Lab 6

Vincent Miceli

11:59PM March 24, 2019

Load the Boston Housing data and create the vector y and the design matrix X.

```
data(Boston, package = 'MASS')
y = as.matrix(Boston$medv)
x = Boston[, 1:13]
x = as.matrix(cbind(1, x))
```

Find the OLS estimate and OLS predictions without using `lm`.

```
b = solve(t(x) %*% x) %*% t(x) %*% y
b
```

```
##                  [,1]
## 1        3.645949e+01
## crim    -1.080114e-01
## zn       4.642046e-02
## indus    2.055863e-02
## chas     2.686734e+00
## nox     -1.776661e+01
## rm       3.809865e+00
## age      6.922246e-04
## dis     -1.475567e+00
## rad      3.060495e-01
## tax     -1.233459e-02
## ptratio -9.527472e-01
## black    9.311683e-03
## lstat   -5.247584e-01
```

```
yhat = x %*% b
```

Write a function spec'd as follows:

```
#' Orthogonal Projection
#'
#' Projects vector a onto v.
#'
#' @param a   the vector to project
#' @param v   the vector projected onto
#'
#' @returns   a list of two vectors, the orthogonal projection parallel to v named a_parallel,
#'            and the orthogonal error orthogonal to v called a_perpendicular
orthogonal_projection = function(a, v){
  a_parallel = (v %*% t(v) %*% a) / (sum(v^2))
  a_perpendicular = a - a_parallel
  list("a_parallel" = a_parallel, "a_perpendicular" = a_perpendicular)
```

```r
}

orthogonal_projection(1:4, 1:4)
```

```
## $a_parallel
##      [,1]
## [1,]    1
## [2,]    2
## [3,]    3
## [4,]    4
##
## $a_perpendicular
##      [,1]
## [1,]    0
## [2,]    0
## [3,]    0
## [4,]    0
```

```r
orthogonal_projection(1:4, c(0,2,0,-1))
```

```
## $a_parallel
##      [,1]
## [1,]    0
## [2,]    0
## [3,]    0
## [4,]    0
##
## $a_perpendicular
##      [,1]
## [1,]    1
## [2,]    2
## [3,]    3
## [4,]    4
```

```r
result = orthogonal_projection(c(2,6,7,3), c(1,3,5,7))

t(result$a_parallel) %*% result$a_perpendicular
```

```
##              [,1]
## [1,] 7.105427e-15
```

```r
sum(result$a_parallel) + sum(result$a_perpendicular)
```

```
## [1] 18
```

```r
result$a_parallel / c(1,3,5,7)
```

```
##           [,1]
## [1,] 0.9047619
## [2,] 0.9047619
## [3,] 0.9047619
## [4,] 0.9047619
```

Try to project onto the column space of X by projecting y on each vector of X individually and adding up the projections. You can use the function `orthogonal_projection`.

```
sum = rep(0, nrow(x))
for (j in 1 : ncol(x)){
  sum = sum + orthogonal_projection(y, x[, j])$a_parallel
}
```

How much double counting occurred? Measure the magnitude relative to the true LS orthogonal projection.

```
d = sum / yhat
```

Convert X into Q where Q has the same column space as X but has orthogonal columns. You can use the function `orthogonal_projection`. This is essentially gram-schmidt.

```
Q = matrix(NA, nrow = nrow(x), ncol = ncol(x))
Q[, 1] = x[, 1]
for(j in 2 : ncol(x)){
  Q[ , j] = x[ , j]

  for(j0 in 1 : (j - 1)){
    Q[ , j] = Q[ , j] - (orthogonal_projection(x[ , j], Q[ , j0])$a_parallel)
  }
}
pacman::p_load(Matrix)
rankMatrix(Q)
```

```
## [1] 14
## attr(,"method")
## [1] "tolNorm2"
## attr(,"useGrad")
## [1] FALSE
## attr(,"tol")
## [1] 1.123546e-13
```

```
dim(Q)
```

```
## [1] 506  14
```

```
ncol(x)
```

```
## [1] 14
```

```
t(Q) %*% Q
```

```
##                [,1]          [,2]          [,3]          [,4]
##  [1,]  5.060000e+02 -1.544542e-12 -8.473222e-13 -1.064282e-11
##  [2,] -1.544542e-12  3.736322e+04  1.833200e-12  1.820544e-12
##  [3,] -8.473222e-13  1.833200e-12  2.636490e+05  4.443779e-12
##  [4,] -1.064282e-11  1.820544e-12  4.443779e-12  1.477223e+04
```

```
##  [5,]  4.116152e-14  3.180789e-14  1.194600e-13  7.313386e-13
##  [6,]  2.738278e-13  2.109771e-13  1.129652e-14  5.964510e-12
##  [7,] -4.435674e-12  2.954414e-12 -1.170175e-12  6.620642e-11
##  [8,] -2.233413e-11 -3.858247e-12  9.720225e-12 -1.070166e-10
##  [9,] -6.893375e-13  3.677059e-12 -1.001865e-12  1.132529e-10
## [10,]  2.939871e-12 -5.329071e-12 -3.808509e-12 -9.987211e-11
## [11,]  4.102674e-11  1.738272e-10 -2.785328e-12  3.081497e-09
## [12,] -1.135136e-11  8.789414e-12  7.247536e-13  2.656571e-10
## [13,]  4.072831e-10  1.519851e-10 -5.897505e-11  4.968760e-09
## [14,] -1.388312e-11  1.529799e-11 -5.783818e-12  3.403625e-10
##               [,5]          [,6]          [,7]          [,8]
##  [1,]  4.116152e-14  2.738278e-13 -4.435674e-12 -2.233413e-11
##  [2,]  3.180789e-14  2.109771e-13  2.954414e-12 -3.858247e-12
##  [3,]  1.194600e-13  1.129652e-14 -1.170175e-12  9.720225e-12
##  [4,]  7.313386e-13  5.964510e-12  6.620642e-11 -1.070166e-10
##  [5,]  3.218831e+01 -2.675475e-14 -1.918830e-13 -2.806644e-13
##  [6,] -2.675475e-14  2.591084e+00 -1.536766e-12 -2.640610e-11
##  [7,] -1.918830e-13 -1.536766e-12  2.029377e+02  3.697231e-10
##  [8,] -2.806644e-13 -2.640610e-11  3.697231e-10  1.617318e+05
##  [9,] -3.403527e-13 -1.304247e-12  2.052783e-11  2.128964e-12
## [10,]  5.884182e-14 -4.850051e-12  3.982170e-11  5.506209e-10
## [11,] -1.479150e-11 -1.340538e-10  6.804788e-10  1.165498e-08
## [12,] -9.342527e-13 -5.017084e-12  5.508982e-11  3.352234e-10
## [13,] -1.553480e-11 -9.259564e-11  1.604291e-09  6.060120e-09
## [14,] -1.191491e-12 -1.036152e-11  1.720823e-11  2.285184e-09
##               [,9]         [,10]         [,11]         [,12]
##  [1,] -6.893375e-13  2.939871e-12  4.102674e-11 -1.135136e-11
##  [2,]  3.677059e-12 -5.329071e-12  1.738272e-10  8.789414e-12
##  [3,] -1.001865e-12 -3.808509e-12 -2.785328e-12  7.247536e-13
##  [4,]  1.132529e-10 -9.987211e-11  3.081497e-09  2.656571e-10
##  [5,] -3.403527e-13  5.884182e-14 -1.479150e-11 -9.342527e-13
##  [6,] -1.304247e-12 -4.850051e-12 -1.340538e-10 -5.017084e-12
##  [7,]  2.052783e-11  3.982170e-11  6.804788e-10  5.508982e-11
##  [8,]  2.128964e-12  5.506209e-10  1.165498e-08  3.352234e-10
##  [9,]  5.742738e+02 -4.222489e-11 -4.938201e-10  1.419753e-11
## [10,] -4.222489e-11  1.664085e+04  2.342631e-09 -6.246736e-11
## [11,] -4.938201e-10  2.342631e-09  1.602478e+06 -1.758217e-09
## [12,]  1.419753e-11 -6.246736e-11 -1.758217e-09  1.319301e+03
## [13,]  3.850618e-10 -2.053042e-09 -1.707542e-08  4.196387e-09
## [14,]  6.702461e-11  2.036771e-10  1.914600e-09  3.358842e-10
##              [,13]         [,14]
##  [1,]  4.072831e-10 -1.388312e-11
##  [2,]  1.519851e-10  1.529799e-11
##  [3,] -5.897505e-11 -5.783818e-12
##  [4,]  4.968760e-09  3.403625e-10
##  [5,] -1.553480e-11 -1.191491e-12
##  [6,] -9.259564e-11 -1.036152e-11
##  [7,]  1.604291e-09  1.720823e-11
##  [8,]  6.060120e-09  2.285184e-09
##  [9,]  3.850618e-10  6.702461e-11
## [10,] -2.053042e-09  2.036771e-10
## [11,] -1.707542e-08  1.914600e-09
## [12,]  4.196387e-09  3.358842e-10
## [13,]  3.198118e+06 -8.166268e-11
```

```
## [14,] -8.166268e-11  8.754864e+03
```

Make Q's columns orthonormal.

```r
for (j in 1 : ncol(Q)){
  Q[, j] = Q[, j] / sqrt(sum(Q[, j]^2))
}
head(Q)
```

```
##            [,1]        [,2]         [,3]        [,4]         [,5]
## [1,] 0.04445542 -0.01866158  0.009106011 -0.05766684 -0.008302544
## [2,] 0.04445542 -0.01855299 -0.025927537 -0.03907578 -0.011665235
## [3,] 0.04445542 -0.01855310 -0.025927558 -0.03907574 -0.011665245
## [4,] 0.04445542 -0.01852682 -0.025922180 -0.07931947 -0.008568726
## [5,] 0.04445542 -0.01833705 -0.025883351 -0.07939459 -0.008550130
## [6,] 0.04445542 -0.01853985 -0.025924848 -0.07931431 -0.008570004
##              [,6]         [,7]         [,8]        [,9]        [,10]
## [1,]  0.055557124 -0.001676246  0.013977978 -0.01965710 -0.030550491
## [2,] -0.026929058  0.005420631  0.049516097  0.03918782 -0.021991211
## [3,] -0.026929034  0.059051140  0.001654837  0.03716313 -0.028975492
## [4,] -0.001935136  0.034898057 -0.025673224  0.05481814 -0.008670817
## [5,] -0.001978314  0.045372004 -0.005504113  0.06345347 -0.008885414
## [6,] -0.001932169 -0.004974848  0.009081307  0.05606882 -0.003505468
##             [,11]       [,12]         [,13]        [,14]
## [1,]  0.055974401 -0.03828158  0.0049925067 -0.043530126
## [2,] -0.015651477 -0.02104506  0.0017126748 -0.023172211
## [3,] -0.008739792 -0.00395963  0.0014648925 -0.025246544
## [4,] -0.008082853  0.02143491  0.0002709019 -0.029225402
## [5,] -0.007774033  0.02166151  0.0016094592 -0.004478635
## [6,] -0.013192315  0.00881971 -0.0015143931 -0.044015945
```

Verify Q^T is the inverse of Q.

```r
t(Q) %*% Q
```

```
##                [,1]          [,2]          [,3]          [,4]
##  [1,]  1.000000e+00 -1.170938e-16  7.329207e-17 -3.932090e-15
##  [2,] -1.170938e-16  1.000000e+00  1.566672e-17  6.763727e-17
##  [3,]  7.329207e-17  1.566672e-17  1.000000e+00 -5.826231e-17
##  [4,] -3.932090e-15  6.763727e-17 -5.826231e-17  1.000000e+00
##  [5,]  3.044440e-16  4.510281e-17  3.794708e-19  1.051744e-15
##  [6,]  7.548107e-15  6.550750e-16  5.526721e-17  3.046028e-14
##  [7,] -1.379756e-14  1.082847e-15 -2.208520e-16  3.826098e-14
##  [8,] -2.475017e-15 -7.361733e-17  5.084908e-17 -2.164291e-15
##  [9,] -1.269384e-15  7.773730e-16  2.385245e-18  3.891581e-14
## [10,]  1.098514e-15 -2.138047e-16 -9.540979e-18 -6.627464e-15
## [11,]  1.463239e-15  7.455516e-16  4.065758e-17  2.017742e-14
## [12,] -1.382228e-14  1.229485e-15  2.602085e-17  6.014552e-14
## [13,]  1.006416e-14  2.636644e-16 -5.095750e-17  2.289555e-14
## [14,] -6.628812e-15  8.515324e-16 -1.021318e-16  2.996148e-14
##                [,5]          [,6]          [,7]          [,8]
##  [1,]  3.044440e-16  7.548107e-15 -1.379756e-14 -2.475017e-15
```

```
##  [2,]  4.510281e-17  6.550750e-16  1.082847e-15 -7.361733e-17
##  [3,]  3.794708e-19  5.526721e-17 -2.208520e-16  5.084908e-17
##  [4,]  1.051744e-15  3.046028e-14  3.826098e-14 -2.164291e-15
##  [5,]  1.000000e+00 -2.882202e-15 -2.479679e-15 -1.329232e-16
##  [6,] -2.882202e-15  1.000000e+00 -6.696465e-14 -4.081119e-14
##  [7,] -2.479679e-15 -6.696465e-14  1.000000e+00  6.453291e-14
##  [8,] -1.329232e-16 -4.081119e-14  6.453291e-14  1.000000e+00
##  [9,] -2.511229e-15 -3.385638e-14  6.016531e-14  1.811702e-16
## [10,]  3.783866e-17 -2.339584e-14  2.159926e-14  1.060024e-14
## [11,] -2.035237e-15 -6.567132e-14  3.771779e-14  2.284709e-14
## [12,] -4.422678e-15 -8.574749e-14  1.065354e-13  2.301436e-14
## [13,] -1.515213e-15 -3.213684e-14  6.298919e-14  8.422896e-15
## [14,] -2.182933e-15 -6.870822e-14  1.289062e-14  6.070513e-14
##                [,9]         [,10]         [,11]         [,12]
##  [1,] -1.269384e-15  1.098514e-15  1.463239e-15 -1.382228e-14
##  [2,]  7.773730e-16 -2.138047e-16  7.455516e-16  1.229485e-15
##  [3,]  2.385245e-18 -9.540979e-18  4.065758e-17  2.602085e-17
##  [4,]  3.891581e-14 -6.627464e-15  2.017742e-14  6.014552e-14
##  [5,] -2.511229e-15  3.783866e-17 -2.035237e-15 -4.422678e-15
##  [6,] -3.385638e-14 -2.339584e-14 -6.567132e-14 -8.574749e-14
##  [7,]  6.016531e-14  2.159926e-14  3.771779e-14  1.065354e-13
##  [8,]  1.811702e-16  1.060024e-14  2.284709e-14  2.301436e-14
##  [9,]  1.000000e+00 -1.368133e-14 -1.628602e-14  1.636278e-14
## [10,] -1.368133e-14  1.000000e+00  1.449112e-14 -1.325676e-14
## [11,] -1.628602e-14  1.449112e-14  1.000000e+00 -3.825694e-14
## [12,]  1.636278e-14 -1.325676e-14 -3.825694e-14  1.000000e+00
## [13,]  8.986952e-15 -8.906396e-15 -7.539284e-15  6.461352e-14
## [14,]  2.987671e-14  1.688667e-14  1.612241e-14  9.881852e-14
##               [,13]         [,14]
##  [1,]  1.006416e-14 -6.628812e-15
##  [2,]  2.636644e-16  8.515324e-16
##  [3,] -5.095750e-17 -1.021318e-16
##  [4,]  2.289555e-14  2.996148e-14
##  [5,] -1.515213e-15 -2.182933e-15
##  [6,] -3.213684e-14 -6.870822e-14
##  [7,]  6.298919e-14  1.289062e-14
##  [8,]  8.422896e-15  6.070513e-14
##  [9,]  8.986952e-15  2.987671e-14
## [10,] -8.906396e-15  1.688667e-14
## [11,] -7.539284e-15  1.612241e-14
## [12,]  6.461352e-14  9.881852e-14
## [13,]  1.000000e+00 -4.839878e-16
## [14,] -4.839878e-16  1.000000e+00
```

Project Y onto Q and verify it is the same as the OLS fit.

```r
cbind(Q %*% t(Q) %*% y, yhat)
```

```
##          [,1]       [,2]
## 1   30.0038434 30.0038434
## 2   25.0255624 25.0255624
## 3   30.5675967 30.5675967
## 4   28.6070365 28.6070365
```

```
## 5   27.9435242 27.9435242
## 6   25.2562845 25.2562845
## 7   23.0018083 23.0018083
## 8   19.5359884 19.5359884
## 9   11.5236369 11.5236369
## 10  18.9202621 18.9202621
## 11  18.9994965 18.9994965
## 12  21.5867957 21.5867957
## 13  20.9065215 20.9065215
## 14  19.5529028 19.5529028
## 15  19.2834821 19.2834821
## 16  19.2974832 19.2974832
## 17  20.5275098 20.5275098
## 18  16.9114013 16.9114013
## 19  16.1780111 16.1780111
## 20  18.4061360 18.4061360
## 21  12.5238575 12.5238575
## 22  17.6710367 17.6710367
## 23  15.8328813 15.8328813
## 24  13.8062853 13.8062853
## 25  15.6783383 15.6783383
## 26  13.3866856 13.3866856
## 27  15.4639765 15.4639765
## 28  14.7084743 14.7084743
## 29  19.5473729 19.5473729
## 30  20.8764282 20.8764282
## 31  11.4551176 11.4551176
## 32  18.0592329 18.0592329
## 33   8.8110574  8.8110574
## 34  14.2827581 14.2827581
## 35  13.7067589 13.7067589
## 36  23.8146353 23.8146353
## 37  22.3419371 22.3419371
## 38  23.1089114 23.1089114
## 39  22.9150261 22.9150261
## 40  31.3576257 31.3576257
## 41  34.2151023 34.2151023
## 42  28.0205641 28.0205641
## 43  25.2038663 25.2038663
## 44  24.6097927 24.6097927
## 45  22.9414918 22.9414918
## 46  22.0966982 22.0966982
## 47  20.4232003 20.4232003
## 48  18.0365509 18.0365509
## 49   9.1065538  9.1065538
## 50  17.2060775 17.2060775
## 51  21.2815254 21.2815254
## 52  23.9722228 23.9722228
## 53  27.6558508 27.6558508
## 54  24.0490181 24.0490181
## 55  15.3618477 15.3618477
## 56  31.1526495 31.1526495
## 57  24.8568698 24.8568698
## 58  33.1091981 33.1091981
```

```
## 59  21.7753799 21.7753799
## 60  21.0849356 21.0849356
## 61  17.8725804 17.8725804
## 62  18.5111021 18.5111021
## 63  23.9874286 23.9874286
## 64  22.5540887 22.5540887
## 65  23.3730864 23.3730864
## 66  30.3614836 30.3614836
## 67  25.5305651 25.5305651
## 68  21.1133856 21.1133856
## 69  17.4215379 17.4215379
## 70  20.7848363 20.7848363
## 71  25.2014886 25.2014886
## 72  21.7426577 21.7426577
## 73  24.5574496 24.5574496
## 74  24.0429571 24.0429571
## 75  25.5049972 25.5049972
## 76  23.9669302 23.9669302
## 77  22.9454540 22.9454540
## 78  23.3569982 23.3569982
## 79  21.2619827 21.2619827
## 80  22.4281737 22.4281737
## 81  28.4057697 28.4057697
## 82  26.9948609 26.9948609
## 83  26.0357630 26.0357630
## 84  25.0587348 25.0587348
## 85  24.7845667 24.7845667
## 86  27.7904920 27.7904920
## 87  22.1685342 22.1685342
## 88  25.8927642 25.8927642
## 89  30.6746183 30.6746183
## 90  30.8311062 30.8311062
## 91  27.1190194 27.1190194
## 92  27.4126673 27.4126673
## 93  28.9412276 28.9412276
## 94  29.0810555 29.0810555
## 95  27.0397736 27.0397736
## 96  28.6245995 28.6245995
## 97  24.7274498 24.7274498
## 98  35.7815952 35.7815952
## 99  35.1145459 35.1145459
## 100 32.2510280 32.2510280
## 101 24.5802202 24.5802202
## 102 25.5941347 25.5941347
## 103 19.7901368 19.7901368
## 104 20.3116713 20.3116713
## 105 21.4348259 21.4348259
## 106 18.5399401 18.5399401
## 107 17.1875599 17.1875599
## 108 20.7504903 20.7504903
## 109 22.6482911 22.6482911
## 110 19.7720367 19.7720367
## 111 20.6496586 20.6496586
## 112 26.5258674 26.5258674
```

```
## 113 20.7732364 20.7732364
## 114 20.7154831 20.7154831
## 115 25.1720888 25.1720888
## 116 20.4302559 20.4302559
## 117 23.3772463 23.3772463
## 118 23.6904326 23.6904326
## 119 20.3357836 20.3357836
## 120 20.7918087 20.7918087
## 121 21.9163207 21.9163207
## 122 22.4710778 22.4710778
## 123 20.5573856 20.5573856
## 124 16.3666198 16.3666198
## 125 20.5609982 20.5609982
## 126 22.4817845 22.4817845
## 127 14.6170663 14.6170663
## 128 15.1787668 15.1787668
## 129 18.9386859 18.9386859
## 130 14.0557329 14.0557329
## 131 20.0352740 20.0352740
## 132 19.4101340 19.4101340
## 133 20.0619157 20.0619157
## 134 15.7580767 15.7580767
## 135 13.2564524 13.2564524
## 136 17.2627773 17.2627773
## 137 15.8784188 15.8784188
## 138 19.3616395 19.3616395
## 139 13.8148390 13.8148390
## 140 16.4488147 16.4488147
## 141 13.5714193 13.5714193
## 142  3.9888551  3.9888551
## 143 14.5949548 14.5949548
## 144 12.1488148 12.1488148
## 145  8.7282236  8.7282236
## 146 12.0358534 12.0358534
## 147 15.8208206 15.8208206
## 148  8.5149902  8.5149902
## 149  9.7184414  9.7184414
## 150 14.8045137 14.8045137
## 151 20.8385815 20.8385815
## 152 18.3010117 18.3010117
## 153 20.1228256 20.1228256
## 154 17.2860189 17.2860189
## 155 22.3660023 22.3660023
## 156 20.1037592 20.1037592
## 157 13.6212589 13.6212589
## 158 33.2598270 33.2598270
## 159 29.0301727 29.0301727
## 160 25.5675277 25.5675277
## 161 32.7082767 32.7082767
## 162 36.7746701 36.7746701
## 163 40.5576584 40.5576584
## 164 41.8472817 41.8472817
## 165 24.7886738 24.7886738
## 166 25.3788924 25.3788924
```

```
## 167 37.2034745 37.2034745
## 168 23.0874875 23.0874875
## 169 26.4027396 26.4027396
## 170 26.6538211 26.6538211
## 171 22.5551466 22.5551466
## 172 24.2908281 24.2908281
## 173 22.9765722 22.9765722
## 174 29.0719431 29.0719431
## 175 26.5219434 26.5219434
## 176 30.7220906 30.7220906
## 177 25.6166931 25.6166931
## 178 29.1374098 29.1374098
## 179 31.4357197 31.4357197
## 180 32.9223157 32.9223157
## 181 34.7244046 34.7244046
## 182 27.7655211 27.7655211
## 183 33.8878732 33.8878732
## 184 30.9923804 30.9923804
## 185 22.7182001 22.7182001
## 186 24.7664781 24.7664781
## 187 35.8849723 35.8849723
## 188 33.4247672 33.4247672
## 189 32.4119915 32.4119915
## 190 34.5150995 34.5150995
## 191 30.7610949 30.7610949
## 192 30.2893414 30.2893414
## 193 32.9191871 32.9191871
## 194 32.1126077 32.1126077
## 195 31.5587100 31.5587100
## 196 40.8455572 40.8455572
## 197 36.1277008 36.1277008
## 198 32.6692081 32.6692081
## 199 34.7046912 34.7046912
## 200 30.0934516 30.0934516
## 201 30.6439391 30.6439391
## 202 29.2871950 29.2871950
## 203 37.0714839 37.0714839
## 204 42.0319312 42.0319312
## 205 43.1894984 43.1894984
## 206 22.6903480 22.6903480
## 207 23.6828471 23.6828471
## 208 17.8544721 17.8544721
## 209 23.4942899 23.4942899
## 210 17.0058772 17.0058772
## 211 22.3925110 22.3925110
## 212 17.0604275 17.0604275
## 213 22.7389292 22.7389292
## 214 25.2194255 25.2194255
## 215 11.1191674 11.1191674
## 216 24.5104915 24.5104915
## 217 26.6033477 26.6033477
## 218 28.3551871 28.3551871
## 219 24.9152546 24.9152546
## 220 29.6865277 29.6865277
```

```
## 221 33.1841975 33.1841975
## 222 23.7745666 23.7745666
## 223 32.1405196 32.1405196
## 224 29.7458199 29.7458199
## 225 38.3710245 38.3710245
## 226 39.8146187 39.8146187
## 227 37.5860575 37.5860575
## 228 32.3995325 32.3995325
## 229 35.4566524 35.4566524
## 230 31.2341151 31.2341151
## 231 24.4844923 24.4844923
## 232 33.2883729 33.2883729
## 233 38.0481048 38.0481048
## 234 37.1632863 37.1632863
## 235 31.7138352 31.7138352
## 236 25.2670557 25.2670557
## 237 30.1001074 30.1001074
## 238 32.7198716 32.7198716
## 239 28.4271706 28.4271706
## 240 28.4294068 28.4294068
## 241 27.2937594 27.2937594
## 242 23.7426248 23.7426248
## 243 24.1200789 24.1200789
## 244 27.4020841 27.4020841
## 245 16.3285756 16.3285756
## 246 13.3989126 13.3989126
## 247 20.0163878 20.0163878
## 248 19.8618443 19.8618443
## 249 21.2883131 21.2883131
## 250 24.0798915 24.0798915
## 251 24.2063355 24.2063355
## 252 25.0421582 25.0421582
## 253 24.9196401 24.9196401
## 254 29.9456337 29.9456337
## 255 23.9722832 23.9722832
## 256 21.6958089 21.6958089
## 257 37.5110924 37.5110924
## 258 43.3023904 43.3023904
## 259 36.4836142 36.4836142
## 260 34.9898859 34.9898859
## 261 34.8121151 34.8121151
## 262 37.1663133 37.1663133
## 263 40.9892850 40.9892850
## 264 34.4463409 34.4463409
## 265 35.8339755 35.8339755
## 266 28.2457430 28.2457430
## 267 31.2267359 31.2267359
## 268 40.8395575 40.8395575
## 269 39.3179239 39.3179239
## 270 25.7081791 25.7081791
## 271 22.3029553 22.3029553
## 272 27.2034097 27.2034097
## 273 28.5116947 28.5116947
## 274 35.4767660 35.4767660
```

```
## 275 36.1063916 36.1063916
## 276 33.7966827 33.7966827
## 277 35.6108586 35.6108586
## 278 34.8399338 34.8399338
## 279 30.3519266 30.3519266
## 280 35.3098070 35.3098070
## 281 38.7975697 38.7975697
## 282 34.3312319 34.3312319
## 283 40.3396307 40.3396307
## 284 44.6730834 44.6730834
## 285 31.5968909 31.5968909
## 286 27.3565923 27.3565923
## 287 20.1017415 20.1017415
## 288 27.0420667 27.0420667
## 289 27.2136458 27.2136458
## 290 26.9139584 26.9139584
## 291 33.4356331 33.4356331
## 292 34.4034963 34.4034963
## 293 31.8333982 31.8333982
## 294 25.8178324 25.8178324
## 295 24.4298235 24.4298235
## 296 28.4576434 28.4576434
## 297 27.3626700 27.3626700
## 298 19.5392876 19.5392876
## 299 29.1130984 29.1130984
## 300 31.9105461 31.9105461
## 301 30.7715945 30.7715945
## 302 28.9427587 28.9427587
## 303 28.8819102 28.8819102
## 304 32.7988723 32.7988723
## 305 33.2090546 33.2090546
## 306 30.7683179 30.7683179
## 307 35.5622686 35.5622686
## 308 32.7090512 32.7090512
## 309 28.6424424 28.6424424
## 310 23.5896583 23.5896583
## 311 18.5426690 18.5426690
## 312 26.8788984 26.8788984
## 313 23.2813398 23.2813398
## 314 25.5458025 25.5458025
## 315 25.4812006 25.4812006
## 316 20.5390990 20.5390990
## 317 17.6157257 17.6157257
## 318 18.3758169 18.3758169
## 319 24.2907028 24.2907028
## 320 21.3252904 21.3252904
## 321 24.8868224 24.8868224
## 322 24.8693728 24.8693728
## 323 22.8695245 22.8695245
## 324 19.4512379 19.4512379
## 325 25.1178340 25.1178340
## 326 24.6678691 24.6678691
## 327 23.6807618 23.6807618
## 328 19.3408962 19.3408962
```

```
## 329 21.1741811 21.1741811
## 330 24.2524907 24.2524907
## 331 21.5926089 21.5926089
## 332 19.9844661 19.9844661
## 333 23.3388800 23.3388800
## 334 22.1406069 22.1406069
## 335 21.5550993 21.5550993
## 336 20.6187291 20.6187291
## 337 20.1609718 20.1609718
## 338 19.2849039 19.2849039
## 339 22.1667232 22.1667232
## 340 21.2496577 21.2496577
## 341 21.4293931 21.4293931
## 342 30.3278880 30.3278880
## 343 22.0473498 22.0473498
## 344 27.7064791 27.7064791
## 345 28.5479412 28.5479412
## 346 16.5450112 16.5450112
## 347 14.7835964 14.7835964
## 348 25.2738008 25.2738008
## 349 27.5420512 27.5420512
## 350 22.1483756 22.1483756
## 351 20.4594409 20.4594409
## 352 20.5460542 20.5460542
## 353 16.8806383 16.8806383
## 354 25.4025351 25.4025351
## 355 14.3248663 14.3248663
## 356 16.5948846 16.5948846
## 357 19.6370469 19.6370469
## 358 22.7180661 22.7180661
## 359 22.2021889 22.2021889
## 360 19.2054806 19.2054806
## 361 22.6661611 22.6661611
## 362 18.9319262 18.9319262
## 363 18.2284680 18.2284680
## 364 20.2315081 20.2315081
## 365 37.4944739 37.4944739
## 366 14.2819073 14.2819073
## 367 15.5428625 15.5428625
## 368 10.8316232 10.8316232
## 369 23.8007290 23.8007290
## 370 32.6440736 32.6440736
## 371 34.6068404 34.6068404
## 372 24.9433133 24.9433133
## 373 25.9998091 25.9998091
## 374  6.1263250  6.1263250
## 375  0.7777981  0.7777981
## 376 25.3071306 25.3071306
## 377 17.7406106 17.7406106
## 378 20.2327441 20.2327441
## 379 15.8333130 15.8333130
## 380 16.8351259 16.8351259
## 381 14.3699483 14.3699483
## 382 18.4768283 18.4768283
```

```
## 383 13.4276828 13.4276828
## 384 13.0617751 13.0617751
## 385  3.2791812  3.2791812
## 386  8.0602217  8.0602217
## 387  6.1284220  6.1284220
## 388  5.6186481  5.6186481
## 389  6.4519857  6.4519857
## 390 14.2076474 14.2076474
## 391 17.2122518 17.2122518
## 392 17.2988727 17.2988727
## 393  9.8911664  9.8911664
## 394 20.2212419 20.2212419
## 395 17.9418118 17.9418118
## 396 20.3044578 20.3044578
## 397 19.2955908 19.2955908
## 398 16.3363278 16.3363278
## 399  6.5516232  6.5516232
## 400 10.8901678 10.8901678
## 401 11.8814587 11.8814587
## 402 17.8117451 17.8117451
## 403 18.2612659 18.2612659
## 404 12.9794878 12.9794878
## 405  7.3781636  7.3781636
## 406  8.2111586  8.2111586
## 407  8.0662619  8.0662619
## 408 19.9829479 19.9829479
## 409 13.7075637 13.7075637
## 410 19.8526845 19.8526845
## 411 15.2230830 15.2230830
## 412 16.9607198 16.9607198
## 413  1.7185181  1.7185181
## 414 11.8057839 11.8057839
## 415 -4.2813107 -4.2813107
## 416  9.5837674  9.5837674
## 417 13.3666081 13.3666081
## 418  6.8956236  6.8956236
## 419  6.1477985  6.1477985
## 420 14.6066179 14.6066179
## 421 19.6000267 19.6000267
## 422 18.1242748 18.1242748
## 423 18.5217713 18.5217713
## 424 13.1752861 13.1752861
## 425 14.6261762 14.6261762
## 426  9.9237498  9.9237498
## 427 16.3459065 16.3459065
## 428 14.0751943 14.0751943
## 429 14.2575624 14.2575624
## 430 13.0423479 13.0423479
## 431 18.1595569 18.1595569
## 432 18.6955435 18.6955435
## 433 21.5272830 21.5272830
## 434 17.0314186 17.0314186
## 435 15.9609044 15.9609044
## 436 13.3614161 13.3614161
```

```
## 437 14.5207938 14.5207938
## 438  8.8197601  8.8197601
## 439  4.8675110  4.8675110
## 440 13.0659131 13.0659131
## 441 12.7060970 12.7060970
## 442 17.2955806 17.2955806
## 443 18.7404850 18.7404850
## 444 18.0590103 18.0590103
## 445 11.5147468 11.5147468
## 446 11.9740036 11.9740036
## 447 17.6834462 17.6834462
## 448 18.1269524 18.1269524
## 449 17.5183465 17.5183465
## 450 17.2274251 17.2274251
## 451 16.5227163 16.5227163
## 452 19.4129110 19.4129110
## 453 18.5821524 18.5821524
## 454 22.4894479 22.4894479
## 455 15.2800013 15.2800013
## 456 15.8208934 15.8208934
## 457 12.6872558 12.6872558
## 458 12.8763379 12.8763379
## 459 17.1866853 17.1866853
## 460 18.5124761 18.5124761
## 461 19.0486053 19.0486053
## 462 20.1720893 20.1720893
## 463 19.7740732 19.7740732
## 464 22.4294077 22.4294077
## 465 20.3191185 20.3191185
## 466 17.8861625 17.8861625
## 467 14.3747852 14.3747852
## 468 16.9477685 16.9477685
## 469 16.9840576 16.9840576
## 470 18.5883840 18.5883840
## 471 20.1671944 20.1671944
## 472 22.9771803 22.9771803
## 473 22.4558073 22.4558073
## 474 25.5782463 25.5782463
## 475 16.3914763 16.3914763
## 476 16.1114628 16.1114628
## 477 20.5348160 20.5348160
## 478 11.5427274 11.5427274
## 479 19.2049630 19.2049630
## 480 21.8627639 21.8627639
## 481 23.4687887 23.4687887
## 482 27.0988732 27.0988732
## 483 28.5699430 28.5699430
## 484 21.0839878 21.0839878
## 485 19.4551620 19.4551620
## 486 22.2222591 22.2222591
## 487 19.6559196 19.6559196
## 488 21.3253610 21.3253610
## 489 11.8558372 11.8558372
## 490  8.2238669  8.2238669
```

```
## 491  3.6639967  3.6639967
## 492 13.7590854 13.7590854
## 493 15.9311855 15.9311855
## 494 20.6266205 20.6266205
## 495 20.6124941 20.6124941
## 496 16.8854196 16.8854196
## 497 14.0132079 14.0132079
## 498 19.1085414 19.1085414
## 499 21.2980517 21.2980517
## 500 18.4549884 18.4549884
## 501 20.4687085 20.4687085
## 502 23.5333405 23.5333405
## 503 22.3757189 22.3757189
## 504 27.6274261 27.6274261
## 505 26.1279668 26.1279668
## 506 22.3442123 22.3442123
```

Project Y onto the columns of Q one by one and verify it sums to be the projection onto the whole space.

```
yq = Q %*% diag(ncol(Q)) %*% t(Q) %*% y
yq_columns = matrix(nrow = nrow(Q), ncol = 0)
for(j in 1:ncol(Q)){
  yq_columns = cbind(yq_columns, Q[, j] %*% t(Q[, j]) %*% y)
}
cbind(head(yq), head(as.matrix(rowSums(yq_columns))))
```

```
##          [,1]     [,2]
## [1,] 30.00384 30.00384
## [2,] 25.02556 25.02556
## [3,] 30.56760 30.56760
## [4,] 28.60704 28.60704
## [5,] 27.94352 27.94352
## [6,] 25.25628 25.25628
```

Verify the OLS fit squared lengh is the sum of squared lengths of each of the orthogonal projections.

```
sum(t(yq_columns) %*% yq_columns)
```

```
## [1] 288547.6
```

```
t(yhat) %*% yhat
```

```
##          [,1]
## [1,] 288547.6
```

Rewrite the "The monotonicity of SSR" demo from the lec06 notes. Comment every line in detail. Write about what the plots means.

```
n = 100
y = rnorm(n)
RMSE = array(NA, n)
```

```
X = matrix(NA, nrow = n, ncol = 0)
X = cbind(1, X)

RMSE[1] =summary(lm(y ~ X))$sigma

for (j in 2 : n){
  X = cbind(X, rnorm(n))
  RMSE[j] = summary(lm(y ~ X))$sigma

pacman::p_load(ggplot2)
base = ggplot(data.frame(j = 1 : n, RMSEs = RMSE))
base + geom_line(aes(x = j, y = RMSE))
}
```

Rewrite the "Overfitting" demo from the lec06 notes. Comment every line in detail. Write about what the plots means.

```
#TO-DO
```