P1

Q1

```
data()
data(iris)
head(iris)
nrow(iris)
ncol(iris)
dim(iris);names(iris)
summary(iris)
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
data = read.csv("titanic.csv",header=TRUE)
View(data)
library("rjson")

data_json <- fromJSON(file="employees.json")
json_data_frame <- as.data.frame(data_json)
print(json_data_from)
View(json_data_from)
dim(data)
dim(json_data_from)
data$Name
json_data_frame$employee.firstName
head(data,2)
head(json_data_frame)
```

Q3

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
data = read.csv("titanic.csv",header=TRUE)
View(data)
is.na(data)
is.na(data$Age)
which(is.na(data$Age))
newdata <- na.omit(data)
View(newdata)
data$Age[is.na(data$Age)]<-0
data = read.csv("titanic.csv",header=TRUE)
m=mean(data$Age,na.rm=TRUE)
data$Age[is.na(data$Age)] <- m
View(data)
data = read.csv("titanic.csv",header=TRUE)
m=median(data$Age,na.rm=TRUE)
data$Age[is.na(data$Age)] <- m
View(data)
```

Q4 1

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
bike_data = read.csv("day.csv",header=TRUE)
boxplot(bike_data[,c('temp','atemp','hum','windspeed')])
median_value<-median(bike_data$hum)
high<-mean(bike_data$hum) + 2* sd(bike_data$hum)
low<-mean(bike_data$hum) - 2 * sd(bike_data$hum)
bike_data$hum<-ifelse(bike_data$hum>high | bike_data$hum<low,
median_value, bike_data$hum)
boxplot(bike_data[,c('temp','atemp','hum','windspeed')])

median_value<-median(bike_data$windspeed)
sd(bike_data$windspeed)
high<-mean(bike_data$windspeed) + 2* sd(bike_data$windspeed)
low<-mean(bike_data$windspeed) - 2* sd(bike_data$windspeed)
bike_data$windspeed<-ifelse(bike_data$windspeed>high |
bike_data$windspeed<low, median_value,bike_data$windspeed)
boxplot(bike_data[,c('temp','atemp','hum','windspeed')])
```

Q4

```
data <- iris[,1:4]
dim(data)
boxplot(data,c(1,2,3,4))
quartiles <- quantile(data$Sepal.Width,probs=c(.25, .75), na.rm=FALSE)
IQR <- IQR(data$Sepal.Width)
Lower <- quartiles[1] - 1.5*IQR
Upper <- quartiles[2] + 1.5*IQR
data_no_outlier <- subset(data, data$Sepal.Width > Lower &
data$Sepal.Width < Upper)
dim(data_no_outlier)
boxplot(data_no_outlier,c(1,2,3,4))
```

Q5

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
install.packages("dplyr")
library("dplyr")
students = read.csv("Students.csv",header=TRUE)
filter(students,TotalMarks<500)
select(students, -Gender)
filter(students,TotalMarks>850,Class=="TYCS")
filter(students, TotalMarks>900, Class %in% c("SYCS","TYCS"))
filter(students, TotalMarks>900, Class=="SYCS" | Class=="TYCS")
filter(students,Class=="TYCS" & Gender=="Male")
```

```
students %>% arrange(Name)
students %>% arrange(desc(Name))
students=read.csv("Students.csv",header=TRUE)
students[order(students$Name),]
students[order(students$Name, decreasing = T),]
students %>% count()
count(students)
students %>% group_by(Class) %>% count()
Maximum_marks=students %>% group_by(Class) %>% summarise(total_marks =
max(TotalMarks),)
Maximum_marks
agg_tbl <- students %>% group_by(Class,Gender) %>%
summarise(Average_marks=mean(TotalMarks),min_marks=min(TotalMarks),max_
marks=max(TotalMarks),)
agg_tbl
```

P2

Q1

```
install.packages("caret")
library(caret)
data(iris)
summary(iris[,1:4])
preProcessParams <- preProcess(iris[,1:4],method=c("center","scale"))
transformed <- predict(preProcessParams, iris[,1:4])
summary(transformed)
View(transformed)

install.packages("caret")
library(caret)
data(iris)
summary(iris[,1:4])
preProcessingParams <- preProcess(iris[,1:4], method=c("range"))
print(preProcessParams)
transformed<-predict(preProcessingParams, iris[,1:4])
summary(transformed)
View(transformed)
```

Q1 1

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
bike_data = read.csv("day.csv",header=TRUE)
boxplot(bike_data[,c('temp','atemp','hum','windspeed')])
iqr <- IQR(bike_data$hum)
up<-quantile(bike_data$hum,0.75) + 1.5*iqr
```

```
low<-quantile(bike_data$hum,0.25) - 1.5*iqr
outlier_ind<-which(bike_data$hum<low | bike_data$hum>up)
bike_data[outlier_ind,"hum"]
bike_data[outlier_ind,]
e<-subset(bike_data,bike_data$hum>low & bike_data$hum<up)
boxplot(e[,c('temp','atemp','hum','windspeed')])


iqr <- IQR(e$windspeed)
up<-quantile(e$windspeed,0.75) + 1.5*iqr
low<-quantile(e$windspeed,0.25) - 1.5*iqr
outlier_ind<-which(e$windspeed<low | e$windspeed>up)
```

```
e[outlier_ind,"hum"]
e[outlier_ind,]
e<-subset(e,e$windspeed>low & e$windspeed<up)
boxplot(e[,c('temp','atemp','hum','windspeed')])
```

Q2

```
data = data.frame(var1=c(120,345,145,122,596,285,211),
                  var2=c(10,15,45,22,53,28,12),
                  var3=c(-34,0.05,0.15,0.12,-6,0.85,0.11))
data
summary(data)

data = data.frame(var1=c(120,345,145,122,596,285,211),
                  var2=c(10,15,45,22,53,28,12),
                  var3=c(-34,0.05,0.15,0.12,-6,0.85,0.11))
```

```
#normalization
```

```
data = data.frame(var1=c(120,345,145,122,596,285,211),
                  var2=c(10,15,45,22,53,28,12),
                  var3=c(-34,0.05,0.15,0.12,-6,0.85,0.11))
```

```
preproc <-preProcess(data, method=c("range"))
norm<-predict(preproc, data)
head(norm)
summary(norm)
```

Q3

```
data(iris)
View(iris)
print(iris$Species)
data_new<-sapply(iris,unclass)
View(data_new)
```

```r
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
Students=read.csv("Students.csv",header=TRUE)
Students$Class <- as.factor(Students$Class)
Students$Gender <- as.factor(Students$Gender)
data_new<-sapply(Students,unclass)
View(data_new)
```

```r
df <- data.frame(team=as.factor(c('A','B','C','D')),
                 conf=as.factor(c('AL','AL','NL','NL')),
                 win=as.factor(c('Yes','No','No','Yes')),
                 points=c(122,98,106,115))
```

```r
df
df$team <- unclass(df$team)
df
```

```r
df <- data.frame(team=as.factor(c('A','B','C','D')),
                 conf=as.factor(c('AL','AL','NL','NL')),
                 win=as.factor(c('Yes','No','No','Yes')),
                 points=c(122,98,106,115))
df[, c('team','win')]<-sapply(df[, c('team','win')],unclass)
df
```

```r
df <- data.frame(team=as.factor(c('A','B','C','D')),
                 conf=as.factor(c('AL','AL','NL','NL')),
                 win=as.factor(c('Yes','No','No','Yes')),
                 points=c(122,98,106,115))
df[sapply(df,is.factor)]<-data.matrix(df[sapply(df,is.factor)])
df
```

```r
df <- data.frame(team=as.factor(c('A','B','C','D')),
                 conf=as.factor(c('AL','AL','NL','NL')),
                 win=as.factor(c('Yes','No','No','Yes')),
                 points=c(122,98,106,115))
df$win=factor(df$win,levels = c('No','Yes'),labels=c(0,1))
df
```

```r
install.packages("CatEncoders")
library(CatEncoders)
df <- data.frame(team=c('A','A','B','B','B','B','C','C'),
                 points=c(25,12,15,14,19,23,25,29))
labs = LabelEncoder.fit(df$team)
df$team=transform(labs,df$team)
df
```

P3

```
x=c(89,88,78,76,78,78,86,83,82,76,72,77,92)
t.test(x,mu=80)

mistime=c(85,95,105,85,90,97,104,95,88,90,94,95)
t.test(mistime,mu=90,alternative="less")
```

```
bot_val=c(484.11,459.49,471.38,)
```

P4

Q1

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
s=read.csv("One way anova.csv",header=TRUE)
t=aov(satindex ~ dept, s)
summary(t)
TukeyHSD()
```

Q2

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
s=read.csv("tyre.csv",header=TRUE)
t=aov(Mileage ~ Brands, s)
summary(t)
TukeyHSD(t)
```

Q3

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
s=read.csv("Heights_Gender.csv",header=TRUE)
t=aov(Heights ~ Gender, s)
summary(t)
TukeyHSD(t)
```

Q4

```
x=c(85,95,100,80,90,97,104,95,88,92,94,99,83,85,96,92,100,104,94,95,88,
90,93,94)
group=c(rep("G1",12),rep("G2",12))
t=aov(x ~ group)
```

```
summary(t)
TukeyHSD(t)
```

Q5

```
data(mtcars)
mtcars$cyl <- as.factor(mtcars$cyl)
t<-aov(wt~cyl, mtcars)
summary(t)
TukeyHSD(t)
```

P5

Q1

```
install.packages("dplyr")
install.packages("ggplot2")
install.packages("ggfortify")

library("ggplot2")
library("dplyr")
library("ggfortify")
head(iris)
data <- select(iris, c(1:4))
kmean <- kmeans(data, centers=2, nstart = 50)
kmean$centers
autoplot(kmean, data, frame=TRUE)
```

Q1

```
install.packages("factoextra")
library(factoextra)
fviz_nbclust(data,kmeans,method = "wss")
fviz_nbclust(data,kmeans,method="elbow")
#fviz_nbclust(data,kmeans,method="silhouette")
```

P6

Linear

```
#Linear regression

install.packages("caTools")
library(caTools)
head(mtcars)
a<- sample.split(mtcars$mpg,SplitRatio=0.7)
training_data <- mtcars[a,]
```

```
testing_data <- mtcars[!a,]
#testing_data
dim(training_data)
dim(testing_data)
plot(mpg~drat, data=mtcars)
Model=lm(mpg~drat, data=training_data)
t=summary(Model)
t
plot(mpg~drat, col="blue",cex=1.3,pch=16,data=training_data)
abline(Model)
#calculate MSE
mean(t$residuals^2)
Test=predict(Model,newdata=testing_data)
testing_data$Test=Test
View(testing_data)
result=predict(Model,data.frame(drat=3.90))
print(result)
```

Multiple

```
#multiple regression

library(caret)
data(mtcars)
head(mtcars)
in_train <- createDataPartition(y=mtcars$mpg,p=0.7,list=FALSE)
training_data <- mtcars[in_train,]
testing_data <- mtcars[-in_train,]
Model=lm(mpg~cyl+disp,data=trainig_data)
model_summ<-summary(Model)
model_summ
#calculate MSE
mean(model_summ$residuals^2)
Test=predict(Model,newdata=testing_data)
testing_data$Test=Test
View(testing_data)
new<-data.frame(cyl=c(6),disp=c(160))
results=predict(Model,newdata=new)
print(results)
```

P7

```
install.packages("caTools")
install.packages("dplyr")
install.packages("rpart")
install.packages("rpart.plot")

library(caTools)
```

```
library(dplyr)
library(rpart)
library(rpart.plot)
library(caret)
```

```
setwd("D:\\TYCS\\B1\\DS_16\\DataSets")
getwd()
x=read.csv("titanic.csv",header=TRUE)
newdata<-na.omit(x)
data=select(newdata,-Cabin)
sample_data=sample.split(data,SplitRatio=0.7)
train_data<-subset(data,sample_data==TRUE)
test_data<-subset(data,sample_data==FALSE)
Model<-glm(Survived~PassengerId+Pclass+Age+SibSp+Parch,data=train_data)
summary(Model)
```

```
predicted<-predict(Model,newdata=test_data)
test_data$predicted=ifelse(predicted>0.5,1,0)
actual=factor(test_data$Survived)
predicted=factor(test_data$predicted)
confusionMatrix(predicted,actual,mode='everything')
```

```
rtree <-
rpart(Survived~PassengerId+Pclass+Age+SibSp+Parch,data=train_data)
y=predict(rtree,newdata=test_data)
test_data$y=ifelse(y>0.5,1,0)
rpart.plot(rtree,main="Decision Tree for Titanic Dataset")
a=factor(test_data$Survived)
p=factor(test_data$y)
confusionMatrix(p,a,mode='everything')
```

P8

```
library(dplyr)
View(iris)
mydata=select(iris,c(1,2,3,4))
#cor(mydata)
mean(cor(mydata))
PCA=princomp(mydata)
summary(PCA)
PCA$loadings
PC=PCA$scores
cor(PC)
```

```
install.packages("factoextra")
library(factoextra)
```

```
get_eig(PCA)
fviz_eig(PCA,addlabels = TRUE)
fviz_pca_var(PCA,col.var="contrib")
fviz_pca_biplot(PCA,col.ind="Blue",geom="point")+labs(title="PCA",x="PC
1",y="PC2")
```

P9

```
data(iris)
dim(iris)
head(iris)
str(iris)
summary(iris)

counts<-table(iris$Species)
counts
barplot(counts)
plot(iris$Petal.Length,iris$Petal.Width,main="Iris Data")
plot(iris$Petal.Length,iris$Petal.Width,main="Iris
Data",pch=21,bg=c("red","green","blue")[unclass(iris$Species)])
legend("topleft",legend=levels(iris$Species),col=("red","green","blue"),
pch=20)
```

```
data<-iris[1:4]
quartiles<-quantile(data$Sepal.Width,probs=c(.25,.75),na.rm=FALSE)
IQR<-IQR(iris$Sepal.Width)
Lower<-quartiles[1] - 1.5*IQR
Upper<-quartiles[2] + 1.5*IQR
data_no_outlier <- subset(data,data$Sepal.Width >
Lower&data$Sepal.Width<Upper)
boxplot(data_no_outlier,c(1,2,3,4))
```

```
boxplot(iris[1:4],notch=T,col=c("red","blue","yellow","pink"))
boxplot(iris[,1]~iris[,5],notch=T,xlab="Species",ylab="Sepal
Length",col="pink")
pie(table(iris$Species),col=rainbow(3),main="Species",radius=0.9)
```

```
plot(density(iris$Sepal.Length),col="blue",lwd=2)
```

```
hist(iris$Sepal.Length,prob=T,col="green",breaks=20,xlim=c(3,9),xlab="S
epal Length")
```

```
set<-iris$Sepal.Width[iris$Species=="setosa"]
ver<-iris$Sepal.Width[iris$Species=="versicolor"]
vir<-iris$Sepal.Width[iris$Species=="virginica"]
```

```
par(mfrow=c(3,1))
iris_breaks<-seq(1.8,4.6,0.1)
iris_ylim<-c(0,12)
iris_xlab<-"Sepal width (cm)"
hist(set,
     breaks=iris_breaks,
     ylim=iris_ylim,
     xlab = iris_xlab,
     main="setosa")
```

```
hist(ver,
     breaks=iris_breaks,
     ylim=iris_ylim,
     xlab = iris_xlab,
     main="versicolor")
```

```
hist(vir,
     breaks=iris_breaks,
     ylim=iris_ylim,
     xlab = iris_xlab,
     main="virginica")
```

```
aveg=aply(iris[,1:4],MARGIN=2,FUN=mean)
barplot(aveg,ylab="Average")
```

```
install.packages("lattice")
library(lattice)
levelplot(Petal.Width~Sepal.Length*Sepal.Width,iris,cuts=9,col.regions=
rainbow(10:1))
```