```
videodata <- read.csv("~/Math 189/videodata.txt", sep="")

##################### Data
summary(videodata)
videodata[videodata == 99] <- NA

##################### Investigation - Scenario 6
grades <- sort(videodata$grade)
table(grades)

# d&f (0&1) - 0
# c (2) - 8/91
# b (3) - 52/91
# a (4) - 31/91

# target distb - 18 A's, 27 B's, 36 C's, and 9 D/F's

letters <- c("A","A","B","B","C","C","D & F","D & F")
values <- c(31,18,52,27,8,36,0,9)
type <- c("Expected","Target","Expected","Target","Expected","Target","Expected","Target")

my_grades <- cbind(letters,values,type)
my_grades <- as.data.frame(my_grades)
my_grades$values = as.double(levels(my_grades$values))[my_grades$values]

ggplot(my_grades, aes(x = letters, y = values, fill = type)) +
  geom_bar(stat="identity",width=.5, position = "dodge") +
  guides(fill=guide_legend(title="")) +
  labs(x = "Letter Grade",
       y = "Density")

##########
new_grades <- c(grades,1,1,1,1)
table(new_grades)

letters <- c("A","A","B","B","C","C","D & F","D & F")
values <- c(31,18,52,27,8,36,4,9)
type <- c("Expected","Target","Expected","Target","Expected","Target","Expected","Target")
my_grades <- cbind(letters,values,type)
my_grades <- as.data.frame(my_grades)
my_grades$values = as.double(levels(my_grades$values))[my_grades$values]

ggplot(my_grades, aes(x = letters, y = values, fill = type)) +
```

```
  geom_bar(stat="identity",width=.5, position = "dodge") +
  guides(fill=guide_legend(title="")) +
  labs(x = "Letter Grade",
       y = "Density")


# target distb - 18 A's, 27 B's, 36 C's, and 9 D/F's

target_grades <- c(1,1,1,1,1,1,1,1,1,
                   2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,
                   3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,
                   4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4,4)
table(target_grades)

ks.test((videodata$grade - mean(videodata$grade))/sd(videodata$grade),
        (target_grades - mean(target_grades))/sd(target_grades))
#4.235e-07

new_grades <- c(grades,1,1,1,1)
table(new_grades)

ks.test((new_grades - mean(new_grades))/sd(new_grades),
        (target_grades - mean(target_grades))/sd(target_grades))
#4.96e-06

###################### Investigation - Scenario 3

ggplot(videodata, aes(x = time)) +
  geom_bar(width = 1, fill = "mediumorchid4") +
  labs(x = "Time spent playing vido games (hours/week)",
       y = "Density")

time.percentage <- mean(videodata$time)
time.percentage
#1.24

boot.population <- rep(videodata$time, length.out = 314)
length(boot.population)
#314

sample1 <- sample(boot.population, size = 91, replace = FALSE)

set.seed(189289)
```

```
B = 400 # the number of bootstrap samples we want
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 = 20, probability = TRUE, density = 20, col = 3, border = 3)
lines(density(boot.mean, adjust = 2), col = 2)

quantile(boot.mean,0.025)
quantile(boot.mean,0.975)

par(pty = 's')
qqnorm(boot.mean)
qqline(boot.mean)

bm <- as.data.frame(boot.mean)

ggplot(bm, aes(x=boot.mean)) +
  geom_histogram(binwidth = 0.1,fill = "mediumorchid4") +
  labs(x = "Time spent playing vido games (hours/week)",
       y = "Density")


library(moments)

kurtosis(boot.mean)
skewness(boot.mean)

normal_kurtosis_time=NULL
for(i in 1:1000){
  normal_kurtosis_time[i]=kurtosis(rnorm(nrow(videodata)))
}

hist(normal_kurtosis_time)

#Histogram
ggplot() + aes(normal_kurtosis_time) +
  geom_histogram(binwidth=0.2, colour="black", fill="mediumorchid4") +
  labs(title="Simulation of Normal Kurtosis",
```

```
      y = "Frequency",
      x = "Normal Kurtosis: Time spent playing video games")

#######
normal_skewness_time=NULL
for(i in 1:1000){
  normal_skewness_time[i]=skewness(rnorm(nrow(videodata)))
}

hist(normal_skewness_time)

#Histogram
ggplot() + aes(normal_skewness_time) +
  geom_histogram(binwidth=0.1, colour="black", fill="mediumorchid4") +
  labs(title="Simulation of Normal Skewness",
      y = "Frequency",
      x = "Normal Skewness: Time spent playing video games")


####################

#Decision Tree
videodata <- read.csv("~/Desktop/videodata.txt", sep="")
videodata[videodata == 99] <- NA

#to fit decision tree, make 'like' to binary 'dis_like'
data$dis_like<- rep(NA, dim(data)[1])
for(i in 1:dim(data)[1]){
+like <- data[i, 'like']
+ if(like==0 || like==4 || like==5){
+ data[i, 'dis_like'] = 0
+ }else{
+ data[i, 'dis_like'] = 1
+ }
+ }
data.tree <- tree(dis_like~educ+sex+age+home+math+work+own+cdrom+grade, data=data)
plot(data.tree, type="uniform")
text(data.tree)
```

```
Cross-Tabulations
data <- read.csv("~/Downloads/videodata.txt", sep="")

data$like[data$like == 99] <- NA
data <- na.omit(data)

data["dislike"] <- rep(NA, dim(data)[1])

for(i in 1:dim(data)[1]){
  like_temp <- data[i, 'like']
  if(like_temp == 1 || like_temp == 4 || like_temp == 5){
    data[i, 'dislike'] = 1
  }else{
    data[i, 'dislike'] = 0
  }
}
chisq.test(data$sex == 0, data$dislike == 1)$obs
chisq.test(data$work == 0, data$dislike == 1)$obs
chisq.test(data$own == 0, data$dislike == 1)$obs

############################
#Scenario 2

videogames <- read.csv("~/Desktop/videogames.csv", sep="")

#Histogram 4.2.1
a <- videogames$time > 0
time <- videogames[a,]
hist(time$freq, main='Students who Spent Time Playing Video Games', xlab = 'Time Played', col
= 'blue')

#Histogram 4.2.2
c <- videogames$time == 0
no.time <- videogames[c,]
hist(no.time$freq, main = 'Students who Spent No Time Playing Video Games', xlab = 'Time
Played', col = 'blue')

#Boxplot 4.2.3
boxplot(time$time, main = 'Students who Played', xlab = 'Students', ylab = 'Time Played', col =
'blue')

#Histogram 4.2.5.
```

```
hist(videogames$busy, main = 'Busy Students Vs. Not Busy Students',xlab="Student
Busyness", col = 'blue')

#Boxplot 4.2.6.
busy <- videogames$busy == 1
busy<-videogames[busy,]
boxplot(busy$freq, main ='Busy Students and their Frequencies', xlab='Busy Students', ylab
='Frequency')

#Boxplot 4.2.7.
nonbusy <- videogames$busy == 0
nonbusy<-videogames[nonbusy,]
boxplot(nonbusy$freq, main ='Nonbusy Students and their Frequencies', xlab='Nonbusy
Students', ylab ='Frequency', col='blue')

#Code to generate analysis of additional question
df_sales = pd.read_csv("vgsales.csv")
df = df_sales.drop(['EU_Sales','JP_Sales', 'Other_Sales', 'Global_Sales'], axis=1)

#Table 4.5.4
df.head(11)

#Figure 4.6.3
Genre=list(df.Genre.unique())
NA_Sales = []
for i in genre:
   value=df[df.Genre==i]
   x=value.NA_Sales.mean()
   NA_Sales.append(x)
df2 = pd.DataFrame({"Genre":Genre,"NA_Sales":NA_Sales})
df2.sort_values("NA_Sales",ascending=False,inplace=True)
plt.figure(figsize=(12,7))
sns.barplot(x="Genre", y="NA_Sales", data=df2)
plt.xticks(rotation= 30)
plt.xlabel("Genre", fontsize=16)
plt.ylabel("NA_Sales", fontsize=16)
plt.title("Sales based on Genre", fontsize=18)
plt.show()
```

```
data <- read.csv("C:/Users/taq19/Downloads/videodata.txt", sep="")

video <- data

video$time[video$time > 0] <- 1


N <- 314
n <- 91

mean.sample <- mean(video$time) #point estimate

confwidth <- 2*(sd(video$time)/(sqrt(n)))
confwidth

conf.sample <- c(mean.sample - confwidth, mean.sample + confwidth)
conf.sample

width <- 1.96 * sqrt(mean.sample*(1-mean.sample)*(N-n)/((n-1)*N))
int.sample <- c(mean.sample - width, mean.sample + width)
int.sample

bootobject= NULL
for ( i in 1:400)
{
  bootobject[i]=mean(sample(as.vector(video$time),size=91,replace=TRUE))
}

ggplot() + aes(bootobject) +
  geom_histogram(bins = 20, colour="black", fill="mediumorchid4") +
  labs(title="Histogram of Bootstrap Mean",
       y = "Frequency",
       x = "Ratio of students who play video games")
hist(bootobject)

s <- sd(bootobject)
int.boot <- c(mean.sample - 1.96*s, mean.sample + 1.96*s)
int.boot

quantile(bootobject, 0.025)
quantile(bootobject, 0.975)
```

```
int.boot <- c(quantile(bootobject, 0.025), quantile(bootobject, 0.975))
int.boot




require(e1071)
kurtosis(bootobject)
skewness(bootobject)

kurtosis_=NULL
for (i in 1:1000)
{
  kurtosis_[i]=kurtosis(rnorm(400))
}

ggplot() + aes(kurtosis_) +
  geom_histogram(binwidth=0.1, colour="black", fill="mediumorchid4") +
  labs(title="Simulation of Normal Kurtosis",
       y = "Frequency",
       x = "Kurtosis of 400 bootstrapped samples")



mean(kurtosis_)



skewness_=NULL
for (i in 1:1000)
{
  skewness_[i]=skewness(rnorm(400))
}
ggplot() + aes(skewness_) +
  geom_histogram(binwidth=0.1, colour="black", fill="mediumorchid4") +
  labs(title="Simulation of Normal Skewness",
       y = "Frequency",
       x = "Skewness of 400 bootstrapped samples")
```