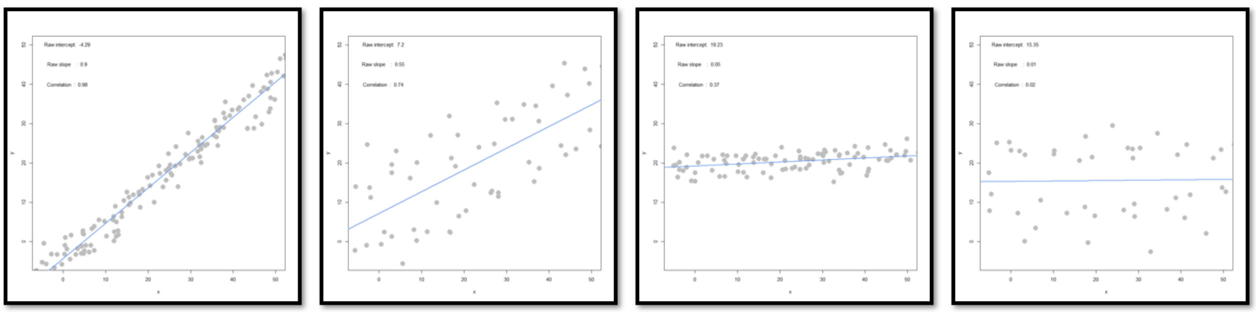
BACS HW10 - 109006234

Credit: 109006278

April 23th 2023

Problem 1

We will use the interactive_regression() function from CompStatsLib again – Windows users please make sure your desktop scaling is set to 100% and RStudio zoom is 100%; alternatively, run R from the Windows Command Prompt.



(a) Comparing scenarios 1 and 2, which do we expect to have a stronger R^2 ?
The R^2 is expected to be stronger for the scenario 1, because a strong R^2 is usually representated by a narrowly dispersed data.

(b) Comparing scenarios 3 and 4, which do we expect to have a stronger R^2 ?
For the same reasoning as (a), the R^2 is expected to be stronger for the scenario 3.

(c) Comparing scenarios 1 and 2, which do we expect has bigger/smaller SSE, SSR, and SST? (intuitively)
The SSE is expected to be smaller for the scenario 1, because there are less variability on the y axis. With small SSE, the greater the SSR and the smaller the SST.

(d) Comparing scenarios 3 and 4, which do we expect has bigger/smaller SSE, SSR, and SST? (intuitively)
For the same reasoning as (c), the SSE is expected to be smaller for the scenario 4. With small SSE, the greater the SSR and the smaller the SST.

Problem 2

(a) Use the lm() function to estimate the regression model

```
dataset <- read.csv(
    "G:/My Drive/111_2_BACS/HW10/programmer_salaries.txt", sep="\t")
salary_reg <- lm(dataset$Salary ~ dataset$Experience +
                            dataset$Score + dataset$Degree)
summary(salary_reg, data=dataset)
```

```
##
## Call:
## lm(formula = dataset$Salary ~ dataset$Experience + dataset$Score +
##     dataset$Degree)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8963 -1.7290 -0.3375  1.9699  5.0480
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)          7.9448     7.3808   1.076   0.2977
## dataset$Experience   1.1476     0.2976   3.856   0.0014 **
## dataset$Score        0.1969     0.0899   2.191   0.0436 *
## dataset$Degree       2.2804     1.9866   1.148   0.2679
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.396 on 16 degrees of freedom
## Multiple R-squared:  0.8468, Adjusted R-squared:  0.8181
## F-statistic: 29.48 on 3 and 16 DF,  p-value: 9.417e-07
```

```r
head(salary_reg$fitted.values, 5)
```

```
##        1        2        3        4        5
## 27.89626 37.95204 26.02901 32.11201 36.34251
```

```r
head(salary_reg$residuals, 5)
```

```
##          1          2          3          4          5
## -3.8962605  5.0479568 -2.3290112  2.1879860 -0.5425072
```

(b) Use only linear algebra and the geometric view of regression to estimate the regression yourself:
(i) Create an X matrix that has a first column of 1s followed by columns of the independent variables

```r
X_mat <- cbind(1, dataset$Experience, dataset$Score, dataset$Degree)
```

(ii) Create a y vector with the Salary values

```r
y <- dataset$Salary
```

(iii) Compute the beta_hat vector of estimated regression coefficients

```r
beta_hat <- solve(t(X_mat) %*% X_mat) %*% t(X_mat) %*% y
```

(iv) Compute a y_hat vector of estimated y hat values, and a res vector of residuals

```r
y_hat <- X_mat %*% beta_hat
res <- y - y_hat
head(y_hat, 5)
```

```
##          [,1]
## [1,] 27.89626
## [2,] 37.95204
## [3,] 26.02901
## [4,] 32.11201
## [5,] 36.34251
```

```r
head(res, 5)
```

```
##            [,1]
## [1,] -3.8962605
## [2,]  5.0479568
## [3,] -2.3290112
## [4,]  2.1879860
## [5,] -0.5425072
```

(v) Using only the results from (i) – (iv), compute SSR, SSE and SST

```r
SSR <- sum((y_hat-mean(y))^2)
SSE <- sum((y - y_hat)^2)
SST <- SSR + SSE
```

```
## [1] "SSR:  507.896013428808"

## [1] "SSE:  91.8894865712009"

## [1] "SST:  599.785500000009"
```

(c) Compute R^2 for in two ways, and confirm you get the same results (i) Use any combination of SSR, SSE, and SST

```r
r2_v1 <- SSR/SST
```

```
## [1] "R-squared:  0.846796085315168"
```

(ii) Use the squared correlation of vectors y and y

```r
r2_v2 <- (cor(y, y_hat))^2
```

```
## [1] "R-squared:  0.846796085315165"
```

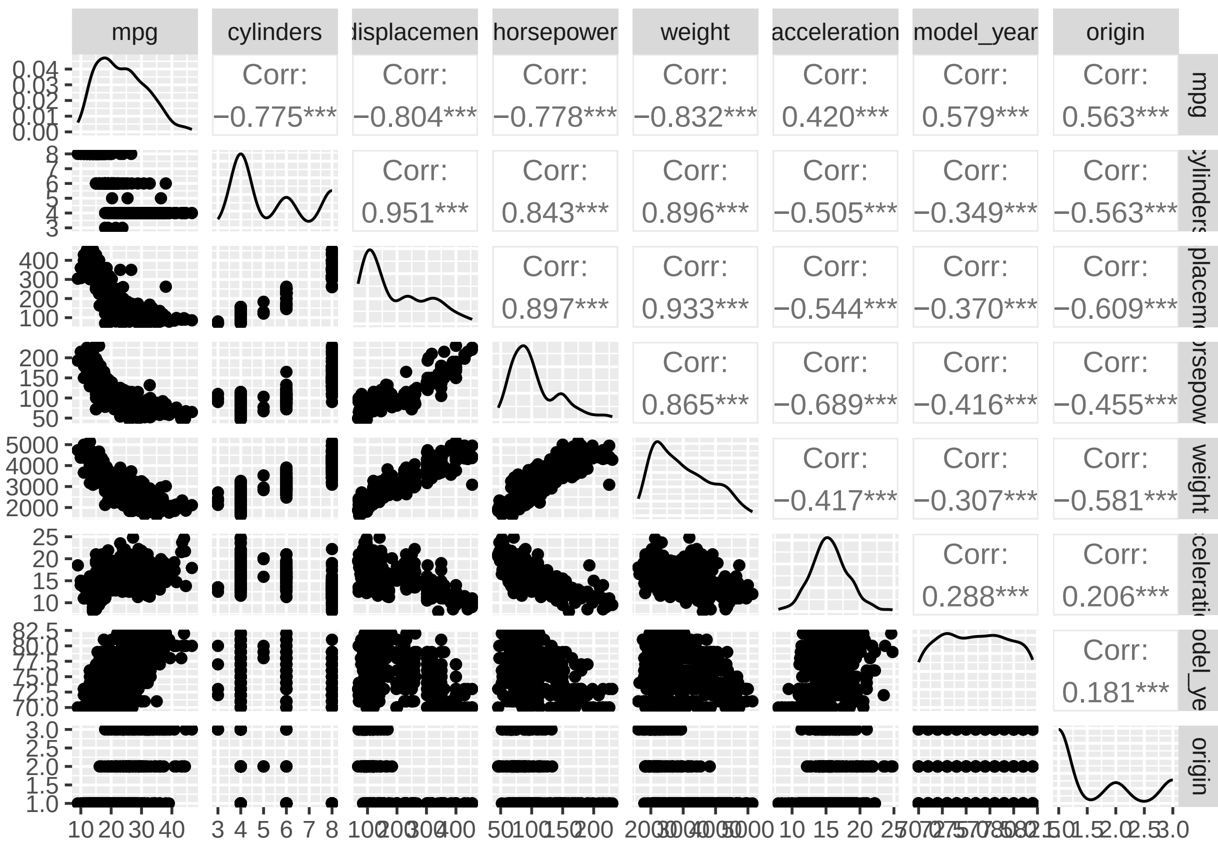
Problem 3

Take a look at the data set in file auto-data.txt. We are interested in explaining what kind of cars have higher fuel efficiency (mpg).

```r
auto <- read.table(
    "G:/My Drive/111_2_BACS/HW10/auto-data.txt",
    header=FALSE, na.strings = "?")
names(auto) <- c("mpg", "cylinders", "displacement",
                 "horsepower","weight","acceleration",
                 "model_year", "origin", "car_name")
```

(a) Let's first try exploring this data and problem:
(i) Visualize the data as you wish

```r
opts_chunk$set()
ggpairs(auto, columns = 1:8)
```



(ii) Report a correlation table of all variables, rounding to two decimal places

```r
round(cor(auto[1:8], use = "pairwise.complete.obs"), 2)
```

```
##                mpg cylinders displacement horsepower weight acceleration
## mpg           1.00     -0.78        -0.80      -0.78  -0.83         0.42
## cylinders    -0.78      1.00         0.95       0.84   0.90        -0.51
## displacement -0.80      0.95         1.00       0.90   0.93        -0.54
## horsepower   -0.78      0.84         0.90       1.00   0.86        -0.69
## weight       -0.83      0.90         0.93       0.86   1.00        -0.42
## acceleration  0.42     -0.51        -0.54      -0.69  -0.42         1.00
```

```
## model_year    0.58     -0.35         -0.37       -0.42  -0.31          0.29
## origin        0.56     -0.56         -0.61       -0.46  -0.58          0.21
##              model_year origin
## mpg                0.58   0.56
## cylinders         -0.35  -0.56
## displacement      -0.37  -0.61
## horsepower        -0.42  -0.46
## weight            -0.31  -0.58
## acceleration       0.29   0.21
## model_year         1.00   0.18
## origin             0.18   1.00
```

(iii) From the visualizations and correlations, which variables appear to relate to mpg?
From the correlation table, we can observe that `mpg` and `weight` strongly correlated to each other. Not only that, `mpg` is highly correlated to `cylinders`, `displacement` and `horsepower`.

(iv) Which relationships might not be linear?
`cylinders` vs. `origin` `model_year` vs. `weight`
`mpg` vs. `origin`
`model_year` vs. `acceleration`

(v) Are there any pairs of independent variables that are highly correlated (r > 0.7)
`cylinders` vs. `displacement`
`cylinders` vs. `horsepower` `cylinders` vs. `weight`
`displacement` vs. `horsepower`
`displacement` vs. `weight`
`horsepower` vs. `weight`

(b) Let's create a linear regression model where mpg is dependent upon all other suitable variables

```r
auto_v1 <- lm(auto$mpg ~
    auto$cylinders + auto$displacement +
    auto$horsepower + auto$weight +
    auto$acceleration + auto$model_year,
    factor(auto$origin))
summary(auto_v1)
```

```
##
## Call:
## lm(formula = auto$mpg ~ auto$cylinders + auto$displacement +
##     auto$horsepower + auto$weight + auto$acceleration + auto$model_year,
##     data = factor(auto$origin))
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.6927 -2.3864 -0.0801  2.0291 14.3607
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -1.454e+01  4.764e+00  -3.051  0.00244 **
## auto$cylinders    -3.299e-01  3.321e-01  -0.993  0.32122
## auto$displacement  7.678e-03  7.358e-03   1.044  0.29733
## auto$horsepower   -3.914e-04  1.384e-02  -0.028  0.97745
## auto$weight       -6.795e-03  6.700e-04 -10.141  < 2e-16 ***
```

```
## auto$acceleration  8.527e-02  1.020e-01   0.836  0.40383
## auto$model_year    7.534e-01  5.262e-02  14.318  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.435 on 385 degrees of freedom
##   (6 observations deleted due to missingness)
## Multiple R-squared:  0.8093, Adjusted R-squared:  0.8063
## F-statistic: 272.2 on 6 and 385 DF,  p-value: < 2.2e-16
```

(i) Which independent variables have a 'significant' relationship with mpg at 1% significance?
From the summary, `weight` and `model_year` have a significant relationship with `mpg` at 1% significance.

(ii) Looking at the coefficients, is it possible to determine which independent variables are the most effective at increasing mpg? If so, which ones, and if not, why not?
In my opinion, by just looking at the variables name, we can determine which independent variable will be most effective at increasing `mpg`. Not only that, from the summary, we can also determine the strongest effect by observing the coefficients.

(c) Let's try to resolve some of the issues with our regression model above.
(i) Create fully standardized regression results: are these slopes easier to compare?

```r
auto_std <- data.frame(scale(auto[1:8]))
auto_std_v2 <- lm(auto_std$mpg ~ auto_std$cylinders + auto_std$displacement +
    auto_std$horsepower + auto_std$weight + auto_std$acceleration +
    auto_std$model_year + auto_std$origin)
summary(auto_std_v2)
```

```
##
## Call:
## lm(formula = auto_std$mpg ~ auto_std$cylinders + auto_std$displacement +
##     auto_std$horsepower + auto_std$weight + auto_std$acceleration +
##     auto_std$model_year + auto_std$origin)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.22701 -0.27591 -0.01496  0.23912  1.67099
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -0.001748   0.021520  -0.081  0.93532
## auto_std$cylinders    -0.107374   0.070356  -1.526  0.12780
## auto_std$displacement  0.265420   0.100256   2.647  0.00844 **
## auto_std$horsepower   -0.083479   0.067896  -1.230  0.21963
## auto_std$weight       -0.701446   0.070648  -9.929  < 2e-16 ***
## auto_std$acceleration  0.028429   0.034875   0.815  0.41548
## auto_std$model_year    0.355179   0.024115  14.729  < 2e-16 ***
## auto_std$origin        0.146347   0.028542   5.127 4.67e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4258 on 384 degrees of freedom
##   (6 observations deleted due to missingness)
## Multiple R-squared:  0.8215, Adjusted R-squared:  0.8182
## F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

Yes. After the standardization, it is easier to observe the slopes of all variables.

(ii) Regress mpg over each nonsignificant independent variable, individually. Which ones become significant when we regress mpg over them individually?

```r
summary(lm(auto_std$mpg ~ auto_std$cylinders))
```

```
##
## Call:
## lm(formula = auto_std$mpg ~ auto_std$cylinders)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.82455 -0.43297 -0.08288  0.32674  2.29046
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.834e-15  3.169e-02    0.00        1
## auto_std$cylinders -7.754e-01  3.173e-02  -24.43   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6323 on 396 degrees of freedom
## Multiple R-squared:  0.6012, Adjusted R-squared:  0.6002
## F-statistic: 597.1 on 1 and 396 DF,  p-value: < 2.2e-16
```

```r
summary(lm(auto_std$mpg ~ auto_std$horsepower))
```

```
##
## Call:
## lm(formula = auto_std$mpg ~ auto_std$horsepower)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.73632 -0.41699 -0.04395  0.35351  2.16531
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -0.008784   0.031701  -0.277    0.782
## auto_std$horsepower -0.777334   0.031742 -24.489   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6277 on 390 degrees of freedom
##   (6 observations deleted due to missingness)
## Multiple R-squared:  0.6059, Adjusted R-squared:  0.6049
## F-statistic: 599.7 on 1 and 390 DF,  p-value: < 2.2e-16
```

```r
summary(lm(auto_std$mpg ~ auto_std$acceleration))
```

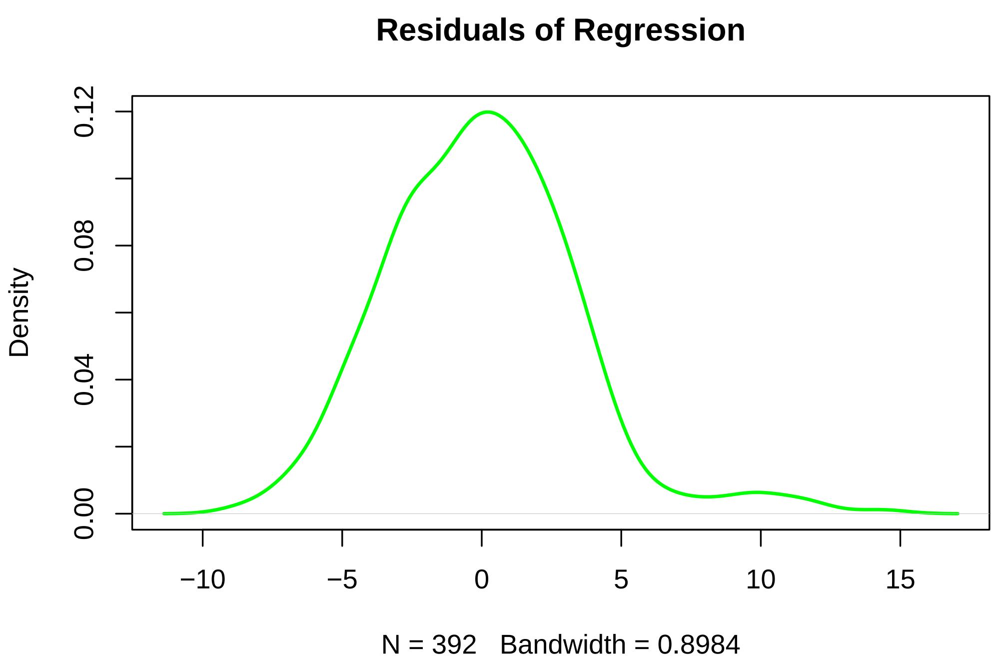
```
##
## Call:
## lm(formula = auto_std$mpg ~ auto_std$acceleration)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.3039 -0.7210 -0.1589  0.6087  2.9672
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)             3.004e-16  4.554e-02   0.000        1
## auto_std$acceleration 4.203e-01  4.560e-02   9.217   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.9085 on 396 degrees of freedom
## Multiple R-squared:  0.1766, Adjusted R-squared:  0.1746
## F-statistic: 84.96 on 1 and 396 DF,  p-value: < 2.2e-16
```

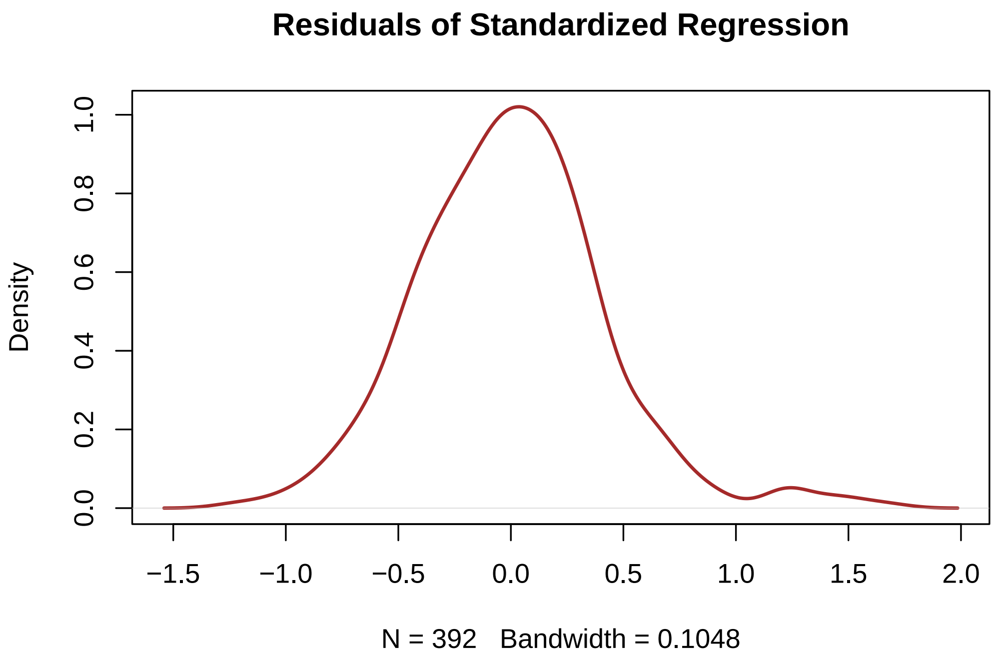
From the summaries showed above, we can see that all three variables are significant.

(iii) Plot the distribution of the residuals: are they normally distributed and centered around zero?

```r
plot(density(auto_v1$residuals),
     main="Residuals of Regression",
     col="green", lwd=2)
```




```r
plot(density(auto_std_v2$residuals),
     main="Residuals of Standardized Regression",
     col="brown", lwd=2)
```



For both plots, it is normally distributed and centered around zero.