QTM 220 HW #1

Author

Veronica Vargas

Exercise #1

Playing with a Gaussian Distribution

(a) Standard Normal Distribution

```
set.seed(42)
sample <- rnorm(100, 0, 1)
```

n = the number of observations (here we see 100 observations).

mean = the average of the data set that is found by dividing the sum of the observations by the total number of observations (the mean is 0 for a normally distributed df).

sd = quantifies the spread of the data, otherwise known as the standard distribution, and is always a positive number.

(b) Summary Statistics

```
mean(sample)
```

```
[1] 0.03251482
```

```
median(sample)
```

```
[1] 0.08979677
```

```
sd(sample)
```

```
[1] 1.041357
```

```
mad(sample)
```

```
[1] 0.936786
```

(c) Plotting the Distribution

```
library(tidyverse)
```

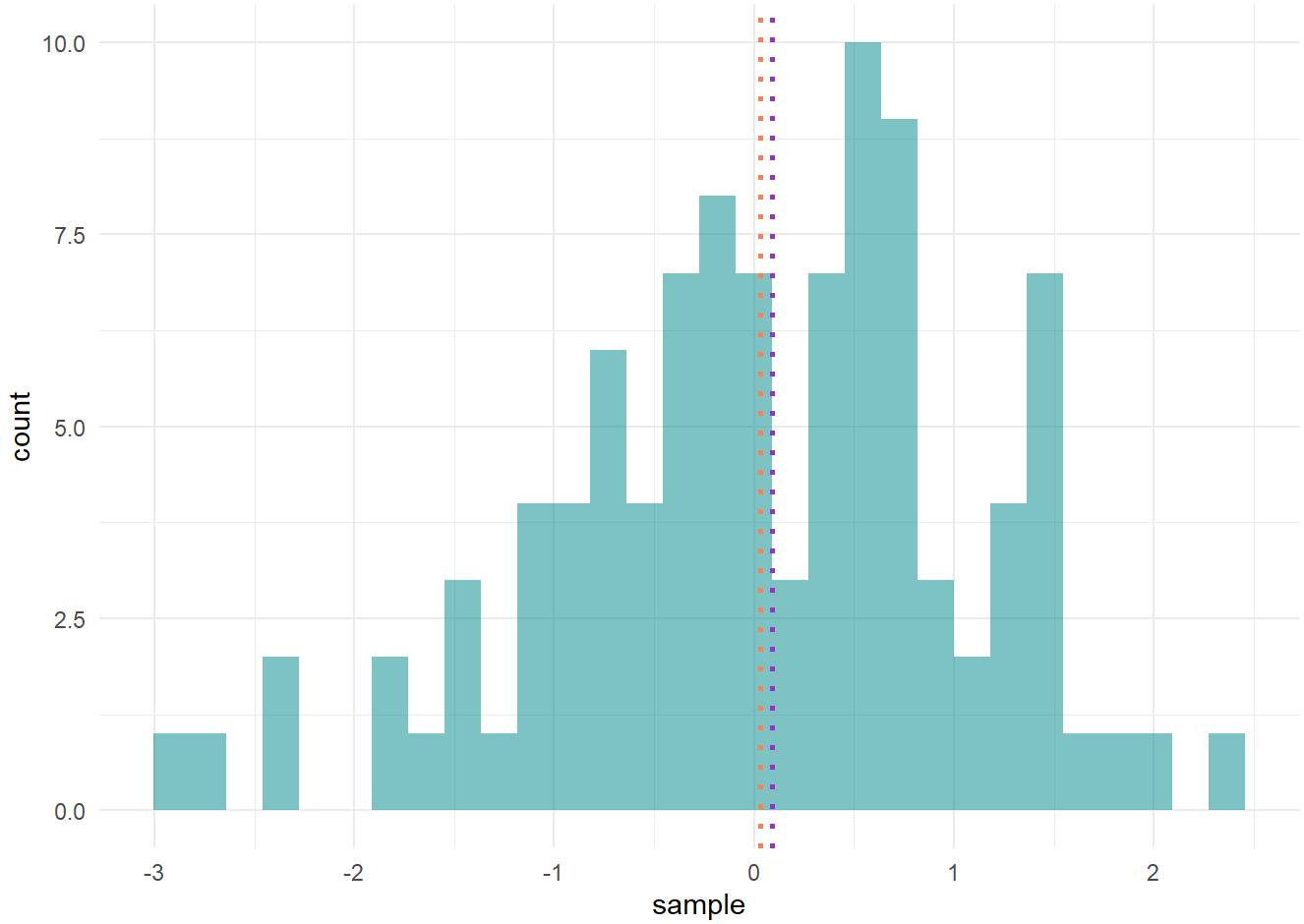
```
Warning: package 'tidyverse' was built under R version 4.3.3
```

```
── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
✓ dplyr     1.1.3     ✓ readr     2.1.4
✓ forcats   1.0.0     ✓ stringr   1.5.0
✓ ggplot2   3.4.3     ✓ tibble    3.2.1
✓ lubridate 1.9.2     ✓ tidyr     1.3.0
✓ purrr     1.0.2
```

```
── Conflicts ─────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
ggplot(data = data.frame(sample = sample), aes(x = sample)) +
  geom_histogram(fill = "cyan4", alpha = 0.5, position = "identity") +
  geom_vline(xintercept = mean(sample), linetype="dotted",
             color = "coral", linewidth=1) +
  geom_vline(xintercept = median(sample), linetype="dotted",
             color = "darkorchid", linewidth=1) +
  theme_minimal()
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



(d) Describe the Distribution

This is a bimodal distribution due the presence of two local maximums. Otherwise, the data is close to a standard normal distribution since the highest values (mode) are almost symmetrical along the mean and the median. That said, due to there being more than one mode, this is not a standard normal distribution.

(e) Comparing Values

The mean obtained in (b) is larger than the expected mean of 0 from part (a). They are not the same. This is because the data is generated at random and will therefore feature some variation. Furthermore, while the value in (b) is larger than 0, it is very small and is relatively close to zero.

Exercise #2

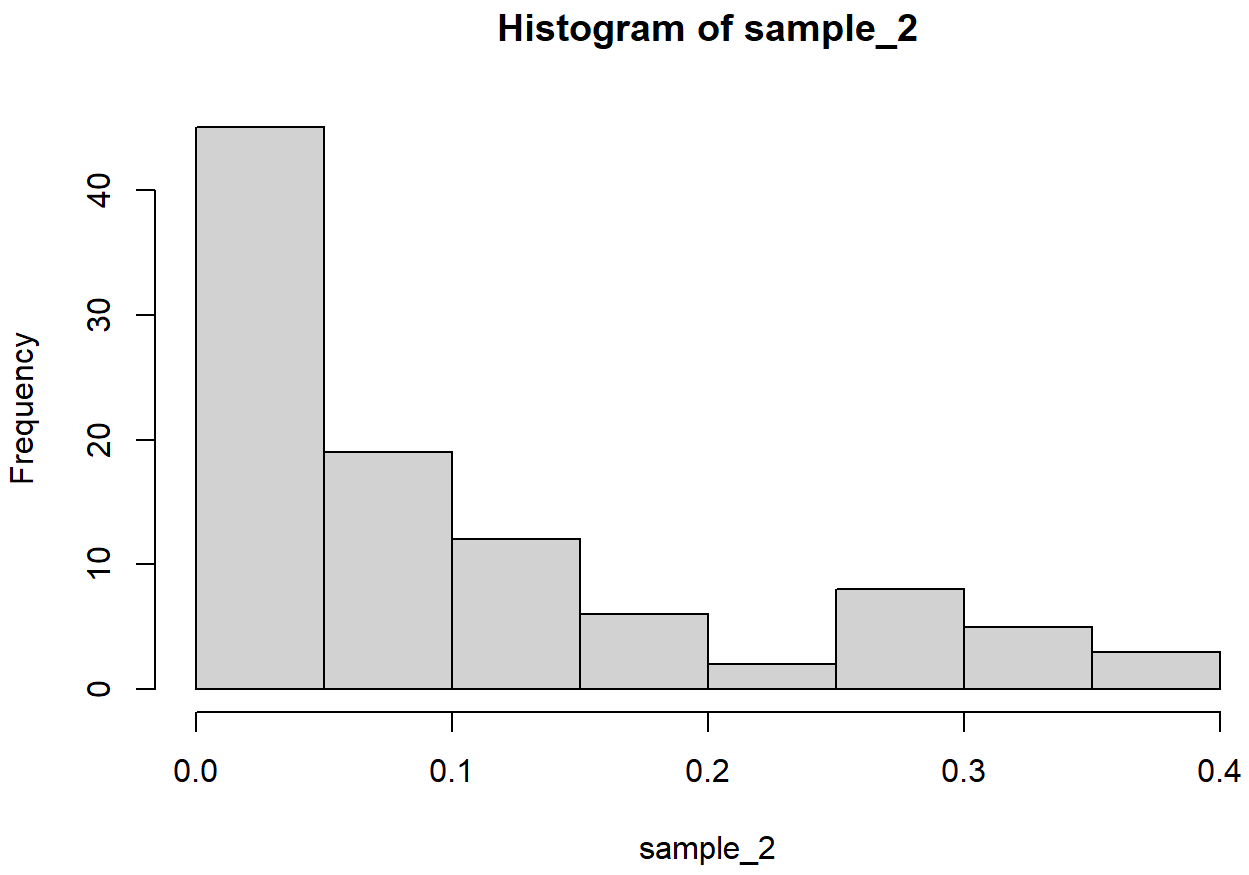
Pearson's Seconds Skewness Coefficient

(a) Computing Pearson's Second Skewness Coefficient

```
custom.pearson <- function(x) {
  mean <- mean(x)
  median <- median(x)
  pearson <- (3*(mean - median))/sd(x)
  pearson
}
```

(b) Simulating Beta Distribution rbeta(1000, 0.45, 5)

```
set.seed(42)
sample_2 <- rbeta(100, 0.45, 5)
hist(sample_2)
```

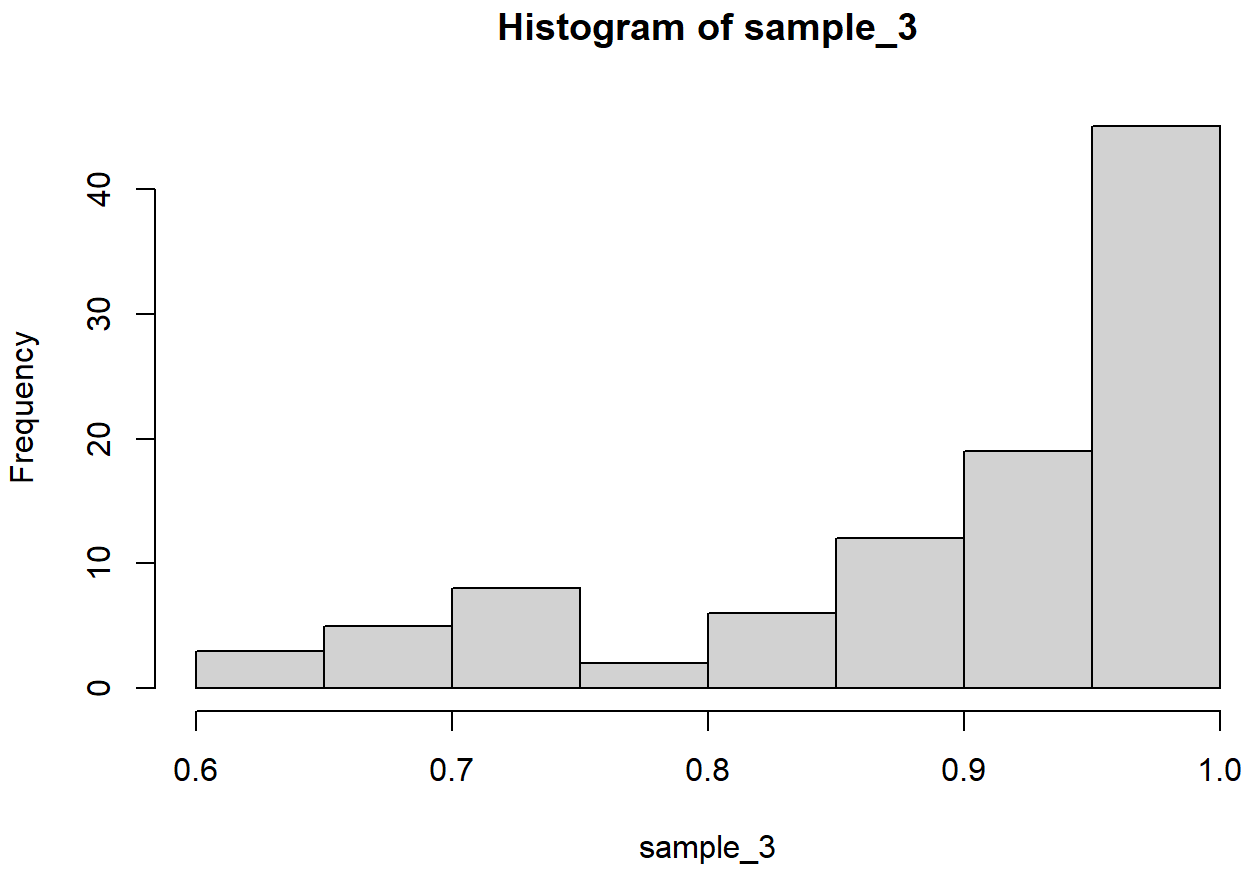



```
custom.pearson(sample_2)
```

```
[1] 1.139
```

(c) Simulating Beta Distribution rbeta(1000, 5, 0.45)

```
set.seed(42)
sample_3 <- rbeta(100, 5, 0.45)
hist(sample_3)
```




```
custom.pearson(sample_3)
```

```
[1] -1.139
```

(d) Describe & Compare

Both histograms feature the same value for Pearson's Second Skewness Coefficients but they are opposite in sign. More specifically, the histogram in (b) has a positive skewness and the histogram in (c) has a negative skewness. Also, both of these values have an absolute value greater than one, meaning that both distributions are heavily skewed.

Exercise #5

Sampling Distributions and Estimators

```
pokemon <- read.csv("C:/Users/13015/OneDrive - Emory University/Documents/Fall 2024/QTM
        220/pokemon.csv")
```

(a) Draw Random Sample w/ size n = 100

```
set.seed(42)
pokemon_sample <- sample(pokemon, 100, replace = T)
mean(pokemon_sample$hp)
```

```
[1] 68.9588
```

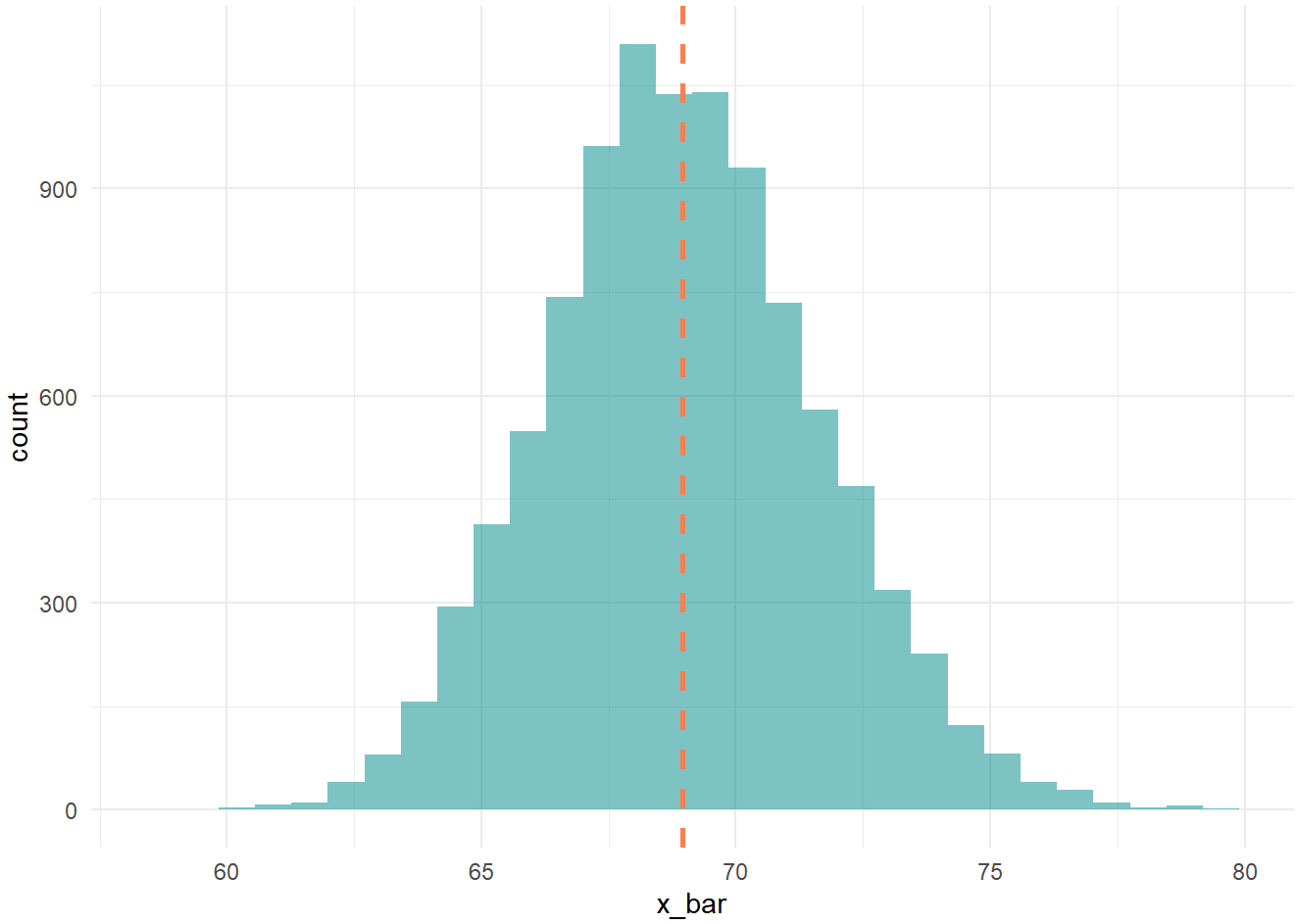
(b) Create Sampling Distribution of Sample

```
n <- 10000
x_bar <-  rep(NA, n)

for(i in 1:n){
  sampled.hp <- sample(pokemon_sample$hp, 100, replace = T)
  x_bar[i] <- mean(sampled.hp)
}
```

```
ggplot(data = data.frame(x_bar = x_bar), aes(x = x_bar)) +
  geom_histogram(fill = "cyan4", alpha = 0.5, position = "identity") +
  geom_vline(xintercept = mean(x_bar), linetype="dashed", color = "coral", linewidth=1) +
  theme_minimal()
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

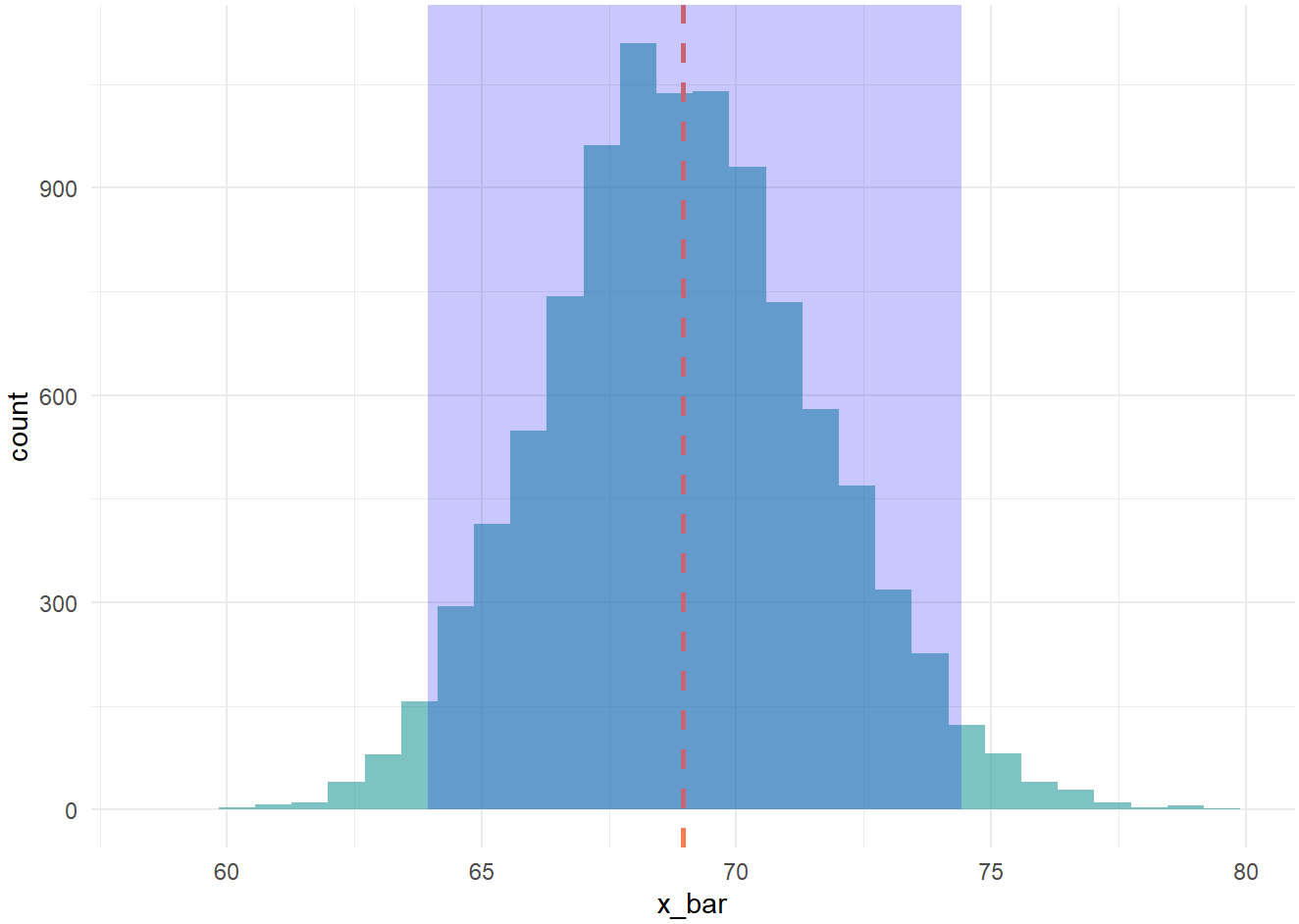


(c) Middle 95% of Sampling Distribution

```
lower_bound <- quantile(x_bar, 0.025)
upper_bound <- quantile(x_bar, 0.975)
```

```
ggplot(data = data.frame(x_bar = x_bar), aes(x = x_bar)) +
  geom_histogram(fill = "cyan4", alpha = 0.5, position = "identity") +
  geom_vline(xintercept = mean(x_bar), linetype="dashed", color = "coral", linewidth=1) +
  annotate("rect", xmin = lower_bound,
                   xmax = upper_bound,
                   ymin = 0, ymax = Inf, fill = "blue",
                   alpha = .2) +
  theme_minimal()
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



We would not know whether is covered the population mean since we do not know our estimand.