```r
library(compstatslib)
library(data.table)
library(tidyr)
library(dplyr)
library(car)
```

```
## Warning:     'car'          R     4.3.3

## Warning:     'carData'          R     4.3.3
```

```r
library(lsa)
```

```
## Warning:     'lsa'          R     4.3.3
```

Question 1

```r
cars <- read.table("auto-data.txt", header=FALSE, na.strings = "?")

names(cars) <- c("mpg", "cylinders", "displacement", "horsepower", "weight",
                 "acceleration", "model_year", "origin", "car_name")

cars_log <- with(cars, data.frame(log(mpg), log(cylinders), log(displacement),
                                  log(horsepower), log(weight), log(acceleration),
                                  model_year, origin))
head(cars_log)
```

```
##   log.mpg. log.cylinders. log.displacement. log.horsepower. log.weight.
## 1 2.890372       2.079442          5.726848        4.867534    8.161660
## 2 2.708050       2.079442          5.857933        5.105945    8.214194
## 3 2.890372       2.079442          5.762051        5.010635    8.142063
## 4 2.772589       2.079442          5.717028        5.010635    8.141190
## 5 2.833213       2.079442          5.710427        4.941642    8.145840
## 6 2.708050       2.079442          6.061457        5.288267    8.375860
##   log.acceleration. model_year origin
## 1          2.484907         70      1
## 2          2.442347         70      1
## 3          2.397895         70      1
## 4          2.484907         70      1
## 5          2.351375         70      1
## 6          2.302585         70      1
```

(a)

```r
model <- lm(log.mpg. ~ factor(origin) + . - origin, data=cars_log)
summary(model)
```

```
##
## Call:
```

```
## lm(formula = log.mpg. ~ factor(origin) + . - origin, data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39727 -0.06880  0.00450  0.06356  0.38542
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.301938   0.361777  20.184  < 2e-16 ***
## factor(origin)2    0.050717   0.020920   2.424  0.01580 *
## factor(origin)3    0.047215   0.020622   2.290  0.02259 *
## log.cylinders.    -0.081915   0.061116  -1.340  0.18094
## log.displacement.  0.020387   0.058369   0.349  0.72707
## log.horsepower.   -0.284751   0.057945  -4.914 1.32e-06 ***
## log.weight.       -0.592955   0.085165  -6.962 1.46e-11 ***
## log.acceleration. -0.169673   0.059649  -2.845  0.00469 **
## model_year         0.030239   0.001771  17.078  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.113 on 383 degrees of freedom
##   (6                  )
## Multiple R-squared:  0.8919, Adjusted R-squared:  0.8897
## F-statistic:   395 on 8 and 383 DF,  p-value: < 2.2e-16
```

(i) Every variable except cylinders and displacement have a significant effect on log.mpg. at 10% significance.

(ii) Horsepower now is significant at alpha=10% and has an effect on mpg. By performing log transform on both sides of regression, we get more linear relationships. I guess the log transform of horsepower had a better effect than on other previously insignificant variables.

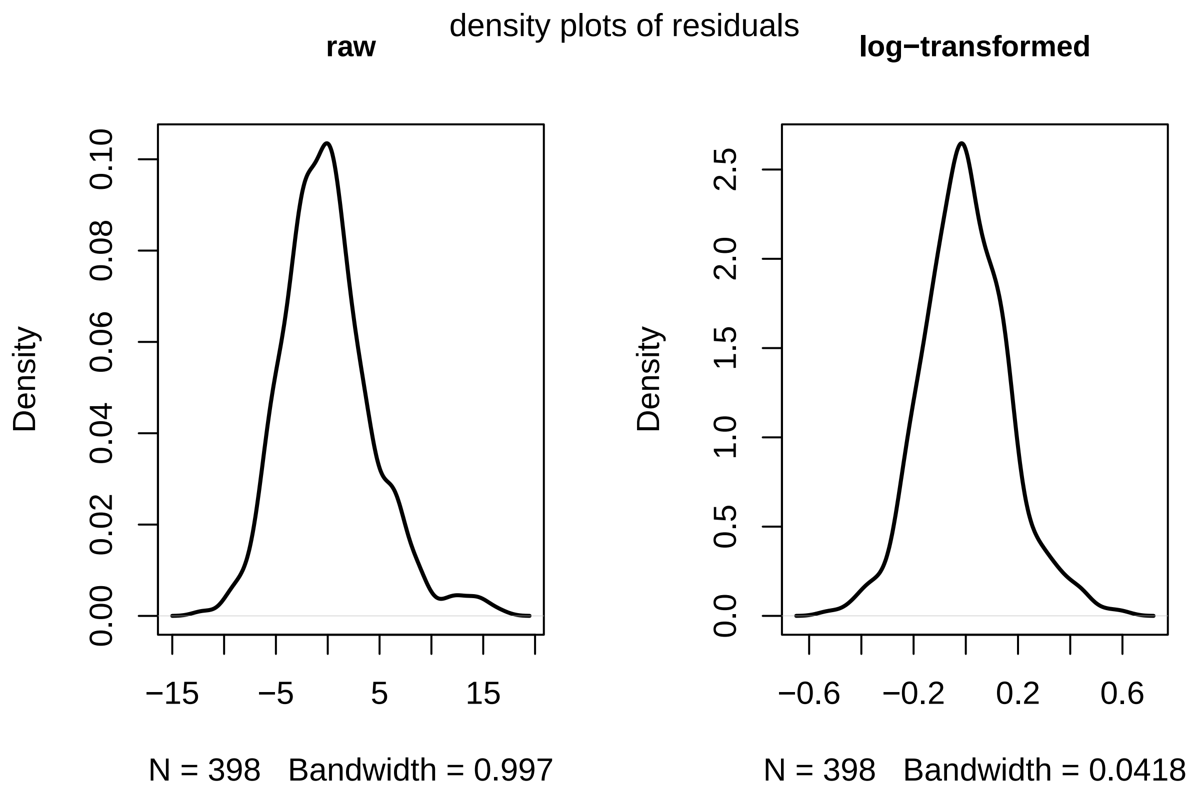
(iii) Cylinders and displacement still have insignificant effects on mpg. As I mentioned earlier, the possible reason could be that log transform wasn't that useful on those variables.

(b)

```r
regr_wt <- lm(cars$mpg ~ cars$weight)
regr_wt_log <- lm(cars_log$log.mpg. ~ cars_log$log.weight.)
```

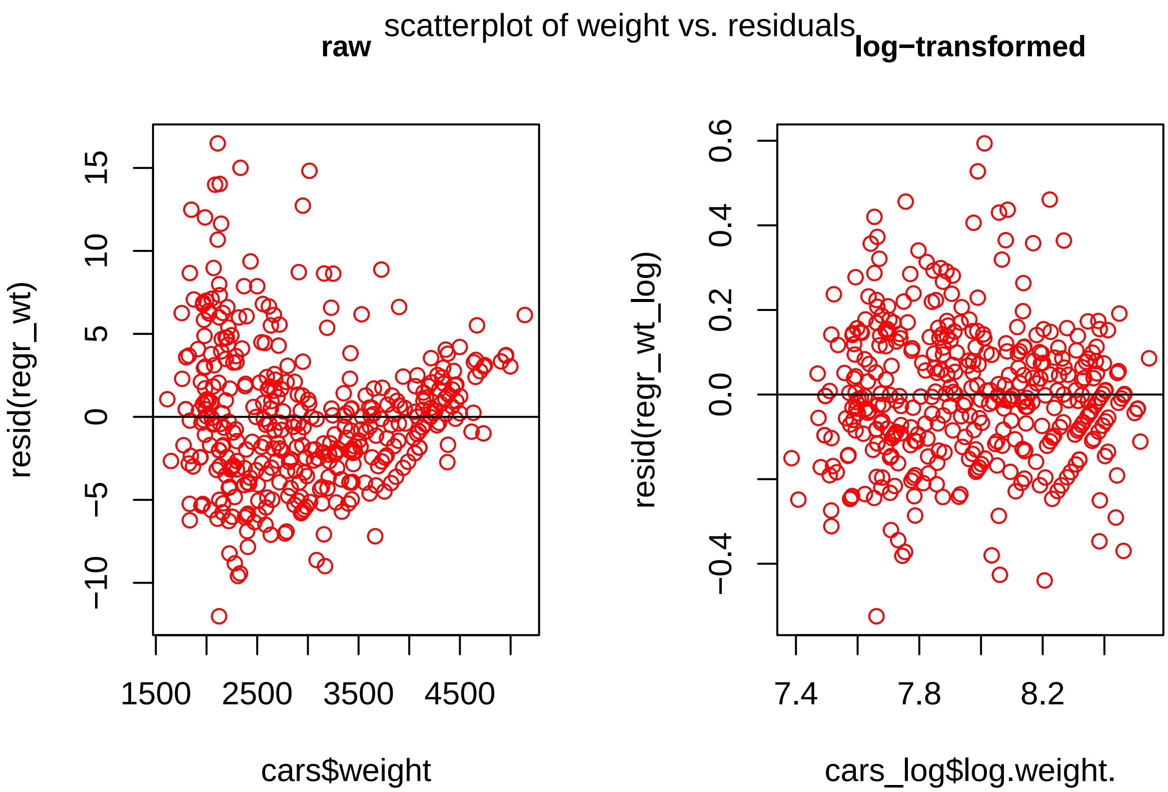
```r
par(mfrow=c(1,2))

plot(density(regr_wt$residuals), lwd=2, main='raw', cex.main=0.9)
plot(density(regr_wt_log$residuals), lwd=2, main='log-transformed', cex.main=0.9)
mtext('density plots of residuals', side=3, line=-2, outer=TRUE)
```




```
par(mfrow=c(1,2))

plot(cars$weight, resid(regr_wt), col="red", main='raw', cex.main=0.9)
abline(h=0)
plot(cars_log$log.weight., resid(regr_wt_log), col='red',
     main='log-transformed', cex.main=0.9)
abline(h=0)
mtext('scatterplot of weight vs. residuals', side=3, line=-2, outer=TRUE)
```



(iv) log-transformed residuals produce better and more normal distribution

```
summary(regr_wt_log)
```

```
##
## Call:
## lm(formula = cars_log$log.mpg. ~ cars_log$log.weight.)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.52408 -0.10441 -0.00805  0.10165  0.59384
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)            11.5219     0.2349   49.06   <2e-16 ***
## cars_log$log.weight.   -1.0583     0.0295  -35.87   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.165 on 396 degrees of freedom
## Multiple R-squared:  0.7647, Adjusted R-squared:  0.7641
## F-statistic:  1287 on 1 and 396 DF,  p-value: < 2.2e-16
```

(v) 1% change in log.weight leads to ~1% decrease in log.mpg

(vi)

```
conf_int <- confint(regr_wt_log)
conf_int
```

```
##                             2.5 %    97.5 %
## (Intercept)          11.060154 11.983659
## cars_log$log.weight. -1.116264 -1.000272
```

The 95% confidence interval for the slope of log.weight. vs log.mpg. is -1.1 to approximately -1.

Question 2

```
regr_log <- lm(log.mpg. ~ log.cylinders. + log.displacement. + log.horsepower. +
                              log.weight. + log.acceleration. + model_year +
                              factor(origin), data=cars_log)

summary(regr_log)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.cylinders. + log.displacement. +
##     log.horsepower. + log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.39727 -0.06880  0.00450  0.06356  0.38542
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.301938   0.361777  20.184  < 2e-16 ***
## log.cylinders.    -0.081915   0.061116  -1.340  0.18094
## log.displacement.  0.020387   0.058369   0.349  0.72707
## log.horsepower.   -0.284751   0.057945  -4.914 1.32e-06 ***
## log.weight.       -0.592955   0.085165  -6.962 1.46e-11 ***
## log.acceleration. -0.169673   0.059649  -2.845  0.00469 **
## model_year         0.030239   0.001771  17.078  < 2e-16 ***
## factor(origin)2    0.050717   0.020920   2.424  0.01580 *
## factor(origin)3    0.047215   0.020622   2.290  0.02259 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.113 on 383 degrees of freedom
##   (6                   )
## Multiple R-squared:  0.8919, Adjusted R-squared:  0.8897
## F-statistic:   395 on 8 and 383 DF,  p-value: < 2.2e-16
```

(a)

```r
weight_regr <- lm(log.weight. ~ log.cylinders. + log.displacement. + log.horsepower. +
                               log.acceleration. + model_year +
                              factor(origin), data=cars_log)
r2_weight <- summary(weight_regr)$r.squared
vif_weight <- 1 / (1 - r2_weight)
cat('VIF of log.weight is', vif_weight, sep=' ')
```

```
## VIF of log.weight is 17.57512
```

(b)

```r
vif(regr_log)
```

```
##                        GVIF Df GVIF^(1/(2*Df))
## log.cylinders.    10.456738  1        3.233688
## log.displacement. 29.625732  1        5.442952
## log.horsepower.   12.132057  1        3.483110
## log.weight.       17.575117  1        4.192269
## log.acceleration.  3.570357  1        1.889539
## model_year         1.303738  1        1.141814
## factor(origin)     2.656795  2        1.276702
```

```r
# eliminate log.displacement.
regr_log <- lm(log.mpg. ~ log.cylinders. + log.horsepower. +
                              log.weight. + log.acceleration. + model_year +
                              factor(origin), data=cars_log)

vif(regr_log)
```

```
##                        GVIF Df GVIF^(1/(2*Df))
## log.cylinders.     5.433107  1        2.330903
## log.horsepower.   12.114475  1        3.480585
## log.weight.       11.239741  1        3.352572
## log.acceleration.  3.327967  1        1.824272
## model_year         1.291741  1        1.136548
## factor(origin)     1.897608  2        1.173685
```

```r
# eliminate log.horsepower.
regr_log <- lm(log.mpg. ~ log.cylinders. + log.weight. +
               log.acceleration. + model_year +
               factor(origin), data=cars_log)

vif(regr_log)
```

```
##                       GVIF Df GVIF^(1/(2*Df))
## log.cylinders.    5.321090  1        2.306749
## log.weight.       4.788498  1        2.188264
## log.acceleration. 1.400111  1        1.183263
## model_year        1.201815  1        1.096273
## factor(origin)    1.792784  2        1.157130
```

```r
# eliminate log.cylinders.
regr_log <- lm(log.mpg. ~ log.weight. + log.acceleration. + model_year +
               factor(origin), data=cars_log)

vif(regr_log)
```

```
##                       GVIF Df GVIF^(1/(2*Df))
## log.weight.       1.926377  1        1.387940
## log.acceleration. 1.303005  1        1.141493
## model_year        1.167241  1        1.080389
## factor(origin)    1.692320  2        1.140567
```

```r
summary(regr_log)
```

```
##
## Call:
## lm(formula = log.mpg. ~ log.weight. + log.acceleration. + model_year +
##     factor(origin), data = cars_log)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.38275 -0.07032  0.00491  0.06470  0.39913
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)        7.431155   0.312248  23.799  < 2e-16 ***
## log.weight.       -0.876608   0.028697 -30.547  < 2e-16 ***
## log.acceleration.  0.051508   0.036652   1.405  0.16072
## model_year         0.032734   0.001696  19.306  < 2e-16 ***
## factor(origin)2    0.057991   0.017885   3.242  0.00129 **
## factor(origin)3    0.032333   0.018279   1.769  0.07770 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1156 on 392 degrees of freedom
## Multiple R-squared:  0.8856, Adjusted R-squared:  0.8841
## F-statistic: 606.8 on 5 and 392 DF,  p-value: < 2.2e-16
```

In the final regression model we have log.weight., log.acceleration., model_year, and origin as independent variables.

(c)

One variable that was previously significant is horsepower. A 1% change in horsepower led to a ~.28% decrease in log.mpg. I don't think by dropping horsepower we decreased the quality of the model, since log.weight. coef. increased.

(d)

If an independent variable has no correlation with other independent variables, its VIF score would be 1.

For VIF scores of 5 or higher, variables would need to be correlated at R-squared = 4/5 at least. To get VIF scores of 10 or higher, variables would need to be correlated at R-squared = 9/10 at least.

Question 3

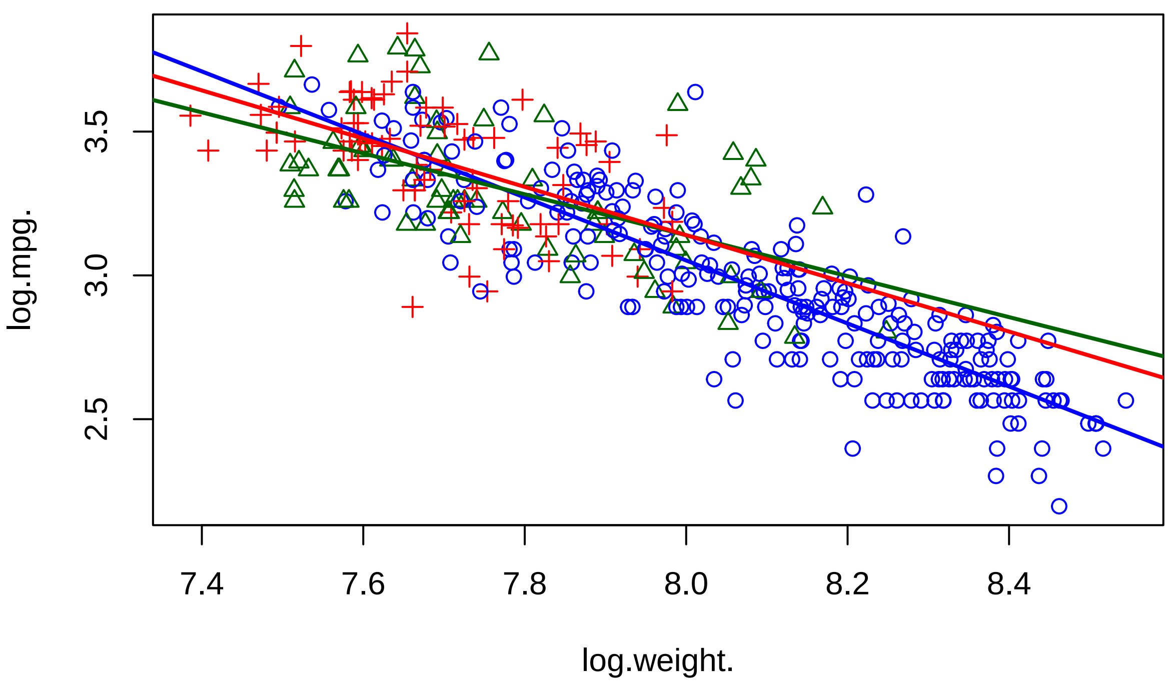
(a)

```r
origin_colors = c("blue", "darkgreen", "red")
with(cars_log, plot(log.weight., log.mpg., pch=origin, col=origin_colors[origin]))

cars_us <- subset(cars_log, origin==1)
wt_regr_us <- lm(log.mpg. ~ log.weight., data=cars_us)
abline(wt_regr_us, col=origin_colors[1], lwd=2)

cars_eu <- subset(cars_log, origin==2)
wt_regr_eu <- lm(cars_eu$log.mpg. ~ cars_eu$log.weight.)
abline(wt_regr_eu, col=origin_colors[2], lwd=2)

cars_jp <- subset(cars_log, origin==3)
wt_regr_jp <- lm(cars_jp$log.mpg. ~ cars_jp$log.weight.)
abline(wt_regr_jp, col=origin_colors[3], lwd=2)
```



(b)

I believe that cars from different origins appear to have similar in a sense weight vs. mpg relationships.