ViF

Gustavo Cepparo and Milica Cudina

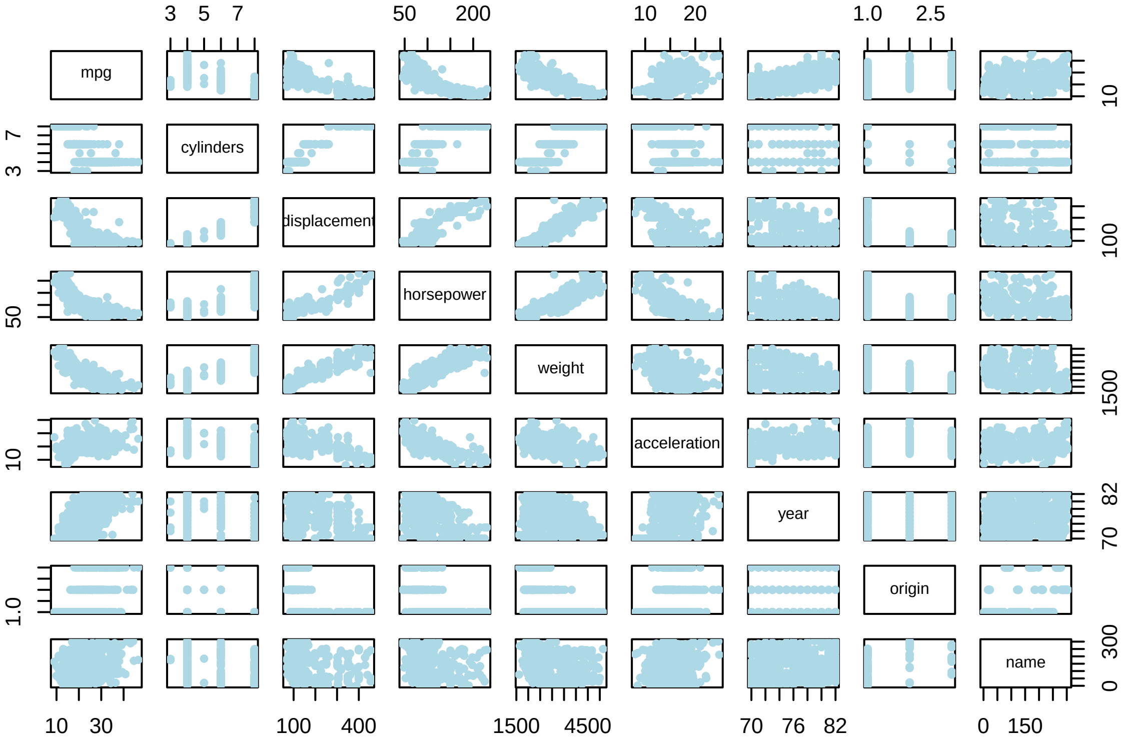
Car efficiency [revisited]

First, let's import the textbook's library.

```
library(ISLR2)
attach(Auto)
```

Let's do some exploratory analysis.

```
pairs(Auto,
      pch=20, col="lightblue")
```



```
round(cor(Auto[,-c(8,9)]),4)
```

```
##                  mpg cylinders displacement horsepower  weight acceleration
## mpg           1.0000   -0.7776      -0.8051    -0.7784 -0.8322       0.4233
## cylinders    -0.7776    1.0000       0.9508     0.8430  0.8975      -0.5047
## displacement -0.8051    0.9508       1.0000     0.8973  0.9330      -0.5438
## horsepower   -0.7784    0.8430       0.8973     1.0000  0.8645      -0.6892
## weight       -0.8322    0.8975       0.9330     0.8645  1.0000      -0.4168
## acceleration  0.4233   -0.5047      -0.5438    -0.6892 -0.4168       1.0000
## year          0.5805   -0.3456      -0.3699    -0.4164 -0.3091       0.2903
##                 year
## mpg           0.5805
## cylinders    -0.3456
## displacement -0.3699
## horsepower   -0.4164
## weight       -0.3091
## acceleration  0.2903
## year          1.0000
```

We notice some pretty sizable correlations. What about a multiple linear regression?

```
lm.fit=lm(mpg~cylinders+horsepower+weight,data=Auto)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = mpg ~ cylinders + horsepower + weight, data = Auto)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.5260  -2.7955  -0.3559   2.2567  16.3209
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 45.7368172  0.7959566  57.461  < 2e-16 ***
## cylinders   -0.3889745  0.2988302  -1.302 0.193806
## horsepower  -0.0427277  0.0116196  -3.677 0.000269 ***
## weight      -0.0052723  0.0006424  -8.208 3.37e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.236 on 388 degrees of freedom
## Multiple R-squared:  0.7077, Adjusted R-squared:  0.7054
## F-statistic: 313.1 on 3 and 388 DF,  p-value: < 2.2e-16
```

We should import the `car` library (nothing to do with vehicles; it's short for *Companion to Applied Regression.*).

```
library(car)
```

```
## Loading required package: carData
```

```
vif(lm.fit)
```

```
##  cylinders horsepower     weight
##   5.660847   4.358007   6.485732
```

The book says: *“. . . a VIF value that exceeds 5 or 10 indicates a problematic amount of collinearity. . . ”

Seat Position

Let's consider another data set. This one is from the `faraway` library.

```
#install.packages("faraway")
library(faraway)
```

```
##
## Attaching package: 'faraway'
```

```
## The following objects are masked from 'package:car':
##
```

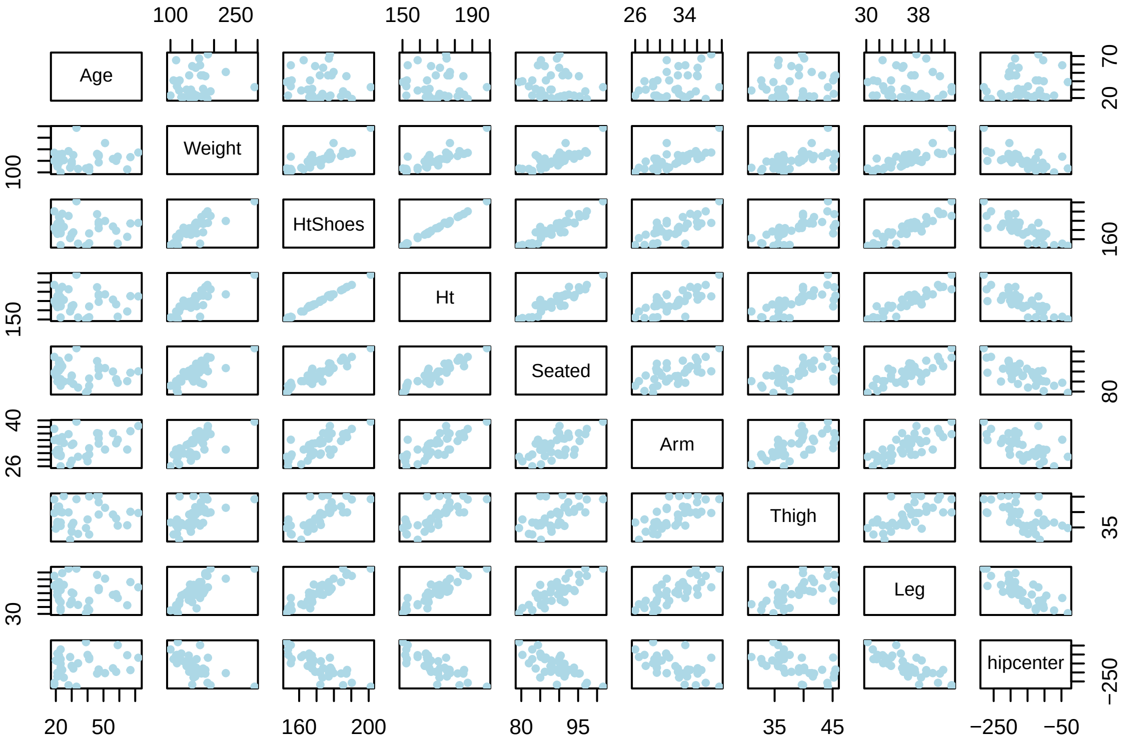
```
##     logit, vif
```

The data set `seatpos` is used to predict the carseat position (`hipcenter`) based on biometric data of the driver.

```
data(seatpos)
```

When we look at the documentation, we see that one of the variables is `HtShoes`, i.e., height in shoes, and another is `Ht`, i.e., height barefoot. These are bound to be incredibly correlated. Similarly, there is the `Seated`, i.e., the seated height, `Weight`, and others that should be heavily positively correlated. Let's do some exploratory data analysis:

```
pairs(seatpos,
      pch=20, col="lightblue")
```



```
round(cor(seatpos),4)
```

```
##                Age  Weight HtShoes      Ht  Seated     Arm   Thigh     Leg
## Age         1.0000  0.0807 -0.0793 -0.0901 -0.1702  0.3595  0.0913 -0.0423
## Weight      0.0807  1.0000  0.8282  0.8285  0.7756  0.6976  0.5726  0.7843
## HtShoes    -0.0793  0.8282  1.0000  0.9981  0.9297  0.7520  0.7249  0.9084
## Ht         -0.0901  0.8285  0.9981  1.0000  0.9282  0.7521  0.7350  0.9098
## Seated     -0.1702  0.7756  0.9297  0.9282  1.0000  0.6252  0.6071  0.8119
## Arm         0.3595  0.6976  0.7520  0.7521  0.6252  1.0000  0.6711  0.7538
## Thigh       0.0913  0.5726  0.7249  0.7350  0.6071  0.6711  1.0000  0.6495
## Leg        -0.0423  0.7843  0.9084  0.9098  0.8119  0.7538  0.6495  1.0000
## hipcenter   0.2052 -0.6403 -0.7966 -0.7989 -0.7313 -0.5851 -0.5912 -0.7872
##           hipcenter
## Age          0.2052
## Weight      -0.6403
## HtShoes     -0.7966
## Ht          -0.7989
## Seated      -0.7313
## Arm         -0.5851
## Thigh       -0.5912
## Leg         -0.7872
## hipcenter    1.0000
```

`Age` appears to be the only predictor not linked with other predictors.

Let's try a multiple linear regression.

```
lm.fit=lm(hipcenter~.,data=seatpos)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = hipcenter ~ ., data = seatpos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -73.827 -22.833  -3.678  25.017  62.337
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 436.43213  166.57162   2.620   0.0138 *
## Age           0.77572    0.57033   1.360   0.1843
## Weight        0.02631    0.33097   0.080   0.9372
## HtShoes      -2.69241    9.75304  -0.276   0.7845
## Ht            0.60134   10.12987   0.059   0.9531
## Seated        0.53375    3.76189   0.142   0.8882
## Arm          -1.32807    3.90020  -0.341   0.7359
## Thigh        -1.14312    2.66002  -0.430   0.6706
## Leg          -6.43905    4.71386  -1.366   0.1824
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.72 on 29 degrees of freedom
## Multiple R-squared:  0.6866, Adjusted R-squared:  0.6001
## F-statistic:  7.94 on 8 and 29 DF,  p-value: 0.00001306
```

As anticipated, we get much nonsense. So, let's check out the variance inflation factors.

```
vif(lm.fit)
```

```
##        Age     Weight    HtShoes         Ht     Seated        Arm      Thigh
##   1.997931   3.647030 307.429378 333.137832   8.951054   4.496368   2.762886
##        Leg
##   6.694291
```

It seems that we really should take a pick between height in shoes and height without.

```
lm.fit.1=lm(hipcenter~. - Ht,data=seatpos)
summary(lm.fit.1)
```

```
##
## Call:
## lm(formula = hipcenter ~ . - Ht, data = seatpos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -74.107 -22.467  -4.207  25.106  62.225
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 436.84207  163.64104   2.670   0.0121 *
## Age           0.76574    0.53590   1.429   0.1634
## Weight        0.02897    0.32244   0.090   0.9290
## HtShoes      -2.13409    2.53896  -0.841   0.4073
## Seated        0.54959    3.68958   0.149   0.8826
## Arm          -1.30087    3.80833  -0.342   0.7350
## Thigh        -1.09039    2.46534  -0.442   0.6615
## Leg          -6.40612    4.60272  -1.392   0.1742
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.09 on 30 degrees of freedom
## Multiple R-squared:  0.6865, Adjusted R-squared:  0.6134
## F-statistic: 9.385 on 7 and 30 DF,  p-value: 0.000004014
```

```
vif(lm.fit.1)
```

```
##       Age    Weight   HtShoes    Seated       Arm     Thigh       Leg
##  1.824607  3.580351 21.550054  8.906032  4.434329  2.454805  6.601632
```

What if we get rid of height in shoes as well?

```
lm.fit.2=lm(hipcenter~. - Ht-HtShoes,data=seatpos)
summary(lm.fit.2)
```

```
##
## Call:
## lm(formula = hipcenter ~ . - Ht - HtShoes, data = seatpos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -68.296 -23.340  -5.672  24.183  74.065
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 409.00851  159.49517   2.564   0.0154 *
## Age           0.83110    0.52771   1.575   0.1254
## Weight       -0.03251    0.31254  -0.104   0.9178
## Seated       -1.73576    2.48225  -0.699   0.4896
## Arm          -2.00541    3.69731  -0.542   0.5914
## Thigh        -1.91970    2.24858  -0.854   0.3998
## Leg          -8.40876    3.91939  -2.145   0.0399 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.91 on 31 degrees of freedom
## Multiple R-squared:  0.6791, Adjusted R-squared:  0.617
## F-statistic: 10.94 on 6 and 31 DF,  p-value: 0.000001571
```

```
vif(lm.fit.2)
```

```
##      Age   Weight   Seated      Arm    Thigh      Leg
## 1.786192 3.396124 4.069626 4.219519 2.061632 4.832701
```

We have not sacrificed anything in terms of R^2 by eliminating the two height variables. Just the `Leg` is still statistically significant with `Age` in the surprising second place.

Just for laughs, how about a simple linear regression on `Leg`?

```r
lm.fit.s<-lm(hipcenter ~ Leg, data=seatpos)
summary(lm.fit.s)
```

```
##
## Call:
## lm(formula = hipcenter ~ Leg, data = seatpos)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -78.10 -26.11  -1.86  18.54  94.42
##
## Coefficients:
##             Estimate Std. Error t value      Pr(>|t|)
## (Intercept)  335.351     65.601   5.112 0.00001066600 ***
## Leg          -13.795      1.801  -7.658 0.00000000459 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 37.29 on 36 degrees of freedom
## Multiple R-squared:  0.6196, Adjusted R-squared:  0.6091
## F-statistic: 58.65 on 1 and 36 DF,  p-value: 0.000000004587
```

So, more predictors are not always better?