Longley

DP

September 6, 2016

To avoid filling up the handout with long summaries I'll use a shortened version, which I call look(). The code is hidden in the pdf file, but is viewable in the Rmd file.

The famous Longley dataset

```
data(longley) # we now have a 16 by 7 data frame called longley
# Also look at  pairs(longley)
```

```
##      GNP.deflator GNP Unemployed Armed.Forces Population Year Employed
## 1947         83.0 234        236          159        108 1947     60.3
## 1948         88.5 259        232          146        109 1948     61.1
## 1949         88.2 258        368          162        110 1949     60.2
```

Longley used this small set of data to illustrate the potentially nasty effects of round-off errors when working with predictors that are "almost linearly dependent". I'll use it to illustrate the effects that small changes can have on a least squares fit when the matrix of predictors is "ill-conditioned".

```
out0 <- lm(Employed ~. , data = longley) ; look(out0)
```

```
## Residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -0.410  -0.158  -0.028   0.000   0.102   0.455
## Rsquared:  0.995
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -3482.259    890.420  -3.911    0.004
## GNP.deflator     0.015      0.085   0.177    0.863
## GNP             -0.036      0.033  -1.070    0.313
## Unemployed      -0.020      0.005  -4.136    0.003
## Armed.Forces    -0.010      0.002  -4.822    0.001
## Population      -0.051      0.226  -0.226    0.826
## Year             1.829      0.455   4.016    0.003
```

A lot of the work is being done by **1** (a column of ones) and the longley$Year predictor:

```
outYear <- lm(Employed ~ Year, data = longley) ; look(outYear)   #summary(outYear)
```

```
## Residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -1.310  -0.709   0.210   0.000   0.424   1.470
## Rsquared:  0.943
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1335.105     91.607   -14.6        0
## Year            0.717      0.047    15.3        0
```

The other predictors improved the fit by only a small amount. Compare totalSS $= \sum_i (y_i - \overline{y})^2$ with the sum of squared residuals for both fits:

```
##   totalSS outYearSS    res0SS
##   185.009    10.457     0.836
```

Remove time trend from five predictors

Replace the columns "GNP.deflator", "GNP","Unemployed", "Armed.Forces", and "Population" by their components orthogonal to **1** and "Year".

```
L1 <- longley
for (nn in names(longley)[1:5]){
  L1[[nn]] <- lm(longley[[nn]] ~ longley[["Year"]])$residual
}
out1 <- lm(Employed ~ . , L1)
```

Look at pairs(L1) to see the effect. Then compare residuals and coefficients with out0:

```
##       Min. 1st Qu.  Median Mean 3rd Qu.  Max.
## out1 -0.41  -0.158 -0.0282    0   0.102 0.455
## out0 -0.41  -0.158 -0.0282    0   0.102 0.455
```

```
##      (Intercept) GNP.deflator     GNP Unemployed Armed.Forces Population
## out1       -1335       0.0151 -0.0358    -0.0202      -0.0103    -0.0511
## out0       -3482       0.0151 -0.0358    -0.0202      -0.0103    -0.0511
##       Year
## out1 0.717
## out0 1.829
```

The least squares fitted vector is the same but some of the coefficients have changed. Why?

Effect of small perturbations

Now I'll make some small changes that, I hope, will have a bad effect on the bhat. Manufacture a small perturbation as suggested on the SVD.pdf handout:

```
Xsvd <- svd(L1[,1:5])
U <- Xsvd$u; U5 <- U[,5]
V <- Xsvd$v; V5 <- V[,5]

E <- U5  %*% t(V5)  ;   dimnames(E) <- dimnames(longley[,1:5])
yincr <- U5 %*% t(U5) %*% y
```

```
L3 <- L2 <- L1 # make changes on  copies
L2[,1:5] <- L1[,1:5] - E
L3[,1:5] <- L1[,1:5] - 2*E
L4 <- L2; L4$Employed <- y + yincr
L5 <- L3; L5$Employed <- y + 2*yincr
# compare:
#  round(L,2); round(LL,2)
```

```
out2 <- lm(Employed ~. , data = L2)
out3 <- lm(Employed ~. , data = L3)
out4 <- lm(Employed ~. , data = L4)
out5 <- lm(Employed ~. , data = L5)
```

Compare the original longley data with the fake data set obtained by applying the L5 perturbations to longley. Do you think any of the differences are particularly noteworthy?

```
##               1947  1948  1949  1950  1951  1952  1953  1954
## GNP.deflator  83.0  88.5  88.2  89.5  96.2  98.1  99.0 100.0
## fake_GNP.d    82.9  88.4  88.2  89.7  96.4  98.1  98.8  99.9
## GNP          234.3 259.4 258.1 284.6 329.0 347.0 365.4 363.1
## fake_GNP     234.3 259.5 258.1 284.5 328.9 347.0 365.5 363.2
## Unemployed   235.6 232.5 368.2 335.1 209.9 193.2 187.0 357.8
## fake_Unemp   235.6 232.5 368.2 335.1 209.9 193.2 187.0 357.8
## Armed.Forces 159.0 145.6 161.6 165.0 309.9 359.4 354.7 335.0
## fake_Armed   159.0 145.6 161.6 165.0 309.9 359.4 354.7 335.0
## Population   107.6 108.6 109.8 110.9 112.1 113.3 115.1 116.2
## fake_Popul   107.4 108.1 109.7 111.5 112.8 113.4 114.3 115.9
## Employed      60.3  61.1  60.2  61.2  63.2  63.6  65.0  63.8
## fake_Emplo    60.3  61.1  60.2  61.2  63.3  63.6  64.9  63.7
```

```
##               1955  1956  1957  1958  1959  1960  1961  1962
## GNP.deflator 101.2 104.6 108.4 110.8 112.6 114.2 115.7 116.9
## fake_GNP.d   101.4 104.6 108.3 110.8 112.8 114.2 115.6 116.8
## GNP          397.5 419.2 442.8 444.6 482.7 502.6 518.2 554.9
## fake_GNP     397.4 419.2 442.8 444.6 482.6 502.6 518.2 554.9
## Unemployed   290.4 282.2 293.6 468.1 381.3 393.1 480.6 400.7
## fake_Unemp   290.4 282.2 293.6 468.1 381.3 393.1 480.6 400.7
## Armed.Forces 304.8 285.7 279.8 263.7 255.2 251.4 257.2 282.7
## fake_Armed   304.8 285.7 279.8 263.7 255.2 251.4 257.2 282.7
## Population   117.4 118.7 120.4 122.0 123.4 125.4 127.8 130.1
## fake_Popul   118.2 118.9 120.1 121.8 124.1 125.4 127.5 129.8
## Employed      66.0  67.9  68.2  66.5  68.7  69.6  69.3  70.5
## fake_Emplo    66.1  67.9  68.2  66.5  68.7  69.6  69.3  70.5
```

Then compare their least squares fits. Larger values in the last column of the coefficients dispaly suggest we should not take a coefficient too seriously. (More about that idea soon.) First longley then fake:

```
## Residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -0.410  -0.158  -0.028   0.000   0.102   0.455
## Rsquared:  0.995
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -3482.259    890.420  -3.911    0.004
## GNP.deflator     0.015      0.085   0.177    0.863
## GNP             -0.036      0.033  -1.070    0.313
## Unemployed      -0.020      0.005  -4.136    0.003
## Armed.Forces    -0.010      0.002  -4.822    0.001
## Population      -0.051      0.226  -0.226    0.826
## Year             1.829      0.455   4.016    0.003
```

```
## Residuals:
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  -0.410  -0.158  -0.028   0.000   0.102   0.455
## Rsquared:  0.995
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -3894.072   1055.762  -3.688    0.005
## GNP.deflator     0.083      0.126   0.657    0.528
## GNP             -0.068      0.055  -1.226    0.251
## Unemployed      -0.024      0.008  -3.221    0.010
## Armed.Forces    -0.011      0.002  -4.805    0.001
## Population       0.202      0.416   0.486    0.638
## Year             2.028      0.532   3.815    0.004
```

The point is that some of the coefficients are very sensitive to changes in the data almost at the level of round-off error.

Just for the comparison, here are the coefficients for the fits with time trends removed:

```
##         (Intercept) GNP.deflator   GNP Unemployed Armed.Forces Population
## longley       -3482         0.02 -0.04      -0.02        -0.01      -0.05
## L1            -1335         0.02 -0.04      -0.02        -0.01      -0.05
## L2            -1335        -0.02 -0.02      -0.02        -0.01      -0.18
## L3            -1335         0.04 -0.05      -0.02        -0.01       0.06
## L4            -1335        -0.07  0.00      -0.02        -0.01      -0.35
## L5            -1335         0.08 -0.07      -0.02        -0.01       0.20
##         Year
## longley 1.83
## L1      0.72
## L2      0.72
## L3      0.72
## L4      0.72
## L5      0.72
```

You might also find it interesting to look at the least squares in more detail.

```
#look(out0);look(out1);look(out2);look(out3);look(out4);look(out5)
```