Big Data Methods for Economists Seminar Spring 2020
Exercises for Topic 1

Jingyan Yang and Marc Richter

Set Up

We first get ready to set up our environment for the following tasks.

```
rm(list = ls())

library(ISLR)  # contains Auto data set
```

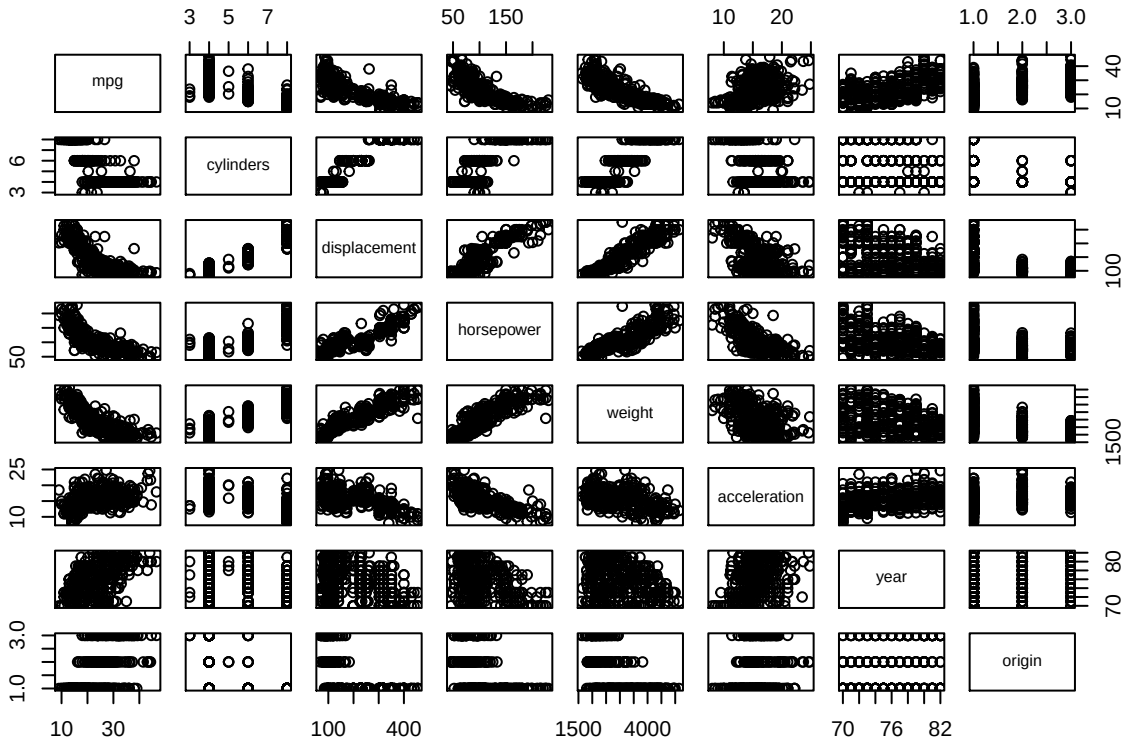
```
## Warning: package 'ISLR' was built under R version 3.6.2
```

```
library(MASS)  # for stepAIC formula

data(Auto)  # load data set
```

a)

We create a scatterplot plotting all variables against each other, except the qualitative variable name.



b)

The cor() function produces a correlation matrix for us.

```
cor(Auto[, !names(Auto) == "name"])  # correlation between all variables , excluding name
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
##              acceleration       year     origin
## mpg             0.4233285  0.5805410  0.5652088
## cylinders      -0.5046834 -0.3456474 -0.5689316
## displacement   -0.5438005 -0.3698552 -0.6145351
## horsepower     -0.6891955 -0.4163615 -0.4551715
## weight         -0.4168392 -0.3091199 -0.5850054
## acceleration    1.0000000  0.2903161  0.2127458
## year            0.2903161  1.0000000  0.1815277
## origin          0.2127458  0.1815277  1.0000000
```

c)

We fit a first mulitlinear model, regressing mpg on all other variables in the data set, excluding the qualitative variable "name".

```
Auto$origin <- as.factor(Auto$origin)  # transform origin variable from integer to factor - this way R t

model1 <- lm(mpg ~ . - name, Auto)

summary(model1)
```

```
##
## Call:
## lm(formula = mpg ~ . - name, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.0095 -2.0785 -0.0982  1.9856 13.3608
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.795e+01  4.677e+00  -3.839 0.000145 ***
## cylinders    -4.897e-01  3.212e-01  -1.524 0.128215
## displacement  2.398e-02  7.653e-03   3.133 0.001863 **
## horsepower   -1.818e-02  1.371e-02  -1.326 0.185488
## weight       -6.710e-03  6.551e-04 -10.243  < 2e-16 ***
## acceleration  7.910e-02  9.822e-02   0.805 0.421101
## year          7.770e-01  5.178e-02  15.005  < 2e-16 ***
## origin2       2.630e+00  5.664e-01   4.643 4.72e-06 ***
## origin3       2.853e+00  5.527e-01   5.162 3.93e-07 ***
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.307 on 383 degrees of freedom
## Multiple R-squared:  0.8242, Adjusted R-squared:  0.8205
## F-statistic: 224.5 on 8 and 383 DF,  p-value: < 2.2e-16
```

We see that some of the predictors are correlated with the response. Displacement, weight, year and origin seem to be related to mpg. For example, a model built one year later, is on average associated with a 0.77 higher mpg, everything else held constant. Cylinders, horsepower and accleration do not seem to be correlated. But we might have first doubts... Why would origin of a car have much to do with mpg? And is horsepower really not related to mpg?

We have to be careful with our estimates, the true relationship between our predictors and the response is

1) very certainly not linear
2) there might also be interactions included - predictors will be dependent on each other

We will now explore both of these issues.

d)

We can include interaction effects of the different variables - let's try it out with some variables.

```
model2 <- lm(mpg ~ . - name + cylinders:displacement + cylinders * origin + horsepower *
    acceleration, Auto)

summary(model2)
```

```
##
## Call:
## lm(formula = mpg ~ . - name + cylinders:displacement + cylinders *
##     origin + horsepower * acceleration, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.5119 -1.8103  0.0816  1.5380 12.2436
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)               -1.168e+01  6.267e+00  -1.863 0.063194 .
## cylinders                 -2.307e+00  5.516e-01  -4.182 3.60e-05 ***
## displacement              -7.872e-02  1.474e-02  -5.340 1.61e-07 ***
## horsepower                 3.663e-02  2.925e-02   1.252 0.211259
## weight                    -3.949e-03  7.201e-04  -5.483 7.64e-08 ***
## acceleration               5.816e-01  1.733e-01   3.356 0.000870 ***
## year                       7.569e-01  4.801e-02  15.765  < 2e-16 ***
## origin2                    1.414e+00  3.485e+00   0.406 0.685242
## origin3                   -4.784e+00  2.921e+00  -1.638 0.102268
## cylinders:displacement     1.207e-02  2.102e-03   5.742 1.92e-08 ***
## cylinders:origin2         -1.681e-01  8.163e-01  -0.206 0.836922
## cylinders:origin3          1.438e+00  6.804e-01   2.114 0.035187 *
## horsepower:acceleration   -6.962e-03  1.970e-03  -3.534 0.000459 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.013 on 379 degrees of freedom
```

```
## Multiple R-squared:  0.8555, Adjusted R-squared:  0.8509
## F-statistic:   187 on 12 and 379 DF,  p-value: < 2.2e-16
```

For this case, the cylinder-displacement and the horsepower interaction are statistically significant.

If we want to , we can include all the possible two-way effects through the following functions:

```
model3 <- lm(mpg ~ (. - name)^2, Auto)

summary(model3)
```

```
##
## Call:
## lm(formula = mpg ~ (. - name)^2, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6008 -1.2863  0.0813  1.2082 12.0382
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  4.401e+01  5.147e+01   0.855 0.393048
## cylinders                    3.302e+00  8.187e+00   0.403 0.686976
## displacement                -3.529e-01  1.974e-01  -1.788 0.074638 .
## horsepower                   5.312e-01  3.390e-01   1.567 0.117970
## weight                      -3.259e-03  1.820e-02  -0.179 0.857980
## acceleration                -6.048e+00  2.147e+00  -2.818 0.005109 **
## year                         4.833e-01  5.923e-01   0.816 0.415119
## origin2                     -3.517e+01  1.260e+01  -2.790 0.005547 **
## origin3                     -3.765e+01  1.426e+01  -2.640 0.008661 **
## cylinders:displacement      -6.316e-03  7.106e-03  -0.889 0.374707
## cylinders:horsepower         1.452e-02  2.457e-02   0.591 0.555109
## cylinders:weight             5.703e-04  9.044e-04   0.631 0.528709
## cylinders:acceleration       3.658e-01  1.671e-01   2.189 0.029261 *
## cylinders:year              -1.447e-01  9.652e-02  -1.499 0.134846
## cylinders:origin2           -7.210e-01  1.088e+00  -0.662 0.508100
## cylinders:origin3            1.226e+00  1.007e+00   1.217 0.224379
## displacement:horsepower     -5.407e-05  2.861e-04  -0.189 0.850212
## displacement:weight          2.659e-05  1.455e-05   1.828 0.068435 .
## displacement:acceleration   -2.547e-03  3.356e-03  -0.759 0.448415
## displacement:year            4.547e-03  2.446e-03   1.859 0.063842 .
## displacement:origin2        -3.364e-02  4.220e-02  -0.797 0.425902
## displacement:origin3         5.375e-02  4.145e-02   1.297 0.195527
## horsepower:weight           -3.407e-05  2.955e-05  -1.153 0.249743
## horsepower:acceleration     -3.445e-03  3.937e-03  -0.875 0.382122
## horsepower:year             -6.427e-03  3.891e-03  -1.652 0.099487 .
## horsepower:origin2          -4.869e-03  5.061e-02  -0.096 0.923408
## horsepower:origin3           2.289e-02  6.252e-02   0.366 0.714533
## weight:acceleration         -6.851e-05  2.385e-04  -0.287 0.774061
## weight:year                 -8.065e-05  2.184e-04  -0.369 0.712223
## weight:origin2               2.277e-03  2.685e-03   0.848 0.397037
## weight:origin3              -4.498e-03  3.481e-03  -1.292 0.197101
## acceleration:year            6.141e-02  2.547e-02   2.412 0.016390 *
## acceleration:origin2         9.234e-01  2.641e-01   3.496 0.000531 ***
## acceleration:origin3         7.159e-01  3.258e-01   2.198 0.028614 *
## year:origin2                 2.932e-01  1.444e-01   2.031 0.043005 *
```

```
## year:origin3               3.139e-01  1.483e-01   2.116 0.035034 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.628 on 356 degrees of freedom
## Multiple R-squared:  0.8967, Adjusted R-squared:  0.8866
## F-statistic: 88.34 on 35 and 356 DF,  p-value: < 2.2e-16
```

However, we see that this leads to nearly all predictors being statistically insignificant - we clearly added a lot of noise here! But we defined new predictors, and could use model selection to select the variables that have predictive power.

e)

We will now transform some variables. We can use the linear model to model non-linear effects - through transformation of the variables themselves!

Let's for example see what happens when we include the log of accleration and the square of horsepower. Having a look at the scatterplot we created in a) and the correlation of these variables with mpg shows that this might be a reasonable thing to do.

Note that we type -horsepower because the horsepower of polynomial 1 get's included automatically through the poly formula.

```
model4 <- lm(mpg ~ . - name - horsepower + log(acceleration) + poly(horsepower, 2),
    Auto)

summary(model4)
```

```
##
## Call:
## lm(formula = mpg ~ . - name - horsepower + log(acceleration) +
##     poly(horsepower, 2), data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.5818 -1.7843 -0.1401  1.6205 12.1677
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)           6.989e+00  1.420e+01   0.492   0.6228
## cylinders             3.764e-01  3.049e-01   1.234   0.2178
## displacement         -9.574e-03  7.923e-03  -1.208   0.2277
## weight               -3.089e-03  7.142e-04  -4.325 1.95e-05 ***
## acceleration          5.932e-01  4.842e-01   1.225   0.2213
## year                  7.448e-01  4.695e-02  15.865  < 2e-16 ***
## origin2               1.284e+00  5.376e-01   2.388   0.0174 *
## origin3               2.009e+00  5.079e-01   3.956 9.09e-05 ***
## log(acceleration)    -1.503e+01  7.772e+00  -1.933   0.0539 .
## poly(horsepower, 2)1 -5.421e+01  1.041e+01  -5.208 3.13e-07 ***
## poly(horsepower, 2)2  3.327e+01  4.008e+00   8.300 1.82e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.99 on 381 degrees of freedom
## Multiple R-squared:  0.857,  Adjusted R-squared:  0.8533
```

```
## F-statistic: 228.4 on 10 and 381 DF,  p-value: < 2.2e-16
```

The second degree polynomial of horsepower does seem to have a strong relationship, whereas the log of acceleration does not. Note that this again goes smoothly with what we could suspect from the scatterplot - a log correlation is not very clearly seen from the scatterplot matrix, but the polynomial relationship appears very strongly there.

f)

Now we want to find the best model based on the the simple model of using all the variables without interaction of transformations.

We use the stepAIC() function to perform one foward and one backward model selection calculating the AIC.

```
model0 <- lm(mpg ~ 1, Auto)  # define 'empty' model as starting point for forward selection

model_forward <- stepAIC(model0, scope = list(lower = "model0", upper = "model1"),
    direction = "forward")
```

```
## Start:  AIC=1611.93
## mpg ~ 1
##
##                Df Sum of Sq     RSS    AIC
## + weight        1   16497.8  7321.2 1151.5
## + displacement  1   15440.2  8378.8 1204.4
## + horsepower    1   14433.1  9385.9 1248.9
## + cylinders     1   14403.1  9415.9 1250.1
## + year          1    8027.7 15791.3 1452.8
## + origin        2    7904.3 15914.7 1457.9
## + acceleration  1    4268.5 19550.5 1536.5
## <none>                      23819.0 1611.9
##
## Step:  AIC=1151.49
## mpg ~ weight
##
##                Df Sum of Sq    RSS     AIC
## + year          1   2752.28 4569.0  968.66
## + horsepower    1    327.39 6993.8 1135.56
## + origin        2    224.14 7097.1 1143.30
## + acceleration  1    168.34 7152.9 1144.37
## + displacement  1    150.93 7170.3 1145.33
## + cylinders     1    115.12 7206.1 1147.28
## <none>                      7321.2 1151.49
##
## Step:  AIC=968.66
## mpg ~ weight + year
##
##                Df Sum of Sq    RSS    AIC
## + origin        2   258.543 4310.4 949.83
## <none>                      4569.0 968.66
## + acceleration  1    10.450 4558.5 969.77
## + cylinders     1     4.958 4564.0 970.24
## + horsepower    1     3.302 4565.7 970.38
## + displacement  1     0.042 4568.9 970.66
##
## Step:  AIC=949.83
```

```
## mpg ~ weight + year + origin
##
##                Df Sum of Sq    RSS    AIC
## + displacement  1    38.445 4272.0 948.32
## <none>                      4310.4 949.83
## + acceleration  1     9.525 4300.9 950.96
## + horsepower    1     9.180 4301.2 950.99
## + cylinders     1     1.119 4309.3 951.73
##
## Step:  AIC=948.32
## mpg ~ weight + year + origin + displacement
##
##                Df Sum of Sq    RSS    AIC
## + horsepower    1    50.631 4221.3 945.64
## + acceleration  1    44.737 4227.2 946.19
## <none>                      4272.0 948.32
## + cylinders     1    18.680 4253.3 948.60
##
## Step:  AIC=945.64
## mpg ~ weight + year + origin + displacement + horsepower
##
##                Df Sum of Sq    RSS    AIC
## + cylinders     1   26.8505 4194.5 945.14
## <none>                      4221.3 945.64
## + acceleration  1    8.5331 4212.8 946.85
##
## Step:  AIC=945.14
## mpg ~ weight + year + origin + displacement + horsepower + cylinders
##
##                Df Sum of Sq    RSS    AIC
## <none>                      4194.5 945.14
## + acceleration  1    7.0916 4187.4 946.48
```

```
model_backward <- stepAIC(model1, direction = "backward")
```

```
## Start:  AIC=946.48
## mpg ~ (cylinders + displacement + horsepower + weight + acceleration +
##     year + origin + name) - name
##
##                Df Sum of Sq    RSS     AIC
## - acceleration  1      7.09 4194.5  945.14
## - horsepower    1     19.24 4206.6  946.28
## <none>                      4187.4  946.48
## - cylinders     1     25.41 4212.8  946.85
## - displacement  1    107.32 4294.7  954.40
## - origin        2    355.96 4543.3  974.46
## - weight        1   1147.04 5334.4 1039.39
## - year          1   2461.64 6649.0 1125.74
##
## Step:  AIC=945.14
## mpg ~ cylinders + displacement + horsepower + weight + year +
##     origin
##
##                Df Sum of Sq    RSS     AIC
## <none>                      4194.5  945.14
```

```
## - cylinders     1     26.85 4221.3  945.64
## - horsepower    1     58.80 4253.3  948.60
## - displacement  1    102.96 4297.4  952.65
## - origin        2    357.11 4551.6  973.17
## - weight        1   1372.52 5567.0 1054.11
## - year          1   2455.71 6650.2 1123.81
```

```
summary(model_forward)
```

```
##
## Call:
## lm(formula = mpg ~ weight + year + origin + displacement + horsepower +
##     cylinders, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.1754 -2.1139 -0.0863  1.9711 13.4207
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.633e+01  4.219e+00  -3.871 0.000127 ***
## weight       -6.460e-03  5.763e-04 -11.209  < 2e-16 ***
## year          7.739e-01  5.161e-02  14.994  < 2e-16 ***
## origin2       2.635e+00  5.661e-01   4.654 4.50e-06 ***
## origin3       2.857e+00  5.525e-01   5.172 3.74e-07 ***
## displacement  2.337e-02  7.613e-03   3.070 0.002292 **
## horsepower   -2.500e-02  1.078e-02  -2.320 0.020855 *
## cylinders    -5.028e-01  3.207e-01  -1.568 0.117742
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.305 on 384 degrees of freedom
## Multiple R-squared:  0.8239, Adjusted R-squared:  0.8207
## F-statistic: 256.7 on 7 and 384 DF,  p-value: < 2.2e-16
```

```
summary(model_backward)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + displacement + horsepower + weight +
##     year + origin, data = Auto)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.1754 -2.1139 -0.0863  1.9711 13.4207
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.633e+01  4.219e+00  -3.871 0.000127 ***
## cylinders    -5.028e-01  3.207e-01  -1.568 0.117742
## displacement  2.337e-02  7.613e-03   3.070 0.002292 **
## horsepower   -2.500e-02  1.078e-02  -2.320 0.020855 *
## weight       -6.460e-03  5.763e-04 -11.209  < 2e-16 ***
## year          7.739e-01  5.161e-02  14.994  < 2e-16 ***
## origin2       2.635e+00  5.661e-01   4.654 4.50e-06 ***
```

```
## origin3       2.857e+00  5.525e-01   5.172 3.74e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.305 on 384 degrees of freedom
## Multiple R-squared:  0.8239, Adjusted R-squared:  0.8207
## F-statistic: 256.7 on 7 and 384 DF,  p-value: < 2.2e-16
```

Both models come to the same conclusion: the only variable not to include should be acceleration ! Note that there is still predictors in the model (like cylinders) which are not statistically significant in the estimation.