Regression

Code

Dataset: Gas Turbine CO and NOx Emission Data Set
(https://archive.ics.uci.edu/ml/datasets/Gas+Turbine+CO+and+NOx+Emission+Data+Set)

The data for this notebook is from the UCI ML Repository. It describes gas turbine carbon monoxide and nitrous oxide emissions from a gas turbine located in North Western Turkey. It contains over 30,000 rows.

The data originally was separated into multiple csv's I wrote a python script to combine all of the data into a single csv, which is the csv included with this project, read in below.

Libraries

Hide

```
library(ggplot2)
library(corrplot)
```

```
corrplot 0.92 loaded
```

Hide

```
library(Metrics)
```

Hide

```
df <- read.csv("pp_gas_emission_2011_2015.csv")
```

Hide

```
head(df)
```

	AT <dbl>	AP <dbl>	AH <dbl>	AFDP <dbl>	GTEP <dbl>	TIT <dbl>	TAT <dbl>	TEY <dbl>	CDP <dbl>
1	4.5878	1018.7	83.675	3.5758	23.979	1086.2	549.83	134.67	11.898
2	4.2932	1018.3	84.235	3.5709	23.951	1086.1	550.05	134.67	11.892
3	3.9045	1018.4	84.858	3.5828	23.990	1086.5	550.19	135.10	12.042
4	3.7436	1018.3	85.434	3.5808	23.911	1086.5	550.17	135.03	11.990
5	3.7516	1017.8	85.182	3.5781	23.917	1085.9	550.00	134.67	11.910
6	3.8858	1017.7	83.946	3.5824	23.903	1086.0	549.98	134.67	11.868

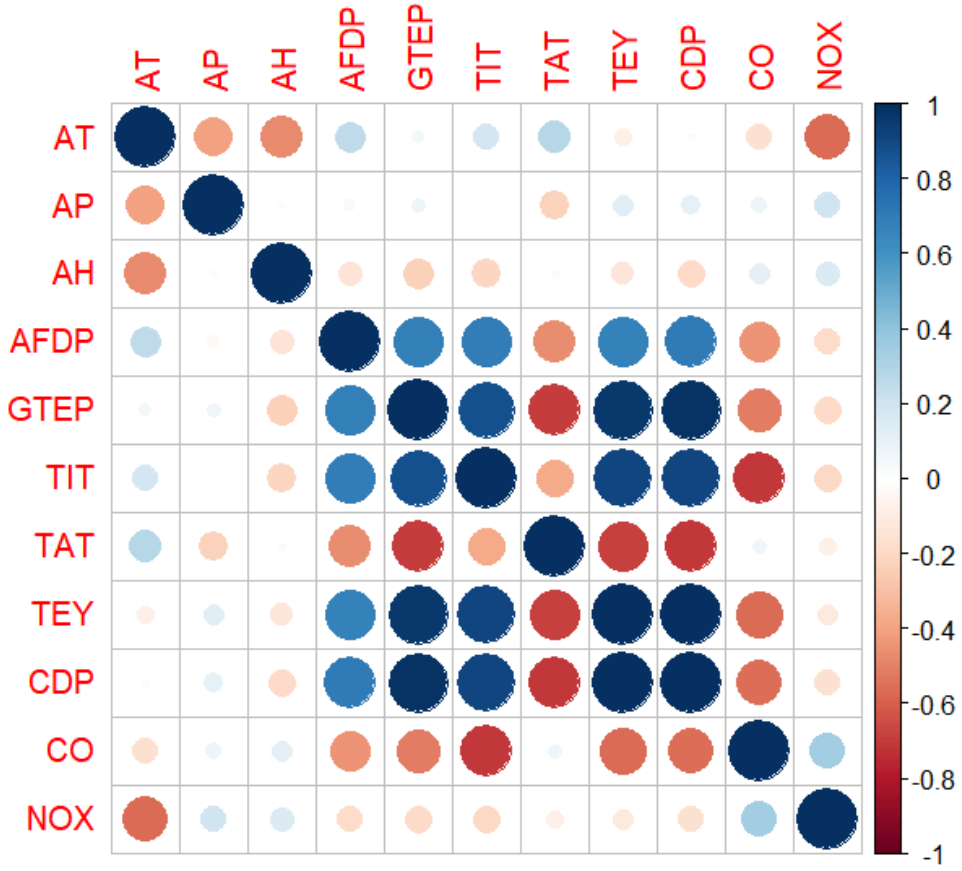
6 rows | 1-10 of 11 columns

All of the columns are represented by symbolic names. To make findings more clear, below is a dictionary of terminology based on symbolic representation:

- AT = Ambient Temperature
- AP = Ambient Pressure
- AH = Ambient Humidity
- AFDP = Air Filter Difference Pressure
- GTEP = Gas Turbine Exhaust Pressure
- TIT = Turbine Inlet Temperature
- TAT = Turbine After Temperature
- CDP = Compressor Discharge Pressure
- TEY = Turbine Energy Yield
- CO = Carbon Monoxide
- NOX = Nitrous Oxides

Hide

```
res <- cor(df)
corrplot(round(res,2))
```



We will choose for the target to be Carbon Monoxide. It appears several variables show negative correlation with Carbon Monoxide emissions. For the first regression analysis, simple linear regression will be used between one predictor and CO. For the next multiple linear regression will be used and several predictors will be used against CO. Finally, a one-versus-all approach against CO will be used.

Train/Test Split

Hide

```
set.seed(1)

# 80/20 split train/test
ind <- sample(1:nrow(df),nrow(df)*0.8,replace=FALSE)
train <- s[ind,]
test <- s[-ind,]
nrow(train)
```

```
[1] 29386
```

Hide

```
nrow(test)
```

```
[1] 7347
```

We're going to use a base model to predict CO emissions.

Hide

```
co.corr <- res[,10]
co.corr
```

```
         AT          AP          AH        AFDP        GTEP         TIT         TAT         TEY
CDP          CO         NOX
-0.17432578  0.06705003  0.10658597 -0.44842461 -0.51890950 -0.70627505  0.05835341 -0.56981256
-0.55102670  1.00000000  0.34060569
```

It appears Turbine Inlet Temperature (TIT) might be a good predictor for carbon monoxide emissions. Let's see.

Hide

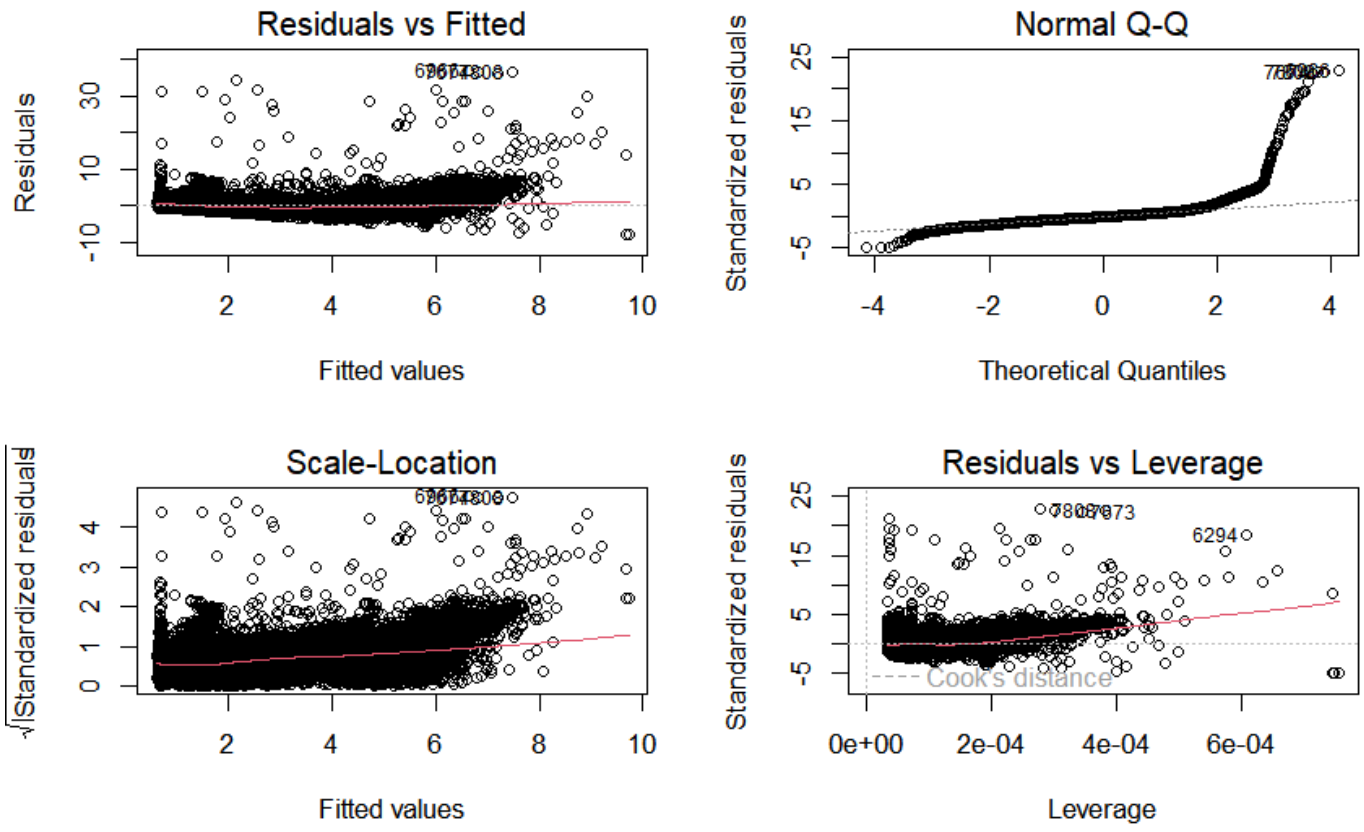
```
mod <- lm(CO ~ TIT, train)
summary(mod)
```

```
Call:
lm(formula = CO ~ TIT, data = train)

Residuals:
   Min     1Q Median     3Q    Max
-8.021 -0.737 -0.153  0.501 36.989

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.010e+02  5.844e-01   172.7   <2e-16 ***
TIT         -9.116e-02  5.403e-04  -168.7   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.621 on 29384 degrees of freedom
Multiple R-squared:  0.492, Adjusted R-squared:  0.492
F-statistic: 2.846e+04 on 1 and 29384 DF,  p-value: < 2.2e-16
```

Though it appears there is some relationship between Turbine Inlet Temperature and Carbon Monoxide, due the p-value being below 0.05, the R^2 and adjusted R^2 are not high for the training data.

Hide

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



A residual is a measure of how far a predicted value is from an actual value on a best-fit line in regression. Part of a regression problem is an assumption that the error in and of itself is normally distributed and independent of the deterministic part of the equation. As part of this assumption then, we should see in the residuals, the stuff left after we train a model, that it follows a normal distribution and is symmetric about the regression line.

In most of the plots above, we see there is a density about the line as the x-axis continues in the growing direction. However, for example in the leverage plot, as the x values grow the density decreases and less points are symmetric about it. This is ultimately used to validate the correctness of the model and we can see that given a different plot such as the scale vs location, there are plenty of points that lie far from the regression line, implying this model's poor performance.

Let's consider making predictions on the test data for the given model.

Hide

```
evaluate <- function(pred, actual, rnd=2) {
  print(paste("RMSE=",round(rmse(pred,actual),rnd)))
  print(paste("MAE=",round(mae(pred,actual),rnd)))
  print(paste("R^2=",round(cor(actual,pred)^2, rnd)))
}
```

Hide

```
new_data <- data.frame(TIT = test$TIT)
co_pred <- predict(mod, new_data)
evaluate(co_pred, test$CO, rnd=3)
```

```
[1] "RMSE= 1.521"
[1] "MAE= 0.895"
[1] "R^2= 0.527"
```

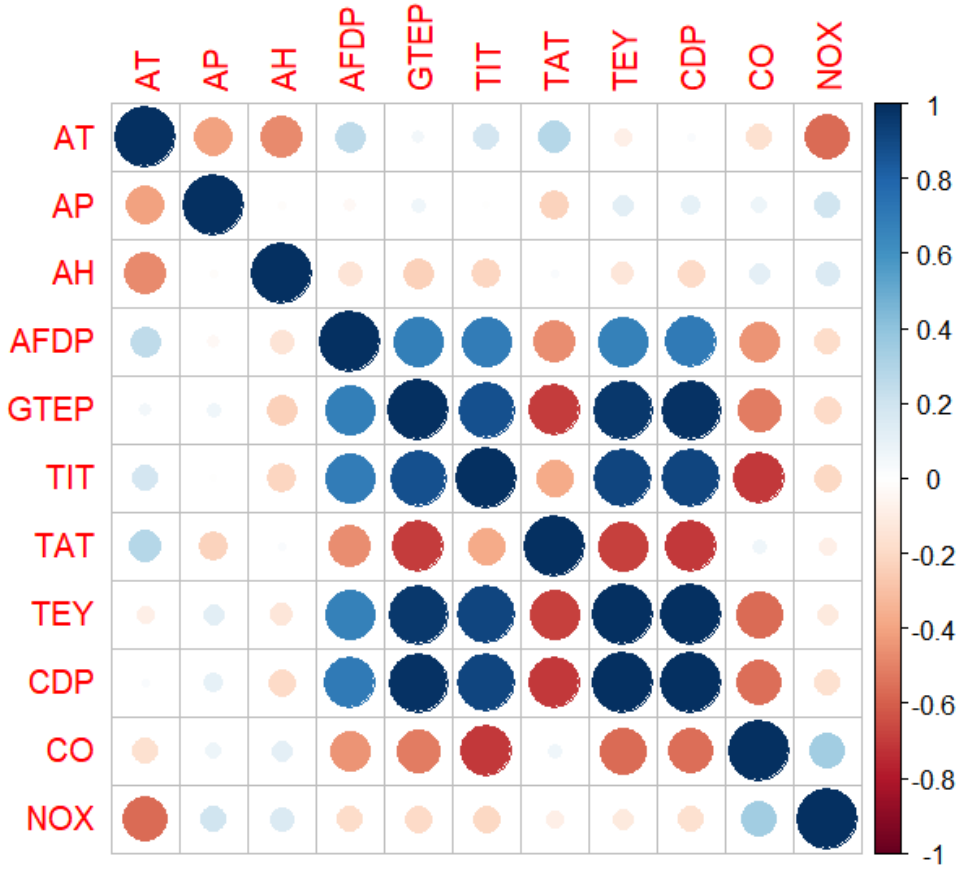
Overall, the R^2 was slightly better than what was shown in the summary for the training of the model than for the test. Turbine Inlet Temperature while its related is not that great at predicting Carbon Monoxide emissions.

Multiple Linear Regression

If we look back at the correlation plot, we can see more attributes that could be used for predicting carbon monoxide emissions.

Hide

```
corrplot(round(res,2))
```



Other attributes include AFDP, GTEP, TEY, CDP. With the inclusion of TIT, all of these features appear to be negatively correlated with carbon monoxide emissions. The next model will include all of these.

Hide

```
mod <- lm(CO ~ TIT + AFDP + GTEP + TEY + CDP, train)
summary(mod)
```

```
Call:
lm(formula = CO ~ TIT + AFDP + GTEP + TEY + CDP, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-11.516  -0.707  -0.054   0.530  35.003

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept) 145.751301   1.226032  118.881  < 2e-16 ***
TIT          -0.146311   0.001294 -113.069  < 2e-16 ***
AFDP         -0.083732   0.017521   -4.779 1.77e-06 ***
GTEP          0.017149   0.010428    1.644      0.1
TEY          -0.079554   0.004139  -19.220  < 2e-16 ***
CDP           2.102488   0.076805   27.374  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.534 on 29380 degrees of freedom
Multiple R-squared:  0.5456,    Adjusted R-squared:  0.5455
F-statistic:  7055 on 5 and 29380 DF,  p-value: < 2.2e-16
```

It appears all of which except for GTEP reject the null hypothesis; pointing at some kind of relationship between the predictor and the target. Dropping GTEP to see how it effects the importance of the other features.

Hide

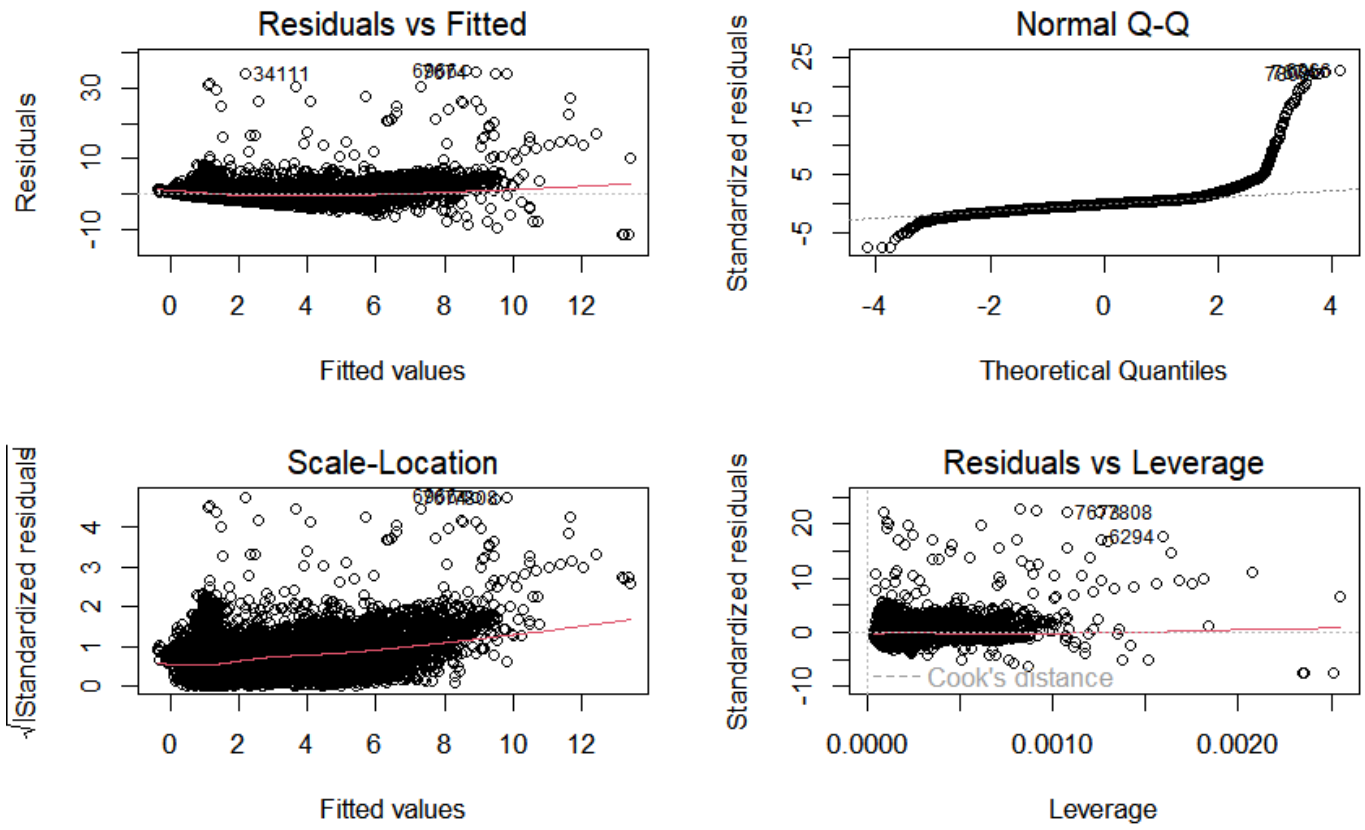
```
mod <- lm(CO ~ TIT + AFDP + TEY + CDP, train)
summary(mod)
```

```
Call:
lm(formula = CO ~ TIT + AFDP + TEY + CDP, data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-11.546  -0.707  -0.056   0.529  34.985

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept) 145.639025   1.224165  118.970  < 2e-16 ***
TIT          -0.146595   0.001282 -114.311  < 2e-16 ***
AFDP         -0.085559   0.017486   -4.893 9.99e-07 ***
TEY          -0.080198   0.004121  -19.462  < 2e-16 ***
CDP           2.181372   0.059984   36.366  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.534 on 29381 degrees of freedom
Multiple R-squared:  0.5456,    Adjusted R-squared:  0.5455
F-statistic:  8818 on 4 and 29381 DF,  p-value: < 2.2e-16
```

It appears everything is significant now and there is some relationship between the predictors and the target. R^2 for the training portion is a tad higher than it was last time, maybe this will imply a higher R^2 once the model is evaluated. Let’s see.

Hide

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



Hide

```
new_data <- data.frame(TIT = test$TIT, AFDP = test$AFDP, TEY = test$TEY, CDP = test$CDP)
co_pred <- predict(mod, new_data)
evaluate(co_pred, test$CO, rnd=3)
```

```
[1] "RMSE= 1.437"
[1] "MAE= 0.86"
[1] "R^2= 0.578"
```

The R^2 slightly increased and the error has decreased from what they were last time using simple linear regression.

Adjusted Multiple Linear Regression

It appears between simple and multiple linear regression, the addition of new features improves the model metrics overall. On this note, I will make a one-versus-all model where all predictors are used to predict carbon monoxide emissions and evaluate from there to see if these scores improve.

Hide

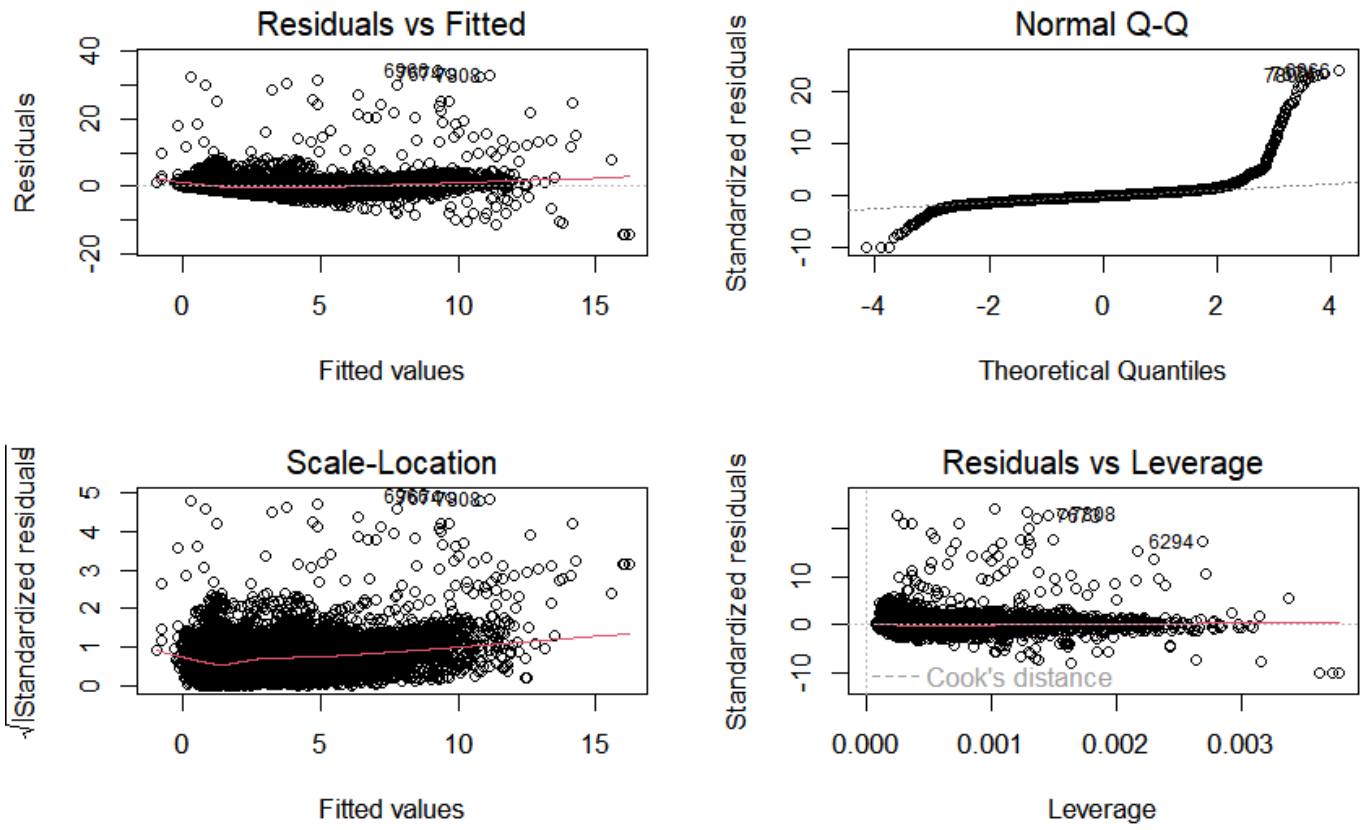
```
mod <- lm(CO ~ ., train)
summary(mod)
```

```
Call:
lm(formula = CO ~ ., data = train)

Residuals:
    Min      1Q  Median      3Q     Max
-14.321  -0.639  -0.089   0.502  34.367

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.334e+02  2.361e+00  56.524  < 2e-16 ***
AT           6.045e-02  3.692e-03  16.372  < 2e-16 ***
AP           1.119e-02  1.609e-03   6.956 3.57e-12 ***
AH           5.872e-03  7.554e-04   7.774 7.87e-15 ***
AFDP        -1.716e-01  1.707e-02 -10.055  < 2e-16 ***
GTEP         9.937e-02  1.075e-02   9.245  < 2e-16 ***
TIT         -1.582e-01  3.279e-03 -48.266  < 2e-16 ***
TAT          1.118e-02  4.203e-03   2.661   0.0078 **
TEY         -6.842e-02  8.745e-03  -7.824 5.28e-15 ***
CDP          2.018e+00  1.206e-01  16.724  < 2e-16 ***
NOX          6.169e-02  1.027e-03  60.058  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.429 on 29375 degrees of freedom
Multiple R-squared:  0.6056,    Adjusted R-squared:  0.6055
F-statistic:  4511 on 10 and 29375 DF,  p-value: < 2.2e-16
```

Per the p-values, it appears all of the features are related to carbon monoxide emissions. In addition the R^2 increased and error decreased relative to the previous model permutation. Let’s see if the same is true for when the model is evaluated against the test data.

Hide

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



Hide

```
new_data <- subset(test, select= -c(CO))
co_pred <- predict(mod, new_data)
evaluate(co_pred, test$CO, rnd=3)
```

```
[1] "RMSE= 1.33"
[1] "MAE= 0.781"
[1] "R^2= 0.639"
```

Once again, the R^2 increased and the error decreased overall.

Conclusion

In this notebook, I explored the affect of attributes onto carbon monoxide emissions in a gas turbine located in Turkey over the years from 2011 to 2015. Starting by combining the data from multiple sets over the years, to training a variety of regression models and evaluating them against unseen data, in general, it appears that all of the attributes in the set correlate to carbon monoxide admissions.

The more variables added into the regression equation, the higher the R^2 and lower the error became. This is both good and bad. The good side of this is that these predictors could be used at some level to predict future admissions, the bad part is that adding more variables can cause the model to overfit to the data. Balance is needed between attributes that are highly correlated and those that aren't in order to assess how well previous information can predict future information.

Overall, the best model was the last one, where a one-versus-all approach was taken and all other features aside from the target, carbon monoxide, were used to predict carbon monoxide emissions. The worst model overall was the simple linear regression using Turbine Inlet Temperature as a gauge for carbon monoxide emissions.

Add a new chunk by clicking the *Insert Chunk* button on the toolbar or by pressing *Ctrl+Alt+I*.