IS622 Week 12 Exercises

Brian Chu | Nov 10, 2015

Exercise 9.4.2: If we wish to start out, as in Fig. 9.10, with all U and V entries set to the same value, what value minimizes the RMSE for the matrix M of our running example?

Each UV element $= x^2 + x^2 = 2x^2$

M - UV =
$[(5-(2x^2)]^2 + (2-2x^2)^2 + (4-2x^2)^2 + ...])]$

Set derivative to zero:
$-8x[(5-2x^2) + (2-2x^2) + (4-2x^2) + ...] = 0$

x cannot be zero so:
$[(5-2x^2) + (2-2x^2) + (4-2x^2) + ...] = 0$
$75 - 46x^2 = 0$
$x = \sqrt{75/46}$
$= 1.2769$

```r
x <- 1.2769

U <- matrix(rep(x,10), nrow=5, ncol=2)
V <- matrix(rep(x,10), nrow=2, ncol=5)
UV <- U %*% V

M <- matrix(c(5,2,4,4,3,
              3,1,2,4,1,
              2,NA,3,1,4,
              2,5,4,3,5,
              4,4,5,4,NA), nrow=5)
M <- t(M)
MUV <- M - UV

# Sum of squared error
print(sum((MUV)^2, na.rm=TRUE))
```

```
## [1] 38.43478
```

Exercise 9.4.3: Starting with the U and V matrices in Fig. 9.16, do the following in order:

```r
U <- matrix(c(2.6,1,
              1.178466, 1,
              1,1,
              1,1,
              1,1), nrow=5, ncol=2)

V <- matrix(c(17.4/10.76, 1,1,1,1,
```

```r
              1,1,1,1,1), nrow=2, ncol=5)

UV <- U %*% V

# Fig. 9.16 starting point
print(U)
```

```
##          [,1] [,2]
## [1,] 2.600000    1
## [2,] 1.000000    1
## [3,] 1.178466    1
## [4,] 1.000000    1
## [5,] 1.000000    1
```

```r
print(V)
```

```
##        [,1] [,2] [,3] [,4] [,5]
## [1,] 1.6171    1    1    1    1
## [2,] 1.0000    1    1    1    1
```

```r
print(UV)
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 5.204461 3.600000 3.600000 3.600000 3.600000
## [2,] 2.617100 2.000000 2.000000 2.000000 2.000000
## [3,] 2.905698 2.178466 2.178466 2.178466 2.178466
## [4,] 2.617100 2.000000 2.000000 2.000000 2.000000
## [5,] 2.617100 2.000000 2.000000 2.000000 2.000000
```

(a) Reconsider the value of u11. Find its new best value, given the changes that have been made so far. From the earlier steps on page 329, we can see the pattern for the derivative equation simplifies to:

sum(M[1,]) - sum(V[2,]) - sum(V[1,])x = 0

$13 - 5.617x = 0$
$x = 13/5.617$
$x = 2.314361$

```r
# Solve for x
u11 <- (sum(M[1,]) - sum(V[2,]))  / sum(V[1,])
print(u11)
```

```
## [1] 2.314361
```

```r
# Updated U
U2 <- U
U2[1,1] <- u11
print(U2)
```

```
##          [,1] [,2]
## [1,] 2.314361    1
## [2,] 1.000000    1
## [3,] 1.178466    1
## [4,] 1.000000    1
## [5,] 1.000000    1
```

```r
# Updated UV
U2V <- U2 %*% V
print(U2V)
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 4.742555 3.314361 3.314361 3.314361 3.314361
## [2,] 2.617100 2.000000 2.000000 2.000000 2.000000
## [3,] 2.905698 2.178466 2.178466 2.178466 2.178466
## [4,] 2.617100 2.000000 2.000000 2.000000 2.000000
## [5,] 2.617100 2.000000 2.000000 2.000000 2.000000
```

```r
# Updated SSE
MU2V <- M - U2V
print(sum((MU2V)^2, na.rm=TRUE))
```

```
## [1] 57.47466
```

(b) Then choose the best value for u52. Similar to equation above, but switch v[1,] and v[2,] because dealing with U_{i2}:

sum(M[5,]) - sum(V[1,]) - sum(V[2,])x = 0
Skip M[5,5] because of NA value

$12.383 - 4x = 0$
$x = 3.095725$

```r
# Solve for x
u52 <- (sum(M[5,-5]) - sum(V[1,-5])) / sum(V[2,-5])
print(u52)
```

```
## [1] 3.095725
```

```r
# Updated U
U3 <- U2
U3[5,2] <- u52
print(U3)
```

```
##          [,1]     [,2]
## [1,] 2.314361 1.000000
## [2,] 1.000000 1.000000
## [3,] 1.178466 1.000000
## [4,] 1.000000 1.000000
## [5,] 1.000000 3.095725
```

```r
# Updated UV
U3V <- U3 %*% V
print(U3V)
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 4.742555 3.314361 3.314361 3.314361 3.314361
## [2,] 2.617100 2.000000 2.000000 2.000000 2.000000
## [3,] 2.905698 2.178466 2.178466 2.178466 2.178466
## [4,] 2.617100 2.000000 2.000000 2.000000 2.000000
## [5,] 4.712825 4.095725 4.095725 4.095725 4.095725
```

```r
# Updated SSE
MU3V <- M - U3V
print(sum((MU3V)^2, na.rm=TRUE))
```

```
## [1] 39.90641
```

(c) Then choose the best value for v22. Same derivative equation, but switch V for U:

sum(M[,2]) - sum(U[,1]) - sum(U[,2])y = 0
Skip M[3,2] because of NA value

$6.686 - 6.096y = 0$
$y = 1.096775$

```r
# Solve for y
v22 <- (sum(M[-3,2]) - sum(U3[-3,1])) / sum(U3[-3,2])
print(v22)
```

```
## [1] 1.096775
```

```r
# Updated V
V2 <- V
V2[2,2] <- v22
print(V2)
```

```
##        [,1]     [,2] [,3] [,4] [,5]
## [1,] 1.6171 1.000000    1    1    1
## [2,] 1.0000 1.096775    1    1    1
```

```r
# Updated UV
U3V2 <- U3 %*% V2
print(U3V2)
```

```
##          [,1]     [,2]     [,3]     [,4]     [,5]
## [1,] 4.742555 3.411136 3.314361 3.314361 3.314361
## [2,] 2.617100 2.096775 2.000000 2.000000 2.000000
## [3,] 2.905698 2.275241 2.178466 2.178466 2.178466
## [4,] 2.617100 2.096775 2.000000 2.000000 2.000000
## [5,] 4.712825 4.395314 4.095725 4.095725 4.095725
```

```r
# Updated SSE
MU3V2 <- M - U3V2
print(sum((MU3V2)^2, na.rm=TRUE))
```

```
## [1] 39.94891
```