University of California, Los Angeles
Department of Statistics

Statistics 100C **Instructor: Nicolas Christou**

Deleting a single point in regression

In this document we will explore the effect of deleting a single point in multiple regression. Let's partition the vector $\mathbf{y}$, the matrix $\mathbf{X}$, and the vector $\boldsymbol{\epsilon}$ as follows:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{Y_{(i)}} \\ Y_i \end{pmatrix} = \mathbf{X}\boldsymbol{\beta} + \begin{pmatrix} \boldsymbol{\epsilon_{(i)}} \\ \epsilon_i \end{pmatrix} = \begin{pmatrix} \mathbf{X_{(i)}} \\ \mathbf{x_i'} \end{pmatrix} \boldsymbol{\beta} + \begin{pmatrix} \boldsymbol{\epsilon_{(i)}} \\ \epsilon_i \end{pmatrix}.$$

Some notation: The subscript (i) means that the ith data point is removed, and $\mathbf{x_i}$ is the ith row of the $\mathbf{X}$ matrix. We know already the solution of least squares when none of the points is removed. The usual OLS solution is: $\hat{\boldsymbol{\beta}} = (\mathbf{X'X})^{-1}\mathbf{X'Y}$. The model we are using here is:

$$\mathbf{Y_{(i)}} = \mathbf{X_{(i)}}\boldsymbol{\beta} + \boldsymbol{\epsilon_{(i)}}.$$

For OLS we will have to simplify $(\mathbf{X_{(i)}}'\mathbf{X_{(i)}})^{-1}$. From the partition of $\mathbf{X}$ above we can get:

$$\mathbf{X'X} = \begin{pmatrix} \mathbf{X_{(i)}}' & \mathbf{x_i} \end{pmatrix} \begin{pmatrix} \mathbf{X_{(i)}} \\ \mathbf{x_i'} \end{pmatrix} = \mathbf{X'_{(i)}X_{(i)}} + \mathbf{x_{(i)}x'_{(i)}} \Rightarrow \mathbf{X'_{(i)}X_{(i)}} = \mathbf{X'X} - \mathbf{x_{(i)}x'_{(i)}}.$$

A useful result from linear algebra will be used here. Let $\mathbf{A}$ be a matrix and $\mathbf{b}$ be a vector. Then,

$$[\mathbf{A} - \mathbf{bb'}]^{-1} = \mathbf{A}^{-1} + \frac{\mathbf{A}^{-1}\mathbf{bb'}\mathbf{A}^{-1}}{1 - \mathbf{b'}\mathbf{A}^{-1}\mathbf{b}}, \text{ provided that } \mathbf{A} \text{ is invertible and } 1 - \mathbf{b'}\mathbf{A}^{-1}\mathbf{b} \neq 0.$$

Similarly,

$$[\mathbf{A} + \mathbf{bb'}]^{-1} = \mathbf{A}^{-1} - \frac{\mathbf{A}^{-1}\mathbf{bb'}\mathbf{A}^{-1}}{1 + \mathbf{b'}\mathbf{A}^{-1}\mathbf{b}}, \text{ provided that } \mathbf{A} \text{ is invertible and } 1 + \mathbf{b'}\mathbf{A}^{-1}\mathbf{b} \neq 0.$$

We can now use the first result to find the inverse of $(\mathbf{X_{(i)}}'\mathbf{X_{(i)}})^{-1}$.

$$\begin{aligned} [\mathbf{X_{(i)}}'\mathbf{X_{(i)}}]^{-1} &= [\mathbf{X'X} - \mathbf{x_{(i)}x'_{(i)}}]^{-1} \\ &= (\mathbf{X'X})^{-1} + \frac{(\mathbf{X'X})^{-1}\mathbf{x_i}\mathbf{x_i'}(\mathbf{X'X})^{-1}}{1 - \mathbf{x_i'}(\mathbf{X'X})^{-1}\mathbf{x_i}}, \end{aligned}$$

The denominator of the last term of the previous expression is the leverage value h_{ii}. Therefore,

$$[\mathbf{X_{(i)}}'\mathbf{X_{(i)}}]^{-1} = (\mathbf{X'X})^{-1} + \frac{(\mathbf{X'X})^{-1}\mathbf{x_i}\mathbf{x_i'}(\mathbf{X'X})^{-1}}{1 - h_{ii}}.$$

Now let's compute the estimate of the $\boldsymbol{\beta}$ vector, which after the deletion of data point i will be denoted with $\boldsymbol{\beta}_{(i)}$. The OLS vector will be denoted $\hat{\boldsymbol{\beta}}_{(i)}$.

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{(i)} &= [\mathbf{X_{(i)}}'\mathbf{X_{(i)}}]^{-1}\mathbf{X'_{(i)}Y_{(i)}} \\ &= \left[(\mathbf{X'X})^{-1} + \frac{(\mathbf{X'X})^{-1}\mathbf{x_i'}\mathbf{x_i}(\mathbf{X'X})^{-1}}{1 - h_{ii}}\right]\mathbf{X'_{(i)}Y_{(i)}} \end{aligned}$$

We can now replace $\mathbf{X'_{(i)}Y_{(i)}}$ as follows:

$$\mathbf{X'Y} = \begin{pmatrix} \mathbf{X_{(i)}}' & \mathbf{x_i} \end{pmatrix} \begin{pmatrix} \mathbf{Y_{(i)}} \\ y_i \end{pmatrix} = \mathbf{X'_{(i)}Y_{(i)}} + \mathbf{x_{(i)}}y_i \Rightarrow \mathbf{X'_{(i)}Y_{(i)}} = \mathbf{X'Y} - \mathbf{x_{(i)}}y_i$$

Therefore,

$$\begin{aligned} \hat{\boldsymbol{\beta}}_{(i)} &= \left[(\mathbf{X'X})^{-1} + \frac{(\mathbf{X'X})^{-1}\mathbf{x_i}\mathbf{x_i'}(\mathbf{X'X})^{-1}}{1 - h_{ii}}\right][\mathbf{X'Y} - \mathbf{x_{(i)}}'y_i] \\ &= \hat{\boldsymbol{\beta}} - (\mathbf{X'X})^{-1}\mathbf{x_i}y_i \\ &+ \frac{(\mathbf{X'X})^{-1}\mathbf{x_i}\mathbf{x_i'}\hat{\boldsymbol{\beta}}}{1 - h_{ii}} \\ &- \frac{(\mathbf{X'X})^{-1}\mathbf{x_i}h_{ii}y_i}{1 - h_{ii}} \end{aligned}$$

$$\begin{aligned}\hat{\boldsymbol{\beta}}_{(i)} &= \hat{\boldsymbol{\beta}} - \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}y_i(1-h_{ii})}{1-h_{ii}} + \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}\mathbf{x_i}'\hat{\boldsymbol{\beta}}}{1-h_{ii}} - \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}h_{ii}y_i}{1-h_{ii}} \\ &= \hat{\boldsymbol{\beta}} - \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}y_i}{1-h_{ii}} + \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}\mathbf{x_i}'\hat{\boldsymbol{\beta}}}{1-h_{ii}} \\ &= \hat{\boldsymbol{\beta}} - \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}}{1-h_{ii}}[y_i - \mathbf{x_i}'\hat{\boldsymbol{\beta}}]\end{aligned}$$

Note: We recognize that $e_i = y_i - \mathbf{x_i}'\hat{\boldsymbol{\beta}}$. Therefore,

$$\hat{\boldsymbol{\beta}}_{(i)} = \hat{\boldsymbol{\beta}} - \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}}{1-h_{ii}}e_i, \text{ and the influence of the } ith \text{ data point on the vector } \hat{\boldsymbol{\beta}} \text{ is given by}$$

$$\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)} = \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}}{1-h_{ii}}e_i.$$

The vector $\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)}$ is often called $DFBETA_i$. If we let $(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i} = (\alpha_{0i}, \alpha_{2i}, \ldots, \alpha_{ki})'$ then the individual component of $DFBETA_i$ is equal to:

$$DFBETA_{ij} = \hat{\beta}_j - \hat{\beta}_j(i) = \alpha_{ji}\frac{e_i}{1-h_{ii}}, \text{ for } i = 1, \ldots, n, \text{ and } j = 0, \ldots, k.$$

The quantity, $\mathbf{X}(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})$ has the units of $\hat{\mathbf{Y}}$ and its squared length is equal to:

$$\begin{aligned}[\mathbf{X}(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})]'[(\mathbf{X}\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})] &= (\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)})'\mathbf{X}'\mathbf{X}(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)}) = \\ &= \frac{e_i^2}{(1-h_{ii})^2}\mathbf{x_i}'(\mathbf{X}'\mathbf{X})^{-1}(\mathbf{X}'\mathbf{X})(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}] = \frac{h_{ii}}{(1-h_{ii})^2}e_i^2.\end{aligned}$$

This squared length is the basis for Cook's distance. It is computed as follows:

$$D_i = \frac{h_{ii}}{(1-h_{ii})^2}\frac{e_i^2}{(k+1)s_e^2}.$$

We can also compute the effect of deleting a data point on the predicted value $\hat{y}_i$. The new predicted value is denoted with $\hat{y}_i(i)$, and the difference between the two predicted vectors is denoted with $DFFITS_i$ and it is computed as follows:

$$DFFITS_i = \hat{y}_i - \hat{y}_i(i) = \mathbf{x_i}'\hat{\boldsymbol{\beta}} - \mathbf{x_i}'\hat{\boldsymbol{\beta}}_{(i)} = \mathbf{x_i}'(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)}) = \mathbf{x_i}'\frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}}{1-h_{ii}}e_i = \frac{h_{ii}}{1-h_{ii}}e_i.$$

We would also like to develop an expression that connects s_e^2 and $s_e^2(i)$, where s_e^2 is the unbiased estimate of σ^2 using all the n data points and $s_e^2(i)$ is the unbiased estimate of σ^2 when the ith data point is deleted. Clearly $s_e^2(i)$ is

$$s_e^2(i) = \frac{1}{n-k-2}\sum_{l=1, l\neq i}^{n}(y_l - \mathbf{x}_l'\hat{\boldsymbol{\beta}}_{(i)})^2,$$

and it should have the properties of s_e^2, i.e., it is unbiased, and also $\frac{(n-k-2)s_e^2(i)}{\sigma^2} \sim \chi^2_{n-k-2}$. The expression of $s_e^2(i)$ can be expressed in terms of s_e^2, e_i, h_{ii} as follows:

The matrix $\mathbf{H}$ is idempotent, therefore $\mathbf{HH} = \mathbf{H}$, which implies that $\sum_{l=1}^{n} h_{il}^2 = hii$. Also, since $\mathbf{He} = \mathbf{H}(\mathbf{I} - \mathbf{H})\mathbf{y} = \mathbf{0}$ it follows that $\sum_{l=1}^{n} h_{il} e_l = 0$. Using these results and also the result from above, $\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)} = \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}'}{1-h_{ii}} e_i$ we get:

$$\begin{aligned}
\sum_{l=1, l\neq i}^{n} (y_l - \mathbf{x}_l'\hat{\boldsymbol{\beta}}_{(i)})^2 &= \sum_{l=1, l\neq i}^{n} (y_l - \mathbf{x}_l'\hat{\boldsymbol{\beta}} + \mathbf{x}_l'\hat{\boldsymbol{\beta}} - \mathbf{x}_l'\hat{\boldsymbol{\beta}}_{(i)})^2 \\
&= \sum_{l=1, l\neq i}^{n} (e_l + \mathbf{x}_l'(\hat{\boldsymbol{\beta}} - \hat{\boldsymbol{\beta}}_{(i)}))^2 = \sum_{l=1, l\neq i}^{n} \left(e_l + \mathbf{x}_l' \frac{(\mathbf{X}'\mathbf{X})^{-1}\mathbf{x_i}'}{1-h_{ii}} e_i \right)^2 \\
&= \sum_{l=1, l\neq i}^{n} (e_l + h_{il}\frac{e_i}{1-h_{ii}})^2 = \sum_{l=1}^{n} (e_l + h_{il}\frac{e_i}{1-h_{ii}})^2 - (e_i + h_{ii}\frac{e_i}{1-h_{ii}})^2 \\
&= \sum_{l=1}^{n} e_l^2 + \frac{e_i^2}{(1-h_{ii})^2} \sum_{l=1}^{n} h_{il}^2 + 2\frac{e_i}{1-h_{ii}} \sum_{l=1}^{n} e_l h_{il} - \frac{e_i^2}{(1-h_{ii})^2} \\
&= \sum_{l=1}^{n} e_l^2 - \frac{e_i^2}{1-h_{ii}}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
s_e^2(i) &= \frac{1}{n-k-2} \sum_{l=1, l\neq i}^{n} (y_l - \mathbf{x}_l'\hat{\boldsymbol{\beta}}_{(i)})^2 = \frac{1}{n-k-2} \left(\sum_{l=1}^{n} e_l^2 - \frac{e_i^2}{1-h_{ii}} \right) \Rightarrow \\
(n-k-2)s_e(i)^2 &= (n-k-1)s_e^2 - \frac{e_i^2}{1-h_{ii}}.
\end{aligned}$$

Example:
Let's compute some of the expressions above.

```
a <- read.table("http://www.stat.ucla.edu/~nchristo/statistics100C/rain_wheat.txt", header=TRUE)
```

```
   rain wheat
1    12   310
2    14   320
3    13   323
4    16   330
5    18   334
6    20   348
7    19   352
8    22   360
9    22   370
10   20   344
11   23   370
12   24   380
13   26   385
14   27   393
15   28   395
16   29   400
17   30   403
18   31   406
19   26   383
20   27   388
21   28   392
22   29   398
23   30   400
24   31   403
25   20   270
26   50   260
```

Let's compute the $DFBETA_i$ vector, $DFFITS_i$ vector, Cook's distance, and $s_e^2(i)$:

```
k <- 1
ones <- rep(1, nrow(a))
X <- as.matrix(cbind(ones, a$rain))
H <- X %*% solve(t(X) %*% X) %*% t(X)
betahat <- solve(t(X) %*% X) %*% t(X) %*% a$wheat
se2 <- (t(a$wheat) %*% a$wheat - t(betahat) %*% t(X) %*% a$wheat) / (nrow(a)-k-1)

e <- a$wheat - X %*% betahat
h <- diag(H)

#Compute DFBETAi vector:
dfbeta <- c(0,0)
for(i in 1:26){
dfb <- t( solve(t(X) %*% X) %*% X[i,] * e[i]/(1-h[i]) )
dfbeta <- rbind(dfbeta, dfb)
}

#Compute DFFITSi vector:
dffits <- rep(0,26)
for(i in 1:26){
dffits[i] <- h[i]*e[i]/(1-h[i])
}

#Compute Cook's distance:
D <- rep(0,26)
for(i in 1:26){
D[i] <- h[i]*e[i]^2/((1-h[i])^2*(k+1)*se2)
}

#Compute se^2(i):
se2i <- rep(0,26)
for(i in 1:26){
se2i[i] <- ( (nrow(a)-k-1)*se2-e[i]^2/(1-h[i]) )/(nrow(a)-k-2)
}
```

```
> head(dfbeta[-1,])
       ones
 -10.656807 0.36677131
  -7.090793 0.23679665
  -6.706132 0.22762529
  -4.333757 0.13865488
  -3.193471 0.09566092
  -1.063000 0.02839604

> dffits
 [1]   -6.2555515   -3.7756396   -3.7470035   -2.1152784   -1.4715742   -0.4950788
 [7]   -0.2445059    0.0261643    0.4687776   -0.7124230    0.3915380    0.7341148
[13]    0.8776548    1.2481878    1.4157437    1.8021571    2.1665339    2.6187348
[19]    0.7940537    1.0240767    1.2677504    1.6914258    1.9775995    2.4020277
[25]   -4.7332902 -119.3509570

> D
 [1] 8.323144e-02 3.865679e-02 3.364939e-02 1.569233e-02 9.877095e-03 1.432397e-03
 [7] 3.098183e-04 4.864369e-06 1.561500e-03 2.966132e-03 1.159646e-03 4.207036e-03
[13] 5.781940e-03 1.093786e-02 1.284058e-02 1.864538e-02 2.386102e-02 3.065808e-02
[19] 4.732884e-03 7.362709e-03 1.029634e-02 1.642449e-02 1.988084e-02 2.579396e-02
[25] 1.309305e-01 9.019692e+00

> se2i
 [1] 1659.489 1687.662 1698.529 1708.213 1712.159 1728.521 1731.602 1732.311
 [9] 1727.234 1724.446 1728.290 1717.193 1712.359 1697.099 1694.751 1683.711
[17] 1677.632 1671.058 1715.982 1708.614 1702.196 1689.502 1686.755 1680.779
[25] 1384.469  296.899
```

All the above can be obtained much easier using:

```
q <- lm(a$wheat ~ a$rain )
> influence(q)
```