BACS HW3

109062710

March 17, 2021

Question 1

Here is the helper functions for Q1

```
standardize <- function(data) {
  standardized <- (data - mean(data)) / sd(data)
  return(standardized)
}

create_density <- function(data, title) {
  mean <- mean(data)

  sd_values = c(
    mean(data) - 2 * sd(data),
    mean(data) - sd(data),
    mean(data) + sd(data),
    mean(data) + 2 * sd(data)
  )

  ggplot(mapping = aes(data)) +
    geom_density(
      fill="#69b3a2",
      color="#e9ecef",
    ) +
    geom_vline(xintercept = mean, col="black") +
    geom_vline(xintercept = sd_values, col="red") +
    ggtitle(title)
}

create_histogram <- function(data, title) {
  n = length(data)

  # Freidman-Darconis' BiUnwidth Rule
  binwidth <- (2 * IQR(data)) / n^(1/3)
  bins <- ceiling(max(data) - min(data)) + binwidth

  ggplot(mapping = aes(data)) +
    geom_histogram(
      fill="#69b3a2",
      color="#e9ecef",
      bins = bins,
```

```
      binwidth = binwidth
    ) +
    ggtitle(title)
}
```

A. create a normal distribution (`mean = 940, sd = 190`) and standardize it

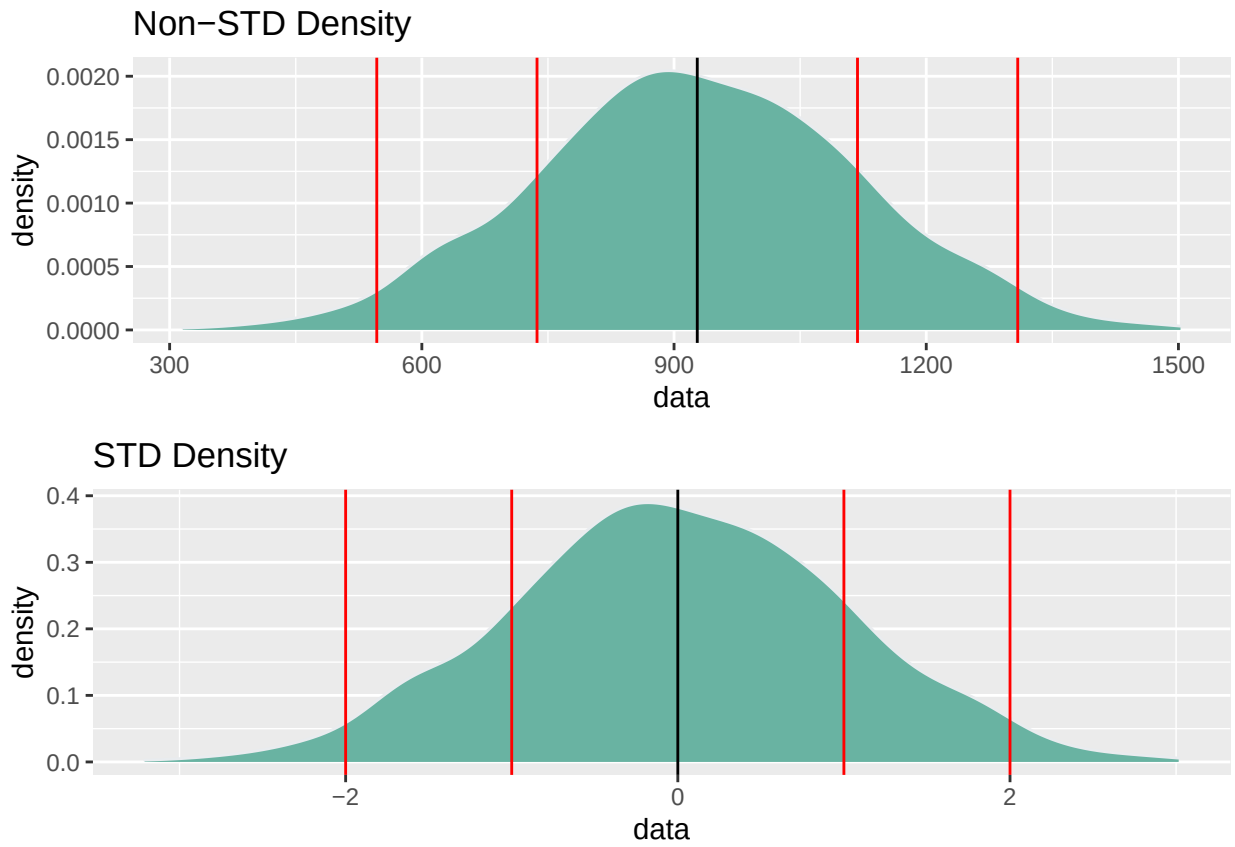
```
rnorm_q1 <- rnorm(1000, mean = 940, sd = 190)
rnorm_std <- standardize(rnorm_q1)
```

i) What should we expect the mean and standard deviation of rnorm_std to be, and why?

```
## The mean of rnorm is 927.622132490757,
## and its standard deviation is 190.619864968953.
```

```
## The mean of rnorm_std is -2.38475038327746e-16,
## and its standard deviation is 1.
```

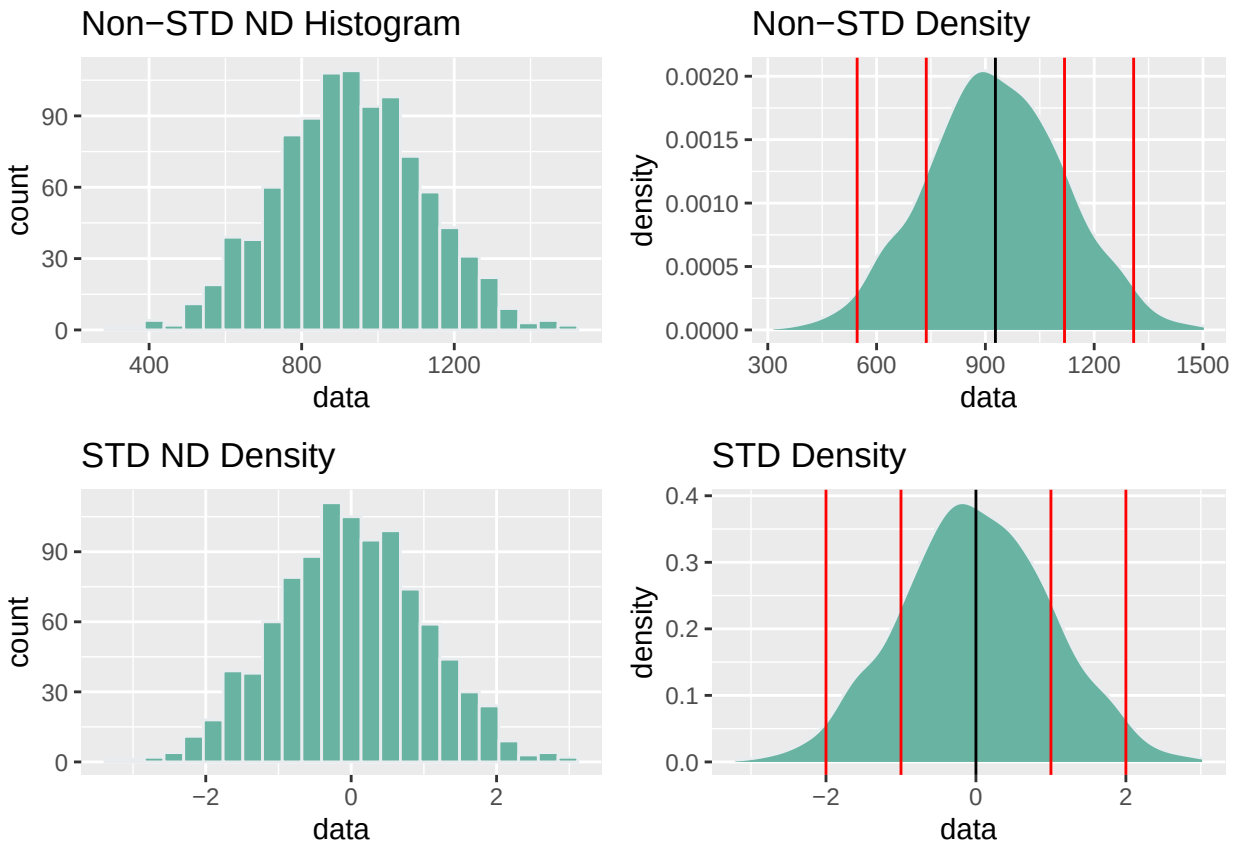
```
grid.arrange(
  rnorm_density,
  rnorm_std_density,
  ncol=1,
  nrow=2
)
```



In this case, the `mean` value is `0`. After standardization, `x_value` has a range of `-3` to `3`. That range represents how far each instance from the mean in STD unit. This happens because standardization scales down everything to STD unit scale.

ii) What should the distribution (shape) of rnorm_std look like, and why?

```
grid.arrange(
  rnorm_hist,
  rnorm_density,
  rnorm_std_hist,
  rnorm_std_density,
  ncol=2,
  nrow=2
)
```



Basically, `rnorm_std` and `rnorm` plots should look entirely the same, but they are not. Let's take the graph above as a reference.

However, there is a worth mentioning here:

1. Non-standardized and standardized histograms look almost the same, but there is a slight difference if you take a close look.
2. The `x_values` range becomes smaller in standardized density plot because standardization scales down everything to STD unit scale.

iii) What do we generally call distributions that are normal and standardized?

It's called **bell-shaped curved** distribution.

B. Create a standardized version of `minday` from the earlier question (let's call it `minday_std`)

```
minday_std <- standardize(minday)
```

i) What should we expect the mean and standard deviation of `minday_std` to be, and why?

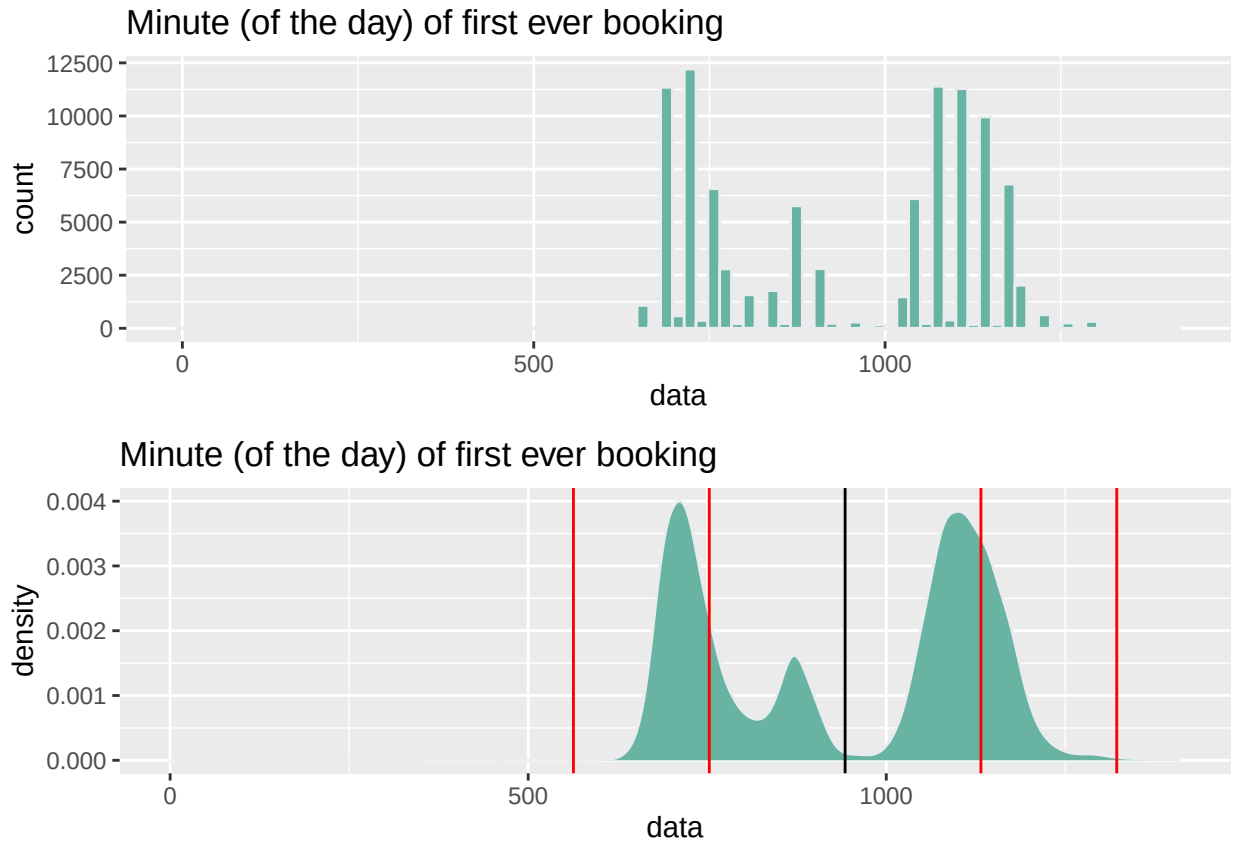
```
## The mean of minday_std -4.25589034500073e-17, while its SD is 1.
```

We expect the mean and the STD values to be really small which are within `-2.5` to `2.5` range after standardization because standardization scales down everything to STD unit scale. In this case, `mean` becomes zero.

ii) What should the distribution of `minday_std` look like compared to `minday`, and why?

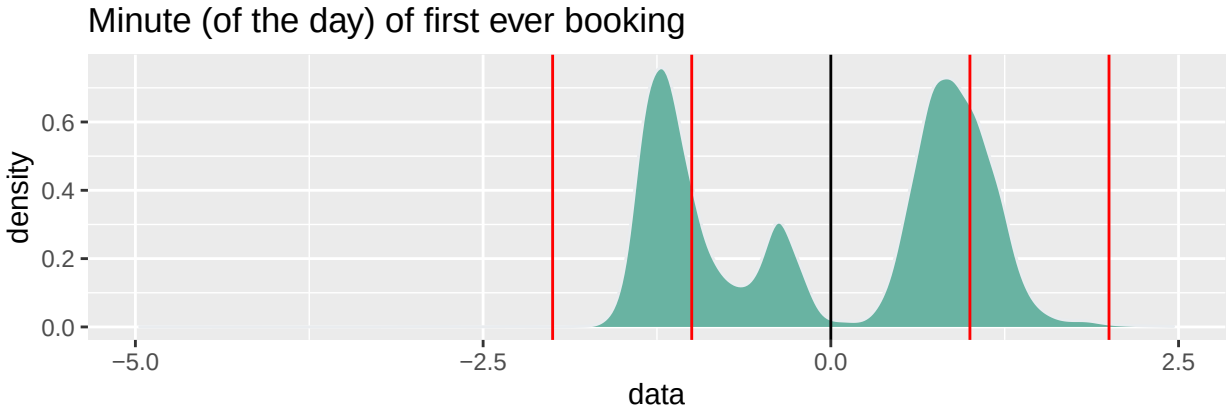
Before standardization,

```
grid.arrange(
  minday_hist,
  minday_density,
  ncol=1,
  nrow=2
)
```



After standardization,

```
grid.arrange(
  minday_std_hist,
  minday_std_density,
  ncol=1,
  nrow=2
)
```



The situation is the similar to the section a, part ii. In the non-standardized data set, the STD lines are far away when we expect them to be. Besides, we have a huge range of `x_value` which is from `0` to `1500`.

However, in the standardized data set, the mean line is exactly in between the STD lines. In addition, we have a smaller range of `x_value` which is from `-4` to `4`.

Question 2

a) Simulate 100 samples (each of size 100), from a normally distributed population of 10,000:

i) How many samples do we expect to NOT include the population mean in its 95% CI?

```
## 5.47
```

Out of 100 simulations, the answer can be rounded down to 5 samples.

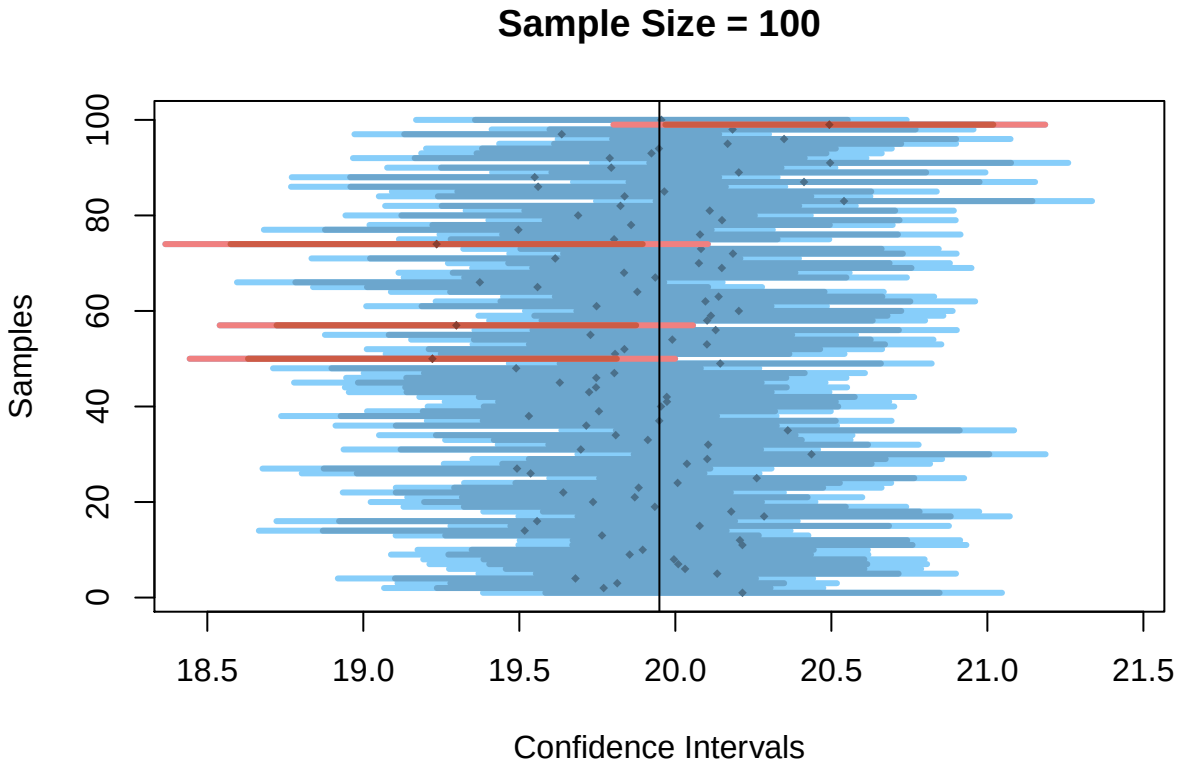
ii) How many samples do we expect to NOT include the population mean in their 99% CI?

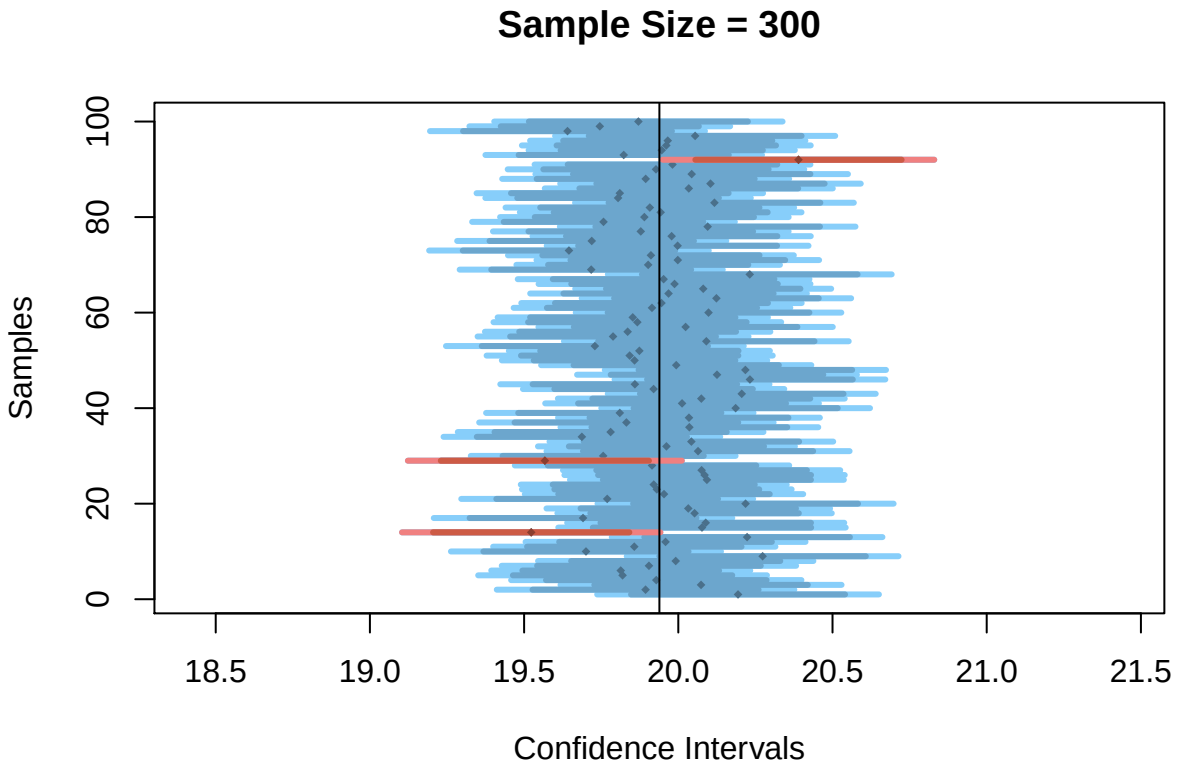
```
## 1.25
```

Out of 100 simulations, the answer can be rounded down to 1 sample.

b) Rerun the previous simulation with larger samples (sample_size=300):

i) Now that the size of each sample has increased, do we expect their 95% and 99% CI to become wider or narrower than before?





As we can see from those two plots above, the 95% and 99% CI become narrower when the sample size increases.

ii) This time, how many samples (out of the 100) would we expect to NOT include the population mean in its 95% CI?

```
## 4.745
```

Out of 100 simulations, the answer can be rounded down to 4 samples.

c) If we ran the above two examples (a and b) using a uniformly distributed population (specify distr_func=runif for visualize_sample_ci), how do you expect your answers to (a) and (b) to change, and why?

`num_sample = 100` & `distr_func = runif`

i) How many samples do we expect to NOT include the population mean in their 95% CI?

```
## 5.575
```

ii) How many samples do we expect to NOT include the population mean in their 99% CI?

```
## 1.24
```

`num_sample = 300` & `distr_func = runif`

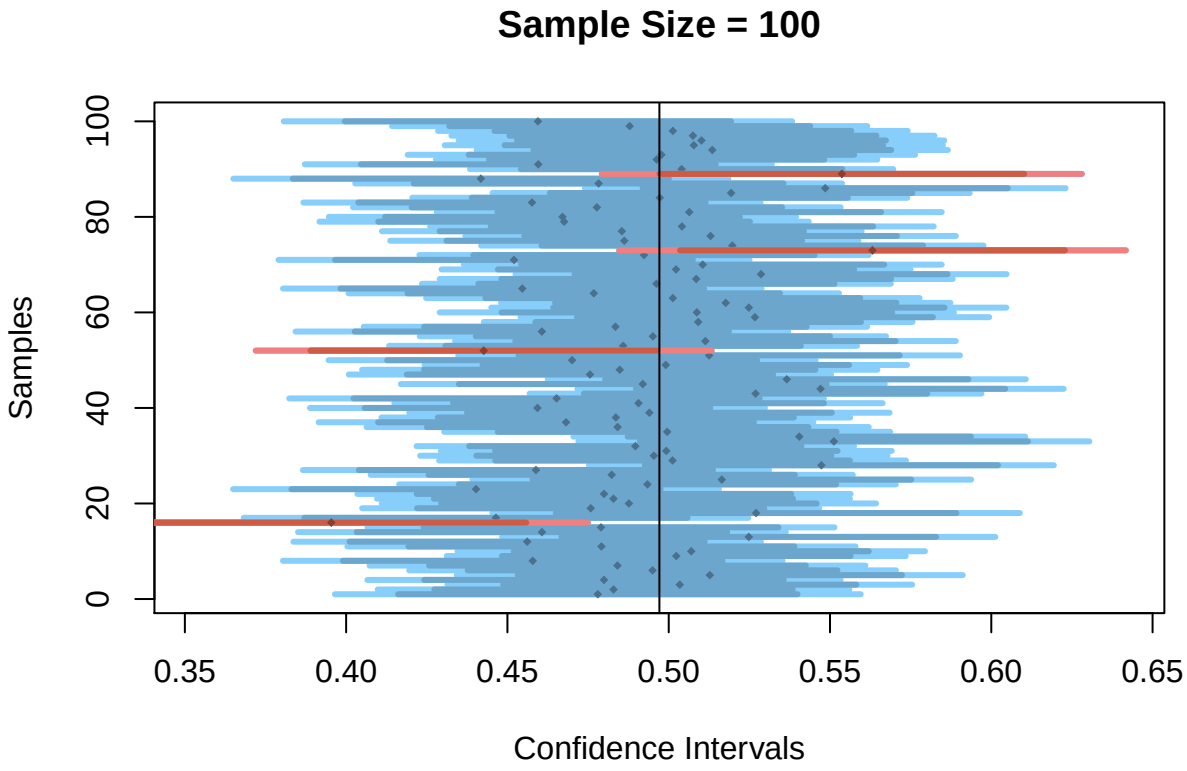
i) How many samples do we expect to NOT include the population mean in their 95% CI?

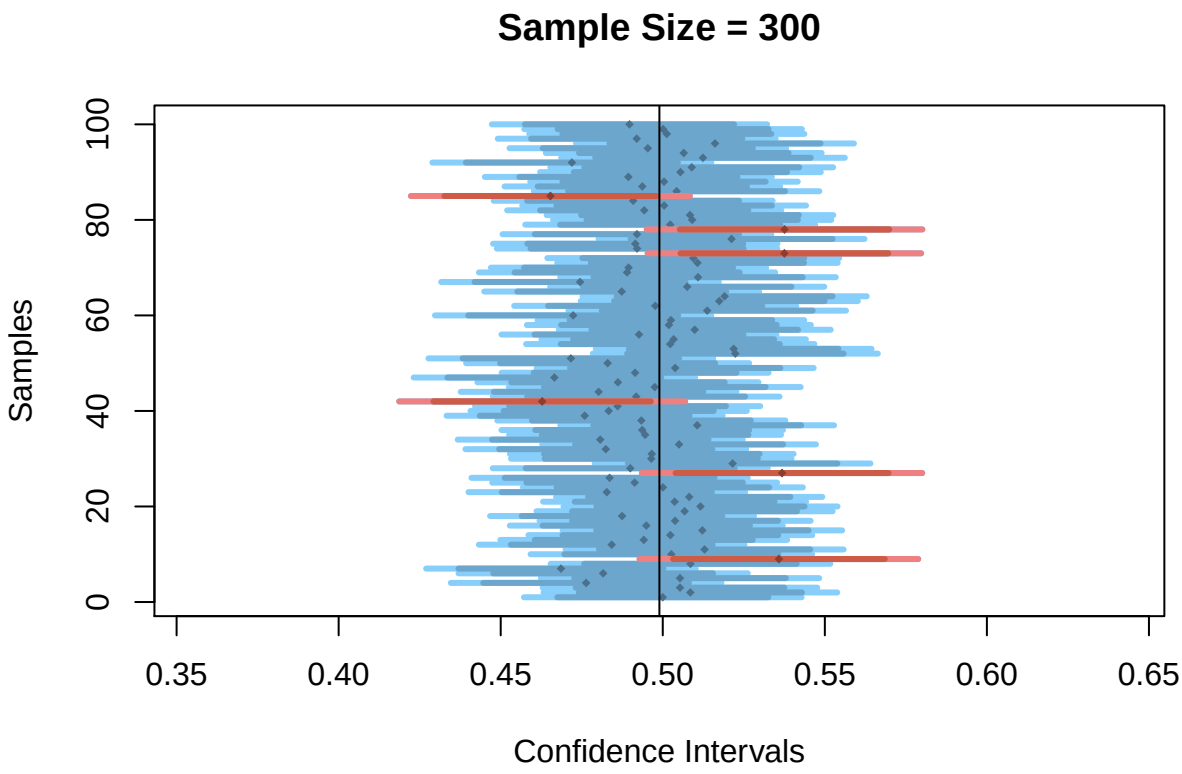
```
## 4.95
```

ii) How many samples do we expect to NOT include the population mean in their 99% CI?

```
## 1.085
```

When we look at those two configurations above, we can see that the changes in mean and standard deviation values both in `rnorm` and `runif` follow the same pattern.


Sample Size = 100
Samples
0
20
40
60
80
100
0.35
0.40
0.45
0.50
0.55
0.60
0.65
Confidence Intervals



However,we can see that the the 95% and 99% CI are concentrated in `0.35 - 0.65` range. Conversely, they are concentrated in `18.5 - 21.5` range. Clearly, the `x_value` range has been reduced in `distr_func = runif`.

Question 3

a) What is the "average" booking time for new members making their first restaurant booking?

i) Use traditional statistical methods to estimate the population mean of minday, its standard error, and the 95% confidence interval (CI) of the sampling means

```
mean <- mean(minday)
sd_error <- sd(minday) / ( length(minday)^0.5 )
ci95_low <- mean - 1.96 * sd_error
ci95_high <- mean + 1.96 * sd_error
```

The mean value is

```
## 942.49635
```

The standard deviation error is

```
## 0.599767314943967
```

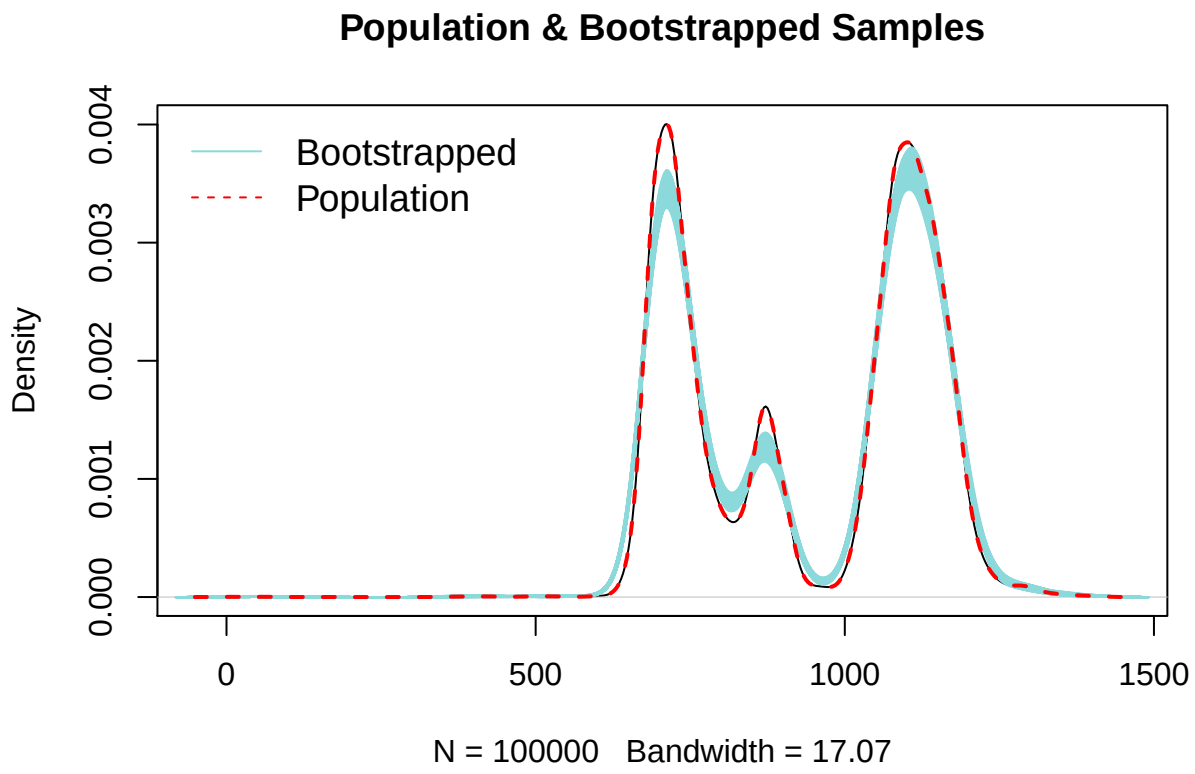
The 95% CI is between

```
## 941.32080606271 & 943.67189393729
```

ii) Bootstrap to produce 2000 new samples from the original sample

```
bootstrapped_samples <- replicate(2000, sample(minday, 10000, replace=TRUE))
```

iii) Visualize the means of the 2000 bootstrapped samples



iv) Estimate the 95% CI of the bootstrapped means.

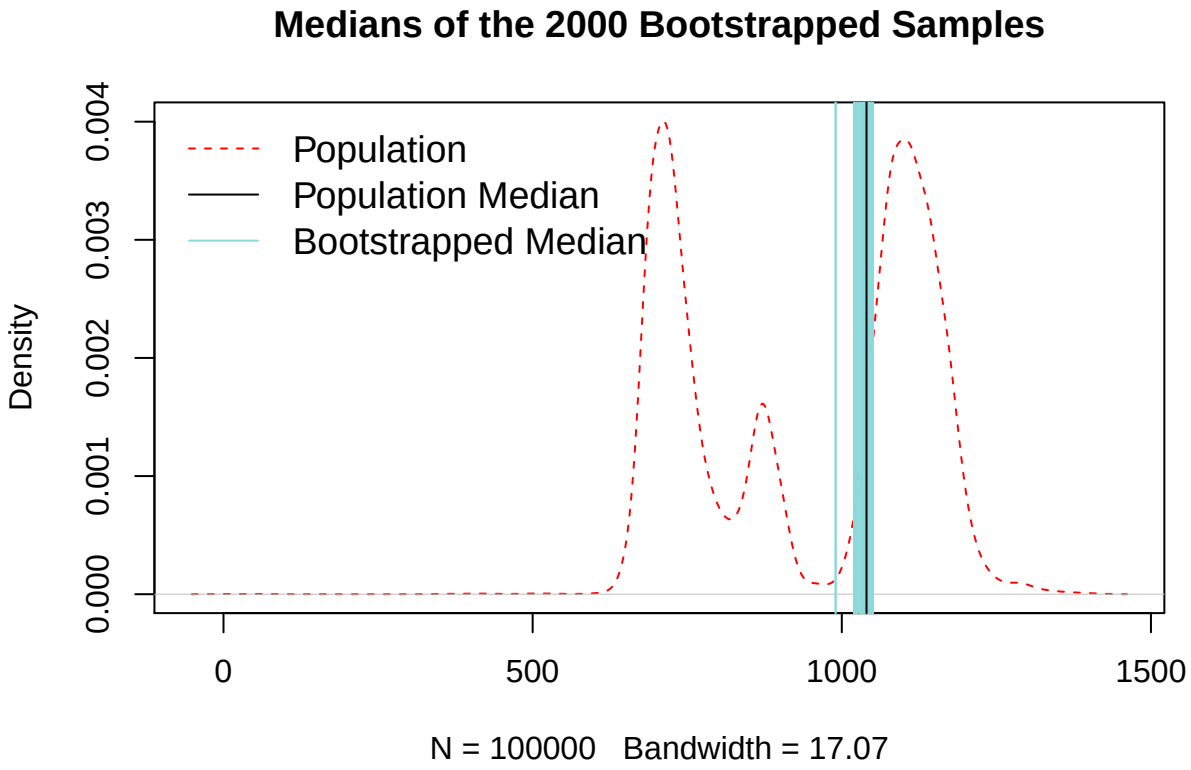
```
## The 95% CI is between 942.324859309738 and 942.491102690262
```

b) By what time of day, have half the new members of the day already arrived at their restaurant?

i) Estimate the median of minday

```
## [1] 1040
```

ii) Visualize the medians of the 2000 bootstrapped samples



iii) Estimate the 95% CI of the bootstrapped medians.

```
## The 95% CI is between 1035.21324801967 and 1036.43425198033
```