(1)

(a) $\sigma^2 = \frac{\sum (x_i - \mu)^2}{N} = \frac{\sum x_i^2 - 2\mu x_i + \mu^2}{N}$

$= \frac{\sum x_i^2}{N} - \frac{\sum 2\mu x_i}{N} + \frac{\sum \mu^2}{N}$

$= \frac{\sum x_i}{N} \cdot - 2\mu \frac{\sum x_i}{N} + \frac{N\mu^2}{N}$ → since θ or 2

$= \mu - 2\mu(\mu) + \mu^2 = \mu - 2\mu^2 + \mu^2 = \mu - \mu^2$

$\Rightarrow = \mu(1-\mu)$ thus $\sigma^2 = \mu(1-\mu)$

(b) For unbiased estimate $\mathbb{E}(\hat{s}^2) - \sigma^2 = 0$

↳ ~~$\mathbb{E}(\hat{s}^2) - \mu(1-\mu) = 0$~~

↳ ~~[illegible]~~

From 4a notes we have $\hat{s}^2$ is unbiased for

$s^2 = \left(1 - \frac{1}{N}\right) \frac{1}{n-2} \sum_{i=2}^{n} (x_i - \bar{x}_n)^2$

$= \left(1 - \frac{1}{N}\right) \frac{1}{n-2} \left(\sum_{i=2}^{n} x_i^2 - 2\bar{x}_n \sum_{i=2}^{n} x_i + \sum_{i=2}^{n} \bar{x}_n^2 \right)$

$= \left(1 - \frac{1}{N}\right) \frac{1}{n-2} \left(n\bar{x}_n - 2n\bar{x}_n^2 + n\bar{x}_n^2 \right)$

$= \left(1 - \frac{1}{N}\right) \frac{n}{n-1} \left(\bar{x}_n - \bar{x}_n^2 \right)$

$= \frac{(N-1)n}{(n-1)N} \left(\bar{x}_n - \bar{x}_n^2 \right)$

(2c)

$$s = \sqrt{\frac{(N-1)n}{(n-1)N}\left(\bar{x}_n - \bar{x}_n^2\right)} =$$

$$se(\bar{x}_n) = \frac{s}{\sqrt{n}}\sqrt{1 - \frac{n-2}{N-2}}$$

$$= \sqrt{\frac{(299)(90)}{(89)(300)}\left(\frac{7}{9} - \frac{7}{9}^2\right)} \simeq .41737$$

$$= \frac{.41737}{\sqrt{90}}\sqrt{1 - \frac{89}{299}} \approx .03687$$

z^* for 95% confidence = 1.96

confidence interval for $\bar{x}_n$ = $\bar{x}_n \pm z^* \cdot se[\bar{x}_n]$

$$\Rightarrow = \frac{7}{9} \pm 1.96 \cdot (.03687)$$

$$= [.7055, .85]$$

The confidence interval for the population proportion given this sample and s^2 from (b) is with 95% confidence & $\mu \in [.7055, .85]$

(2)

Population variance worst case scenario $\rightarrow p = .5$

$\sigma^2 = p(1-p) = (.5)^2 = .25$

$se[\bar{X}_n] = \sqrt{\frac{\sigma^2}{n}}$ → ignore finite population correction

Confidence interval $= \bar{X}_n \pm 1.96 \cdot se[\bar{X}_n]$ ← 95% z^*

$||CI|| = 2 \cdot 1.96 \cdot se[\bar{X}_n] \leq .04$

$se[\bar{X}_n] \leq \frac{.04}{3.92} = \frac{1}{98}$

$se[\bar{X}_n] = \frac{\sigma}{\sqrt{n}} \leq \frac{1}{98} \Rightarrow 98\sigma \leq \sqrt{n}$

$98\sigma = 98\left(\frac{1}{2}\right) = 49 \leq \sqrt{n}$

$\Rightarrow n \geq 49^2 = 2401$

Thus, ignoring finite population correction, we conclude a 95% confidence interval to have at most a .04 width, the minimum sample size must be $n \geq 2401 = 49^2$ students

4

(a) Given $\hat{\theta} = 2\bar{x}$, the bias is given by $E(\hat{\theta}) - \theta = E(2\bar{x}) - \theta$, notice given uniform distribution $E(\bar{x}) = E(x) = \frac{\theta}{2}$

so $E(2\bar{x}) = 2E(\bar{x}) = 2\frac{\theta}{2} = \theta$

$\Rightarrow E(\hat{\theta}) - \theta = \theta - \theta = 0 \Rightarrow$ bias$(\hat{\theta} = 2\bar{x}) = 0$

$MSE = \text{bias}^2 + \text{var}(\hat{\theta}) = 0 + \text{var}(\hat{\theta}) = \text{var}(\hat{\theta})$

$MSE = \text{var}(\hat{\theta}) = \sqrt{se}$

$\text{var}(\hat{\theta}) = \text{var}(2\bar{x}) = 4\,\text{var}(\bar{x}) = 4\,\text{var}\left(\frac{\sum x_i}{n}\right)$

$\Rightarrow 4\,\text{var}\left(\frac{\sum x_i}{n}\right) = \frac{4}{n^2}\text{var}(\sum x_i)$ x_i are independent:

$\frac{4}{n^2}\text{var}\left(\sum_n x_i\right) = \frac{4n}{n^2}\text{var}(x_i) = \frac{4}{n}\left(\frac{\theta^2}{12}\right) = \frac{\theta^2}{3n}$

Bias $= 0$, $MSE = \frac{\theta^2}{3n}$, $SE = \sqrt{\frac{\theta^2}{3n}}$

4

(c) $\frac{\theta^2}{3n} = \frac{2\theta^2}{n^2+3n+2} \Rightarrow n = 1, 2, >$ for $n > 2$

The MSE's of both estimates $2\bar{x}_n$, $\max(x_n)$ are equal for $n = 1, 2$ but $\max(x_n)$ has lower MSE for $n > 2$, so $\max(x_n) = \hat{\theta}$ is the more efficient estimate

4

(b) $\text{Bias} = E(\hat{\theta}) - \theta$

$F(\hat{\theta}) \Rightarrow \text{CDF of } \theta = \left(\frac{x}{\theta}\right)^n \Rightarrow \text{PDF} = \frac{d}{dx}\left(\frac{x}{\theta}\right)^n = \frac{n}{\theta}\left(\frac{x}{\theta}\right)^{n-1}$

$$E(\hat{\theta}) = \int_0^\theta x \cdot p(x)\,dx = \int_0^\theta \frac{n}{\theta^n} x^n\,dx = \frac{n}{\theta^n}\int_0^\theta x^n\,dx$$

$$\Rightarrow = \frac{n}{\theta^n}\left[\frac{1}{n+2} x^{n+2}\right] = \frac{n}{n+2}\left(\frac{\theta^{n+2}}{\theta^n}\right) = \theta\left(\frac{n}{n+2}\right) = E(\hat{\theta})$$

$$E(\hat{\theta}) - \theta = \theta\left(\frac{n}{n+2}\right) - \theta = \boxed{\frac{-\theta}{n+2} = \text{bias}(\hat{\theta})}$$

$$MSE = \text{bias}^2 + \text{var} = \text{bias}^2 + se^2$$

$$\text{var} = E(\hat{\theta})^2 - E(\hat{\theta})^2 = \int_0^\theta \frac{n x^2}{\theta}\left(\frac{x}{\theta}\right)^{n-1} dx - \left[\theta\left(\frac{n}{n+2}\right)\right]^2$$

$$\Rightarrow \frac{n}{\theta^n}\int_0^\theta x^{n+1} = \frac{n\theta^2}{n+2} \Rightarrow \text{var} = \frac{n\theta^2}{n+2} - \frac{\theta^2 n^2}{(n+2)^2}$$

$$se = \sqrt{\text{var}} = \sqrt{\frac{n\theta^2}{n+2} - \frac{\theta^2 n^2}{(n+2)^2}} = \boxed{\frac{\theta}{n+2}\sqrt{\frac{n}{n+2}}}$$

$$\text{var} = \frac{\theta^2 n}{(n+2)(n+2)^2} \Rightarrow \text{var} + \text{bias}^2 = MSE$$

$$\Rightarrow \frac{\theta^2 n}{(n+2)(n+2)^2} + \left[\frac{-\theta \cdot 2}{(n+2)}\right]^2 = \boxed{\frac{2\theta^2}{n^2+3n+2}} = MSE$$

```
clc; clear; close all;
```

Problem 3

```
data = textread("C:\Users\sayuj\OneDrive - California Institute of
Technology\ACM_157\birth.txt");
birth_weights = data(:, 1);
birth_weights = birth_weights(birth_weights ~= 999);
birth_weights = birth_weights * 0.0283495;
```

Part A

```
population_mean = mean(birth_weights)
```

```
population_mean = 3.3899
```

```
n = 100;
sample = datasample(birth_weights, 100, 'Replace', false);
sample_mean = mean(sample)
```

```
sample_mean = 3.4357
```

```
exact_se = std(birth_weights) / sqrt(n) * sqrt(1 - (n - 1) / (length(birth_weights)
- 1))
```

```
exact_se = 0.0496
```

From the code above, we have calculated the population mean for the birth weights to be 3.3899, the sample mean for a size n = 100 sample to vary closesly around that value, and the exact standard error of mean estimate for n = 100 samples to be .0496

Part B results

```
b_results = bootstrap_algo_b(100, birth_weights, sample)
```

```
b_results = 0.0495
```

From the results of running this (B) bootstrapping algorithm multiple times it appears that the standard error of this estimation varies around the true exact value.

Part C results

```
c_results = bootstrap_algo_c(100, birth_weights, sample)
```

```
c_results = 0.0491
```

From the results of running this (C) bootstrapping algorithm multiple times it appears that the standard error of this estimation varies around the true exact value.

Part D results

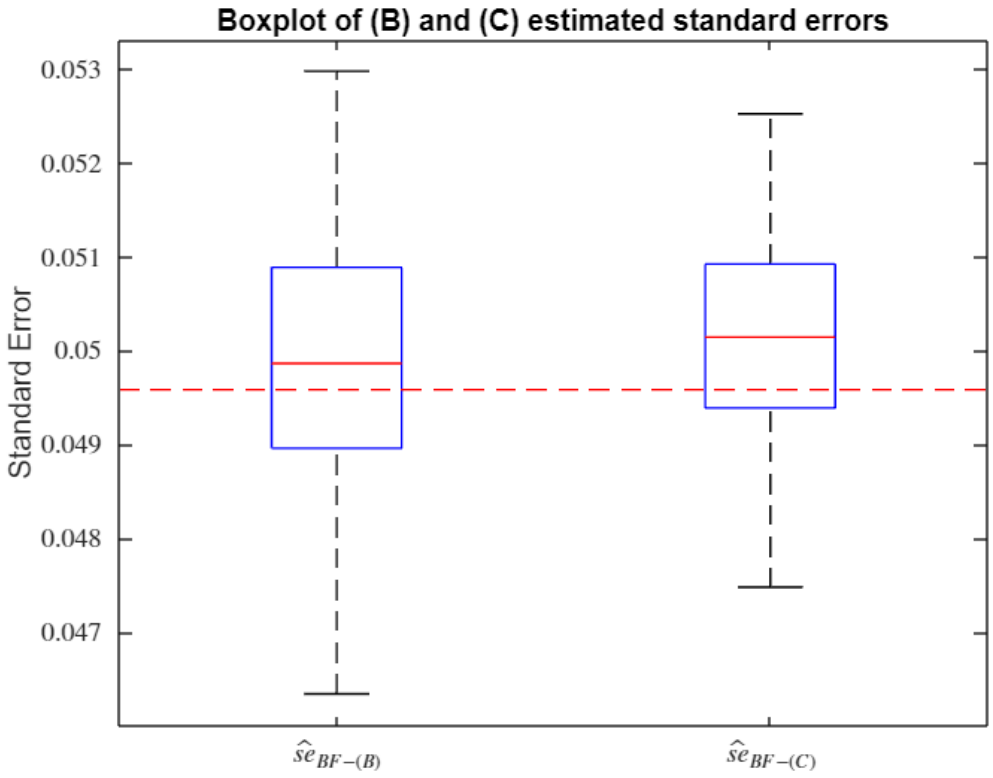
```
algo_b_se = zeros(100, 1);
algo_c_se = zeros(100, 1);
```

```
for s = 1:100
    algo_b_se(s) = bootstrap_algo_b(100, birth_weights, sample);
    algo_c_se(s) = bootstrap_algo_c(100, birth_weights, sample);
end

boxplot([algo_b_se, algo_c_se])

hold on;
line(xlim, [exact_se, exact_se], 'Color', 'r', 'LineStyle', '--');

ylabel('Standard Error');
title('Boxplot of (B) and (C) estimated standard errors');
labels = {'$\hat{se}_{BF-(B)}$', '$\hat{se}_{BF-(C)}$'};
set(gca, 'XTickLabel', labels, 'TickLabelInterpreter', 'latex')
```



From running bootstrapping algorithms B and C 100 times on the same sample and analyzing the boxplot of the results, both estimates seem to accurately estimate the exact standard error pretty well as the true exact standard error (dashed red line) is contained within the mid 50% quartiles of results for both bootstrapping algorithms. Based on the boxplots it also seems possible that the variance of algorithm C's results is a little smaller than algorithm B but there is not conclusive statisticaly evidence that this is true.

Part B Function Code

```
function bootstrap_se = bootstrap_algo_b(n, birth_weights, sample)
    bootstrap_reps = floor(length(birth_weights) / n);
    bootstrapped_population = repelem(sample, bootstrap_reps);

    bootstrap_samples = 1000;
    sample_means = zeros(bootstrap_samples, 1);

    for i = 1:bootstrap_samples
        sample_data = datasample(bootstrapped_population, n, 'Replace', false);

        sample_mean = mean(sample_data);

        sample_means(i) = sample_mean;
    end

    bootstrap_se = sqrt(sum((sample_means - (mean(sample_means))).^2) /
bootstrap_samples);
end
```

Part C Function Code

```
function new_bootstrap_se = bootstrap_algo_c(n, birth_weights, sample)
    bootstrap_reps = (length(birth_weights) / n);
    bootstrapped_population_1 = repelem(sample, floor(bootstrap_reps));
    bootstrapped_population_2 = repelem(sample, ceil(bootstrap_reps));

    r = length(birth_weights) - (floor(bootstrap_reps) * n);

    p = (1 - r / n) * (1 - r / (length(birth_weights) + 1));



    bootstrap_samples = 1000;
    sample_means = zeros(bootstrap_samples, 1);

    for i = 1:bootstrap_samples
        if rand > p
            sample_data = datasample(bootstrapped_population_1, n, 'Replace',
false);

            sample_mean = mean(sample_data);

            sample_means(i) = sample_mean;
        else
            sample_data = datasample(bootstrapped_population_2, n, 'Replace',
false);

            sample_mean = mean(sample_data);

            sample_means(i) = sample_mean;
```

```
        end
    end

    % sum((sample_means - (mean(sample_means))).^2)

    new_bootstrap_se = sqrt(sum((sample_means - (mean(sample_means))).^2) /
bootstrap_samples);
end
```