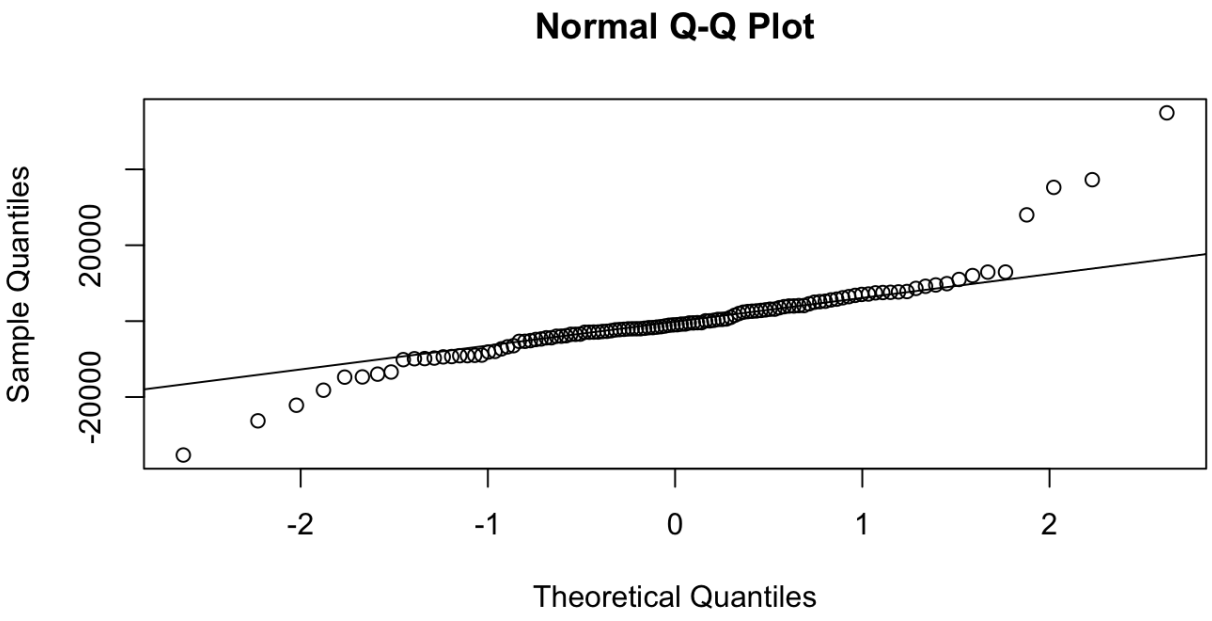
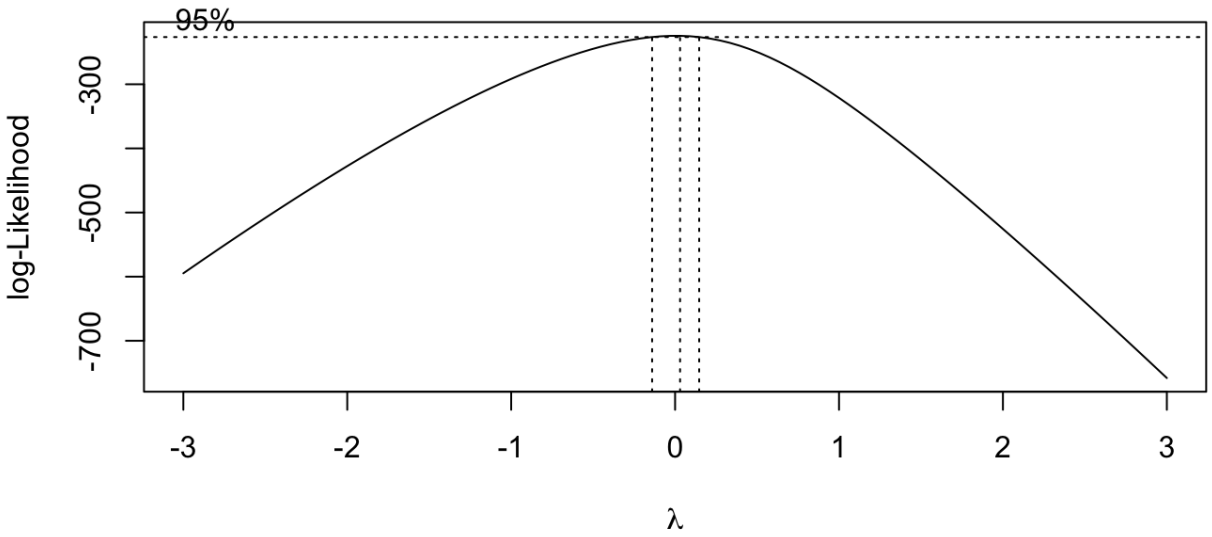
Question 2: Do complete series and first dose have the same impact on infected cases, given death cases?

```
> #Load dataset
> library(car)
> library(carData)
> library(zoo)
> library(MASS)
> library(lmtest)
> library(boot)
> library(fmsb)
> library(leaps)
> library(caret)
> library(lmridge)

> #Load dataset
> covid = read.csv("/Users/taiyangfurenmrssun/Desktop/Academics/2023 Spring/STAT 512/
Project/covid.csv")

> #Build model
> model = lm(infection ~ Doseone + Series + death, data = covid)
> new = subset(covid, select = -c(TotalVac, Booster, Mask))

> #Check for constant variance
> bptest(model)

        studentized Breusch-Pagan test

data:  model
BP = 35.205, df = 3, p-value = 1.103e-07
Residuals have non-constant variance

> #Check for normality
shapiro.test(residuals(model))

        Shapiro-Wilk normality test

data:  residuals(model)
W = 0.85227, p-value = 2.153e-09
Residuals are not normally distributed
> qqnorm(residuals(model))
> qqline(residuals(model))
```



```
> #Transformation on x
> new$Doseone = log(new$Doseone)
> new$Series = log(new$Series)
> new$death = log(new$death)
> modelNew = lm(infection ~ Doseone + Series + death, data = new)
```
Take natural log of x to transform x

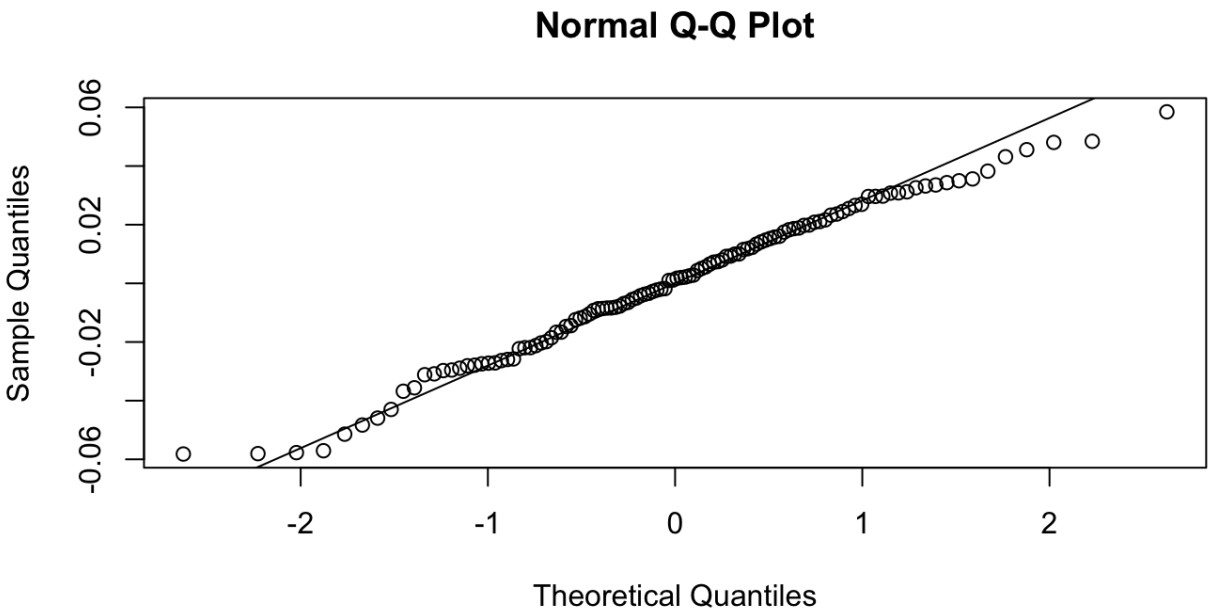
```
> #Transformation on y
> bc = boxcox(modelNew, lambda = seq(-3, 3,by = 0.1))
```




```
> lambda = bc$x[which.max(bc$y)]
> lambda
[1] 0.03030303
> new$newY = new$infection ^ lambda
> modelNew = lm(new$newY ~ Doseone + Series + death, data = new)
> new = new[, -3]
```
Box-cox transformation on y, lambda = 0.0303

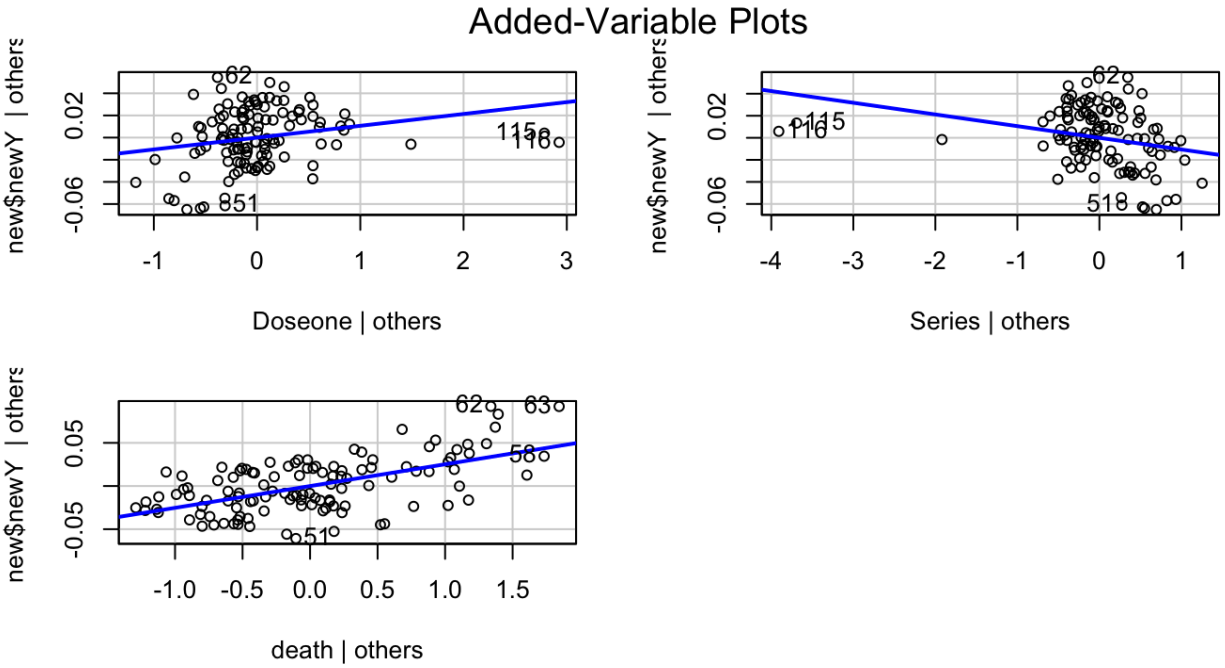
```
> #Check for constant variance after transformation
> bptest(modelNew)

        studentized Breusch-Pagan test

data:  modelNew
BP = 7.5565, df = 3, p-value = 0.05612
```
Residuals now have constant variance

```
> #Check for normality after transformation
> shapiro.test(residuals(modelNew))

        Shapiro-Wilk normality test

data:  residuals(modelNew)
W = 0.98693, p-value = 0.327
```
Residuals are now normally distributed.




```
> qqnorm(residuals(modelNew))
> qqline(residuals(modelNew))
```

```
> #Check for marginal effect of predictors
> avPlots(modelNew)
```



All predictors have marginal contribution

```
> #Check for x/y outliers
According to studentized residuals, there are no y outliers
> sum(abs(rstudent(modelNew)) > 2)
[1] 0
> infl = lm.influence(modelNew)$hat
```

According to hat matrix, there are 3 x outliers

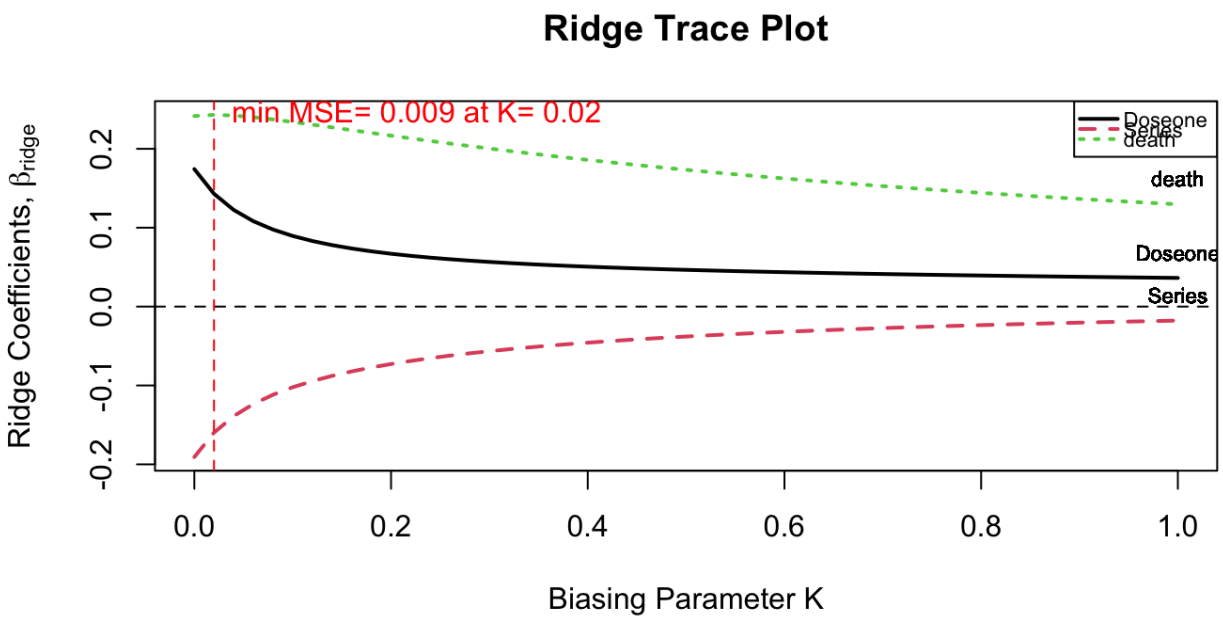
```
> length(which(infl[] > 2 * (4 / 116)))
[1] 3

> #Check for influential points
> d = dfbetas(modelNew)
> sum(d[which(abs(d[, 2]) > 1 & abs(d[, 3])> 1 & abs(d[, 4]) > 1)])
[1] 0
> dff = dffits(modelNew)
> length(dff[dff > 1])
[1] 0
> minor = qf(0.2, df1 = 4, df2 = 116 - 4)
> major = qf(0.5, df1 = 4, df2 = 116 - 4)
> Cooksdistance = cooks.distance(modelNew)
> sum(Cooksdistance > minor)
[1] 0
> sum(Cooksdistance > major)
[1] 0
```

According to DFFITS, DFBETAS, and Cook's distance, there are no influential points

```
> #Check for multicollinearity
> VIF(lm(Doseone~Series+death, new))
[1] 7.283964
> VIF(lm(Series~Doseone+death, new))
[1] 6.085945
> VIF(lm(death~Doseone+Series, new))
[1] 1.531526
```

According to Variance Inflation Factor, minor multicollinearity exist

```
> #Ridge regression for multicollinearity
K = 0.14
> rid = lmridge(newY ~ Doseone + Series + death, data = new, K = seq(0, 1, 0.02))
> plot(rid)
```




```
> vif(rid)
      Doseone  Series   death
k=0   7.28396 6.08595 1.53153
```

k=0.02 4.65084 3.95688 1.31869
k=0.04 3.25454 2.82358 1.18521
k=0.06 2.42528 2.14714 1.08976
k=0.08 1.89220 1.70962 1.01549
k=0.1 1.52885 1.40920 0.95437
k=0.12 1.26971 1.19316 0.90216
k=0.14 1.07811 1.03194 0.85640
k=0.16 0.93221 0.90792 0.81555
k=0.18 0.81835 0.81007 0.77862
k=0.2 0.72762 0.73121 0.74489
k=0.22 0.65401 0.66647 0.71385
k=0.24 0.59337 0.61248 0.68513
k=0.26 0.54271 0.56681 0.65843
k=0.28 0.49988 0.52771 0.63350
k=0.3 0.46328 0.49387 0.61017
k=0.32 0.43170 0.46430 0.58826
k=0.34 0.40420 0.43824 0.56764
k=0.36 0.38008 0.41509 0.54819
k=0.38 0.35876 0.39439 0.52982
k=0.4 0.33980 0.37575 0.51244
k=0.42 0.32282 0.35888 0.49597
k=0.44 0.30755 0.34353 0.48033
k=0.46 0.29373 0.32950 0.46548
k=0.48 0.28117 0.31661 0.45135
k=0.5 0.26971 0.30473 0.43789
k=0.52 0.25920 0.29374 0.42506
k=0.54 0.24953 0.28353 0.41282
k=0.56 0.24060 0.27403 0.40113
k=0.58 0.23233 0.26515 0.38995
k=0.6 0.22464 0.25684 0.37926
k=0.62 0.21748 0.24904 0.36902
k=0.64 0.21078 0.24169 0.35921
k=0.66 0.20451 0.23477 0.34981
k=0.68 0.19862 0.22823 0.34079
k=0.7 0.19308 0.22204 0.33213
k=0.72 0.18785 0.21616 0.32380
k=0.74 0.18291 0.21059 0.31580
k=0.76 0.17823 0.20528 0.30811
k=0.78 0.17380 0.20023 0.30070
k=0.8 0.16958 0.19541 0.29357
k=0.82 0.16558 0.19080 0.28670
k=0.84 0.16176 0.18640 0.28008
k=0.86 0.15811 0.18218 0.27369
k=0.88 0.15463 0.17814 0.26753
k=0.9 0.15129 0.17426 0.26158
k=0.92 0.14810 0.17054 0.25583
k=0.94 0.14504 0.16696 0.25028
k=0.96 0.14210 0.16352 0.24492
k=0.98 0.13928 0.16020 0.23973
k=1   0.13657 0.15700 0.23471
> summary(lmridge(newY ~ Doseone + Series + death,data = new, K = 0.14))

Call:
lmridge.default(formula = newY ~ Doseone + Series + death, data = new,
    K = 0.14)

Coefficients: for Ridge parameter K= 0.14
          Estimate Estimate (Sc) StdErr (Sc) t-value (Sc) Pr(>|t|)
Intercept   1.2066   0.3447   0.3763   0.9162   0.3615

```
Doseone     0.0048       0.0779      0.0274       2.8452   0.0053 **
Series     -0.0049      -0.0879      0.0268      -3.2780   0.0014 **
death       0.0238       0.2272      0.0244       9.3040   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Ridge Summary
       R2     adj-R2   DF ridge         F        AIC        BIC
  0.37720    0.36610    2.13765   38.30602 -841.63815 -284.33547
Ridge minimum MSE= 0.02213321 at K= 0.14
P-value for F-test ( 2.13765 , 113.4471 ) = 5.317365e-14
------------------------------------------------------------------
```

According to ridge regression, linear impact of dose one is strictly positive, linear impact of complete series is negative.

```
> #Bootstrapping for nornormality, on top of ridge regression
> boot.Rid = function(data, indices, maxit = 100){
+   data<-data[indices,]
+   mod<-lmridge(newY ~ Doseone + Series + death, data = data, maxit = maxit, K = 0.14)
+   return(coefficients(mod))
+ }
> modelrid = boot(data=new,statistic = boot.Rid, R = 100, maxit = 100)
> boot.ci(modelrid, index = 2, type="perc")
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = modelrid, type = "perc", index = 2)

Intervals :
Level     Percentile
95%   ( 0.0024,  0.0080 ) Bootstrap confidence interval suggests linear impact of dose one is strictly positive
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable
> boot.ci(modelrid, index = 3, type="perc")
BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
Based on 100 bootstrap replicates

CALL :
boot.ci(boot.out = modelrid, type = "perc", index = 3)

Intervals :
Level     Percentile
95%   (-0.0069, -0.0032 ) Bootstrap confidence interval suggests linear impact of complete series is strictly negative
Calculations and Intervals on Original Scale
Some percentile intervals may be unstable

> #Reduced model(under H0)
> combined = rowSums(new[, c("Doseone", "Series")])
> reduced = lm(newY ~ combined + death, data = new)
> #Full model(under Ha)
> full = lm(newY ~ Doseone + Series + death, data = new)
> summary(full)

Call:
lm(formula = newY ~ Doseone + Series + death, data = new)

Residuals:
```

```
      Min        1Q    Median        3Q       Max
-0.058194 -0.018880  0.001447  0.019094  0.058479

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  1.196178   0.016614  71.998  < 2e-16 ***
Doseone      0.010701   0.004322   2.476  0.01478 *
Series      -0.010609   0.003580  -2.964  0.00371 **
death        0.025270   0.003377   7.483 1.77e-11 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.02608 on 112 degrees of freedom
Multiple R-squared:  0.5122,  Adjusted R-squared:  0.4991
F-statistic: 39.19 on 3 and 112 DF,  p-value: < 2.2e-16
```

According to OLS, linear impact of dose one is strictly positive, linear impact of complete series is negative.

```
> #F test for hypothesis testing
> MSR = (sum(reduced$residuals ^ 2) - sum(full$residuals ^ 2))/ (reduced$df.residual -
full$df.residual)
> MSE = sum(full$residuals ^ 2) / full$df.residual
> FS = MSR / MSE
> FS
[1] 7.627347 Test statistic
> qf(1 - 0.05, reduced$df.residual - full$df.residual, full$df.residual)
[1] 3.925834 Critical value
> p = 1 - pf(FS, reduced$df.residual - full$df.residual, full$df.residual)
> p
[1] 0.006720598 p-value
```

Since test statistic is greater than critical value and p-value is less than alpha = 0.05, we reject the null hypothesis and conclude that complete series and first dose do not have the same impact on infected cases, given death cases.

```
> #K-fold cross validation to check predictability
> set.seed(123)
> train.control<-trainControl(method = 'cv', number = 5)
> step.model1 = train(newY ~ Doseone + Series + death, data = new, method="leapBackward",
tuneGrid = data.frame(nvmax = 4), trControl = train.control)
> step.model1$results
  nvmax       RMSE Rsquared        MAE      RMSESD RsquaredSD       MAESD
1     4 0.02722844 0.487252 0.02208553 0.002404326  0.1818767 0.002062165
```

Model has good predictive power.