Scienrios 3

```
data<-read.table("videodata.txt", header=TRUE)
head(data)
summary(data)
data[data == 99]<- NA
sum(is.na(data))
summary(data$time)
hist(data$time)
### Bootstrap ####
time.mean <- mean(data$time)
time.mean
boot.population <- rep(data$time,length.out=314)
length(boot.population)
sample1<-sample(boot.population,size = 91, replace = FALSE)
set.seed(189289)
B = 300
boot.sample <- array(dim = c(B, 91))
for (i in 1:B) {
  boot.sample[i, ] <- sample(boot.population, size = 91, replace = FALSE)
}
boot.mean <- apply(X = boot.sample, MARGIN = 1, FUN = mean)
head(boot.mean)
hist(boot.mean, breaks = 22, probability = TRUE, density = 40, col = 4, border = 4)
lines(density(boot.mean, adjust = 2), col = 2)
### Kolmogorov-Smirnov test ###
par(pty = 's')
qqnorm(boot.mean)
qqline(boot.mean)
ks.test((boot.mean - mean(boot.mean))/sd(boot.mean), pnorm)
### Confidence interval ###
boot.sd<-sd(boot.mean)
time.mean + c(-1,1)*1.96*boot.sd
boot.sd
# method 2
int.boot <- c(quantile(boot.mean, 0.025), quantile(boot.mean, 0.975))
Int.boot
### kurtosis ###
install.packages('moments')
library(moments)
### n = 2000 ###
time1 <- data$time
kurtosis(time1)
```

```
normal_kurtosis= NULL
for (i in 1:2000)
  normal_kurtosis[i]=kurtosis(rnorm(91))
hist(normal_kurtosis)
mean(normal_kurtosis)
### n = 4000 ###
time1 <- data$time
kurtosis(time1)
normal_kurtosis= NULL
for (i in 1:4000)
  normal_kurtosis[i]=kurtosis(rnorm(91))
hist(normal_kurtosis)
mean(normal_kurtosis)
##############
```

S2:

```
title: "case study 2"
author: "barryd"
date: "2018年1月31日"
output: html_document
---
```

loading data

```{r}
setwd("C:/Users/barry/Desktop/Math 189")
data <- read.table("videodata.txt", header=TRUE)
head(data)
```

attach data and get arrays of hours played a week before the survey and the array of reported frequency.Then get rid of value of 99 to clean the data

```{r}
attach(data)
time <- time[time != 99]
freq <- freq[freq != 99]
```

plot bar plot(instead of histogram because of discrete data) for time and freq, since in freq 4 means the least frequent and 1 means the most frequent, reverse the scale of horizontal axis

```{r}
counts1 <-table(time)
barplot(counts1,main = "time of playing video games a week before the survey")
counts2 <-table(freq)
barplot(rev(counts2),main = "frequency of playing video
games",names.arg=c("semesterly","monthly","weekly","daily"),ylim = c(0,50))

irreg.index <- which(data$freq == 99)
reg.ind <- setdiff(rownames(data), irreg.index)
data <- data[reg.ind,]
boxplot(time~freq,data, xlab = "frequency of play", ylab = "time of play a week before the survey",names=c("daily","weekly","monthly","semesterly"))

daily.ind <- which(data['freq'] == 1)
data.daily <- data[daily.ind,]
hist(data.daily$time,freq = F,breaks = c(0,5,10,15,20,25,30),ylim = c(0,0.3))
weekly.ind <- which(data['freq'] == 2)
data.weekly <- data[weekly.ind,]
hist(data.weekly$time,freq = F,breaks = c(0,5,10,15,20,25,30),ylim = c(0,0.3))
monthly.ind <- which(data['freq'] == 3)
data.monthly <- data[monthly.ind,]
hist(data.monthly$time,freq = F,breaks = c(0,5,10,15,20,25,30),ylim = c(0,0.3))
semesterly.ind <- which(data['freq'] == 4)
data.semesterly <- data[semesterly.ind,]
hist(data.semesterly$time,freq = F, breaks = c(0,5,10,15,20,25,30),ylim = c(0,0.3))
```

S4:

```{r}
```

```
attach(data)
attitude <- like[like != 1 & like != 99]
count.attitude <- table(attitude)
barplot(count.attitude,main = "attitude towards playing video games",names.arg = c("very
much","somewhat","not really","not at all"))

```
```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['like'] != 99 & data['like'] != 1)
data <- data[reg.ind,]
male.ind <- which(data['sex'] == 1)
data.male <- data[male.ind,]
female.ind <- which(data['sex']== 0)
data.female <- data[female.ind,]
male.count <- table(data.male$like)
female.count <- table(data.female$like)
barplot(male.count,main = "male attitude towards playing video games",names.arg = c("very
much","somewhat","not really","not at all"))
barplot(female.count,main = "female attitude towards playing video games",names.arg = c("very
much","somewhat","not really","not at all"),ylim = c(0,25))
```
```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['like'] != 99 & data['like'] != 1)
data <- data[reg.ind,]
own.ind <- which(data['own'] == 1)
data.own <- data[own.ind,]
notown.ind <- which(data['own']== 0)
data.notown <- data[notown.ind,]
own.count <- table(data.own$like)
notown.count <- table(data.notown$like)
barplot(own.count,main = "PC owners attitude towards playing video games",names.arg =
c("very much","somewhat","not really","not at all"))
barplot(notown.count,main = "non PC owners attitude towards playing video games",names.arg
= c("very much","somewhat","not really","not at all"),ylim = c(0,30))
```
```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['like'] != 99 & data['like'] != 1 & data['educ'] != 99)
data <- data[reg.ind,]
educ.count <- round(prop.table(table(data$educ,data$like),1),2)
barplot(educ.count,main ="barplot for educational&attitute towardsplaying video games", xlab =
```

"Preference", ylab = "Percantage", col = c("red", "blue"),beside = T, names.arg = c("very much","somewhat","not really"))

```

```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['like'] != 99 & data['like'] != 1)
data <- data[reg.ind,]
home.count <- round(prop.table(table(data$home,data$like),1),2)
barplot(home.count,main ="barplot for home&attitute towardsplaying video games", xlab = "Preference", ylab = "Percantage",  col = c("red", "blue"),beside = T, names.arg = c("very much","somewhat","not really","not really"),ylim = c(0,0.6))
legend("topleft",c("PC","no PC"),cex=1.3,bty = "n",fill = c("red","blue"))
```

```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['like'] != 99 & data['like'] != 1)
data <- data[reg.ind,]
data['age.category'] <- rep(NA, dim(data)[1])
for(i in 1:dim(data)[1]){
  age <- data[i, 'age']
  if(age < 20){
    data[i, 'age.category'] = 0
  }else{
    data[i, 'age.category'] = 1
  }
}
age.count <- round(prop.table(table(data$age.category,data$like),1),2)
barplot(age.count,main ="barplot for age & attitute towardsplaying video games", xlab = "Preference", ylab = "Percantage",  col = c("red", "blue"),beside = T, names.arg = c("very much","somewhat","not really","not really"),ylim = c(0,0.6))
legend("topleft",c("< 20",">= 20"),cex=1.3,bty = "n",fill = c("red","blue"))
```
```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['like'] != 99 & data['like'] != 1 & data['educ'] != 99)
data <- data[reg.ind,]
data['age.category'] <- rep(NA, dim(data)[1])
for(i in 1:dim(data)[1]){
  age <- data[i, 'age']
  if(age < 20){
```
```

```
    data[i, 'age.category'] = 0
  }else{
    data[i, 'age.category'] = 1
  }
}
data['dis_like'] <- rep(NA, dim(data)[1])
for(i in 1:dim(data)[1]){
  like <- data[i, 'like']
  if(like==4 || like==5){
    data[i, 'dis_like'] = 0
  }else{
    data[i, 'dis_like'] = 1
  }
}
install.packages("tree")
library(tree)
data.tree <- tree(dis_like~educ+sex+age.category+home+own, data=data)
plot(data.tree, type="uniform")
text(data.tree)
```
```{r}
data <- read.table("videodata.txt", header=TRUE)
reg.ind <- which(data['freq'] != 99)
data <- data[reg.ind,]
data['freq_group'] <- rep(NA, dim(data)[1])
for(i in 1:dim(data)[1]){
  freq <- data[i, 'freq']
  if(freq < 3){
    data[i, 'freq_group'] = 1
  }else{
    data[i, 'freq_group'] = 2
  }
}
freqgroup.count <- round(prop.table(table(data$freq_group,data$grade),1),2)
barplot(freqgroup.count,main ="barplot for frequence of playing video games & grades", xlab =
"grade", ylab = "frequency of playing videogames ",  col = c("red", "blue"),beside = T, names.arg
= c("C","B","A"),ylim = c(0,0.7))
legend("topleft",c("daily or weekly","monthlu or semesterly"),cex=1.3,bty = "n",fill =
c("red","blue"))
```

```
(su ZHANG)
setwd("~/Desktop")
data <- read.table("videodata.txt", header = TRUE)
data$grade
data['grade']
attach(data)
grade
i <- 91
while(i < 95){
  grade[i+1] = 0
  i = i +1
  print(grade[i])
  }
grade[92]
length(grade)
count <- table(grade)
barplot(count,main = "expected grade distribution",names.arg=c("D","C","B","A"),ylim = c(0,60),
col = c(1,4,2,3), xlab = "grade expected", ylab = "number of people")
mean.grade <- mean(grade)
mean.grade


new_vector <- c(31/91,0.2,31/95,52/91,0.3,52/95,8/91,0.4,8/95,0,0.1,5/95)
barplot(new_vector,col = c(rgb(0.3,0.1,0.4,0.6) , rgb(0.3,0.5,0.4,0.6),
rgb=(3)),xlab="grade",ylab="grade assigned percentage",ylim = c(0,1),main = "expected grade
distribution",names.arg=c("A","A*","A", "B","B*","B", "C","C*","C",
"D","D*","D"),space=c(0,0,0,0.3,0,0,0.3,0,0,0.3,0,0))
legend("topleft", legend = c("old expected grade","target grade","new expected grade" ) ,
       col = c(rgb(0.3,0.1,0.4,0.6) , rgb(0.3,0.5,0.4,0.6),rgb=(3)) ,
```

```
      bty = "n", pch=20 , pt.cex = 2, cex = 0.8, horiz = FALSE, inset = c(0.05, 0.05))

data <- read.table("videodata.txt", header = TRUE)
grade4.ind <- which(data['grade']==4)
data.grade4 <- data[grade4.ind,]
grade3.ind <- which(data['grade']==3)
data.grade3 <- data[grade3.ind,]
grade2.ind <- which(data['grade']==2)
data.grade2 <- data[grade2.ind,]
grade1.ind <- which(data['grade']==1)
data.grade1 <- data[grade1.ind,]
new_vector <- c(31/91,0.2,52/91,0.3,8/91,0.4,0,0.1)
barplot(new_vector,col = c(rgb(0.3,0.1,0.4,0.6) , rgb(0.3,0.5,0.4,0.6)),xlab="grade",ylab="grade
assigned percentage",ylim = c(0,1),main = "expected grade
distribution",names.arg=c("A","A*","B","B*","C","C*","D","D*"),space=c(0.2,0,0.2,0,0.2,0,0.2,0))
legend("topleft", legend = c("expected grade","target grade" ) ,
      col = c(rgb(0.3,0.1,0.4,0.6) , rgb(0.3,0.5,0.4,0.6)) ,
      bty = "n", pch=20 , pt.cex = 2, cex = 0.8, horiz = FALSE, inset = c(0.05, 0.05))
summary(grade)
grade
target2 <- array(1:95)
i <- 0
while(i < 95*0.2){
  target2[i+1] = 4
  i = i + 1

}
while(i < 95*0.5){
  target2[i+1] =3
  i = i+1
}
while(i < 95*0.9){
  target2[i+1]=2
  i = i +1

}
while(i < 95){
  target2[i+1] =1
  i = i +1
}
target1 <- array(1:91)
i <- 0
while(i < 91*0.2){
```

```
  target1[i+1] = 4
  i = i + 1

}
while(i < 91*0.5){
  target1[i+1] =3
  i = i+1
}
while(i < 91*0.9){
  target1[i+1]=2
  i = i +1

}
while(i < 91){
  target1[i+1] =1
  i = i +1
}
ks.test(grade, target2)
ks.test(data$grade, target1)
mean(target1)
mean(target2)
```