HomeWork_2_LexBrunett_STAT408

Lex Miguel Brunett Chavez

2023-09-29

STAT408 HOMEWORK 2 Due by 9/29/2023 (50 Points)

Please provide detailed calculation and explanation in your solution. Points will be deducted for skimpily written answers. This homework will also require coding in R. On the coding part, the homework solutions should also include detailed description, R code, and output. Write your answers, scan them, and combine to a single pdf file. Name this file as yourname_hw2 and upload to Sakai.

1. (10 points). Consider a simple linear regression model y = $\beta0$ + $\beta1x$ + ε. We fit this model based on a dataset with test score (y) and training hours (x). The fitted model is y = 10 + 0.56x.

a. What is the fitted value of the response variable corresponding to x = 7?

To answer this question, we need to replace x in the model, giving the next value:

$$y = 10 + 0.56(7)$$

$$y = 10 + 3.92$$

y = 13.92 score with 7 training hours

b. What is the residual corresponding to the data point with x = 7 and y = 17?

The residual value corresponds to the fitted value vs the response, giving:

Residual = y - ŷ

Residual = 17 - 13.92

Residual = 17 - 13.92

Residual = 3.08

The residual score between 17 and 13.92 gives 3.08 of score.

c. If the number of training hours is increased by 1, how is the expected test score affected?

Based on the equation for the model with the training hours increased by 1 gives:

$$y = 10 + (0.56+1) *X$$

$$y = 10 + 0.56*X+1*X$$

Comparing the models with the example given in an example:

$$y = 10 + 0.56*(7) +1*(7)$$

$$y = 10 + 0.56*(7) +7$$

$$y = 10 + 3.92 +7$$

$$y = 20.92$$

The score by 7 of the score, so increasing the training hours will increase the expected score.

d. Consider the data point in part b. An additional test score is to be obtained for a new observation at x = 7. Would the test score for the new observation necessarily be 17? Explain.

Using x = 7 would not necessarily be 17. The regression model does not guarantee that the observed value will match the predicted value exactly.

In part b, we calculated the residual with x = 7 and y = 17, and it was approximately 3.08. This means that the observed test score for that data point was 3.08 units higher than what the regression model predicted based on the number of training hours.

For a new observation at x = 7, the predicted value using the training hours changed to 1, the model fitted the value to 17, the predicted value for this exercise.

2. (10 points) In this question, we will still use the teengamb dataset. It concerns a study of teenage gambling in Britain. Each row is one teenager's records. Download this dataset from Sakai and read it into R. Below is the variable description:

a. Fit a regression model with the expenditure on gambling as the response and the sex, status, income and verbal score as predictors. Save the model output to a "model" object. Use the summary function to show the model output.

b. What percentage of variation in the response is explained by these predictors?

c. Use model$residuals to show the residuals. Which observation has the largest (positive) residual?

d. Use model$fitted.values to show the fitted response. Compute the correlation of the residuals with the fitted response.

e. Compute the correlation of the residuals with the income.

f. If all other predictors held constant, what would be the difference in predicted expenditure on gambling for a male compared to a female?

```
teengamb_df <- suppressWarnings(read.csv("https://raw.githubusercontent.com/L
exoBrunett/STAT408_LexoBrunett/main/Datasets/teengamb.csv"))

# a fit the model

# Fitting the linear model of gambling based on the predictors, status, sex,
income and verbal
model <- lm(gamble~status+sex+income+verbal, data= teengamb_df)

# Using the summary to observe the results
summary(model)

##
## Call:
## lm(formula = gamble ~ status + sex + income + verbal, data = teengamb_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.082 -11.320  -1.451   9.452  94.252
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.55565   17.19680   1.312   0.1968
## status        0.05223    0.28111   0.186   0.8535
## sex         -22.11833    8.21111  -2.694   0.0101 *
## income        4.96198    1.02539   4.839 1.79e-05 ***
## verbal       -2.95949    2.17215  -1.362   0.1803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06

# b What percentage of variation in the response is explained by these predic
tors? using the R-squared value as the variance

summary(model)

##
## Call:
## lm(formula = gamble ~ status + sex + income + verbal, data = teengamb_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.082 -11.320  -1.451   9.452  94.252 
## 
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)    
## (Intercept)  22.55565   17.19680   1.312   0.1968    
## status        0.05223    0.28111   0.186   0.8535    
## sex         -22.11833    8.21111  -2.694   0.0101 *  
## income        4.96198    1.02539   4.839 1.79e-05 ***
## verbal       -2.95949    2.17215  -1.362   0.1803    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816 
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06
```

```
# c observation with the largest residual

maximun_residual <- max(model$residuals)
print(maximun_residual)
```

```
## [1] 94.25222
```

```
max_index_residual <- which.max(model$residuals)
print(max_index_residual)
```

```
## 24 
## 24
```

```
# d Compute the correlation of the residuals with the fitted response.

fitted_values <- model$fitted.values

correlation_fittedvalues <- cor(model$residuals, model$fitted.values)
print(correlation_fittedvalues)
```

```
## [1] -7.609513e-17
```

```
# e Compute the correlation of the residuals with the income.

correlation_income <- cor(model$residuals, teengamb_df$income)
print(correlation_income)
```

```
## [1] 2.319757e-17
```

```
# f what would be the difference in predicted expenditure on gambling for a m
ale compared to a female?, lets use the coefficient of sex

summary(model)
```

```
## 
## Call:
```

```
## lm(formula = gamble ~ status + sex + income + verbal, data = teengamb_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -51.082 -11.320  -1.451   9.452  94.252
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  22.55565   17.19680   1.312   0.1968
## status        0.05223    0.28111   0.186   0.8535
## sex         -22.11833    8.21111  -2.694   0.0101 *
## income        4.96198    1.02539   4.839 1.79e-05 ***
## verbal       -2.95949    2.17215  -1.362   0.1803
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 22.69 on 42 degrees of freedom
## Multiple R-squared:  0.5267, Adjusted R-squared:  0.4816
## F-statistic: 11.69 on 4 and 42 DF,  p-value: 1.815e-06
```

Basing on the R-squared value from the summary of the model fitted, we see a value of 0.5267 meaning a variance of 52.67% with the gamble variable explained by the predictors used in the dataset.

The index with the highest positive residual is the case number 24 with a value of 94.2522174

The correlation of the residuals with the fitted values obtained from the model is -7.6095127^{-17}.

The correlation of the residuals with the income values of the dataframe is 2.3197573^{-17}.

Using the summary function of the model fitted we can see the coefficient between sex variable from the dataset (0 to male and 1 to female) giving a difference for female gamblers expending $22.11833 less than male gamblers.

3. (10 points) The dataset prostate comes from a study on 97 men with prostate cancer who were due to receive a radical prostatectomy. The description of each variable can be found at https://rafalab.github.io/pages/649/prostate.html. Download and import this dataset from Sakai, answer following questions.

a. Fit a regression model with lpsa as the response and lcavol as the predictor. Show the residual sum of square (RSS) and the R^2 of this model (hint: check deviance function for RSS).

b. Add lweight, svi, lbph, age, lcp, pgg45 and gleason as predictors to the regression model. Show the residual sum of square (RSS) and the R^2 of this model.

c. Compare the RSS and R^2 of these two models. Explain why you observe such a comparison result.

d. Use the method introduced in lecture slides to manually fit the model in b. First construct a design matrix X, then a response vector y, and finally use the formula of parameter estimation. Compare the manually estimated parameters with the result from the lm function.

```
prostate_df <- suppressWarnings(read.csv("https://raw.githubusercontent.com/L
exoBrunett/STAT408_LexoBrunett/main/Datasets/prostate.csv"))

# a Fit a regression model with lpsa as the response and lcavol as the predic
tor.Show the residual sum of square (RSS) and the R^2 of this model

linear_model_obj <- lm(lpsa ~ lcavol, data = prostate_df)
summary(linear_model_obj)

##
## Call:
## lm(formula = lpsa ~ lcavol, data = prostate_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.67625 -0.41648  0.09859  0.50709  1.89673
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.50730    0.12194   12.36   <2e-16 ***
## lcavol       0.71932    0.06819   10.55   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7875 on 95 degrees of freedom
## Multiple R-squared:  0.5394, Adjusted R-squared:  0.5346
## F-statistic: 111.3 on 1 and 95 DF,  p-value: < 2.2e-16

# obtain the RSS from deviance function
deviance(linear_model_obj)

## [1] 58.91476

anova(linear_model_obj)

## Analysis of Variance Table
##
## Response: lpsa
##           Df Sum Sq Mean Sq F value    Pr(>F)
## lcavol     1 69.003  69.003  111.27 < 2.2e-16 ***
## Residuals 95 58.915   0.620
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
# obtaing the r squared
print(paste0("R-squared: ", summary(linear_model_obj)$r.squared))
```

```
## [1] "R-squared: 0.53943190877902"
```

```r
# making the new fitted model with new predictors
prostate_lm <- lm(lpsa ~ lcavol + lweight + svi + lbph + age + lcp + pgg45 +
gleason, data = prostate_df)

summary(prostate_lm)
```

```
##
## Call:
## lm(formula = lpsa ~ lcavol + lweight + svi + lbph + age + lcp +
##     pgg45 + gleason, data = prostate_df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -1.7331 -0.3713 -0.0170  0.4141  1.6381
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.669337   1.296387   0.516  0.60693
## lcavol       0.587022   0.087920   6.677 2.11e-09 ***
## lweight      0.454467   0.170012   2.673  0.00896 **
## svi          0.766157   0.244309   3.136  0.00233 **
## lbph         0.107054   0.058449   1.832  0.07040 .
## age         -0.019637   0.011173  -1.758  0.08229 .
## lcp         -0.105474   0.091013  -1.159  0.24964
## pgg45        0.004525   0.004421   1.024  0.30886
## gleason      0.045142   0.157465   0.287  0.77503
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7084 on 88 degrees of freedom
## Multiple R-squared:  0.6548, Adjusted R-squared:  0.6234
## F-statistic: 20.86 on 8 and 88 DF,  p-value: < 2.2e-16
```

```r
# obtain the RSS from deviance function
deviance(prostate_lm)
```

```
## [1] 44.16302
```

```r
# sum((new_linear_model_obj$residuals) ^ 2)
anova(prostate_lm)
```

```
## Analysis of Variance Table
##
## Response: lpsa
##           Df Sum Sq Mean Sq  F value    Pr(>F)
## lcavol     1 69.003  69.003 137.4962 < 2.2e-16 ***
## lweight    1  5.949   5.949  11.8531 0.0008832 ***
## svi        1  5.181   5.181  10.3245 0.0018350 **
## lbph       1  1.300   1.300   2.5905 0.1110872
## age        1  0.959   0.959   1.9114 0.1703058
## lcp        1  0.129   0.129   0.2576 0.6130533
## pgg45      1  1.192   1.192   2.3752 0.1268606
## gleason    1  0.041   0.041   0.0822 0.7750328
## Residuals 88 44.163   0.502
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

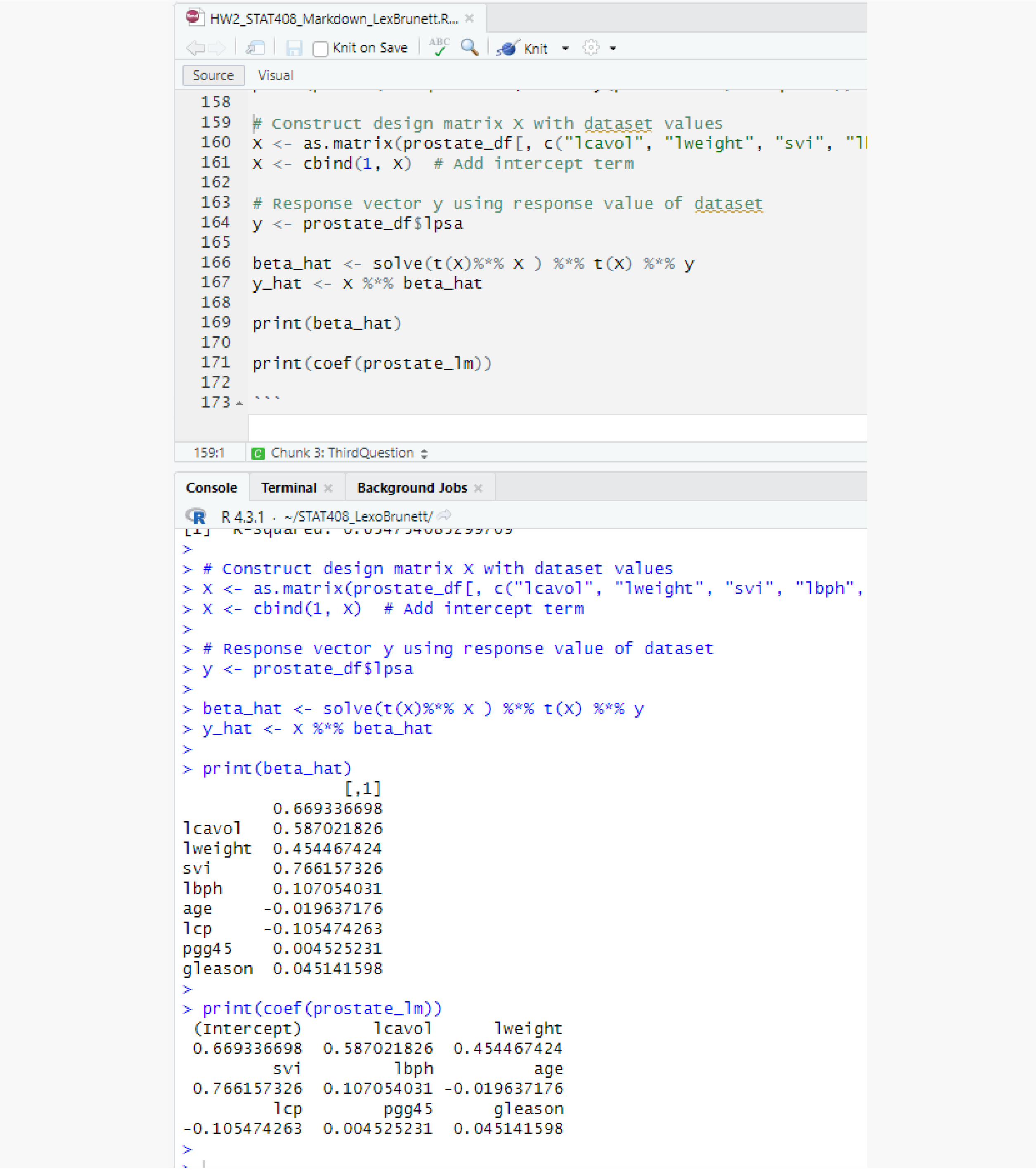
```r
# obtaing the r squared
print(paste0("R-squared: ", summary(prostate_lm)$r.squared))
```

```
## [1] "R-squared: 0.654754085299709"
```

```r
# Construct design matrix X with dataset values

X <- as.matrix(prostate_df[, c("lcavol", "lweight", "svi", "lbph", "age", "lc
p", "pgg45", "gleason")])

X <- cbind(1, X)  # Add intercept term



# Response vector y using response value of dataset

y <- prostate_df$lpsa
```

$$\hat{y} = X\hat{\beta} = X(X^TX)^{-1}X^Ty$$

```r
beta_hat <- solve(t(X)%*% X ) %*% t(X) %*% y

y_hat <- X %*% beta_hat


print(beta_hat)

print(coef(prostate_lm))
```



```r

# Construct design matrix X with dataset values
X <- as.matrix(prostate_df[, c("lcavol", "lweight", "svi", "l
X <- cbind(1, X)  # Add intercept term

# Response vector y using response value of dataset
y <- prostate_df$lpsa

beta_hat <- solve(t(X)%*% X ) %*% t(X) %*% y
y_hat <- X %*% beta_hat

print(beta_hat)

print(coef(prostate_lm))

```
```

```
[1] R-squared: 0.654754085299709
>
> # Construct design matrix X with dataset values
> X <- as.matrix(prostate_df[, c("lcavol", "lweight", "svi", "lbph",
> X <- cbind(1, X)  # Add intercept term
>
> # Response vector y using response value of dataset
> y <- prostate_df$lpsa
>
> beta_hat <- solve(t(X)%*% X ) %*% t(X) %*% y
> y_hat <- X %*% beta_hat
>
> print(beta_hat)
                [,1]
         0.669336698
lcavol   0.587021826
lweight  0.454467424
svi      0.766157326
lbph     0.107054031
age     -0.019637176
lcp     -0.105474263
pgg45    0.004525231
gleason  0.045141598
>
> print(coef(prostate_lm))
 (Intercept)       lcavol      lweight 
 0.669336698  0.587021826  0.454467424 
         svi         lbph          age 
 0.766157326  0.107054031 -0.019637176 
         lcp        pgg45      gleason 
-0.105474263  0.004525231  0.045141598 
>
```

For the regression model with the lpsa predictor the RSS value is 58.91476 the R-squared for this model 0.53943190877902.

In the regression model adding more predictors change the RSS value to 44.16302 and the R-squared value for this model give 0.654754085299709.

Comparing the model with the RSS and R-squared we can se that the model with more predictors is better accurate for the model based in the R-squared (higher R-squared) and
```

with the RSS we can conclude that the second model fits better the data with a minimal data variation, meaning that the with these two statistics values the model with more predictors is better.

In the exercise, we create the matrix X and calcjulate with the formula given in class, the manual and the lm function values are exact the same, so we can asure that using the lm function with the coefficient method we can obtain the same result than the theorical formula given in class.

4. (10 points) Thirty samples of cheddar cheese were analyzed for their content of acetic acid, hydrogen sulfide and lactic acid. Each sample was tasted and scored by a panel of judges and the average taste score produced. Use the cheddar dataset from Sakai to answer the following questions.

a. Fit a regression model with taste as the response and the three chemical contents as predictors. Report the values of the regression coefficients.

b. Compute the correlation between the fitted values and the true response. What information can you learn from this correlation?

c. How do you interpret the value of intercept in this model? Does this value make sense in this setting (tasting cheese)?

```
cheddarCheese_df <- read.csv("https://raw.githubusercontent.com/LexoBrunett/S
TAT408_LexoBrunett/main/Datasets/cheddar.csv")

cheddar_model <- lm(taste ~ Acetic + H2S + Lactic, data = cheddarCheese_df)

coefficients(cheddar_model)
```

```
(Intercept)      Acetic         H2S      Lactic 
-28.8767696   0.3277413   3.9118411  19.6705434
```

```
summary(cheddar_model)
```

```
## 
## Call:
## lm(formula = taste ~ Acetic + H2S + Lactic, data = cheddarCheese_df)
## 
## Residuals:
##     Min      1Q  Median      3Q     Max 
## -17.390  -6.612  -1.009   4.908  25.449 
## 
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -28.8768    19.7354  -1.463  0.15540   
## Acetic        0.3277     4.4598   0.073  0.94198   
## H2S           3.9118     1.2484   3.133  0.00425 **
## Lactic       19.6705     8.6291   2.280  0.03108 * 
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 10.13 on 26 degrees of freedom
## Multiple R-squared:  0.6518, Adjusted R-squared:  0.6116
## F-statistic: 16.22 on 3 and 26 DF,  p-value: 3.81e-06

predicted_values_cheddar <- cheddar_model$fitted.values

correlation_fittedvalues_cheddar <- cor(cheddarCheese_df$taste, predicted_val
ues_cheddar)

print(correlation_fittedvalues_cheddar)

[1] 0.8073256
```

Using the lm function in R with the predictors the coefficients report is showed with the summary function or the coefficients function, in this case we use both to show that they show the same.

```
(Intercept)      Acetic         H2S      Lactic
-28.8767696   0.3277413   3.9118411  19.6705434
```

Using the model fitted values and comparing with the taste values we can obtain the correlation using the cor function, giving the correlation equal to 0.8073256.

The intercept value in a regression model gives the value when predictors are zero, if we think about how the taste of a cheddar cheese with zero quantity of Acetic acid is, H2S and lactic acid, making it to think that the cheese will have no tase.

5. (10 points) Run the following R code, generated x and y, fit the following two linear models and explain:

lm1 <- lm(y~x) lm2 <- lm(y~x+I(x^2))

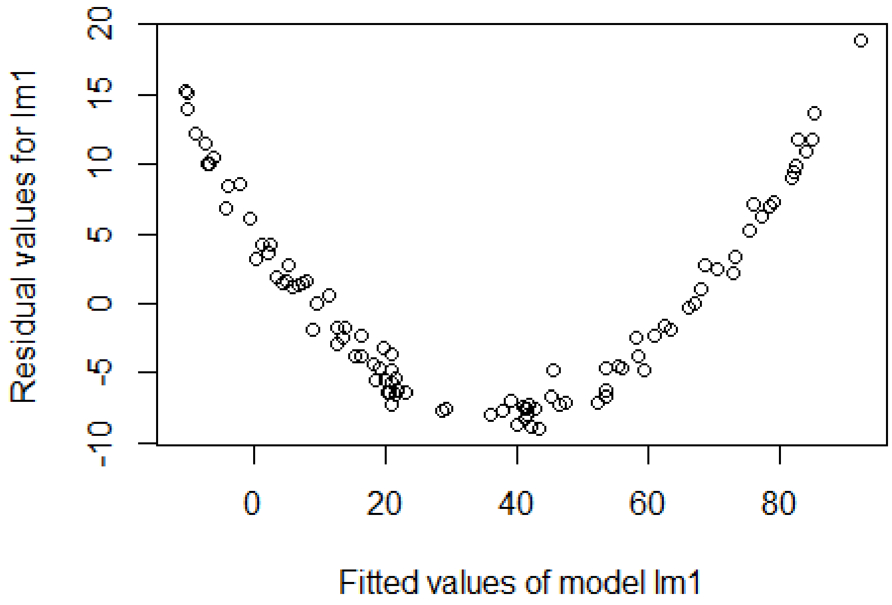
a. Explain what the code does. Use ?function_name() or Google if you do not know the meaning of any function.

b. For both models, plot the residual versus the fitted response. Describe the pattern you observed in the plots.

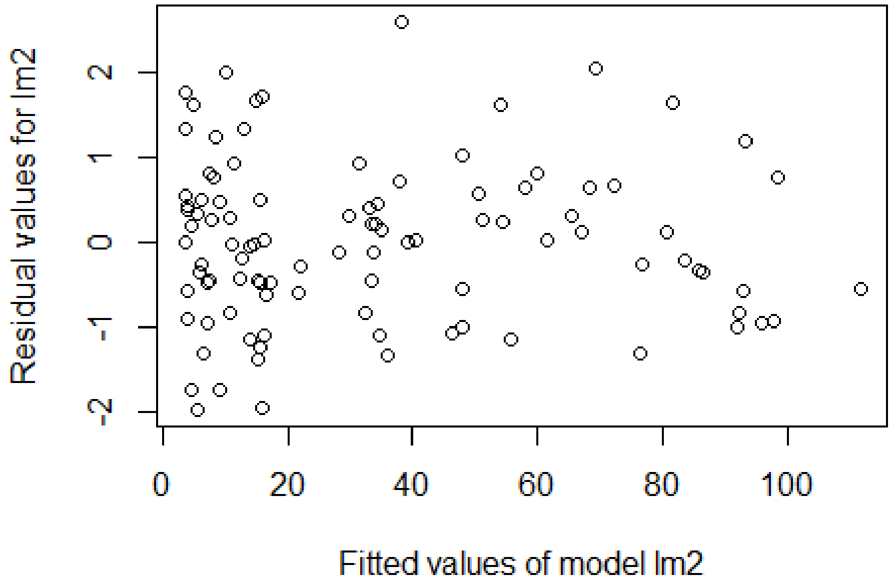
c. Which model is better? Give your reason.

```
# Run the following R code:
set.seed(1234)
x <- runif(100,0,10)
y <- 3+x+x^2+rnorm(100,0,1)

#print(x)
#print(y)

# Once you have generated x and y, fit the following two linear models:
```

```r
lm1 <- lm(y~x)
lm2 <- lm(y~x+I(x^2))

print(lm1)

##
## Call:
## lm(formula = y ~ x)
##
## Coefficients:
## (Intercept)            x
##      -11.45        10.45

print(lm2)

##
## Call:
## lm(formula = y ~ x + I(x^2))
##
## Coefficients:
## (Intercept)            x       I(x^2)
##      3.3280       0.8468       1.0157

# plot the fitted values with residuals
plot(lm1$fitted.values, lm1$residuals, xlab = "Fitted values of model lm1", ylab = "Residual values for lm1")
```



```
plot(lm2$fitted.values, lm2$residuals, xlab = "Fitted values of model lm2", y
lab = "Residual values for lm2")
```



```
r_squared_lm1 <- summary(lm1)$r.squared
r_squared_lm2 <- summary(lm2)$r.squared

print(r_squared_lm1)

## [1] 0.9447891

print(r_squared_lm2)

## [1] 0.9990073
```

The first line stablish a seed for random numbers, using this function asures reproductibility with the results (create a population for the exercise),For the second line the function runif helps to create a vector of one hundred random numbers with a uniform distributed, The third line makes a vector with the cuadratic equation plus a vector of 100 random number which follows a normal distribution and the fourth line makes two fitted models lm1 and lm2 where the second model uses a indepent value.

Looking the plots comparing the fitted values and the residuals values we can se that for the lm1 model there is a curve formed compared to the lm2 model, this first evidence is not a good sign to say that the lm1 model is better, so we create a variable to save the R squared to conclude which model is better and lm1 R squared value is 0.9447891 compared to the lm2 squared value 0.9990073, so to conclude lm2 is a better model, this is because we use an extra predictor, more predictors better model, but too many predictor will give noise to the plots.