
#title: "Assignment 3"

#output: pdf_document

#author: Colm Mooney 20325583

---

1. Which of the following regression models are linear? Give reasons.

(a) $y = \beta_0 + \beta_1 x + \beta_2 x^2 + \beta_3 x^3 + \epsilon$

(b) $y = \beta_0 + \beta_1 10^x + \epsilon$

(c) $y = (\beta_0 + \beta_1 x)/(\beta_0 + \beta_2 x) + \epsilon$

(d) $y = \exp(\beta_0 + \beta_1 x + \epsilon)$

(e) $y = \exp(\beta_0 + \beta_1 x) + \epsilon$

Linear regression is a method that shows the relationship between a dependent variable & one or more independent variables. It only models the relationship between linear variables, and it's not suited to some problems. As a predictor variable increases, the response either increase/decreases at the same rate. If this relationship holds the same for any values of variables, a straight-line pattern will form when graphed.

The X values are the independent variables. E stands for Random Error.

#(a) A clean and simple linear regression model. Probably the most plain to see from the examples, think of this like A + B(var1) + C(var2) + D(var3) + e.

#(b) This is a linear regression model. The value of y can be drastically different depending on the value of x, this is since the independent variable is being used to the power of.

#(c) This is not a linear regression model. While I don't like the fact the variables are being divided, with it could still very much have linear regression. But the Constant is there twice and the Random error is separate from the model.

#(d) If log is applied to this, regression model It becomes linear.

#(e) This is not a linear regression model. It is similar to part c, in this case, e is not within the exp() brackets. If it was, it would be identical to (d) and thus be linear.

2. Suppose you have the following data

i	y_i	x_{i1}	x_{i2}
1	62	2	6
2	60	9	10
3	57	6	4
4	48	3	13
5	23	5	2

and want to fit the model $y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \epsilon_i$.

(a) Write down the model in matrix format and specify what each matrix / vector is.

(b) Calculate the least squares estimates using matrix manipulations.

(c) Calculate the fitted values and residuals.

(d) What is the estimate of σ^2?

```
> x2 <- matrix(c(1,1,1,1,1,2,9,6,3,5,6,10,4,13,2), nrow=5)
> x2
     [,1] [,2] [,3]
[1,]    1    2    6
[2,]    1    9   10
[3,]    1    6    4
[4,]    1    3   13
[5,]    1    5    2
> y2 <- c(62,60,57,48,23)
> y2
[1] 62 60 57 48 23
```

Code Used:

```
#Q2

#(a)

x2 <- matrix(c(1,1,1,1,1,2,9,6,3,5,6,10,4,13,2), nrow=5)

x2

y2 <- c(62,60,57,48,23)

y2
```

```
#(b)

xtx2 <- crossprod(x2)

xtxi2 <- solve(xtx2)

round(xtxi2,3)

xty2 <- crossprod(x2,y2)


betahat2 <- xtxi2 %*% xty2

betahat2


yhat2 <- x2 %*% betahat2

yhat2

ehat2 <- y2 - yhat2

ehat2

n2 <- nrow(x2)

p2 <- ncol(x2)

MSE2 <- crossprod(ehat2)/(n2-p2)

MSE2

#-------------

H2 <- x2 %*% xtxi2 %*% t(x2)

SSE2 <- t(y2) %*% (diag(nrow(x2)))

sigmahatTest2 <- SSE2/(n2-p2)

sigmahatTest2


#(c)

summary(lm(y2 ~ x2))

ols_data <- data.frame(x2,y2)

ols_data


#(c) Calculate fitted values & residuals.

#The residuals are: 15.0 3.5 11.0 -10.0 -19.5
```

dtf2 <- data.frame(y=y2, x2[,-1])

fit2 <- lm(y ~., data = dtf2) # Estimate linear regression model

summary(fit2) # Summary of linear regression model

#Fitted Values:

fits <- fitted(fit2)

head(fits)

(a) The first column is B0 which is always 1. The second column are the xi1 values and the third column is the xi2 values.

```
> #(a)
> x2 <- matrix(c(1,1,1,1,1,2,9,6,3,5,6,10,4,13,2), nrow=5)
> x2
     [,1] [,2] [,3]
[1,]    1    2    6
[2,]    1    9   10
[3,]    1    6    4
[4,]    1    3   13
[5,]    1    5    2
> y2 <- c(62,60,57,48,23)
> y2
[1] 62 60 57 48 23
```

(b)

```
> xtx2 <- crossprod(x2)
> xtxi2 <- solve(xtx2)
> #(a)
> x2 <- matrix(c(1,1,1,1,1,2,9,6,3,5,6,10,4,13,2), nrow=5)
> x2
     [,1] [,2] [,3]
[1,]    1    2    6
[2,]    1    9   10
[3,]    1    6    4
[4,]    1    3   13
[5,]    1    5    2
> y2 <- c(62,60,57,48,23)
> y2
[1] 62 60 57 48 23
> #(b)
> xtx2 <- crossprod(x2)
> xtxi2 <- solve(xtx2)
> round(xtxi2,3)
       [,1]   [,2]   [,3]
[1,]  1.646 -0.167 -0.088
[2,] -0.167  0.033  0.000
[3,] -0.088  0.000  0.013
> xty2 <- crossprod(x2,y2)
> betahat2 <- xtxi2 %*% xty2
> betahat2
     [,1]
[1,] 37.0
[2,]  0.5
[3,]  1.5
> yhat2 <- x2 %*% betahat2
> yhat2
     [,1]
[1,] 47.0
[2,] 56.5
[3,] 46.0
[4,] 58.0
[5,] 42.5
> ehat2 <- y2 - yhat2
> ehat2
      [,1]
[1,]  15.0
[2,]   3.5
[3,]  11.0
[4,] -10.0
[5,] -19.5
> 
```

```
> n2 <- nrow(x2)
> p2 <- ncol(x2)
> MSE2 <- crossprod(ehat2)/(n2-p2)
> MSE2
      [,1]
[1,] 419.25
> #-------------
> H2 <- x2 %*% xtxi2 %*% t(x2)
> SSE2 <- t(y2) %*% (diag(nrow(x2)))
> sigmahatTest2 <- SSE2/(n2-p2)
> sigmahatTest2
     [,1] [,2] [,3] [,4] [,5]
[1,]   31   30 28.5   24 11.5
> 
```

(c)

---

summary(lm(y2 ~ x2))

ols_data <- data.frame(x2,y2)

ols_data

#(c) & (d)

#The residuals are: 15.0 3.5 11.0 -10.0 -19.5

```
> summary(lm(y2 ~ x2))

Call:
lm(formula = y2 ~ x2)

Residuals:
    1     2     3     4     5 
 15.0   3.5  11.0 -10.0 -19.5 

Coefficients: (1 not defined because of singularities)
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   37.000     26.268   1.409    0.294
x21               NA         NA      NA       NA
x22            0.500      3.738   0.134    0.906
x23            1.500      2.289   0.655    0.580

Residual standard error: 20.48 on 2 degrees of freedom
Multiple R-squared:  0.1827,	Adjusted R-squared:  -0.6345 
F-statistic: 0.2236 on 2 and 2 DF,  p-value: 0.8173

> ols_data <- data.frame(x2,y2)
> ols_data
  X1 X2 X3 y2
1  1  2  6 62
2  1  9 10 60
3  1  6  4 57
4  1  3 13 48
5  1  5  2 23
> dtf2 <- data.frame(y=y2, x2[,-1])
```

```
> dtf2 <- data.frame(y=y2, x2[,-1])
> dtf2 <- data.frame(y=y2, x2[,-1])
> fit2 <- lm(y ~., data = dtf2) # Estimate linear regression model
> summary(fit2) # Summary of linear regression model

Call:
lm(formula = y ~ ., data = dtf2)

Residuals:
    1     2     3     4     5 
 15.0   3.5  11.0 -10.0 -19.5 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   37.000     26.268   1.409    0.294
X1             0.500      3.738   0.134    0.906
X2             1.500      2.289   0.655    0.580

Residual standard error: 20.48 on 2 degrees of freedom
Multiple R-squared:  0.1827,	Adjusted R-squared:  -0.6345 
F-statistic: 0.2236 on 2 and 2 DF,  p-value: 0.8173

> #Fitted Values:
> fits <- fitted(fit2)
> head(fits)
   1    2    3    4    5 
47.0 56.5 46.0 58.0 42.5 
> 
```

#While Question 3 code is available in (All used code), we are going to skip it as it isn't part of the submission.

Code used:

#Q4

#a

#i

library(tidyverse)

library(dplyr)

bodyfat <- read.table("SharedFiles/ST303/data/bodyfat.txt",header = TRUE)

head(bodyfat)

pairs(bodyfat)

fit <- lm(bfat~., data = bodyfat)

summary(fit)

```
#Positive impact:Age,Abdomen, Knee,Ankle

#Negative impact on a negative body fat: Weight, Height, Neck

#Significant Variables: Weight, Age, Ankle, Knee.

#Insignificant Variables: Abdomen, Height




#ii.

#plot(bodyfat$bfat, bodyfat$age)

#t.test(x = bodyfat$bfat, y = bodyfat$age, conf.level = 0.95)

FatAge <- lm(bfat ~ age, data = bodyfat)

confint(FatAge,'age')



#iii


fitbit <- lm(bfat~., data = bodyfat)

summary(fitbit)

confint(fitbit)


TomData <- data.frame(age = 49,weight = 188,height= 68,neck = 37,abdomen = 90,knee= 38,ankle=
24)

predict(fitbit , TomData, interval="confidence")


BillData <- data.frame(age = 40,weight = 220,height= 76,neck = 40,abdomen = 113,knee= 34,ankle=
20)

predict(fitbit, BillData, interval = "confidence")

#Tom's data is predicted more precisely. This is because his collective values are not as big as Bills.

#We can see this in action when the predict is used, Tom's lwr and upr fit are closer together than
Bills.


#(b)
```

```
#i

fit4 <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)

summary(fit4)

#Tom's Data is the more precise between the two. This is because the interval is narrower.

#Positve impact: Age,Abdomen,Knee

#Negative impact: Height, Neck

#Significant Variables: Knee

#Insignificant Variables: Age, Height, Neck, Abdomen

#While the impacts are the same, the amount of significant Variables went drastically down leaving only Knee left.

#They of course, became insignificant variables.


#ii

FatAge <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)

confint(FatAge,'age')

#Yes these finding are completely different.


#iii

TomData <- data.frame(age = 49,height= 68,neck = 37,abdomen = 90,knee= 38)

predict(fit4 , TomData, interval="confidence")


BillData <- data.frame(age = 40,height= 76,neck = 40,abdomen = 113,knee= 34)

predict(fit4, BillData, interval = "confidence")
```

(a) (i)

```
library(tidyverse)

library(dplyr)

bodyfat <- read.table("SharedFiles/ST303/data/bodyfat.txt",header = TRUE)

head(bodyfat)

fit <- lm(bfat~., data = bodyfat)

summary(fit)
```

```
> fit <- lm(bfat~., data = bodyfat)
> summary(fit)

Call:
lm(formula = bfat ~ ., data = bodyfat)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.1016 -2.9148 -0.7392  3.3351  8.4197 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) -5.63842   23.98381  -0.235  0.81474    
age          0.04397    0.03812   1.154  0.25212    
weight      -0.07294    0.06756  -1.080  0.28356    
height      -0.69440    0.26036  -2.667  0.00926 ** 
neck        -0.56477    0.33051  -1.709  0.09137 .  
abdomen      0.86684    0.12923   6.708 2.57e-09 ***
knee         0.47829    0.36361   1.315  0.19214    
ankle        0.31149    0.25267   1.233  0.22126    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.188 on 80 degrees of freedom
Multiple R-squared:  0.8097,	Adjusted R-squared:  0.7931 
F-statistic: 48.64 on 7 and 80 DF,  p-value: < 2.2e-16
```

#Positive impact:Age,Abdomen, Knee,Ankle

#Negative impact on a negative body fat: Weight, Height, Neck

#Significant Variables: Weight, Age, Ankle, Knee.

#Insignificant Variables: Abdomen, Height

(ii)

#plot(bodyfat$bfat, bodyfat$age)

#t.test(x = bodyfat$bfat, y = bodyfat$age, conf.level = 0.95)

FatAge <- lm(bfat ~ age, data = bodyfat)

confint(FatAge,'age')

```
> #ii.
> #plot(bodyfat$bfat, bodyfat$age)
> #t.test(x = bodyfat$bfat, y = bodyfat$age, conf.level = 0.95)
> FatAge <- lm(bfat ~ age, data = bodyfat)
> confint(FatAge,'age')
        2.5 %    97.5 %
age 0.1079472 0.3443517
> 
```

(iii)

#iii

fitbit <- lm(bfat~., data = bodyfat)

summary(fitbit)

confint(fitbit)

TomData <- data.frame(age = 49,weight = 188,height= 68,neck = 37,abdomen = 90,knee= 38,ankle= 24)

predict(fitbit , TomData, interval="confidence")

BillData <- data.frame(age = 40,weight = 220,height= 76,neck = 40,abdomen = 113,knee= 34,ankle= 20)

predict(fitbit, BillData, interval = "confidence")

#Tom's data is predicted more precisely. This is because his collective values are not as big as Bills.

#We can see this in action when the predict is used, Tom's lwr and upr fit are closer together than Bills.

(b)

(i)

```
> fit4 <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)
> summary(fit4)

Call:
lm(formula = bfat ~ age + height + neck + abdomen + knee, data = bodyfat)

Residuals:
    Min      1Q  Median      3Q     Max 
-7.7629 -3.2343 -0.8899  3.4161  8.2253 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 14.88456   13.77162   1.081 0.282948    
age          0.06437    0.03213   2.003 0.048444 *  
height      -0.82698    0.20481  -4.038 0.000121 ***
neck        -0.71606    0.28407  -2.521 0.013650 *  
abdomen      0.75235    0.07481  10.057 5.77e-16 ***
knee         0.43745    0.34144   1.281 0.203740    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 4.191 on 82 degrees of freedom
Multiple R-squared:  0.8047,	Adjusted R-squared:  0.7928 
F-statistic: 67.57 on 5 and 82 DF,  p-value: < 2.2e-16
```

fit4 <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)

summary(fit4)

#Tom's Data is the more precise between the two. This is because the interval is narrower.

#Positve impact: Age,Abdomen,Knee

#Negative impact: Height, Neck

#Significant Variables: Knee

#Insignificant Variables: Age, Height, Neck, Abdomen

#While the impacts are the same, the amount of significant Variables went drastically down leaving only Knee left.

#They of course, became insignificant variables.

(ii)

#A Comparison between bfat&age questions:

```
> #ii.
> #plot(bodyfat$bfat, bodyfat$age)
> #t.test(x = bodyfat$bfat, y = bodyfat$age, conf.level = 0.95)
> FatAge <- lm(bfat ~ age, data = bodyfat)
> confint(FatAge,'age')
        2.5 %    97.5 %
age 0.1079472 0.3443517
> #ii
> FatAge <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)
> confint(FatAge,'age')
           2.5 %    97.5 %
age 0.0004505887 0.1282917
>
```

(iii)

A comparison:

```
> TomData <- data.frame(age = 49,weight = 188,height= 68,neck = 37,abdomen = 90,knee= 38,ankle= 24)
> predict(fitbit , TomData, interval="confidence")
      fit     lwr     upr
1 18.3547 15.1175 21.5919
> BillData <- data.frame(age = 40,weight = 220,height= 76,neck = 40,abdomen = 113,knee= 34,ankle= 20)
> predict(fitbit, BillData, interval = "confidence")
       fit     lwr      upr
1 25.15361 18.2605 32.04672
> #iii
> TomData <- data.frame(age = 49,height= 68,neck = 37,abdomen = 90,knee= 38)
> predict(fit4 , TomData, interval="confidence")
       fit      lwr      upr
1 19.64509 18.51735 20.77283
> #Again, Tom's Data is more precise for the same reasons as last time.
> BillData <- data.frame(age = 40,height= 76,neck = 40,abdomen = 113,knee= 34)
> predict(fit4, BillData, interval = "confidence")
       fit      lwr     upr
1 25.85607 19.12253 32.5896
>
```

#Tom's data is predicted more precisely. This is because his collective values are not as big as Bills.

#We can see this in action when the predict is used, Tom's lwr and upr fit are closer together than Bills.

All code:

#title: "Assignment 3"

#output: pdf_document

#author: Colm Mooney 20325583

#Q1

#(a) This is a linear regression model.

#(b) This is a linear regression model.

#(c) This is not a linear regression model.

#(d) If log is applied, then this would look linear.

#(e) This is not a linear regression model.

```
#Q2

#(a)
x2 <- matrix(c(1,1,1,1,1,2,9,6,3,5,6,10,4,13,2), nrow=5)
x2
y2 <- c(62,60,57,48,23)
y2

#(b)
xtx2 <- crossprod(x2)
xtxi2 <- solve(xtx2)
round(xtxi2,3)
xty2 <- crossprod(x2,y2)

betahat2 <- xtxi2 %*% xty2
betahat2

yhat2 <- x2 %*% betahat2
yhat2
ehat2 <- y2 - yhat2
ehat2
n2 <- nrow(x2)
p2 <- ncol(x2)
MSE2 <- crossprod(ehat2)/(n2-p2)
MSE2
#-------------
H2 <- x2 %*% xtxi2 %*% t(x2)
SSE2 <- t(y2) %*% (diag(nrow(x2)))
sigmahatTest2 <- SSE2/(n2-p2)
sigmahatTest2
```

```
#(c)
summary(lm(y2 ~ x2))
ols_data <- data.frame(x2,y2)
ols_data

#(c) Calculate fitted values & residuals.
#The residuals are: 15.0 3.5 11.0 -10.0 -19.5

dtf2 <- data.frame(y=y2, x2[,-1])
fit2 <- lm(y ~., data = dtf2) # Estimate linear regression model
summary(fit2) # Summary of linear regression model
#Fitted Values:
fits <- fitted(fit2)
head(fits)



#Q3
#(a)
x <- matrix(c(1,1,1,1,1,1,1,2,3,1,2,3,1,4,9,1,4,9,0,0,0,1,1,1), nrow=6)
x
#B0 <- all 1s, #B1 All x1 values, #B2 <- x1^2 #B3<-(0,0,0,1,1,1)
y <- c(6.1,5.4,7.2,8.9,9.1,11.6)
y

#(b)
xtx <- crossprod(x)
xtxi <- solve(xtx)
#xti <- format(xtxi, scientific = FALSE)
round(xtxi,3)

xty <- crossprod(x,y)
```

```
#c

betahat <- xtxi %*% xty
betahat

yhat <- x %*% betahat
yhat
ehat <- y - yhat
ehat



n <- nrow(x)
p <- ncol(x)
MSE <- crossprod(ehat)/(n-p)
-----------
#Or - Hard version, cool to see, don't do tho
H <- x %*% xtxi %*% t(x)
SSE <- t(y) %*% (diag(nrow(x)))
sigmahat2 <- SSE/(n-p)
sigmahat2
-------------------------

#(d)

dtf <- data.frame(y=y, x[,-1])
fit1 <- lm(y ~., data = dtf) #~. is a formula consturct which R recognises
#as 'everything in the data argument except the response.
summary(fit1)

#I think this is the interval stuff, (e),(f),(g)
x0 <- rep(1,4) #Note this observation belongs to the design matrix already
```

```
yhat_0 <- x0 %*% betahat

yhat_0


SEfit_yhat <- sqrt(MSE) * sqrt(t(x0) %*% xtxi %*% x0)

SEfit_yhat


c(yhat_0 - qt(0.975, n-p) * SEfit_yhat,

  yhat_0 + qt(0.975, n-p) * SEfit_yhat)


SEpred_yhat <- sqrt(MSE) * sqrt(1 + t(x0) %*% xtxi %*% x0)

SEpred_yhat


c(yhat_0 - qt(0.975, n-p) * SEpred_yhat,

  yhat_0 + qt(0.975, n-p) * SEpred_yhat)


#To Verify these results

CI <- predict(fit1, newdata = data.frame(t(x0)), interval = c("confidence"), level = 0.95,
type="response")

PI <- predict(fit1, newdata = data.frame(data.frame(t(x0))), interval = c("prediction"), level =
0.95,type="response")


CI

PI


n <- nrow(dtf)

SST <- sum(dtf$y^2) - n * mean(dtf$y)^2

#alternatively

#SST <- sum(anova(fit1)[,2])

#or

#SST <- crossprod(dtf$y) - (1/n) * t(dtf$y) %*% matrix(1, nrow=n,ncol=n) %*% dtf$y

R2a <- 1 - (MSE/(SST/(n-1)))

all.equal(summary(fit1)$adj.r.squared, as.numeric(R2a))
```

```
#a
#i

library(tidyverse)
library(dplyr)
bodyfat <- read.table("SharedFiles/ST303/data/bodyfat.txt",header = TRUE)
head(bodyfat)
pairs(bodyfat)
fit <- lm(bfat~., data = bodyfat)
summary(fit)
#Positive impact:Age,Abdomen, Knee,Ankle
#Negative impact on a negative body fat: Weight, Height, Neck
#Significant Variables: Weight, Age, Ankle, Knee.
#Insignificant Variables: Abdomen, Height


#ii.
#plot(bodyfat$bfat, bodyfat$age)
#t.test(x = bodyfat$bfat, y = bodyfat$age, conf.level = 0.95)
FatAge <- lm(bfat ~ age, data = bodyfat)
confint(FatAge,'age')


#iii

fitbit <- lm(bfat~., data = bodyfat)
summary(fitbit)
confint(fitbit)
```

TomData <- data.frame(age = 49,weight = 188,height= 68,neck = 37,abdomen = 90,knee= 38,ankle= 24)

predict(fitbit , TomData, interval="confidence")

BillData <- data.frame(age = 40,weight = 220,height= 76,neck = 40,abdomen = 113,knee= 34,ankle= 20)

predict(fitbit, BillData, interval = "confidence")

#Tom's data is predicted more precisely. This is because his collective values are not as big as Bills.

#We can see this in action when the predict is used, Tom's lwr and upr fit are closer together than Bills.

#(b)

#i

fit4 <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)

summary(fit4)

#Tom's Data is the more precise between the two. This is because the interval is narrower.

#Positve impact: Age,Abdomen,Knee

#Negative impact: Height, Neck

#Significant Variables: Knee

#Insignificant Variables: Age, Height, Neck, Abdomen

#While the impacts are the same, the amount of significant Variables went drastically down leaving only Knee left.

#They of course, became insignificant variables.

#ii

FatAge <- lm(bfat ~ age +height+neck+abdomen+knee, data = bodyfat)

confint(FatAge,'age')

#Yes these finding are completely different.

#iii

TomData <- data.frame(age = 49,height= 68,neck = 37,abdomen = 90,knee= 38)

predict(fit4 , TomData, interval="confidence")

```
BillData <- data.frame(age = 40,height= 76,neck = 40,abdomen = 113,knee= 34)
predict(fit4, BillData, interval = "confidence")
```