Multicollinearity

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \qquad \epsilon_i \sim N(0, \sigma^2)$$

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\boldsymbol{Y} \qquad E[\hat{\boldsymbol{\beta}}] = \boldsymbol{\beta} \qquad Var[\hat{\boldsymbol{\beta}}] = \boldsymbol{\sigma}^2\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}$$

Let's create a dataset where one of the predictors, x_3, is a linear combination of the other predictors.

```r
x1 <- rnorm(100,80,10)
x2 <- rnorm(100,70,5)
x3 <- 2*x1 + 4*x2 + 3
y <- 3 + x1 + x2 + rnorm(100)
```

What happens when we fit a regression model in `R`?

```r
fit <- lm(y ~ x1 + x2 + x3)
summary(fit)
```

```
##
## Call:
## lm(formula = y ~ x1 + x2 + x3)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.50553 -0.84309 -0.03273  0.77138  2.27616
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.08252    1.78761   0.606    0.546
## x1           1.00166    0.01113  90.035   <2e-16 ***
## x2           1.02398    0.02177  47.030   <2e-16 ***
## x3                NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.088 on 97 degrees of freedom
## Multiple R-squared:  0.9906, Adjusted R-squared:  0.9904
## F-statistic:  5133 on 2 and 97 DF,  p-value: < 2.2e-16
```

We see that `R` simply decides to exclude the variable. Why is this happening?

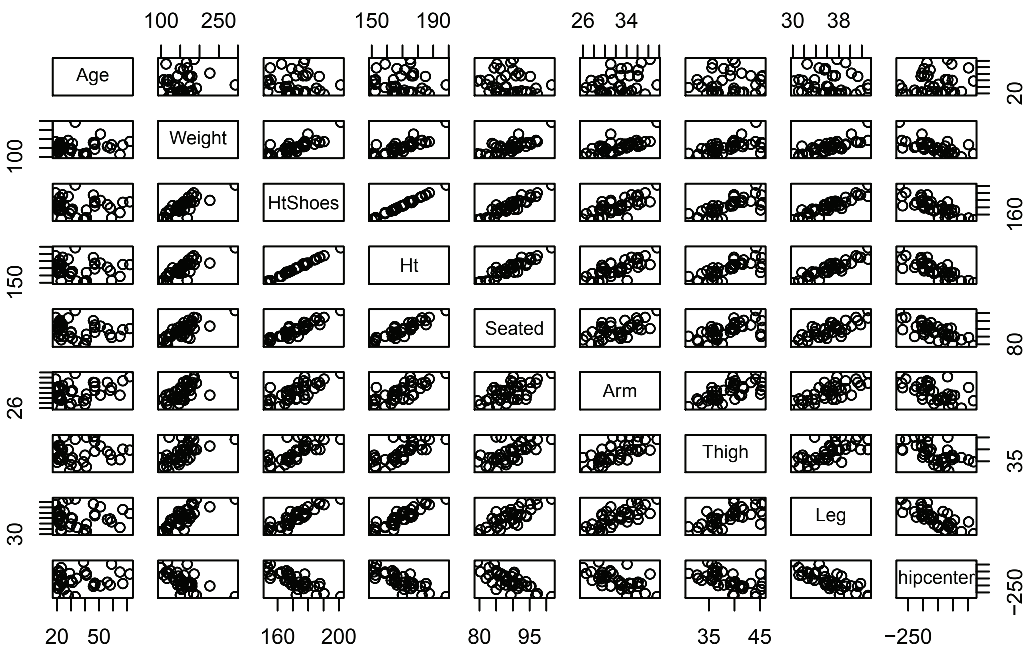
```r
x0 <- rep(1,100)
X <- cbind(x0,x1,x2,x3)
solve(t(X) %*% X)
```

If we attempt to find $\hat{\boldsymbol{\beta}}$ using $\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}$, we see that this is not possible, due to the fact that the columns of $\boldsymbol{X}$ are linearly dependent.

When this happens, we say there is exact collinearity. Exact collinearity is an extreme example of multicollinearity, which occurs in multiple regression when predictor variables are highly correlated.

Looking at the `seatpos` dataset, we can see an example of this. For example, we expect a person's height to be highly correlated to their height when wearing shoes.

```r
library(faraway)
data(seatpos)
pairs(seatpos)
```




```r
round(cor(seatpos),2)
```

```
##             Age Weight HtShoes    Ht Seated   Arm Thigh   Leg hipcenter
## Age        1.00   0.08   -0.08 -0.09  -0.17  0.36  0.09 -0.04      0.21
## Weight     0.08   1.00    0.83  0.83   0.78  0.70  0.57  0.78     -0.64
## HtShoes   -0.08   0.83    1.00  1.00   0.93  0.75  0.72  0.91     -0.80
## Ht        -0.09   0.83    1.00  1.00   0.93  0.75  0.73  0.91     -0.80
## Seated    -0.17   0.78    0.93  0.93   1.00  0.63  0.61  0.81     -0.73
## Arm        0.36   0.70    0.75  0.75   0.63  1.00  0.67  0.75     -0.59
## Thigh      0.09   0.57    0.72  0.73   0.61  0.67  1.00  0.65     -0.59
## Leg       -0.04   0.78    0.91  0.91   0.81  0.75  0.65  1.00     -0.79
## hipcenter  0.21  -0.64   -0.80 -0.80  -0.73 -0.59 -0.59 -0.79      1.00
```

Unlike exact collinearity, here we can still fit a model with all of the predictors, but what effect does this have?

```r
fit <- lm(hipcenter ~ ., data = seatpos)
summary(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: 1.306e-05
```

One of the first things we should notice is that the F-test for the regression tells us that the regression is significant, however each individual predictor is not. Another interesting result is the opposite signs of the coefficients for `Ht` and `HtShoes`. This should seem rather counter-intuitive.

This happens as a result of the predictors being highly correlated. For example, the `HtShoe` variable explains a large amount of the variation in `Ht`. When they are both in the model, their effects on the response are lessened individually, but together they still explain a large portion of the variation of `hipcenter`.

Define R_j^2 to be the proportion of observed variation in the jth predictor explained by the other predictors. In other words R_j^2 is the multiple R-Squared for the regression of x_j on each of the other predictors.

```r
fitHtShoes <- lm(HtShoes ~ . -hipcenter, data = seatpos)
summary(fitHtShoes)
```

```
##
## Call:
## lm(formula = HtShoes ~ . - hipcenter, data = seatpos)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.40812 -0.35387  0.07312  0.31358  1.55182
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  0.231451   3.117889   0.074    0.941
## Age          0.014462   0.010345   1.398    0.172
## Weight      -0.002405   0.006180  -0.389    0.700
## Ht           1.001574   0.050206  19.949   <2e-16 ***
## Seated       0.048687   0.069858   0.697    0.491
## Arm         -0.022155   0.072898  -0.304    0.763
## Thigh       -0.060584   0.048551  -1.248    0.222
## Leg          0.010946   0.088220   0.124    0.902
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7061 on 30 degrees of freedom
## Multiple R-squared:  0.9967, Adjusted R-squared:  0.996
## F-statistic:  1313 on 7 and 30 DF,  p-value: < 2.2e-16
```

Here we see that the other predictors explain 99.67% of the variation in `HtShoe`.

Now note that the variance of $\hat{\beta}_j$ can be written as

$$Var(\hat{\beta}_j) = \sigma^2 \left(\frac{1}{1 - R_j^2} \right) \frac{1}{SX_jX_j}$$

where $SX_jX_j = \sum(x_{ij} - \bar{x_j})^2$. This gives us a way to understand how multicollinearity affects our regression estimates.

We will call,

$$\frac{1}{1 - R_j^2}$$

the **variance inflation factor.** The variance inflation factor quantifies the effect of multicollinearity on the variance of our regression estimates. When R_j^2 is large, x_j is well explained by the other predictors. With a large R_j^2 the variance inflation factor becomes large. This tells us that when x_j is highly correlated with other predictors, our estimate of β_j is highly variable.

The `vif` function from the `faraway` package calculates the VIFs for each of the predictors.

```
vif(fit)
```

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

In practice it is common to say that any VIF greater than 5 is problematic. So in this example we see there is a huge multicollinearity issue.

If we add a small amount of noise to the data, we see that the estimates of the coefficients change drastically.

```
fitNoise <- lm(hipcenter+10*rnorm(38) ~ ., data = seatpos)
fit
```

```
##
## Call:
## lm(formula = hipcenter ~ ., data = seatpos)
##
## Coefficients:
## (Intercept)          Age       Weight      HtShoes           Ht
##   436.43213      0.77572      0.02631     -2.69241      0.60134
##      Seated          Arm        Thigh          Leg
##     0.53375     -1.32807     -1.14312     -6.43905
```

```
fitNoise
```

```
##
## Call:
## lm(formula = hipcenter + 10 * rnorm(38) ~ ., data = seatpos)
##
## Coefficients:
## (Intercept)          Age       Weight      HtShoes           Ht
##   483.38266      0.79614      0.07326     -6.80782      4.63232
##      Seated          Arm        Thigh          Leg
##     0.50248     -1.63447     -1.62828     -6.54831
```

Let's now look at a smaller model,

```
fit2 <- lm(hipcenter ~ Age + Weight + Ht, data = seatpos)
summary(fit2)
```

```
##
## Call:
## lm(formula = hipcenter ~ Age + Weight + Ht, data = seatpos)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -91.526 -23.005   2.164  24.950  53.982
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 528.297729 135.312947   3.904 0.000426 ***
## Age           0.519504   0.408039   1.273 0.211593
## Weight        0.004271   0.311720   0.014 0.989149
## Ht           -4.211905   0.999056  -4.216 0.000174 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 36.49 on 34 degrees of freedom
## Multiple R-squared:  0.6562, Adjusted R-squared:  0.6258
## F-statistic: 21.63 on 3 and 34 DF,  p-value: 5.125e-08
```

```
vif(fit2)
```

```
##      Age   Weight       Ht
## 1.093018 3.457681 3.463303
```

Immediately we see that multicollinearity isn't an issue here.

Let's now look at the effect of adding another variable to this model. Specifically we want to look at adding the variable `HtShoes`. So now our possible predictors are `HtShoes`, `Age`, `Weight`, and `Ht`. Our response is still `hipcenter`.

To quantify this effect we will look at a **variable added plot** and a **partial correlation coefficient**. For both of these, we will look at the residuals of two models.

- Regress the response against all of the predictors except the predictor of interest.
- Regress the predictor of interest against the other predictors.

```
fit3 <- lm(HtShoes ~ Age + Weight + Ht, data = seatpos)
```

So now, the residuals of `fit2` give us the variation of `hipcenter` that is unexplained by `Age`, `Weight`, and `Ht`. Similarly, the residuals of `fit3` give us the variation of `HtShoes` unexplained by `Age`, `Weight`, and `Ht`.

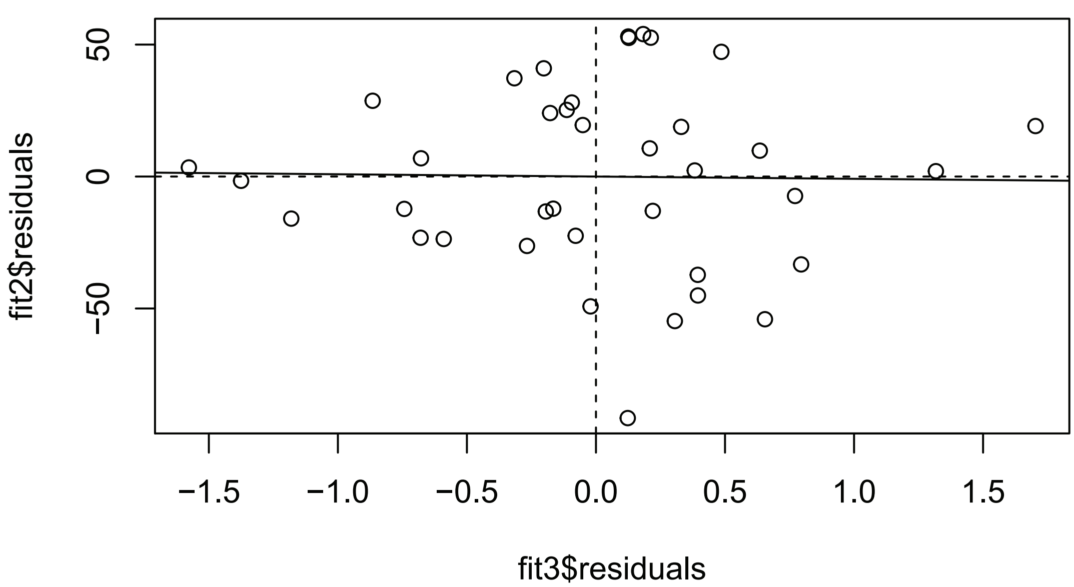
The correlation of these two residuals gives us the **partial correlation coefficient** of `HtShoes` and `hipcenter` with the effects of `Age`, `Weight`, and `Ht` removed.

```
cor(fit3$residuals, fit2$residuals)
```

```
## [1] -0.01650317
```

Similarly the **variable added plot** plots these residuals against each other. It is also helpful to regress the residuals of the response against the residuals of the predictor and add the regression line to the plot.

```
plot(fit3$residuals, fit2$residuals)
abline(h=0,lty=2)
abline(v=0,lty=2)
abline(lm(fit2$residuals ~ fit3$residuals))
```



Here the variable added plot shows almost no linear relationship and the partial correlation is very low. This tells us that adding `HtShoes` to the model would probably not be worthwhile. Since its variation is largely explained by the other predictors, adding it to the model will not do much to improve the model. However it will increase the variation of the estimates and make the model much harder to interpret.

This trade off is mostly true in general. As a model gets more predictors,errors will get smaller and its prediction will be better, but it will be harder to interpret. This is why, if we are interested in the relationship between the predictors and the response, we often want a model that fits well with a small number of predictors.

In class we also discussed the following `bodyfat` example. While with all the predictors prediction is good, we have to be careful to remember how much of the space of X we have seen and where it is a good idea to actually make predictions with correlated data.

```r
tricep <- c(19.5,24.7,30.7,29.8,19.1,25.6,31.4,27.9,22.1,25.5,
            31.1,30.4,18.7,19.7,14.6,29.5,27.7,30.2,22.7,25.2)
thigh <- c(43.1,49.8,51.9,54.3,42.2,53.9,58.5,52.1,49.9,53.5,
           56.6,56.7,46.5,44.2,42.7,54.4,55.3,58.6,48.2,51.0)
midarm <- c(29.1,28.2,37.0,31.1,30.9,23.7,27.6,30.6,23.2,24.8,
            30.0,28.3,23.0,28.6,21.3,30.1,25.7,24.6,27.1,27.5)
bodyfat <- c(11.9,22.8,18.7,20.1,12.9,21.7,27.1,25.4,21.3,19.3,
             25.4,27.2,11.7,17.8,12.8,23.9,22.6,25.4,14.8,21.1)
bodyfat <- data.frame(bodyfat, tricep, thigh, midarm)



fit <- lm(bodyfat ~., data = bodyfat)
summary(fit)
```

```
##
## Call:
## lm(formula = bodyfat ~ ., data = bodyfat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.7263 -1.6111  0.3923  1.4656  4.1277
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  117.085     99.782   1.173    0.258
## tricep         4.334      3.016   1.437    0.170
## thigh         -2.857      2.582  -1.106    0.285
## midarm        -2.186      1.595  -1.370    0.190
##
## Residual standard error: 2.48 on 16 degrees of freedom
## Multiple R-squared:  0.8014, Adjusted R-squared:  0.7641
## F-statistic: 21.52 on 3 and 16 DF,  p-value: 7.343e-06
```

```r
vif(fit)
```

```
##   tricep    thigh   midarm
## 708.8429 564.3434 104.6060
```

```r
fit4 <- lm(bodyfat ~ tricep, data = bodyfat)
summary(fit4)
```

```
##
## Call:
## lm(formula = bodyfat ~ tricep, data = bodyfat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.1195 -2.1904  0.6735  1.9383  3.8523
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -1.4961     3.3192  -0.451    0.658
## tricep        0.8572     0.1288   6.656 3.02e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.82 on 18 degrees of freedom
## Multiple R-squared:  0.7111, Adjusted R-squared:  0.695
## F-statistic:  44.3 on 1 and 18 DF,  p-value: 3.024e-06
```

```
vif(fit4)
```

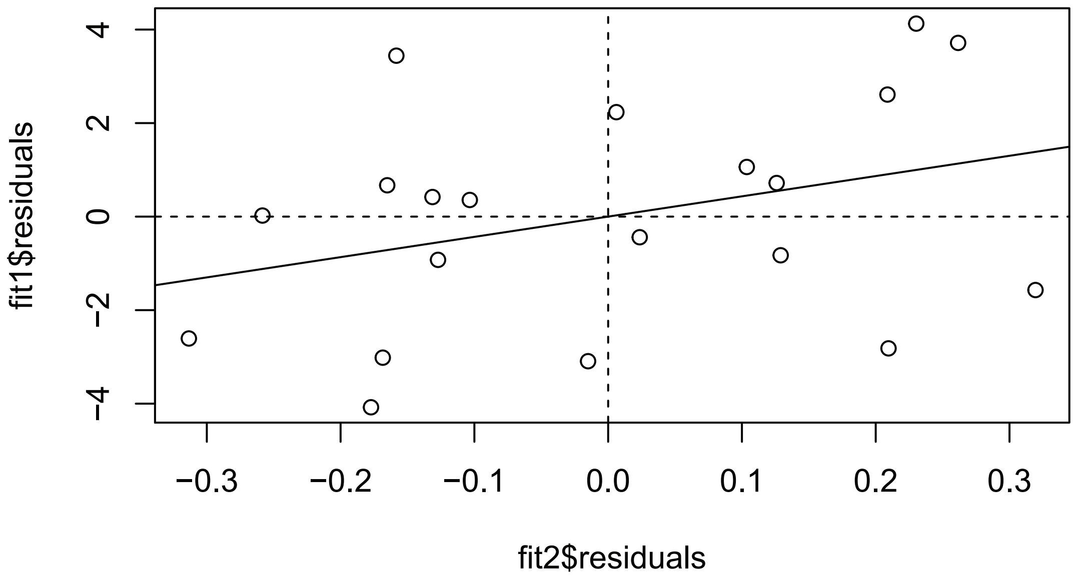
```
## tricep
##      1
```

```
fit1 <- lm(bodyfat ~ thigh + midarm, data = bodyfat)
fit2 <- lm(tricep ~ thigh + midarm, data = bodyfat)

cor(fit2$residuals,fit1$residuals)
```

```
## [1] 0.33815
```

```
plot(fit2$residuals,fit1$residuals)
abline(h=0,lty=2)
abline(v=0,lty=2)
fit3 <- lm(fit1$residuals ~ fit2$residuals)
abline(fit3)
```




```r
summary(bodyfat$tricep)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   14.60   21.50   25.55   25.30   29.90   31.40
```

```r
summary(bodyfat$thigh)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   42.20   47.78   52.00   51.17   54.62   58.60
```

```r
plot(bodyfat$tricep,bodyfat$thigh)
```