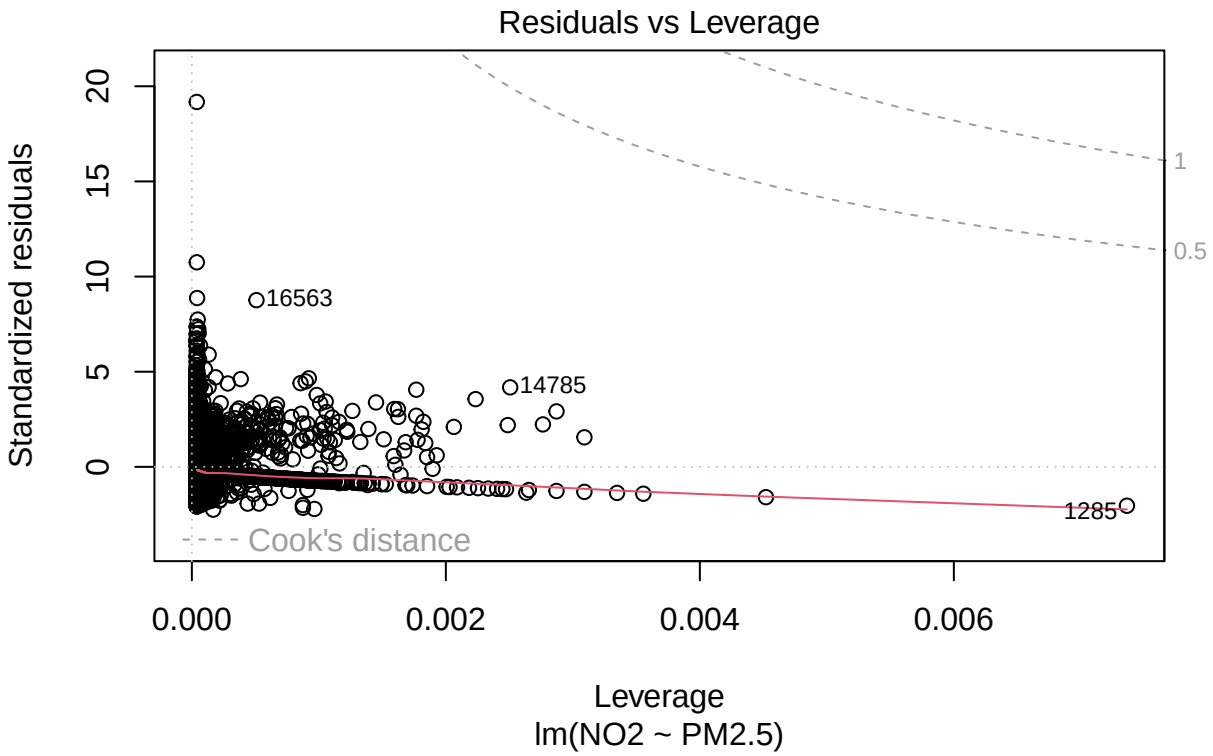
of freedom, this states that roughly the model's predictions are about 10 units off. The multiple r-squared error is 0.02057, this should be as close to 1 as possible. In most cases anything under .4 is a poor model at explaining variance in the data. However our p-value is fairly low and very close to 0, according to this we have a good model that is significant.

```
plot(lm1)
```




Normal Q−Q
Standardized residuals
0
5
10
15
20
16518
16683
16562
−4
−2
0
2
4
Theoretical Quantiles
lm(NO2 ~ PM2.5)


Scale−Location
16518
16683
16562
√|Standardized residuals|
0
1
2
3
4
20
25
30
35
40
Fitted values
lm(NO2 ~ PM2.5)



The residual plots above are summarized below:

Residuals vs Fitted: In this plot it shows whether the linearity holds, indicated by the mean residual value for every fitted value region being close to 0 (the red line being close to the dashed line). As you can see the redline is very close to the dashed line close to 20, but drifts downwards as we get close to 40 and 0. We can also see that we may have outliers at point 16687 and the other two below (we cannot read them because of the way R printed them ontop of eachother). Overall it shows that our model is a good model.

Normal Q-Q: This graph shows the standardized residuals and theoretical quantiles, if they come from the same distribution then the points should form a roughly straight line. Our data set has a very straight line until around (4, 10) where we once again see our outlier points 16687 and the couple below. As you can see it once again steers away from the gray line in a strong curve shape. Although the line is still fairly straight overall but we may have some skewed data in our dataset (possibly from having to replace many missing values with the mean). It can also mean since it is curved at the edge that our dataset contains more extreme values.

Scale Location: Our scale-location graph is very uneven and is basically a black blob. The residuals are supposed to be spread evenly across the range of predictors, which can show the assumption of equal variance. As you can tell ours is definitely not good, this shows that our assumption of equal variance is very incorrect.

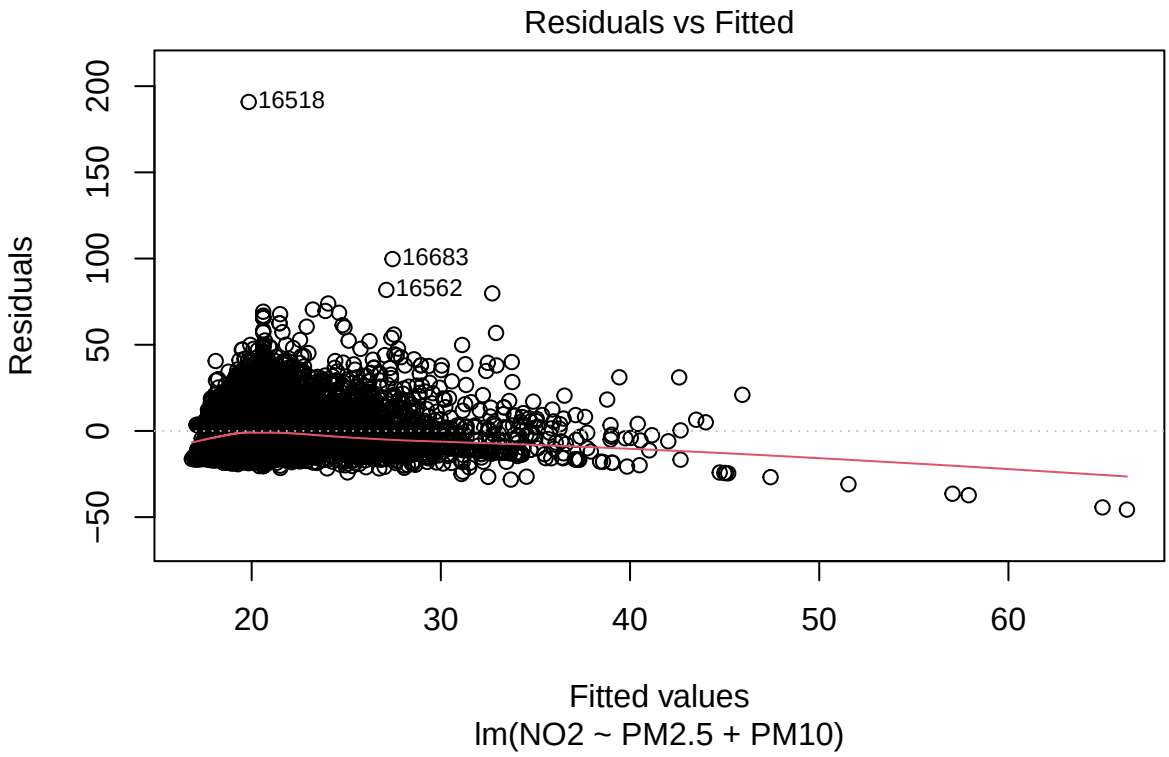
Leverage vs Residuals: In this plot the leverage is the extent to which the coefficients in the regression model would change if a particular observation was removed from the dataset. Thus observations with high leverage have strong influence on the coefficients of the regression model. The standardized residuals refer to the standardized difference between a predicted value and actual value of an observation. You can tell in the plots above that we have no influential points as all the points of the data are within the cook's distance (dotted lines). But we may have outliers at points 16563, 14785, and 1285.

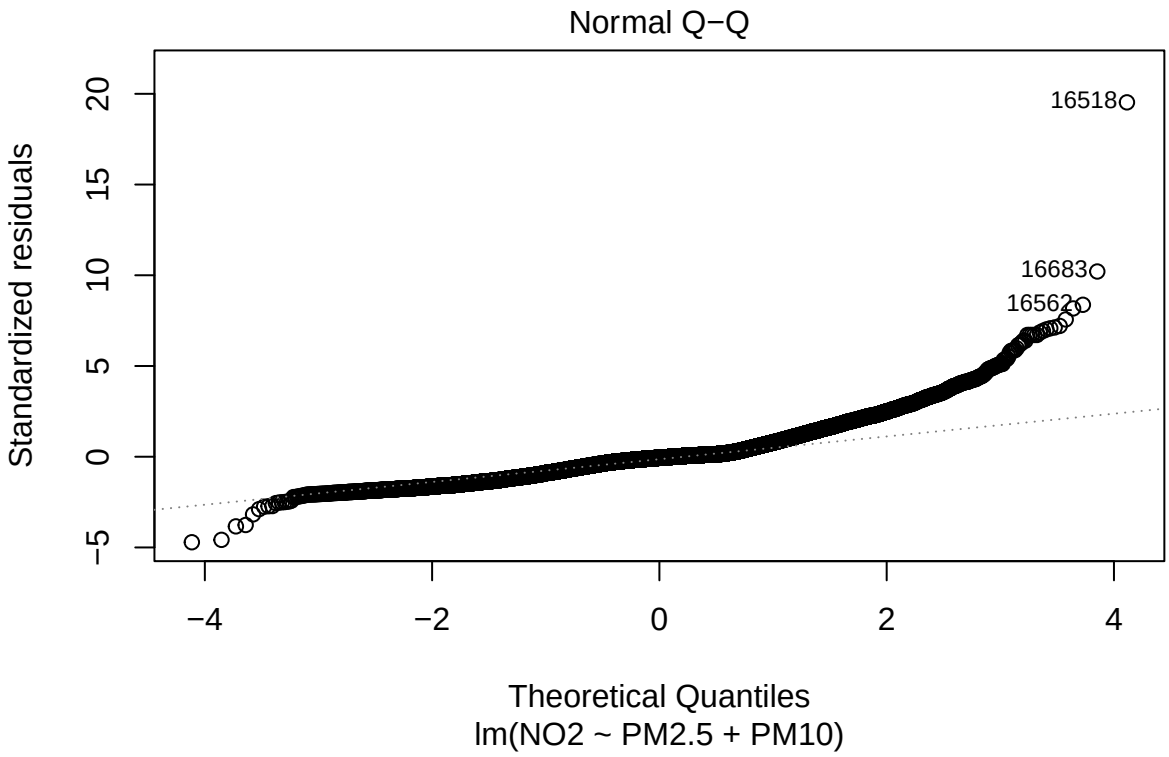
```r
lm2 <- lm(NO2~PM2.5+PM10, data=train)
summary(lm2)
```

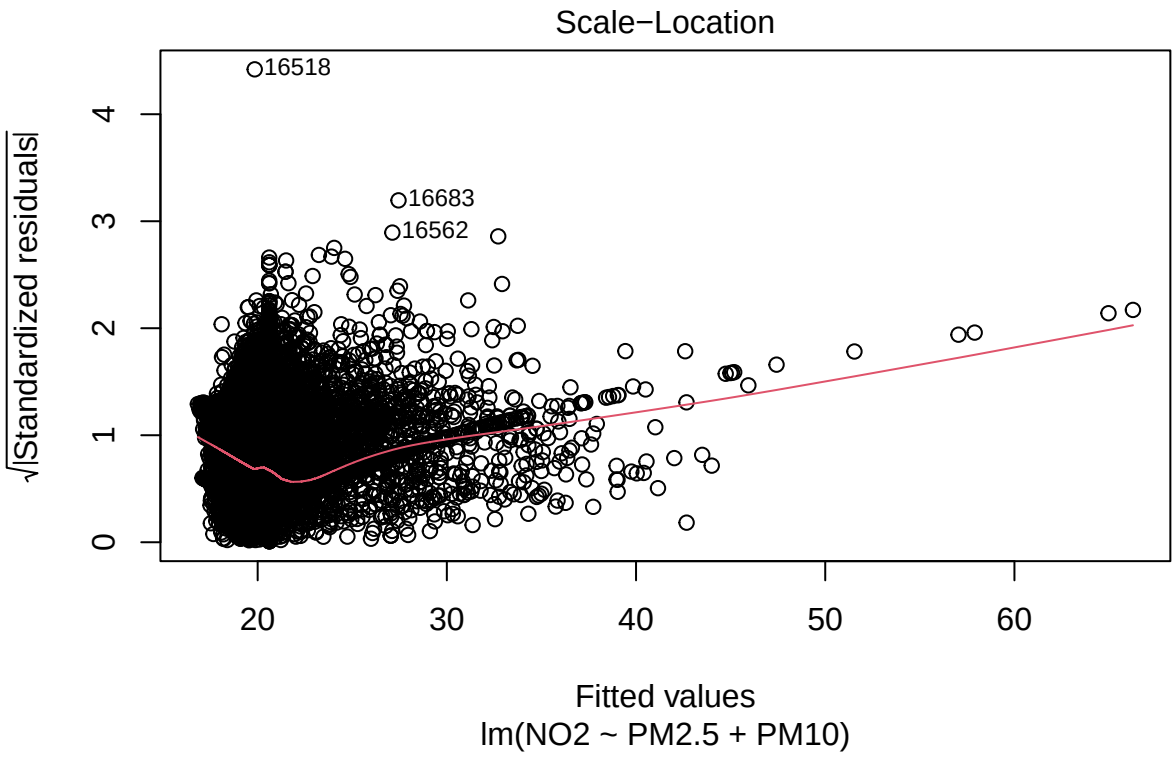
```
##
## Call:
## lm(formula = NO2 ~ PM2.5 + PM10, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -45.646  -5.475  -0.573   2.791 190.836
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 16.563514   0.137331  120.61   <2e-16 ***
## PM2.5        0.078761   0.005166   15.24   <2e-16 ***
## PM10         0.073456   0.002683   27.38   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.772 on 25749 degrees of freedom
## Multiple R-squared:  0.04916,    Adjusted R-squared:  0.04909
## F-statistic: 665.7 on 2 and 25749 DF,  p-value: < 2.2e-16
```

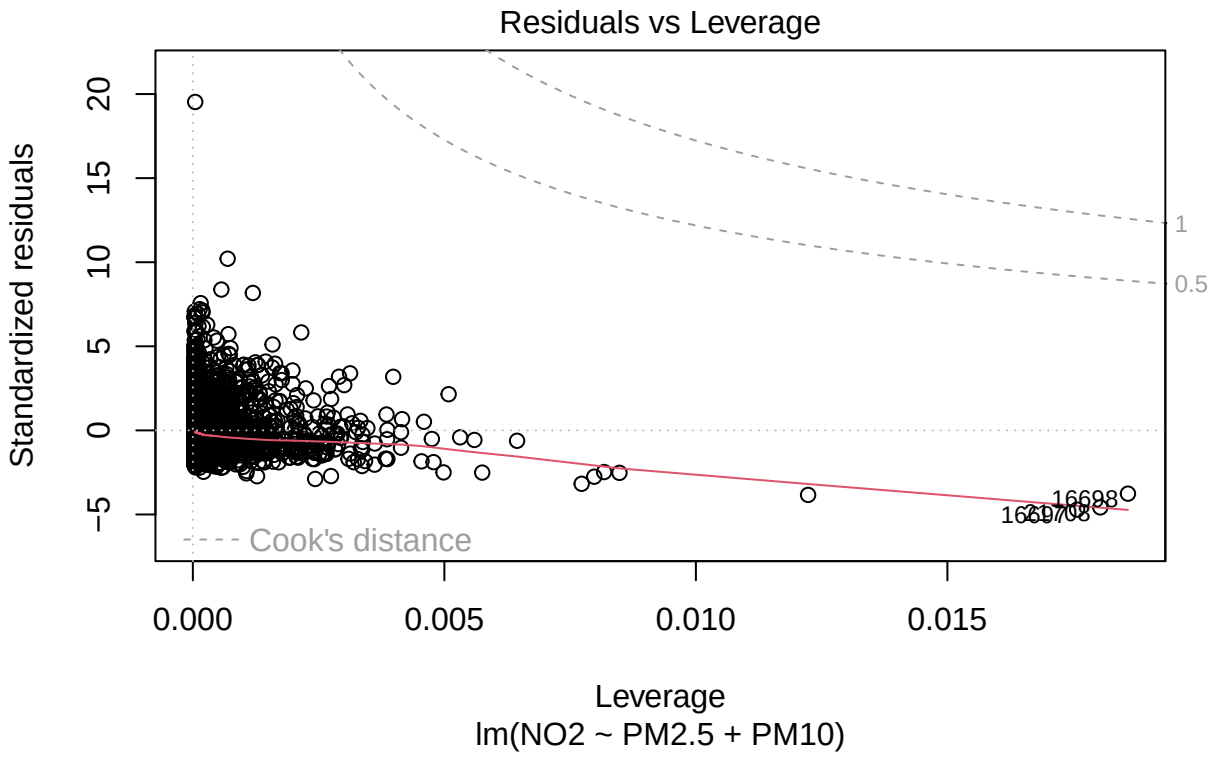
The summary above can be used to compare this model with the previous one above. The residual standard error is 9.82 instead of 9.974 on about the same degrees of freedom, this states that roughly the model's predictions are still about 10 units off but slightly less than the previous model. The multiple r-squared error is 0.05059 instead of 0.02057, this should be as close to 1 as possible so it is very much an improvement compared to the model above. However, it is still well under .4 and therefore is still a poor model at explaining variance in the data. The p-value is for both models is the same that is significant.

```r
plot(lm2)
```


Residuals vs Fitted
Residuals
Fitted values
lm(NO2 ~ PM2.5 + PM10)
16518
16683
16562
200
150
100
50
0
−50
20
30
40
50
60


Normal Q−Q
Standardized residuals
20
15
10
5
0
−5
16518
16683
16562
−4
−2
0
2
4
Theoretical Quantiles
lm(NO2 ~ PM2.5 + PM10)


Scale−Location
16518
16683
16562
√|Standardized residuals|
0
1
2
3
4
20
30
40
50
60
Fitted values
lm(NO2 ~ PM2.5 + PM10)



As you can see above these plots are even more extreme in every graph in the wrong ways. Residuals vs fitted is less of a straight line and curves dramatically at the end. Normal Q-Q is very much curved at the edges telling us our dataset is skewed. The scale location is about the same of a condensed black blob as above. Finally in Residuals vs Leverage we have even more extreme outliers at both ends, but still no influential values (barely). Overall with these results this model is worse than the original above.

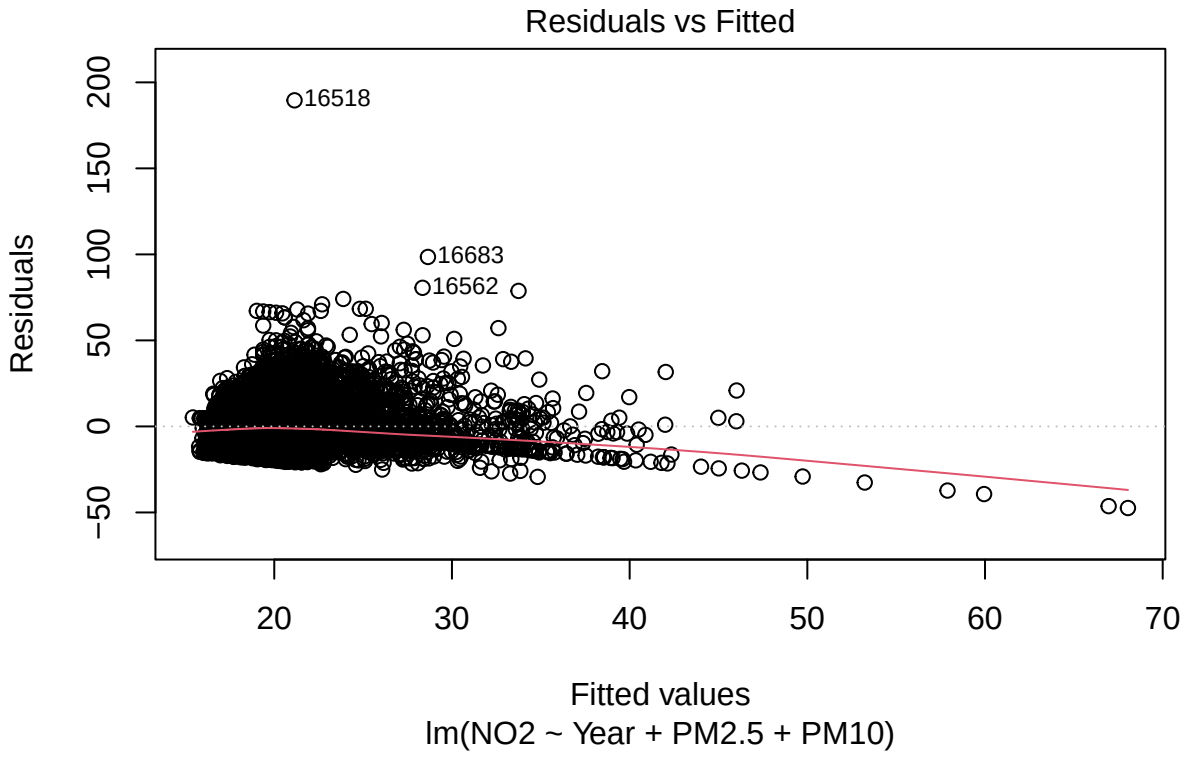
```
lm3 <- lm(NO2~Year+PM2.5+PM10, data=train)
summary(lm3)
```

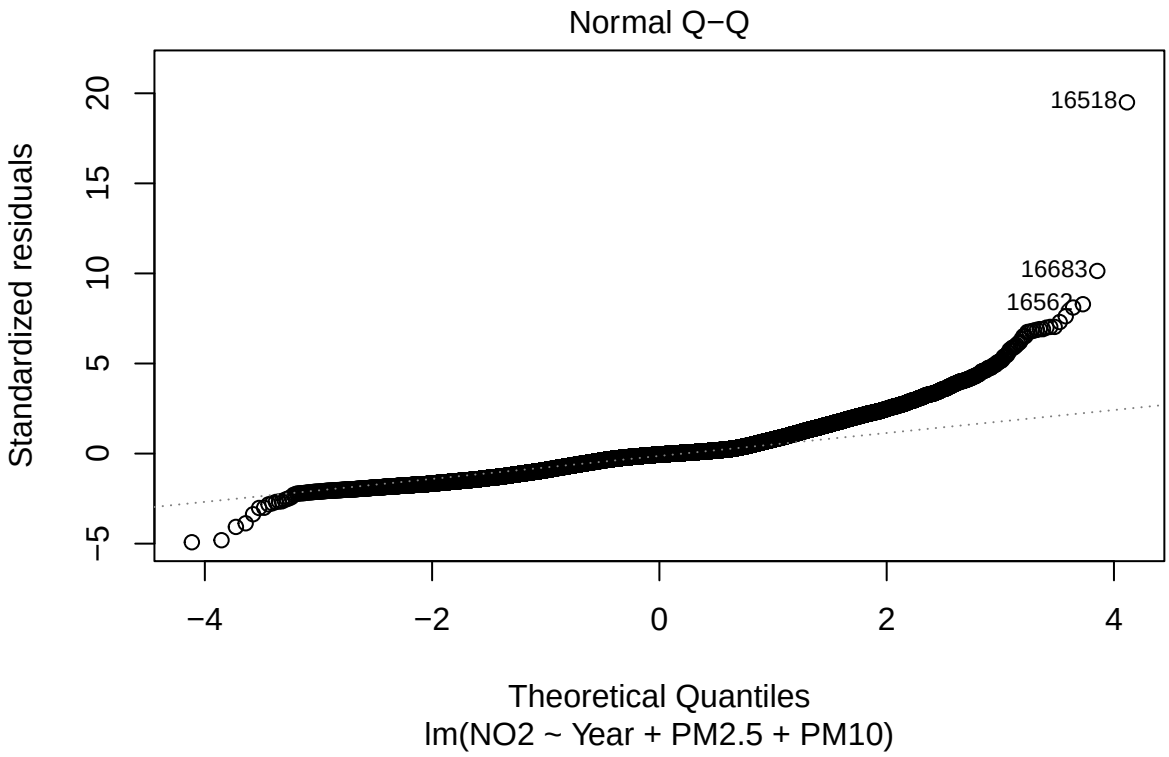
```
##
## Call:
## lm(formula = NO2 ~ Year + PM2.5 + PM10, data = train)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -47.426  -5.458  -0.536   2.899 189.541
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 742.164397  44.492973   16.68   <2e-16 ***
## Year         -0.360040   0.022077  -16.31   <2e-16 ***
## PM2.5         0.083661   0.005149   16.25   <2e-16 ***
## PM10          0.072570   0.002670   27.18   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 9.722 on 25748 degrees of freedom
```

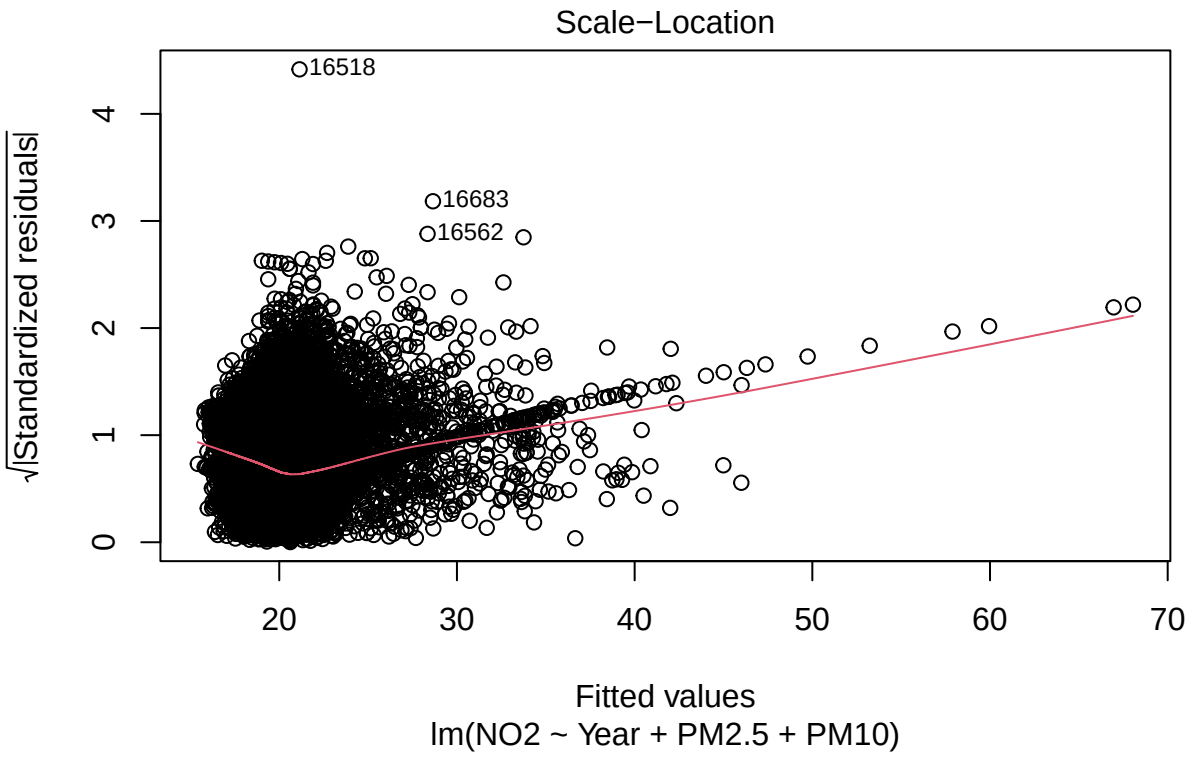
```
## Multiple R-squared:  0.05888,    Adjusted R-squared:  0.05877
## F-statistic:   537 on 3 and 25748 DF,  p-value: < 2.2e-16
```

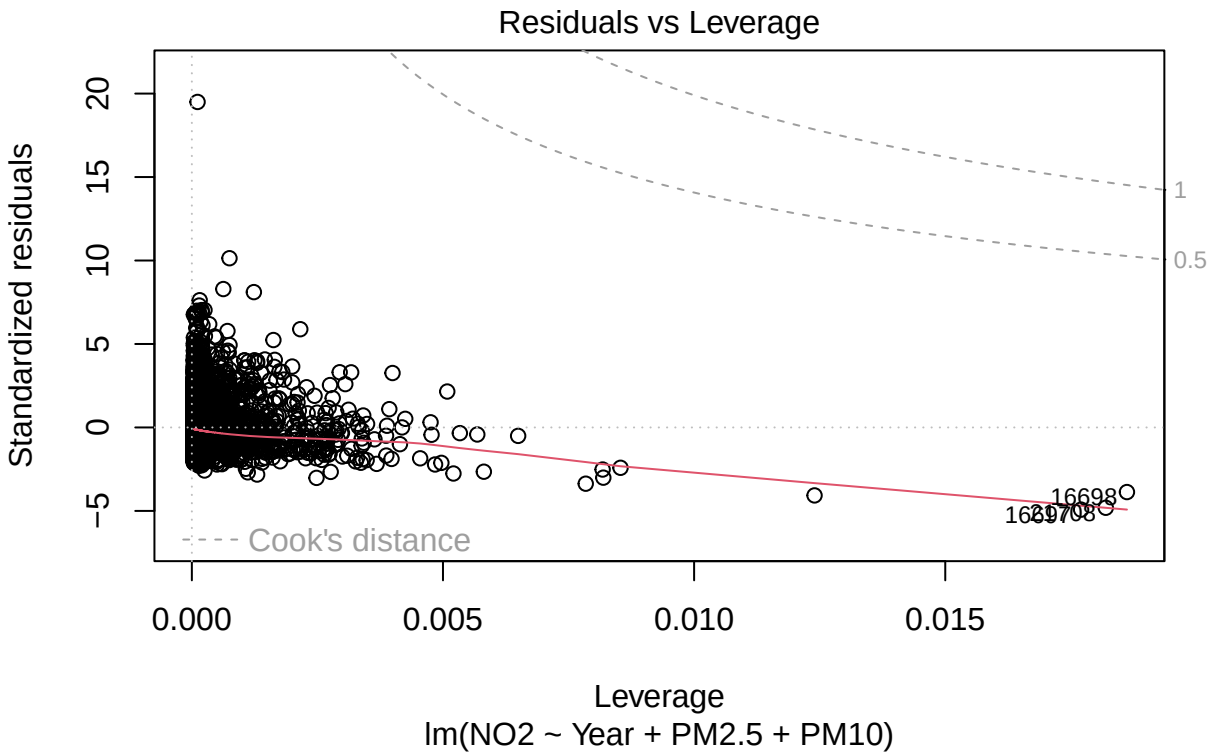
The summary above can be used to compare this model with the previous models above. The residual standard error is 9.737 instead of 9.974 and 9.82 on about the same degrees of freedom, this states that roughly the model's predictions are still about 10 units off being slightly less than the previous models. The multiple r-squared error is 0.06038, which is much better than 0.02057 from the first model, and slightly better than 0.05059 from the second model, this should be as close to 1 as possible so it is not an improvement compared to the models above. Still being well under .4 and therefore is still a poor model at explaining variance in the data. The p-value is for all models is the same that is significant.

```
plot(lm3)
```




Normal Q−Q
Standardized residuals
20
15
10
5
0
−5
16518
16683
16562
−4
−2
0
2
4
Theoretical Quantiles
lm(NO2 ~ Year + PM2.5 + PM10)


Scale−Location
16518
16683
16562
√|Standardized residuals|
0
1
2
3
4
20
30
40
50
60
70
Fitted values
lm(NO2 ~ Year + PM2.5 + PM10)



As you can see above these plots vary more than the 2nd model. Residuals vs fitted is less of a straight line and curves more at the end than the 1st model, but is more straight than the 2nd model. Normal Q-Q is very much curved at the ending edge telling us our dataset is skewed, but not as bad as the 2nd model. The scale location is about the same of a condensed black blob as above. Finally in Residuals vs Leverage we have even more extreme outliers at both ends and has one influential value at 16687. Overall with these results this model is worse than the models above.

Overall, all of the models do a very poor job at explaining the variation of the data. Which makes sense as there are many outside factors not in the dataset that are causing/changing the air pollution/quality. But the models do manage to have a low p-value and thus does have a significant meaning and isn't useless. The models aren't very good in general but could be used for something specific in predicting the NO2 of areas given PM2.5 or PM10. It is a close call but we think the 3rd model is the best out of the three as it has a higher multiple r-squared error and a lower residual standard error.

Here is the testing of the three models:

```
pred1 <- predict(lm1, newdata=test)
pred2 <- predict(lm2, newdata=test)
pred3 <- predict(lm3, newdata=test)
```

Here are the test results of the three models: Linear model 1:

```
correlation1 <- cor(pred1, test$NO2)
print(paste("correlation: ", correlation1))
```

```
## [1] "correlation:  0.120559837288455"
```

```
mse <- mean((pred1 - test$NO2)^2)
print(paste("mse: ", mse))
```

```
## [1] "mse:  104.477283787824"
```

```
rmse <- sqrt(mse)
print(paste("rmse: ", rmse))
```

```
## [1] "rmse:  10.2214130034856"
```

Linear model 2:

```
correlation2 <- cor(pred2, test$NO2)
print(paste("correlation: ", correlation2))
```

```
## [1] "correlation:  0.237807166220592"
```

```
mse <- mean((pred2 - test$NO2)^2)
print(paste("mse: ", mse))
```

```
## [1] "mse:  100.044651057672"
```

```
rmse <- sqrt(mse)
print(paste("rmse: ", rmse))
```

```
## [1] "rmse:  10.0022323037246"
```

Linear model 3:

```
correlation3 <- cor(pred3, test$NO2)
print(paste("correlation: ", correlation1))
```

```
## [1] "correlation:  0.120559837288455"
```

```
mse <- mean((pred3 - test$NO2)^2)
print(paste("mse: ", mse))
```

```
## [1] "mse:  99.0308872270996"
```

```
rmse <- sqrt(mse)
print(paste("rmse: ", rmse))
```

```
## [1] "rmse:  9.95142639158325"
```

As expected the correlation for all models is quite low, the highest is model 2, but is still very unacceptable. The RMSE tells us that our model is off around 10 units as expected from above. We believe these results happened because there was a lot of missing data in the dataset that we had to fill in with medians, as well as the unpredictable differences between these data values. They are primarily caused by outside forces, not necessarily by each other therefore there is likely little accurate correlation between them and using them to predict another is ignoring outside factors. There was also a very high amount of variance in the data that very much affected the ability for the linear model to fit the dataset well.