$88/100 = 17.6/20$

# STA3301 Statistical Models

## ASSIGNMENT 1(20%)

Due: 3 September 2020

Name: Justin Ross

ID: 0050040865

## Notes on assignments

(a) Datasets can be found on the StudyDesk.

(b) Submit one file only in pdf format. Proof read your file after conversion to pdf and before submission.

(c) Working and R output must be included. Full marks will not be awarded for correct answers with insufficient working or interpretation of results.

(d) Any R output should be accompanied by a written explanation and key results from R output should be identified in the text. R output should be included in the body of the report, not an Appendix.

(e) Do not include an Appendix, it will not be marked.

(f)This assignment covers material from Modules 1 to 3.

(g) Please note that referencing text books and other resources is not the goal of this assess- ment. This work requires students to demonstrate their understanding of the analysis and interpretation, not provide quotes from resources.

---

# Question 1 (30 marks)

This data has been adapted from a subset of data collected from women by a survey team of Demographic and Health Survey (DHS) in a developing country. This dataset contains information on five variables measured on 411 women. The data file *dhs.dat* is available in the StudyDesk.

LIST OF VARIABLES:

| Variable | Description |
|---|---|
| age | Age of the women |
| weight | Weight (in kg) of women |
| height | Height (in cm) of women |
| distance | Distance (km) from home to nearest service facility |
| residence | Type of place of residence, 1=Urban, 2=Rural |

A researcher is interested in identifying if distance from home to nearest service facility, age and height of the women can be used to predict the weight of the women in thissurvey.
Using R software:

(a)( **3 marks**) Plot weight against all other quantitative variables, and comment on any relationships you see.

**R Code:**

```
par(mfrow = c(2, 2))
plot(weight ~ age, xlab="Age (years)", ylab="Weight (kgs)", main = "Weight (kgs) vs Age (years)", data=dhsVar)
plot(weight ~ height, xlab="Height (cm)", ylab="Weight (kgs)", main = "Weight (kgs) vs Height (cm)", data=dhsVar)
plot(weight ~ distance, xlab="Distance (km)", ylab="Weight (kgs)", main = "Weight (kgs) vs Distance (km)", data=dhsVar)
plot(weight ~ residence, xlab="Residence (Type)", ylab="Weight (kgs)", main = "Weight (kgs) vs Residence (Type)", data=dhsVar)
par(mfrow = c(1, 1))
```

**R Output:**

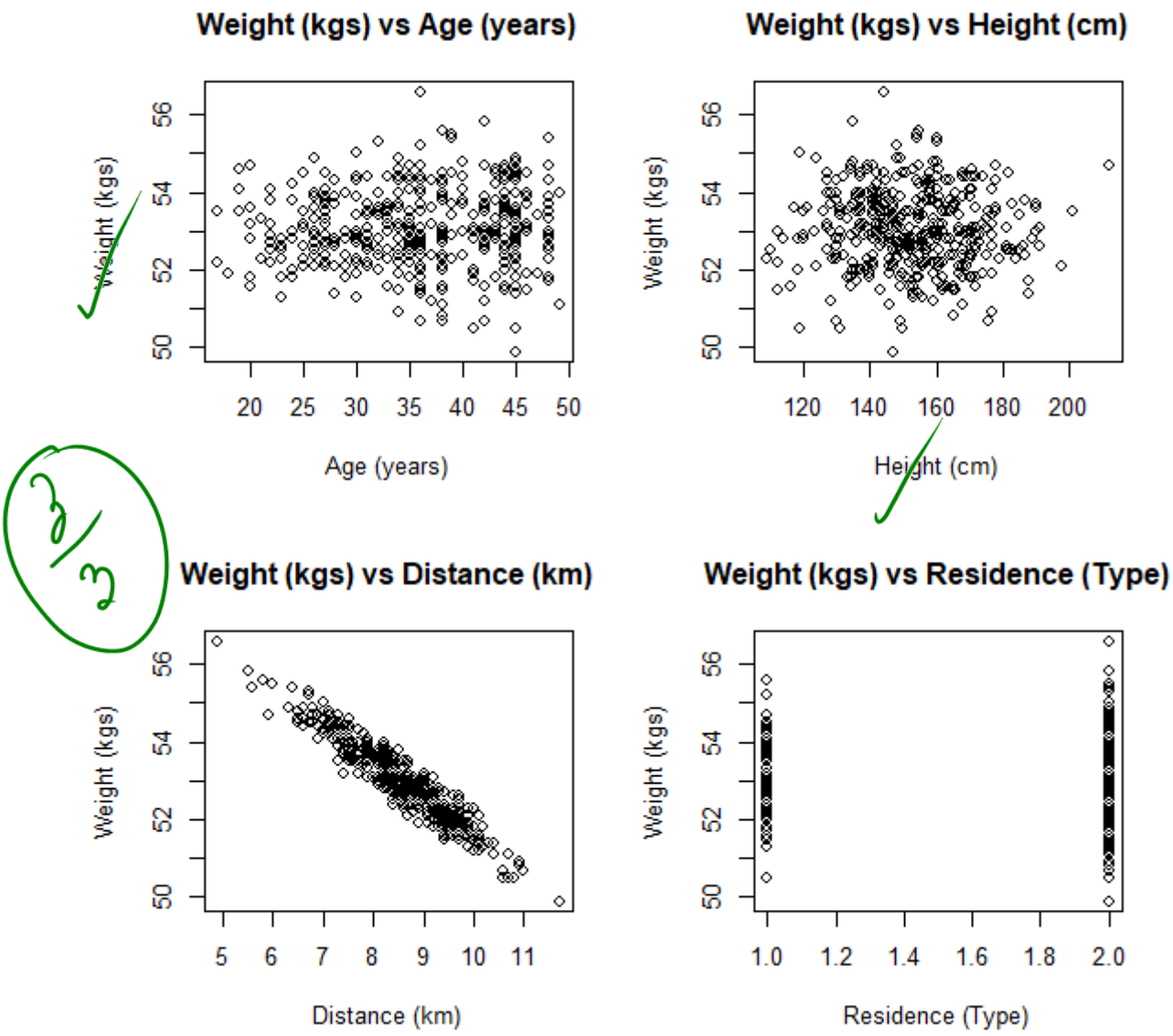



**Figure 1**

**Discussion:**

At a glance it seems that age and height have no clear impact on weight as per the top row of figure 1. Residence type seems to have an effect on scatter, with greater variance, but likely not a difference on average. The only variable that seems to have a clear relationship with weight is distance, with a clear negative correlation. The intuition appears to be that the more distance to the nearest service facility, the lower the weight of the woman.

(b)( **3 marks**) Fit a multiple linear regression model to the data (including all three quantitative independent variables). Show the final fitted model.

**R Code:**

```
le.lm.1 <- lm(weight ~ age + height + distance, data=dhsVar)
le.lm.1
summary(le.lm.1)
```

**R Output:**

```
Call:
lm(formula = weight ~ age + height + distance, data = dhsVar)

Coefficients:
(Intercept)          age       height     distance
 61.3149681   -0.0007112   -0.0013232   -0.9455581
```

```
Call:
lm(formula = weight ~ age + height + distance, data = dhsVar)

Residuals:
     Min       1Q   Median       3Q      Max
-0.86414 -0.19599  0.02437  0.21732  0.72449

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 61.3149681  0.2027979 302.345   <2e-16 ***
age         -0.0007112  0.0019792  -0.359    0.720
height      -0.0013232  0.0009060  -1.461    0.145
distance    -0.9455581  0.0155607 -60.766   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3162 on 407 degrees of freedom
Multiple R-squared:  0.9009,   Adjusted R-squared:  0.9001
F-statistic:  1233 on 3 and 407 DF,  p-value: < 2.2e-16
```

**Discussion:**

3/3

$$\widehat{weight} = \beta_0 + \beta_1 age + \beta_2 height + \beta_1 distance$$

$$\widehat{weight} = 61.3149681 + -0.0007112 age + -0.0013232 height + -0.9455581 distance$$

(c)( **2 marks**) Estimate $\sigma^2$ for the model fitted in (b).

**R Code:**

anova(le.lm.1)

**R Output:**

```
Analysis of Variance Table

Response: weight
           Df Sum Sq Mean Sq   F value Pr(>F)
age         1   0.07    0.07    0.6985 0.4038
height      1   0.55    0.55    5.5361 0.0191 *
distance    1 369.26  369.26 3692.4630 <2e-16 ***
Residuals 407  40.70    0.10
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

**Discussion:**

$\sigma^2 \approx 0.10$

1.5/2

(d)( **4 marks**) Calculate the analysis of variance table for the model in (b) to test the significance of the regression at a significance level of $\alpha = 0.05$. What conclusions can you draw? Place the R output correctly into an ANOVA table.

**R Code:**

anova(le.lm.1)

**R Output:**

```
Analysis of Variance Table

Response: weight
           Df Sum Sq Mean Sq   F value Pr(>F)
age         1   0.07    0.07    0.6985 0.4038
height      1   0.55    0.55    5.5361 0.0191 *
distance    1 369.26  369.26 3692.4630 <2e-16 ***
Residuals 407  40.70    0.10
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

**Discussion:** at least one of the regressors statistically different from zero

| ANOVA Table | | | | |
|---|---|---|---|---|
| Source | Sum-of Squares | df | Mean-Square | F-Ratio |
| Regression | 0.07 + 0.55 + 369.26 = 369.88 | 3 | 369.88/1 = 369.88 | 369.88/0.1 = 3,698.8 |
| Error | 40.7 | 407 | 0.1 | |
| Total | 410.58 | 408 | | |

2/4

$H_0: \beta = 0$

$H_a:$ at least one of them is not zero

$\beta = (\beta_0, \beta_1, \beta_2, \beta_3)$

(e)( **2 marks**) Using the model in (b) predict *weight* when distance = 7.5*km*, age = 35*years* and height = 132*cm*.

**R Code:**

```
predict(le.lm.1, newdata = data.frame(distance = 7.5, age = 35, height = 132))
```

**R Output:**

```
       1
54.02373
```

**Discussion:**

The model predicts that weight will be 54.02373 kgs when distance = 7.5km, age = 35years and height = 132cm.

(f)( **2 marks**) Find a 99% prediction interval for *weight* when distance = 7.5*km*, age = 35*years* and height = 132*cm*.

**R Code:**

```
ge.lm.300 <- predict(le.lm.1, newdata = data.frame(distance = 7.5, age = 35, height = 132),
interval = "prediction", level=0.99)
ge.lm.300
```

**R Output:**

```
       fit      lwr      upr
1 54.02373 53.20194 54.84553
```

**Discussion:**

This 99% prediction interval suggests that the weight for an individual is likely to be between 54.02373 kgs and 54.84553 kgs when distance = 7.5km, age = 35years and height = 132cm.

(g)( **2 marks**) Find a 99% confidence interval for the mean *weight* when distance = 7.5*km*, age = 35*years* and height = 132*cm*. Explain the meaning of a confidence interval. How do your prediction and confidence intervals compare?

**R Code:**

```
ge.lm.400 <- predict(le.lm.1, newdata = data.frame(distance = 7.5, age = 35, height = 132),
interval = "confidence", level=0.99)
ge.lm.400
```

**R Output:**

```
       fit      lwr      upr
1 54.02373 53.94914 54.09832
```

**Discussion:**

This 95% confidence interval suggests that the mean weight is between 54.02373 kgs and 54.09832 kgs when distance = 7.5*km*, age = 35*years* and height = 132*cm*.

(h)( **2 marks**) Compute the correlation matrix for the three quantitative regressors. Comment on the strength and direction of correlation between the regressors.

**R Code:**

```
dhsVarThreeCor <- dhsVar[, c(1,3,4)]
cor(dhsVarThreeCor)
```

**R Output:**

```
                 age      height    distance
age        1.00000000 0.03301367 -0.02045978
height     0.03301367 1.00000000  0.01399430
distance -0.02045978 0.01399430  1.00000000
```

**Discussion:**

There is not particularly strong relationship between the regressors.

The strongest of the correlations is between age and height, which is to be expected given that the age value starts at 0 and hence as someone grows older they also grow taller and then start to decline in height with age. However, this is still not a significant figure with a value of 0.03301367 likely because most of the women in the data are adults with only a few adolescents.

There is a negative correlation between age and distance and, but it is not a strong value at -0.02045978. This suggests that perhaps people move closer to their service facilities as they age, though not a significant correlation.

There is a slight positive correlation between height and distance, though a very weak correlation with a coefficient of 0.01399430.

(i)( **4 marks**) Calculate the partial correlation of each quantitative regressor with the dependent variable. Explain what these partial correlations mean. How do they relate to the relationships you identified in part (a)?

**R Code:**

```
partialr <- function(y, xi, otherx) {
  y.otherx.lm <- lm(y ~ otherx)
  y.otherx.res <- resid(y.otherx.lm)
  xi.otherx.lm <- lm(xi ~ otherx)
  xi.otherx.res <- resid(xi.otherx.lm)
  partial.corr <- cor(y.otherx.res, xi.otherx.res)
  return(partial.corr)
}

partialr(dhsVar$weight, dhsVar$age, cbind(dhsVar$height, dhsVar$distance))
partialr(dhsVar$weight, dhsVar$height, cbind(dhsVar$age, dhsVar$distance))
partialr(dhsVar$weight, dhsVar$distance, cbind(dhsVar$age, dhsVar$height))
```

4/4

**R Output:**

```
> partialr(dhsVar$weight, dhsVar$age, cbind(dhsVar$height, dhsVar$distance))
[1] -0.01780894
>
> partialr(dhsVar$weight, dhsVar$height, cbind(dhsVar$age, dhsVar$distance))
[1] -0.07220731
>
> partialr(dhsVar$weight, dhsVar$distance, cbind(dhsVar$age, dhsVar$height))
[1] -0.949062
```

**Discussion:**

The partial correlation of weight with age is: -0.01780894
The partial correlation of weight with height is: -0.07220731
The partial correlation of weight with distance is: -0.949062

The partial correlation measures the correlation of a variable with the response variable after removing the effects of other variables. In other words, when the impact of the other variables has been nullified, how much does the given variable correlation with the response variable.

Clearly, the strongest relationship is between distance and weight, which visual inspection of the data suggested in part a. The relationship is clearly negative with a partial correlation of -0.949062. This value represents the relationship between distance and weight when the effects of age and height have been removed.

(j)( **2 marks**) Based on the results from (h) and (i), which regressors are likely to be important in predicting the response?

Distance is likely to be the only important regressor when predicting weight as the response variable. The reason for this is that there is little correlation between regressors and the partial correlation of age and height with weight was weak verging on non-existent, whereas the partial correlation of distance with weight was strongly negative with a coefficient of -0.949062.

(k)( **2 marks**) After fitting the model in (b) test the statistical significance of each regressor (one at a time) at the $\alpha = 0.05$ level.

**R Code:**
summary(le.lm.1)

**R Output:**

```
Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 61.3149681  0.2027979 302.345   <2e-16 ***
age         -0.0007112  0.0019792  -0.359    0.720
height      -0.0013232  0.0009060  -1.461    0.145
distance    -0.9455581  0.0155607 -60.766   <2e-16 ***
```

**Discussion:**

Distance seems very important with a p-value approaching 0. The other items have p-values above the level of significance of 0.05 with the significance of the relationship between age and weight having a p-value of 0.720 and the significance of the relationship between height and weight having a p-value of 0.145.

2/2

(l)( **2 marks**) From the available data what model do you think would provide the 'best' prediction of the outcome of interest (*weight*)? Explain why. State the final fitted model

**R Code:**

```
dhsfinlm <- lm(weight ~ distance, data=dhsVar)
dhsfinlm
summary(dhsfinlm)
```

**R Output:**

```
Call:
lm(formula = weight ~ distance, data = dhsVar)

Coefficients:
(Intercept)     distance
    61.0891      -0.9458
```

```
Call:
lm(formula = weight ~ distance, data = dhsVar)

Residuals:
     Min       1Q   Median       3Q      Max
-0.89047 -0.19742  0.02275  0.21495  0.74987

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 61.08911    0.13319  458.67   <2e-16 ***
distance    -0.94576    0.01556  -60.78   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.3164 on 409 degrees of freedom
Multiple R-squared:  0.9003,   Adjusted R-squared:  0.9001
F-statistic:  3694 on 1 and 409 DF,  p-value: < 2.2e-16
```

**Discussion:**

A simple model that only includes distance as an independent variable as follows:

$$\widehat{weight} = \beta_0 + \beta_1 distance$$

$$\widehat{weight} = 61.08911 - 0.94576\ distance$$

The reason this model is likely to be best is that there does not particularly appear to be any correlation between age and weight, or height and weight, and so including those variables in the model would not particularly improve it. The only significant correlation is between weight and distance, as the above R output indicates with the significance of the relationship having a p-value approaching 0.

# Question 2 (30 marks)

The Boston Housing Dataset is derived from information collected by the U.S. Census Service concerning housing in the area of Boston MA. The following describes the variables listed in the dataset. The data file *housing.dat* is available in the StudyDesk.

LIST OF VARIABLES:

| Variable | Description |
|---|---|
| zn | proportion of residential land zoned for lots over 25,000 sq.ft. |
| indus | proportion of non-retail business acres per town. |
| nox | nitric oxides concentration (parts per 10 million) |
| rm | average number of rooms per dwelling |
| age | proportion of owner-occupied units built prior to 1940 |
| dis | weighted distances to five Boston employment centres |
| medv | Median value of owner-occupied homes in $1000's |

(a)( **4 marks**) Plot medv against any four of the other variables that you think might be strongly related to Median value of owner-occupied homes (in $1000's). Comment on the relationships you observe.

**R Code:**

```
par(mfrow = c(2, 2))
plot(medv ~ nox + rm + age + dis, data=housingVar)
par(mfrow = c(1, 1))
```

**R Output:**

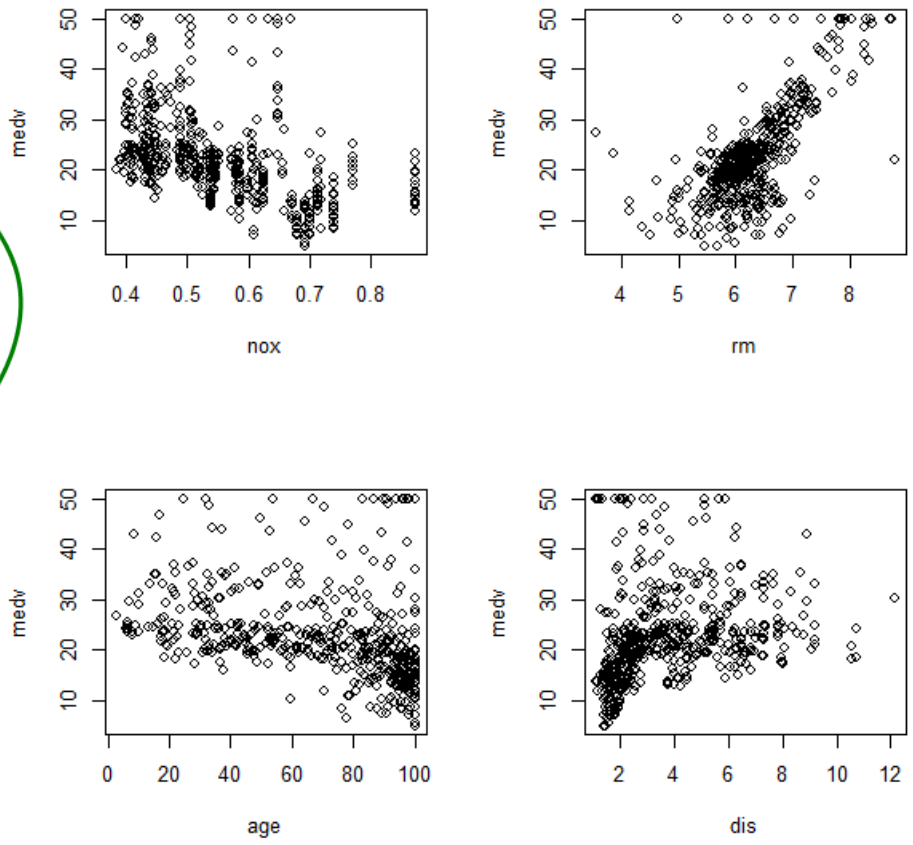



**Figure 2**

**Discussion:**

For the variables I have chosen nox, rm, age and dis.

Nitric Oxide (nox) concentrates would presumably have been a concern. Although there is a slight negative correlation between the median value and nitric oxide as per the plot on the top left of figure 2, the correlation is not particularly strong. It more seems like the high value homes start to drop off but the low value do not.

Average rooms per dwelling (rm) is an indicator of the size of a home and thus should have a correlation with its median value. The plotted data appears to show this relationship clearly as per the top right graph of figure 2.

Age of the property (age) was chosen on the basis that older homes might have been worth less, but the distribution based on age is quite horizontal and there seems to be some quite high value homes that are old, likely some sort of heritage value being added to the properties offsetting their lack of modernity.

Distance to the nearest employment centre (dis) ought to be an indicator of distance from the city in general. It seemed that this would have a clear correlation based on this hypothesis, but curiously there seems to be a lot of low value residences, likely units, close to the city center.

(b)( **5 marks**) Using all possible predictors select a model using forward regression. Give the final fitted model. Explain each step of the forward selection process from your output. How did you decide on the best model?

**R Code:**

step(lm(housingVar$medv ~ 1), ~housingVar$zn + housingVar$indus + housingVar$nox + housingVar$rm + housingVar$age + housingVar$dis, test = "F", direction = "forward")

**R Output:**

```
Start:  AIC=2246.51
housingVar$medv ~ 1

                   Df Sum of Sq   RSS    AIC F value    Pr(>F)
+ housingVar$rm     1   20654.4 22062 1914.2 471.847 < 2.2e-16 ***
+ housingVar$indus  1    9995.2 32721 2113.6 153.955 < 2.2e-16 ***
+ housingVar$nox    1    7800.1 34916 2146.5 112.591 < 2.2e-16 ***
+ housingVar$age    1    6069.8 36647 2171.0  83.478 < 2.2e-16 ***
+ housingVar$zn     1    5549.7 37167 2178.1  75.258 < 2.2e-16 ***
+ housingVar$dis    1    2668.2 40048 2215.9  33.580 1.207e-08 ***
<none>                          42716 2246.5
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=1914.19
housingVar$medv ~ housingVar$rm

                   Df Sum of Sq   RSS    AIC F value    Pr(>F)
+ housingVar$indus  1   2254.26 19808 1861.7  57.245 1.846e-13 ***
+ housingVar$nox    1   2217.51 19844 1862.6  56.208 2.966e-13 ***
+ housingVar$age    1   1996.99 20065 1868.2  50.062 5.025e-12 ***
+ housingVar$zn     1    974.47 21087 1893.3  23.244 1.893e-06 ***
+ housingVar$dis    1    512.56 21549 1904.3  11.964 0.0005884 ***
<none>                          22062 1914.2
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=1861.65
housingVar$medv ~ housingVar$rm + housingVar$indus

                  Df Sum of Sq   RSS    AIC F value   Pr(>F)
+ housingVar$age   1    383.32 19424 1853.8  9.9064 0.001745 **
+ housingVar$nox   1    322.30 19485 1855.3  8.3034 0.004127 **
+ housingVar$dis   1    212.44 19595 1858.2  5.4425 0.020047 *
+ housingVar$zn    1    100.86 19707 1861.1  2.5692 0.109591
<none>                         19808 1861.7
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=1853.76
housingVar$medv ~ housingVar$rm + housingVar$indus + housingVar$age

                  Df Sum of Sq   RSS    AIC F value   Pr(>F)
+ housingVar$dis   1    896.19 18528 1831.9 24.2331 1.16e-06 ***
+ housingVar$nox   1     93.64 19331 1853.3  2.4268   0.1199
<none>                         19424 1853.8
+ housingVar$zn    1     11.06 19413 1855.5  0.2854   0.5934
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=1831.86
housingVar$medv ~ housingVar$rm + housingVar$indus + housingVar$age +
    housingVar$dis

                  Df Sum of Sq   RSS    AIC F value    Pr(>F)
+ housingVar$nox   1    429.01 18099 1822.0 11.8517 0.0006244 ***
+ housingVar$zn    1    271.22 18257 1826.4  7.4279 0.0066474 **
<none>                         18528 1831.9
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Step:  AIC=1822
housingVar$medv ~ housingVar$rm + housingVar$indus + housingVar$age +
    housingVar$dis + housingVar$nox

                 Df Sum of Sq   RSS  AIC F value   Pr(>F)
+ housingVar$zn   1    354.99 17744 1814  9.9831 0.001676 **
<none>                        18099 1822
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Step:  AIC=1813.98
housingVar$medv ~ housingVar$rm + housingVar$indus + housingVar$age +
    housingVar$dis + housingVar$nox + housingVar$zn


Call:
lm(formula = housingVar$medv ~ housingVar$rm + housingVar$indus +
    housingVar$age + housingVar$dis + housingVar$nox + housingVar$zn)

Coefficients:
     (Intercept)      housingVar$rm  housingVar$indus    housingVar$age    housingVar$dis
housingVar$nox     housingVar$zn
         -1.19094           7.22274          -0.26083          -0.05798          -1.64846         -
16.40970           0.05021
```

**Discussion:**

Forward selection starts with only the model intercept and no further predictors and then adds them one-by-one until they no longer improve the model to a statistically significant extent. The model which achieves the lowest AIC is considered the best one.

It started with only an intercept and then added rm as that had the lowest AIC (1914.19). Afterwards it re-examined the correlations and added indus, as indus had the lowest AIC after rm was removed from selection choices with a value of 1861.65. This patterned continued with the adding of age, dis, nox and zn. The ordering of choices is determined by the lowest AIC, but the decision to add or not to add is based on the p-value being below 0.05 at that particular step. As every step of the way the p-value was lower than the alpha 0.05 the full model was the model selected.

The model selected by forward regression was the full model:

$$\widehat{medv} = \beta_0 + \beta_1 rm + \beta_2 indus + \beta_3 age + \beta_4 dis + \beta_5 nox + \beta_6 zn$$

$$\widehat{medv} = -1.19094 + 7.22274 rm + -0.26083 indus + -0.05798 age + -1.64846 dis + -16.40970 nox + 0.05021 zn$$

(c)( **5 marks**) Select and describe the best model using backward elimination. Explain the backward regression process in general terms (i.e. not in terms of your output, simply define the process).

**R Code:**

step(lm(medv ~ zn + indus + nox + rm + age + dis, data=housingVar), test = "F", direction = "backward")

**R Output:**

```
Start:  AIC=1813.98
medv ~ zn + indus + nox + rm + age + dis

        Df Sum of Sq   RSS    AIC  F value    Pr(>F)
<none>               17744 1814.0
- zn     1     355.0 18099 1822.0   9.9831 0.0016757 **
- age    1     498.4 18243 1826.0  14.0154 0.0002024 ***
- nox    1     512.8 18257 1826.4  14.4205 0.0001642 ***
- indus  1     560.9 18305 1827.7  15.7746 8.186e-05 ***
- dis    1    1577.7 19322 1855.1  44.3668 7.208e-11 ***
- rm     1   10309.5 28054 2043.8 289.9230 < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Call:
lm(formula = medv ~ zn + indus + nox + rm + age + dis, data = housingVar)

Coefficients:
(Intercept)           zn        indus          nox           rm          age          dis
   -1.19094      0.05021     -0.26083    -16.40970      7.22274     -0.05798     -1.64846
```

**Discussion:**

The model selected by backward elimination was also the full model, though ordered slightly differently due to selection ordering as follows:

$$\widehat{medv} = \beta_0 + \beta_1 zn + \beta_2 indus + \beta_3 nox + \beta_4 rm + \beta_5 age + \beta_6 dis$$

$$\widehat{medv} = -1.19094 + 0.05021 zn + -0.26083 indus + -16.40970 nox + 7.22274 rm + -0.05798 age + -1.64846 dis$$

The backward selection process starts with all regressors, otherwise called the full model, and then starts by eliminating those that do not have a statistically significant effect on the response variable.

(d)( **6 marks**) Exclude zn and age from the possible predictors to consider. Compute adjusted $R^2$, *PRESS*, *AIC*, $C_m$ and *m* for all the remaining possible models (except the model with only the constant term). On the basis of this information, reduce the pool of models from 15 to a small subset for further consideration. Select a model, justifying your choice. Give the final fitted model.

**R Code:**

```
model.info <- function(model, s2) {
  R2.adj <- summary(model)$adj.r.squared
  PRESS <- sum((resid(model)/(1 - hatvalues(model)))^2)
  SSE <- sum(resid(model)^2)
  m <- length(coef(model))
  Cm <- SSE/s2 - (length(resid(model)) - 2 * m)
  aic <- AIC(model)
  return(matrix(data = c(R2.adj, PRESS, Cm, m, AIC = aic), ncol = 5, nrow = 1))
}

co.md.full <- lm(medv ~ indus + nox + rm + dis, data=housingVar)
s2 <- summary(co.md.full)$sigma^2
info.matrix <- array(dim = c(15, 5))
colnames(info.matrix) <- c("adjusted R^2", "PRESS", "Cm","m", "AIC")

info.matrix[1, ] <- model.info(lm(medv ~ indus, data=housingVar), s2 = s2)
info.matrix[2, ] <- model.info(lm(medv ~ nox, data=housingVar), s2 = s2)
info.matrix[3, ] <- model.info(lm(medv ~ rm, data=housingVar), s2 = s2)
info.matrix[4, ] <- model.info(lm(medv ~ dis, data=housingVar), s2 = s2)
info.matrix[5, ] <- model.info(lm(medv ~ indus + nox, data=housingVar), s2 = s2)
info.matrix[6, ] <- model.info(lm(medv ~ indus + rm, data=housingVar), s2 = s2)
info.matrix[7, ] <- model.info(lm(medv ~ indus + dis, data=housingVar), s2 = s2)
info.matrix[8, ] <- model.info(lm(medv ~ nox + rm, data=housingVar), s2 = s2)
info.matrix[9, ] <- model.info(lm(medv ~ nox + dis, data=housingVar), s2 = s2)
info.matrix[10, ] <- model.info(lm(medv ~ rm + dis, data=housingVar), s2 = s2)
info.matrix[11, ] <- model.info(lm(medv ~ indus + nox + rm, data=housingVar), s2 = s2)
info.matrix[12, ] <- model.info(lm(medv ~ indus + nox + dis, data=housingVar), s2 = s2)
info.matrix[13, ] <- model.info(lm(medv ~ indus + rm + dis, data=housingVar), s2 = s2)
info.matrix[14, ] <- model.info(lm(medv ~ nox + rm + dis, data=housingVar), s2 = s2)
info.matrix[15, ] <- model.info(lm(medv ~ indus + nox + rm + dis, data=housingVar), s2 = s2)

info.matrix
```

**R Output:**

```
      adjusted R^2    PRESS        Cm m      AIC
 [1,]   0.23247017 32999.00 373.00184 2 3551.601
 [2,]   0.18098122 35165.35 431.70043 2 3584.455
 [3,]   0.48250070 22373.63  87.96122 2 3352.151
 [4,]   0.06060418 40358.06 568.93305 2 3653.843
 [5,]   0.23902526 32756.48 365.80759 3 3548.256
 [6,]   0.53445466 20210.18  29.67947 3 3299.612
 [7,]   0.24819267 32376.91 355.37727 3 3542.123
 [8,]   0.53359085 20240.52  30.66227 3 3300.550
 [9,]   0.19461399 34640.07 416.33702 3 3576.957
[10,]   0.49351878 21951.30  76.25472 3 3342.257
[11,]   0.54111748 19943.91  23.06080 4 3293.311
[12,]   0.27914900 31081.27 320.52583 4 3521.840
[13,]   0.53853036 20032.71  25.99846 4 3296.156
[14,]   0.54209884 19895.51  21.94646 4 3292.228
[15,]   0.55790375 19227.15   5.00000 5 3275.445
```

**Discussion:**

Model 6, 11, 13, 14, 15 are a good subset of models to choose from as they have somewhat high $R^2$ and low PRESS, $C_m$ and AIC values. Of them model 15 is the best choice as it has the highest $R^2$ and the lowest Press, $C_m$ and AIC values as follows:

Using $R^2$ the best model is: model 15 - 0.55790375
Using PRESS the best model is: model 15 - 19227.15
Using $C_m$ the best model is: model 15 - 5.00000
Using AIC the best model is: model 15 - 3275.445

The model selected by $R^2$, PRESS, $C_m$, AIC was model 15, which in this case was the full model:

$$\widehat{medv} = \beta_0 + \beta_1 indus + \beta_2 nox + \beta_3 rm + \beta_4 dis$$

$$\widehat{medv} = -1.19094 + -0.26083 indus + -16.40970 nox + 7.22274 rm + -1.64846 dis$$

(e)( **5 marks**) Using the *cv.glm ()* function in R, determine the variables that seem to be important in each case of Ridge, LASSO and Elastic-Net regression considering all data and compare the results. Write down the full model in each case.

**R Code:**

```
#Use the following r code for finding optimum value of λ using all data.
set.seed(1234)
y_md<-matrix(housingVar$medv) # y_md is the dependent variable, Time (4th variable).
#y_md<-matrix(md[, 4]) # y_md is the dependent variable, Time (4th variable).
#x_md_factor<-
model.matrix(housingVar$medv~housingVar$nox+housingVar$rm+housingVar$age+housing
Var$dis)[,-1]
x_md_factor<-model.matrix()[,-1]
# model.matrix is used to use categorical variables in cv.glmnet.

#x_md<-as.matrix(data.frame(Dose, x_md_factor))

x_md<-as.matrix(data.frame(housingVar$zn + housingVar$nox + housingVar$rm +
housingVar$age + housingVar$dis,x_md_factor))
# x_md is a matrix of data that we will use Ridge, Lasso and
# Elastic-Net Regression to predict y_md.

alpha0.fit_md<-cv.glmnet(as.matrix(x_md), y_md, alpha=0)
# The "cv" part means we want to use Cross Validation to obtain the optimal
# values for lambda. This call to cv.glmnet() will fit a Linear Regression
# with a Ridge Regression penalty using 10-fold Cross # Validation to find
# optimal value of lambda.
# best value of lambda.
best_lam_md=alpha0.fit_md$lambda.min
best_lam_md

# The best value of lambda is 44.15745.
predict(alpha0.fit_md, type="coefficients", s=best_lam_md, as.matrix(x_md))
# Regression coefficients using optimum lambda
#5 x 1 sparse Matrix of class "dgCMatrix"

#The ridge regression suggests that all four variables (clinic, status, prison and dose)
#should be in the model. This is because all the coefficients are moderately large.

# Prediction
alpha0.predicted_md<-predict(alpha0.fit_md, s=best_lam_md, as.matrix(x_md))
# predicted values
sqrt(mean((y_md-alpha0.predicted_md)^2)) # Root Mean Squared Error (RMSE)


### LASSO ###

set.seed(1234)
alpha1.fit_md<-cv.glmnet(as.matrix(x_md), y_md, alpha=1)
best_lam_lasso_md=alpha1.fit_md$lambda.min
predict(alpha1.fit_md, type="coefficients", s=best_lam_lasso_md)
best_lam_lasso_md

# Prediction
```

```
alpha1.predicted_md<-predict(alpha1.fit_md, s=best_lam_lasso_md, as.matrix(x_md))
# predicted values
sqrt(mean((y_md-alpha1.predicted_md)^2)) # RMSE


### Elastic-Net ###

set.seed(1234)
list.of.fits<-list() # Empty list that will store a bunch of Elastic Net Regression fits.
for (i in 0:10){
  fit.name<-paste0("alpha", i/10)
  list.of.fits[[fit.name]]<-
    cv.glmnet(as.matrix(x_md), y_md, alpha=i/10)
}
# For loop is used to try different values for alpha. We created the Elastic-Net fit
# using the cv.glmnrt() function. When i=0, the result in Ridge regression, and
# when i=1, resulting in Lasso regression.
results<-data.frame() # Empty data.frame that will store MSE and other things.
for (i in 0:10){
  fit.name<-paste0("alpha", i/10)
  predicted_en<-predict(list.of.fits[[fit.name]],
                        s=list.of.fits[[fit.name]]$lambda.min, as.matrix(x_md))
  rmse<-sqrt(mean((y_md-predicted_en)^2))
  temp<-data.frame(alpha=i/10, rmse=rmse, fit.name=fit.name)
  results<-rbind(results,temp)
}
results

alpha0.8.fit_en_md<-cv.glmnet(as.matrix(x_md), y_md, alpha=0.8)
best_lam_en_md=alpha0.8.fit_en_md$lambda.min
best_lam_en_md
predict(alpha0.8.fit_en_md, type="coefficients", s=best_lam_en_md)
# Regression coefficients using optimum lambda and alpha


# Prediction
alpha0.8.predicted_en_md<-predict(alpha0.8.fit_en_md,
                                  s=best_lam_en_md, as.matrix(x_md)) # predicted values
sqrt(mean((y_md-alpha0.8.predicted_en_md)^2)) # RMSE
```

**R Output:**

```
### Ridge regression ###

> best_lam_md
[1] 0.6388975

> predict(alpha0.fit_md, type="coefficients", s=best_lam_md, as.matrix(x_md))
6 x 1 sparse Matrix of class "dgCMatrix"
                                                                            1
(Intercept)                                                      -6.32162828
housingVar.zn...housingVar.nox...housingVar.rm...housingVar.age...  0.03210478
housingVar.nox                                                  -19.60485291
housingVar.rm                                                     7.33083725
housingVar.age                                                   -0.08247277
housingVar.dis                                                   -0.94722084

> # Prediction
> alpha0.predicted_md<-predict(alpha0.fit_md, s=best_lam_md, as.matrix(x_md))
> # predicted values
```

```
> sqrt(mean((y_md-alpha0.predicted_md)^2)) # Root Mean Squared Error (RMSE)
[1] 6.054196
>
> ### LASSO ###
>
> predict(alpha1.fit_md, type="coefficients", s=best_lam_lasso_md)
6 x 1 sparse Matrix of class "dgCMatrix"
                                                                              1
(Intercept)                                                         -4.06768850
housingVar.zn...housingVar.nox...housingVar.rm...housingVar.age...   0.05112522
housingVar.nox                                                     -23.09247767
housingVar.rm                                                        7.62695807
housingVar.age                                                      -0.11205607
housingVar.dis                                                      -1.44114079

> best_lam_lasso_md
[1] 0.01041244

> # Prediction
> sqrt(mean((y_md-alpha1.predicted_md)^2)) # RMSE
[1] 6.014927

> ### Elastic-Net ###

> set.seed(1234)
> list.of.fits<-list() # Empty list that will store a bunch of Elastic Net
Regression fits.
> results
   alpha     rmse fit.name
1    0.0 6.054196   alpha0
2    0.1 6.015485 alpha0.1
3    0.2 6.015193 alpha0.2
4    0.3 6.014859 alpha0.3
5    0.4 6.014865 alpha0.4
6    0.5 6.014998 alpha0.5
7    0.6 6.014867 alpha0.6
8    0.7 6.014925 alpha0.7
9    0.8 6.014913 alpha0.8
10   0.9 6.014854 alpha0.9
11   1.0 6.014927   alpha1

> best_lam_en_md
[1] 0.01185928

> predict(alpha0.8.fit_en_md, type="coefficients", s=best_lam_en_md)
6 x 1 sparse Matrix of class "dgCMatrix"
                                                                              1
(Intercept)                                                         -4.04982653
housingVar.zn...housingVar.nox...housingVar.rm...housingVar.age...   0.05121134
housingVar.nox                                                     -23.10318404
housingVar.rm                                                        7.62545250
housingVar.age                                                      -0.11214248
housingVar.dis                                                      -1.44228334

> # Regression coefficients using optimum lambda and alpha
> # Prediction
> sqrt(mean((y_md-alpha0.8.predicted_en_md)^2)) # RMSE
[1] 6.014913
```

**Discussion:**

The ridge analysis suggests that rm and nox are the most important with values of 7.33083725 and -19.60485291 respectively. The model suggested is subsequently:

$$\widehat{medv} = \beta_0 + \beta_1 nox + \beta_2 rm$$

$$\widehat{medv} = -6.32162828 + -19.60485291 nox + 7.33083725 rm$$

The lasso analysis suggests similar with rm and nox the stand outs with values of 7.62695807 and -23.09247767. The model suggested is subsequently:

$$\widehat{medv} = \beta_0 + \beta_1 nox + \beta_2 rm$$

$$\widehat{medv} = -4.06768850 + -23.09247767 nox + 7.62695807 rm$$

The elastic-net analysis suggests that rm and nox are the most important variable with a 7.6254 positive correlation for rm and a negative correlation of -23.1031 (rounded down) for nox. The model suggested is subsequently:

$$\widehat{medv} = \beta_0 + \beta_1 nox + \beta_2 rm$$

$$\widehat{medv} = -4.04982653 + -23.10318404 nox + 7.62545250 rm$$

(f)( **5 marks**) Using the *cv.glm ()* function in R, determine the variables that seem to be important in each case of Linear, Ridge, LASSO and Elastic-Net regressions considering 80%, 20% split (80% training and 20% testing data) and compare the results. Write down the full model in each case.

**R Code:**

```
### Machine Learning Training ###


md <- read.table("C://Users//Justin//Desktop//Semester 2 2020//STA3301 Statistical
Models//Assignment 1//dhs.dat", header = TRUE)

set.seed(1234) # we use this so that we can repeatable result
md_new<-md
attach(md_new)
ind<-sample(2, nrow(md_new), replace=T, prob=c(0.8, 0.2))
# We are going to take two independent samples (sampling with
# replacement) with 80% 20% split.
train<-md_new[ind==1,] # 80% goes to training
test<-md_new[ind==2,] # 20% goes to testing
#Libraries Needed
custom<-trainControl(method="repeatedcv", number=10, repeats=10, verboseIter = T)
# VerboseIter is used so that we know what is going on when the model is running.

### Linear ###

set.seed(1234)
lm_md<-train(Time~., train, method='lm', trControl=custom)
# Results
lm_md$results
lm_md
summary(lm_md)


### Ridge regression ###

set.seed(1234)
ridge_md<-train(Time~., train, method='glmnet',
tuneGrid=expand.grid(alpha=0,lambda=seq(0.1, 300, length=30)), trControl=custom)
# The best value for lamba is 20.8.
plot(ridge_md)

print(ridge_md)

plot(ridge_md$finalModel, xvar="lambda",label=T)


plot(ridge_md$finalModel, xvar="dev",label=T) # Fraction deviance explained
plot(varImp(ridge_md, scale=T))

# best model
ridge_md$bestTune

best_md<-ridge_md$finalModel
```

```
coef(best_md, s=ridge_md$bestTune$lambda)

# Prediction
p1_ridge_md<-predict(ridge_md, train)
sqrt(mean((train$Time-p1_ridge_md)^2)) # RMSE for training data
p2_ridge_md<-predict(ridge_md, test)
sqrt(mean((test$Time-p2_ridge_md)^2)) # RMSE for test data


### LASSO regression ###

set.seed(1234)
lasso_md<-train(Time~., train, method='glmnet', tuneGrid=expand.grid(alpha=1,
lambda=seq(0.001, 40, length=400)), trControl=custom)
# The best value for lambda is 0.402.
plot(lasso_md) # For higher value of lamda, the error increases, we
# get the best value of lambda is 0.402.
print(lasso_md)
plot(lasso_md$finalModel, xvar="lambda",label=T)

plot(lasso_md$finalModel, xvar="dev",label=T)
# About 20% of the variability explained by three variables.
plot(varImp(lasso_md, scale=T))
# Variable importance: Similar to ridge regression, Prison is the most important
# variable, followed by Status and Clinic.
# best model
lasso_md$bestTune

best_lasso_md<-lasso_md$finalModel
coef(best_lasso_md, s=lasso_md$bestTune$lambda)

# Prediction
p1_lasso_md<-predict(lasso_md, train)
sqrt(mean((train$Time-p1_lasso_md)^2)) # RMSE for training data

p2_lasso_md<-predict(lasso_md, test)
sqrt(mean((test$Time-p2_lasso_md)^2)) # RMSE for test data

### Elastic Net Regression ###

set.seed(1234)
en_md<-train(Time~., train, method='glmnet', tuneGrid=expand.grid(alpha=seq(0.001, 1,
length=10), lambda=seq(0.01, 40, length=40)), trControl=custom)
en_md
# It gives us optimal values of the parameters alpha= 0.001, and lambda=23.6.
# plot results
plot(en_md) # This is the mixing percentage of alpha and lambda.
print(en_md)

plot(en_md$finalModel, xvar="lambda",label=T)
# when log lambda is over 10, we can see all coefficients are almost 0.
# and the model has no variable. When log lambda is about less than 10,
# all coefficients grow and we have all four variables in the model.
plot(en_md$finalModel, xvar="dev",label=T)
```

```
# We can see similar pattern that have been seen ridge and lasso.
plot(varImp(en_md, scale=T))
# Variable importance: Prison is the most important variable, followed
# by Status and Prison. Similar to Ridge and LASSO.
# best model
en_md$bestTune

best_en_md<-en_md$finalModel
coef(best_en_md, s=en_md$bestTune$lambda)

# Prediction
p1_en_md<-predict(en_md, train)
sqrt(mean((train$Time-p1_en_md)^2)) # RMSE for training data

p2_en_md<-predict(en_md, test)
sqrt(mean((test$Time-p2_en_md)^2)) # RMSE for test data


# compare models using taring data
model_list_md<-list(LinearModel=lm_md, Ridge=ridge_md,
                    Lasso=lasso_md, ElasticNet=en_md)
res_md<-resamples(model_list_md)
# This is a par of caret package that will help us to compare the models.
summary(res_md)
# It gives us summary in the form of Min, Q1, Median,Mean, Q3 Max values
# for MAE, RMSE and R^2 for each of the model.
```

3/5

**R Output:**

✓ See Solution

**Discussion:**

I ran out of time on this question and my custom coding of the question beforehand made copying the code difficult. I just added the code above to demonstrate that I understood the machine learning 80/20 split as well as the implementation of the ridge, lasso and elastic-net scripts. I just could not feed the data into it.

# Question 3 (24 marks)

Consider the *dhs.dat* dataset again. The dataset also contains information of residence from Urban and Rural women. Based on the information in the dataset, answer the following questions.

(a)( **6 marks**) Fit a single regression model to predict weight of women from distance (ignore other variables), with separate slopes and intercepts for Urban and Rural residence. Give each of the final fitted models.

**R Code:**

```
pt.lm.3 <- lm(dhsVarResFact$weight ~ dhsVarResFact$distance *
dhsVarResFact$residence)
pt.lm.3
summary(pt.lm.3)
```

**R Output:**

```
Call:
lm(formula = dhsVarResFact$weight ~ dhsVarResFact$distance *
    dhsVarResFact$residence)

Coefficients:
                                    (Intercept)
dhsVarResFact$distance                                  dhsVarResFact$residence2
                                        60.6886                                                    -
0.9014                                                   0.5411
dhsVarResFact$distance:dhsVarResFact$residence2
                                        -0.0599
```

```
Call:
lm(formula = dhsVarResFact$weight ~ dhsVarResFact$distance *
    dhsVarResFact$residence)

Residuals:
     Min       1Q   Median       3Q      Max
-0.81834 -0.19724  0.03342  0.21406  0.74117

Coefficients:
                                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                                  60.68856    0.26188 231.739   <2e-16 ***
dhsVarResFact$distance                       -0.90138    0.03067 -29.391   <2e-16 ***
dhsVarResFact$residence2                      0.54113    0.30394   1.780   0.0758 .
dhsVarResFact$distance:dhsVarResFact$residence2 -0.05990    0.03557  -1.684   0.0929 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3157 on 407 degrees of freedom
Multiple R-squared:  0.9012,      Adjusted R-squared:  0.9005
F-statistic:  1238 on 3 and 407 DF,  p-value: < 2.2e-16
```

**Discussion:**

When residence=1 the model is:

$$\widehat{weight} = \beta_0 + \beta_1 distance + (\beta_2 * 1)$$

$$\widehat{weight} = 60.68856 + -0.90138 distance + (0.54113 * 1)$$

When residence=2 the model is:

$$\widehat{weight} = \beta_0 + \beta_1 distance + (\beta_2 * 2)$$

$$\widehat{weight} = 60.68856 + -0.90138 distance + (0.54113 * 2)$$

(b)( **4 marks**) Determine if the slopes are different for Urban and Rural residence.

**R Code:**

dhs.lm <- lm(weight ~ distance * residence, data=dhsVarResFact) # This model produces separate intercepts and slopes for the two Locations.
dhs.lm
summary(dhs.lm)

**R Output:**

```
Call:
lm(formula = weight ~ distance * residence, data = dhsVarResFact)

Coefficients:
        (Intercept)             distance           residence2
distance:residence2
            60.6886              -0.9014               0.5411              -
0.0599
```

```
Call:
lm(formula = weight ~ distance * residence, data = dhsVarResFact)

Residuals:
     Min       1Q   Median       3Q      Max
-0.81834 -0.19724  0.03342  0.21406  0.74117

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         60.68856    0.26188 231.739   <2e-16 ***
distance            -0.90138    0.03067 -29.391   <2e-16 ***
residence2           0.54113    0.30394   1.780   0.0758 .
distance:residence2 -0.05990    0.03557  -1.684   0.0929 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3157 on 407 degrees of freedom
Multiple R-squared:  0.9012,   Adjusted R-squared:  0.9005
F-statistic:  1238 on 3 and 407 DF,  p-value: < 2.2e-16
```

**Discussion:**

For residence one the slope is: -0.90138
For residence two the slope is: -0.96128

This difference is not considered significant as $\beta_4$ has a p-value of 0.0929 which is greater than our level of significance of $\alpha$=0.05. This in turn means that there is no significant difference between the slopes based on rural and urban residences.

(c)( **4 marks**) Determine if the intercepts are different for Urban and Rural residence.

**R Code:**

```
dhs.lm2 <- lm(weight ~ distance + residence, data=dhsVarResFact)
dhs.lm2
summary(dhs.lm2)
```

**R Output:**

```
Call:
lm(formula = weight ~ distance + residence, data = dhsVarResFact)

Coefficients:
(Intercept)     distance   residence2
   61.06631     -0.94592      0.03274
```

```
Call:
lm(formula = weight ~ distance + residence, data = dhsVarResFact)

Residuals:
     Min       1Q   Median       3Q      Max
-0.86651 -0.20200  0.02267  0.21290  0.74126

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 61.06631    0.13548 450.735   <2e-16 ***
distance    -0.94592    0.01557 -60.771   <2e-16 ***
residence2   0.03274    0.03546   0.923    0.356
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3164 on 408 degrees of freedom
Multiple R-squared:  0.9005,   Adjusted R-squared:    0.9
F-statistic:  1847 on 2 and 408 DF,  p-value: < 2.2e-16
```

**Discussion:**

As $H_0$: $\beta_2 = 0$ is not significant, with a p-value of 0.356 being greater than our critical value of $\alpha$=0.05, there is no significant difference between the intercepts of urban and rural residences.

(d)( **6 marks**) Does there appear to be a difference in the Urban and Rural residence?

**R Code:**

```
ms.lm.3 <- lm(weight ~ distance * residence, data=dhsVarResFact)
ms.lm.3
summary(ms.lm.3)
```

**R Output:**

```
Call:
lm(formula = weight ~ distance * residence, data = dhsVarResFact)

Coefficients:
        (Intercept)            distance           residence2
distance:residence2
            60.6886             -0.9014              0.5411             -
0.0599
```

```
Call:
lm(formula = weight ~ distance * residence, data = dhsVarResFact)

Residuals:
     Min       1Q   Median       3Q      Max
-0.81834 -0.19724  0.03342  0.21406  0.74117

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)
(Intercept)         60.68856    0.26188 231.739   <2e-16 ***
distance            -0.90138    0.03067 -29.391   <2e-16 ***
residence2           0.54113    0.30394   1.780   0.0758 .
distance:residence2 -0.05990    0.03557  -1.684   0.0929 .
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.3157 on 407 degrees of freedom
Multiple R-squared:  0.9012,   Adjusted R-squared:  0.9005
F-statistic:  1238 on 3 and 407 DF,  p-value: < 2.2e-16
```

**Discussion:**

3/6

There does not appear to be a significant difference between the Urban and Rural residences as both the intercepts and slopes were not significantly different enough.

Test $H_0: (\beta_1, \beta_3) = (0, 0)$

$H_a: (\beta_1, \beta_3) \neq (0, 0)$

(e)( **4 marks**) Now, plot the data (using a different symbol for Urban and Rural residence) and the fitted regression line(s), and comment on any differences between the Urban and Rural residence.

**R Code:**

```
plot(dhsVarResFact$weight ~ dhsVarResFact$distance, las = 1, type = "n", xlab =
"Distance (in km)", ylab = "Weight (in kg)")
points(dhsVarResFact$weight[dhsVarResFact$residence == 1] ~
dhsVarResFact$distance[dhsVarResFact$residence == 1], pch = 1)
points(dhsVarResFact$weight[dhsVarResFact$residence == 2] ~
dhsVarResFact$distance[dhsVarResFact$residence == 2], pch = 19)
legend("topright", pch = c(1, 19), legend = c("Urban Residence (1)", "Rural Residence
(2)"))


par(mar=c(4, 4, 1, 0.5))
plot(dhsVarResFact$weight ~ dhsVarResFact$distance, las = 1, type = "n", xlim=c(0,15),
ylim=c(50,65), xlab = "distance (kms)",
    ylab = "Weight (in kgs)")
points(dhsVarResFact$weight[dhsVarResFact$residence == 1] ~
dhsVarResFact$distance[dhsVarResFact$residence == 1], pch = 1)
points(dhsVarResFact$weight[dhsVarResFact$residence == 2] ~
dhsVarResFact$distance[dhsVarResFact$residence == 2], pch = 19)
legend("topright", pch = c(1, 19), legend = c("Urban Residence (1)", "Rural Residence (2)"))
x<-seq(from=-5, to=25, by=0.5)
locA<-ms.lm.3$coef[1]+(ms.lm.3$coef[2]*x)
lines(x,locA,lwd=1)
locB<-ms.lm.3$coef[1]+ ms.lm.3$coef[3]+((ms.lm.3$coef[2] + ms.lm.3$coef[4])*x)
lines(x,locB,lwd=2)
```

**R Output:**

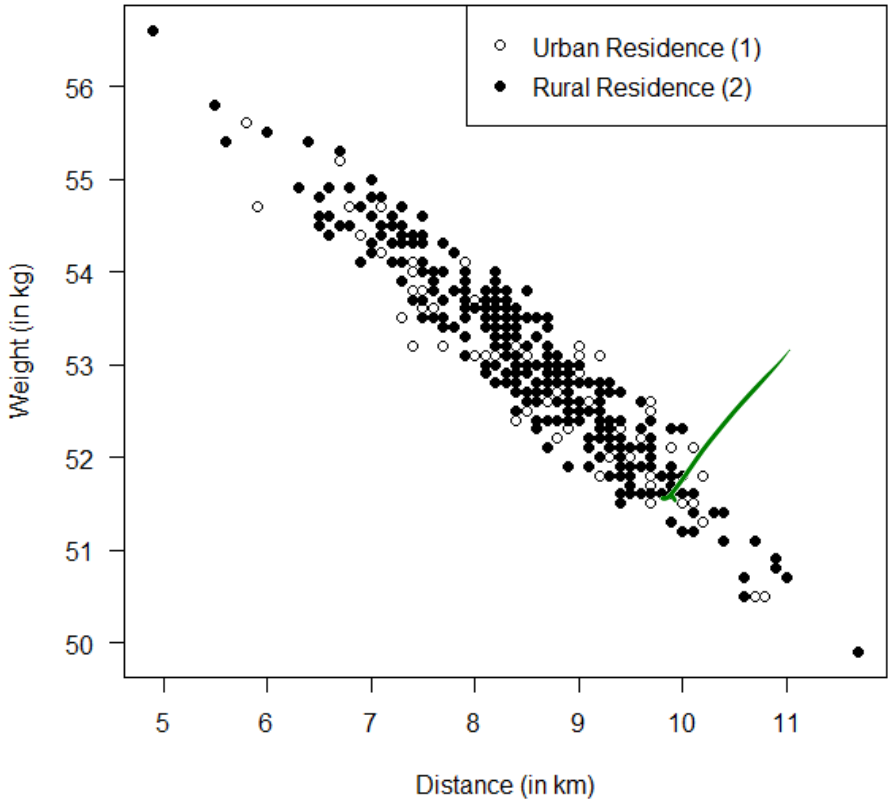



**Figure 3**

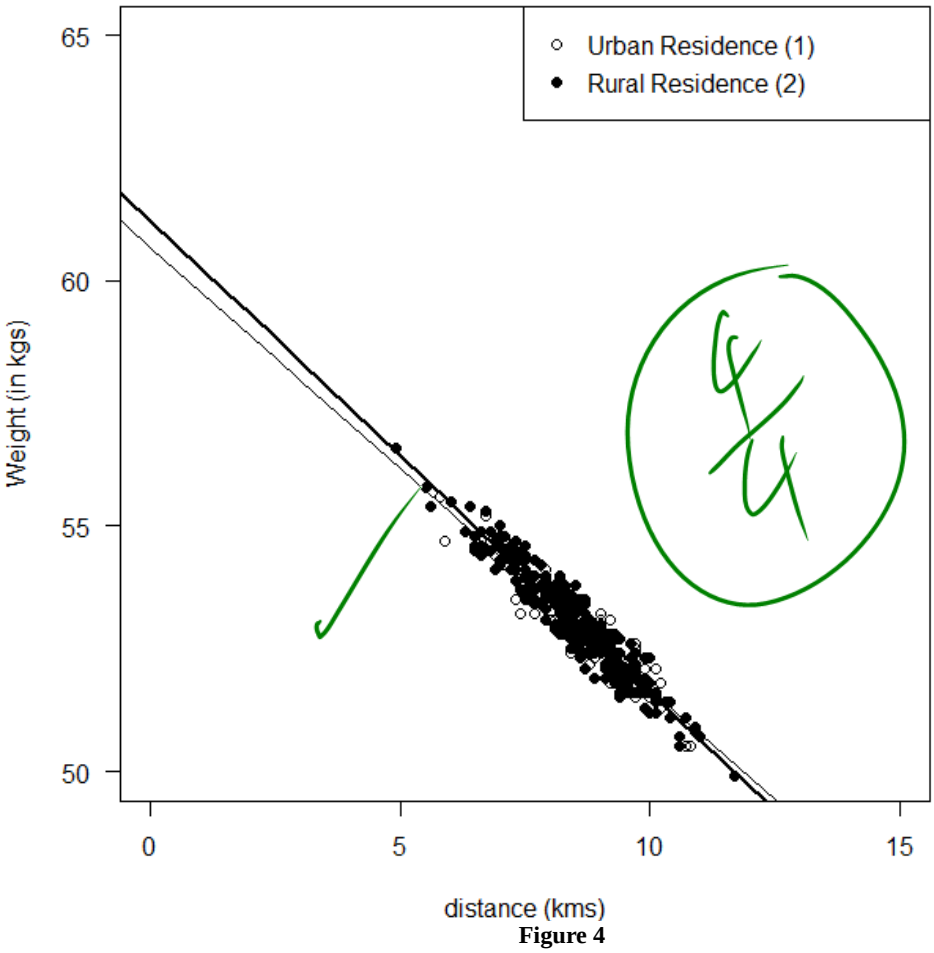



**Figure 4**

**Discussion:**

There appears to be more variance in weight for the rural residence, but there does not appear to be a strong difference between the two. Overall, there is not a strong difference between urban and rural residence types when it comes to weight after visually inspecting the data. At a glance the lines of fit are slightly different, with different points of intercepts, but the analysis concluded that the difference in slopes and intercepts was not significant. Thus we can conclude that there is no significant difference between urban and rural residences insofar as determining weight is concerned.

# Question 4 (16 marks)

Consider the data set *grade.dat*. The data set consists of four assignments and exam marks of 380 students along with their gender and faculty. The variable of interest here exam mark.
The data set *grade.dat* is available in the StudyDesk.

LIST OF VARIABLES:

| Variable | Description |
|---|---|
| gender | Gender of students, 1=Male, 2=Female. |
| assign1 | Assignment 1 Marks of students. |
| assign2 | Assignment 2 Marks of students. |
| assign3 | Assignment 3 Marks of students. |
| assign4 | Assignment 1 Marks of students. |
| faculty | Faculty of students, 1= BELA-Business, Education, Law and Arts<br>2= HES- Health, Engineering, and Sciences |
| exam | Exam Marks of students. |

(a)( **4 marks**) Obtain five additional predictors in *R* by calculating the product of the variable *gender* with each of the variables *assign*1, *assign*2, *assign*3, *assign*4 and *faculty* (this means you will now have 11 possible predictors). Using the six independent variables as well as the five additional predictors obtain a linear regression model for *exam* marks. State and comment on the final fitted model.

**R Code:**

```
gradeVar.lm <-lm(exam ~ factor(gender) + assign1 + assign2 + assign3 + assign4 +
factor(faculty) + factor(gender)*assign1 + factor(gender)*assign2 + factor(gender)*assign3 +
factor(gender)*assign4 + factor(gender)*factor(faculty), data = gradeVar)

gradeVar.lm
summary(gradeVar.lm)
```

**R Output:**

```
Call:
lm(formula = exam ~ factor(gender) + assign1 + assign2 + assign3 +
    assign4 + factor(faculty) + factor(gender) * assign1 + factor(gender) *
    assign2 + factor(gender) * assign3 + factor(gender) * assign4 +
    factor(gender) * factor(faculty), data = gradeVar)

Coefficients:
                  (Intercept)                 factor(gender)2                         assign1
assign2
                      -0.72374                      -0.61329                         0.01643
-0.08480
                       assign3                       assign4              factor(faculty)2
factor(gender)2:assign1
                       0.12646                       0.84693                      -2.85040
-0.05250
       factor(gender)2:assign2       factor(gender)2:assign3       factor(gender)2:assign4
factor(gender)2:factor(faculty)2
                       0.05049                       0.05782                      -0.05624
-0.76122
```

```
Call:
lm(formula = exam ~ factor(gender) + assign1 + assign2 + assign3 +
    assign4 + factor(faculty) + factor(gender) * assign1 + factor(gender) *
    assign2 + factor(gender) * assign3 + factor(gender) * assign4 +
    factor(gender) * factor(faculty), data = gradeVar)

Residuals:
    Min      1Q  Median      3Q     Max
-30.850  -6.830   0.724   6.099  37.071

Coefficients:
                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                   -0.72374    1.85919  -0.389   0.6973
factor(gender)2               -0.61329    2.81801  -0.218   0.8278
assign1                        0.01643    0.05129   0.320   0.7490
assign2                       -0.08480    0.05658  -1.499   0.1348
assign3                        0.12646    0.04895   2.584   0.0102 *
assign4                        0.84693    0.04754  17.813   <2e-16 ***
factor(faculty)2              -2.85040    1.71393  -1.663   0.0971 .
factor(gender)2:assign1       -0.05250    0.08066  -0.651   0.5155
factor(gender)2:assign2        0.05049    0.08572   0.589   0.5562
factor(gender)2:assign3        0.05782    0.07533   0.767   0.4433
factor(gender)2:assign4       -0.05624    0.06987  -0.805   0.4214
factor(gender)2:factor(faculty)2 -0.76122    2.51653  -0.302   0.7624
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 11.93 on 368 degrees of freedom
Multiple R-squared:  0.8406,   Adjusted R-squared:  0.8358
F-statistic: 176.4 on 11 and 368 DF,  p-value: < 2.2e-16
```

**Discussion:**

When gender=1 and faculty=1 the model is:

$$\widehat{exam} = \beta_0 + (\beta_1 * 1) + \beta_2 assign1 + \beta_3 assign2 + \beta_4 assign3 + \beta_5 assign4 + (\beta_6 * 1) + (\beta_7 assign1 * 1) + (\beta_8 assign2 * 1) + (\beta_9 assign3 * 1) + (\beta_{10} assign4 * 1) + (1 * 1)$$

When gender=1 and faculty=2 the model is:

$$\widehat{exam} = \beta_0 + (\beta_1 * 1) + \beta_2 assign1 + \beta_3 assign2 + \beta_4 assign3 + \beta_5 assign4 + (\beta_6 * 2) + (\beta_7 assign1 * 1) + (\beta_8 assign2 * 1) + (\beta_9 assign3 * 1) + (\beta_{10} assign4 * 1) + (1 * 2)$$

When gender=2 and faculty=1 the model is:

$$\widehat{exam} = \beta_0 + (\beta_1 * 2) + \beta_2 assign1 + \beta_3 assign2 + \beta_4 assign3 + \beta_5 assign4 + (\beta_6 * 1) + (\beta_7 assign1 * 2) + (\beta_8 assign2 * 2) + (\beta_9 assign3 * 2) + (\beta_{10} assign4 * 2) + (2 * 1)$$

When gender=2 and faculty=2 the model is:

$$\widehat{exam} = \beta_0 + (\beta_1 * 2) + \beta_2 assign1 + \beta_3 assign2 + \beta_4 assign3 + \beta_5 assign4 + (\beta_6 * 2) + (\beta_7 assign1 * 2) + (\beta_8 assign2 * 2) + (\beta_9 assign3 * 2) + (\beta_{10} assign4 * 2) + (2 * 2)$$

(b)( **4 marks**) For the model in (a) compute standardised residuals and use these residuals in the following plots to assess the model:

- normal probability plot
- residuals against the fitted values
- residuals against any two quantitative predictors

Based on this residual analysis, comment on whether or not there appear to be 'suspect' data points (that is, assignment marks for which data should be checked).

**R Code:**

```
gradeVar.stdres = rstandard(gradeVar.lm)

qqnorm(gradeVar.stdres, ylab="Standardized Residuals", xlab="Normal Scores",  main="QQ
plot of Standardized Grades")
qqline(gradeVar.stdres)

plot(rstandard(gradeVar.lm) ~ fitted(gradeVar.lm))

par(mfrow = c(1, 1))
plot(rstandard(gradeVar.lm) ~ gradeVar$assign1 + gradeVar$assign2)
par(mfrow = c(1, 1))
```

**R Output:**

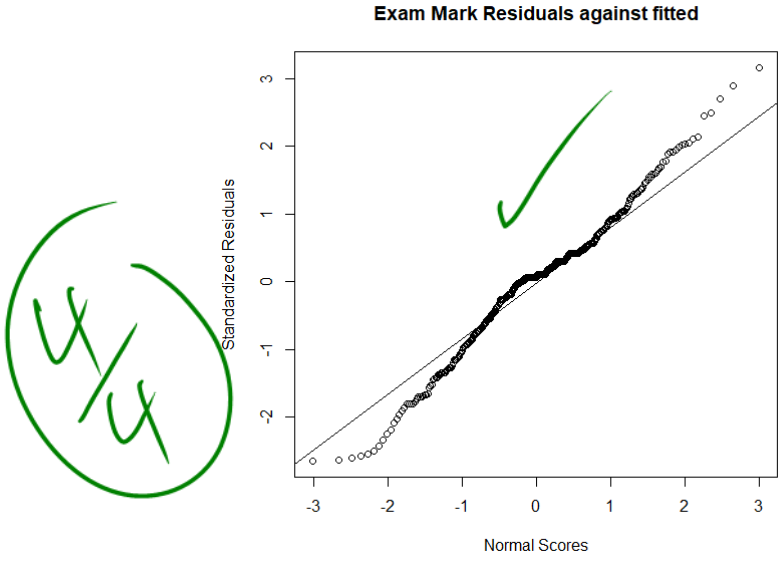



**Figure 5**

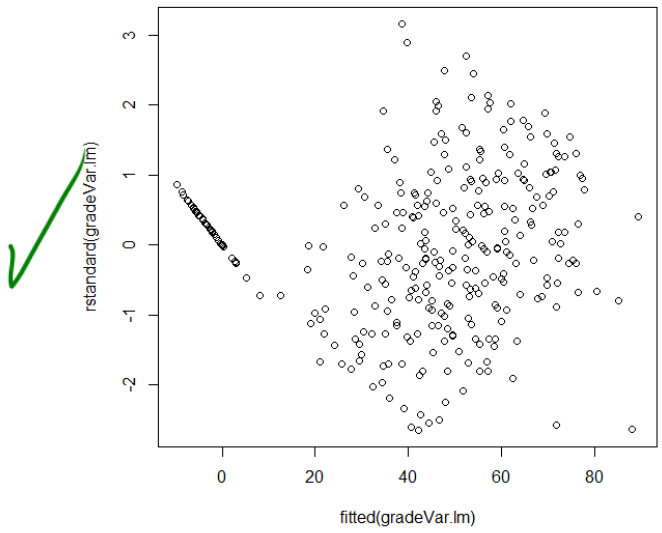



**Figure 6**

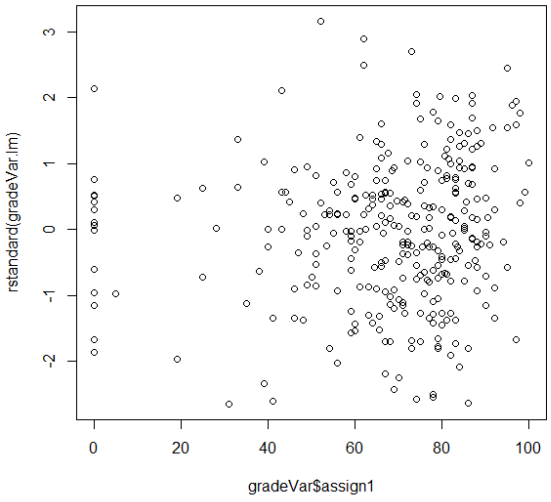



**Figure 7**

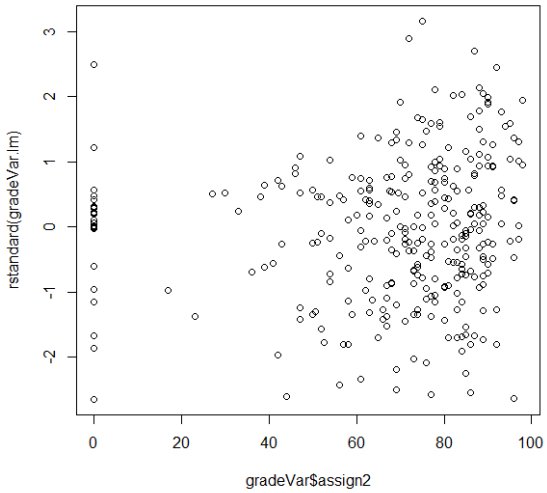



**Figure 8**

**Discussion:**

The normal probability plot (figure 5) has a bit of an s-shape and is slightly flattened on both ends, indicating that it has a shorter tail than the normality distribution. This in turn suggests that normality has been violated.

When the residuals were plotted against the normal values in figure 6 we can begin to see further issues. There is something of a vertical funnel shape that seems to have developed largely due to a collection of grades at the 0 mark, as indicated by an almost straight line on the middle-left of the plot. This suggests more grades around the 0 mark than would be expected from the regression.

When the residuals were plotted against two of the quantitative predictors, namely assign1 and assign2 in figures 7 and 8 respectively, some oddities did emerge. Overall this seems to suggest that there are more grades around the 0 mark than the regression would suggest.

The main issue identified is that same of the grades that are 0 are perhaps undeserving of that mark. It is hard to say, though, as these might simply be assessment items that were not completed.

(c)( **5 marks**) Continue the analysis of this model by computing the following influence diagnostics:

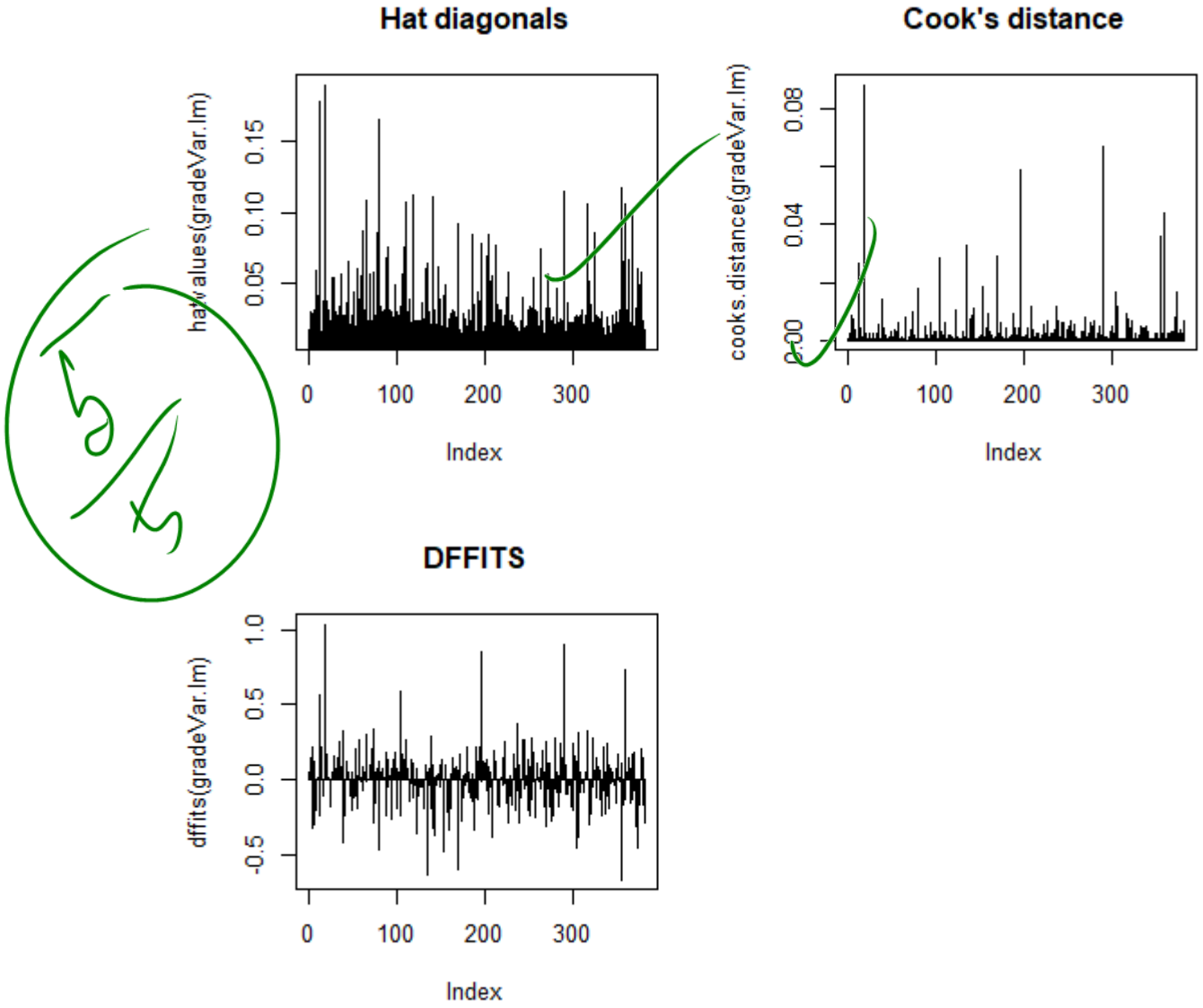
- HAT diagonals,
- Cook's Distance,
- *DFFITS*.

Discuss which, if any, days appear influential.

**R Code:**

```
par(mfrow = c(2, 2))
plot(hatvalues(gradeVar.lm), type = "h", main = "Hat diagonals")
plot(cooks.distance(gradeVar.lm), type = "h", main = "Cook's distance")
plot(dffits(gradeVar.lm), type = "h", main = "DFFITS")
par(mfrow = c(1, 1))
```

**R Output:**




**Discussion:**

The Hat diagonals, cook's distance and dffits test all appear at a glance to show that there are quite a few grades that ought to be investigated. According to the tests there seems to be two

grades roughly around observation 20 that should be investigated. One of the grades around observation 100 appears worth investigating according to the hat diagonals and another at 200 and 300 according to Cook's and DFFits. There also appears to be two observations roughly around 350 that ought to be investigated.

(d)( **3 marks**) Check for any evidence of multi-collinearity among the regressors. Show your analyses and state your conclusions.

**R Code:**

```
cor(gradeVar)

Xmat <- cbind(gradeVar$gender, gradeVar$assign1, gradeVar$assign2, gradeVar$assign3,
gradeVar$assign4, gradeVar$faculty)
cor(Xmat)

eigen(t(Xmat) %*% Xmat)

evals <- eigen(t(Xmat) %*% Xmat)$values

# condition number

max(evals)/min(evals)
```

**R Output:**

Correlation matrix:

```
              gender      assign1     assign2     assign3     assign4     faculty        exam
gender   1.000000000  0.003488757  0.09266716  0.08714460  0.00341535 -0.03039257  0.004098183
assign1  0.003488757  1.000000000  0.85372458  0.80873996  0.73282347 -0.04730448  0.689793211
assign2  0.092667159  0.853724583  1.00000000  0.85844687  0.76525813 -0.04606978  0.714912328
assign3  0.087144598  0.808739959  0.85844687  1.00000000  0.76684429 -0.04910358  0.752532106
assign4  0.003415350  0.732823469  0.76525813  0.76684429  1.00000000 -0.02485790  0.910108291
faculty -0.030392573 -0.047304476 -0.04606978 -0.04910358 -0.02485790  1.00000000 -0.079331009
exam     0.004098183  0.689793211  0.71491233  0.75253211  0.91010829 -0.07933101  1.000000000
```

Eigen Decomposition:

```
eigen() decomposition
$values
[1] 5.387794e+06 1.042444e+05 8.755183e+04 5.636046e+04 4.662468e+02 9.374216e+01

$vectors
            [,1]         [,2]         [,3]          [,4]         [,5]         [,6]
[1,] -0.01077290  0.008579272 -0.007948891 -0.0030056839 -0.709066924  0.704955473
[2,] -0.53608393  0.488032929 -0.487681859  0.4859916524  0.020216141  0.002775579
[3,] -0.54504913  0.247589583  0.050795211 -0.7993197445  0.010867026 -0.003247276
[4,] -0.49461107 -0.109048298  0.786470390  0.3533950642 -0.006306436 -0.002199804
[5,] -0.41313015 -0.829765272 -0.375232207  0.0008725357  0.002763436  0.002337136
[6,] -0.01013763  0.006907756 -0.013717961  0.0014808794 -0.704733925 -0.709231437
```

Condition Number:

```
[1] 57474.6
```

**Discussion:**

The correlation matrix above shows that there is a lot of evidence of multi-collinearity among the regressors. The multi-collinearity seems largely isolated to the numerical variables assign1, assign2, assign3 and assign4.

Assign1 and assign2 have a strong linear association with a correlation coefficient of 0.85372458. Assign1 and assign3 have a strong association with a correlation coefficient of 0.80873996. Assign 2 and assign3 also have strong associations with a correlation coefficient of 0.85844687. As all of these values are above the rule of thumb of 0.8, it can be concluded that there is significant multi-collinearity among assign1, assign 2 and assign 3. The exception among the assign variables is assign4, which does not reach the rule of thumb of 0.8, but approach it. It is thus probably fair to conclude that all the assign variables show signs of multi-collinearity.

The condition number is also somewhat high, suggesting multi-collinearity. There are greater examples of such a value however.