Statistics & Bootstrapping – Part 2

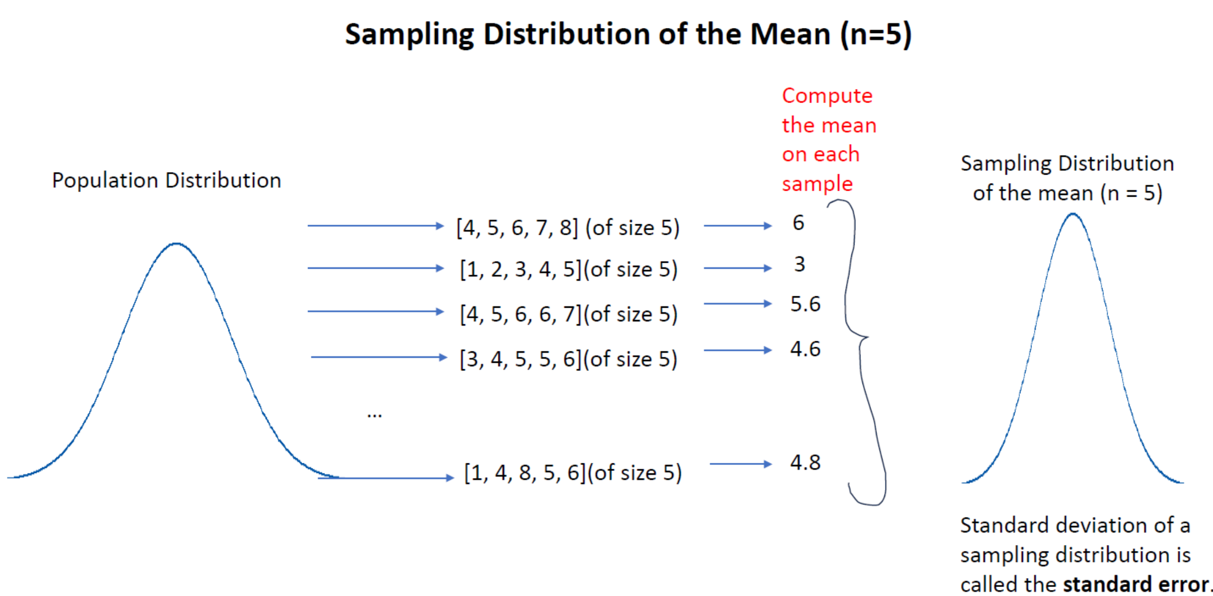
Sampling With Replacement: When you sample with replacement, the object you selected is put back into the pool before another object is sampled.

Example:

```
vector <- 1:6
sample(vector,  3,  replace = TRUE)
```

Output:

Statistical Foundations



Standard Error: it's the standard deviation of a sampling distribution.

Review: Computing the Standard Error of a Sampling Distribution of the Mean with n = 100

Population

```
normal_pop <- rnorm(1000000, mean = 100, sd = 15)
```

Function

```
get_one_sample_mean  <- function(i, population_vector, n) {
  one_sample <- sample(population_vector, n)
  one_sample_mean <- mean(one_sample)
  return(one_sample_mean)
}
```

Sampling Distribution of the Mean with n = 100

```
sampling_distribution <- map_dbl(1:10000, get_one_sample_mean,
population_vector= normal_pop, n = 100)
```

Standard Error

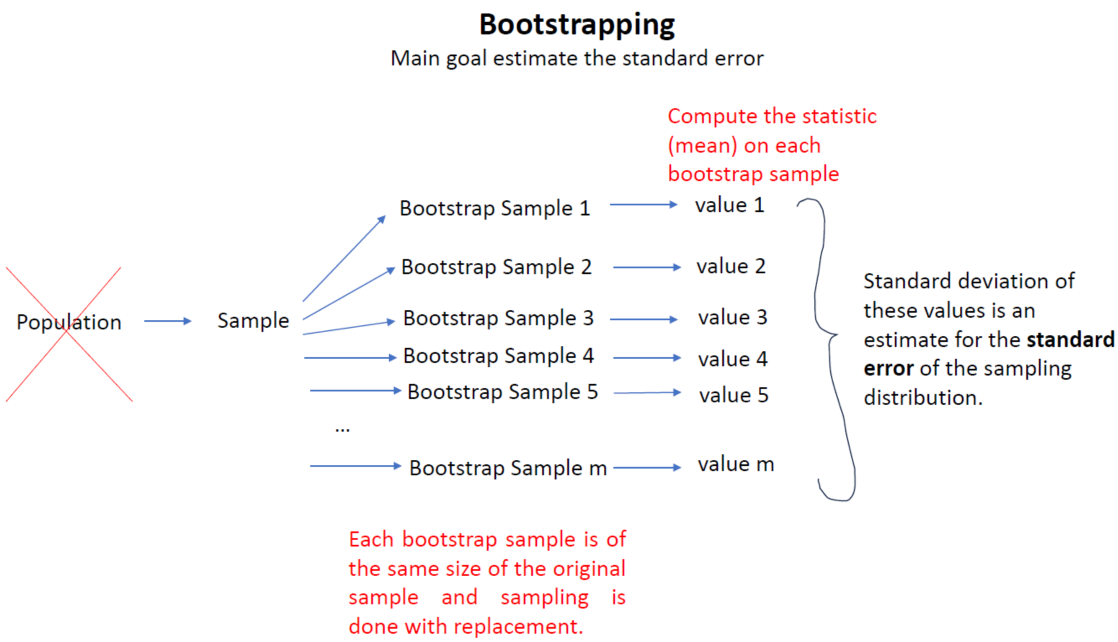
```
st_error <- sd(sampling_distribution)
st_error
```

Output: 1.501716

Bootstrapping

The bootstrap is a resampling technique used to estimate standard errors and confidence intervals for sample statistics. It's particularly useful when you don't know the underlying data distribution.

A bootstrap sample is a sample of the same size of the original sample and sampling is done with replacement.

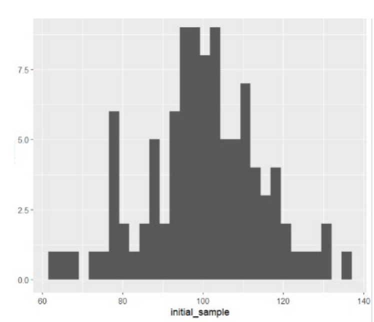


Step 1: Start with a sample.

```
initial_sample <- sample(normal_pop, 100)
```

Plot of the Initial Sample

```
sample_plot <- data.frame(initial_sample) %>%
  ggplot(aes(x=initial_sample)) +
  geom_histogram()
sample_plot
```



Step 2: Create a function that draws one bootstrap sample from a given sample and computes a statistic on the bootstrap sample. In this example the statistic is the mean.

```
boot_mean <- function(i, y){
  boot_sample <- sample(y, length(y), replace = TRUE)
  value <- mean(boot_sample)
  return(value)
}
```

Step 3: Using the function from Step 2, draw 10000 bootstrap samples and compute the mean on each.

```
all_values <- map_dbl(1:10000, boot_mean, y = initial_sample)
```

Step 4: Compute the standard deviation of the 10000 bootstrap sample means from Step 3. That is the estimated standard error.

```
sd(all_values)
```

Output: 1.435214