

Check-in 7

[Code](#)

AUTHOR

Maxim Dokukin

Remember, **follow the instructions below and use R Markdown to create a pdf document with your code and answers to the following questions on Gradescope.** You may find a template file by clicking "Code" in the top right corner of this page.

Load the flips data using the following code:

```
library(tidyverse)
flips <- read_csv("https://math167r-s24.github.io/static/flips.csv")
head(flips)
```

A tibble: 6 × 5

	A	B	C	D	E
	<chr>	<chr>	<chr>	<chr>	<chr>
1	T	H	H	T	H
2	H	H	H	H	H
3	T	H	T	H	H
4	H	T	H	H	H
5	H	T	T	H	T
6	T	H	T	T	H

```
flips_long <- flips |>
  mutate(id = 1:200) |>
  pivot_longer(-id, names_to = "Sequence", values_to = "Flip")
```

1. Design your own hypothesis test to try to identify the sequence of real flips. Design your own test statistic and simulate the null distribution of your test statistic. Can you identify which sequence is the real one?

Test 1 - Longest Streak

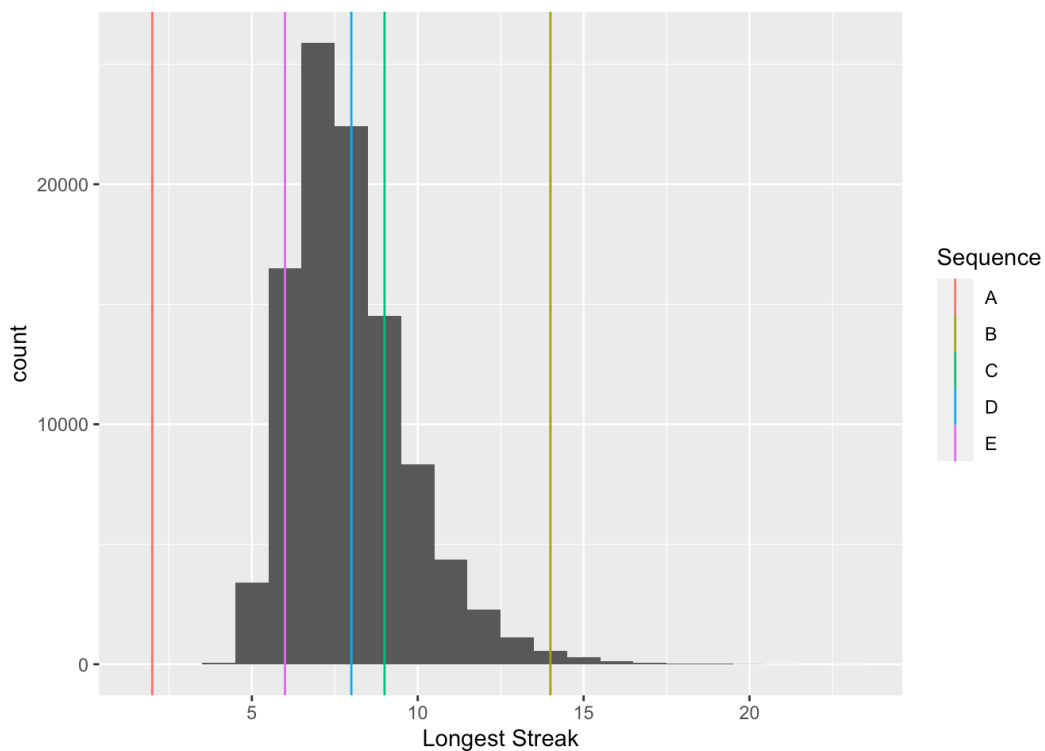
```

long_flip_streaks <- flips_long |>
  group_by(Sequence) |>
  summarize(longest_streak = max(rle(Flip)$length))

long_sim_streaks <-
  data.frame(
    x = replicate(100000,
                  max(rle(sample(c("H", "T"), size = 200, replac
    )
  )

ggplot() +
  geom_histogram(data = long_sim_streaks, aes(x = x), binwidth = 1) +
  geom_vline(data = long_flip_streaks,
             aes(color = Sequence, xintercept = longest_streak)) +
  xlab("Longest Streak")

```



```

long_p_vals <- sapply(long_flip_streaks$longest_streak,
                      function(x) mean(long_sim_streaks$x < x))
print(long_p_vals)

```

```
[1] 0.00000 0.98875 0.68288 0.45885 0.03476
```

```
long_dist_from_center <- abs(long_p_vals - 0.5)
print(paste("Sample closest to simulated distribution:",
            c('A', 'B', 'C', 'D', 'E')[which.min(long_dist_from_center)]))
```

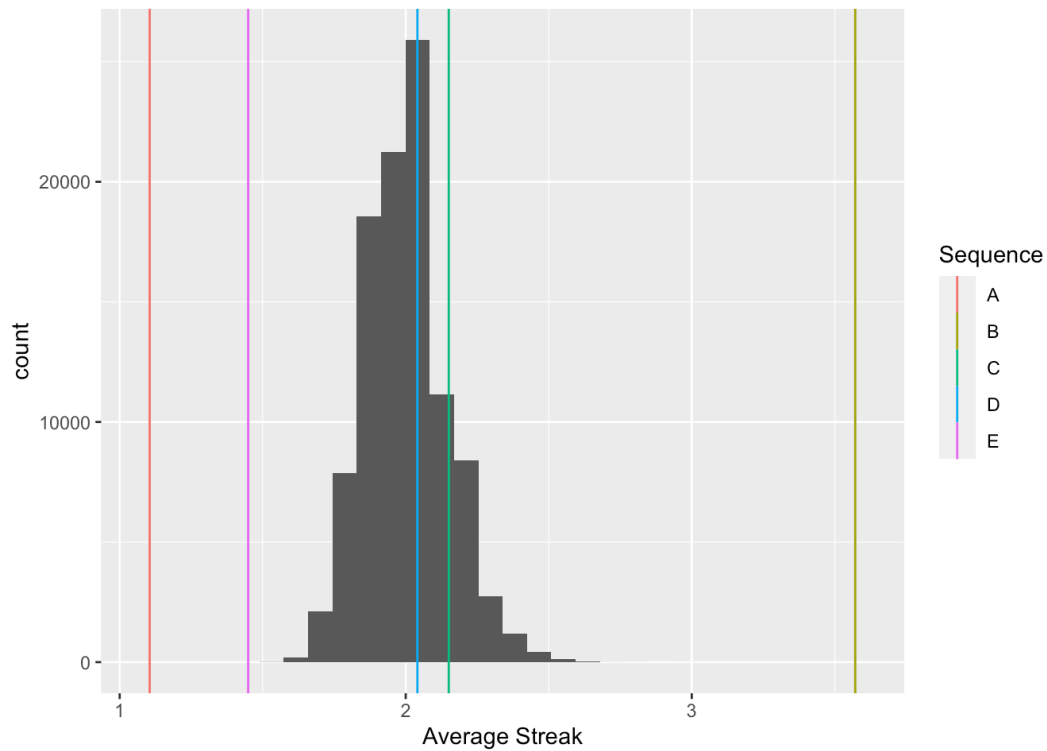
```
[1] "Sample closest to simulated distribution: D"
```

Test 2 - Average Streak

```
avg_flip_streaks <- flips_long |>
  group_by(Sequence) |>
  summarize(avg_streak = mean(rle(Flip)$length))

avg_sim_streaks <-
  data.frame(
    x = replicate(100000,
                  mean(rle(sample(c("H", "T"), size = 200, replace = TRUE))$length))
  )

ggplot() +
  geom_histogram(data = avg_sim_streaks, aes(x = x)) +
  geom_vline(data = avg_flip_streaks,
            aes(color = Sequence, xintercept = avg_streak)) +
  xlab("Average Streak")
```



```
avg_p_vals <- sapply(avg_flip_streaks$avg_streak,
                     function (x) mean(avg_sim_streaks$x < x))
print(avg_p_vals)
```

```
[1] 0.00000 1.00000 0.83882 0.60997 0.00000
```

```
avg_dist_from_center <- abs(avg_p_vals - 0.5)
print(paste("Sample closest to simulated distribution:",
            c('A', 'B', 'C', 'D', 'E')[which.min(avg_dist_from_center)]))
```

```
[1] "Sample closest to simulated distribution: D"
```

Test 3 - H/T Transition Count

```
trans_counts <- flips_long |>
```

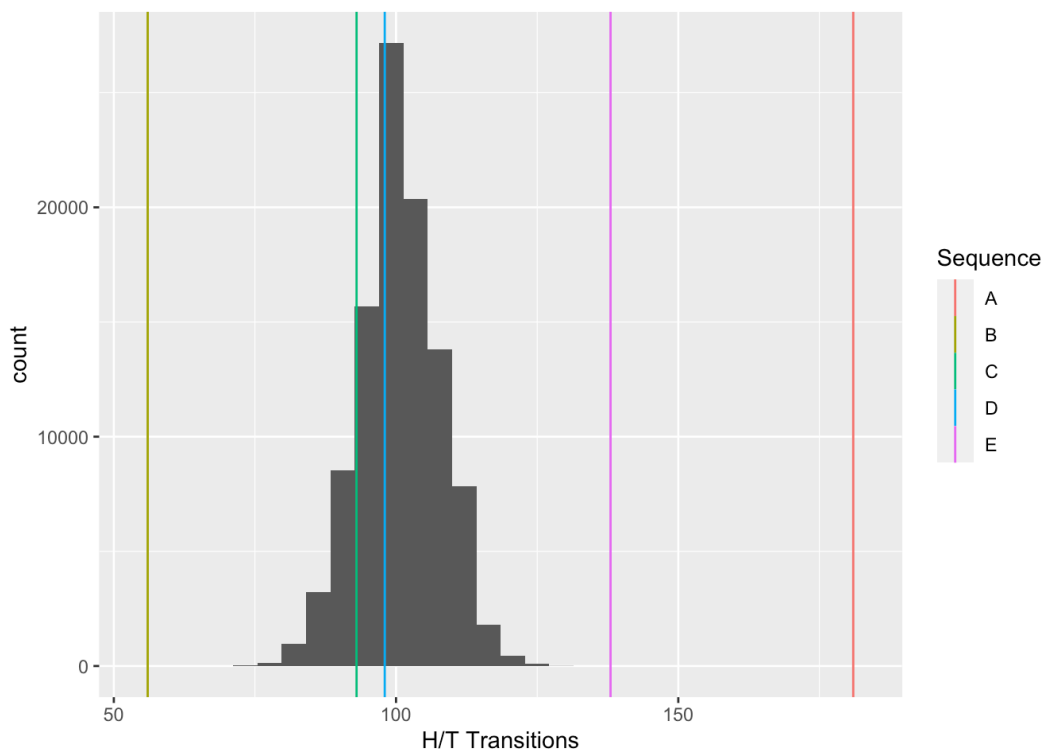
```

group_by(Sequence) |>
  summarize(transitions = length(rle(Flip)$length))

trans_counts_sim <-
  data.frame(
    x = replicate(100000,
                  length(rle(sample(c("H", "T"), size = 200, rep
    )
  )

ggplot() +
  geom_histogram(data = trans_counts_sim, aes(x = x)) +
  geom_vline(data = trans_counts,
             aes(color = Sequence, xintercept = transitions)) +
  xlab("H/T Transitions")

```



```

trans_p_vals <- sapply(trans_counts$transitions,
                       function(x) mean(trans_counts_sim$x < x))
print(trans_p_vals)

```

```
[1] 1.00000 0.00000 0.12844 0.33490 1.00000
```

```
trans_dist_from_center <- abs(trans_p_vals - 0.5)
```

```
print(paste("Sample closest to simulated distribution:",  
           c('A', 'B', 'C', 'D', 'E')[which.min(trans_dist_from_center)]))
```

```
[1] "Sample closest to simulated distribution: D"
```

Distance from center summary

```
total_dist <- long_dist_from_center + avg_dist_from_center + trans_dist_from_center  
  
print(paste("Sample closest to simulated distribution:",  
           c('A', 'B', 'C', 'D', 'E')[which.min(total_dist)]))
```

```
[1] "Sample closest to simulated distribution: D"
```