Seminar 6

Linear Regression Model

From this tutorial document, we will learn how to develop a linear regression model in R. For the exercises, we will use the same Motor Trends dataset (http://www.jstor.org/stable/2530428) that we used in the previous tutorial. The dataset was compiled from 1974 issues of Motor Trends magazine and is included with R Base package. Let's start with loading the dataset. You don't need to import any data from you project folder for this exercise.

```
> data(mtcars)
```

You can check the detailed descriptions of the variables.

```
> ?mtcars
```

We will use the lm() function to fit a linear regression model to data with R. This function is included in the 'stats' package, which should be included and loaded by default with your R installation.

```
> help(lm)
```

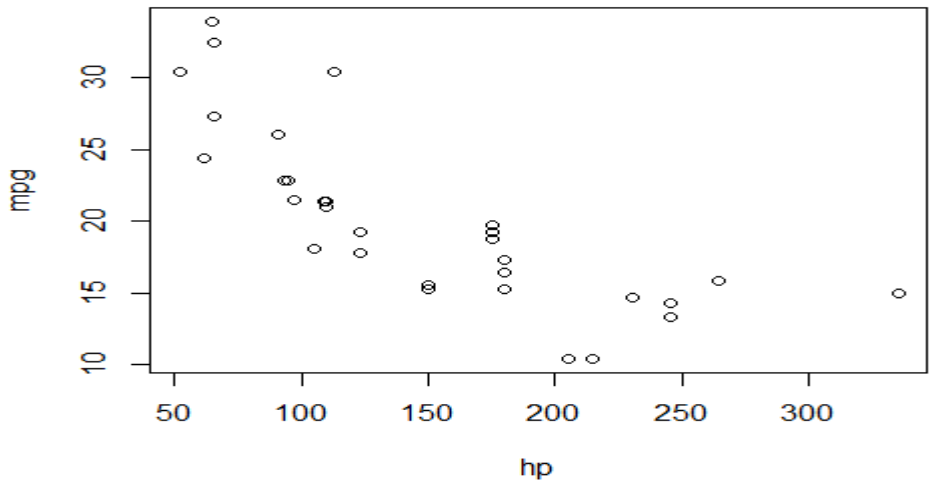
Simple Regression Model

A simple regression model include only one independent variable to estimate/predict the dependent variable. Below I declare a simple linear regression model where I use horse power (*hp*) to predict miles per gallon (*mpg*).

```
mpg ~ hp
```

First, let's see the relationship between *mpg* and *hp* from the below scatter plot.

```
> plot(mpg~hp, data = mtcars)
```



It seems *hp* is negatively associated with *mpg* (as *hp* increases, *mpg* decreases). The correlation between *hp* and *mpg* is -0.7761684 (i.e., cor(mtcars$hp, mtcars$mpg))

- **Model Setup**

Then, let's run a simple regression model. Here the left hand side of the tilde (~) is the dependent variable, and the right hand side is the predictor.

> reg1 <- lm(mpg ~ hp, data = mtcars)

> reg1

```
Call:
lm(formula = mpg ~ hp, data = mtcars)

Coefficients:
(Intercept)           hp
   30.09886     -0.06823
```

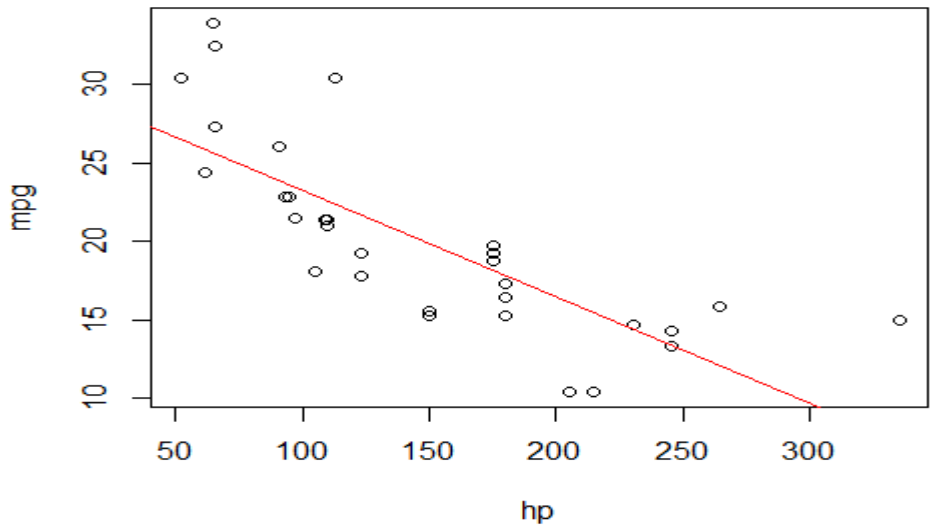
- **Interpretation**

Understanding the regression coefficients is very straightforward. The intercept is the predicted values of mpg when the independent variable is equal to zero. As is the case here, quite often the intercept has little value alone because it is impossible to have value(s) of zero for the independent variable(s). For example, since no car exists with zero horse power (*hp* = 0), the intercept has no real-world interpretation.

The slope coefficient indicates (*hp= -0.06823*) the estimated increase in *mpg* for an increase of one in *hp*. That is, a one-unit increase in *hp* decreases *mpg* by an average of 0.06823.

Then, let's present the estimated regression line on the scatter plot.

> abline(reg1, col="red")



It seems the regression line fits the data.

- **Model Evaluation**

By using the summary function, we can check the detailed regression estimation outcomes.

> summary(reg1)

```
Call:
lm(formula = mpg ~ hp, data = mtcars)

Residuals:
    Min      1Q  Median      3Q     Max
-5.7121 -2.1122 -0.8854  1.5819  8.2360

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 30.09886    1.63392  18.421  < 2e-16 ***
hp          -0.06823    0.01012  -6.742 1.79e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.863 on 30 degrees of freedom
Multiple R-squared:  0.6024,    Adjusted R-squared:  0.5892
F-statistic: 45.46 on 1 and 30 DF,  p-value: 1.788e-07
```

A

B

C

As indicated by the labels (A, B, C) in the above output, the output presents the three key ways to evaluate the performance (fit) of our regression model.

A: The first section shows summary statistics for the errors in the predictions. Since a residual is equal to the actual value minus the predicted value (see the lecture slides), the maximum error of 8.32360 suggest that the model under-predicted by nearly *8 mpg* for at least one observation. On the other hand, 50 percent of errors fall within the 1Q (25th percentile) and 3Q (75th percentile) values, so the majority of predictions were between *2.1122 mpg* over the true value and *1.5819 mpg* under the true value.

B: For each estimated regression coefficients, the p-value provides an estimate of the probability that the true coefficient is zero given the value of the estimate, denoted **$Pr(B_1 = 0 \mid hp = -0.06823)$**. Small p-values suggest that the true coefficient is very unlikely to be zero, which means that the predictor is extremely unlikely to have no relationship with the dependent variable.

Note that some of p-values have stars (***), which correspond to the footnotes to indicate the significant level met by the estimate. This level is a threshold, chosen prior to building the model, which will be used to indicate "real" findings. In this example, a negative association between *hp* and *mpg* is ***statistically significant*** (***) and the real finding from the model.

C: The R-squared value (or, the coefficient of determination) provides a measure of how well the model as a whole explains the values of the dependent variable. The closer the value is to 1.0, the better the model perfectly explains the data. Since the R-squared value is 0.6024, we know that the model including only one predictor (i.e., *hp*) explains about 60% percent of variation in the dependent variable (i.e., *mpg*). As models with more predators always explain more variation, the adjusted- R-squared value corrects R-squared by penalizing models with a large number of predictors. Note that it is useful for comparing the performance of models with different numbers of predictors.

Finally, the F-statistic presents whether all coefficients in the regression model are equal to zero ($B_1 = B_2 = \ldots = 0$), which means the model does not capture any significant associations with the dependent variables (i.e., *zero* association with *mpg*). Simply, the lower p-value indicates that the predictors in the model have non-zero effects on the dependent variable. Here, the p-value of 1.788e-07 means the predictor, *hp*, is not highly likely to have a zero effect on *mpg*.

Multiple Regression Model

A multiple regression model includes multiple independent variables (or predictors) to estimate the dependent variable.

- **Model Setup**

The following command fits a linear regression model relating the three independent variables (horse power (*hp*), cylinders (*cyl*), and transmission type (*am*)) to the miles per gallon (*mpg*).

```
> reg2 <- lm( mpg ~ hp + cyl + am, data = mtcars)
```

Here the left hand side of the tilde (~) is the dependent variable, and the right hand side has the three predictors separated by plus signs (+).

```
> reg2

Call:
lm(formula = mpg ~ hp + cyl + am, data = mtcars)

Coefficients:
(Intercept)           hp          cyl           am
   30.88834     -0.03688     -1.12721      3.90428
```

- **Interpretation**

As we did in the simple regression model, we can interpret the estimated slope coefficients for the three predictors as follows:

- A one-unit increase in *hp* decreases *mpg* by an average of 0.03688.
- A one-unit increase in *cyl* decreases *mpg* by an average of 1.12721.
- A one-unit increase (i.e., the use of automatic transmission) in *am* increase *mpg* by an average of 3.90428.

- **Model Evaluation**

Let's see the detailed regression estimation outcomes by using the summary function.

> summary(reg2)

```
Call:
lm(formula = mpg ~ hp + cyl + am, data = mtcars)

Residuals:
   Min     1Q Median     3Q    Max
-4.864 -1.811 -0.158  1.492  6.013

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 30.88834    2.78422  11.094 9.27e-12 ***
hp          -0.03688    0.01452  -2.540  0.01693 *
cyl         -1.12721    0.63417  -1.777  0.08636 .
am           3.90428    1.29659   3.011  0.00546 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.807 on 28 degrees of freedom
Multiple R-squared:  0.8041,    Adjusted R-squared:  0.7831
F-statistic: 38.32 on 3 and 28 DF,  p-value: 4.791e-10
```

A

B

C

I skip the detailed explanation for the above output.

A: The maximum error of 6.013 suggest that the model under-predicted by nearly *6 mpg* for at least one observation. On the other hand, 50 percent of errors fall within the 1Q and 3Q values, so the majority of predictions were between *0.158 mpg* over the true value and *1.492 mpg* under the true value. As you can see, the prediction errors (residuals) have been decreased as compared with the simple regression model had.

B: The p-values for the slope coefficients of *hp, cyl*, and *am* indicate that while *hp* (*) and *am (**)* are ***statistically significant*** in predicting mpg, *cyl* is not significant. Therefore, we find that *hp* and *am* have meaningful associations with *mpg*.

C: Overall, the model explanatory power (adjusted R-squared values) has been improved from the simple regression model (0.5892) to the multiple regression model (0.7831). The model with the three predictors explains about 78% percent of variation in the dependent variable (i.e., *mpg*). The F-statistic indicates that the three predictors do not have zero values. Therefore, the model specification is not incorrect.

- **Prediction**

Finally, we have the predictive model from the regression models. Here is the predicted model from the multiple regression model with the three predictors:

$$\hat{mpg} = -4.864 - 0.03688 * hp - 1.12721 * cyl + 3.90428 * am$$

Let's see how we can compare the actual values and the predicted values.

Mazda RX4 is features with the actual *mpg* = 21.0 (with *hp* = 110, *cyl* = 6, and *am* = 1).

```
> mtcars[1,] # 1st observatgion is Mazda RX4
           mpg cyl disp  hp drat   wt  qsec vs am gear carb
Mazda RX4   21   6  160 110  3.9 2.62 16.46  0  1    4    4
```

However, the estimated *mpg* for Mazda RX4 is 23.973 = 30.88834 – 0.03688*(110) - 1.12721*(6) + 3.90428*(1)

You can also see all the predicted values (fitted values) as follows. I used the head function to limit the output.

```
> head(reg2$fitted.values)
       Mazda RX4    Mazda RX4 Wag       Datsun 710   Hornet 4 Drive Hornet Sportabout          Valiant
        23.97302         23.97302         26.85433         20.06874         15.41740         20.25312
```

You can use the predicted model to predict new datasets, here I am modifying Datsun 710 in the red circle to see how the miles per gas (*mpg*) may have been influenced if the car was automatic instead of manual transmission.

```
> newCar <- mtcars[3,] # 3rd observation is Datsun 710
> newCar$am <- 0 # what if the car was automatic?
> predict(reg2, newdata = newCar) # The prdict function will return the predicted values
Datsun 710
  22.95005
```

The prediction (*mpg* = 22.95005) went down by about 4 miles.