Applied-1-prelim-2021

Vi Ly

8/17/2021

Source: <https://rpubs.com/Abhilash333/396604> ## Applied regression Framework: # 1. Loading data:

setwd("C:/Users/lykha/OneDrive/Documents/1\_Qualify-exam-review/5\_Applied-regression (Applied)/Prelim-2021-submission")  
data <- read.csv("PH1821\_data.csv")  
data

## ID AGE GENDER EDU Hypertension Diabetes Cholesterol HeartAttack Stroke  
## 1 1 51 2 2 1 1 20 1 1  
## 2 2 61 2 3 2 2 24 2 2  
## 3 3 57 2 2 2 2 14 2 2  
## 4 4 56 1 2 2 2 11 2 2  
## 5 5 56 2 2 2 2 32 2 2

# Not including the ID column  
data <- data[0:nrow(data), 2:ncol(data)]

## 2. Summary stats:

## 5 number summary stats:  
summary(data)

## AGE GENDER EDU Hypertension   
## Min. :18.00 Min. :1.000 Min. :1.000 Min. :1.000   
## 1st Qu.:38.50 1st Qu.:1.000 1st Qu.:3.000 1st Qu.:2.000   
## Median :47.00 Median :2.000 Median :3.000 Median :2.000   
## Mean :46.89 Mean :1.632 Mean :3.071 Mean :1.847   
## 3rd Qu.:55.00 3rd Qu.:2.000 3rd Qu.:3.000 3rd Qu.:2.000   
## Max. :92.00 Max. :2.000 Max. :4.000 Max. :2.000   
## NA's :3 NA's :2 NA's :6 NA's :1   
## Diabetes Cholesterol HeartAttack Stroke   
## Min. :1.000 Min. :11.00 Min. :1.000 Min. :1.000   
## 1st Qu.:2.000 1st Qu.:19.50 1st Qu.:2.000 1st Qu.:2.000   
## Median :2.000 Median :25.00 Median :2.000 Median :2.000   
## Mean :1.941 Mean :24.53 Mean :1.992 Mean :1.995   
## 3rd Qu.:2.000 3rd Qu.:30.00 3rd Qu.:2.000 3rd Qu.:2.000   
## Max. :2.000 Max. :52.00 Max. :2.000 Max. :2.000   
## NA's :1 NA's :7   
## Cardiovascular Biomarker   
## Min. :1.000 Min. : 0.0   
## 1st Qu.:2.000 1st Qu.: 0.0   
## Median :2.000 Median : 40.0   
## Mean :1.989 Mean : 101.3   
## 3rd Qu.:2.000 3rd Qu.: 160.0   
## Max. :2.000 Max. :1080.0   
## NA's :27

## NA omit to drop the row with missing data:

data <- na.omit(data)  
nrow(data)

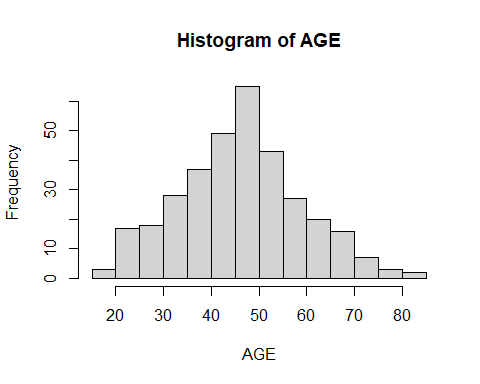
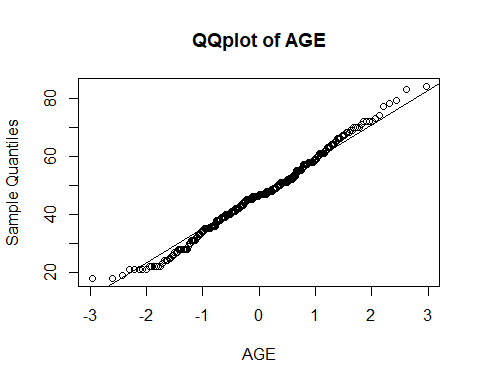
## [1] 335

## 3.1 Normalily checking for continous covariates:

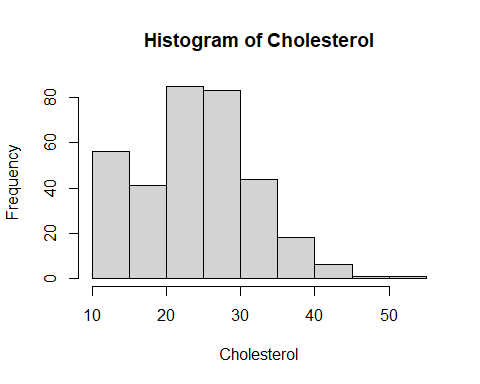
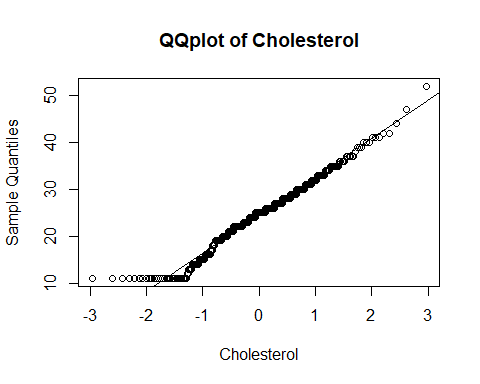
s <- data[, c("AGE", "Cholesterol", "Biomarker")]  
colnames(s)

## [1] "AGE" "Cholesterol" "Biomarker"

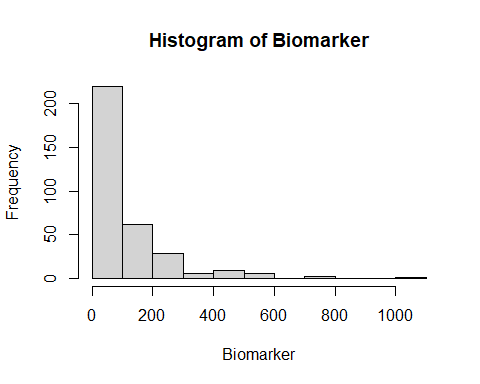
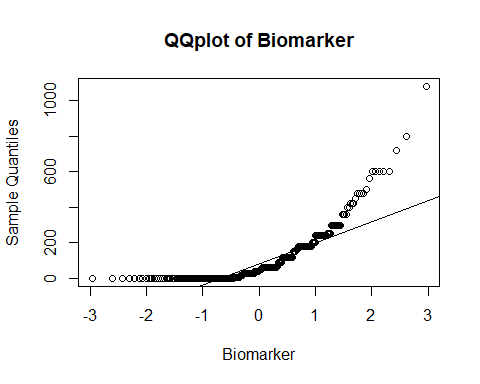
for (i in colnames(s)){  
 qqnorm(s[,i], main = paste("QQplot of", i), xlab=i)  
 qqline(s[,i])  
 hist(s[, i], main=paste("Histogram of", i), xlab=i)  
 print(paste("Shapiro test for", i))  
 print(shapiro.test(s[,i]))  
}



## [1] "Shapiro test for AGE"  
##   
## Shapiro-Wilk normality test  
##   
## data: s[, i]  
## W = 0.9914, p-value = 0.04838



## [1] "Shapiro test for Cholesterol"  
##   
## Shapiro-Wilk normality test  
##   
## data: s[, i]  
## W = 0.97614, p-value = 2.366e-05



## [1] "Shapiro test for Biomarker"  
##   
## Shapiro-Wilk normality test  
##   
## data: s[, i]  
## W = 0.70935, p-value < 2.2e-16

## 3.2 Frequency table of categorical variables:

table\_x <- table(data$GENDER)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c("Male", "Female")  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## Male 129 38.51 129 38.51  
## Female 206 61.49 335 100.00

table\_x <- table(data$EDU)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c(1, 2, 3, 4)  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## 1 7 2.09 7 2.09  
## 2 34 10.15 41 12.24  
## 3 207 61.79 248 74.03  
## 4 87 25.97 335 100.00

table\_x <- table(data$Hypertension)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c("No", "Yes")  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## No 51 15.22 51 15.22  
## Yes 284 84.78 335 100.00

table\_x <- table(data$Diabetes)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c("No", "Yes")  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## No 19 5.67 19 5.67  
## Yes 316 94.33 335 100.00

table\_x <- table(data$HeartAttack)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c("No", "Yes")  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## No 3 0.9 3 0.9  
## Yes 332 99.1 335 100.0

table\_x <- table(data$Stroke)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c("No", "Yes")  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## No 2 0.6 2 0.6  
## Yes 333 99.4 335 100.0

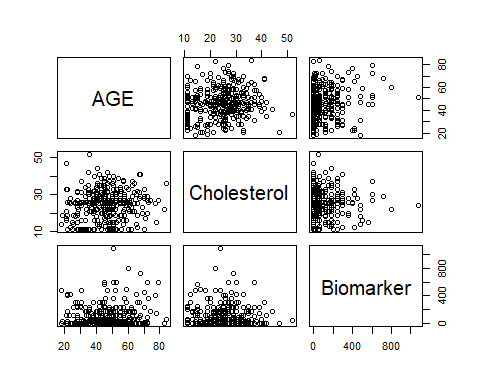
table\_x <- table(data$Cardiovascular)  
cumsum\_table\_x <- cumsum(table\_x)   
n <- nrow(data)  
data\_freq <- data.frame(Freq = as.numeric(table\_x), # Create data frame with relevant values  
 Percent = round(as.numeric(table\_x / n)\*100, 2),  
 Culmulative\_freq = as.numeric(cumsum\_table\_x),  
 Culmulative\_percent = round(as.numeric(cumsum\_table\_x / n)\*100, 2))  
rownames(data\_freq) <- c("No", "Yes")  
data\_freq

## Freq Percent Culmulative\_freq Culmulative\_percent  
## No 4 1.19 4 1.19  
## Yes 331 98.81 335 100.00

Row names are always 0 then 1 #find correct matched categories using the given description

## 3.3 Scatter plot of the continous data:

pairs(s)

 Spearman Correlation Coefficients:

for (i in colnames(s)){  
 print(paste("Correlation test for:", i))  
 print(cor.test(data$Cholesterol, s[, i]))  
}

## [1] "Correlation test for: AGE"  
##   
## Pearson's product-moment correlation  
##   
## data: data$Cholesterol and s[, i]  
## t = 1.9497, df = 333, p-value = 0.05205  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.000926544 0.210990197  
## sample estimates:  
## cor   
## 0.106238   
##   
## [1] "Correlation test for: Cholesterol"  
##   
## Pearson's product-moment correlation  
##   
## data: data$Cholesterol and s[, i]  
## t = Inf, df = 333, p-value < 2.2e-16  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## 1 1  
## sample estimates:  
## cor   
## 1   
##   
## [1] "Correlation test for: Biomarker"  
##   
## Pearson's product-moment correlation  
##   
## data: data$Cholesterol and s[, i]  
## t = 0.44256, df = 333, p-value = 0.6584  
## alternative hypothesis: true correlation is not equal to 0  
## 95 percent confidence interval:  
## -0.08312477 0.13105877  
## sample estimates:  
## cor   
## 0.02424523

## 4. Regression model:

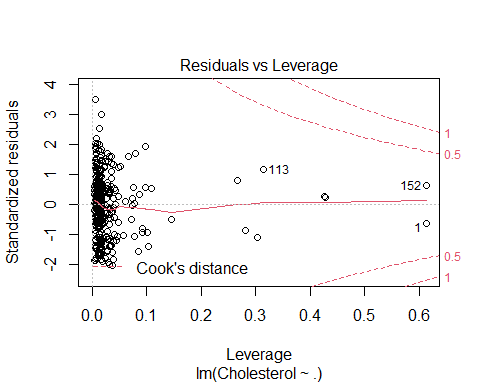
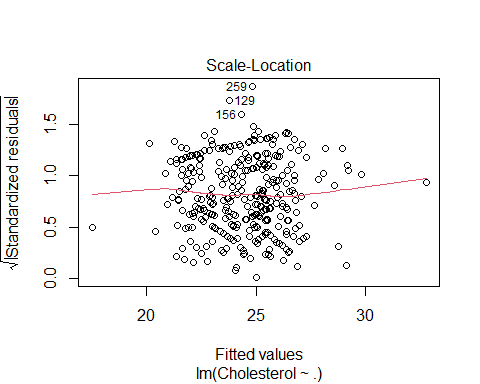
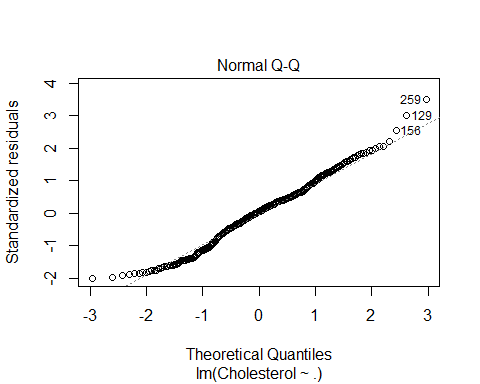
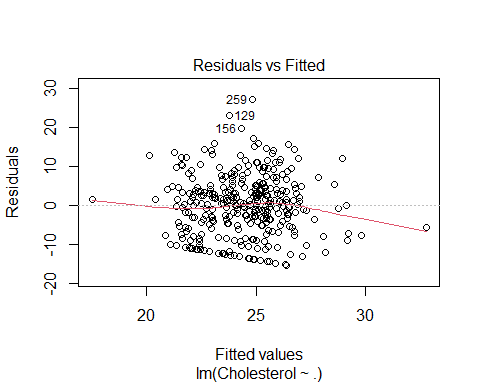
attach(data)  
mod <- lm(Cholesterol ~ ., data = data)  
summary(mod)

##   
## Call:  
## lm(formula = Cholesterol ~ ., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -15.4071 -5.1523 0.4122 4.5665 27.1733   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 23.951540 14.998770 1.597 0.1113   
## AGE 0.063415 0.035540 1.784 0.0753 .  
## GENDER 2.328209 0.924255 2.519 0.0122 \*  
## EDU -1.047214 0.683895 -1.531 0.1267   
## Hypertension 0.390768 1.319737 0.296 0.7673   
## Diabetes 1.787375 2.080401 0.859 0.3909   
## HeartAttack 4.117977 5.031286 0.818 0.4137   
## Stroke -2.530779 6.293550 -0.402 0.6879   
## Cardiovascular -5.271969 4.026906 -1.309 0.1914   
## Biomarker 0.001814 0.002946 0.616 0.5385   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 7.806 on 325 degrees of freedom  
## Multiple R-squared: 0.05448, Adjusted R-squared: 0.0283   
## F-statistic: 2.081 on 9 and 325 DF, p-value: 0.03079

## 5. Checking the regression model assumptions on residuals:

1.linearity 2.normality 3.homoscadescity 4.independency

plot(mod)

 Here in residuals vs fitted plot the red line is almost lying near to zero residual value and is almost horizontal and all the fitted values are scattered around it without any systematic relationship. Therefore , LINEARITY is met on residuals

In normal q-q plot drawn, the residuals are almost linearly distributed.(but lets check normaly futher using other tests)

In scale-location plot,all the residuals are scattered(i.e none of the points are clustered at one spot). Therefore, HOMOSCADESCITY IS MET on residuals.

## Checking for normality on residuals:

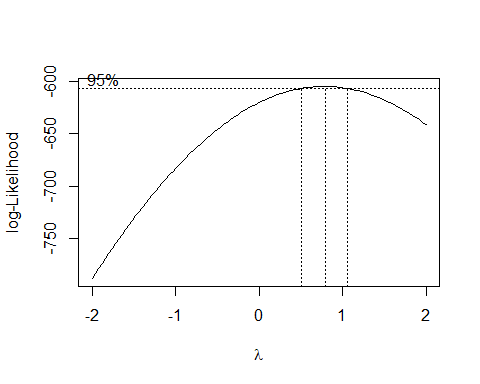
shapiro.test(mod$residuals)

##   
## Shapiro-Wilk normality test  
##   
## data: mod$residuals  
## W = 0.98705, p-value = 0.00432

Here the probability value of both Shapiro Wilk is more than 0.05 hence, we accept null hypothesis saying that the residual data is normally distributed. And we also have skewness nearly equal to zero and kurtosis nearly equal to 3 where we can say that residual data is normally distributed. Therefore, NORMALITY IS MET on residuals

No violation so no need for Box-cox transforamtion. # Box-cox transformation (if necessary): #find optimal lambda for Box-Cox transformation

library(MASS)  
bc <- boxcox(Cholesterol ~., data = data)



(lambda <- bc$x[which.max(bc$y)])

## [1] 0.7878788

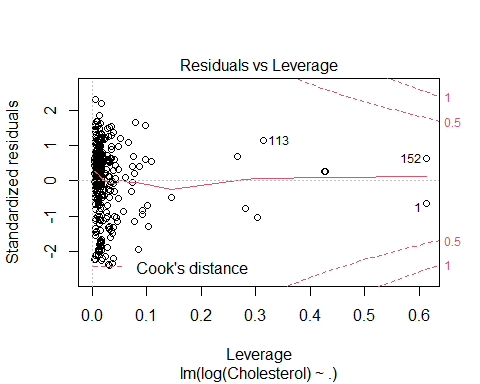
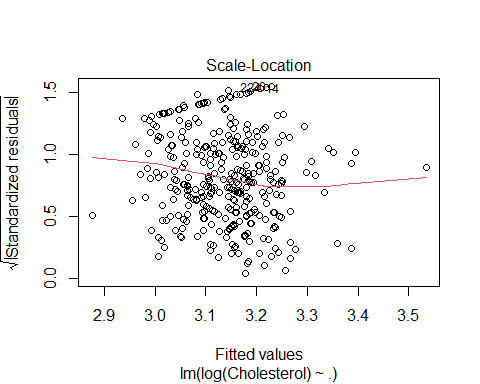
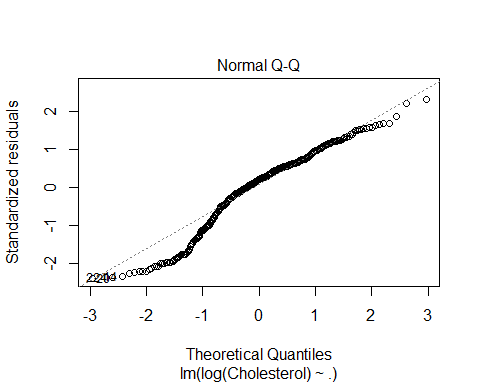
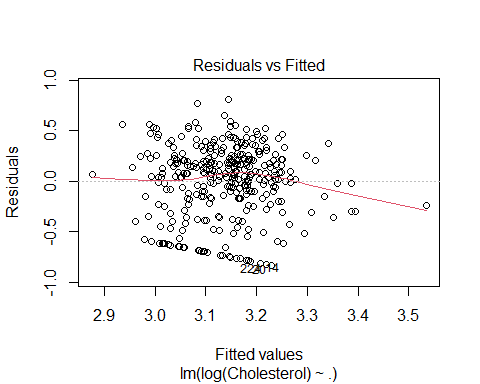
#fit new linear regression model using the Box-Cox transformation  
new\_model <- lm(((Cholesterol^lambda-1)/lambda) ~ ., data=data)

## For convinient, the log transformation is performed to save time:

trans.mod <- lm(log(Cholesterol) ~., data = data)  
summary(trans.mod)

##   
## Call:  
## lm(formula = log(Cholesterol) ~ ., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.83105 -0.17185 0.06837 0.22225 0.80741   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3.2037268 0.6786322 4.721 3.5e-06 \*\*\*  
## AGE 0.0035566 0.0016080 2.212 0.0277 \*   
## GENDER 0.1027026 0.0418187 2.456 0.0146 \*   
## EDU -0.0413272 0.0309434 -1.336 0.1826   
## Hypertension 0.0271980 0.0597126 0.455 0.6491   
## Diabetes 0.0824822 0.0941295 0.876 0.3815   
## HeartAttack 0.1296525 0.2276448 0.570 0.5694   
## Stroke -0.1392443 0.2847570 -0.489 0.6252   
## Cardiovascular -0.2380750 0.1822008 -1.307 0.1923   
## Biomarker 0.0001323 0.0001333 0.992 0.3217   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3532 on 325 degrees of freedom  
## Multiple R-squared: 0.05724, Adjusted R-squared: 0.03113   
## F-statistic: 2.192 on 9 and 325 DF, p-value: 0.02229

plot(trans.mod)



shapiro.test(mod$residuals)

##   
## Shapiro-Wilk normality test  
##   
## data: mod$residuals  
## W = 0.98705, p-value = 0.00432

# 6. Stepwise selection for multiple covariates:

library(MASS)  
# Fit the full model   
full.model <- lm(log(Cholesterol) ~., data = data)  
# Stepwise regression model  
step.model <- stepAIC(full.model, direction = "both", scope = list(lower = ~ Biomarker),  
 trace = FALSE)  
summary(step.model)

##   
## Call:  
## lm(formula = log(Cholesterol) ~ AGE + GENDER + Biomarker, data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.85837 -0.18054 0.06003 0.23485 0.81283   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 2.7654163 0.0990560 27.918 < 2e-16 \*\*\*  
## AGE 0.0035852 0.0015156 2.366 0.01858 \*   
## GENDER 0.1185996 0.0400153 2.964 0.00326 \*\*   
## Biomarker 0.0001346 0.0001314 1.024 0.30645   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.3525 on 331 degrees of freedom  
## Multiple R-squared: 0.04344, Adjusted R-squared: 0.03477   
## F-statistic: 5.011 on 3 and 331 DF, p-value: 0.00207

Final model

exp(0.0001346)

## [1] 1.000135