Name: CHAN King Yeung
SID: 1155119394
STAT4001 Homework 2

Question 1

a) $E(\hat{\beta}_1^{LS}) = E\left[\frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sum(x_i-\bar{x})^2}\right]$

$= \frac{\sum(x_i-\bar{x})(\beta_0+\beta_1 x_i-\beta_0+\beta_1\bar{x})}{\sum(x_i-\bar{x})^2}$ given $E(\bar{y}) = E\left(\frac{\sum y_i}{n}\right) = \frac{\sum(\beta_0+\beta_1 x_i)}{n} = \beta_0 + \beta_1\bar{x}$

$= \beta_1$

$E(\hat{\beta}_0^{LS}) = E(\bar{y} - \hat{\beta}_1^{LS}\bar{x})$

$= \beta_0 + \beta_1\bar{x} - \beta_1\bar{x}$

$= \beta_0$

$\therefore$ both $\hat{\beta}_0^{LS}$ and $\hat{\beta}_1^{LS}$ are unbiased estimators

b) $E(\hat{\beta}_1^{Ridge}) = E\left[\frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sum(x_i-\bar{x})^2+\lambda}\right]$

$= \frac{\sum(x_i-\bar{x})(\beta_0+\beta_1 x_i-\beta_0+\beta_1\bar{x})}{\sum(x_i-\bar{x})^2+\lambda}$

$= \beta_1\frac{\sum(x_i-\bar{x})^2}{\sum(x_i-\bar{x})^2+\lambda}$

$\neq \beta_1$

$E(\hat{\beta}_0^{Ridge}) = E(\bar{y} - \hat{\beta}_1^{Ridge}\bar{x})$

$= \beta_0 + \beta_1\bar{x} - \beta_1\bar{x}\frac{\sum(x_i-\bar{x})^2}{\sum(x_i-\bar{x})^2+\lambda}$

$= \beta_0 + \beta_1\bar{x}\left[1 - \frac{\sum(x_i-\bar{x})^2}{\sum(x_i-\bar{x})^2+\lambda}\right]$

$\neq \beta_0$

$\therefore$ both $\hat{\beta}_0^{Ridge}$ and $\hat{\beta}_1^{Ridge}$ are biased estimators

---

Question 2

a)

i. $\hat{\beta}_1^{LS} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}$ $\qquad$ $\hat{\beta}_0^{LS} = \bar{y} - \hat{\beta}_1^{LS}\bar{x}$

$= \frac{20.2 - 3(2)(3.1)}{14 - 3(2)^2}$ $\qquad$ $= 3.1 - 0.8(2)$

$= 0.8$ $\qquad$ $= 1.5$

$\hat{\beta}_1^{Ridge} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2 + \lambda}$ $\qquad$ $\hat{\beta}_0^{Ridge} = \bar{y} - \hat{\beta}_1^{Ridge}\bar{x}$

$= \frac{20.2 - 3(2)(3.1)}{14 - 3(2)^2 + 1}$ $\qquad$ $= 3.1 - 0.5333(2)$

$= 0.5333$ $\qquad$ $= 2.0333$

ii. $\hat{\boldsymbol{y}}^{LS} = (2.3 \quad 3.1 \quad 3.9)'$

$\hat{\boldsymbol{y}}^{Ridge} = (2.5666 \quad 3.0999 \quad 3.6332)'$

b)

i. $\hat{\beta}_1^{L} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2}$ $\qquad$ $\hat{\beta}_0^{L} = \bar{y} - \hat{\beta}_1^{L}\bar{x}$

$= \frac{202 - 3(20)(3.1)}{1400 - 3(20)^2}$ $\qquad$ $= 3.1 - 0.08(20)$

$= 0.08$ $\qquad$ $= 1.5$

$\hat{\beta}_1^{R} = \frac{\sum x_i y_i - n\bar{x}\bar{y}}{\sum x_i^2 - n\bar{x}^2 + \lambda}$ $\qquad$ $\hat{\beta}_0^{R} = \bar{y} - \hat{\beta}_1^{R}\bar{x}$

$= \frac{202 - 3(20)(3.1)}{1400 - 3(20)^2 + 1}$ $\qquad$ $= 3.1 - 0.0796(20)$

$= 0.0796$ $\qquad$ $= 1.508$

ii. Comparing with $\left(\hat{\beta}_0^{Ridge}, \frac{\hat{\beta}_1^{Ridge}}{10}\right)$ and $\left(\hat{\beta}_0^{R}, \hat{\beta}_1^{R}\right)$, the coefficient estimates change substantially given there is a scaling constant on $\boldsymbol{X}$ and penalty terms is independent of the column space of $\boldsymbol{X}$.

Comparing with $\left(\hat{\beta}_0^{LS}, \frac{\hat{\beta}_1^{LS}}{10}\right)$ and $\left(\hat{\beta}_0^{L}, \hat{\beta}_1^{L}\right)$, the coefficient estimates are scale invariant which is expected since linear transformation does not alter the column space of $\boldsymbol{X}$. In other words, $\boldsymbol{X}\hat{\boldsymbol{\beta}}$ remains unchanged.

iii. $\hat{\boldsymbol{y}}^{L} = (2.3 \quad 3.1 \quad 3.9)'$

$\hat{\boldsymbol{y}}^{R} = (2.304 \quad 3.1 \quad 3.896)'$

Comparing $\hat{\boldsymbol{y}}^{Ridge}$ and $\hat{\boldsymbol{y}}^{R}$, the predicated values also changed due to the variant property of ridge regression. Standardising $\boldsymbol{X}$ is a recommended way to reduce the effect while scaling $\boldsymbol{X}$.

Comparing $\hat{\boldsymbol{y}}^{LS}$ and $\hat{\boldsymbol{y}}^{L}$, the predicated values remain the same given the fact that the column space of $\boldsymbol{X}$ does not change over the linear transformation on it.

Question 3

Given $b < -\frac{\lambda}{2} < 0$, we have $\hat{\beta}_j = \frac{2b+\lambda}{2a} < 0$, and we can prove that $\hat{\beta}_j$ minimises $f = a\beta_j^2 - 2b\beta_j + \lambda\left|\beta_j\right|$ at $\frac{2b+\lambda}{2a}$.

1. When $\hat{\beta}_j < 0$, we have $f = a\beta_j^2 - 2b\beta_j - \lambda\beta_j$. To minimise f, we formulate

   $$\frac{\partial f}{\partial \beta}\Big|_{\hat{\beta}} = 0$$
   $$0 = 2a\hat{\beta}_j - 2b - \lambda$$
   $$\hat{\beta} = \frac{2b+\lambda}{2a}$$

   $\therefore \hat{\beta} = \frac{2b+\lambda}{2a} < 0$ minimises f.

2. When $\hat{\beta}_j > 0$, we plug-in $\hat{\beta}$ and $-\hat{\beta}$ into f

   $$f\left(\beta_j = \hat{\beta}_j\right) \coloneqq a\beta_j^2 - 2b\beta_j - \lambda\beta_j$$
   $$f\left(\beta_j = -\hat{\beta}_j\right) \coloneqq a\beta_j^2 + 2b\beta_j + \lambda\beta_j$$

   Since $f\left(\beta_j = \hat{\beta}_j\right) < f\left(\beta_j = -\hat{\beta}_j\right)$ which contradicts the minimisation, $\hat{\beta}_j$ cannot minimise $f(\cdot)$ if $\hat{\beta}_j > 0$.

3. When $\hat{\beta}_j = 0$, we plug-in 0 into f

   $$f\left(\beta_j = 0\right) = a(0)^2 - 2b(0) + \lambda|0|$$
   $$= 0$$

   In addition, $f = \frac{\beta_j}{a}\left(\beta_j - \frac{2b+\lambda}{a}\right)$. We plug-in $\frac{2b+\lambda}{2a}$ into f

   $$f\left(\beta_j = \frac{2b+\lambda}{2a}\right) = \frac{\hat{\beta}_j}{a}\left(\frac{2b+\lambda}{2a} - \frac{2b+\lambda}{a}\right)$$
   $$= \frac{\hat{\beta}_j}{a}\left(-\frac{2b+\lambda}{2a}\right)$$
   $$> 0$$

   $\therefore \hat{\beta} = \frac{2b+\lambda}{2a} < 0$ minimises f.

Question 4

a) $bias^2 = [y_0 - E(\hat{y}_0)]^2$

$= \left[\beta_0 + \beta_1 x_0 - E\left(\hat{\beta}_0 + \hat{\beta}_1 x_0 + \varepsilon_0\right)\right]^2$

$= [\beta_0 + \beta_1 x_0 - \beta_0 - \beta_1 x_0 - 0]^2$ given $E\left(\hat{\beta}_0\right) = \beta_0, E\left(\hat{\beta}_1\right) = \beta_1$ (see Question 1a) and $E(\varepsilon_0) = 0$

$= 0$

$variance = Var(\hat{y}_0)$

$= Var\left(\hat{\beta}_0 + \hat{\beta}_1 x_0 + \varepsilon_0\right)$

$= Var\left(\hat{\beta}_0\right) + x_0^2 Var\left(\hat{\beta}_1\right) + 2x_0 Cov\left(\hat{\beta}_0, \hat{\beta}_1\right) + Var(\varepsilon_0)$ given $\varepsilon_0 \perp\!\!\!\perp \hat{\beta}_0 + \hat{\beta}_1 x_0$

$= \sigma^2 \left(\frac{1}{n} + \frac{\bar{x}^2}{SXX}\right) + x_0^2 \sigma^2 \left(\frac{1}{SXX}\right) - 2x_0 \sigma^2 \left(\frac{\bar{x}}{SXX}\right) + \sigma^2$

given $Cov\left(\hat{\beta}_0, \hat{\beta}_1\right) = E\left\{\frac{\sum(x_i - \bar{x})y_i}{SXX}\left[\bar{y} - \frac{\sum(x_i - \bar{x})y_i}{SXX}\bar{x}\right]\right\} = -\frac{\sigma^2 \bar{x}}{SXX}$ and $SXX = \sum(x_i - \bar{x})^2$

$= \sigma^2 \left[1 + \frac{1}{n} + \frac{(x_0 - \bar{x})^2}{SXX}\right]$

b) $bias^2 = [y_0 - E(\hat{y}_0)]^2$

$= \left[\beta_0 + \beta_1 x_0 - E\left(\hat{\beta}_0 + \hat{\beta}_1 x_0 + \varepsilon_0\right)\right]^2$

$= \left[\beta_0 + \beta_1 x_0 - \beta_0 - \beta_1 \bar{x}\left(1 - \frac{SXX}{SXX+\lambda}\right) - \beta_1 x_0 \frac{SXX}{SXX+\lambda} - 0\right]^2$

given $E\left(\hat{\beta}_0\right) = \beta_0 + \beta_1 \bar{x}\left(1 - \frac{SXX}{SXX+\lambda}\right)$ and $E\left(\hat{\beta}_0\right) = \beta_1 \frac{SXX}{SXX+\lambda}$ (see Question 1b)

$= \left[\frac{\beta_1 \lambda (x_0 - \bar{x})}{SXX+\lambda}\right]^2$

$variance = Var(\hat{y}_0)$

$= Var\left(\hat{\beta}_0 + \hat{\beta}_1 x_0 + \varepsilon_0\right)$

$= Var\left(\hat{\beta}_0\right) + x_0^2 Var\left(\hat{\beta}_1\right) + 2x_0 Cov\left(\hat{\beta}_0, \hat{\beta}_1\right) + Var(\varepsilon_0)$

$= \sigma^2 \left[\frac{1}{n} + \frac{\bar{x}^2 (SXX)}{(SXX+\lambda)^2}\right] + x_0^2 \sigma^2 \left[\frac{SXX}{(SXX+\lambda)^2}\right] - 2x_0 \sigma^2 \left[\frac{\bar{x}(SXX)}{(SXX+\lambda)^2}\right] + \sigma^2$

given $Var\left(\hat{\beta}_1\right) = Var\left[\frac{\sum(x_i - \bar{x})y_i}{SXX+\lambda}\right] = \sigma^2 \left[\frac{SXX}{(SXX+\lambda)^2}\right]$, $Var\left(\hat{\beta}_0\right) = Var\left(\bar{y} - \bar{x}\hat{\beta}_1\right) = \sigma^2 \left[\frac{1}{n} + \frac{\bar{x}^2 (SXX)}{(SXX+\lambda)^2}\right]$,

$Cov\left(\hat{\beta}_0, \hat{\beta}_1\right) = E\left\{\frac{\sum(x_i - \bar{x})y_i}{SXX}\left[\bar{y} - \frac{\sum(x_i - \bar{x})y_i}{SXX}\bar{x}\right]\right\} = -\frac{\sigma^2 \bar{x}(SXX)}{(SXX+\lambda)^2}$

$= \sigma^2 \left[1 + \frac{1}{n} + \frac{SXX(x_0 - \bar{x})^2}{(SXX+\lambda)^2}\right]$

Question 5

a) Please refer to the following console of output (see Appendix for your reference)

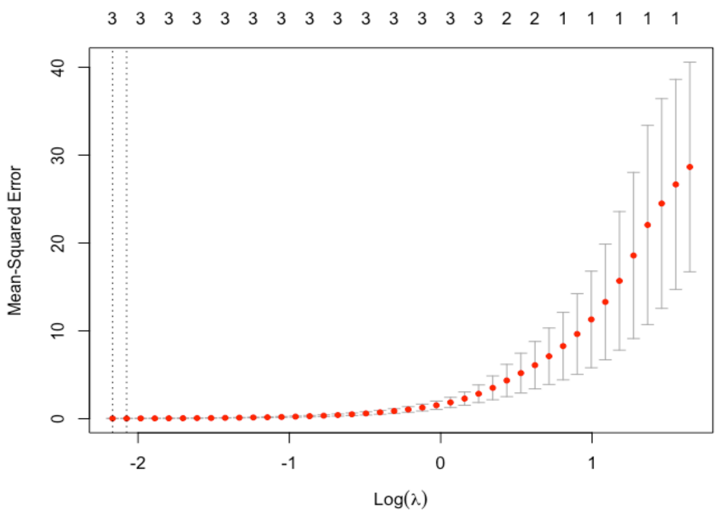
```
> set.seed(4001)
> x = rnorm(100, 0, 1)
> e = rnorm(100, 0, 0.1)
```

b) Please refer to the following console of output (see Appendix for your reference)

```
> y = 1 + x + x ^ 2 + x ^ 3 + e
```

c) Please refer to the following console of output (see Appendix for your reference)

```
> library(glmnet)
>
> X = data.frame(x)
> names(X) = "X"
> for(i in 2:10) {
+   X = cbind(X, x ^ i)
+   names(X)[i] = paste0("X ^ ", i)
+ }
> X = as.matrix(X)
>
> cv = cv.glmnet(X, y, alpha = 1)
> plot(cv)
```

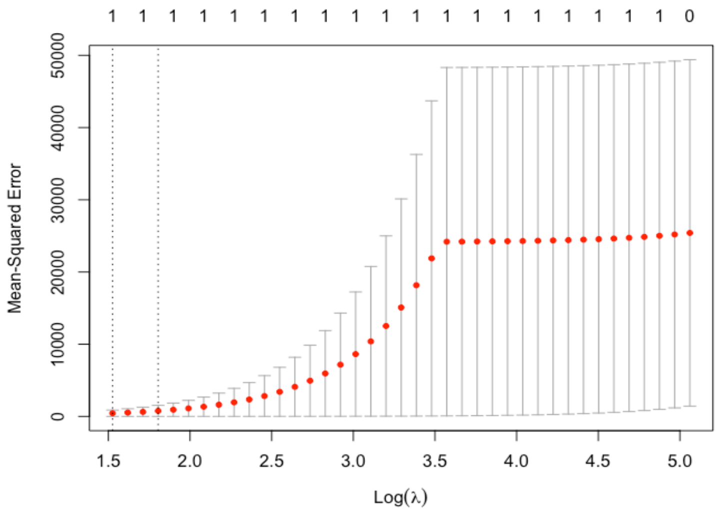



```
> cv$lambda.min
[1] 0.1143797
>
> lasso = glmnet(X, y, alpha = 1)
> predict(lasso, type = "coefficient", s = cv$lambda.min)
11 x 1 sparse Matrix of class "dgCMatrix"
                    1
(Intercept) 1.0570168
X           0.9242384
X ^ 2       0.9292638
X ^ 3       1.0034351
X ^ 4       .
X ^ 5       .
X ^ 6       .
X ^ 7       .
X ^ 8       .
X ^ 9       .
X ^ 10      .
```

The optimal value of λ is 0.1144.
The resulting coefficient estimates are $\boldsymbol{\beta} = (1.057 \quad 0.9242 \quad 0.9293 \quad 1.0034 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0)'$

d) Please refer to the following console of output (see Appendix for your reference)

```
> y = 1 + x ^ 7 + e
>
> cv = cv.glmnet(X, y, alpha = 1)
> plot(cv)
```




```
> cv$lambda.min
[1] 4.597186
>
> lasso = glmnet(X, y, alpha = 1)
> predict(lasso, type = "coefficient", s = cv$lambda.min)
11 x 1 sparse Matrix of class "dgCMatrix"
                    1
(Intercept) 1.620063
X           .
X ^ 2       .
X ^ 3       .
X ^ 4       .
X ^ 5       .
X ^ 6       .
X ^ 7       0.970864
X ^ 8       .
X ^ 9       .
X ^ 10      .
```

The optimal value of λ is 4.5972.
The resulting coefficient estimates are $\boldsymbol{\beta} = (1.6201 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0.9709 \quad 0 \quad 0 \quad 0)'$

Appendix

```
## Question 5
# Part a
set.seed(4001)
x = rnorm(100, 0, 1)
e = rnorm(100, 0, 0.1)

# Part b
y = 1 + x + x ^ 2 + x ^ 3 + e

# Part c
library(glmnet)

X = data.frame(x)
names(X) = "X"
for(i in 2:10) {
  X = cbind(X, x ^ i)
  names(X)[i] = paste0("X ^ ", i)
}
X = as.matrix(X)

cv = cv.glmnet(X, y, alpha = 1)
plot(cv)
cv$lambda.min

lasso = glmnet(X, y, alpha = 1)
predict(lasso, type = "coefficient", s = cv$lambda.min)

# Part d
y = 1 + x ^ 7 + e

cv = cv.glmnet(X, y, alpha = 1)
plot(cv)
cv$lambda.min

lasso = glmnet(X, y, alpha = 1)
predict(lasso, type = "coefficient", s = cv$lambda.min)
```