Exercise 2 - Multiple Linear Regression

Load Dataset

We first load ozone dataset and store it into df variable. The dataset consists of 14 columns. First column is the *date* of the observation but won't be used in our analysis.

```
df <- read.table("ozone.txt", header = TRUE, sep=" ")
df <- df[, 2:ncol(df)]
attach(df)
head(df)
```

```
##   maxO3   T9  T12  T15 Ne9 Ne12 Ne15     Wx9    Wx12    Wx15 maxO3y  wind  rain
## 1    87 15.6 18.5 18.4   4    4    8  0.6946 -1.7101 -0.6946     84 North   Dry
## 2    82 17.0 18.4 17.7   5    5    7 -4.3301 -4.0000 -3.0000     87 North   Dry
## 3    92 15.3 17.6 19.5   2    5    4  2.9544  1.8794  0.5209     82  East   Dry
## 4   114 16.2 19.7 22.5   1    1    0  0.9848  0.3473 -0.1736     92 North   Dry
## 5    94 17.4 20.5 20.4   8    8    7 -0.5000 -2.9544 -4.3301    114  West   Dry
## 6    80 17.7 19.8 18.3   6    6    7 -5.6382 -5.0000 -6.0000     94  West Rainy
```

Second column labeled as *max03* is the depended variable which we would like to predict by constructing a model from the remaining 12 variables, *temperature(T)*, *neon(Ne)*, *Wx*, *Wind* and *Rain*.

Then we make a simple check for potential missing values in our dataset.

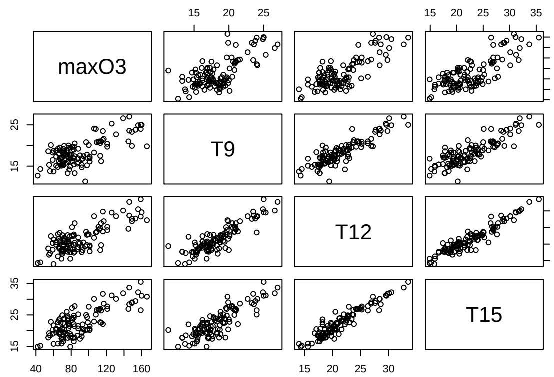
```
head(is.na.data.frame(df))
```

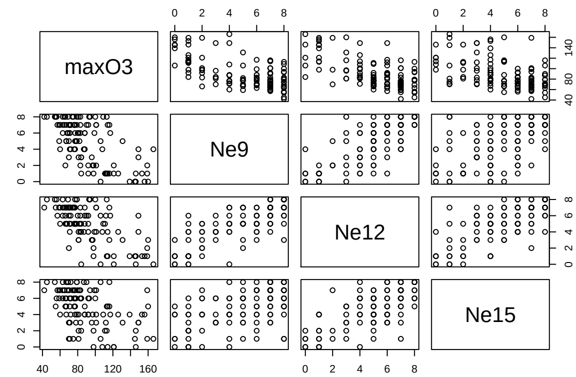
```
##      maxO3    T9   T12   T15   Ne9  Ne12  Ne15   Wx9  Wx12  Wx15 maxO3y  wind
## [1,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  FALSE FALSE
## [2,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  FALSE FALSE
## [3,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  FALSE FALSE
## [4,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  FALSE FALSE
## [5,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  FALSE FALSE
## [6,] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  FALSE FALSE
##       rain
## [1,] FALSE
## [2,] FALSE
## [3,] FALSE
## [4,] FALSE
## [5,] FALSE
## [6,] FALSE
```

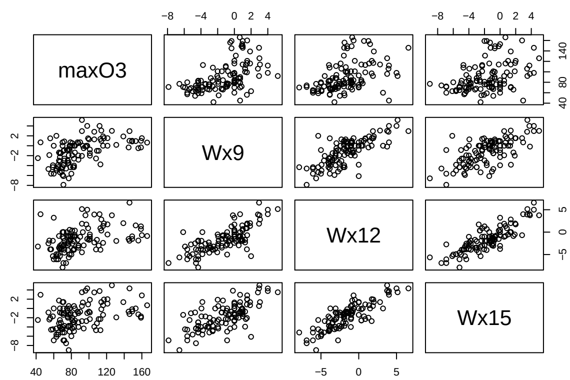
Undestanding the data

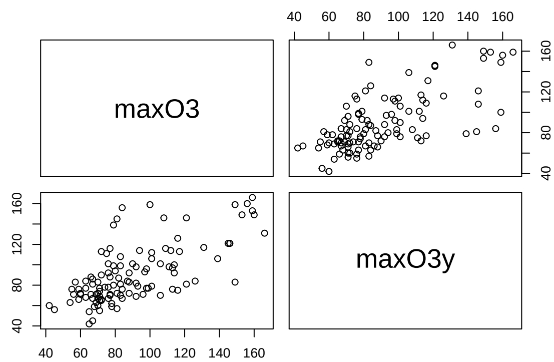
Once we have validated the integrity of the dataset, we plot the dataset in pairs. We use *pairs* command. In order to keep our plots simple and readable, we will plot *max03* with the 3 main variables, *Temperature*, *Ne*, *Wx* separately.

```r
pairs(subset(df, select = c(1,2,3,4)))
pairs(subset(df, select = c(1,5,6,7)))
pairs(subset(df, select = c(1,8,9,10)))
pairs(subset(df, select = c(1,11)))
```









Intuitively, we could say that max03 levels are more correlated with temperature compared to *Wx9* and *Ne*.

Pearson coefficient

We've seen before that temperature has a strong correlation with *max03*. In order to measure that relationship we will calculate the Pearson correlation coefficient R. Pearson coefficient measures the strength and direction of a linear relationship between two variables. The value of R is always between +1 and –1. Closer to +1 values means there is a very strong positive correlation between variables while closer to -1 a very strong negative correlation. 0 indicates that there is no linear correlation.

In order to compute coefficient R we use built-in method *cor* and we explicitly ask for pearson method.

```r
cor_9 = cor(df$T9, df$max03, method = c("pearson"))
cor_12 = cor(df$T12, df$max03, method = c("pearson"))
cor_15 = cor(df$T15, df$max03, method = c("pearson"))
```

	T9	T12	T15
r-coeff(max03)	0.6993865	0.7842623	0.77457

We observe that R9, R12 and R15, 0.6993865, 0.7842623, 0.77457 respectively are all positive and close to 1 which indicates a strong positive linear relationship.

Regression

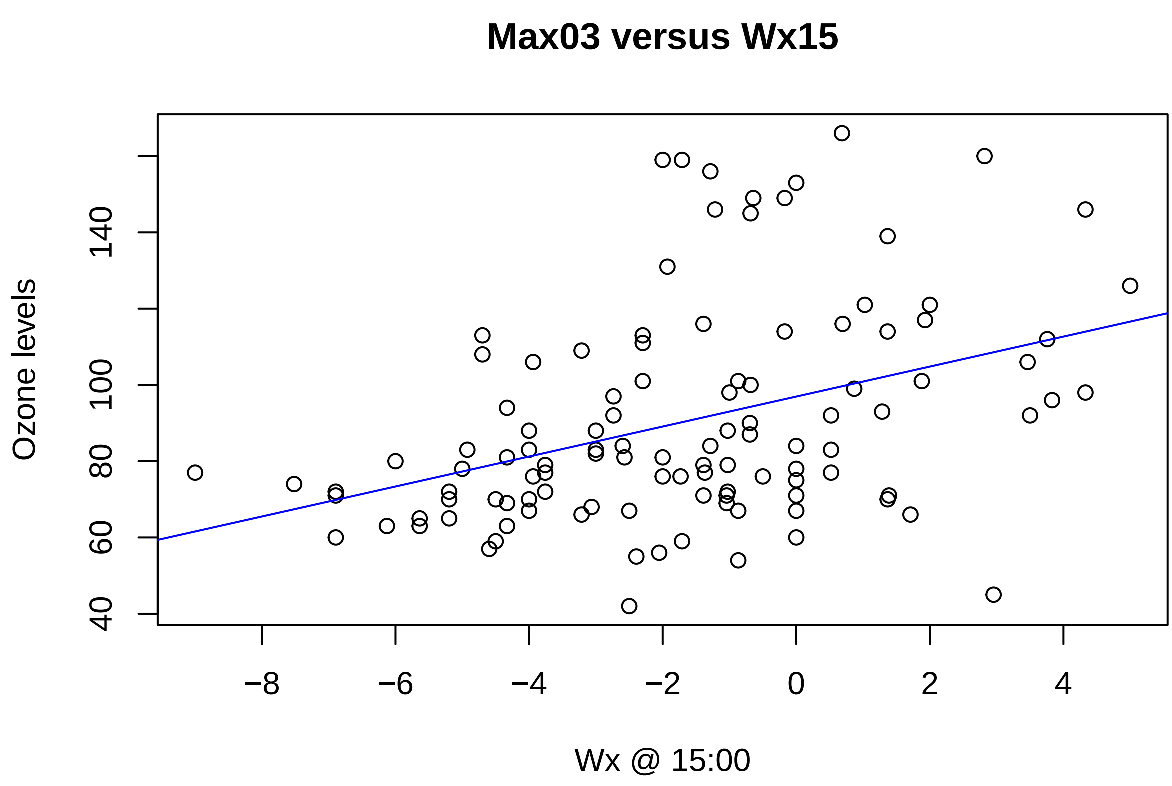
Simple Linear Regression

Previous analysis hinted that *Temperature* varialbe has the strongest correlation with *max03*. For the purposes of the excercise, we will use *Wx15* as a regressor to construct a simple linear model. A simple linear model is expressed mathematically as shown below.

$$\hat{y} = \beta_0 + \beta_1 x + \epsilon \tag{1}$$

In R, we simply use the command *lm* (linear model) to fit a linear model to observations and *summary* to get basics statistics of the fit.

```
simple.model <- lm(df$max03 ~ df$Wx15)
plot(df$Wx15, df$max03, main = "Max03 versus Wx15", xlab = "Wx @ 15:00", ylab = "Ozone levels")
abline(simple.model, col="blue")
```




```
summary(simple.model)
```

```
##
## Call:
## lm(formula = df$max03 ~ df$Wx15)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -63.563 -16.230  -5.045  11.872  69.912
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  96.9494     2.8761  33.709  < 2e-16 ***
## df$Wx15       3.9309     0.8799   4.468 1.93e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 26.05 on 110 degrees of freedom
## Multiple R-squared:  0.1536, Adjusted R-squared:  0.1459
## F-statistic: 19.96 on 1 and 110 DF,  p-value: 1.927e-05
```

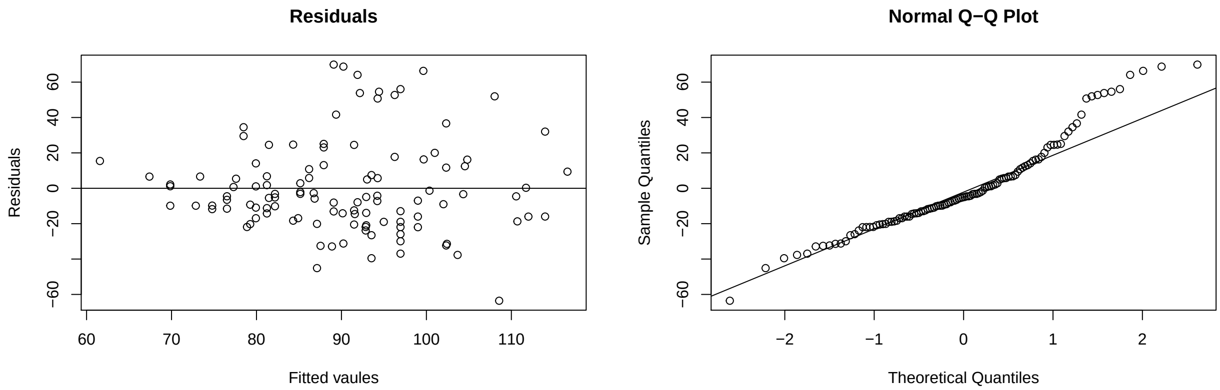
Generally, R-squared is the percentage in variation in dependent variable, in this case *max03* that can be explained by the model. It is defined as follows:

$$R^2 = \frac{\text{Variance explained by the model}}{\text{Total variance}} \tag{2}$$

Usually, larger R-squared value indicates a better linear model that fits the observations. Visually, it means that the observed data points are closer to the regression line. Limitation of R-squared coefficient is that it does not provide any information whether our model is biased to the data. R-squared can be misleading when you assess the goodness-of-fit for linear regression analysis. A good model could have a low R-squared value which we will deal with it later by performing a residuals plots analysis.

Residual Plots

```r
simple.model.residuals = resid(simple.model)
simple.model.fitted = fitted.values(simple.model)
plot(simple.model.fitted, simple.model.residuals,
          ylab = "Residuals",
          xlab = "Fitted vaules",
          main = "Residuals")
abline(0,0)
#create Q-Q plot for residuals
qqnorm(simple.model.residuals)
#add a straight diagonal line to the plot
qqline(simple.model.residuals)
```



The x-axis displays the fitted values and the y-axis displays the residuals. From the plot we can see that the spread of the residuals tends to be higher for higher fitted values. Q-Q plot shows that the residuals don't follow a normal distribution as the upper tail tends to stray away for the line.

Multiple Linear Regression

We will use multiple regressors to predict max03 taking into consideration the 3 variables *T12*, *Ne12* and *Wx12*. Multiple linear regression model is as expressed similarly to (1) but with number of regressors p>1.

$$\hat{y} = \beta_0 + \beta_1 x_9 + \beta_2 x_{12} + \beta_3 x_{15} \tag{3}$$

Matrix notation:

$$Y = X\beta + \epsilon \tag{4}$$

In R we include the additional variables *Ne12* and *Wx12*.

```r
multi.model <- lm(df$max03 ~ df$T12 + df$Ne12 + df$Wx12)
summary(multi.model)
```

```
##
## Call:
## lm(formula = df$max03 ~ df$T12 + df$Ne12 + df$Wx12)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.462 -11.448  -0.722   8.908  46.331
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   3.8958    14.8243   0.263   0.7932
## df$T12        4.5132     0.5203   8.674 4.71e-14 ***
## df$Ne12      -1.6189     1.0181  -1.590   0.1147
## df$Wx12       1.6290     0.6571   2.479   0.0147 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.63 on 108 degrees of freedom
```

```
## Multiple R-squared:  0.6612, Adjusted R-squared:  0.6518
## F-statistic: 70.25 on 3 and 108 DF,  p-value: < 2.2e-16
```

As expected, R-squared value is higher than the *simple.model* as it never decreases when new predictors are added. R-squared is encouraging us to make more complex model for the prediction of *max03*. Though, that would result to *overfitting*. Adjusted R-square coefficient is defined as shown below.

$$AdjustedR^2 = 1 - (1 - R^2)\frac{n-1}{n-p-1} \tag{5}$$

p - number of regressors.
n - sample size.

For every regressor added in the model there is a penalty factor. As the denominator decreases the fraction increases, thus R^{2}-adjusted gets smaller. In case R^{2} is significantly larger with the addition of new regressors then adding new variables to the model was worth it.