INSH5301 Intro Computational Statistics

Ali Banijamali

03/02/2020

```
# Required Libraries:
library(ggplot2)
```

Load the mtcars dataset in R. (use data(mtcars)) Using this sample of cars, we are interested in knowing if:

(a) there is any relation between the miles per gallon (mpg) and the gross horsepower (hp),

(b) the sign of the relation, and,

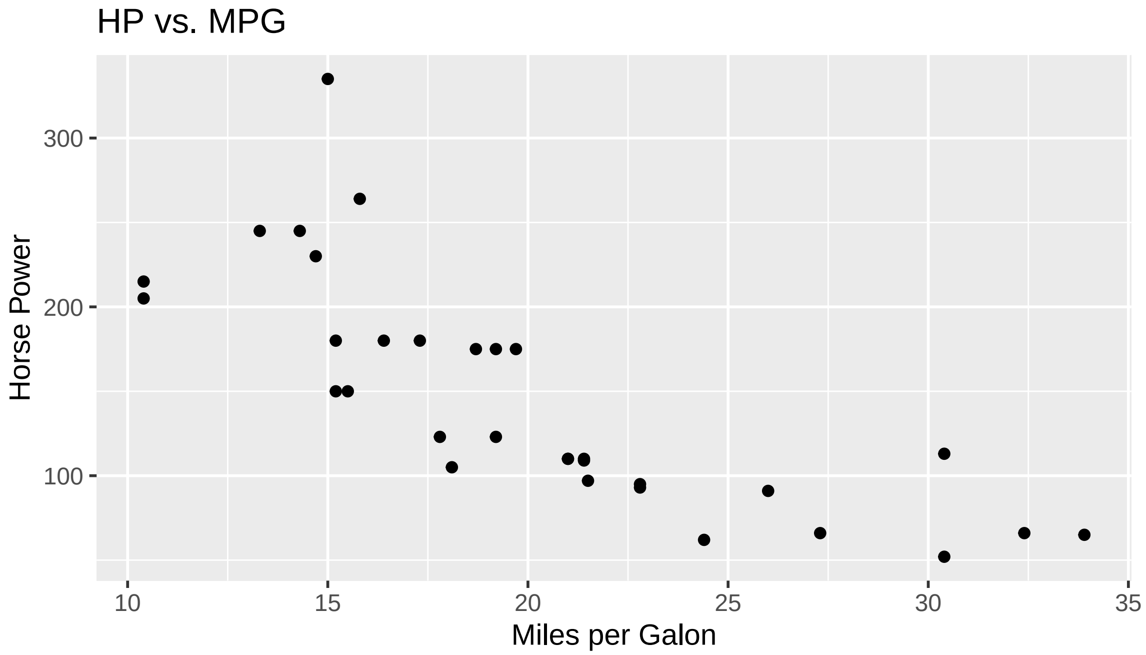
(c) the magnitude - strenght - in which this variables are associated.

```
cars <- as.data.frame(mtcars)
```

1. Using ggplot2, make a scatter plot with mpg on x-axis and hp on the y-axis. Using the graph as evidence, do you think the variables are positively, negatively or not related?

ANS.

```
ggplot(cars, aes(x=mpg, y=hp)) +
  geom_point() +
  xlab('Miles per Galon') + ylab('Horse Power') + ggtitle('HP vs. MPG')
```



Based on the plot, hp and mpg are negatively correlated. This also makes sense, a more powerful engine, generally consumes more fuel and that brings down mpg.

2. Compute the covariance between the two variables. How can we use the covariance to answer (a,b,c)?

ANS.

```
# Relationship between mpg and hp (Based on COV):
cov(cars$mpg, cars$hp)
```

```
## [1] -320.7321
```

Covarience is well distant from zero, this means that the variables are related. Also, the covarience is below zero, so they are negatively related. However, we can hardly say anything about the strength of the relationship based on the covarience. Here is where correlation is useful because it is scaled and always between -1 and 1.

3. Calculate the correlation between the two variables. How can we use the correlation coefficient to answer (a,b,c)?

ANS.

```
# Relationship between mpg and hp (Based on COR):
cor(cars$mpg, cars$hp)
```

```
## [1] -0.7761684
```

Here, again we can see a good correlation value (-0.7761684). This is a rather meaningful correlation. It is also negative, meaning that the variables are negatively correlated.

4. Estimate the regression coefficients Beta0 and Beta1. What's the interpretation of Beta1?

ANS.

```
bivariate_model <- lm(hp~mpg, data=cars)
summary(bivariate_model)
```

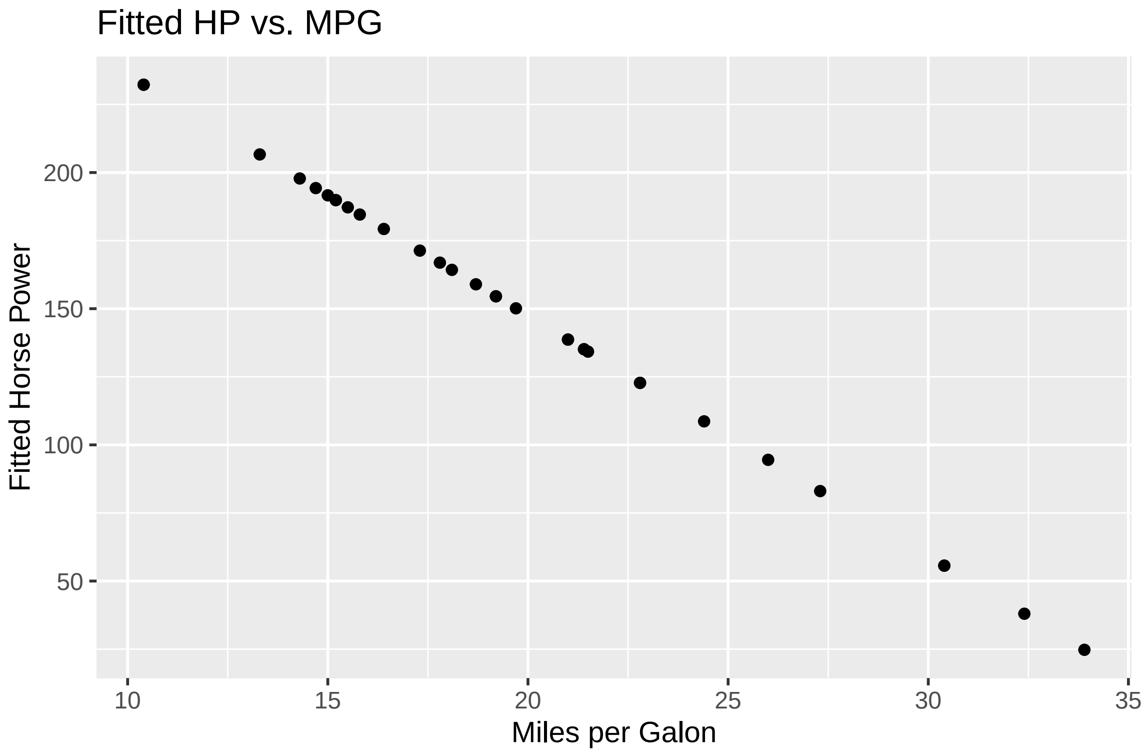
```
##
## Call:
## lm(formula = hp ~ mpg, data = cars)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.26 -28.93 -13.45  25.65 143.36
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   324.08      27.43  11.813 8.25e-13 ***
## mpg            -8.83       1.31  -6.742 1.79e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 43.95 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
```

According to the results, $\beta_0 = 324.08$ and $\beta_1 = -8.83$. β_1 is the slope of the regression line. $\beta_1 = -8.83$ means for every 1 unit of mpg, hp decreases by 8.83 units. Meaning if you want to have a better fuel efficiency, you should opt for a car which has a lower power engine (hp), which completely makes sense.

5. Write a function that, using the estimated coefficients, returns the fitted value of hp given some mpg. Find all the fitted values using all observations of mpg. Make a scatterplot of mpg and the fitted values.

ANS.

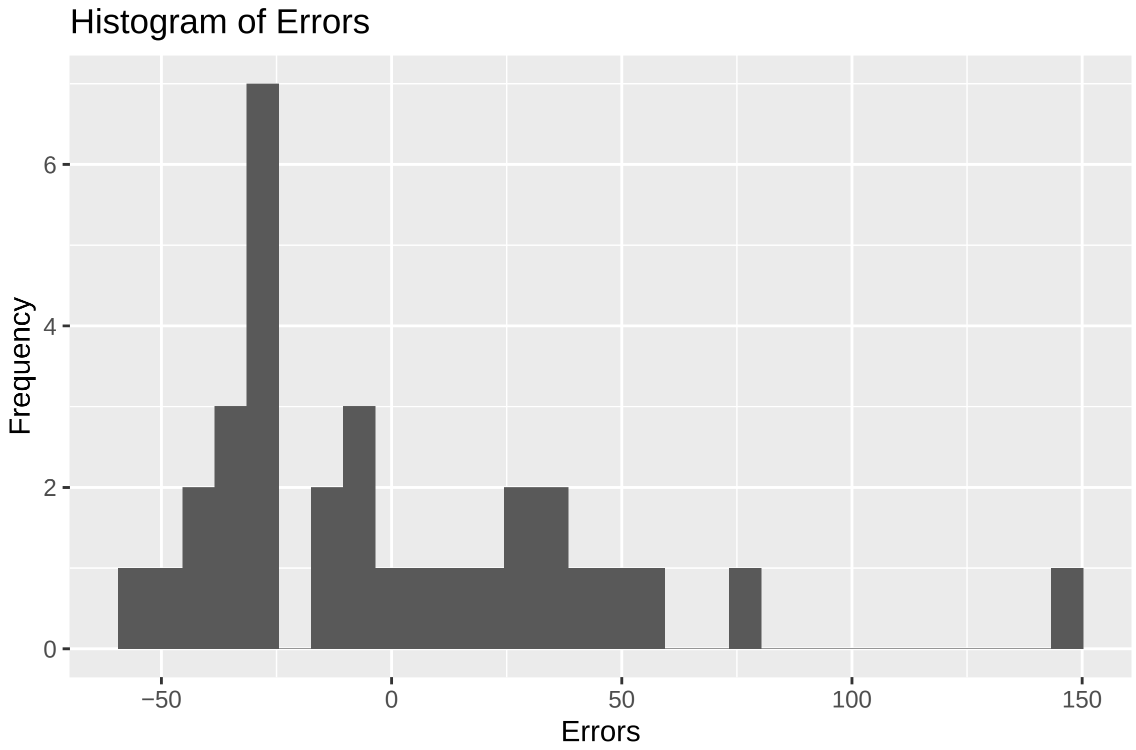
```r
fit.vals <- function(x.data, y.data){
  bivar.model <- lm(y.data~x.data)

  intercept <- as.numeric(bivar.model$coefficients[1])
  slope <- as.numeric(bivar.model$coefficients[2])

  fitted.y <- slope*(x.data) + intercept

  all.in.one <- data.frame(x=x.data,
                           actual.y=y.data,
                           fitted.y=fitted.y)
  return(all.in.one)
}

mpg.hp.fitting <- fit.vals(x.data=cars$mpg, y.data=cars$hp)

ggplot(mpg.hp.fitting, aes(x=x, y=fitted.y)) +
  geom_point() +
  xlab('Miles per Galon') + ylab('Fitted Horse Power') + ggtitle('Fitted HP vs. MPG')
```



6. Calculate the errors of the regression, i.e. compute the difference between the fitted values and the observed values. Make an histogram with the errors. Do they look normally distributed?

ANS.

```
# Calculating the errors:
errors <- data.frame(Errors=mpg.hp.fitting$actual.y - mpg.hp.fitting$fitted.y)

# Histogram of the errors:
ggplot(errors, aes(x=Errors)) + geom_histogram() +
    ggtitle(paste("Histogram of Errors")) + xlab("Errors") + ylab('Frequency')
```



Yes, they do look to be normally distributed, as we expected them to be that way!

7. Compute the standard error of Beta1 and use a t-test to check wheter Beta1 is significant - statistically different than zero -. Calculate the p-value and 95% confidence interval to verify that you reach the same conclusion.

ANS.

```
bivariate_model <- lm(hp~mpg, data=cars)
summary(bivariate_model)
```

```
##
## Call:
## lm(formula = hp ~ mpg, data = cars)
##
## Residuals:
##     Min      1Q Median     3Q    Max
## -59.26 -28.93 -13.45  25.65 143.36
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   324.08      27.43  11.813 8.25e-13 ***
## mpg            -8.83       1.31  -6.742 1.79e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 43.95 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
```

The standard error of β_1 is 1.31. According to the test, the p-value for β_1 is 1.79e-07, which is quite significant. However we could calculate it by:

```r
# p-value for Beta1 according to the previous test:
2*pt(-6.742, 30, lower.tail=T) # D.O.F=30
```

```
## [1] 1.789735e-07
```

This (1.789735e-07) confirms the previous calculations. Now let's calculate the %95 confidence interval:

```r
# %95 confidence interval:
# Any t-test greater than:
qt(0.975, 30) # D.O.F=30
```

```
## [1] 2.042272
```

```r
# Or, any T-test lower than:
qt(0.025, 30)
```

```
## [1] -2.042272
```

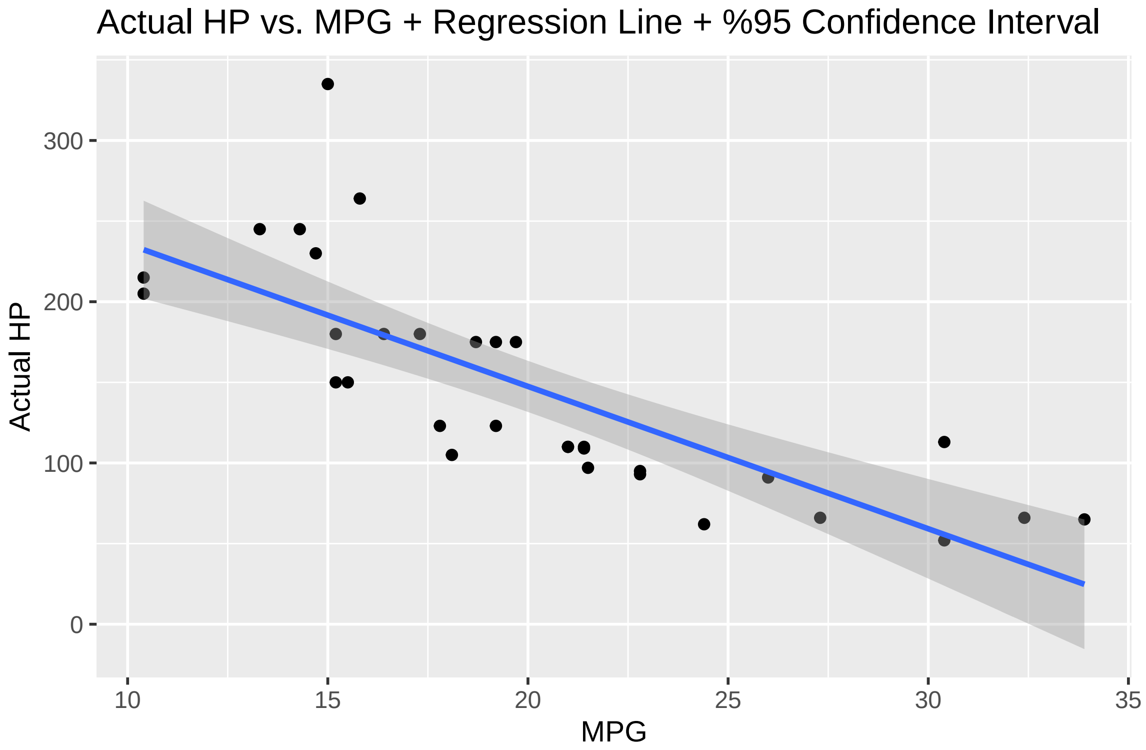
For our β_1, T-test is -6.742, which is clearly smaller than -2.042272.

8. Calculte the R2 coefficient and interpret it's value.

ANS.
R^2 is calculated in question 7 and is 0.6024. The adjusted R^2 is 0.5892. R-squared tells us how much of the variation of hp is predicted by mpg. A value of ≈ 0.6 means a good portion ($\approx$ %60) of the variation is explained by mpg. (Note that this value is between 0 to 1.)

9. Make a new graph with ggplot2, this time make a scatter-plot with the regression line and a 95% C.I. for the fitted values.

```r
ggplot(mpg.hp.fitting, aes(x=x, y=actual.y)) + geom_point() + geom_smooth(method=lm) +
  xlab('MPG') + ylab('Actual HP') +
  ggtitle('Actual HP vs. MPG + Regression Line + %95 Confidence Interval')
```


Actual HP vs. MPG + Regression Line + %95 Confidence Interval
Actual HP
300
200
100
0
10
15
20
25
30
35
MPG