FinStats1 supplement

ArunK

11/3/2020

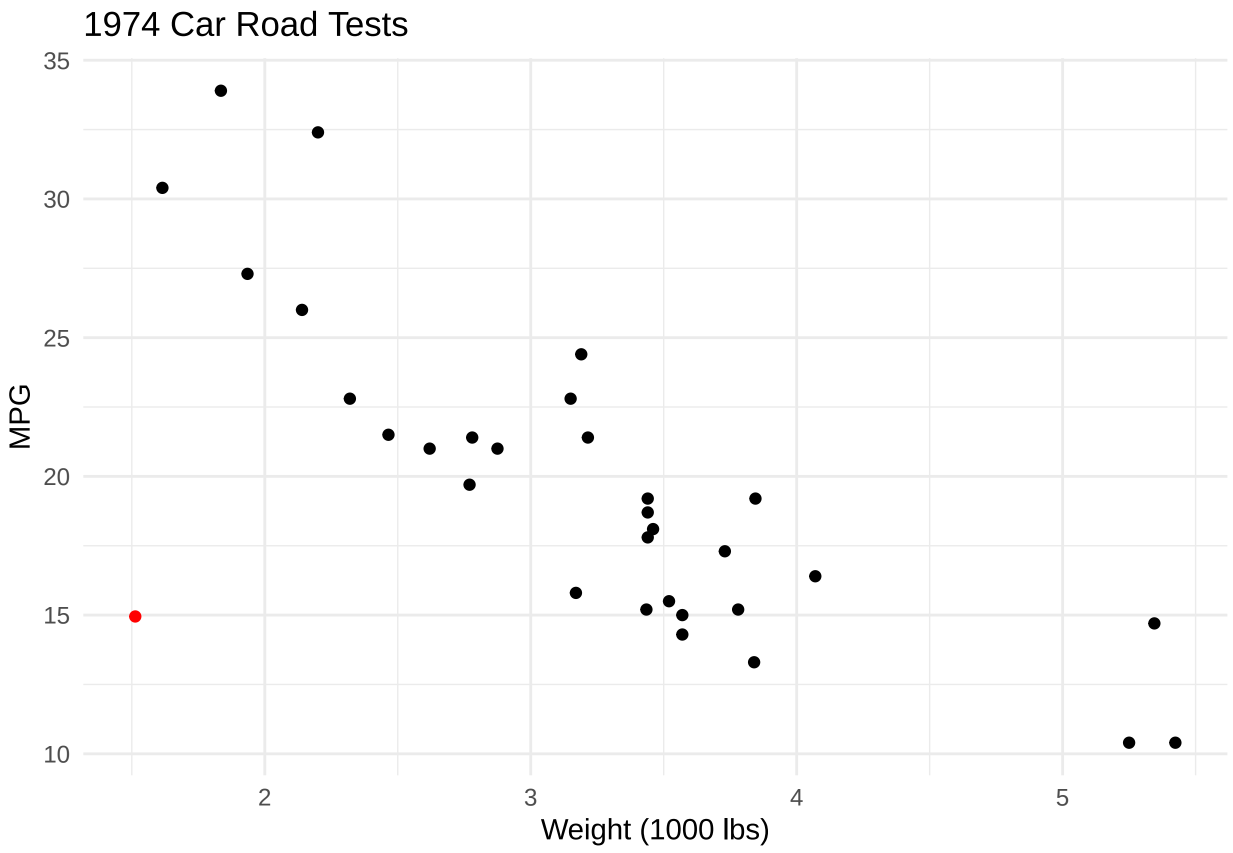
```r
library(tidyverse);
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

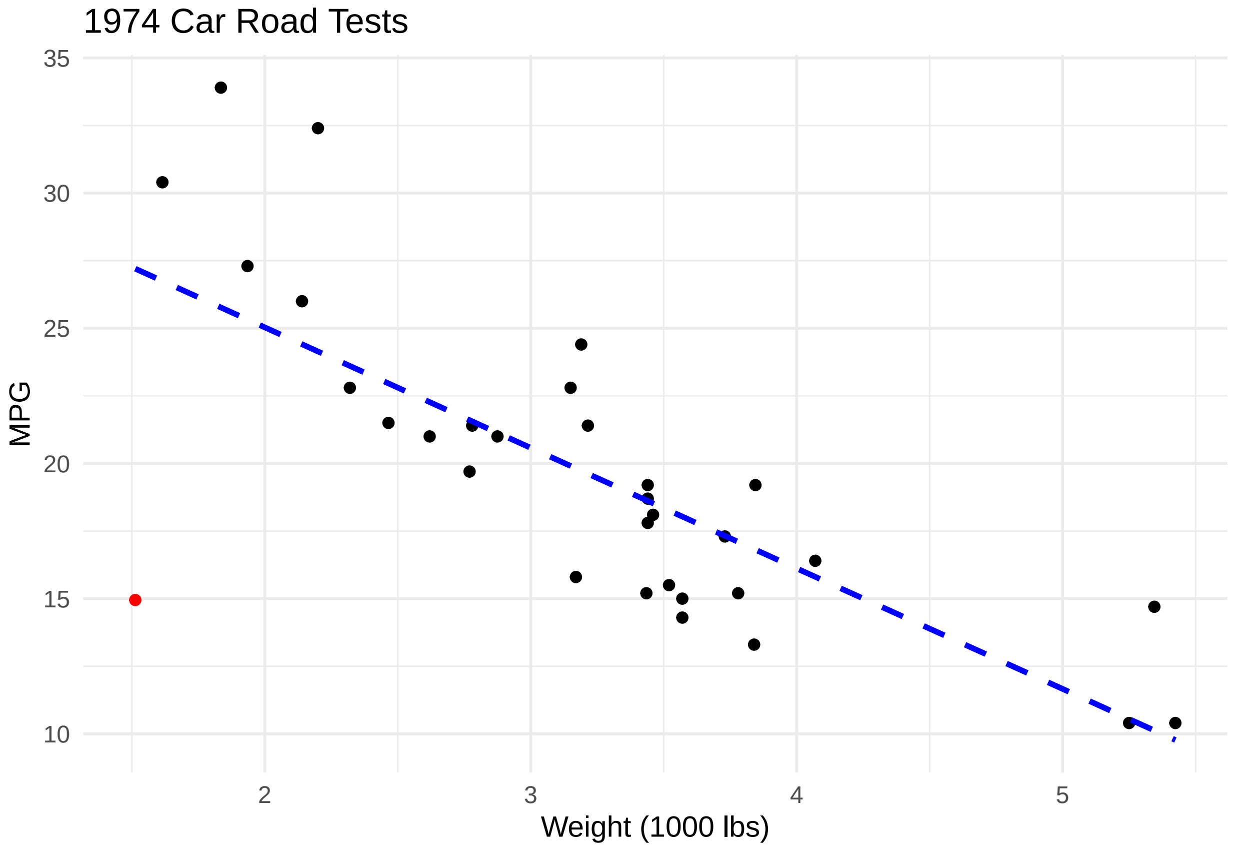
```r
mtcars["Lotus Europa", ]$mpg <- 14.95
mtcars$dum <- mtcars$mpg == 14.95
mtcars %>% ggplot(aes(x = wt, y = mpg, color = dum)) + geom_point() + xlab("Weight (1000 lbs)") + ylab(
```



I'll be working through a sample regression for the mtcars dataset, which has performance and design data for 32 cars collected from the 1974 Motor Trends Magazine. We're interested in exploring factors that might be useful in predicting miles per gallon for 1973-1974 cars; my first guess is that weight and mpg are somehow linearly related, and it looks like there is a such relationship, despite the presence of an outlier (in red).

```
mtcars %>% ggplot(aes(x = wt, y = mpg, color = dum)) + geom_point() + xlab("Weight (1000 lbs)") + ylab(
```

```
## `geom_smooth()` using formula 'y ~ x'
```



Let's quantify the fit of a single variable linear regression (Ordinary Least Squares):

```
lm(mtcars$mpg~mtcars$wt) %>% summary
```

```
##
## Call:
## lm(formula = mtcars$mpg ~ mtcars$wt)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -12.2541  -1.8999  -0.1403   1.8308   8.2580
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  33.9480     2.3851  14.233 7.01e-15 ***
## mtcars$wt    -4.4573     0.7102  -6.276 6.46e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## Residual standard error: 3.869 on 30 degrees of freedom
## Multiple R-squared:  0.5677, Adjusted R-squared:  0.5532
## F-statistic: 39.39 on 1 and 30 DF,  p-value: 6.462e-07
```

We can see that the slope coefficient is significant, that is there is a non-zero change in mpg for one unit change in car weight (in context +1000 lbs decreases the predicted mpg by 4.4573); there is less than 1E-06 chance of observing the given data if the null of no relationship was true. Looking at the model fit (R-squared and Adjusted-R-Squared), we interpret the following: the linear model explains .56 or .55 of the variation seen in the sample (adjusted R-squared penalizes higher model complexity, is usually lower and better to use to be conservative).

What about if we were interested in how both weight and horsepower impact mpg?

```
lm(mtcars$mpg~mtcars$wt+mtcars$hp) %>% summary
```

```
##
## Call:
## lm(formula = mtcars$mpg ~ mtcars$wt + mtcars$hp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -10.4460  -1.5824  -0.1857   1.1637   7.4123
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 33.87481    2.03436  16.651  < 2e-16 ***
## mtcars$wt   -2.60175    0.80511  -3.232  0.00306 **
## mtcars$hp   -0.04020    0.01149  -3.499  0.00153 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.3 on 29 degrees of freedom
## Multiple R-squared:  0.696,  Adjusted R-squared:  0.675
## F-statistic: 33.19 on 2 and 29 DF,  p-value: 3.178e-08
```

Both independent variables are significant again: as weight increases by 1 unit, holding horsepower constant, we predict a 2.60175 decrease in mpg; as horsepower increases by 1 unit, holding weight constant, we predict a 0.04020 decrease in mpg. This model explains 0.675 of the observed sample variation.

What about interaction?

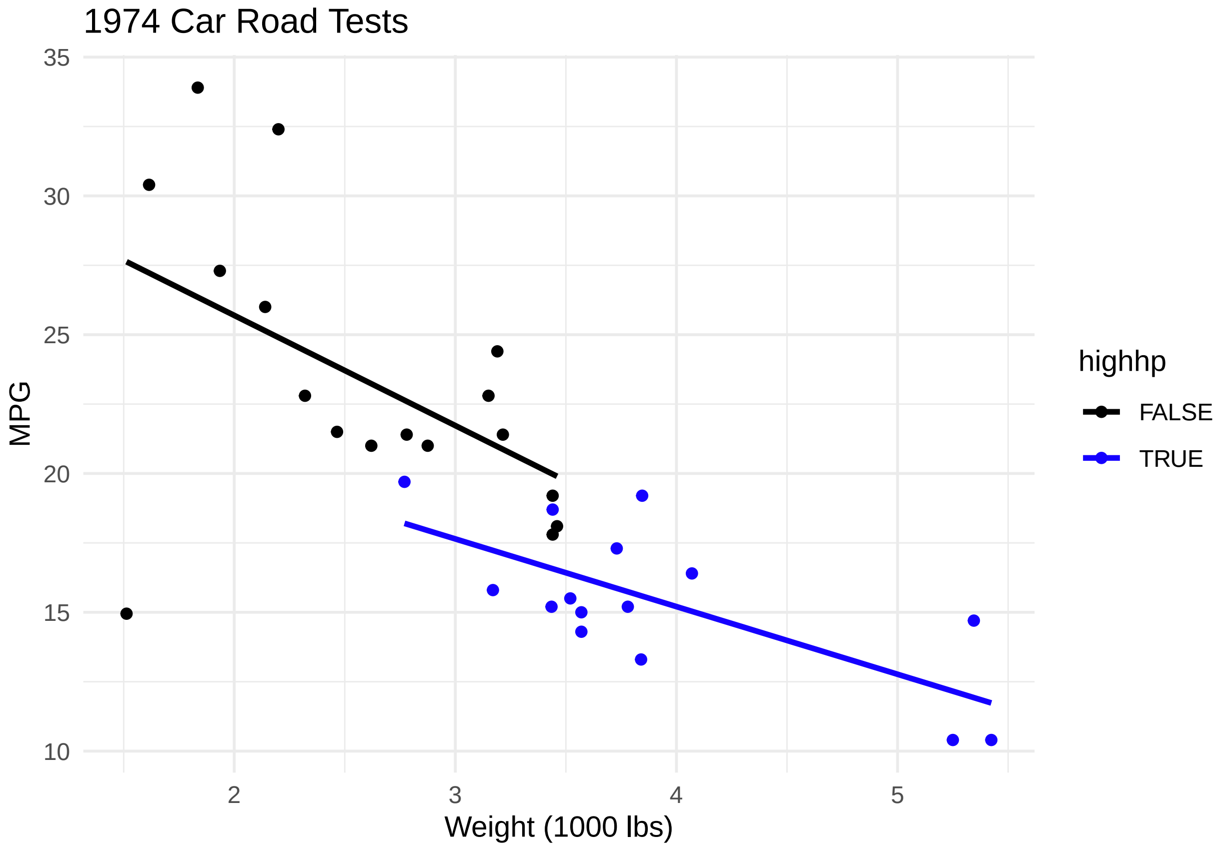
```
lm(mtcars$mpg~mtcars$wt*mtcars$hp) %>% summary
```

```
##
## Call:
## lm(formula = mtcars$mpg ~ mtcars$wt * mtcars$hp)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -11.0924  -1.2529  -0.0796   1.3987   5.8880
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)         44.08441    5.22733   8.433 3.59e-09 ***
## mtcars$wt           -6.12268    1.84103  -3.326  0.00247 **
## mtcars$hp           -0.11188    0.03581  -3.124  0.00412 **
## mtcars$wt:mtcars$hp  0.02260    0.01076   2.101  0.04480 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.121 on 28 degrees of freedom
## Multiple R-squared:  0.7374, Adjusted R-squared:  0.7092
## F-statistic:  26.2 on 3 and 28 DF,  p-value: 2.797e-08
```

The model with interaction explains a bit more variation than the model without; an increase in weight by 1 unit holding horsepower constant decreases the predicted mpg by 6.12269. Increasing horsepower by 1 holding weight constant decreases predicted mpg by 0.11188. The interaction effect is interpreted something like this: as horsepower increases by 1 unit the slope effect of weight on mpg increases by 0.02260, meaning it becomes less severe. We can see this if we split the data into high and low horsepower groups.

```
mtcars$highhp <- mtcars$hp >= mean(mtcars$hp)
mtcars %>% ggplot(aes(x = wt, y = mpg, group = highhp, color = highhp, fill = highhp)) + geom_point() +
```

```
## `geom_smooth()` using formula 'y ~ x'
```



The slope of the regression for the high hp group is less extreme than that for the low hp group (the coercion of continuous hp to discrete groups is common, and what happens under the hood; I picked 2 groups arbitrarily); now that I have a result that's interesting, I should validate my assumptions for the final model:

- Linearity: looks good at a glance + high model R-squared for linear regressions suggests that this is fine
- Outliers: The outlier I mentioned could be a valid observation (that is a light car that has low mpg) and if it is we want to include it in our sample; but if it's not something we normally expect to happen (ie model was recalled, had major mechanical issues, etc) then we might want to remove it. In this case I introduced the outlier right at the bound of what should be included, so we'll keep it but recognize it changes our model fit.

- Homoskedastic: I'll use the Breusch-Pagan test here, but there's tons of other heteroskedasticity tests that may be better based on your needs (White's test, Levene's, etc. see here)

```r
suppressWarnings(library(lmtest))
```

```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```

```r
bptest(mtcars$mpg~mtcars$wt*mtcars$hp)
```

```
##
##  studentized Breusch-Pagan test
##
## data:  mtcars$mpg ~ mtcars$wt * mtcars$hp
## BP = 6.0172, df = 3, p-value = 0.1108
```

The p-value of 0.1108 indicates that we can't reject the null hypothesis of homoskedasticity even at the 90% level, so we're ok with this assumption.

- Multicollinearity: let's check if our two independent variable are associated at all:

```r
cor.test(mtcars$wt, mtcars$hp)
```

```
##
##  Pearson's product-moment correlation
##
## data:  mtcars$wt and mtcars$hp
## t = 4.7957, df = 30, p-value = 4.146e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4025113 0.8192573
## sample estimates:
##       cor
## 0.6587479
```

It looks like they are linearly correlated with high significance, meaning that we should go back and adjust our model to account for this.

Note: We should have really done these assumptions before running any regressions, but I think this ordering is easier to follow.