Statistical Data Analysis
Assignment 4

Matt Kuodis & James Zoryk.
2640428 : 2663347
Group: 36
Due Date: 05.04.2022.

Question 4.1.

Part a.

In this section we shall consider the data from 'birthweight.txt" which contains data about birth weights in grams. We shall begin by analyzing the distribution of the birth weight data, which is achieved by generating a numerical analysis (Table 1) of the data along with a histogram (Figure 2).

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
709	2414	2977	2945	3475	4990

Table 1: Numerical summary of 'birthweights.txt'.

In Table 1, we observe that the median is less than the mean, however this difference is negligible as it is small in comparison with he data size. Therefore we can say that the data is not skewed.

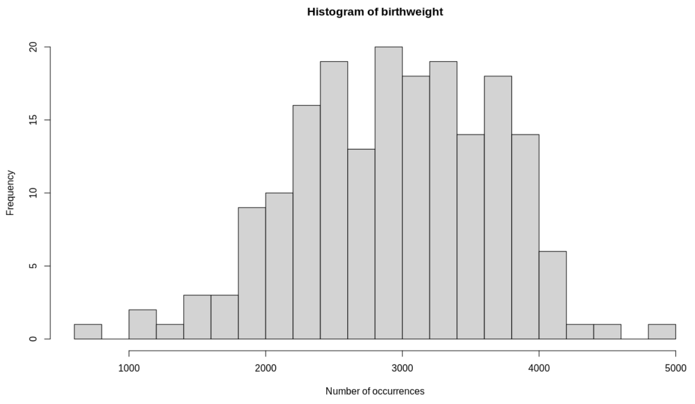

Figure 1: Histogram plots of 'birthweights.txt'

From the histogram we can see that the shape of the data is quite similar to that of the normal distribution. This is because the data is very symmetrical for both tails, with a central mass peak. Furthermore, we also consider that the data originates for a natural measurement which typically has a normal distribution, especially with a large sample which we have in this case. In order to test this hypothesis we shall compare the data with a QQ-plot, we also plot other Location Scale Family distributions as well.

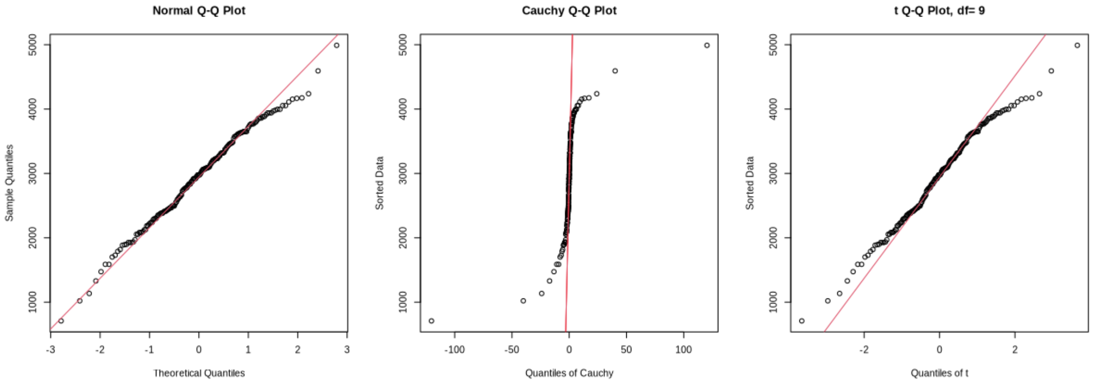

Figure 2: QQplot of 'birthweights.txt'

As we can see from the QQ-plot figure, the normal distribution gives nearly all of the data points along the straight line $y = x$. Hence there is evidence to suggest that the data comes from a normal distribution. We finish this section by computing the location and scale parameters for our data which is $\mathcal{N}(2945, 729)$.

Part b.

We can estimate the 10% - quantile of the birth weights by using the the R-function `quantile` and setting the parameter to 0.1. Computing this we have the estimate as xx. Now, for the estimate of the median absolute deviation "MAD" of the 10% quantile estimator, we use the empirical bootstrap method. Doing so, we obtain the following results (Table 2)

10% - quantile	MAD bootstrap
2038	110.90

Table 2: 10% - quantile of the data and the 'MAD' value of the bootstrap.

The reason for choosing the empirical bootstrap is that we do not need to assume more information about the data distribution. We are always guaranteed a reasonable result when we are conducting the empirical bootstrap. Here we should note that from part [a] there is strong evidence that the data originates from the normal distribution location scale family. However, if we are incorrect with this hypothesis then the whole bootstrap estimation would be computed with an error, undermining the accuracy of our analysis of the data. Therefore, we choose the side of caution and use the empirical bootstrap to mitigate potential errors.

Part c.

In this section we conduct a parametric bootstrap with the assumption that our sample data originated from the exponential distribution. The purpose of this is to compare the results with part [b]. Computing the parametric bootstrap yields the following results (Table 3).

MAD parameter bootstrap (exp):	72.67

Table 3: parameter bootstrap exponential

Here we see that there is a large difference between the ‘MAD’ value of the empirical bootstrap and the parametric exponential. This is due to using the wrong parametric to estimate the the distribution, which leads to the large difference we see in the ‘MAD’ valves. Therefore the empirical bootstrap is the safer option.

Part d.

Here we are tasked with writing the bootstrap estimate in one line of R-studio code. This is achieved by the line of code on line 45 of ”R code for exercise 4.1” in the Appendix.

Question 4.2.

Part a.

For this tasked we followed the method stated in the course syllabus. Namely, that we use a large enough sample $B = 1000$ for our bootstrap estimator. In this specific case we are interested in estimating the location of the distribution, as in doing so we can estimate the mean and the median by means of a 90% bootstrap confidence interval. This leads us to obtain the following results We also get the following histogram

Statistic (PRRP)	Value
Mean	61.56
ci Mean	53.57, 69.72

Table 4: The mean and confidence interval for PRRP.

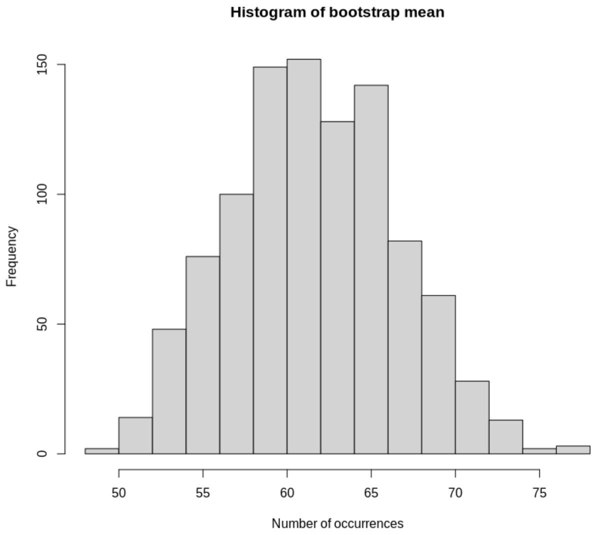

Figure 3: Histogram plots of the mean bootstrap

Part b.

We repeat our work as we have done in part 2.a but this time we compute the median instead of the mean which leads to the following numerical results: With which we also obtain the histogram:

Part c.

In order to decide which statistic is better suited to the given data, we will use the the span of the confidence interval as metric. From Table 4 we compute the span for

Statistic (PRRP)	Value
Median	54.25
ci Median	44.5, 68.5

Table 5: The median and confidence interval for PRRP.

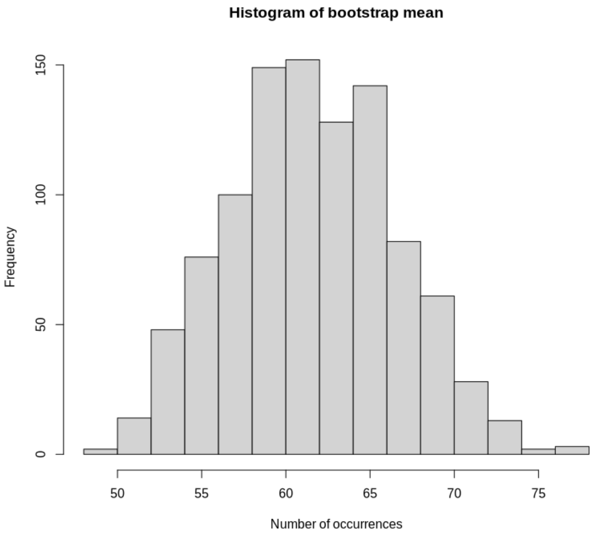

Figure 4: Histogram plots of the median bootstrap

the mean as $69.72 - 53.57 = 16.15$. Then from Table 5 we compute the span for the median as $68.5 - 44.5 = 24$. Therefore, by this metric we see that the mean statics is the better one to use since the span is smaller than that of the median.

Part d.

For this task we are interested in getting a confidence interval of the difference of the two samples 'PRRP' and 'SDRP'. Noting that these samples have difference lengths. In order to compute this we follow the modified test. Therefore, with a 90% confidence level we have the following results Table 6 and Figure 6.

mean(RPPP) -mean(SDRP)	-0.3643
ci mean diff	-0.2517, 0.2524

Table 6: Difference of the two samples 'PRRP' and 'SDRP' and confidence intervals.

From Figure 6 we see that the histogram is rather symmetrical and centred roughly around zero. This is also indicated in the data in Table 6, we compute the span metric as $0.2524 + 0.2517 = 0.5041$. Consider the present data and the fact the sample size is small, there is evidence to suggest that there is no difference between means of the two samples.

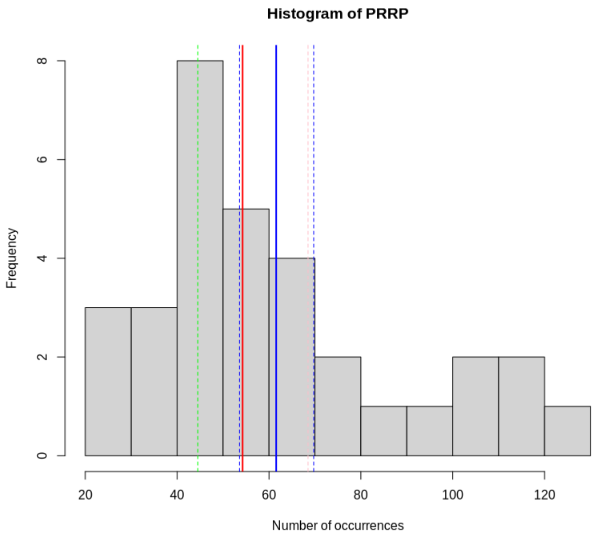

Figure 5: blue = mean, red = median with the dashed lines belonging to mean (dashed red/blue) median (green/pink)

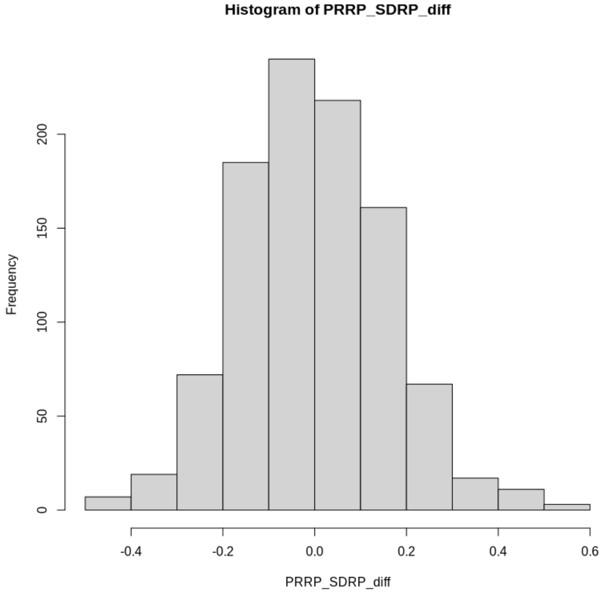

Figure 6: Histogram of the difference between between PRRP and SDRP

Question 4.3.

Part a.

We note that the Kolmogorov-Smirnov statistic is distribution free, as it is generally given by computing the largest deviation between the empirical distribution of a given data set, and the empirical distribution of a hypothetical parametric distribution of some F_0, which in our case is $\mathcal{N}(\texttt{mean}(Data), \texttt{sd}(Data))$. The statistic is therefore distribution free, as it compares discrepancies between the accumulation of observed and expected data.

The adjustments made to the test statistic D_n are simply that we have normalized our hypothetical sample data, but since we are comparing cumulative empirical distributions, it does not matter which parameters are being compared, our interest lies simply in $\tilde{D}_n$ which quantifies how well the spread of observed data fits to the expected distribution.

Since the cumulative distribution simply shows the rate by which a data sample grows, the distribution, location, and scale will be irrelevant, and thus the adjusted Kolmogorov-Smirnov statistic, in exactly the same way as the standard one, will be distribution free.

Part b.

We first take a look at our regular data, using the histogram plots. This gives us the following graphs.

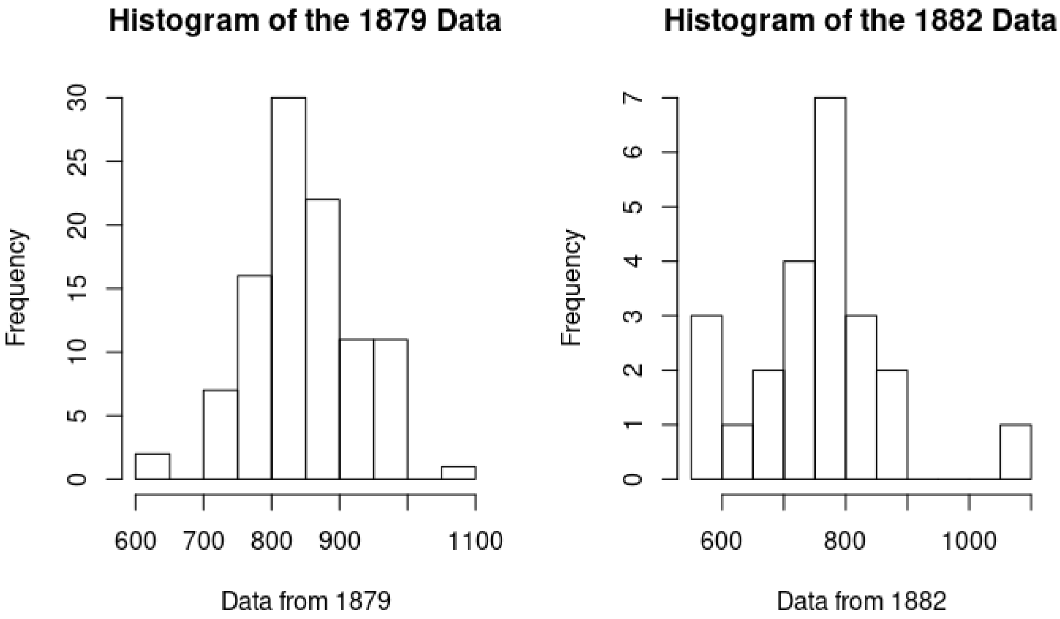

Figure 7: Plots of the data sets from 1879 and 1882.

We use the built in `ks.test` for the adjusted statistic of the data sets, and find that $\tilde{D}_{1879} = 0.083424$ with p-value $= 0.4896$, and $\tilde{D}_{1882} = 0.14533$ with p-value $= 0.7162$. In either case, the p-values strongly support the claim for normality of the data.

Now we perform the bootstrap on $\tilde{D}_n$ for the two data sets, with $B = 1000$, and look at the data plots for the generated bootstrap samples $\tilde{D}_n^*$, which give us the following histograms.

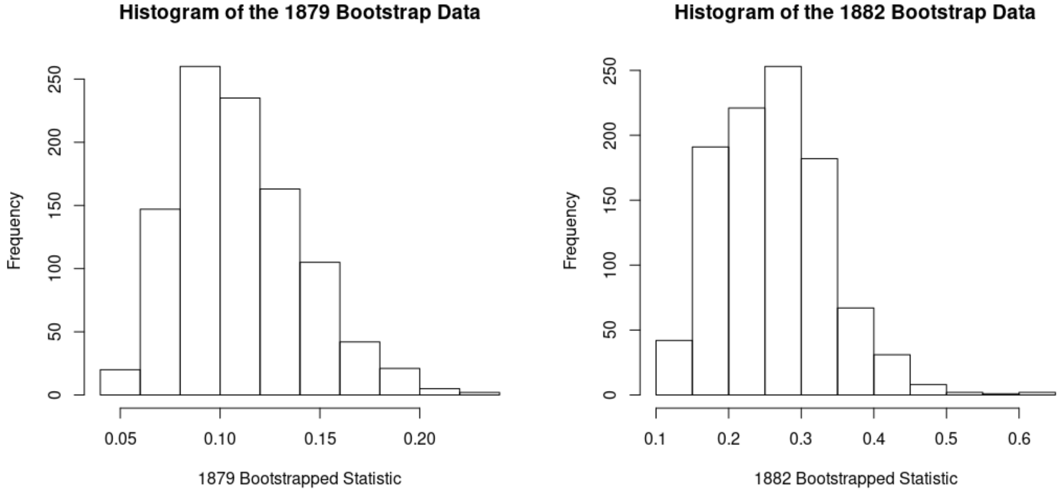

Figure 8: Plots of the bootstrapped samples $\tilde{D}_{1879}$ and $\tilde{D}_{1882}$.

Using these samples we can compute the p-values that our statistics $\tilde{D}_{1879}$ and $\tilde{D}_{1882}$ are reasonable values for our test statistic. We do this through the command line `1-ecdf("Bootstrap Statistic")("Regular Statistic")` which computes the probability of more extreme p-values than those we computed as our regular statistics. Doing the same for the bootstrapped data from 1882, we get the following p-values, namely $\tilde{p}^*$-value$_{1879} = 0.806$ and $\tilde{p}^*$-value$_{1879} = 0.962$ which strongly supports the claim for normality, yet again.

Part c.

The differences in obtained p-values, which are far higher for the bootstrapped samples, might be explained by the fact that the data sets themselves are rather spread out. We can see in Figure 1 that the sample from 1879 has a few data points at the tail ends, while the sample from 1882 seems to have even more extreme values at the tails, and a substantial gap towards the right side. As the bootstrapped samples take fewer data points, they likely also sample less of these outliers, and thus the bootstrapped data begins to resemble a normal distribution far quicker than the actual data might. This would explain the large p-values for the bootstrapped samples that support normality, while the p-values of the actual data, though large enough to also support the claim for normality, are far less certain.

R code for exercise 4.1

```
###################################################
# SDA HW4
# Exercise: 1
###################################################
# Pull data from disk
setwd("~/Maths/SDA/Assignment_4")
#read file
source("functions_Ch3.txt")
source("functions_Ch5.txt")



birthweight <- t(read.table("birthweight.txt"))
# get data summary
summary(birthweight)

hist(birthweight)

hist(birthweight, prob = TRUE, col = "grey",
     ylim = c(0, 0.001),
     main = c("Histogram with density curve:"),
     xlab= "data: birthwieghts",
)
lines(density(birthweight,),
      col="tomato",
      lwd= 1.0)

abline(v = median(birthweight), col = "red",lwd = 1) # median line
abline(v = mean(birthweight), col = "green",lwd = 1) # mean line
text(x = median(birthweight)+200, y= 15,"Mean" , col="green")
text(x = median(birthweight)-200, y= 20,"Median" , col="red")


meanBS =  sd(bootstrap(birthweight, median, B = 5000))
mu = mean(birthweight)
sdBS <- mean(bootstrap(birthweight, sd, B = 5000))
sigma = sd(birthweight)
meanBS
mu
sdBS
sigma

par(mfrow=c(1,2))
par(pty="s")
qqnorm(birthweight,
       ylab = "Sorted Sqrt Data")
qqline(birthweight,  distribution = qnorm, col="red")
qqchisq(birthweight, df=28)
qqline(birthweight,  distribution = qchisq, col="red")
qqchisq(birthweight, df=3)
qqchisq(birthweight, df=2000)


```

```


# b)
bs = bootstrap(birthweight, quantile , probs=0.1 , B=1000)
hist(bs)
mad(bs)

abline(v = mean(bs), col = ”red”,lwd = 1) # median line
abline(v = median(bs), col = ”green”,lwd = 1) # median line



med = median(birthweight)
med
medBS <- mean(bootstrap(birthweight, median, B = 5000))

sdofthemed = sd(bootstrap(birthweight, median, B = 5000))
meanBS <- bootstrap(birthweight, median, B = 5000)
sdBS <- mean(bootstrap(meanBS, sd, B = 5000))
sdBS
sdofthemed


hist(bootstrap(birthweight, median, B = 5000))

mean(empBS)
sdBS <- bootstrap(empBS, sd, B = 5000)
hist(sdBS)
mean(sdBS)

rexp
meanBS =  mean(bootstrap(birthweight, mean, B = 5000))
lambda = 1 / mean(birthweight)

#
```

Code/sda_hw4_ex1.R

R code for exercise 4.2

```

setwd(”~/Desktop/P4_and_P5_2022/SDA_Assignments/SDA4”)

data <- source(”thromboglobulin.txt”)
source(”functions_Ch3.txt”)
source(”functions_Ch5.txt”)

PRRP_data = thromboglobulin$PRRP

#a.
PRRP_means = bootstrap(PRRP_data, mean, B = 1000)
PRRP_quants = quantile(PRRP_means, probs = c(0.05, 0.95))
PRRP_quants
sqrt(var(PRRP_quants))

```

```r
#b.
PRRP_medians = bootstrap(PRRP_data, median, B = 1000)
PRRP_quants2 = quantile(PRRP_medians, probs = c(0.05, 0.95))
PRRP_quants2
sqrt(var(PRRP_quants2))

#c.
PRRP_mean = mean(PRRP_data)
PRRP_bootstrap_mean = mean(PRRP_means)
PRRP_median = median(PRRP_data)
PRRP_Bootstrap_median = mean(PRRP_medians)

hist(PRRP_data, breaks = 10)
abline(v=c(PRRP_mean, PRRP_median), col=c("blue", "red"), lw = 2)
abline(v=PRRP_quants, col=c("blue", "blue"), lty = 2)
abline(v=PRRP_quants2, col=c("red", "red"), lty = 2)

mean_interval_width = PRRP_quants[2] - PRRP_quants[1]
median_interval_width = PRRP_quants2[2] - PRRP_quants2[1]

#d.
SDRP_data = thromboglobulin$SDRP
iterations = 1000

PRRP_SDRP_diff = numeric(iterations)

for (i in 1:iterations) {
  PRRP_means2 = bootstrap(PRRP_data, mean, B=1000)
  SDRP_means = bootstrap(SDRP_data, mean, B=1000)

  PRRP_SDRP_diff[i] = mean(PRRP_means2) - mean(SDRP_means) - (mean(PRRP_data) - mean(SDRP_means))
}

PRRP_SDRP_diff_quant = quantile(PRRP_SDRP_diff, probs = c(0.05, 0.95))
PRRP_SDRP_diff_quant

hist(PRRP_SDRP_diff)

#Conclusion: The difference is normally distributed
```

Code/SDA_hw4_ex2.R

R code for exercise 4.3

```r

setwd("~/Desktop/P4_and_P5_2022/SDA_Assignments/SDA4")

data <- source("light.txt")
source("functions_Ch3.txt")
source("functions_Ch5.txt")

# a. Because it is normalized so that X_i ~ N(0, 1)

# b.
```

```
data_1879 = light$`1879`
data_1882 = light$`1882`

par(mfcol=c(1,2))
hist(data_1879, breaks = 10, main = "Histogram of the 1879 Data", xlab = "Data from 1879")
hist(data_1882, breaks = 10, main = "Histogram of the 1882 Data", xlab = "Data from 1882")

mean_1879 = mean(data_1879)
sd_1879 = sd(data_1879)
mean_1879
sd_1879

mean_1882 = mean(data_1882)
sd_1882 = sd(data_1882)
mean_1882
sd_1882

# compute D and p-values for the data with the KS test
statistic_1879 = ks.test(data_1879, "pnorm", mean = mean_1879, sd = sd_1879)$statistic
statistic_1882 = ks.test(data_1882, "pnorm", mean = mean_1882, sd = sd_1882)$statistic
ks.test(data_1879, "pnorm", mean = mean_1879, sd = sd_1879)
ks.test(data_1882, "pnorm", mean = mean_1882, sd = sd_1882)

mean_parameter=c(mean_1879, mean_1882)
sd_parameter=c(sd_1882, sd_1882)

par(mfcol=c(1,2))

ks_statistic_test_1879 = function(input_data) {

  D_value_1879 = ks.test(input_data, "pnorm", mean = mean_parameter[1], sd = sd_parameter[1])$statistic
  D_value_1879

}

ks_statistic_test_1882 = function(input_data) {

  D_value_1882 = ks.test(input_data, "pnorm", mean = mean_parameter[2], sd = sd_parameter[2])$statistic
  D_value_1882

}

normal_sample = rnorm(length(data_1879), mean = mean_parameter[1], sd = sd_parameter[1])
bootstrap_statistic_1879 = bootstrap(normal_sample, ks_statistic_test_1879, B=1000)
hist(bootstrap_statistic_1879, main = "Histogram of the 1879 Bootstrap Data", xlab = "1879 Bootstrapped Statistic")
```

```

normal_sample = rnorm(length(data_1882), mean = mean_parameter[2], sd = sd_parameter[2])
bootstrap_statistic_1882 = bootstrap(normal_sample, ks_statistic_test_1882, B=1000)
hist(bootstrap_statistic_1882, main = "Histogram of the 1882 Bootstrap Data", xlab = "1882 Bootstrapped Statistic")

p_value_1879 = 1-ecdf(bootstrap_statistic_1879)(statistic_1879)
p_value_1882 = 1-ecdf(bootstrap_statistic_1882)(statistic_1882)

#c.

# Compare p-values with bootstrap, and note that bootstrap gets pretty nice p-values,
# but ks.test on datasets gave p-values large enough to not reject H_0
# that F is normally distributed
```

Code/SDA_hw4_ex3.R