Advanced Methods Problem Set 2

Bohao Tang

1 Inference

1: Denote $\vec{\beta} = \begin{pmatrix} \vec{\beta}_1 \\ \vec{\beta}_2 \end{pmatrix}$ and L for $L'\vec{\beta} = \vec{\beta}_2$

Then $XL = X_2$

Now the Likelihood $f(\vec{y};\vec{\beta}) = \left(\frac{1}{\sqrt{2\pi}\,\sigma}\right)^{n} e^{-\frac{(\vec{Y}-X\vec{\beta})'(\vec{Y}-X\vec{\beta})}{2\sigma^2}}$

$$\ell(\vec{\beta}) = \log f(\vec{y},\vec{\beta}) = -n \ln\sqrt{2\pi}\,\sigma - \frac{1}{2\sigma^2}(\vec{Y}-X\vec{\beta})'(\vec{Y}-X\vec{\beta})$$

So $\vec{s}(\vec{\beta}) = \frac{\partial \ell(\vec{\beta})}{\partial \vec{\beta}} = \frac{1}{\sigma^2} X'(\vec{Y}-X\vec{\beta})$

and $\mathcal{J}(\vec{\beta}) = -E\left[\frac{\partial^2 \ell(\vec{\beta})}{\partial \vec{\beta}^2}\right] = E\left[\frac{1}{\sigma^2}X'X\right] = \frac{X'X}{\sigma^2}$

Now $\hat{\beta} = (X'X)^{-1}X'\vec{Y}$, $\ell(\hat{\beta}) = -n\ln\sqrt{2\pi}\,\sigma - \frac{1}{2\sigma^2}\vec{Y}'(I-H_X)\vec{Y}$

where $H_X = X(X'X)^{-1}X'$

$\tilde{\beta} = \begin{bmatrix} (X_1'X_1)^{-1}X_1'\vec{Y} \\ \vec{0} \end{bmatrix}$, $\ell(\tilde{\beta}) = -n\ln\sqrt{2\pi}\,\sigma - \frac{1}{2\sigma^2}\vec{Y}'(I-H_{X_1})\vec{Y}$

where $H_{X_1} = X_1(X_1'X_1)^{-1}X_1'$

so the likelihood ratio test statistic is

$$T_{lr} = \frac{1}{\sigma^2}\vec{Y}'(H_X - H_{X_1})\vec{Y}$$

And the Wald test statistic is

$$T_W = (L'\hat{\beta} - \vec{0})' \left(L' \frac{(X'X)^{-1}}{\sigma^2} L \right)^{-1} (L'\hat{\beta} - \vec{0})$$

Recall that $X = [X_1\ X_2]$ then $(X'X)^{-1} =$

$$\begin{pmatrix} (X_1'X_1)^{-1} + FE^{-1}F', & -FE^{-1} \\ -E^{-1}F', & E^{-1} \end{pmatrix} \quad ①$$

where $E = X_2'(I - H_{X_1})X_2$ and $F = (X_1'X_1)^{-1}X_1'X_2$.

Then $T_W = \vec{Y}'[X_1\ X_2]\left(\cdots \right) L \left(L' \frac{(\cdots)}{\sigma^2} L \right)^{-1} L' \left(\cdots \right) X' \vec{Y}$

$$= \vec{Y}'[X_1, X_2] \begin{bmatrix} -FE^{-1} \\ E^{-1} \end{bmatrix} \left[\left(\frac{E}{\sigma^2} \right)^{-1} \right]^{-1} [-E^{-1}F', E^{-1}] \begin{bmatrix} X_1' \\ X_2' \end{bmatrix} \vec{Y}$$

$$= \frac{1}{\sigma^2} \vec{Y}' (I - H_{X_1}) X_2 [X_2'(I - H_{X_1}) X_2]^{-1} X_2' (I - H_{X_1}) \vec{Y}$$

And the score test statistic is

$$T_S = \vec{S}(\tilde{\beta})' \mathcal{J}(\tilde{\beta})^{-1} \vec{S}(\tilde{\beta})$$

$$\cancel{= \frac{1}{\sigma^2} \vec{Y}' [I - H_{X_1}] [X'X]^{-1} [I - H_{X_1}] \vec{Y}}$$

$$= \frac{1}{\sigma^2} \vec{Y}' (I - H_{X_1}) [X_1\ X_2] [X'X]^{-1} \begin{bmatrix} X_1' \\ X_2' \end{bmatrix} [I - H_{X_1}] \vec{Y}$$

$$= \frac{1}{\sigma^2} \vec{Y}' (I - H_{X_1}) H_X (I - H_{X_1}) \vec{Y}$$

$$= \frac{1}{\sigma^2} \vec{Y}' (H_X - H_{X_1}) \vec{Y}$$

~~use~~ Comment: use $(X'X)^{-1} = \begin{pmatrix} (X_1'X_1)^{-1}+FE^{-1}F', & -FE^{-1} \\ -E^{-1}F' & , E^{-1} \end{pmatrix}$

to write out $H_X - H_{X_1}$

we find that $H_X - H_{X_1} = (I-H_{X_1}) X_2 [X_2'(I-H_{X_1})X_2]^{-1} X_2'(I-H_{X_1})$

So we have that $T_{lr} = T_W = T_S$

three tests are equivalent.

2 Residuals:

1: A: $(\hat{\vec{\beta}}, \hat{\Delta})$ is the minimum of $\|\vec{y} - X\vec{\beta} - \vec{\delta}\Delta\|^2$

use derivative ~~use~~ of Δ we get a normal equation.

$$\vec{\delta}'(\vec{y} - X\hat{\vec{\beta}} - \vec{\delta}\hat{\Delta}) = 0$$

since $\delta_i = 0$ $i \neq i_0$ and $\delta_{i_0} = 1$

$$\vec{\delta}'(\vec{y} - X\hat{\vec{\beta}} - \vec{\delta}\hat{\Delta}) = (\vec{y} - X\hat{\vec{\beta}} - \vec{\delta}\hat{\Delta})_{i_0} = \hat{\varepsilon}_{i_0} = 0$$

Therefore i_0 ~~th~~ residual is 0.

B: Denote $\hat{\vec{\beta}}_{\Delta}$ be the estimate of $\vec{\beta}$ using whole data and model with Δ

and $\hat{\beta}_{-i_0}$ be the estimate of $\vec{\beta}$ using data without i_0th and usual model.

$\vec{y}_{-i_0}$ be the $\vec{y}$ without i_0th element.

~~Then~~ and X_{i_0} be the ith row of X

Then:

$$\begin{pmatrix} \hat{\beta}_\Delta \\ \hat{\Delta} \end{pmatrix} = \left([X,\delta]'[X,\delta]\right)^{-1} \begin{bmatrix} X' \\ \delta' \end{bmatrix} \vec{y}$$

$$= \begin{pmatrix} X'X, & X_{i_0}' \\ X_{i_0}, & 1 \end{pmatrix}^{-1} \begin{pmatrix} X'\vec{y} \\ \vec{y}_{i_0} \end{pmatrix}$$

$$\Rightarrow \hat{\beta}_\Delta = \left[(X'X)^{-1} + FE^{-1}F'\right] X'\vec{y} - FE^{-1} y_{i_0}$$

where $F = (X'X)^{-1}X'\delta = (X'X)^{-1}X_{i_0}'$

$E = \delta'(I-H_X)\delta = (I-H_X)_{i_0,i_0}$

$$\Rightarrow \hat{\beta}_\Delta = (X'X)^{-1}X'\vec{y} + \frac{X_{i_0}'X_{i_0}}{(I-H_X)_{i_0i_0}} (X'X)^{-1}(X'X)^{-1}X'\vec{y}$$

$$- \frac{y_{i_0}}{(I-H_X)_{i_0i_0}} (X'X)^{-1}X_{i_0}'$$

$$= (X'X)^{-1}X'\vec{y} + \frac{(X'X)^{-1}X_{i_0}'}{(I-H_X)_{i_0i_0}} \left(X_{i_0}(X'X)^{-1}X'\vec{y} - y_{i_0}\right)$$

denote $\hat{\vec{y}}$, $\hat{\beta}$ be the fitted value of $\vec{y}$ and $\vec{\beta}$ using full data in model $\vec{y} = X\vec{\beta} + \vec{\varepsilon}$

Then we have $\hat{\beta}_\Delta = \hat{\beta} - \dfrac{(X'X)^{-1}X_{i_0}'}{1 - H_{X\,i_0i_0}} (y_{i_0} - \hat{y}_{i_0})$

recall the result in slides of Lecture 4. we have that

$\hat{\beta}_{-i_0} = \hat{\beta}_\Delta$ which ends the proof.

C: $\delta'\vec{y} = \delta' X\vec{\beta} + \Delta + \delta'\vec{\varepsilon}$

$\Rightarrow \Delta = (\delta'\vec{y} - \delta' X\vec{\beta}) - \delta'\vec{\varepsilon}$

$\Rightarrow$ to test $\Delta=0$ we ~~cant~~ can use its $\overset{\text{standardized}}{\vee}$ mean value

while its mean value is just $\delta'\vec{y} - \delta' X\vec{\beta}_\Delta$

$= y_{i_0} - X_{i_0}\vec{\beta}_{-i_0}$ = PRESS residuals of i_0

So standardized i_0 th PRESS residuals can be a test statistic. for $\Delta = 0$

2: internally studentized residuals $r_{i_0} = \dfrac{e_{i_0}}{S\sqrt{1-h_{i_0i_0}}}$

externally studentized residuals $t_{i_0} = \dfrac{e_{i_0}}{S_{(i_0)}\sqrt{1-h_{i_0i_0}}}$

so $\dfrac{t_{i_0}}{r_{i_0}} = \dfrac{S}{S_{(i_0)}} = \sqrt{\dfrac{\vec{Y}'(I-H_X)\vec{Y}/(n-p)}{\vec{Y}'_{-i_0}(I-H_{X_{-i_0}})\vec{Y}_{-i_0}/(n-p-1)}}$ X_{-i_0} is X without i_0th row.

notice that $\vec{Y}'_{-i_0}(I-H_{X_{-i_0}})\vec{Y}_{-i_0} = \|\vec{Y}_{-i_0} - X_{-i_0}\hat{\vec{\beta}}_{-i_0}\|^2$

$= \left\|\vec{e}_{-i_0} + \dfrac{X_{-i_0}(X'X)^{-1}X'_{i_0}}{1-h_{i_0i_0}} e_{i_0}\right\|^2$ $\vec{e}_{-i_0}$ is the $\overset{\text{original}}{\vee}$ residual $\vec{e}$ without i_0th element.

$= \|\vec{e}_{-i_0}\|^2 + 2\dfrac{\vec{e}'_{-i_0}X_{-i_0}(X'X)^{-1}X'_{i_0}e_{i_0}}{1-h_{i_0i_0}} + \dfrac{X_{i_0}(X'X)^{-1}X'_{-i_0}X_{-i_0}(X'X)^{-1}X'_{i_0}}{(1-h_{i_0i_0})^2}e_{i_0}^2$

$= \|\vec{e}\|^2 + \left[\dfrac{X_{i_0}(X'X)^{-1}X'_{i_0}X_{-i_0}(X'X)^{-1}X'_{i_0}}{(1-h_{i_0i_0})^2} - 1\right]e_{i_0}^2 + \dfrac{2\vec{e}'_{-i_0}X_{-i_0}(X'X)^{-1}X'_{i_0}e_{i_0}}{1-h_{i_0i_0}}$.

Recall that we have $\vec{e}' X = \vec{0}$

$$\Rightarrow \vec{e}'_{-i_0} X_{-i_0} + e_{i_0} X_{i_0} = \vec{0}$$

~~So~~ $\cancel{X_{i_0} (X'X)^{-1} X'_{-i_0} X_{-i_0} X_{i_0}}$

Also $X'_{-i_0} X_{-i_0} = X'X - X'_{i_0} X_{i_0}$.

So $$\frac{X_{i_0} (X'X)^{-1} X'_{-i_0} X_{-i_0} (X'X)^{-1} X'_{i_0}}{(1-h_{i_0 i_0})^2} - 1$$

$$= \frac{X_{i_0} (X'X)^{-1} (X'X - X'_{i_0} X_{i_0}) (X'X)^{-1} X'_{i_0}}{(1-h_{i_0 i_0})^2} - 1$$

$$= \frac{h_{i_0 i_0} - h^2_{i_0 i_0}}{(1-h_{i_0 i_0})^2} - 1 = \frac{h_{i_0 i_0}}{1-h_{i_0 i_0}} - 1 = \frac{2h_{i_0 i_0} - 1}{1-h_{i_0 i_0}}.$$

and $$\frac{2\vec{e}'_{-i_0} X_{-i_0} (X'X)^{-1} X'_{i_0} e_{i_0}}{1-h_{i_0 i_0}}.$$

$$= \frac{2 \cdot [- e_{i_0} X_{i_0} (X'X)^{-1} X'_{i_0} e_{i_0}]}{1-h_{i_0 i_0}} = \frac{-2h_{i_0 i_0} e^2_{i_0}}{1-h_{i_0 i_0}}$$

$$\Rightarrow \vec{Y}'_{-i_0} (I - H_{X_{-i_0}}) \vec{Y}_{-i_0} = \vec{Y}'(I-H_X)\vec{Y} - \frac{e^2_{i_0}}{1-h_{i_0 i_0}}$$

$$\Rightarrow \frac{t_i}{r_i} = \sqrt{\frac{n-p-1}{\left[n-p-\frac{e^2_{i_0}}{(1-h_{i_0 i_0}) \cdot \frac{\vec{Y}'(I-H_X)\vec{Y}}{n-p}}\right]}} = \sqrt{\frac{n-p-1}{n-p-r^2_{i_0}}}$$

3 Inference under incorrectly specified models

1:
$$\hat{\beta}_1 = \left[(x_1, x_2 \cdots x_n)\begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}\right]^{-1} (x_1 \cdots x_n)\begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix}$$

$$= \frac{\sum x_i y_i}{\sum_{i=1}^{n} x_i^2}$$

$$\Rightarrow E[\hat{\beta}_1] = \frac{\sum x_i(\beta_0 + \beta_1 x_i)}{\sum x_i^2} = \beta_1 + \beta_0 \frac{\sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2}$$

$$\Rightarrow \text{bias}[\hat{\beta}_1] = \beta_0 \frac{\sum_{i=1}^{n} x_i}{\sum_{i=1}^{n} x_i^2}$$

4: Multiple Comparisons:

Denote $FR = \#\{\text{false positive}\}$ $R = \#\{\text{null hypothesis we reject}\}$

Then $FWER = P(FR \geq 1)$ $FDR = \cancel{E[FR/R]}\ E[FR/R]$ (define: $0/0 = 0$)

1: When all null hypothesis are true, then every hypothesis we reject is false positive. so $FR/R = \cancel{1_{FR}}\ 1_{R\geq 1} = 1_{FR \geq 1}$.

$$\text{so } FDR = E[FR/R] = E[1_{\{FR \geq 1\}}] = P(FR \geq 1) = FWER$$

2: Since when $FR = 0$ $FR/R = 0$

and FR/R always ≤ 1 ($FR \leq R$)

So $FR/R \leq 1_{\{FR \geq 1\}} \Rightarrow FDR = E[FR/R] \leq E[1_{\{FR \geq 1\}}] = FWER$

So when every FWER is controlled FDR is also controlled.

Coding and data analysis exercises

1.

```
require(stats)

mylm <- function(Y,X){

    Y = as.matrix(Y); Xnew = as.matrix(X)

    # Check if numeric
    if(!is.numeric(Y) | !is.numeric(Xnew))
        stop("Y or X is not numeric!\n")

    # Check for dimensions
    dy = dim(Y) ; dx = dim(Xnew)
    if(dy[2] != 1 | dy[1] != dx[1])
        stop("Y or X has wrong dimensions\n")

    # Check for ill conditioned elements
    # we can use is.finite to response only to finite real numbers
    if(FALSE %in% is.finite(Y) | FALSE %in% is.finite(Xnew))
        warning("Y or X is ill conditioned\n")

    # Check if of full rank
    D = cbind(1,Xnew)
    DtD = t(D) %*% D
    if(det(DtD) == 0)
        stop("Design matrix is not full rank\n")

    # Regressing
    DtD.inv = solve(DtD)
    hat.matrix = D %*% DtD.inv %*% t(D)
    beta = DtD.inv %*% t(D) %*% Y
    fitted = D %*% beta
    residuals = Y - fitted

    SS.tot = sum((Y - mean(Y))^2)
    if(SS.tot == 0)
        warning("Y is constant!\n")
    SS.res = sum((Y - fitted)^2)
    SS.reg = SS.tot - SS.res
    R2 = SS.reg / SS.tot

    df = dim(D)[1] - dim(D)[2]
    df2 = dim(D)[2] - 1

    s2 = SS.res / df
    std_error = sqrt(s2 * diag(DtD.inv))
    t_value = beta / std_error
    P_value = 1 - pt(abs(t_value), df) + pt(-abs(t_value), df)
```

```r
    K = cbind(rep(0, df2), diag(df2))
    Kbeta = K %*% beta
    Fstat = t(Kbeta) %*% solve(K %*% DtD.inv %*% t(K)) %*% Kbeta
    Fstat = Fstat / (df2 * s2)
    P_value_F = 1 - pf(Fstat, df2, df)

    beta_names = c("(Interception)")
    for(i in 1:(dim(D)[2] - 1)){
        beta_names = c(beta_names, sprintf("beta%d",i))
    }
    t_summary = data.frame("Estimate"=beta, "Std.Error"=std_error,
                           "t.value"=t_value, "Pvalue"=P_value)
    row.names(t_summary) = beta_names

    summary <- function(){
        cat("T table:\n")
        print(t_summary)
        cat("\nOverall F test:\n")
        cat(sprintf("F-statistics: %f on %d and %d DF,  p-value: %f",
                    Fstat, df2, df, P_value_F))
    }

    internal.t.res = c()
    external.t.res = c()
    PRESS.res = c()
    Cooks.dist = c()
    for(i in 1:dx[1]){
        ir = residuals[i] / sqrt(s2 * (1 - hat.matrix[i,i]))
        er = ir * sqrt((df - 1) / (df - ir^2))
        pr = residuals[i] / (1 - hat.matrix[i,i])
        cd = ir^2 / dim(D)[2] * (hat.matrix[i,i] / (1 - hat.matrix[i,i]))
        internal.t.res = c(internal.t.res, ir)
        external.t.res = c(external.t.res, er)
        PRESS.res = c(PRESS.res, pr)
        Cooks.dist = c(Cooks.dist, cd)
    }

    # Return result
    result = list(beta = beta,
                  fitted = fitted,
                  residuals = residuals,
                  R2 = R2,
                  hatdiag = diag(hat.matrix),
                  summary = summary,
                  internal.t.res = internal.t.res,
                  external.t.res = external.t.res,
                  PRESS.res = PRESS.res,
                  Cooks.distance = Cooks.dist)
    return(result)
}

test.X = cbind(sample(1:100),sample(1:100),sample(1:100))
beta = c(5,-1,0.01,2)
```

```r
test.y = cbind(1,test.X) %*% beta + rnorm(100,0,5)
test.y[50] = test.y[50] + 10
model = lm(test.y ~ test.X)
mymodel = mylm(test.y, test.X)

ir = rstandard(model)
er = rstudent(model)
cook = cooks.distance(model)
pr = model$residuals / (1 - hatvalues(model))

#Test internal standardized residuals:
print(sum(abs(ir - mymodel$internal.t.res)))
```

```
## [1] 9.470064e-13
```

```r
#Test external standardized residuals:
print(sum(abs(er - mymodel$iexternal.t.res)))
```

```
## [1] 0
```

```r
#Test PRESS residuals:
print(sum(abs(pr - mymodel$PRESS.res)))
```

```
## [1] 5.78998e-12
```

```r
#Test Cook's distance:
print(sum(abs(cook - mymodel$Cooks.distance)))
```

```
## [1] 1.669646e-14
```