Compare R Linear Model (lm) OLS regression coefficient estimates with QR Decomposition (QR Matrix Factorization)

Brad Nott

November 29, 2018

In live session we briefly touched on how the regression coefficients are calculated in matrix form. I did some digging and discovered that the linear model (`lm`) function in R calls the lm.fit function which has a default method parameter of "qr". So the approach doesn't quite follow what is implied in the "Matrix Derivation of the OLS Estimator" asynch lecture (11.9).

To follow the approach in the asynch video the matrix $X^T X$ must be invertible (columns are linearly independent; no features that are linear combinations of each other; no perfect multicollinearity). In other words, the matrix equation $(X^T X)\beta = X^T Y$ has a single solution if $X^T X$ is invertible. If $X^T X$ does not have an inverse then an approximation for its inverse, a pseudoinverse, can be used to solve the system.

One way to get the pseudoinverse is to decompose the matrix of regressors with a factorization such as QR decomposition, and this is what R does when you fit a linear model using the `lm` function.

For (way) more detail on how this works in R, here is a lengthy blog post.

http://madrury.github.io/jekyll/update/statistics/2016/07/20/lm-in-R.html

Careful…it's a deep rabbit hole.

Comparison

For a kind of analogy to what is going on under the hood we can instead find the coefficients using the `qr.solve` function and then compare their values with what `lm` provides.

```r
data <- mtcars

# Define design (regressor) matrix
x <- as.matrix(data[-(1)])

# Add column of ones to the design matrix to represent the intercept
x <- cbind(rep(1, dim(x)[1]), x)
colnames(x)[1] <- 'intercept'

# Define output matrix
y <- as.matrix(data[1])
colnames(y) <- NULL

# Compute coefficient estimates through QR Matrix Factorization
QR_betas <- qr.solve(x, y)
colnames(QR_betas) <- 'Estimate'

# Compare QR Decomposition results with estimates computed with lm function
lm_betas <- as.matrix(summary(lm(mpg ~ ., data = data))$coefficients[,1])
```

```
lm_betas
```

```
##                    [,1]
## (Intercept) 12.30337416
## cyl         -0.11144048
## disp         0.01333524
## hp          -0.02148212
## drat         0.78711097
## wt          -3.71530393
## qsec         0.82104075
## vs           0.31776281
## am           2.52022689
## gear         0.65541302
## carb        -0.19941925
```

```
QR_betas
```

```
##              Estimate
## intercept 12.30337416
## cyl       -0.11144048
## disp       0.01333524
## hp        -0.02148212
## drat       0.78711097
## wt        -3.71530393
## qsec       0.82104075
## vs         0.31776281
## am         2.52022689
## gear       0.65541302
## carb      -0.19941925
```