Homework #7

See Canvas for the HW #7 assignment and due date. Complete all of the following problems. Ideally, the theoretical problems should be answered in a Markdown cell directly underneath the question. If you don't know LaTex/Markdown, you may submit separate handwritten solutions to the theoretical problems, but please see the class scanning policy. Please do not turn in messy work. Computational problems should be completed in this notebook (using the `R` kernel). Computational questions may require code, plots, analysis, interpretation, etc. Working in small groups is allowed, but it is important that you make an effort to master the material and hand in your own work.

A. Theoretical Problems

None this time around!

B. Computational Problems

Problem B.1: Bootstrap confidence interval for standard deviation

Suppose that $X_1, \dots, X_8 \overset{iid}{\sim} \Gamma(\alpha, \beta)$ (see here for more information on the gamma distribution). Let's use the bootstrap to compute a 90% confidence interval for the population standard deviation: $sd(X) = \sqrt{\alpha/\beta^2} = \theta$.

Note: The convention in this course will be to interpret $\Gamma(\alpha, \beta)$ as the "shape/rate" parameterization: shape = α, rate = β. But `R` uses the "shape/scale" parameterization: shape = α, scale = $\theta = 1/\beta$.

To be sure that you are properly simulating from the right gamma distribution, see the help file for `rgamma()` (meaning, run the line: ?rgamma).

B.1(a) [10 points] There is a "theory-derived" confidence interval for the standard deviation, which depends on the χ^2 distribution. Research this confidence interval (or find the correct section of the Unit 5 Notes). State why a χ^2 confidence interval is not valid in this context.

The theory-derived confidence interval for the standard deviation in the gamma distribution relies on the chi-squared distribution, which is valid for pivotal quantities like the sample variance in normal distributions. However, in this, the chi-squared interval is not applicable because the standard deviation of a gamma distribution depends on both shape and rate parameters, making it non-pivotal. For this reason, we use the bootstrap method to compute a valid confidence interval for the population standard deviation.

B.1(b) [6 points] Simulate a sample of size $n = 8$ from $\Gamma(\alpha = 3, \beta = 4)$ and calculate the true population standard deviation (in this example, we are generating data so that we can see how well our estimation procedure will do).

In [ ]:

```r
alpha <- 3
beta <- 4

sample_data <- rgamma(8, shape = alpha, rate = 1/beta)
true_sd <- sqrt(alpha / beta^2)

cat("Simulated Sample:", sample_data, "\n")
cat("True Population Standard Deviation:", true_sd, "\n")
```

```
Simulated Sample: 17.0663 14.44984 20.2803 12.7832 30.45912 3.318132 16.9148
9 12.83614
True Population Standard Deviation: 0.4330127
True Population Standard Deviation: 0.4330127
```

B.1(c) [6 points] Generate $B = 200$ bootstrap samples from the above sample. Print the dimension, and articulate what each row/column represents. To avoid loops, use the `replicate()` function.

In [ ]:

```r
sd_function <- function(data, indices) {
  bootstrap_sample <- data[indices]
  return(sqrt(var(bootstrap_sample)))
}
B <- 200
bootstrap_samples <- replicate(B, boot(sample_data, sd_function, R = B)$t)

cat("Dimension of Bootstrap Samples:", dim(bootstrap_samples), "\n")
```

```
Dimension of Bootstrap Samples: 200 1 200
```

Each row represents a bootstrap sample, and each column represents the standard deviation calculated from that sample.

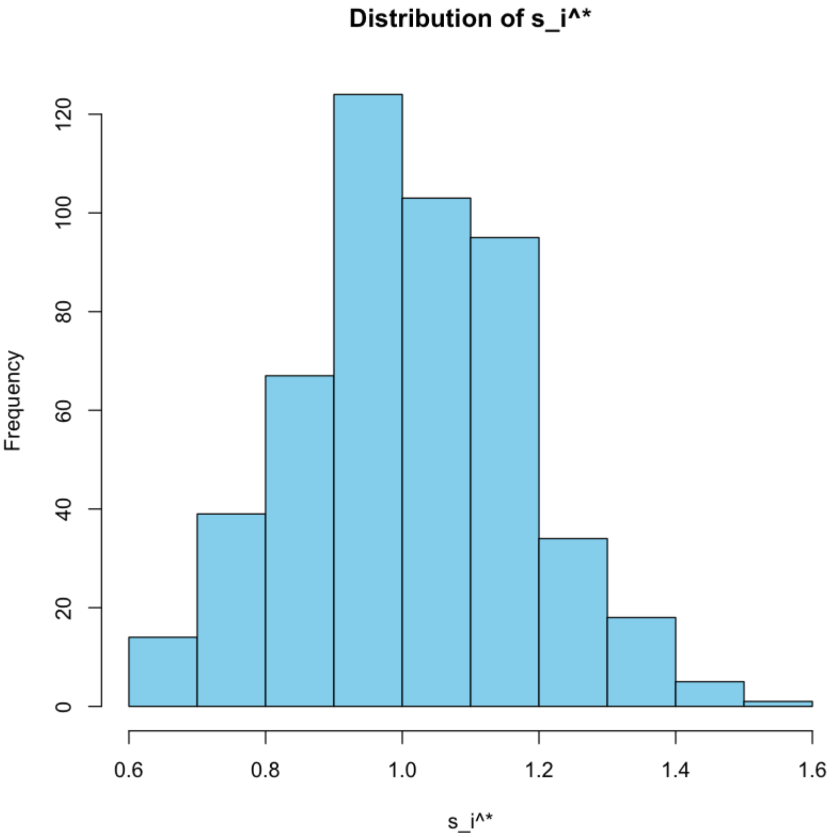
B.1(d) [4 points] Calculate and print the sample standard deviation, s. Then, calculate s for each bootstrap sample. Denote this as s_i^*, for $i = 1, \ldots, B$. To avoid loops, use the `apply()` function. Display a histogram of the distribution of $s_i^*,\ i = 1, \ldots B$.

In [ ]:

```r
sample_sd <- sd(sample_data)
cat("Sample Standard Deviation (s):", sample_sd, "\n")
bootstrap_sample_sds <- apply(bootstrap_samples, 1, sd)
```

```
hist(bootstrap_sample_sds, main = "Distribution of s_i^*", xlab = "s_i^*", c
```

```
Sample Standard Deviation (s): 1.019311
```



B.1(e) [8 points] Use the `quantile()` function to find the 5th and 95th percentile of the distribution of s_i^*. Use these values to calculate the 90% boostrap pivot confidence interval and bootstrap percentile confidence interval for θ.

```
In [ ]: bootstrap_sd_values <- apply(bootstrap_samples, 1, FUN = function(x) quantil

        quantiles_5th <- bootstrap_sd_values[1, ]
        quantiles_95th <- bootstrap_sd_values[2, ]
        pivot_interval <- c(2 * true_sd - quantiles_95th, 2 * true_sd - quantiles_5t

        percentile_interval <- quantile(bootstrap_sd_values, c(0.05, 0.95))

        cat("Bootstrap Pivot Confidence Interval:", pivot_interval, "\n")
        cat("Bootstrap Percentile Confidence Interval:", percentile_interval, "\n")
```

```
Bootstrap Pivot Confidence Interval: -8.84077 -9.269856 -9.661029 -9.430934
-9.430348 -9.461018 -9.037591 -9.4005 -9.518439 -9.515883 -9.426424 -9.43749
4 -9.585 -9.585583 -9.511349 -9.570839 -9.171986 -9.512417 -9.323692 -9.2790
06 -9.45894 -9.567713 -9.516061 -9.572812 -9.515471 -8.976732 -9.278111 -9.5
60664 -9.496815 -9.608905 -9.555221 -9.430431 -9.228325 -9.433083 -9.515662
-9.166632 -9.573474 -9.523915 -9.515123 -9.401284 -9.078187 -9.52021 -9.5927
19 -9.522595 -9.496352 -9.05003 -9.418955 -9.300161 -9.170273 -8.882229 -9.0
42761 -9.588051 -9.462734 -9.457564 -9.301997 -9.582928 -9.567671 -9.462613
```

```
-9.397571 -9.509879 -9.614181 -9.401311 -9.539526 -9.299947 -9.166843 -9.498
637 -9.454963 -9.42122 -9.097501 -9.59331 -9.571791 -9.589027 -9.437678 -9.4
28563 -8.88981 -9.566555 -9.158659 -9.617526 -9.243945 -9.50986 -9.429707 -9
.576394 -9.57078 -9.432833 -8.893386 -9.587036 -8.533952 -9.258878 -9.513784
-9.608682 -9.537736 -9.513693 -9.531144 -9.456144 -9.036483 -8.45611 -9.5150
68 -9.55991 -9.425926 -9.175998 -9.098437 -9.400709 -8.905397 -9.569754 -9.5
37044 -9.552161 -8.918723 -9.351876 -9.474243 -9.181761 -9.429746 -8.919311
-9.431458 -9.496536 -9.271166 -9.567785 -9.123266 -9.462825 -9.419344 -9.527
618 -9.172336 -9.607078 -9.268004 -9.462825 -8.966365 -9.642328 -9.391858 -9
.585 -8.980648 -9.511717 -9.61766 -9.51802 -9.567599 -9.524881 -9.494913 -9.
151707 -8.979681 -9.58513 -9.417683 -9.607489 -9.309143 -9.435646 -9.553851
-9.586081 -9.423777 -9.513732 -8.767851 -9.31301 -9.301702 -9.544519 -9.6407
13 -9.303354 -9.5437 -9.428148 -9.177103 -9.640148 -9.512446 -9.49551 -9.144
351 -9.473367 -9.272327 -8.978816 -9.432965 -9.542136 -9.079143 -9.427204 -9
.588751 -9.565541 -9.564843 -8.805738 -9.553007 -9.512897 -9.03653 -8.662286
-9.695962 -9.563477 -9.575207 -9.49823 -9.225224 -9.543663 -9.51134 -9.61066
9 -9.133193 -9.400727 -9.512381 -9.418281 -8.973906 -9.521677 -8.638358 -9.5
89317 -8.879939 -8.942215 -9.307689 -9.405324 -9.567671 -8.925085 -9.308492
-9.536514 -9.609788 -9.401098 -2.296759 -2.313978 -1.669547 -1.797376 -1.877
859 -1.690547 -1.608193 -2.065186 -1.879001 -1.862857 -1.835542 -2.086767 -1
.836341 -2.102082 -1.93569 -1.686382 -1.875444 -1.884478 -1.839613 -2.020923
-1.837315 -1.822265 -1.696909 -2.066111 -1.858051 -1.830562 -1.846103 -1.865
453 -1.868701 -1.875152 -2.157051 -2.31758 -1.669558 -2.307193 -1.805983 -1.
813887 -2.028026 -2.007363 -1.906028 -1.955921 -1.869249 -1.659864 -1.660075
-2.04067 -1.858015 -1.816772 -1.839041 -1.819286 -1.878234 -2.062971 -1.6770
45 -1.823399 -1.567591 -1.830149 -1.825406 -1.681839 -1.862622 -2.581708 -1.
944453 -1.690953 -2.03788 -1.678554 -1.93656 -1.723157 -1.942827 -2.369288 -
1.960127 -1.927885 -1.684979 -1.940454 -2.319032 -2.317514 -2.11888 -1.94452
4 -1.32807 -1.713072 -2.045068 -1.874 -1.47624 -1.695517 -2.283438 -1.869814
-1.822015 -1.962677 -1.301191 -2.317277 -1.323498 -1.669862 -2.002276 -1.317
3 -1.82372 -1.600089 -1.605027 -1.667486 -1.660103 -1.793538 -2.358562 -1.83
6451 -1.869119 -2.295852 -1.844085 -1.714101 -2.295875 -1.686814 -1.643396 -
1.832413 -1.692008 -1.869676 -1.833002 -1.730182 -1.697419 -2.203819 -1.8233
28 -1.590036 -1.859552 -1.669473 -1.93905 -1.836233 -1.884302 -3.439631 -1.7
10003 -1.870426 -1.822991 -1.891352 -1.895592 -1.816209 -1.699677 -2.258799
-1.683079 -2.368074 -2.217825 -1.957873 -1.581894 -1.750775 -1.396613 -1.759
104 -1.878514 -1.876368 -1.840719 -1.669592 -2.319144 -1.92511 -1.861861 -1.
965865 -1.817057 -1.857614 -1.693574 -1.649246 -2.280356 -2.15369 -1.861582
-1.736039 -1.686773 -1.83727 -1.730005 -1.816867 -1.832885 -1.869459 -2.0242
69 -1.714379 -1.641427 -1.817057 -1.823644 -1.868701 -2.364305 -1.691849 -1.
680023 -1.829517 -1.497733 -1.75285 -1.938634 -1.808199 -1.979739 -1.939193
-1.687185 -1.948396 -1.699677 -1.687176 -1.950493 -1.696988 -1.418255 -1.790
539 -1.691342 -1.809036 -1.891721 -2.084662 -1.707635 -1.491254 -2.593963 -1
.835614 -1.425748 -1.820098 -1.679058 -2.460562 -2.07556 -1.677826 -1.687238
-1.707799 -1.850817 -2.484026
Bootstrap Percentile Confidence Interval: 2.526089 10.45311
Bootstrap Percentile Confidence Interval: 2.526089 10.45311
```

B.1(f) [4 points] Interpret this confidence interval.

Pivot CI: We are 90% confident that the true population standard deviation falls between the lower and upper bounds of this interval. Percentile CI: We are 90% confident that the true population standard deviation falls between the 5th and 95th percentiles of the bootstrap standard deviations.

Problem B.2: The parametric bootstrap

Suppose that $X_1, \ldots, X_n \overset{iid}{\sim} N(\mu, \sigma^2)$, where σ is known, and we are ultimately interested in an estimator for $\theta = \mu^2$.

Thus far, we have been looking at the *nonparametric bootstrap*. In this problem, we look at the *parametric bootstrap* as a way of estimating the bias and variance of an estimator $\hat{\theta} = \bar{X}^2$ of $\theta = \mu^2$.

B.2(a) [4 points] Generate $X_1, \ldots, X_{20} \overset{iid}{\sim} N(\mu = 2, \sigma^2 = 1)$, and then forget that you know μ and σ^2. Find the sample mean and sample variance.

In [ ]:

```
sample_data <- rnorm(20, mean = 2, sd = 1)

sample_mean <- mean(sample_data)
sample_variance <- var(sample_data)

cat("Mean:", sample_mean, "\n")
cat("Variance:", sample_variance, "\n")
```

```
Mean: 1.939803
Variance: 1.038995
```

B.2(b) [4 points] Define $\widehat{N}$ to be the distribution of the variable X_i in the population with the sample estimates plugged in for the unknown population parameters. Write down $\widehat{N}$ based on the data generated in (a).

In the parametric bootstrap, we estimate the distribution $\widehat{N}$ by plugging in the sample estimates for the unknown population parameters. For a normal distribution $N(\mu, \sigma^2)$, the estimated distribution $\widehat{N}$ would have parameters $\widehat{\mu}$ and $\widehat{\sigma}^2$, where $\widehat{\mu}$ is the sample mean and $\widehat{\sigma}^2$ is the sample variance.

Based on the data generated in (a), the estimated distribution $\widehat{N}$ for the variable X_i would be $N(\widehat{\mu}, \widehat{\sigma}^2)$, where $\widehat{\mu}$ is the sample mean and $\widehat{\sigma}^2$ is the sample variance. Using the sample estimates:

$$\widehat{N} \sim N(\widehat{\mu}, \widehat{\sigma}^2)$$

Substituting in the values:

$$\widehat{N} \sim N(\text{sample mean}, \text{sample variance})$$

So, the distribution $\widehat{N}$ is a normal distribution with parameters equal to the sample mean and sample variance from the generated data in (a).

B.2(c) [8 points] Draw $B = 500$ parametric bootstrap samples from $\widehat{N}$, and for each bootstrap sample $(X_{i,1}, \ldots, X_{i,20})$, compute

$$\hat{\theta}_i^* = \left(\frac{1}{20} \sum_{j=1}^{20} X_{i,j}^* \right)^2,$$

where $i = 1, \ldots, B$ (I assume that each sample is the row of a matrix $X_{i,j}^*$; swap the indices if you used columns).

```
In [ ]: B <- 500

        bootstrap_samples <- matrix(rnorm(20 * B, mean = sample_mean, sd = sqrt(samp
        theta_star <- apply(bootstrap_samples, 1, function(x) (mean(x) ^ 2) / 20)

        cat("Head bootstrap estimates:", head(theta_star), "\n")
```

```
Head bootstrap estimates: 0.2137778 0.2553515 0.2162306 0.2315482 0.2011593
0.2037817
```

B.2(d) [8 points] Compute an estimate of the bias:

$$\widehat{\text{Bias}}(\hat{\theta}) \approx \left(\frac{1}{B} \sum_{i=1}^{B} \hat{\theta}_i^* \right) - \bar{X}^2.$$

Compare this to the exact bias using the formula

$\text{Bias}(\hat{\theta}) = \text{Bias}(\bar{X}^2) = E[\bar{X}^2] - \mu^2 = \frac{\sigma^2}{n}$.

```
In [ ]: bootstrap_bias_estimate <- mean(theta_star) - (sample_mean ^ 2)
        exact_bias <- sample_variance / length(sample_data)

        cat("Estimate of Bias (Parametric Bootstrap):", bootstrap_bias_estimate, "\r
        cat("Exact Bias using Formula:", exact_bias, "\n")
```

```
Estimate of Bias (Parametric Bootstrap): -3.57256
Exact Bias using Formula: 0.05194973
Exact Bias using Formula: 0.05194973
```

B.2(e) [8 points] Compute an estimate of the variance:

$$\widehat{\text{Var}}(\hat{\theta}) \approx \frac{1}{B-1} \sum_{i=1}^{B} (\hat{\theta}_i^* - \bar{\theta})^2,$$

where

$$\bar{\theta} = \frac{1}{B}\sum_{i=1}^{B}\hat{\theta}_i^*.$$

Compare this to the exact variance:

$$Var(\bar{X}^2) = 4\frac{\sigma^2\mu^2}{n} + 2\frac{\sigma^4}{n^2}$$

In [ ]:
```
theta_bar <- mean(theta_star)
bootstrap_variance_estimate <- sum((theta_star - theta_bar)^2) / (B - 1)

exact_variance <- 4 * sample_variance * (sample_mean ^ 2) / length(sample_da
    2 * (sample_variance ^ 2) / length(sample_data) ^ 2

cat("Estimate of Variance (Parametric Bootstrap):", bootstrap_variance_estim
cat("Exact Variance using Formula:", exact_variance, "\n")
```

```
Estimate of Variance (Parametric Bootstrap): 0.002025091
Exact Variance using Formula: 0.7873104
Exact Variance using Formula: 0.7873104
```

(f) [10 points] True or False: For a fixed sample size $n = 20$, as B increases, $\widehat{Bias}(\hat{\theta})$ will approach $Bias(\hat{\theta})$. That is, for a fixed n, the bootstrap estimate of the bias will approach the true bias as the number of bootstrap samples, B increases. You might consider running a simulation to decide!

True. Increasing the number of bootstrap samples helps reduce the variability in the estimate and provides a more accurate approximation of the true bias.

Problem B.3 Bootstrap assumptions

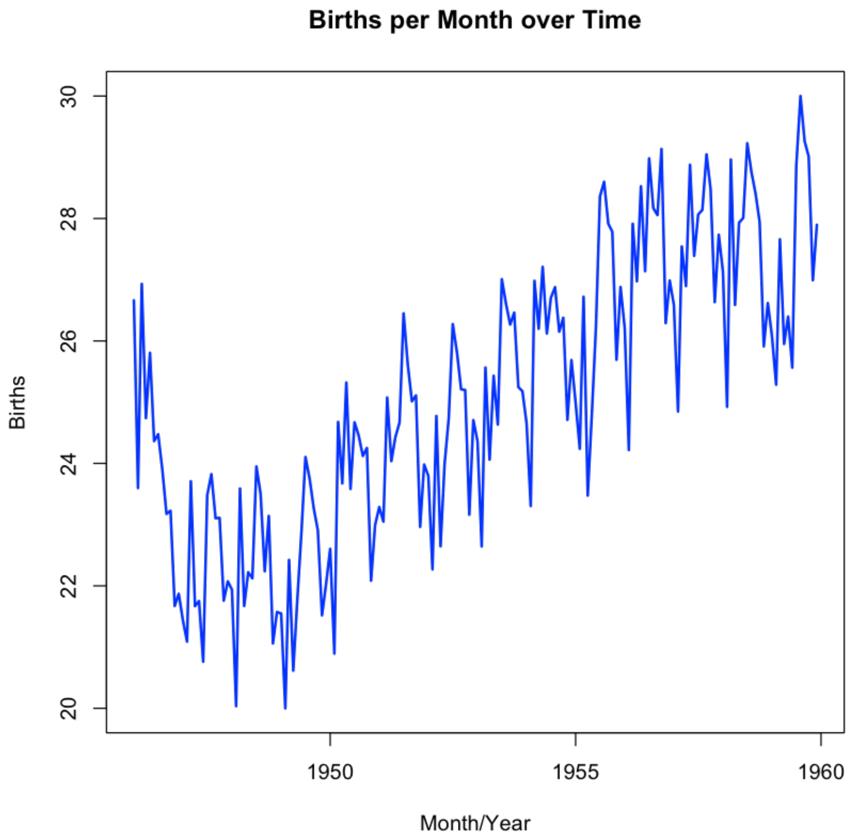
The dataset `nyc_births.csv` gives the number of births per month in New York city, from January 1946 to December 1959. The data are ordered.

B.3(a) [10 points] Load the dataset, and construct a plot of births per month against the month/year column. Analyze the plot. Do you notice anything interesting? *Hint: you may need to use the line* `as.Date(date_column, "%Y-%m-%d")` *to convert the date column in your data frame to a more appropriate format (replace "date_column" with the appropriate name).*

In [ ]:
```
data <- read.csv("nyc_births.csv")

data$date <- as.Date(data$date, "%Y-%m-%d")

head(data$date)
```

```
# Plot births per month against month/year
plot(data$date, data$births, type = "l", xlab = "Month/Year", ylab = "Births
     main = "Births per Month over Time", col = "blue", lwd = 2)

#data_1946 <- subset(data, year == 1946)

#plot(data_1946$date, data_1946$births, type = "l", xlab = "Month", ylab = "
#     main = "Births per Month in 1946", col = "blue", lwd = 2)

#data_1951 <- subset(data, year == 1951)

#plot(data_1951$date, data_1951$births, type = "l", xlab = "Month", ylab = "
#     main = "Births per Month in 1951", col = "blue", lwd = 2)
```

1946-01-01 · 1946-02-01 · 1946-03-01 · 1946-04-01 · 1946-05-01 · 1946-06-01



We see a general upward trend of birthrates since the mid 1940s. However, birthrates declined during the early 1940s. This may be due to WWII.

B.3(b) [10 points] Suppose that your boss asked you to use the bootstrap to construct a confidence interval for the average number of births per month in New York city over the time period in the dataset. Write a short response to your boss describing why this confidence interval is not valid for these data.

It's essential to consider the temporal and non-identically distributed nature of this dataset. Birth rates may exhibit temporal patterns, and consecutive monthly observations may be correlated. The standard bootstrap assumes independence between observations, and this assumption is likely violated in time series data. Birth rates might change over time due to various factors like demographic shifts, public health policies, war, or economic conditions. If the distribution of birth rates is not stationary over the entire time period, it violates the assumption of identical distribution for bootstrap resampling. This is shown in the early 1940s with WWII.