MAST30025 Linear Statistical Models
Assignment 2

Student Code: 1079860

April 2023

Question 1

$E\,[y^* - (\vec{x}^*)^T\ \vec{b}\,]$

$= E[(\vec{x}^*)^T\ \vec{\beta}\,] + E[\vec{\epsilon}^*] - E[(\vec{x}^*)^T\ \vec{b}\,]$

$= (\vec{x}^*)^T\vec{\beta} + 0 - (\vec{x}^*)^T\vec{\beta}$

$= 0$

$Var[y^* - (\vec{x}^*)^T\ \vec{b}\,] = Var[\vec{\epsilon}^*] + Var[(\vec{x}^*)^T\ \vec{b}\,]$

$= \sigma^2 + (\vec{x}^*)^T(X^TX)^{-1}\sigma^2\vec{x}^*$

$= \sigma^2 + (\vec{x}^*)^T(X^TX)^{-1}\sigma^2\vec{x}^*$

$= [1 + (\vec{x}^*)^T(X^TX)^{-1}\vec{x}^*]\ \sigma^2$

$\implies \frac{y^*-(\vec{x}^*)^T\ \vec{b}}{\sqrt{[1+(\vec{x}^*)^T(X^TX)^{-1}\vec{x}^*]\ \sigma^2}} \sim Z\ \ as\ \ y^*-(\vec{x}^*)^T\ \ \vec{b}\ \ is\ \ normally\ \ distributed\ \ (linear\ \ combination\ \ of\ \ b_i s)$

$\frac{SS_{Res}}{\sigma^2} \sim \chi^2_{n-p}\ \ according\ \ to\ \ Theorem\ \ 4.13$

$\implies \sqrt{\frac{\frac{SS_{Res}}{\sigma^2}}{n-p}} = \frac{s}{\sigma} = \sqrt{\frac{\chi^2_{n-p}}{n-p}}$

$\implies \frac{y^*-(\vec{x}^*)^T\ \vec{b}}{\frac{s}{\sigma}\sqrt{[1+(\vec{x}^*)^T(X^TX)^{-1}\vec{x}^*]}\ \sigma} = \frac{y^*-(\vec{x}^*)^T\ \vec{b}}{s\sqrt{[1+(\vec{x}^*)^T(X^TX)^{-1}\vec{x}^*]}} \sim t_{n-p}\ \ by\ \ Definition\ \ 4.15$

Question 2

[R code]

```
y = c(37.9, 42.2, 47.3, 43.1, 54.8, 47.1, 40.3)
X = matrix(c(rep(1,7),32,19.5,13.3,13.3,5,7.1,34.5,84.9,306.6,
        562.0,562.0,390.6,2175.0,623.5,10,9,5,5,5,3,7), 7, 4)
```

(a)

[R code]

```
n = nrow(X)
p = ncol(X)
b = solve(t(X) %*% X, t(X) %*% y)

e = y-X %*% b
SSRes = sum(e^2)
s2 = SSRes/(n-p)
```

$$\vec{b} = \begin{bmatrix} 58.369 \\ -0.346 \\ -0.003 \\ -0.888 \end{bmatrix}$$

$s^2 = 13.069$

(b)

[R code]

```
covar = solve(t(X) %*% X)
vars = diag(covar)
alpha = 0.05
ta = qt(1-alpha/2, df = n-p)
for (i in c(1:4)) {
    print((b[i] + c(-1,1)*ta*sqrt(s2*vars[i])))
}
```

$b_0 : [\,34.102,\ 82.637\,]$
$b_1 : [\,-0.997,\ 0.305\,]$
$b_2 : [\,-0.013,\ 0.007\,]$
$b_3 : [\,-4.818,\ 3.043\,]$

(c)

[R code]

```
xstar = c(1,5,100,6)
alpha = 0.1
ta = qt(1-alpha/2, df = n-p)
print(t(xstar) %*% b + c(-1,1)*ta*sqrt(s2*(1+t(xstar) %*% covar %*%
                                                                        xstar)))
```

$y^* : [\,39.869,\ 62.175\,]$

(d)

$H_0 : \beta_1 = -1$

[R code]

```
C = t(c(0, 1, 0, 0))
dstar = c(-1)

Fstat = t(C %*% b - dstar) %*% solve(C %*% covar %*% t(C)) %*%
                                          (C %*% b - dstar) / s2

alpha = 0.05

pval = pf(Fstat, 1, n-p, lower.tail = FALSE)
```

$p-value = 0.0495 < 0.05 \implies \textit{Reject } H_0 \textit{ at the } 5\% \textit{ significance level}$

(e)

$H_0 : \beta_1 = \beta_2 = \beta_3 = 0$

[R code]

```
fullModel = lm(y~X)
null = lm(y~1)
anova(null, fullModel)

X2 = X[,1]
n = nrow(X)
p = ncol(X)
b2 = solve(t(X2)%*%X2, t(X2)%*%y)

SSTotal = sum(y^2)
SSReg = SSTotal - SSRes
SSRes2 = sum((y-X2%*%b2)^2)

Rg2 = SSTotal - SSRes2

Rg1g2 = SSReg - Rg2

r = 3
Fstat = (Rg1g2/r) / (SSRes/(n-p))

pval = pf(Fstat, r, n-p, lower.tail = FALSE)
```

$p{-}value = 0.1501 > 0.05 \implies \textit{Cannot reject } H_0 \textit{ at the } 5\% \textit{ significance level}$

Question 3

$-2log(Likelihood) + 2p \ \ where \ \ likelihood \ \ is \ \ the \ \ maximised \ \ likelihood$

$= -\frac{-2n}{2}log(2\pi\sigma^2) - \frac{1}{2\sigma^2}(\vec{y} - X\vec{\beta})^T(\vec{y} - X\vec{\beta}) + 2p$

$= n(log(\sigma^2) + log(2\pi)) - \frac{1}{2\sigma^2}(\vec{y} - X\vec{\beta})^T(\vec{y} - X\vec{\beta}) + 2p$

$= n(log(\frac{SS_{Res}}{n}) + log(2\pi)) - \frac{1}{2\frac{SS_{Res}}{n}}SS_{Res} + 2p$

$= nlog(\frac{SS_{Res}}{n}) + nlog(2\pi) - \frac{n}{2} + 2p$

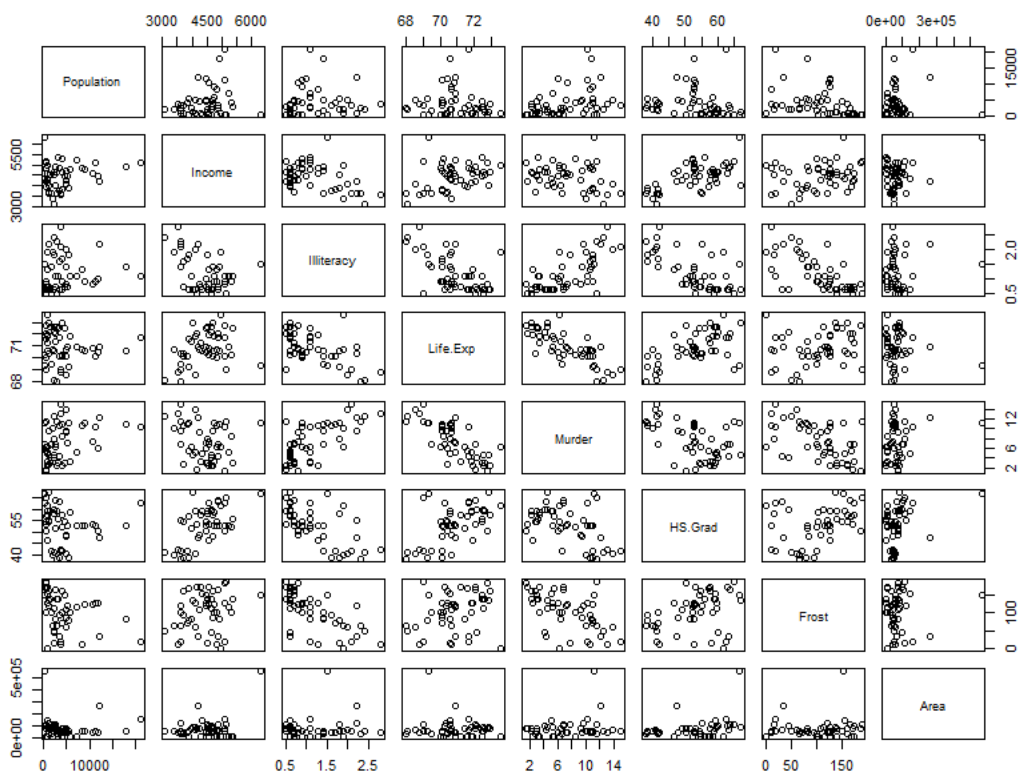
$= nlog(\frac{SS_{Res}}{n}) + 2p + const$

Question 4

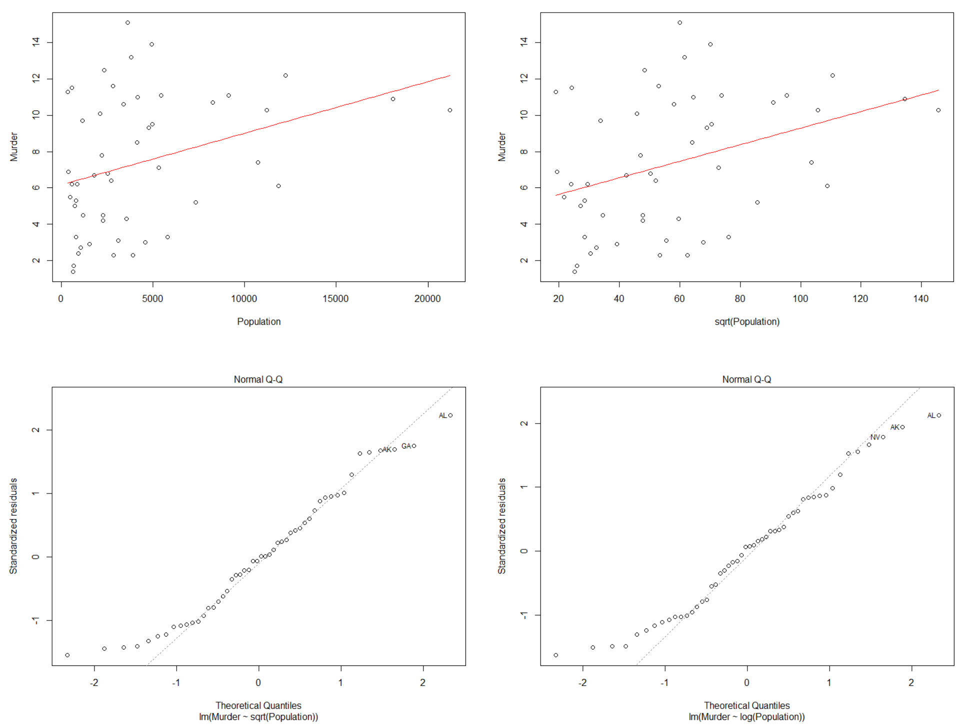
(a)

[R code]

```
data(state)
statedata = data.frame(state.x77, row.names=state.abb, check.names=TRUE)
pairs(statedata)
```

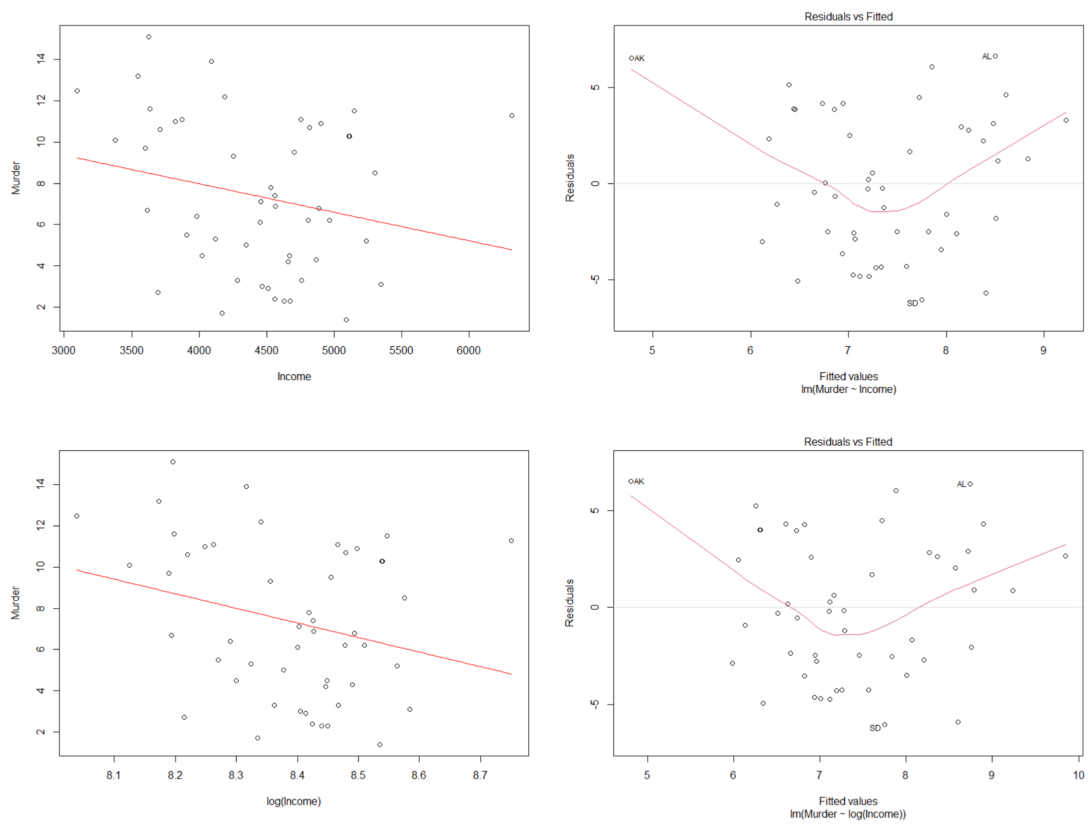


```
plot(Murder~Population, data=statedata)
m = lm(Murder~Population, data=statedata)
curve(m$coeff[1]+m$coeff[2]*x,add=T, col="red")

plot(Murder~sqrt(Population), data=statedata)
m = lm(Murder~sqrt(Population), data=statedata)
curve(m$coeff[1]+m$coeff[2]*x,add=T, col="red")
plot(m, which=2)

m = lm(Murder~log(Population), data=statedata)
plot(m, which=2)
```



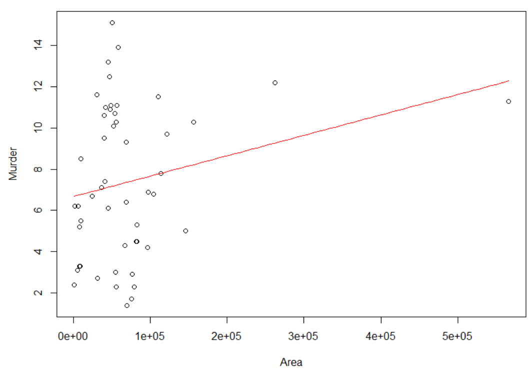
The untransformed distribution appears to be right skewed and population and murder are constrained to be positive, so a square root or logarithmic transformation appears justified. sqrt(Population) residuals seem to follow a slightly more normal distribution than log(Population) residuals and produced lower AIC scores in stepwise selection testing so a square root transformation will be applied.

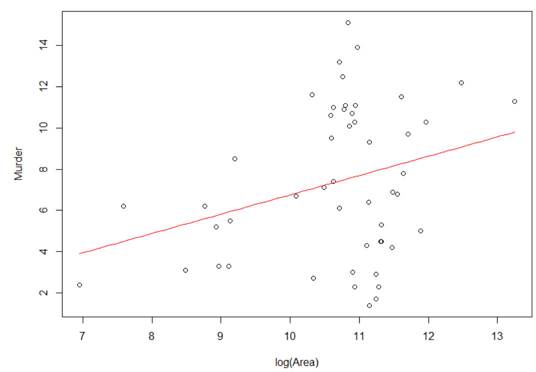
```
plot(Murder~Income, data=statedata)
m = lm(Murder~Income, data=statedata)
curve(m$coeff[1]+m$coeff[2]*x,add=T,col="red")
plot(m, which=1)


plot(Murder~log(Income), data=statedata)
m = lm(Murder~log(Income), data=statedata)
curve(m$coeff[1]+m$coeff[2]*x,add=T,col="red")
plot(m, which=1)
```



The residuals get larger on both sides of the residuals vs fitted plot, although this could be partially attributed to outliers. log(income) seems to be a slightly better fit but doesn't fully resolve the curve present on the residuals vs fitted plot.

```
plot(Murder~Area, data=statedata)
m = lm(Murder~Area, data=statedata)
curve(m$coeff[1]+m$coeff[2]*x,add=T,col="red")

plot(Murder~log(Area), data=statedata)
m = lm(Murder~log(Area), data=statedata)
curve(m$coeff[1]+m$coeff[2]*x,add=T,col="red")
```





Fit seems much better for log(Area) over untransformed Area which presents an extremely right skewed distribution. Similarly to the case of population, Area is constrained to be positive, so a logarithmic or square root transformation again seems justified.

All other variables seem to have a reasonably linear relationship with murder and don't require any transformation.

(b)

[R code]

```
> fsbasemodel = lm(Murder~1, data=statedata)
>
> add1(fsbasemodel, scope = ~ . + sqrt(Population) + log(Income) + Illiteracy +
+         Life.Exp + HS.Grad + Frost + log(Area), test="F")
Single term additions

Model:
Murder ~ 1
                 Df Sum of Sq    RSS     AIC F value    Pr(>F)
<none>                        667.75 131.594
sqrt(Population)  1     91.70 576.05 126.208  7.6411 0.0080693 **
log(Income)       1     48.01 619.74 129.864  3.7181 0.0597518 .
Illiteracy        1    329.98 337.76  99.516 46.8943 1.258e-08 ***
Life.Exp          1    407.14 260.61  86.550 74.9887 2.260e-11 ***
HS.Grad           1    159.00 508.75 119.996 15.0017 0.0003248 ***
Frost             1    193.91 473.84 116.442 19.6433 5.405e-05 ***
log(Area)         1     58.63 609.12 128.999  4.6201 0.0366687 *
---
Signif. codes:  0   ***   0.001   **   0.01   *   0.05   .   0.1
>
> fsmodel2 = lm(Murder ~ Life.Exp, data=statedata)
>
> add1(fsmodel2, scope = ~ . + sqrt(Population) + log(Income) + Illiteracy +
+         HS.Grad + Frost + log(Area), test="F")
Single term additions

Model:
Murder ~ Life.Exp
                 Df Sum of Sq    RSS    AIC F value    Pr(>F)
<none>                        260.61 86.550
sqrt(Population)  1    57.427 203.18 76.104 13.2841 0.0006673 ***
log(Income)       1     0.782 259.83 88.399  0.1414 0.7085864
Illiteracy        1    60.549 200.06 75.329 14.2249 0.0004533 ***
HS.Grad           1     1.124 259.48 88.334  0.2035 0.6539823
Frost             1    80.104 180.50 70.187 20.8575 3.576e-05 ***
log(Area)         1    30.223 230.38 82.386  6.1656 0.0166517 *
---
Signif. codes:  0   ***   0.001   **   0.01   *   0.05   .   0.1
```

```
>
> fsmodel3 = lm(Murder ~ Life.Exp + Frost, data=statedata)
>
> add1(fsmodel3, scope = ~ . + sqrt(Population) + log(Income) + Illiteracy +
+        HS.Grad  + log(Area), test="F")
Single term additions

Model:
Murder ~ Life.Exp + Frost
                 Df Sum of Sq    RSS    AIC F value   Pr(>F)
<none>                        180.50 70.187
sqrt(Population)  1   20.1383 160.37 66.272  5.7765 0.020330 *
log(Income)       1    5.1077 175.40 70.751  1.3396 0.253084
Illiteracy        1    6.0663 174.44 70.477  1.5997 0.212315
HS.Grad           1    2.0679 178.44 71.610  0.5331 0.469015
log(Area)         1   30.9733 149.53 62.774  9.5283 0.003422 **
---
Signif. codes:  0   ***   0.001   **   0.01   *   0.05   .   0.1
>
> fsmodel4 = lm(Murder ~ Life.Exp + Frost + log(Area), data=statedata)
>
> add1(fsmodel4, scope = ~ . + sqrt(Population) + log(Income) + Illiteracy +
+        HS.Grad, test="F")
Single term additions

Model:
Murder ~ Life.Exp + Frost + log(Area)
                 Df Sum of Sq    RSS    AIC F value  Pr(>F)
<none>                        149.53 62.774
sqrt(Population)  1   14.4861 135.04 59.679  4.8271 0.03321 *
log(Income)       1    4.6252 144.91 63.203  1.4364 0.23700
Illiteracy        1    8.7371 140.79 61.764  2.7925 0.10165
HS.Grad           1    0.1900 149.34 64.710  0.0572 0.81200
---
Signif. codes:  0   ***   0.001   **   0.01   *   0.05   .   0.1
```

```
>
> fsmodel5 = lm(Murder ~ Life.Exp + Frost + log(Area) + sqrt(Population),
+                data=statedata)
>
> add1(fsmodel5, scope = ~ . + log(Income) + Illiteracy + HS.Grad
+       , test="F")
Single term additions

Model:
Murder ~ Life.Exp + Frost + log(Area) + sqrt(Population)
            Df Sum of Sq    RSS    AIC F value   Pr(>F)
<none>                   135.04 59.679
log(Income)  1    1.1138 133.93 61.265  0.3659 0.54835
Illiteracy   1   13.6068 121.44 56.369  4.9301 0.03159 *
HS.Grad      1    0.0166 135.03 61.673  0.0054 0.94166
---
Signif. codes:  0   ***   0.001   **   0.01   *   0.05   .   0.1
>
> fsmodel6 = lm(Murder ~ Life.Exp + Frost + log(Area) + sqrt(Population) +
+                Illiteracy, data=statedata)
>
> add1(fsmodel6, scope = ~ . + log(Income) + HS.Grad
+       , test="F")
Single term additions

Model:
Murder ~ Life.Exp + Frost + log(Area) + sqrt(Population) + Illiteracy
            Df Sum of Sq    RSS    AIC F value Pr(>F)
<none>                   121.44 56.369
log(Income)  1    4.9259 116.51 56.299  1.8180 0.1846
HS.Grad      1    3.9559 117.48 56.713  1.4479 0.2354
```

Final model using forward selection is Murder = 107.199 -1.534*Life.Exp - 0.011*Frost + 0.654*log(Area) + 0.023*sqrt(Population) + 1.458*Illiteracy

(c)

[R code]

```
> ssfullmodel = lm(Murder ~ sqrt(Population) + log(Income) + Illiteracy
+                   Life.Exp + HS.Grad + Frost + log(Area), data=statedata)
>
> ssmodel2 = step(ssfullmodel, scope = ~ .)
Start:  AIC=58
Murder ~ sqrt(Population) + log(Income) + Illiteracy + Life.Exp +
    HS.Grad + Frost + log(Area)

                   Df Sum of Sq    RSS    AIC
- HS.Grad           1     0.702 116.51 56.299
- log(Income)       1     1.672 117.48 56.713
<none>                          115.81 57.997
- Frost             1     5.729 121.54 58.411
- sqrt(Population)  1    13.384 129.19 61.465
- Illiteracy        1    17.626 133.44 63.080
- log(Area)         1    19.300 135.11 63.704
- Life.Exp          1   122.295 238.11 92.035

Step:  AIC=56.3
Murder ~ sqrt(Population) + log(Income) + Illiteracy + Life.Exp +
    Frost + log(Area)

                   Df Sum of Sq    RSS    AIC
<none>                          116.51 56.299
- log(Income)       1     4.926 121.44 56.369
- Frost             1     6.762 123.27 57.119
+ HS.Grad           1     0.702 115.81 57.997
- sqrt(Population)  1    13.451 129.96 59.761
- Illiteracy        1    17.419 133.93 61.265
- log(Area)         1    28.557 145.07 65.259
- Life.Exp          1   130.189 246.70 91.808
```

Final model using stepwise selection is Murder = 87.228 + 0.020*sqrt(Population)
+ 2.720*log(Income) + 1.723*Illiteracy - 1.577*Life.Exp - 0.011*Frost + 0.665*log(Area)

(d)

[R code]

```
> extractAIC(ssmodel2)
[1]  7.00000 56.29858
> extractAIC(fsmodel6)
[1]  6.00000 56.36903

> summary(ssmodel2)

Call:
lm(formula = Murder ~ sqrt(Population) + log(Income) + Illiteracy +
    Life.Exp + Frost + log(Area), data = statedata)

Residuals:
    Min      1Q  Median      3Q     Max
-2.8251 -1.0773 -0.1556  0.8982  3.0617

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)      87.227514  22.619725   3.856  0.00038 ***
sqrt(Population)  0.020202   0.009067   2.228  0.03116 *
log(Income)       2.719763   2.017150   1.348  0.18462
Illiteracy        1.723245   0.679654   2.535  0.01495 *
Life.Exp         -1.577480   0.227577  -6.932 1.62e-08 ***
Frost            -0.010822   0.006851  -1.580  0.12151
log(Area)         0.664847   0.204794   3.246  0.00227 **
---
Signif. codes:  0  ***  0.001  **  0.01  *  0.05  .  0.1

Residual standard error: 1.646 on 43 degrees of freedom
Multiple R-squared:  0.8255,    Adjusted R-squared:  0.8012
F-statistic: 33.91 on 6 and 43 DF,  p-value: 9.155e-15
```

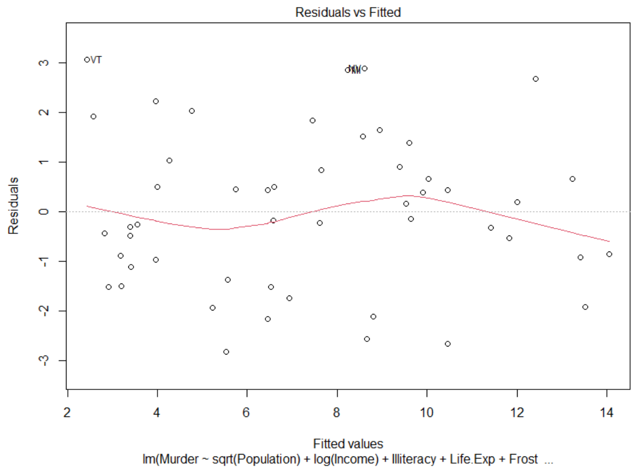
AIC of stepwise selection model < AIC of forward selection model. Therefore, we choose the stepwise selection model (from 4c) as the slightly better model.

Final fitted model:
$Murder = 87.228 + 0.020 * sqrt(Population) + 2.720 * log(Income) + 1.723 * Illiteracy - 1.577 * Life.Exp - 0.011 * Frost + 0.665 * log(Area)$

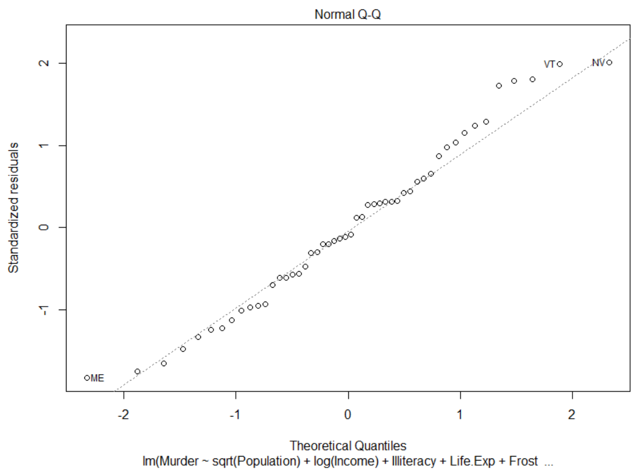
(e)

[R code]

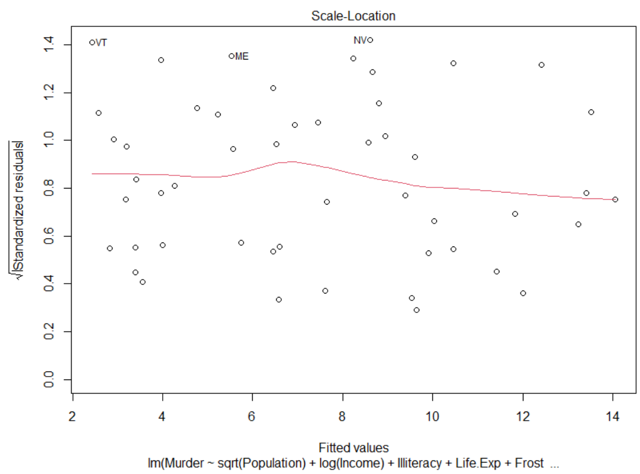
```
> plot(ssmodel2)
```



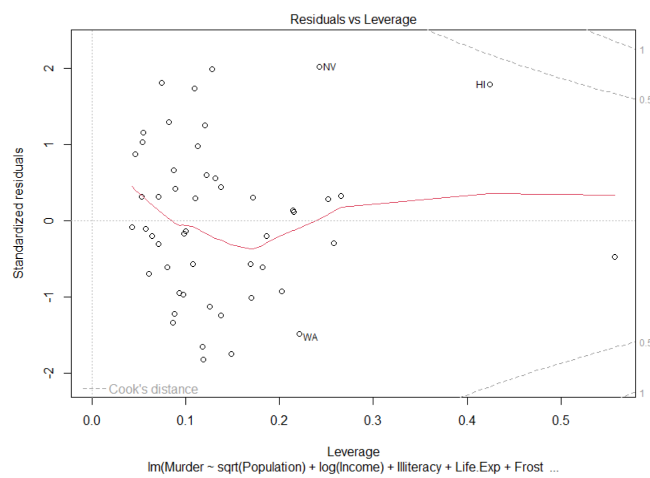
Residuals vs fitted plot seems to present fairly constant variance and seems to average around 0 without any noticeable trend.



The normal Q-Q plot shows that the standardised residuals follow a normal distribution relatively well.



The standardised residuals don't appear to follow any trend and have constant variance.



The residuals vs leverage plot appears fine.

Overall, the final fitted model doesn't appear to violate any linear model assumptions.

Question 5

(a)

$\sum_{i=1}^{n} e_i^2 + \lambda \sum_{j=0}^{k} b_j^2$
$= \vec{e}^T \vec{e} + \lambda \vec{b}^T \vec{b}$
$= (\vec{y} - X\vec{b})^T (\vec{y} - X\vec{b}) + \lambda \vec{b}^T \vec{b}$
$= \vec{y}^T \vec{y} - 2(X^T \vec{y})^T \vec{b} + \vec{b}^T (X^T X) \vec{b} + \lambda \vec{b}^T \vec{b}$

$We\ \ need\ \ \frac{\partial \vec{e}^T \vec{e}}{\partial \vec{b}} + \frac{\partial \lambda \vec{b}^T \vec{b}}{\partial \vec{b}} = 0$

$\frac{\partial}{\partial \vec{b}} \vec{y}^T \vec{y} = 0$
$\frac{\partial}{\partial \vec{b}} - 2(X^T \vec{y})^T \vec{b} = -2X^T \vec{y}$
$\frac{\partial}{\partial \vec{b}} \vec{b}^T (X^T X) \vec{b} = 2(X^T X) \vec{b}$
$\frac{\partial}{\partial \vec{b}} \lambda \vec{b}^T \vec{b} = 2\lambda \vec{b}$
$\implies -2X^T \vec{y} + 2(X^T X)\vec{b} + 2\lambda \vec{b} = 0$
$\implies (X^T X + \lambda I)\vec{b} = X^T \vec{y}$
$\implies \vec{b} = (X^T X + \lambda I)^{-1} X^T \vec{y}$

(b)

[R code]

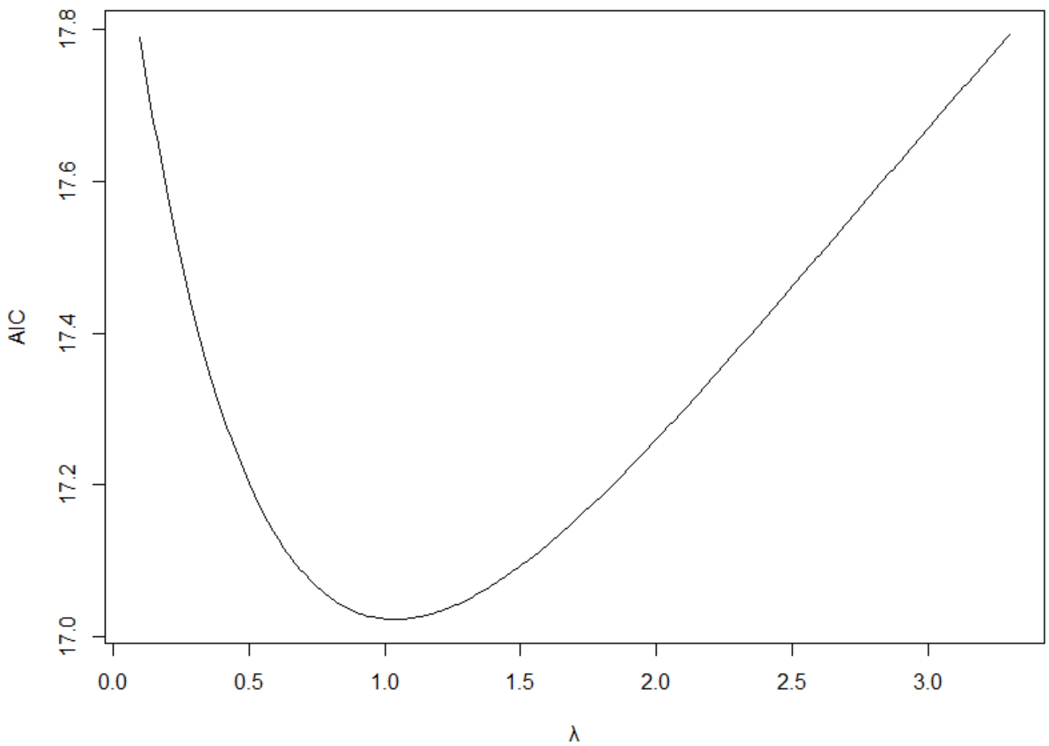
```
y = c(37.9, 42.2, 47.3, 43.1, 54.8, 47.1, 40.3)
X = matrix(c(32,19.5,13.3,13.3,5,7.1,34.5,84.9,306.6,562.0,562.0,390.6,
             2175.0,623.5,10,9,5,5,5,3,7), 7, 3)

lambda = 1.5

y = scale(y, center=TRUE, scale=FALSE)
X = scale(X, center=TRUE, scale=TRUE)


bRR = solve(t(X) %*% X + lambda * diag(3)) %*% t(X) %*% y
```

$\vec{b} = \begin{bmatrix} -3.158 \\ -1.003 \\ -1.713 \end{bmatrix}$

(c)

[R code]

```
y = c(37.9, 42.2, 47.3, 43.1, 54.8, 47.1, 40.3)
X = matrix(c(32,19.5,13.3,13.3,5,7.1,34.5,84.9,306.6,562.0,562.0,390.6,
             2175.0,623.5,10,9,5,5,5,3,7), 7, 3)

y = scale(y, center=TRUE, scale=FALSE)
X = scale(X, center=TRUE, scale=TRUE)

n = nrow(X)

AICfunction = function(lambda){n*log(SSRes/n) +
        2 * sum(diag((X %*%  solve(t(X) %*% X + lambda * diag(3)) %*% t(X))))}

xvals = seq(0.1, 3.3, 0.1)
yvals = c()

for(lambda in xvals){
  bRR = solve(t(X) %*% X + lambda * diag(3)) %*% t(X) %*% y
  e = y-X%*%bRR
  SSRes = sum(e^2)

  yvals = append(yvals, AICfunction(lambda))

}
plot(x=xvals, y=yvals, xlab="$\lambda$", ylab="AIC", type="l")
xvals[which(yvals == min(yvals))]
```



The optimal value for λ is approximately 1.04 with an AIC of 17.02245