# Final Exam 2023(Long Answer)

Student Full Name

2023-04-22

Question 1: Suppose you have a dataset called "students" containing information on the age, gender, and test scores of 100 students.

(a) Create a dataframe with three columns: age (22, 20, 19, 19, 18, 20, 17), gender (Female, Male, Male, Female, Male, Female, Female), and test scores (7 sample data between 40 to 100) using the sample function and set.seed(25).

(b) Calculate the mean and standard deviation of the test scores.

(c) Create a boxplot of the test scores.

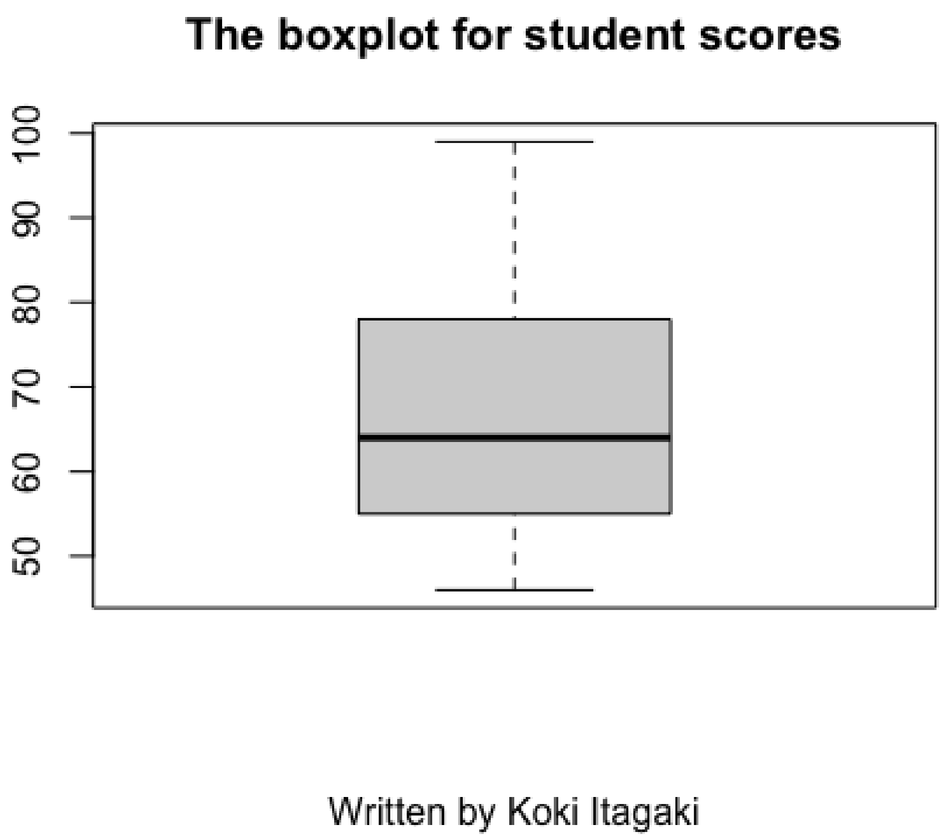
(d) Calculate the correlation between age and test scores.

(e) Create a scatter plot to show the relationship between age and test scores, where the latter is the response variable in the dataframe.

(f) Add a regression line to your scatter plot.

```
# Answer (a) below:
set.seed(25)
test_scores<-sample(40:100,7,replace = TRUE)
test_scores

## [1] 46 68 63 99 64 88 47

students <- data.frame(age = c(22, 20, 19, 19, 18, 20, 17),gender = c(
  "Female", "Male", "Male", "Female", "Male", "Female", "Female"),
  testScores = test_scores)
students

##   age gender testScores
## 1  22 Female         46
## 2  20   Male         68
## 3  19   Male         63
## 4  19 Female         99
## 5  18   Male         64
## 6  20 Female         88
## 7  17 Female         47

# Answer (b) below:
mean_score = mean(students$testScores)
mean_score
```

```
## [1] 67.85714

sd_score = sd(students$testScores)
sd_score

## [1] 19.69288

# Answer (c) below:
boxplot(students$testScores,main = "The boxplot for student scores",sub =
          "Written by Koki Itagaki")
```




```
# Answer (d) below:
cor<-cor(students$age,students$testScores)
cor

## [1] 0.001507941

# Answer (e) below:

plot(x = students$age, y = students$testScores,main = "The scatter plot of
age and scores",
     xlab = "age of students", ylab = "test scores of students")
```

```
# Answer (f) below:

abline(lm(students$testScores~students$age),col = "red")
```


Question 2: Download, save, and read in the file “insurance.csv” from Brightspace. This dataset contains the age, the gender, body mass index of the person who has purchased the insurance policy, the number of children/dependents the insured person has, and the amount charged for the insurance policy, which is the response variable in this dataset.

(a) Create 3 plots illustrating the relationship between the response variable and and each explanatory variables.

(b) Fit a linear regression model including all of the explanatory variables. Be sure to write out the regression equation.

(c) Determine which variable(s) (if any) are not significant in the model using 0.05 as the criteria.

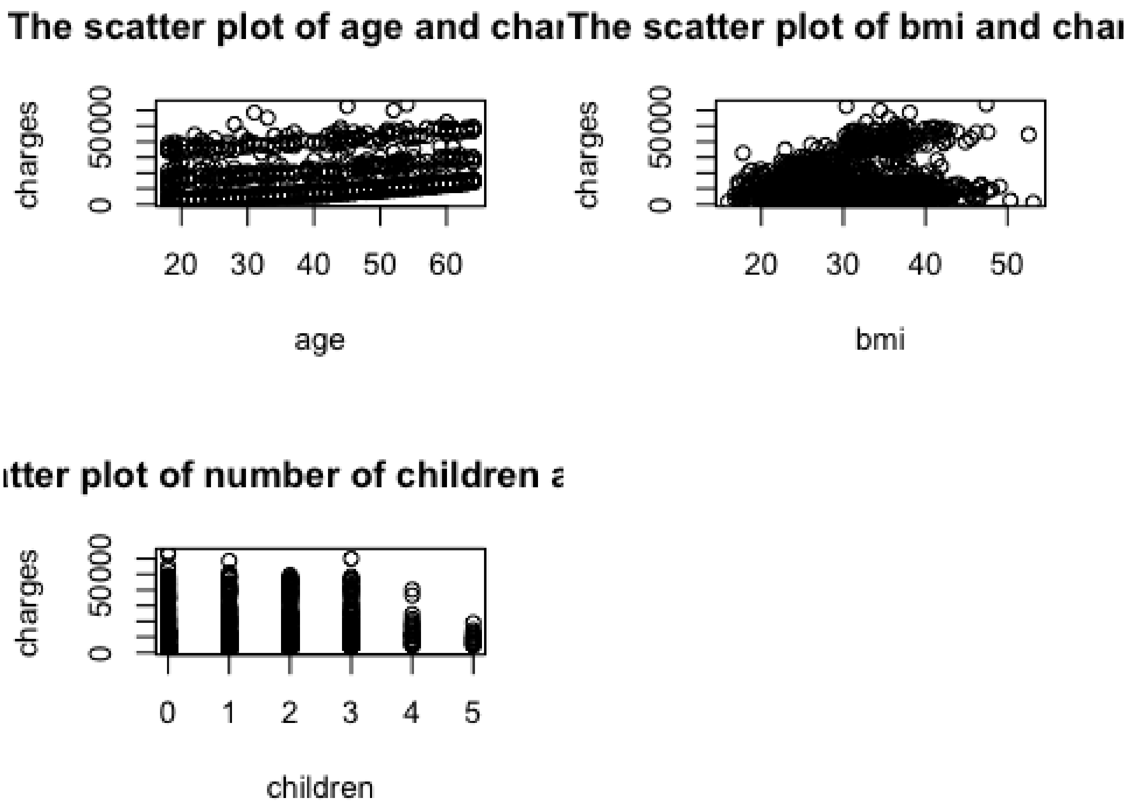
(d) Using the model from part (c) to predict the amount of charges for a female aged 30 with 1 child, a bmi of 34.2.

```
insurance= read.csv("/Users/itagakikouki/stat123/insurance.csv")
age<-insurance$age
bmi<-insurance$bmi
children<-insurance$children
charges<-insurance$charges
# Answer (a) below:
par(mfrow = c(2, 2))
plot(x=age, y = charges,main = "The scatter plot of age and charges")
plot(x=bmi, y = charges,main = "The scatter plot of bmi and charges")
plot(x=children, y = charges,main = "The scatter plot of number of children
and charges")
```

```
# Answer (b) below:
lm_charge<-lm(charges~age+bmi+children)
lm_charge

##
## Call:
## lm(formula = charges ~ age + bmi + children)
##
## Coefficients:
## (Intercept)          age          bmi     children
##     -6916.2        240.0        332.1        542.9

#The regression equation is y = b0 + b1x1 + b2x2 + b3x3
#Now we know the numbers of b0,b1,b2,and b3.
#So, the regression equation is now y = -6916.2 + 240.0*x1 + 332.1*x2 +
542.9*x3
```

```
# Answer (c) below:

summary(lm_charge)

##
## Call:
## lm(formula = charges ~ age + bmi + children)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -13884  -6994  -5092   7125  48627
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -6916.24    1757.48  -3.935 8.74e-05 ***
## age           239.99      22.29  10.767  < 2e-16 ***
```

```
## bmi           332.08      51.31   6.472 1.35e-10 ***
## children      542.86     258.24   2.102   0.0357 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11370 on 1334 degrees of freedom
## Multiple R-squared:  0.1201, Adjusted R-squared:  0.1181
## F-statistic: 60.69 on 3 and 1334 DF,  p-value: < 2.2e-16

#All of the variables are significant which means the p-values are less
#than 0.05, so I do not to remove any variables.




# Answer (d) below:

#y = -6916.2 + 240.0*x1 + 332.1*x2 + 542.9*x3
#When the data is a female aged 30 with 1 child, a bmi of 34.2.

y = -6916.2 + 240.0*30 + 332.1*34.2 + 542.9*1
cat(paste("The charges for  a female aged 30 with 1 child, a bmi of 34.2 is",
y))

## The charges for  a female aged 30 with 1 child, a bmi of 34.2 is 12184.52
```




Question 3: Consider the gapminder dataset (available by either loading into the R session or reading in the .csv file available in Brightspace).

(a) Create a variable called Europe_1957 which contains all of the rows of the gapminder data set corresponding to the continent Europe in the year 1957 You may subset the data in any way that you please.

(b) Plot the distribution of the life expectancy in European countries in 1957. You do not need any titles for your plot.

(c) Describe the shape of the distribution (symmetry, skewness, etc.).

(d) What is the best measure of the centre of the distribution? Compute this value.

(e) What is the best measure of the spread of the distribution? Compute the value(s).

(f) Suppose we are interested in a statistic that takes the minimum life expectancy value + the maximum life expectancy value and then divides that sum by 2. We will call this statistic “midpoint”. Compute the observed value of the midpoint statistic for the sample of European life expectancies in 1957.

(g) Bootstrap 10000 sample midpoints of European life expectancies in 1957. Save the bootstrapped vector as boot_midpoint.

** Note ** If you are unable to bootstrap this particular statistic, then bootstrap the median instead in order to be able to answer the remainder of the question.

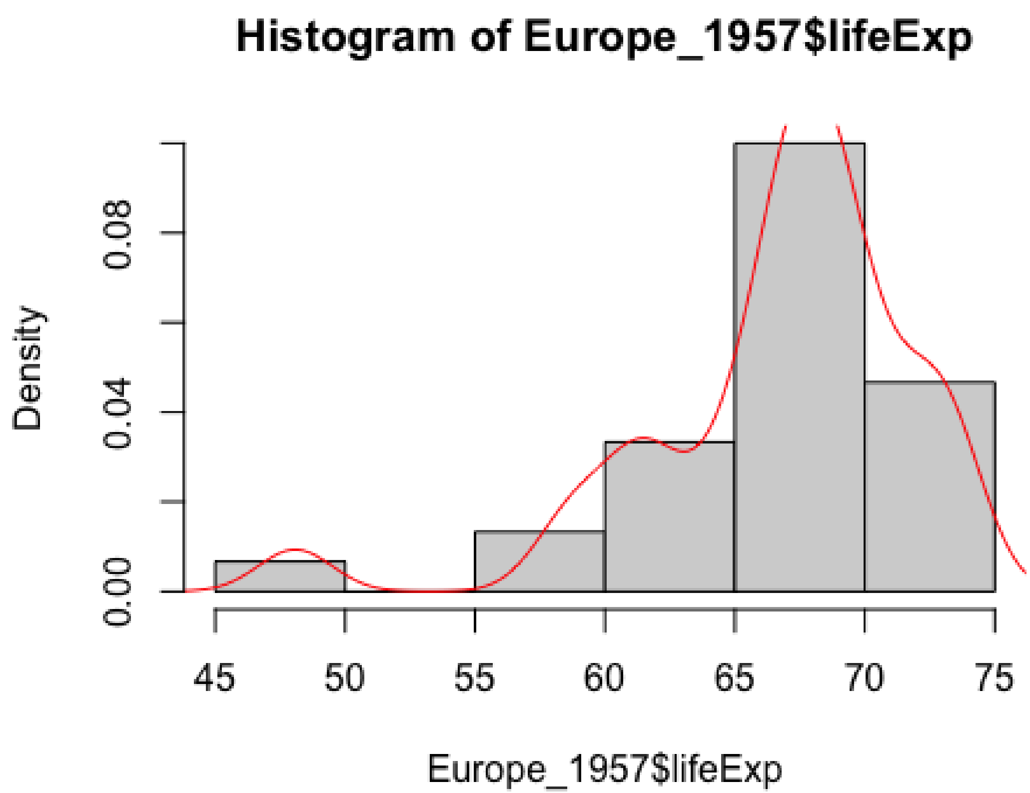
(h) Plot the distribution of the bootstrapped midpoints. You do not need any titles for your plot.

(i) Describe the shape of the distribution. Does it appear normally distributed?

```
library(dplyr,ggplot2)

## 
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union

gapminderr<- read.csv("/Users/itagakikouki/stat123/gapminder.csv")

# Answer (a) below:
Europe_1957 = gapminderr%>%filter(continent == "Europe", year == 1957)
Europe_1957

##       X                country continent year lifeExp      pop gdpPercap
## 1    14                Albania    Europe 1957  59.280  1476505  1942.284
## 2    74                Austria    Europe 1957  67.480  6965860  8842.598
## 3   110                Belgium    Europe 1957  69.240  8989111  9714.961
## 4   146 Bosnia and Herzegovina    Europe 1957  58.450  3076000  1353.989
## 5   182               Bulgaria    Europe 1957  66.610  7651254  3008.671
## 6   374                Croatia    Europe 1957  64.770  3991242  4338.232
## 7   398         Czech Republic    Europe 1957  69.030  9513758  8256.344
## 8   410                Denmark    Europe 1957  71.810  4487831 11099.659
## 9   518                Finland    Europe 1957  67.490  4324000  7545.415
## 10  530                 France    Europe 1957  68.930 44310863  8662.835
## 11  566                Germany    Europe 1957  69.100 71019069 10187.827
## 12  590                 Greece    Europe 1957  67.860  8096218  4916.300
## 13  674                Hungary    Europe 1957  66.410  9839000  6040.180
## 14  686                Iceland    Europe 1957  73.470   165110  9244.001
## 15  746                Ireland    Europe 1957  68.900  2878220  5599.078
## 16  770                  Italy    Europe 1957  67.810 49182000  6248.656
## 17 1010             Montenegro    Europe 1957  61.448   442829  3682.260
## 18 1082            Netherlands    Europe 1957  72.990 11026383 11276.193
## 19 1142                 Norway    Europe 1957  73.440  3491938 11653.973
## 20 1226                 Poland    Europe 1957  65.770 28235346  4734.253
## 21 1238               Portugal    Europe 1957  61.510  8817650  3774.572
## 22 1274                Romania    Europe 1957  64.100 17829327  3943.370
## 23 1334                 Serbia    Europe 1957  61.685  7271135  4981.091
```

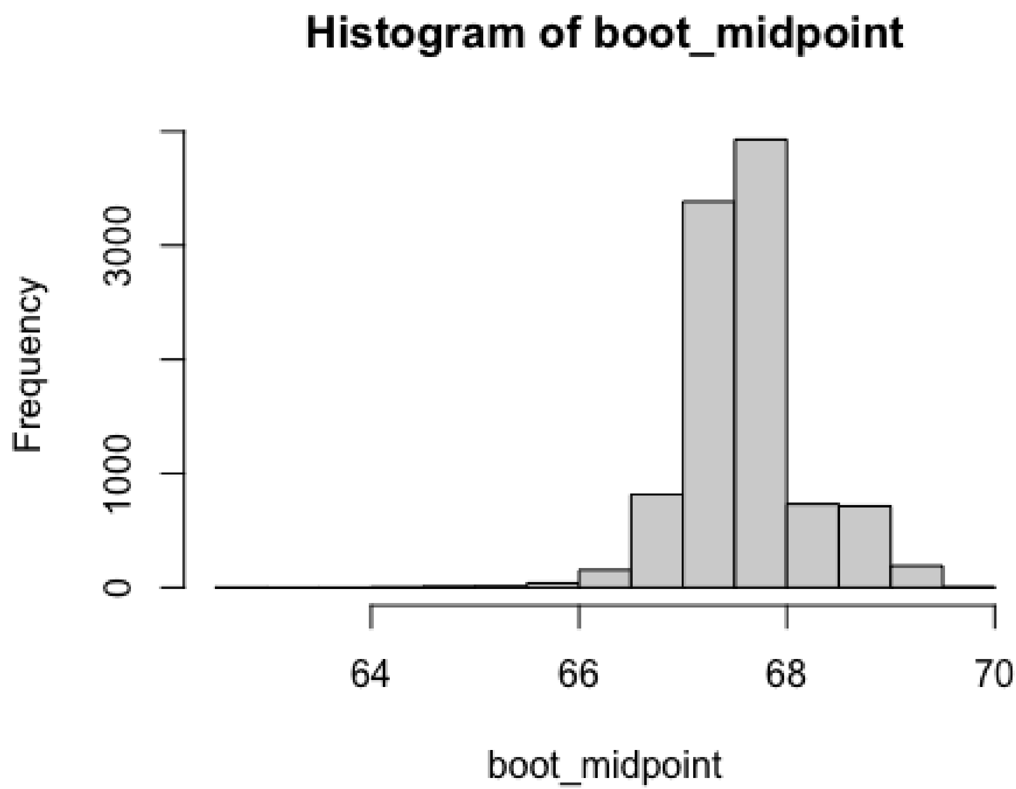
```
## 24 1370        Slovak Republic    Europe 1957  67.450  3844277  6093.263
## 25 1382               Slovenia    Europe 1957  67.850  1533070  5862.277
## 26 1418                  Spain    Europe 1957  66.660 29841614  4564.802
## 27 1466                 Sweden    Europe 1957  72.490  7363802  9911.878
## 28 1478            Switzerland    Europe 1957  70.560  5126000 17909.490
## 29 1574                 Turkey    Europe 1957  48.079 25670939  2218.754
## 30 1598         United Kingdom    Europe 1957  70.420 51430000 11283.178
```

```
# Answer (b) below:
hist(Europe_1957$lifeExp,prob = TRUE)
lines(density(Europe_1957$lifeExp),col = "red")
```




```
# Answer (c) below:

#The distribution is left-skewed




# Answer (d) below:
#Since the distribution is not symmentric and there might be a outlier around
#45 to 50, we should use median to describe centre of the distribution.
median(Europe_1957$lifeExp)
```

```
## [1] 67.65

# Answer (e) below:
#Since the distribution might not be normally distributed, I should use 
quantile function
#to describe the spread of the distribution
quantile(Europe_1957$lifeExp)

##     0%    25%    50%    75%   100% 
## 48.079 65.020 67.650 69.205 73.470

# Answer (f) below:
midpoint<-function(x){
  (min(x)+max(x))/2
}

round(midpoint(Europe_1957$lifeExp),2)

## [1] 60.77

# Answer (g) below:z
n = length(Europe_1957$lifeExp)
boot_midpoint = numeric()
for(i in 1:10000){
  temp_sample= sample(Europe_1957$lifeExp,n,replace = TRUE)
  boot_midpoint[i] = median(temp_sample)
}




# Answer (h) below:
hist(boot_midpoint)
```




```
# Answer (i) below:
#The distribution looks a little bit left skewed, so I do not think I can say
#this is perfectly normally distributed.
```

Question 4: The built-in Titanic data set is a 4-dimensional array that contains the following information:

• Dimension 1: Class of the passenger (1 = 1st, 2 = 2nd, 3 = 3rd, 4 = Crew member) • Dimension 2: Sex of the passenger (1 = male, 2 = female) • Dimension 3: Age of the passenger (1 = child, 2 = adult) • Dimension 4: Survival of the passenger (1 = died, 2 = survived)

(a) Create (and print out) a table which contains the adult passengers (of all classes and genders) who survived.

(b) Create (and print out) a vector called survived which contains all adult passengers (of all classes and genders) who survived.

(c) Create a barplot displaying the survived vector. Make sure to include a main title and to label your x-axis. Also, make sure that each bar is a different colour.

(d) Create (and print out) a vector called died which contains the adult passengers who did not survive.

(e) Create (and print out) a vector called percent.Survived which contains the percentage of adult passengers who survived in each class, Using the sum(survived) in part (b).

(f) Create a pie chart that displays the percent.Survived data. Be sure to include a main title for your pie chart.

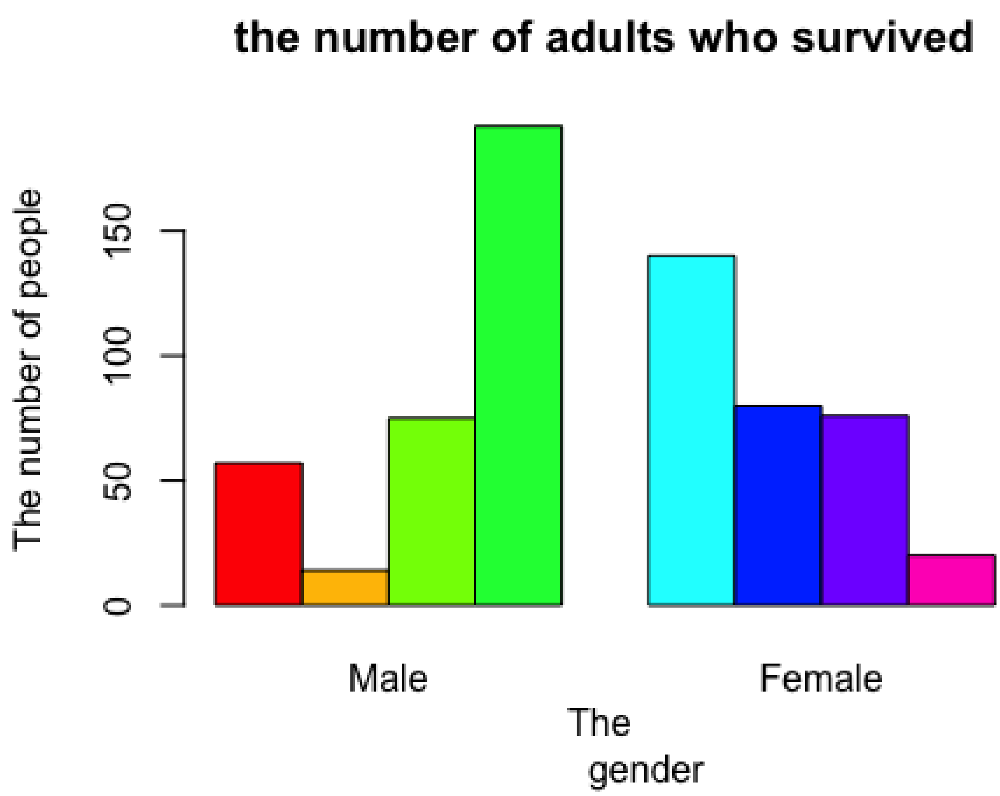
(g) Estimate the proportion of the female passengers (of all classes and ages) who survived using the table created in part (a).

(h) Determine a 90% confidence interval for the proportion estimated in part (g) (round to 3 decimal places)

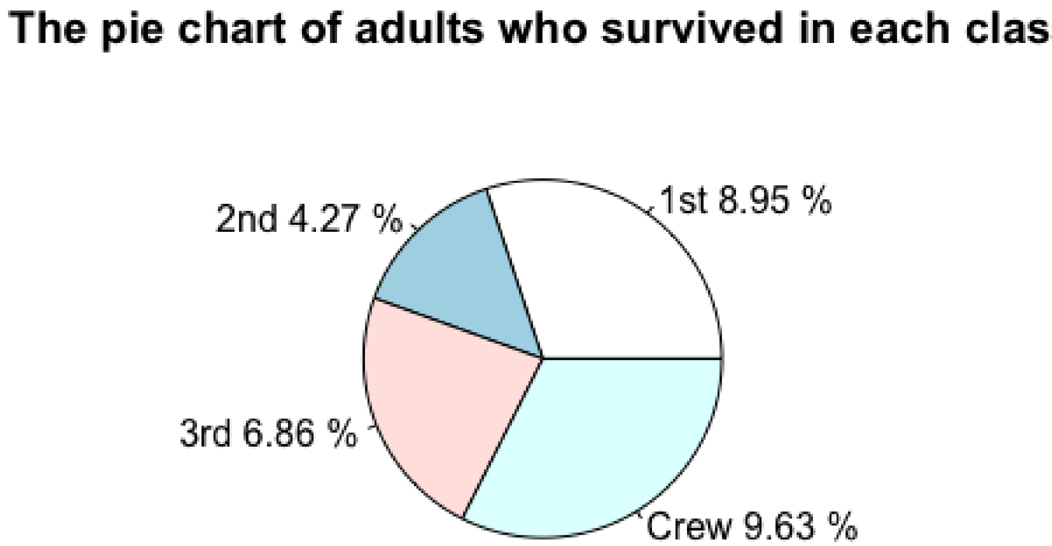
(i) Compute the margin of error.

```
head(Titanic)

## , , Age = Child, Survived = No
##
##       Sex
## Class  Male Female
##   1st     0      0
##   2nd     0      0
##   3rd    35     17
##   Crew    0      0
##
## , , Age = Adult, Survived = No
##
##       Sex
## Class  Male Female
##   1st   118      4
##   2nd   154     13
##   3rd   387     89
##   Crew  670      3
##
## , , Age = Child, Survived = Yes
##
##       Sex
## Class  Male Female
##   1st     5      1
##   2nd    11     13
##   3rd    13     14
##   Crew    0      0
##
## , , Age = Adult, Survived = Yes
##
##       Sex
## Class  Male Female
##   1st    57    140
##   2nd    14     80
```

```
##    3rd    75      76
##    Crew  192      20

# Answer (a) below:
adult_survived<-table(Titanic[,,"Adult","Yes"])
adult_survived

##
##  14   20   57   75   76   80  140  192
##   1    1    1    1    1    1    1    1

# Answer (b) below:
survived<-Titanic[,,"Adult","Yes"]




# Answer (c) below:
barplot(survived,main = "the number of adults who survived",xlab = "The
        gender",ylab = "The number of people"
        ,col = rainbow(length(survived)),beside  = TRUE)
```



```r
# Answer (d) below:
died<-Titanic[,,"Adult","No"]
died

##       Sex
## Class  Male Female
##   1st   118      4
##   2nd   154     13
##   3rd   387     89
##   Crew  670      3

# Answer (e) below:


#I use for loop to put the proportion of survived adults out of all
passengers
#who dead or survived and to show it in each class, i created a list.
percent.Survived<-numeric()
for(i in 1:4){
  percent.Survived[i]<-round(sum(survived[i,])/sum(Titanic)*100,2)

}
classes<-c("1st", "2nd","3rd","Crew")
percent.Survived

## [1] 8.95 4.27 6.86 9.63

for(i in 1:4){
cat("The percentage of survived adults in", classes[i], "is",
percent.Survived[i])
}

## The percentage of survived adults in 1st is 8.95The percentage of survived
adults in 2nd is 4.27The percentage of survived adults in 3rd is 6.86The
percentage of survived adults in Crew is 9.63

# Answer (f) below:



pie(percent.Survived,main = "The pie chart of adults who survived in each
class",
    labels = paste(classes,percent.Survived,"%"))
```




```
# Answer (g) below:
female_survived<-Titanic[,"Female",,"Yes"]/sum(Titanic)
```

```
# Answer (h) below:
n = length(female_survived)
p = mean(female_survived)
p

## [1] 0.01953657

sd = sqrt(p*(1-p)/n)
sd

## [1] 0.04893222

#Since this is the sample size is small I will use quantile function
q = quantile(p,0.95)
q

##        95%
## 0.01953657
```

```r
upper = round(p + q*sd,3)
lower = round(p -q*sd,3)
cat(paste("The 90% confidence interval is ",lower, ",", upper))

## The 90% confidence interval is  0.019 , 0.02

# Answer (i) below:
sd = sqrt(p*(1-p)/n)
sd

## [1] 0.04893222

#Since this is the sample size is small I will use quantile function

q = quantile(p,0.95)
q

##        95%
## 0.01953657

#The margin of the error is
moe<-q*sd
```