HW3

109006206

helped by : 109060082

Question 1

a) Create a normal distribution (mean=940, sd=190) and standardize it (let's call it rnorm_std)

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

Answer:

We should expect the mean is 0 and standard deviation is 1 because it was standardized.

```
standardize <-function(numbers) {
  numbers <-(numbers -mean(numbers)) / sd(numbers)
  return(numbers)
  }
temp <- rnorm(1000,mean = 940, sd = 190)
rnorm_std <- standardize(temp)
print(paste("Mean :",mean(rnorm_std)))
```

```
## [1] "Mean : -2.4520836749975e-16"
```

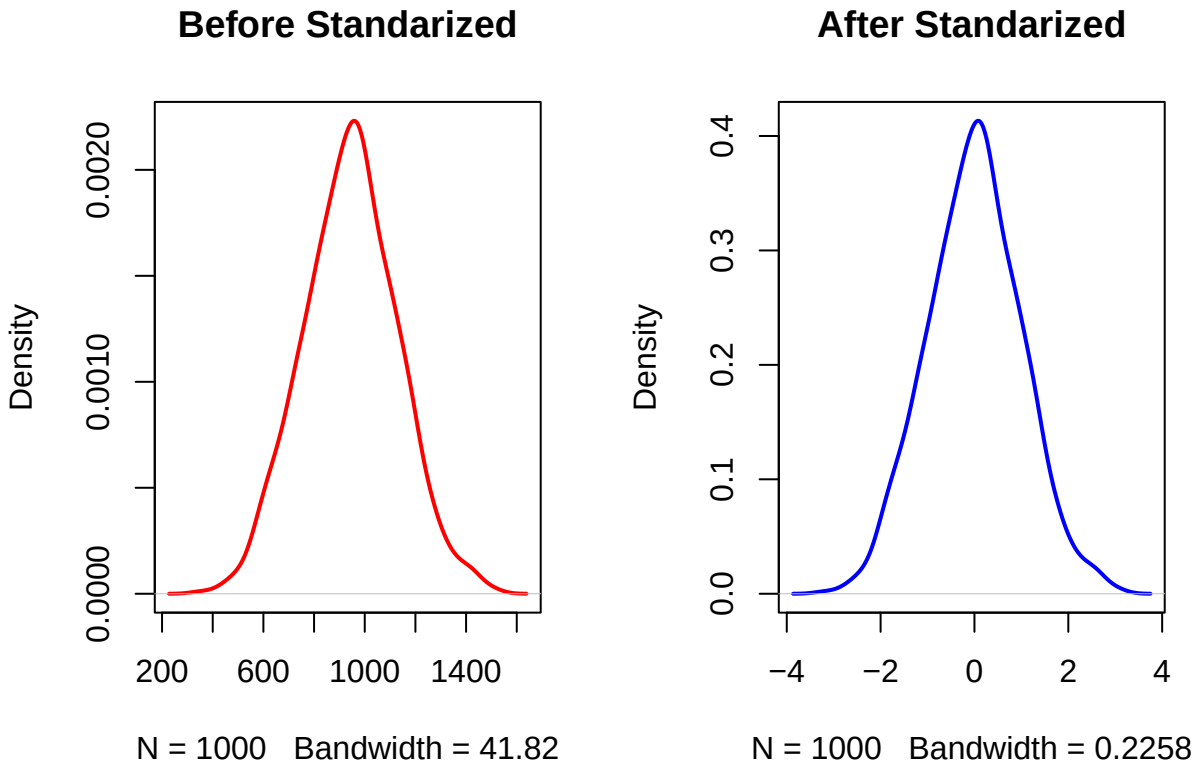
```
print(paste("Standard Deviation:",sd(rnorm_std)))
```

```
## [1] "Standard Deviation: 1"
```

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

Answer:
It should also be a normal distribution since "Standardization" shifts and scales a distribution and won't change it's shape.

```
par(mfrow=c(1,2))
plot(density(temp),lwd=2,col="red",main="Before Standarized")
plot(density(rnorm_std),lwd=2,col="blue",main="After Standarized")
```



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

Answer:
It's called Standard Normal Distributions

b) Create a standardized version of minday discussed in question 3 (let's call it minday_std)

```
bookings <- read.table("first_bookings_datetime_sample.txt", header=TRUE)
hours  <- as.POSIXlt(bookings$datetime, format="%m/%d/%Y %H:%M")$hour
mins   <- as.POSIXlt(bookings$datetime, format="%m/%d/%Y %H:%M")$min
minday <- hours*60 + mins
```

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

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

Answer:
We should expect the mean is 0 and standard deviation is 1 because it was standardized

```
print(paste("Mean :",mean(minday_std)))
```

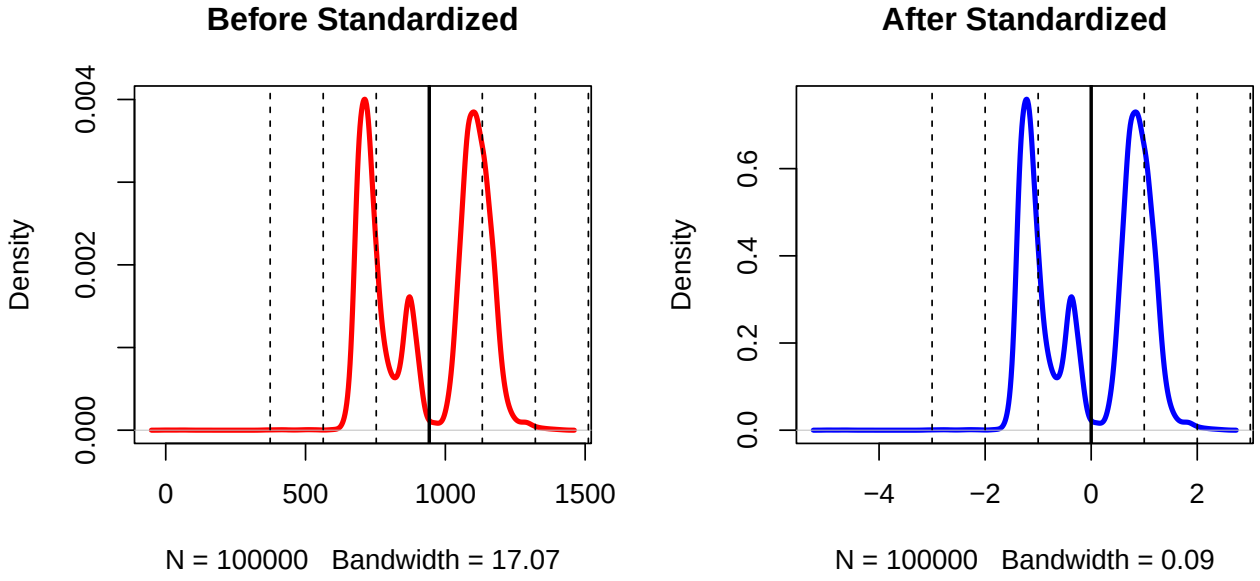
```
## [1] "Mean : -4.25589034500073e-17"
```

```
print(paste("Standard Deviation:",sd(minday_std)))
```

```
## [1] "Standard Deviation: 1"
```

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

```
par(mfrow=c(1,2))
plot(density(minday),lwd=3,col="red",main="Before Standardized")
abline(v=mean(minday),lwd = 2)
lines <- seq(-3,-1)
abline(v=mean(minday)-sd(minday)*lines,lty = "dashed")
lines2 <- seq(1,3)
abline(v=mean(minday)-sd(minday)*lines2,lty = "dashed")
plot(density(minday_std),lwd=3,col="blue",main="After Standardized")
abline(v=mean(minday_std),lwd = 2)
lines <- seq(-3,-1)
abline(v=mean(minday_std)-sd(minday_std)*lines,lty = "dashed")
lines2 <- seq(1,3)
abline(v=mean(minday_std)-sd(minday_std)*lines2,lty = "dashed")
```



Question 2

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

plot_sample_ci(num_samples = 100, sample_size = 100, pop_size=10000, distr_func=rnorm, mean=20, sd=3)

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

Answer:

```
library("compstatslib")
n_samples<-100
cat("Number of samples we expect to NOT include the population mean in its 95% CI :"
    ,n_samples*0.05)
```

```
## Number of samples we expect to NOT include the population mean in its 95% CI : 5
```

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

Answer:

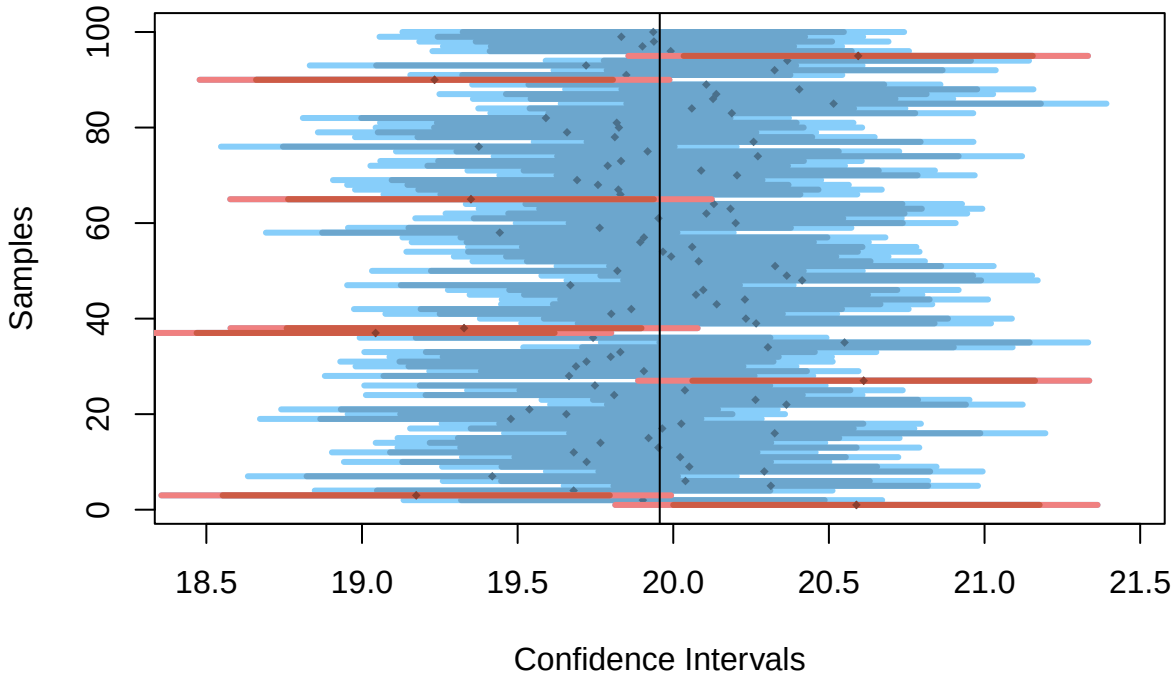
```
n_samples<-100
cat("Number of samples we expect to NOT include the population mean in its 99% CI :"
    ,n_samples*0.01)
```

```
## Number of samples we expect to NOT include the population mean in its 99% CI : 1
```

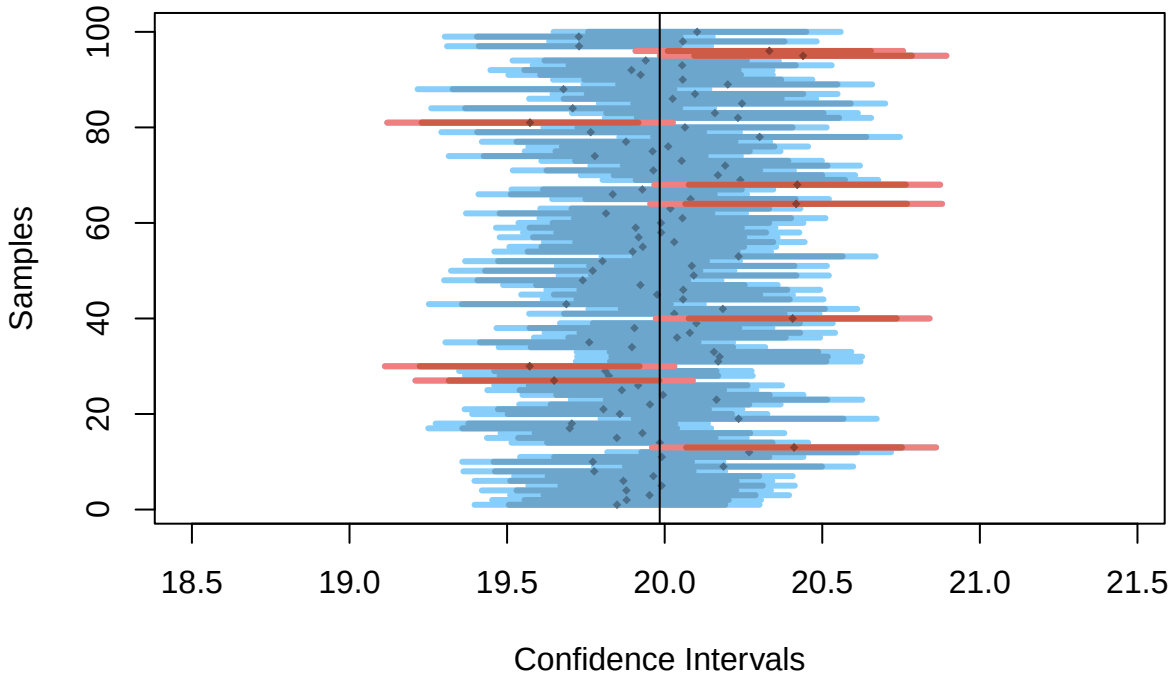
b) Rerun the previous simulation with the same number of samples, but larger sample size (sample_size=300):

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?

```
plot_sample_ci(num_samples = 100, sample_size = 100, pop_size=10000,
               distr_func=rnorm, mean=20, sd=3)
```




```
plot_sample_ci(num_samples = 100, sample_size = 300, pop_size=10000,
               distr_func=rnorm, mean=20, sd=3)
```



We can see that the 95% and 99% CI become more narrower as the sample size increases

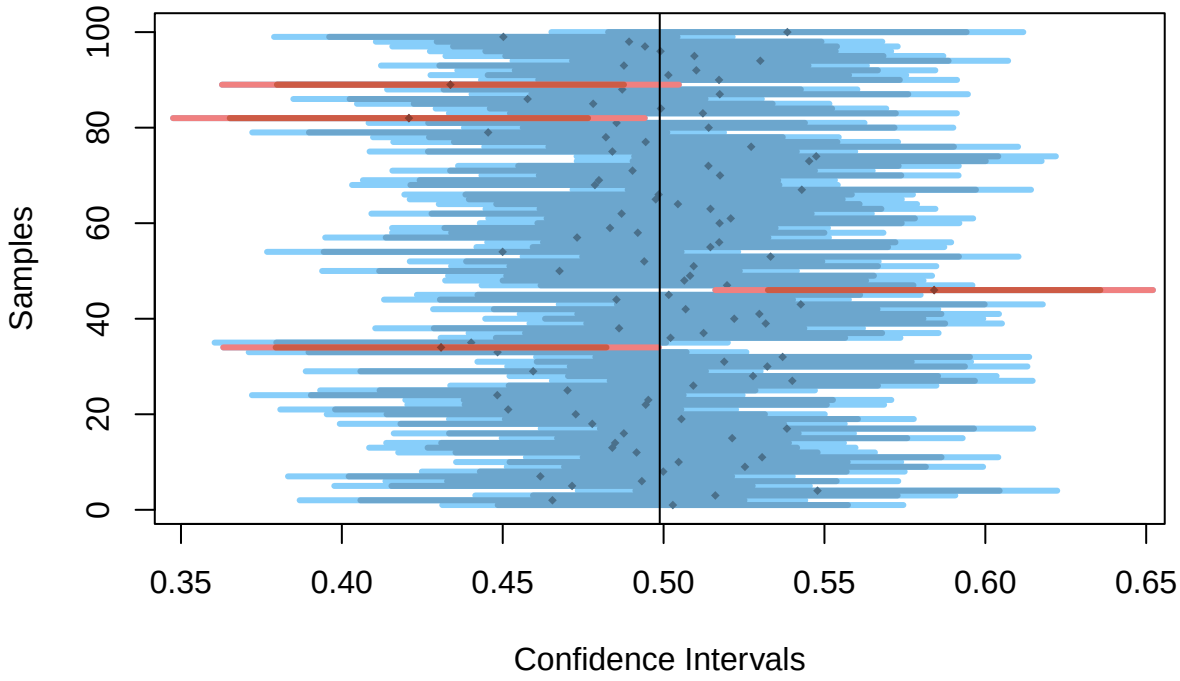
ii) This time, how many samples (out of the 100) would we expect to NOT include the population mean in its 95% CI?

```
n_samples<-100
cat("Number of samples we expect to NOT include the population mean in its 95% CI :"
    ,n_samples*0.05)
```

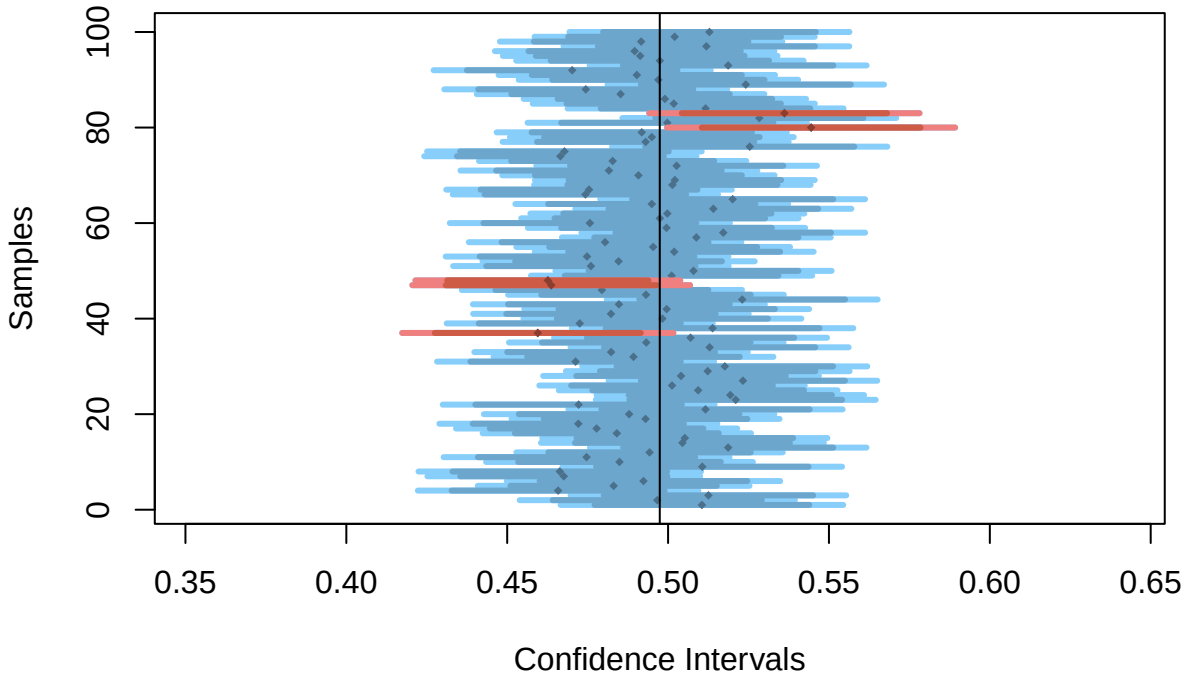
```
## Number of samples we expect to NOT include the population mean in its 95% CI : 5
```

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

```
plot_sample_ci(num_samples = 100, sample_size = 100, pop_size=10000,
               distr_func=runif)
```




```
plot_sample_ci(num_samples = 100, sample_size = 300, pop_size=10000,
               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_minday<-mean(minday)
sd_error<-sd(minday)/(length(minday)^0.5)
ci95_low <- mean_minday-(1.96*sd_error)
ci95_high<- mean_minday+(1.96*sd_error)
cat("Mean :",mean_minday)
```

```
## Mean : 942.4964
```

```
cat("Standard Deviation Error :",sd_error)
```

```
## Standard Deviation Error : 0.5997673
```

```
cat("95% Confidence Interval :",ci95_low," to ",ci95_high)
```

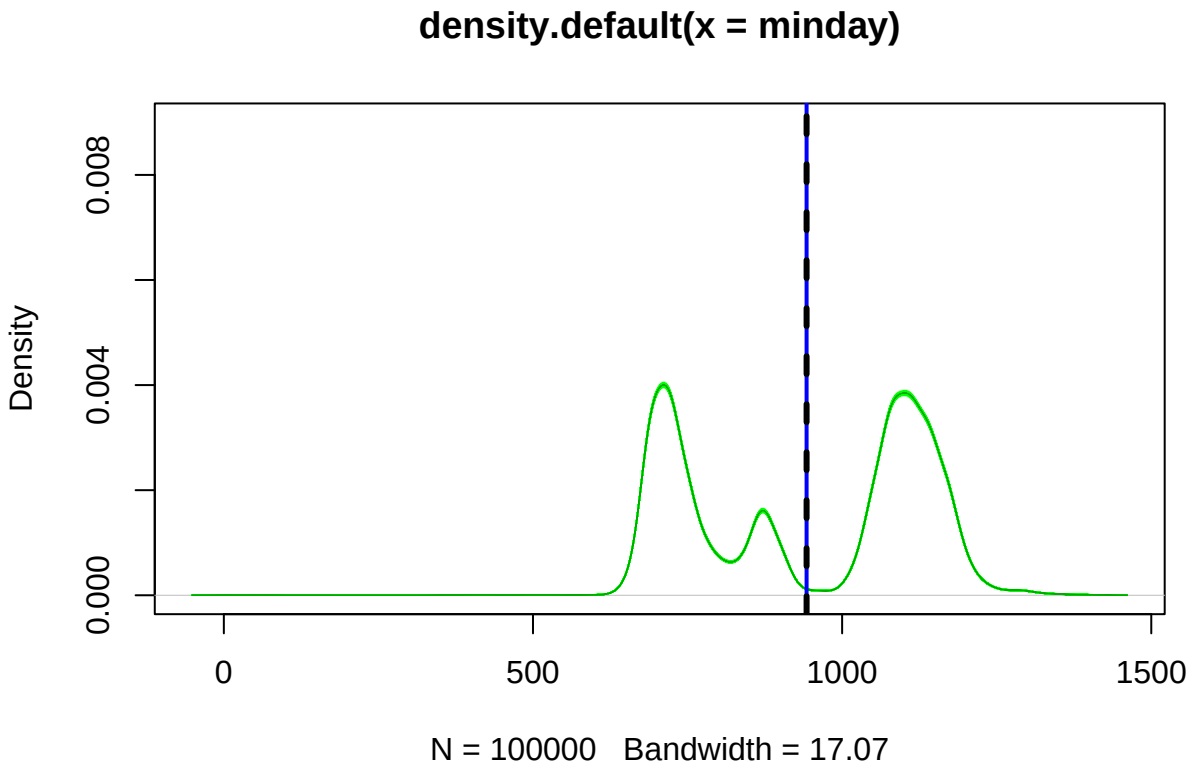
```
## 95% Confidence Interval : 941.3208  to  943.6719
```

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

```
resamples<-replicate(2000,sample(minday,length(minday),replace = TRUE))
```

iii) Visualize the means of the 2000 bootstrapped samples

```
plot(density(minday), lwd=0, ylim=c(0, 0.009))
# Draws lines for each sampling mean
plot_resample_density<-function(sample_i) {
  lines(density(sample_i), col=rgb(0.0,1, 0.0, 0.01))
  return(mean(sample_i))
  }
sample_means<-apply(resamples, 2, FUN=plot_resample_density)
abline(v=mean(sample_means),col="blue",lwd=2)
abline(v=mean(minday),lty="dashed",lwd=3)
```



iv) Estimate the 95% CI of the bootstrapped means using the quantile function

```
quantile(sample_means, probs=c(0.025, 0.975))
```

```
##     2.5%    97.5%
## 941.3572 943.6727
```

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

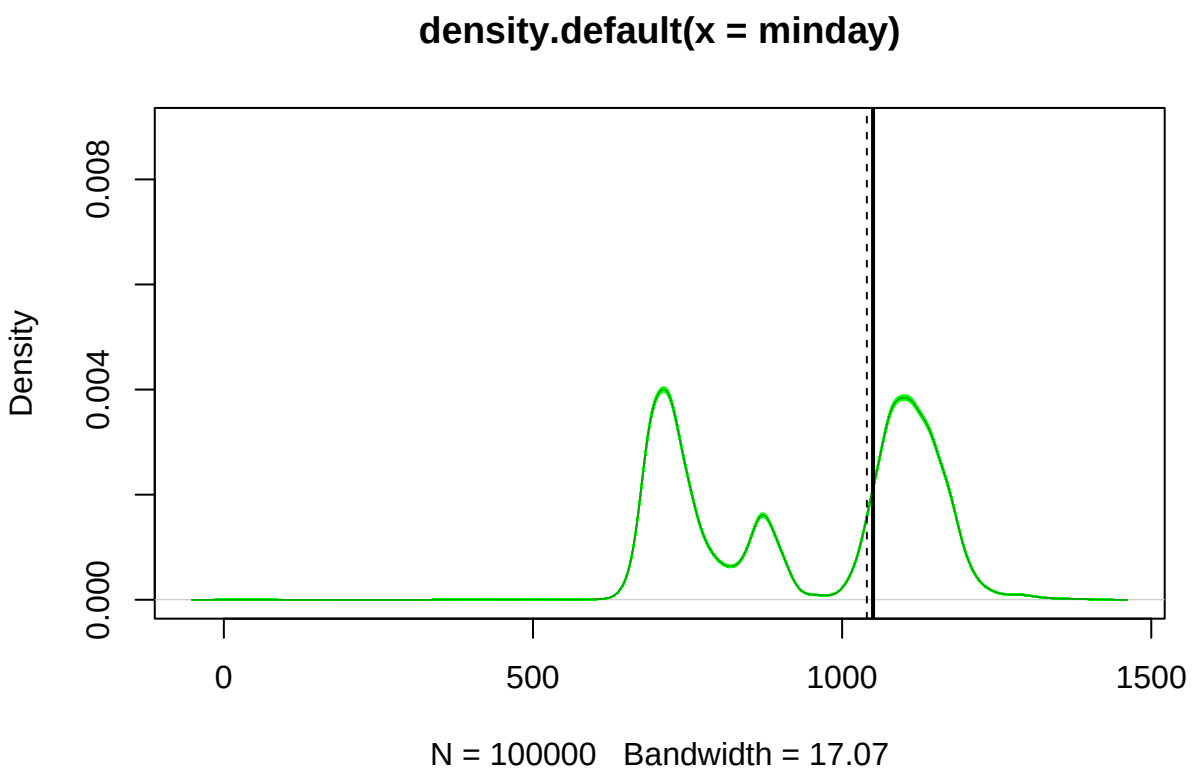
i) Estimate the median of minday

```
median(minday)
```

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

ii) Visualize the medians of the 2000 bootstrapped samples

```
resamples2<-replicate(2000,sample(minday,length(minday),replace = TRUE))
plot(density(minday), lwd=0, ylim=c(0, 0.009))
# Draws lines for each sampling mean
plot_resample_density2<-function(sample_i) {
  lines(density(sample_i), col=rgb(0.0, 1, 0.0, 0.01))
  return(median(sample_i))
  }
sample_means2<-apply(resamples2, 2, FUN=plot_resample_density2)
abline(v=median(sample_means2), lwd=2)
abline(v=median(minday), lty="dashed")
```



iii) Estimate the 95% CI of the bootstrapped medians using the quantile function

```
quantile(sample_means2, probs=c(0.025, 0.975))
```

```
##  2.5% 97.5%
##  1020  1050
```