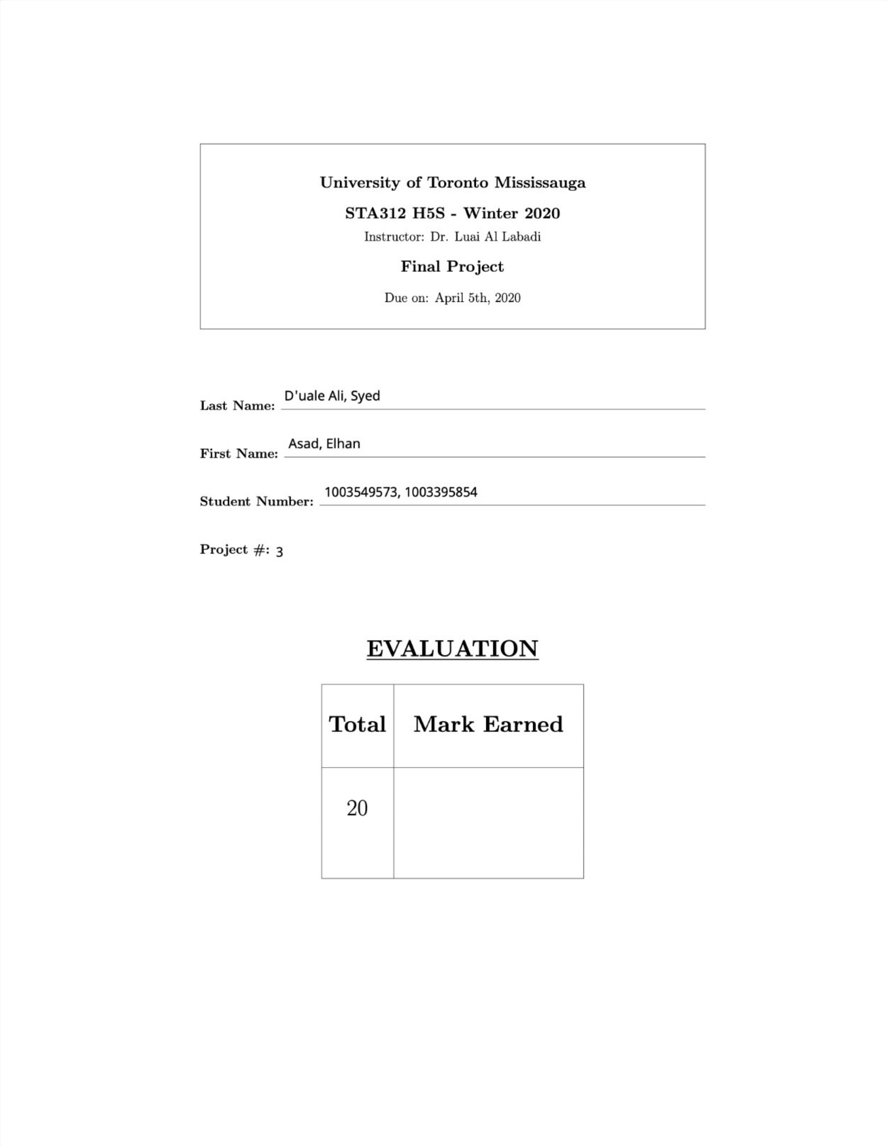


University of Toronto Mississauga

STA312 H5S - Winter 2020

Instructor: Dr. Luai Al Labadi

Final Project

Due on: April 5th, 2020

Last Name: D'uale Ali, Syed

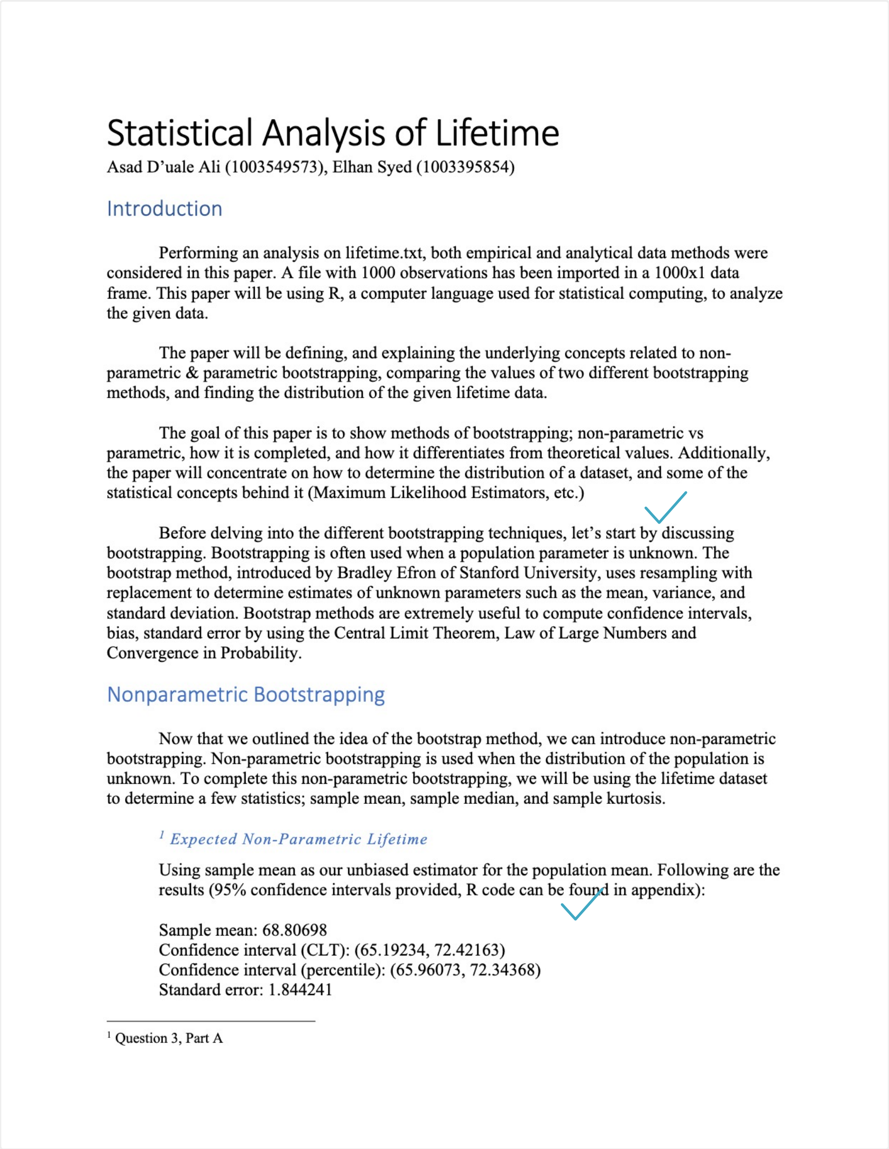
First Name: Asad, Elhan

Student Number: 1003549573, 1003395854

Project #: 3

EVALUATION

Total	Mark Earned
20	



Statistical Analysis of Lifetime

Asad D’uale Ali (1003549573), Elhan Syed (1003395854)

Introduction

Performing an analysis on lifetime.txt, both empirical and analytical data methods were considered in this paper. A file with 1000 observations has been imported in a 1000x1 data frame. This paper will be using R, a computer language used for statistical computing, to analyze the given data.

The paper will be defining, and explaining the underlying concepts related to non-parametric & parametric bootstrapping, comparing the values of two different bootstrapping methods, and finding the distribution of the given lifetime data.

The goal of this paper is to show methods of bootstrapping; non-parametric vs parametric, how it is completed, and how it differentiates from theoretical values. Additionally, the paper will concentrate on how to determine the distribution of a dataset, and some of the statistical concepts behind it (Maximum Likelihood Estimators, etc.) ✓

Before delving into the different bootstrapping techniques, let’s start by discussing bootstrapping. Bootstrapping is often used when a population parameter is unknown. The bootstrap method, introduced by Bradley Efron of Stanford University, uses resampling with replacement to determine estimates of unknown parameters such as the mean, variance, and standard deviation. Bootstrap methods are extremely useful to compute confidence intervals, bias, standard error by using the Central Limit Theorem, Law of Large Numbers and Convergence in Probability.

Nonparametric Bootstrapping

Now that we outlined the idea of the bootstrap method, we can introduce non-parametric bootstrapping. Non-parametric bootstrapping is used when the distribution of the population is unknown. To complete this non-parametric bootstrapping, we will be using the lifetime dataset to determine a few statistics; sample mean, sample median, and sample kurtosis.

[1] *Expected Non-Parametric Lifetime*

Using sample mean as our unbiased estimator for the population mean. Following are the results (95% confidence intervals provided, R code can be found in appendix): ✓

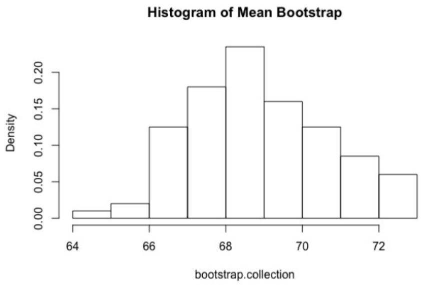
Sample mean: 68.80698
Confidence interval (CLT): (65.19234, 72.42163)
Confidence interval (percentile): (65.96073, 72.34368)
Standard error: 1.844241

[1] Question 3, Part A

These values were achieved by performing a non-parametric bootstrap with 1000 replications. The mean of each replication was taken and stored into a vector. The standard error was then calculated from this, and the confidence intervals were as well. ✓

The expected lifetime was determined to be 65.19234 - 72.42163 using the Central Limit Theorem, and 65.96073 - 72.34368 using percentiles, with a standard error of 1.844241.

Following is a histogram to visualize the bootstrap medians.



[2] *Median Non-Parametric Lifetime*

Using sample median as our unbiased estimator for the population median, we produced the following results (95% confidence intervals provided, R code can be found in appendix):

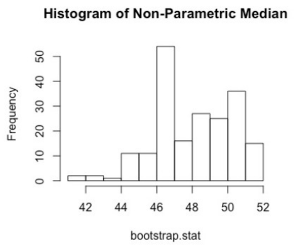
Sample median: 48.41656
Standard error: 2.406337
Confidence interval (percentile): (43.38875, 52.25755)

These values were achieved by performing a non-parametric bootstrap with 1000 replications. The median of each replication was taken and stored into a vector. The standard error was then calculated from this, and the confidence intervals were as well. ✓

The median lifetime was determined to be 43.70022 - 53.13289 using the Central Limit Theorem, and 43.38875 - 52.25755 using percentiles, with a standard error of 1.844241.

[2] Question 3, Part B (Median)

Following is a histogram to visualize the bootstrap medians.

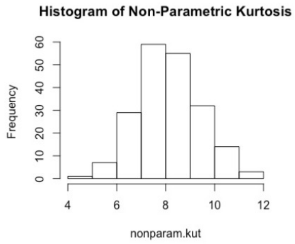


[3] *Kurtosis Non-Parametric Lifetime*

Kurtosis is the fourth standardized moment that is used to measure the distribution of a curve. More specifically, kurtosis gives insights about the size of the tail of the distribution. Kurtosis is mathematically defined as $k_4 = \frac{\mu_4}{\sigma^4}$, however we will be using its estimator, $\hat{k}_4 = \frac{\frac{1}{n}\sum_{i=0}^{n}(X_i - \bar{X})^4}{s^4}$.

The kurtosis of the data was determined to be 8.000926, with a confidence interval of (5.667378, 10.734500).

✓



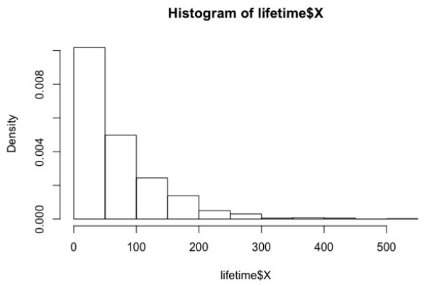
Determining the Distribution

[3] Question 3, Part B (Kurtosis)

Now that we have done some bootstrapping and have used lifetime data to calculate various statistics, let’s attempt to determine its distribution.

[4] *Visualizing the Data*

Following is a histogram showing the distribution of the lifetime dataset:



Just looking at the above histogram, it’s very noticeable that the data follows an exponential distribution.

[5] *Determining the λ value*

While determining the true λ parameter is impossible with only a sample, we will attempt to find $\hat{\lambda}$, the unbiased estimator of λ.

To do so, we can start by finding the MLE (maximum likelihood estimator) of the exponential distribution.

$$f(x) = \lambda e^{-\lambda x}$$

$$\prod_{i=0}^{n} f(x) = \lambda^{n} e^{-\lambda \sum_{j=0}^{n} x_j} = L(\lambda)$$

$$\ln\big(L(\lambda)\big) = nln(\lambda) - \lambda \sum_{j=0}^{n} x_j$$

[4] Question 3, Part C (Verifying data follows exponential)
[5] Question 3, Part C (find $\hat{\lambda}$)

$$0 = \frac{d(\ln(L(\lambda)))}{d\lambda} = \frac{n}{\lambda} - \sum_{j=0}^{n} x_j$$

$$\hat{\lambda} = \frac{1}{\sum_{j=0}^{n} x_j} = \frac{1}{\bar{x}}$$

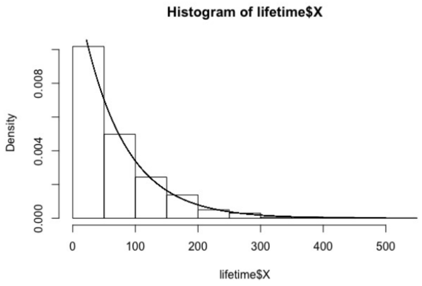
✓

Thus, from this we see that the MLE for λ in an exponential distribution is $\frac{1}{\bar{x}}$.

Using this MLE, we can now determine, through R, the value for $\hat{\lambda}$, which is found to be 0.01453341. This is also found in the appendix.

Is $\hat{\lambda}$ a Good Fit?

Below is a graph of the distribution of the lifetime data, alongside the exponential distribution using the $\hat{\lambda}$ value determined above.



✓

As expected, visibly the line closely follows the trend of the data. To determine the goodness of the fit of the exponential distribution to the lifetime dataset, we can run a K-S test. When ran, the KS test returned a p-value of 0.828 which mean that the null hypothesis that the datasets are sampled from the same distribution cannot be rejected. This ultimately means that exp($\hat{\lambda}$ = 0.01453341) closely resembles the distribution of the lifetime dataset.

Parametric Bootstrapping

Based on the work done above, $\hat{\lambda}$ seems to follow an exponential distribution. With that assumption, parametric bootstrapping can be done. Parametric bootstrapping assumes the data comes from a known distribution with unknown parameters. $\hat{\lambda}$ will be estimating the population

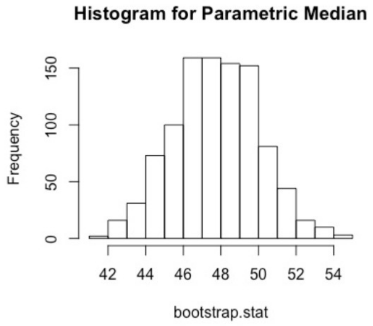
✓

parameter, λ. Estimating the parameters from the data we have, we will now use the estimated distribution to simulate samples.

[6] *Median Parametric Lifetime*

Knowing ƛ = 0.01453341, tests will be run with this assumption that lifetime follows an exponential distribution.

Sample median: 49.31878
Standard error 2.490158
Confidence interval (percentile): (38.76744, 56.60349)



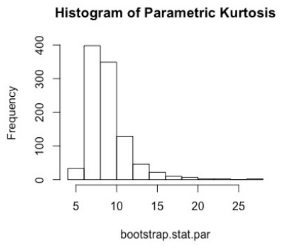
Here it is more noticeable that the median is more normally distributed than it was with non-parametric bootstrapping. These values were achieved by using parametric bootstrapping with a 1000 cases. Each replication was sampled from a random exponential generation with ƛ and stored into a vector. The median of that vector was then made into a confidence interval.

[7] *Kurtosis Parametric Lifetime*

Measuring the "tailed-ness" of a known probability distribution, exponential, the Kurtosis was taken.

✓

[6] Question 3, Part D (Median)
[7] Question 3, Part D (Kurtosis)



The kurtosis of the data was 8.893668 within a confidence interval of (5.91674, 15.52047). The data is more skewed to the right which is similar to the exponential distribution.

[8] Comparison

A quick little comparison between the theoretical and bootstrap values. Let's start by first determining the theoretical value for median. We know the cumulative distribution function of exponential, so we can easily find the median, which is as follows:

$$median = P(X \leq x) = 0.5$$
$$0.5 = 1 - e^{-\lambda x}$$
$$e^{-\lambda x} = 0.5$$
$$x = \frac{-\ln{(0.5)}}{0.01453341} \approx 47.69336 = median$$

✓

Now that we have the theoretical median value, let's find the theoretical kurtosis.

$$k_4 = \frac{E\big(x - E(x)\big)^4}{(E(x - E(X)^2)^2)}$$
$$Start\ by\ only\ simplifying\ the\ numerator{:}\ E\big(x - E(x)\big)^4$$
$$= E(x^4 + 4x^3E(x) + 6x^2E(x)^2 + 4xE(x) + E(x)^4$$
$$= E(x^4) - 4\frac{1}{\lambda}E(x^3) + 6\frac{1}{\lambda^2}E(x^2) - 4\frac{1}{\lambda^3}E(x) + \frac{1}{\lambda^4}$$
$$= \frac{9}{\lambda^4}$$
$$Combining\ the\ numerator\ and\ denominator,$$

[8] Question 3, Part E

✓

$$k_4 = \frac{\frac{9}{\lambda^4}}{(E(x-E(X)^2)^2)} = \frac{\frac{9}{\lambda^4}}{V(x)^2} = \frac{\frac{9}{\lambda^4}}{\lambda^4} = 9$$

We have now determined that the theoretical median is 47.69336 and the theoretical kurtosis is 9. When compared to the median, and kurtosis determined through bootstrapping, we see that they lie very close to their respective theoretical value. Below is a chart representing their absolute difference.

	Theoretical	Bootstrap	Absolute Difference
Median	47.69336	49.31878	1.62542
Kurtosis	9	8.893668	0.106332

Comparing their confidence intervals?

-0.5

Conclusion

From the comparison above, bootstrapping is an incredibly efficient method of estimate statistics on a population. By sampling a sub-dataset with replacement, summary statistics such as mean, median, and confidence intervals can be derived. The kurtosis showed how heavy ended the tails were, since it followed an exponential distribution the number was incredibly high.

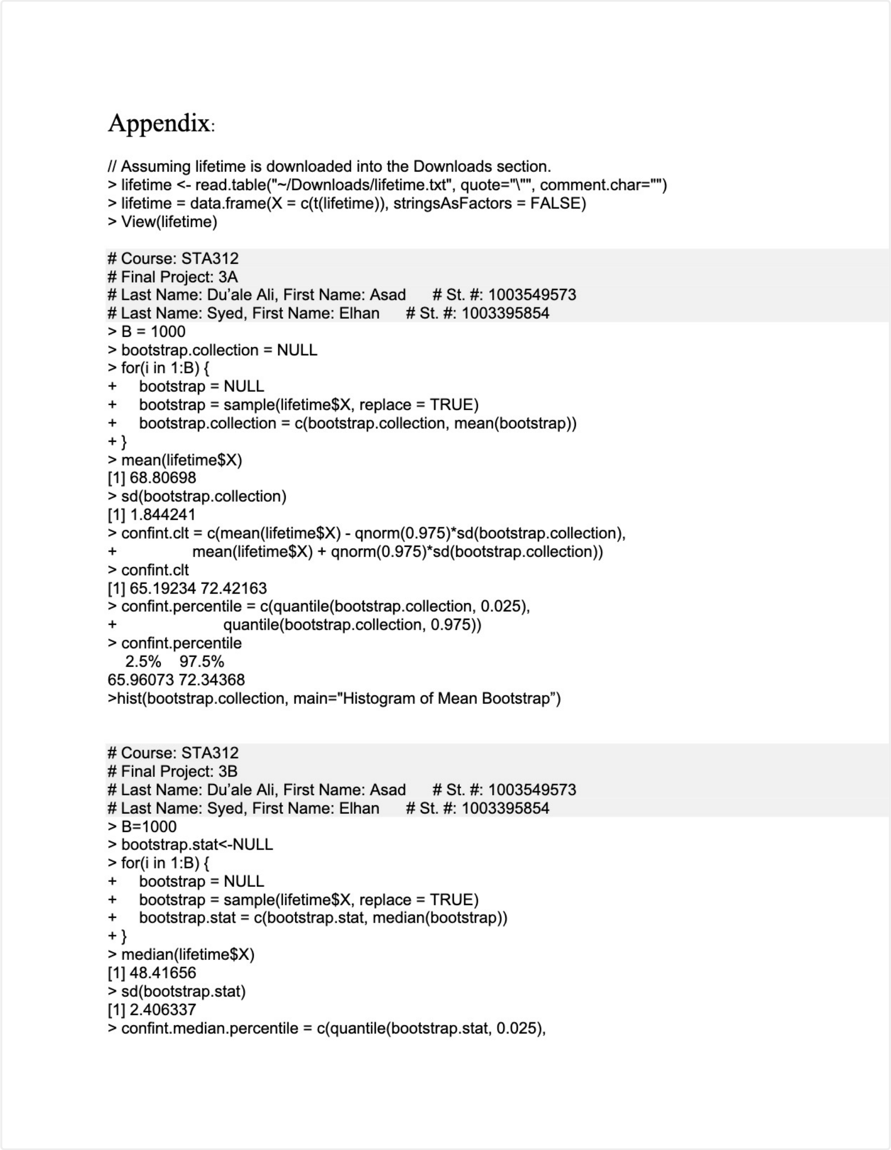
This was overall an extremely well done project. Good

However you are missing a table of contents

-0.5

Also, it would have been nice if you left chunks of code next to your answers, this would remove the need for me to scroll all the way down, look at your code, then all the way back up

-0.5

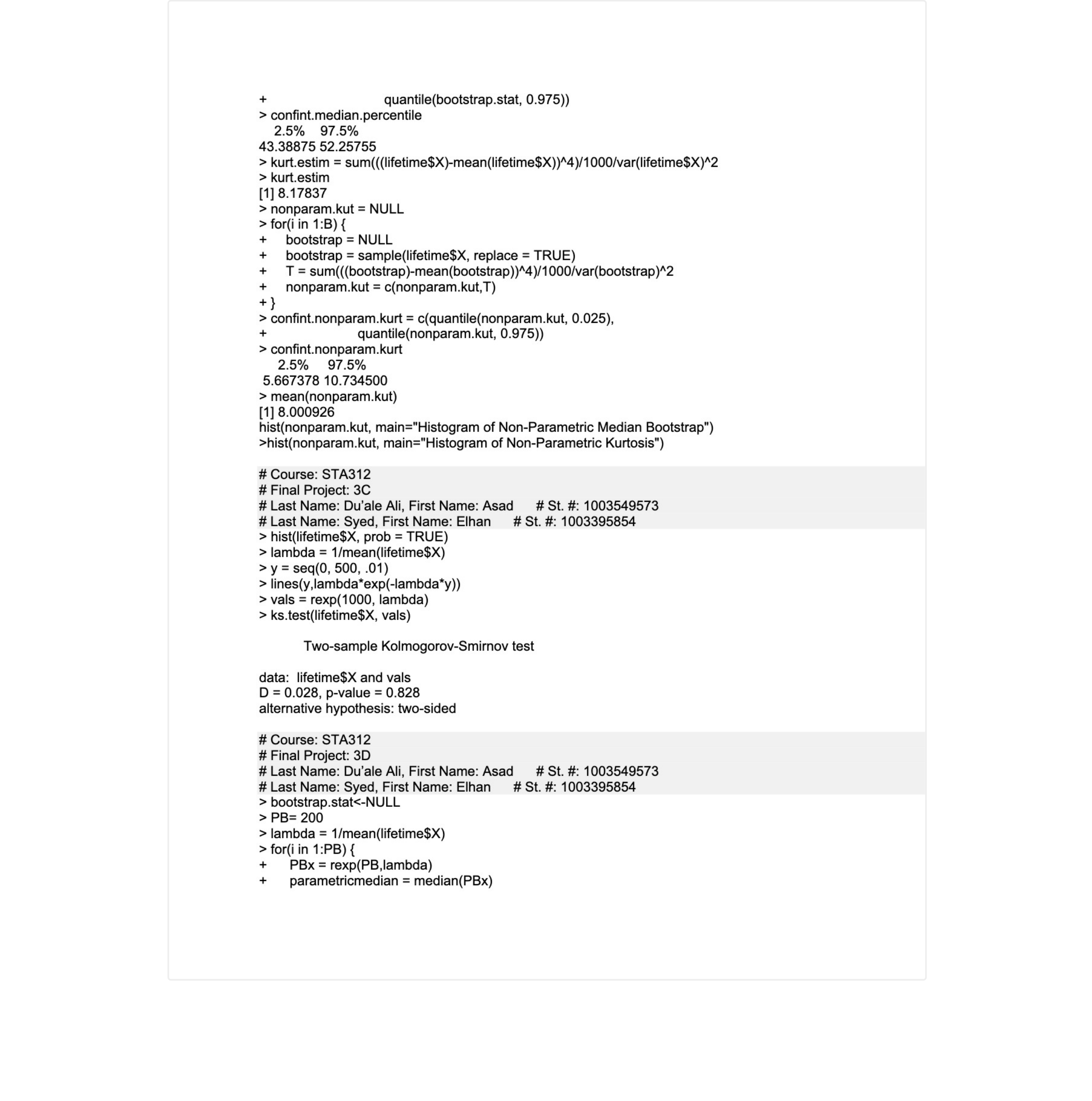


Appendix:

```
// Assuming lifetime is downloaded into the Downloads section.
> lifetime <- read.table("~/Downloads/lifetime.txt", quote="\"", comment.char="")
> lifetime = data.frame(X = c(t(lifetime)), stringsAsFactors = FALSE)
> View(lifetime)
```

```
# Course: STA312
# Final Project: 3A
# Last Name: Du’ale Ali, First Name: Asad      # St. #: 1003549573
# Last Name: Syed, First Name: Elhan      # St. #: 1003395854
> B = 1000
> bootstrap.collection = NULL
> for(i in 1:B) {
+     bootstrap = NULL
+     bootstrap = sample(lifetime$X, replace = TRUE)
+     bootstrap.collection = c(bootstrap.collection, mean(bootstrap))
+ }
> mean(lifetime$X)
[1] 68.80698
> sd(bootstrap.collection)
[1] 1.844241
> confint.clt = c(mean(lifetime$X) - qnorm(0.975)*sd(bootstrap.collection),
+                 mean(lifetime$X) + qnorm(0.975)*sd(bootstrap.collection))
> confint.clt
[1] 65.19234 72.42163
> confint.percentile = c(quantile(bootstrap.collection, 0.025),
+                        quantile(bootstrap.collection, 0.975))
> confint.percentile
    2.5%    97.5%
65.96073 72.34368
>hist(bootstrap.collection, main="Histogram of Mean Bootstrap”)
```

```
# Course: STA312
# Final Project: 3B
# Last Name: Du’ale Ali, First Name: Asad      # St. #: 1003549573
# Last Name: Syed, First Name: Elhan      # St. #: 1003395854
> B=1000
> bootstrap.stat<-NULL
> for(i in 1:B) {
+     bootstrap = NULL
+     bootstrap = sample(lifetime$X, replace = TRUE)
+     bootstrap.stat = c(bootstrap.stat, median(bootstrap))
+ }
> median(lifetime$X)
[1] 48.41656
> sd(bootstrap.stat)
[1] 2.406337
> confint.median.percentile = c(quantile(bootstrap.stat, 0.025),
```



```
+                           quantile(bootstrap.stat, 0.975))
> confint.median.percentile
    2.5%    97.5%
43.38875 52.25755
> kurt.estim = sum(((lifetime$X)-mean(lifetime$X))^4)/1000/var(lifetime$X)^2
> kurt.estim
[1] 8.17837
> nonparam.kut = NULL
> for(i in 1:B) {
+     bootstrap = NULL
+     bootstrap = sample(lifetime$X, replace = TRUE)
+     T = sum(((bootstrap)-mean(bootstrap))^4)/1000/var(bootstrap)^2
+     nonparam.kut = c(nonparam.kut,T)
+ }
> confint.nonparam.kurt = c(quantile(nonparam.kut, 0.025),
+                         quantile(nonparam.kut, 0.975))
> confint.nonparam.kurt
     2.5%     97.5%
 5.667378 10.734500
> mean(nonparam.kut)
[1] 8.000926
hist(nonparam.kut, main="Histogram of Non-Parametric Median Bootstrap")
>hist(nonparam.kut, main="Histogram of Non-Parametric Kurtosis")

# Course: STA312
# Final Project: 3C
# Last Name: Du'ale Ali, First Name: Asad      # St. #: 1003549573
# Last Name: Syed, First Name: Elhan      # St. #: 1003395854
> hist(lifetime$X, prob = TRUE)
> lambda = 1/mean(lifetime$X)
> y = seq(0, 500, .01)
> lines(y,lambda*exp(-lambda*y))
> vals = rexp(1000, lambda)
> ks.test(lifetime$X, vals)

        Two-sample Kolmogorov-Smirnov test

data:  lifetime$X and vals
D = 0.028, p-value = 0.828
alternative hypothesis: two-sided

# Course: STA312
# Final Project: 3D
# Last Name: Du'ale Ali, First Name: Asad      # St. #: 1003549573
# Last Name: Syed, First Name: Elhan      # St. #: 1003395854
> bootstrap.stat<-NULL
> PB= 200
> lambda = 1/mean(lifetime$X)
> for(i in 1:PB) {
+      PBx = rexp(PB,lambda)
+      parametricmedian = median(PBx)
+      bootstrap.stat=c(bootstrap.stat, parametricmedian)
+       }
> alpha=0.05
> q.low <- quantile(bootstrap.stat,alpha/2)
> q.high <- quantile(bootstrap.stat,1-alpha/2)
> CI = c(q.low,q.high)
> CI
    2.5%    97.5%
38.76744 56.60349
> parametricmedian
[1] 49.31878
> median(lifetime$X)
[1] 48.41656
> sd(bootstrap.stat)
[1] 2.490158
//kurtosis
> B = 1000
> bootstrap.stat.par = NULL
> lambda = 1/mean(lifetime$X)
> for ( i in 1:B){
+   boot.s.par = rexp(B,lambda)
+   fourth.moment=sum((boot.s.par-mean(boot.s.par))^4)/B
+   F = fourth.moment / var(boot.s.par)^2
+   bootstrap.stat.par = c(bootstrap.stat.par, F)
+ }
> hist(bootstrap.stat.par, main=”Histogram for Parametric Kurtosis”)
> hist(bootstrap.stat, main=”Histogram for Parametric Median”)
> confint.param.kurt = c(quantile(bootstrap.stat.par, 0.025), quantile(bootstrap.stat.par, 0.975))
> confint.param.kurt
    2.5%    97.5%
 5.91674 15.52047
> mean(bootstrap.stat.par)
[1] 8.893668


# Course: STA312
# Final Project: 3E
# Last Name: Du'ale Ali, First Name: Asad      # St. #: 1003549573
# Last Name: Syed, First Name: Elhan      # St. #: 1003395854

> X = rexp(1000, lambda)
> median(X)
[1] 45.8488
> kurtosis_theoretical = sum((X-mean(X))^4)/1000/var(X)^2
> kurtosis_theoretical
[1] 8.956054
```