Statistical Modelling
Coursework 2

Stefan A. Obada - *so1118*

March 2020

Abstract

This paper solves the second Coursework available on Blackboard.

Question 1

a)

As we are working under (FR) and (SOA) we can solve for $\hat{\gamma}_{LSE}$ as: $\hat{\gamma}_{LSE} = (X^T X)^{-1} X^T \mathbf{Y}$, where X is the design matrix. In our case:

$$X = \begin{bmatrix} 1 & w_1 & x_1 \\ 1 & w_2 & x_2 \\ \vdots & \vdots & \vdots \\ 1 & w_n & x_n \end{bmatrix}$$

$$X^T X = \begin{bmatrix} n & n\overline{w} & n\overline{x} \\ n\overline{w} & \sum w_i^2 & \sum w_i x_i \\ n\overline{x} & \sum w_i x_i & \sum x_i^2 \end{bmatrix}$$

Using the following 2 properties for the 3x3 matrix inverse and for S_{ab} we will deduce the form of $(X^T X)^{-1}$ and $X^T \mathbf{Y}$. We will skip the matrix calculation part.

$$\begin{aligned} \begin{pmatrix} a & b & c \\ d & e & f \\ g & h & i \end{pmatrix}^{-1} &= \frac{1}{\det()} \begin{pmatrix} ei - fh & -(bi - ch) & bf - ce \\ -(di - fg) & ai - cg & -(af - cd) \\ dh - eg & -(ah - bg) & ae - bd \end{pmatrix} \\ &= \frac{1}{\det()} \begin{pmatrix} a' & b' & c' \\ d' & e' & f' \\ g' & h' & i' \end{pmatrix} \end{aligned} \tag{1}$$

$$\begin{aligned} & S_{ab} = \sum_i (a_i - \overline{a})(b_i - \overline{b}) = \sum (a_i b_i - a_i \overline{b} - \overline{a} b_i + \overline{a}\overline{b}) = \\ & \sum a_i b_i - 2n\overline{a}\overline{b} + n\overline{a}\overline{b} = \sum a_i b_i - n\overline{a}\overline{b}. \\ & Moreover : S_{aa} = \sum a_i^2 - n\overline{a}^2 \end{aligned} \tag{2}$$

Therefore, we deduce:

$$(X^T X)^{-1} = \frac{1}{n(S_{xx} S_{ww} - S_{xw}^2)} \begin{bmatrix} a' & b' & c' \\ d' & e' & f' \\ g' & h' & i' \end{bmatrix}$$

$$X^T \mathbf{Y} = \begin{bmatrix} \overline{Y} \\ S_{wy} + n\overline{w}\overline{Y} \\ S_{xy} + n\overline{x}\overline{Y} \end{bmatrix}$$

$$(X^T X)^{-1} X^T \mathbf{Y} = \frac{1}{n(S_{xx} S_{ww} - S_{xw}^2)} \begin{bmatrix} a'\overline{Y} + b'(S_{wy} + n\overline{w}\overline{Y}) + c'(S_{xy} + n\overline{x}\overline{Y}) \\ d'\overline{Y} + e'(S_{wy} + n\overline{w}\overline{Y}) + f'(S_{xy} + n\overline{x}\overline{Y}) \\ g'\overline{Y} + h'(S_{wy} + n\overline{w}\overline{Y}) + i'(S_{xy} + n\overline{x}\overline{Y}) \end{bmatrix}$$

Now plugging the values for $(a', b' ... i')$ (see Appendix) we find:

$$\begin{aligned}\hat{\gamma}_{LSE} =&(X^TX)^{-1}X^T\mathbf{Y} = \\ =&\frac{1}{(S_{xx}S_{ww}-S_{xw}^2)}\begin{bmatrix}(S_{ww}S_{wx}-S_{xw}^2)\overline{Y}-\overline{w}(S_{wy}S_{xx}-S_{wx}S_{xy})-\overline{x}(S_{xy}S_{ww}-S_{wx}S_{xy})\\ S_{wy}S_{xx}-S_{wx}S_{xy}\\ S_{xy}S_{ww}-S_{wx}S_{xy}\end{bmatrix}\end{aligned}$$

Therefore, we obtain the final estimators as:

$$\begin{aligned}\hat{\gamma}_0 =&\overline{Y}-\overline{w}\hat{\gamma}_W-\overline{x}\hat{\gamma}_X\\ \hat{\gamma}_W =&\frac{S_{wy}S_{xx}-S_{wx}S_{xy}}{S_{xx}S_{ww}-S_{xw}^2}\\ \hat{\gamma}_X =&\frac{S_{xy}S_{ww}-S_{wx}S_{xy}}{S_{xx}S_{ww}-S_{xw}^2}\end{aligned}$$

b)

As proved in lectures, we have the following equation for $\hat{\beta}_X$:

$$\hat{\beta}_X = \frac{S_{xy}}{S_{xx}}$$

Therefore we can condition on $\mathbf{x}-\overline{\mathbf{x}}$ and $\mathbf{w}-\overline{\mathbf{w}}$ as it follows:

$$\begin{aligned}\hat{\beta}_X = \hat{\gamma}_X \Leftrightarrow&\frac{S_{xy}S_{ww}-S_{wx}S_{xy}}{S_{xx}S_{ww}-S_{xw}^2} = \frac{S_{xy}}{S_{xx}} \Leftrightarrow\\ \Leftrightarrow&S_{xx}S_{xy}S_{ww}-S_{xx}S_{wx}S_{xy} = S_{xy}S_{xx}S_{ww}-S_{xw}^2S_{xy} \Leftrightarrow\\ \Leftrightarrow&S_{xx} = S_{wx} \Leftrightarrow (\mathbf{x}-\overline{\mathbf{x}}-(\mathbf{w}-\overline{\mathbf{w}}))^T(\mathbf{x}-\overline{\mathbf{x}}) = \mathbf{0} \Leftrightarrow\\ \Leftrightarrow&(\mathbf{w}-\overline{\mathbf{w}})^T(\mathbf{x}-\overline{\mathbf{x}}) = |(\mathbf{x}-\overline{\mathbf{x}})|^2\end{aligned}$$

So if given $\mathbf{x}$, then $\mathbf{w}-\overline{\mathbf{w}}$ has to lie on a hyperplane whose normal is $\mathbf{n} = \mathbf{x}-\overline{\mathbf{x}}$ and passes through the point $\mathbf{x}-\overline{\mathbf{x}}$. Note that we worked under the assumption that $S_{xy} \neq 0$, as otherwise the model would be trivial.

c)

Again, as proved in lectures we have:

$$\begin{aligned}cov(\hat{\beta}_X) &= \frac{\sigma^2}{S_{xx}}\\ cov(\hat{\gamma}_{LSE}) &= \sigma^2(X^TX)^{-1} \Rightarrow cov(\hat{\gamma}_X) = \frac{\sigma^2 i'}{n(S_{xx}S_{ww}-S_{xw}^2)} =\\ &= \frac{n\sigma^2 S_{ww}}{n(S_{xx}S_{ww}-S_{xw}^2)} = \frac{\sigma^2}{S_{xx}-S_{xw}^2/S_{ww}}\\ &= \frac{\sigma^2}{S_{xx}}\cdot(1-\frac{S_{xw}^2}{S_{xx}S_{ww}})^{-1}\end{aligned}$$

- Under the condition from **1b)** we have that $S_{xx} = S_{wx}$ and therefore they will differ by a factor of: $(1-\frac{S_{xx}}{S_{ww}})^{-1}$.

- $\underline{var(\hat{\gamma}_X) < var(\hat{\beta}_X)} \Leftrightarrow (1 - \frac{S_{xw}^2}{S_{xx}S_{ww}})^{-1} < 1 \Leftrightarrow 1 - \frac{S_{xw}^2}{S_{xx}S_{ww}} < 0 \Leftrightarrow$
  $\Leftrightarrow \underline{S_{xw}^2 > S_{xx}S_{ww}}$ (since otherwise $\frac{S_{xw}^2}{S_{xx}S_{ww}} < 0$ but $S_{aa} > 0$). However this cannot ever happen since by letting $\mathbf{u} = \mathbf{x} - \overline{\mathbf{x}}$ and $\mathbf{v} = \mathbf{w} - \overline{\mathbf{w}}$, the last relation is therefore equivalent to: $\langle \mathbf{u}, \mathbf{v} \rangle \langle \mathbf{u}, \mathbf{v} \rangle > \langle \mathbf{u}, \mathbf{u} \rangle \langle \mathbf{v}, \mathbf{v} \rangle$ which contradicts Cauchy–Bunyakovsky–Schwarz inequality.

- $\underline{var(\hat{\gamma}_X) > var(\hat{\beta}_X)} \Leftrightarrow \underline{S_{xw}^2 < S_{xx}S_{ww}}$ by a similar argument. This explains that by adding a new parameter in a linear model results in an increased variance of the initial parameters.

Question 2

a)

To compute the least squares estimators we will use the LinearRegression() model of *sklearn* in python, in order to avoid too long code. It simply calculates the ordinary least squares estimators as $(X^TX)^{-1}X^T\mathbf{Y}$, also handling automatically adding the columns of ones to the design matrix. $\hat{\beta}_0$ and $\hat{\beta}_1$ will therefore be the *intercept* and the *coef* of the model.

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression

# Import data
df = pd.read_csv(’fev.csv’)
X = df[[’smoke’, ’age’]]
y = df[’fev’].values.reshape(-1,1)

# 2 (a) #
X_fev = df.smoke.values.reshape(-1,1) # Design matrix with only
                                        smoke as covariate

## Build the model
model = LinearRegression() # We use sklearn to avoid long matrix
                                        calculation

## Fit the model
model.fit(X_fev, y)

beta0 = model.intercept_[0]
beta1 = model.coef_[0][0]

print(f’beta0 = {beta0}’)
print(f’beta1 = {beta1}’)
```

Output:

```
beta0 = 3.9875804623220583
beta1 = -0.71071892386052
```

Moreover, we can compute $E(fev|smoke = 1) - E(fev|smoke = 2)$ as:

```
## Mean difference
exp_diff = (beta0 + 1*beta1) - (beta0 + 2*beta1)
print(’Fev(smokers) - Fev(nonsmokers): {}’.format(exp_diff))
##
```

Output:

```
Fev(smokers) - Fev(nonsmokers): 0.71071892386052
```

Conclusion: According to the estimates, smoking does not seem to impair lung function in children and it even contradicts the initial hypothesis that FEV is should be smaller for smokers. However, we will see that this result is not credible and we have to take into account another covariate: *age*. Explore in **??** more about this.

b)

Proceeding similarly as before, we will use *LinearRegression()*.

```
# 2 (b)
# Recall that X = df[['smoke', 'age']]

## Fitting the model
model_b = LinearRegression()
model_b.fit(X, y)

gamma0, gamma1, gamma2 = model_b.intercept_[0], model_b.coef_[0][
                         0], model_b.coef_[0][1]
print(f'gamma0 = {gamma0}')
print(f'gamma1 = {gamma1}')
print(f'gamma2 = {gamma2}')
##
```

Output:

```
gamma0 = -0.05061671981585736
gamma1 = 0.20899487720480997
gamma2 = 0.2306045731168579
```

And the mean difference:

```
## Mean difference
exp_diff = gamma1*1-gamma1*2
print('Fev(smokers) - Fev(nonsmokers): {}'.format(exp_diff))
##
```

Output:

```
Fev(smokers) - Fev(nonsmokers): -0.20899487720480997
```

This results are indeed more realistic and shows that among children, smokers tend to have a *0.2* decrease in FEV compared to non-smokers.

Moreover, according to **1c)** we should have $\frac{var(\hat{\beta}_1)}{var(\hat{\gamma}_1)} < 1$. We compute it with the previous formulae: $\frac{var(\hat{\beta}_1)}{var(\hat{\gamma}_1)} = 1 - \frac{S_{xw}^2}{S_{xx}S_{ww}}$.

```
## Variability
def S(a: np.array, b: np.array):
    '''Return S_{ab} as in the coursework'''
    return (a-a.mean()).dot(b-b.mean())

print('var(beta1)/var(gamma1) = {}'.format((1-S(X.smoke,X.age)**2
                         /(S(X.smoke,X.smoke)*S(X.age, X.
                         age)))))
```

Output:

```
var(beta1)/var(gamma1) = 0.8365799359665773
# Which is smaller than 1.
```

c)

d)

Appendix: values of $a', b'...i'$

$$
\begin{aligned}
a' &= n \\
b' &= n\overline{w} \\
c' &= n\overline{x} \\
d' &= n\overline{w} \\
e' &= \sum w_i^2 = S_{ww} + n\overline{w}^2 \\
f' &= \sum w_i x_i = S_{wx} + n\overline{w}\overline{x} \\
g' &= n\overline{x} \\
h' &= \sum w_i x_i = S_{wx} + n\overline{w}\overline{x} \\
i' &= \sum x_i^2 = S_{xx} + n\overline{x}^2
\end{aligned}
$$