Jennifer Starling
SDS 385
23 Aug 2016

Exercises 1

Part 1. Linear Regression

A) The WLS objective function, rewritten in matrix form, is

$$\hat{\beta} = \text{argmin}_{\beta\in\mathcal{R}^P} \frac{1}{2}(Y - X'\beta)'W(Y - X'\beta) = \frac{1}{2}(Y' - \beta'X')W(Y - X'\beta)$$

To satisfy the 'argmin' part of the expression, take the derivative of $\hat{\beta}$ with respect to β, set equal to zero, and solve as follows.

$$\frac{\delta}{\delta\beta}\left[\frac{1}{2}(Y' - \beta'X')W(Y - X'\beta)\right] = \left(\frac{1}{2}\right)\frac{\delta}{\delta\beta}[Y'WY - 2Y'WX\beta + \beta'XwX'\beta] = 0$$

The derivatives of each term are as follows.

1. $\frac{\delta}{\delta\beta}[Y'WY] = 0$ since is a constant wrt β.
2. $\frac{\delta}{\delta\beta}[-2Y'WX\beta] = -2Y'WX$, since derivative has form $\frac{\delta}{\delta\beta}c\beta = c$ where $c = -2Y'WX$. Then since W diagonal and X and Y vectors, $-2Y'WX = -2X'WY$.
   a. Derivation: $f(\beta) = c'\beta = c_1\beta_1 + \cdots + c_k\beta_k \rightarrow \frac{\delta f(\beta)}{\delta\beta} = \begin{bmatrix}\frac{\delta(c_1\beta_1+\cdots+c_k\beta_k)}{\delta\beta_1} \\ \cdots \\ \frac{\delta(c_1\beta_1+\cdots+c_k\beta_k)}{\delta\beta_k}\end{bmatrix} = \begin{bmatrix}c_1 \\ \cdots \\ c_k\end{bmatrix} = c$
3. $\frac{\delta}{\delta\beta}[\beta'XwX'\beta] = 2XWX'\beta$, since derivative has quadratic form $\frac{\delta}{\delta\beta}(\beta'V\beta) = (V + V')\beta = 2V\beta$, where $V = XWX'$. The last equality (=$2V\beta$) applies only when V symmetric, which holds here.

Then the derivative, subbing in $\hat{\beta}$ for β, is $\left(\frac{1}{2}\right)\left[-2X'WY + 2XWX'\hat{\beta}\right] = 0 \rightarrow\ XWX'\hat{\beta} = X'WY \rightarrow \hat{\beta} = (XWX')^{-1}X'WY$

Therefore $\hat{\beta} = (XWX')^{-1}X'WY$.

To show that $\hat{\beta} = (XWX')^{-1}X'WY$ is the solution to the linear system: $(X'WX)\hat{\beta} = (X'WX)(XWX')^{-1}X'WY = IX'WY = X'WY$

B)

C)

D)

Testing with N=6, P=3

```
> inv_method(X,W,y)
          [,1]
[1,] -0.2285076
[2,]  0.4334117
[3,]  0.2829263
> cholesky_method(X,W,y)
          [,1]
[1,] -0.2285076
[2,]  0.4334117
[3,]  0.2829263
> lu_method(X,W,y)
3 x 1 Matrix of class "dgeMatrix"
          [,1]
[1,] -0.2285076
[2,]  0.4334117
[3,]  0.2829263
```

```
> perf_results
$`N=10,P=5`
Unit: microseconds
                      expr    min      lq     mean  median      uq     max neval cld
      inv_method(X, W, y) 37.611 41.8100 45.06385 44.7085 47.2725  64.006   100  a 
       lu_method(X, W, y) 84.483 90.8865 96.81388 94.6975 99.6330 180.891   100   b
 cholesky_method(X, W, y) 77.703 81.4315 93.98773 85.9800 92.2835 741.216   100   b

$`N=100,P=50`
Unit: milliseconds
                      expr      min       lq     mean   median       uq      max neval cld
      inv_method(X, W, y) 1.341674 1.353068 1.475150 1.373673 1.494854 2.583980   100   b
       lu_method(X, W, y) 1.058799 1.080653 1.224056 1.141064 1.244571 3.091794   100  a 
 cholesky_method(X, W, y) 1.307661 1.333443 1.510231 1.364680 1.511695 5.079535   100   b

$`N=500,P=250`
Unit: milliseconds
                      expr      min       lq     mean   median       uq      max neval cld
      inv_method(X, W, y) 140.5526 146.9083 150.7572 149.6457 152.7927 241.2390   100   c
       lu_method(X, W, y) 101.7165 106.7253 110.1241 109.2953 112.2400 189.4535   100 a  
 cholesky_method(X, W, y) 126.9278 133.7628 136.5326 136.3793 138.8827 145.9535   100  b 

$`N=1000,P=500`
Unit: milliseconds
                      expr       min        lq      mean    median        uq      max neval cld
      inv_method(X, W, y) 1184.4662 1213.3566 1240.0551 1227.1508 1255.5876 1336.853   100   c
       lu_method(X, W, y)  879.8264  897.5688  932.9878  909.2964  983.9252 1026.348   100 a  
 cholesky_method(X, W, y) 1075.1613 1100.6688 1134.6080 1113.2897 1185.2184 1240.524   100  b 
```