

```

library(readxl)
Parameters.data <- read_excel("2020 - assignment 2.xlsx", sheet="Parameters")

#install packages

install.packages("car")
install.packages("psych")
install.packages("carData")
install.packages("mvtnorm")
install.packages("tidyverse")
install.packages("survival")
install.packages("TH.data")
install.packages("MASS")
install.packages("emmeans")

library(tidyverse)

library(car)
library(carData)
library(psych)
library(lattice)

library(data.table)
library(plyr)
library(doBy)

library(car)
library(carData)
library(afex)

library(multcomp)
library(lsmeans)
library(emmeans)
library(doBy)
library(multcomp)

# a. Try different models [ $y \sim x$ ;  $\log(y) \sim x$  and  $\sqrt{y} \sim x$ ] and check the
assumptions of each
## First model

#Rename Variables
Parameters.data<- rename (Parameters.data,c("CPM_Treatment" = "CPM", "Growth
(g)" = "Growth"))
Parameters.data<- rename (Parameters.data,c("Weight start (g)" =
"Weight.start", "Weight end (g)" = "Weight.end"))

Parameters.data<- rename (Parameters.data,c("S-methylglutathione" =
"methylglutathione"))

#One way ANOVA

```

```

summaryBy (Growth ~factor(CPM), data=Parameters.data,
           FUN=c(mean,sd,length))

#Graphically
boxplot(Growth ~factor(CPM), data=Parameters.data, ylab="Growth")

#T test

pairwise.t.test(Parameters.data$Growth, Parameters.data$CPM,
p.adj="none",paired=F)

# one way ANOVA

riaz1=aov(Growth ~ CPM, data=Parameters.data)
summary.lm(riaz1)
summary (riaz1)

fit15<- lm (Growth ~ CPM, data=Parameters.data)
summary (fit15)

AIC(fit15)
BIC(fit15)
anova(fit15)

#mODEL Diagnosis
# variance are homogenous

leveneTest(Parameters.data$Growth, Parameters.data$CPM)

#The Shapiro Wilk's W test
shapiro.test(residuals(fit15))

studres=rstudent(fit15)
hist(studres,
     probability=T,
     col="lightgrey",
     xlim=c(-6,6),
     breaks=12,
     main="Distribution of Studentized Residuals",
     xlab="Studentized residuals")
xfit=seq(-6,6,length=100)
yfit=dnorm(xfit) # normal fit
lines(xfit, yfit, col="red",lwd=2)

```

```
shapiro.test(residuals(riaz1))
```

```
#Graphically
```

```
studres=rstudent(riaz1)
hist(studres,
      probability=T,
      col="lightgrey",
      xlim=c(-6,6),
      breaks=12,
      main="Distribution of Studentized Residuals",
      xlab="Studentized residuals")
xfit=seq(-6,6,length=100)
yfit=dnorm(xfit) # normal fit
lines(xfit, yfit, col="red",lwd=2)
```

```
#####
#####
```

```
#Does CPM and weight affect the level of S-methylglutathione? In other words,
#is there an effect of CPM on the level of S-methylglutathione, after
accounting for the effect of weight?
```

```
#effect of CPM on the level of S-methylglutathione before considering for the
effect of weight
```

```
fit5<- lm (methylglutathione ~ CPM, data=Parameters.data)
summary (fit5)
```

```
summary(fit5)$coefficients
summary(fit5)$r.squared
summary(fit5)$adj.r.squared
AIC(fit5)
BIC(fit5)
anova(fit5)
```

```
# Testing assumptions of linear model
```

```
plot(residuals(fit5))
#Normality of residuals
```

```
# variances along the regression line
```

```
hist(residuals(fit5))
#The Shapiro Wilk's W test
shapiro.test(residuals(fit5))
```

```

studres=rstudent(fit5)
hist(studres,
      probability=T,
      col="lightgrey",
      xlim=c(-6,6),
      breaks=12,
      main="Distribution of Studentized Residuals",
      xlab="Studentized residuals")
xfit=seq(-6,6,length=100)
yfit=dnorm(xfit) # normal fit
lines(xfit, yfit, col="red",lwd=2)

```

```

#effect of CPM on the level of S-methylglutathione after considering the
effect of weight
# multiple linear regression model
plot(methylglutathione ~ factor(CPM), data=Parameters.data)
plot(methylglutathione ~ Weight.start, data=Parameters.data)
plot(methylglutathione ~ Weight.end, data=Parameters.data)

```

```

fit14<- lm (methylglutathione ~ CPM, data=Parameters.data)
summary (fit14)
AIC(fit14)
BIC(fit14)
anova(fit14)

```

```

# Testing assumptions of linear model

```

```

plot(residuals(fit14))

```

```

#Normality of residuals

```

```

# variances along the regression line

```

```

hist(residuals(fit14))

```

```

#The Shapiro Wilk's W test
shapiro.test(residuals(fit14))

```

```

studres=rstudent(fit14)
hist(studres,
      probability=T,
      col="lightgrey",
      xlim=c(-6,6),
      breaks=12,
      main="Distribution of Studentized Residuals",
      xlab="Studentized residuals")

```

```

xfit=seq(-6,6,length=100)
yfit=dnorm(xfit) # normal fit
lines(xfit, yfit, col="red",lwd=2)

```

```

#####

```

```

#####question no

```

```

2#####

```

```

#After considering weight in the model

```

```

summaryBy(methylglutathione~CPM+Weight.start+Weight.end, data=Parameters.data,
          FUN=c(mean,sd,length))

```

```

bwplot(methylglutathione~Weight.start|factor(CPM), data=Parameters.data,
fill="blue")

```

```

bwplot(methylglutathione~Weight.end|factor(CPM), data=Parameters.data,
fill="blue")

```

```

bwplot(methylglutathione~factor(CPM), data=Parameters.data, fill="blue")

```

```

plot(effect(term="methylglutathione:CPM",
mod=fit5,default.levels=20),multiline=TRUE)
leveneTest(Parameters.data$methylglutathione, Parameters.data$CPM)
leveneTest(Growth~factor(CPM), data=Parameters.data.riaz)

```

```

fit12<- lm (methylglutathione ~ CPM+Weight.start+Weight.end,
data=Parameters.data)
summary (fit12)

```

```

summary(fit12)$coefficients
summary(fit12)$r.squared
summary(fit12)$adj.r.squared
AIC(fit12)
BIC(fit12)
anova(fit12)

```

```

#The Shapiro Wilk's W test
shapiro.test(residuals(fit12))

```

```

studres=rstudent(fit12)

```

```
hist(studres,  
      probability=T,  
      col="lightgrey",  
      xlim=c(-6,6),  
      breaks=12,  
      main="Distribution of Studentized Residuals",  
      xlab="Studentized residuals")  
xfit=seq(-6,6,length=100)  
yfit=dnorm(xfit) # normal fit  
lines(xfit, yfit, col="red",lwd=2)
```

```
fit13<- lm (methylglutathione ~ CPM+Weight.start: CPM +Weight.end:CPM,  
data=Parameters.data)
```

```
summary (fit13)  
AIC(fit13)  
BIC(fit13)  
anova(fit13)
```

```
#The Shapiro Wilk's W test  
shapiro.test(residuals(fit13))
```

```
# Graphical histogram  
studres=rstudent(fit13)  
hist(studres,  
      probability=T,  
      col="lightgrey",  
      xlim=c(-6,6),  
      breaks=12,  
      main="Distribution of Studentized Residuals",  
      xlab="Studentized residuals")  
xfit=seq(-6,6,length=100)  
yfit=dnorm(xfit) # normal fit  
lines(xfit, yfit, col="red",lwd=2)
```