Week 11: Linear Regression Using R

Christian Thieme

11/6/2020

// Using the "cars" dataset in R, build a linear model for stopping distance as a function of speed.

R has a built in dataset named cars. This dataset has 50 observations and 2 columns:

- speed - a numeric field for the speed in miles per hour (mph)
- dist - a numeric field for the stopping distance in feet (ft)

We'll import the dataset here:

```
library(tidyverse)

head(cars)
```

```
##   speed dist
## 1     4    2
## 2     4   10
## 3     7    4
## 4     7   22
## 5     8   16
## 6     9   10
```

Before doing our analysis, let's do a quick check of the data to make sure everything is in order. Let's take a look and see if we have any missing data.

```
colSums(is.na(cars))
```

```
## speed  dist
##     0     0
```

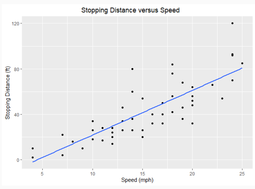
It looks like our data is complete. Next, we'll run some quick summary statistics.

```
summary(cars)
```

```
##      speed           dist       
##  Min.   : 4.0   Min.   :  2.00  
##  1st Qu.:12.0   1st Qu.: 26.00  
##  Median :15.0   Median : 36.00  
##  Mean   :15.4   Mean   : 42.98  
##  3rd Qu.:19.0   3rd Qu.: 56.00  
##  Max.   :25.0   Max.   :120.00
```

Looking at the above summary statistics we can see that the speed column has a relatively small range (21 mph). Additionally 50% of the data falls between 12 mph and 19 mph, making the interquartile range 7 mph. Distance on the other hand has a much wider range (118 ft). We also notice that 75% of the data falls between 2 ft and 56 ft, but the max is 120 ft, which may mean we have some outliers we need to be aware of. With this first initial glance, lets visualize our data. We'll do this using ggplots `geom_point` function adding in `geom_smooth` for our least squares line. As our arguments to `geom_point`, X will be our explanatory variable, "Speed", and Y will be our response variable, "Distance". Conventionally, we'll plot the response variable on the Y axis.

```
ggplot(cars) +
  aes(x = speed, y = dist) +
  geom_point() +
  geom_smooth(method = lm, se = FALSE) +
  labs(title = "Stopping Distance versus Speed") +
  ylab("Stopping Distance (ft)") +
  xlab("Speed (mph)") +
  theme(
        plot.title = element_text(hjust = 0.45)
  )
```



Looking at the above scatter plot, we can definitely see that there is fairly strong linear realtionship between these two variables. As speed increases along the X axis, stopping distance also increases - which logically makes sense.

Let's now use linear regression to build a model using speed as a predictor for stopping distance. We'll use R's `lm` function to build a linear model. The call to `lm` is different than our call to ggplot. Here, we'll first identify our response variable, then we'll identify the explanatory variable.

```
model <- lm(dist~speed, data = cars)
model
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Coefficients:
## (Intercept)        speed
##     -17.579        3.932
```

The output of lm gives us our y-intercept as well as our slope. We can re-write the above as the following function:

$$y = 3.932x - 17.579$$

How can we tell if our model above is a good one? The first thing we can do is take a look at some of the model diagnostics using the `summary` function on the model object:

```
summary(model)
```

```
##
## Call:
## lm(formula = dist ~ speed, data = cars)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -29.069  -9.525  -2.272   9.215  43.201
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.5791     6.7584  -2.601  0.0123 *
## speed         3.9324     0.4155   9.464 1.49e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 15.38 on 48 degrees of freedom
## Multiple R-squared:  0.6511, Adjusted R-squared:  0.6438
## F-statistic: 89.57 on 1 and 48 DF,  p-value: 1.49e-12
```

Looking at the first line displaying the residuals, we can see the median is very close to 0 which is promising. Additionally the 1st quartile and 3rd quartile are both very close and so also suggest that variation within the residuals is fairly evenly distributed. Looking at the max and min, we'd also assume that these numbers be fairly similar, however it looks like the max value is quite a bit larger (absolute value) than the minimum. We'll investigate the residuals shortly with some plots to see if we can get an better view of this.

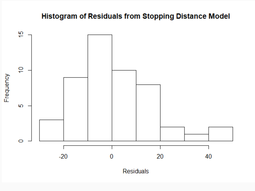
The next section of the output are the coefficients. For a good model, we'd expect the standard error to be at least 5-10 times smlaler than the corresponding coefficient. For speed, the standard error is 0.4155 and the coefficient is 3.9324 making the error 9.47 (3.9324/0.4155) times smaller than the coefficient. Having a large ratio gives us some assurance that there is relatively low variability in our slope estimate. The standard error for our slope estimate is 6.7584 which is a little less than 1/3 the value of our intercept. This suggests that the estimate for the y-intercept can vary quite a bit.

The last column shows the probability that the coefficient is NOT relevant to the model. For our model, speed's p-value is incredibly small, meaning there is an incredibly small chance that speed is NOT relevant to determining stopping distance. The p-value for our intercept is 0.0123, which is less than the generally accepted value of 0.05. This indicates that there is about a 1% chance that this intercept value is NOT relevant to our model.

Lastly, we'll look at the the Adjusted R-Squared value of 0.6438. This value tells us that using our simple linear regression where we try to predict stopping distance based on speed, we are able to explain about 64% of the variation of stopping distance with our single variable, speed.

Given that our Adjusted R-Squared is fairly good, can we rely on this model? When using linear regression, there are several assumptions that need to be met. One of these assumptions requires that we investigate the residuals to make sure that they are nearly normal. Let's do that now:

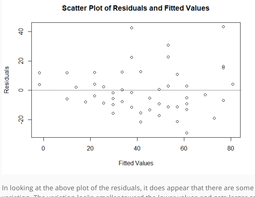
```
hist(model$residuals, main = "Histogram of Residuals from Stopping Distance Model",
     xlab = "Residuals")
```



Looking at the above histogram, we can see that our residuals are *nearly* normal. However, there appears to be some outliers at the far right of the plot.

Next, let's check to see if the variability in our residuals in nearly constant.

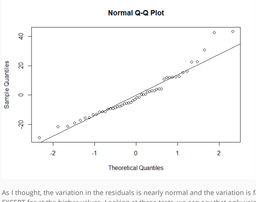
```
plot(model$residuals ~ model$fitted.values,
     main = "Scatter Plot of Residuals and Fitted Values",
     xlab = "Fitted Values", ylab = "Residuals")
abline(h = 0, lty = 3)
```



In looking at the above plot of the residuals, it does appear that there are some differences in variation. The variation looks smaller toward the lower values and gets larger as the values increase.

Let's also look at the *quantile-versus-quantile* plot.

```
qqnorm(resid(model))
qqline(resid(model))
```



As I thought, the variation in the residuals is nearly normal and the variation is fairly constant EXCEPT for at the higher values. Looking at these tests, we can say that only using speed as a predictor for our model would be insufficient to build an accurate model.

To make this model more accurate, we could take a look at the outliers and see if they are something that should be removed. However, caution and judgement should be used when removing any observations from the data. The more reasonable approach would be to add in additional factors to see if added complexity to the model would be able to better explain the data.