# Set 5

Oluwanifemi

2024-10-24

```
library(ggplot2)
library(stats)
library(readxl)
```

## Question One

```
birth_dat <- read_excel("birth_dat.xlsx")
head(birth_dat)
```

```
## # A tibble: 6 x 5
##   Nation      Birthrate PerCapIncome PopFarms InfantMort
##   <chr>           <dbl>        <dbl>    <dbl>      <dbl>
## 1 Venezuela        46.4          392     0.4        68.5
## 2 Mexico           45.7          118     0.61       87.8
## 3 Ecuador          45.3           44     0.53      116.
## 4 Colombia         38.6          158     0.53      107.
## 5 Ceylon           37.2           81     0.53       71.6
## 6 Puerto Rico      35            374     0.37       60.2
```

```
X <- model.matrix(~ PerCapIncome + PopFarms + InfantMort, data= birth_dat)
head(X)
```

```
##   (Intercept) PerCapIncome PopFarms InfantMort
## 1           1          392     0.40       68.5
## 2           1          118     0.61       87.8
## 3           1           44     0.53      115.8
## 4           1          158     0.53      106.8
## 5           1           81     0.53       71.6
## 6           1          374     0.37       60.2
```

```
y = birth_dat$Birthrate
XtX <- t(X) %*% X
beta.hat <- solve(XtX, t(X)%*%y)
beta.hat
```

```
##                     [,1]
## (Intercept)  5.553868478
## PerCapIncome 0.006566304
## PopFarms     9.104754908
## InfantMort   0.242689935
```

```r
model <- lm(Birthrate ~ PerCapIncome + PopFarms + InfantMort, data = birth_dat)
coef(model)
```

```
##  (Intercept) PerCapIncome     PopFarms   InfantMort
##  5.553868478  0.006566304  9.104754908  0.242689935
```

```r
#The results are the similar so in comparison, they are equal to each other.

mexico_row <- birth_dat[birth_dat$Nation == "Mexico", ]
mexico_row
```

```
## # A tibble: 1 x 5
##   Nation Birthrate PerCapIncome PopFarms InfantMort
##   <chr>      <dbl>        <dbl>    <dbl>      <dbl>
## 1 Mexico      45.7          118     0.61       87.8
```

```r
X_mexico <- model.matrix(~ PerCapIncome + PopFarms + InfantMort, data = mexico_row)
fitted_mexico <- X_mexico %*% beta.hat
fitted_mexico
```

```
##       [,1]
## 1 33.19077
```

```r
new_country <- data.frame(
  PerCapIncome = 600,
  PopFarms = 0.2,
  InfantMort = 20
)

predicted_value <- predict(model, newdata = new_country)
predicted_value
```

```
##       1
## 16.1684
```

```r
residuals <- residuals(model)
RSS <- (sum(residuals^2))
RSS
```

```
## [1] 1423.854
```

## Question 2

```r
x <- 1:20
set.seed(463)
y <- x + rnorm(20)


model1 <-lm(y~x)
beta_model1 <- coef(model1)
beta_model1
```

```
## (Intercept)           x
##   0.0876554   1.0136850
```

```
X <- cbind(1, x)
XtX <- t(X) %*% X
beta_hat <- solve(XtX, t(X)%*%y)
beta_hat
```

```
##          [,1]
##   0.0876554
## x 1.0136850
```

```
x_2<- x^2
model2 <-lm(y~x + x_2)
coef(model2)
```

```
##  (Intercept)            x          x_2
## -0.298169776  1.118910079 -0.005010717
```

```
X <- model.matrix(~ x + x_2)
#head(X)
XtX1 <- t(X) %*% X
beta_hat1 <- solve(XtX1, t(X)%*%y)
beta_hat1
```

```
##                     [,1]
## (Intercept) -0.298169776
## x            1.118910079
## x_2         -0.005010717
```

```
# Function to calculate estimates for polynomial terms up to degree 10
calculate_polynomial_estimates <- function(degree) {

  X_poly <- sapply(0:degree, function(d) x^d)

  # Attempting direct inversion
  tryCatch({
    XtX1 <- t(X_poly) %*% X_poly
    beta_direct <- solve(XtX1, t(X_poly)%*%y)
    return(beta_direct)
  }, error = function(e) {
    return("Direct inversion failed")
  })
}


for (degree in 0:10) {
  cat("Degree:", degree, "\n")
  print(calculate_polynomial_estimates(degree))
}
```

```
## Degree: 0
##          [,1]
## [1,] 10.73135
## Degree: 1
##           [,1]
## [1,] 0.0876554
## [2,] 1.0136850
## Degree: 2
##              [,1]
## [1,] -0.298169776
## [2,]  1.118910079
## [3,] -0.005010717
## Degree: 3
##              [,1]
## [1,] -1.134761912
## [2,]  1.545787791
## [3,] -0.054611041
## [4,]  0.001574613
## Degree: 4
##               [,1]
## [1,] -1.490596e+00
## [2,]  1.824149e+00
## [3,] -1.109391e-01
## [4,]  5.676782e-03
## [5,] -9.767067e-05
## Degree: 5
##               [,1]
## [1,] -1.280435e+00
## [2,]  1.600212e+00
## [3,] -4.334297e-02
## [4,] -2.574586e-03
## [5,]  3.384335e-04
## [6,] -8.306746e-06
## Degree: 6
## [1] "Direct inversion failed"
## Degree: 7
## [1] "Direct inversion failed"
## Degree: 8
## [1] "Direct inversion failed"
## Degree: 9
## [1] "Direct inversion failed"
## Degree: 10
## [1] "Direct inversion failed"
```

```r
#It fails when the degree is 6.


set.seed(463)
x2 <- x + rnorm(20, sd = 0.1)
model3 <- lm(y ~ x + x2)
correlation_x_x2 <- cor(x, x2)
beta_lm3 <- coef(model3)
correlation_x_x2
```

```
## [1] 0.9999045
```

```
beta_lm3
```

```
##   (Intercept)             x            x2
##   5.08423e-14 -9.00000e+00  1.00000e+01
```

```
#Both of them are highly correlated


set.seed(463)
x3 <- x + rnorm(20, sd = 0.001)
model4 <- lm(y ~ x + x3)
beta_lm4 <- coef(model4)
cor(x, x3)
```

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

```
beta_lm4
```

```
##    (Intercept)             x             x3
##   4.880861e-12 -9.990000e+02  1.000000e+03
```

**c) It fails when the number of degree is 6.**

**d) They are highly correlated (although this might raise the concern for multicollinearity).**

**e) Using the correlation being "1", there might be an issue of multicollinearity.The coefficent for x is approximately -999 and the coefficent for x3 is approximately 1000, htis might show that an increase in x csn be an almost exactly offset by an increase in x3, btu it mgith seems the model is assigning high sensitivity for small chnages in data.**