

V S N Sai Krishna Mohan Kocherlakota
Counting and Probability
24th October 2023

INTRODUCTION

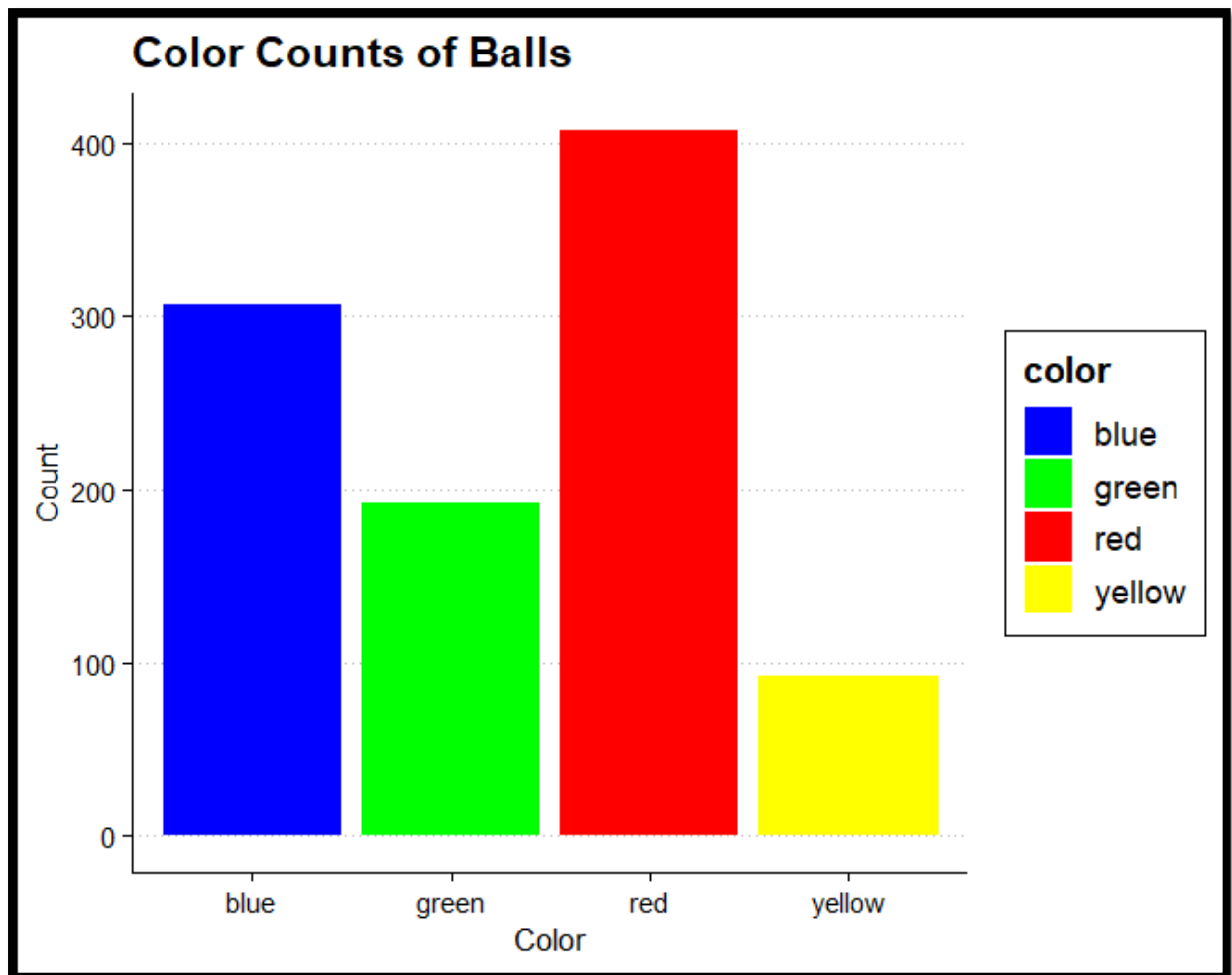
In this project, I utilized a single dataset to create bar plots and calculate the probability of each color or label based on the data. I successfully employed new functions such as 'count()' and 'choose()' to achieve these tasks. This project has deepened my understanding of the connection between statistics, probability, and programming.

KEY-FINDINGS

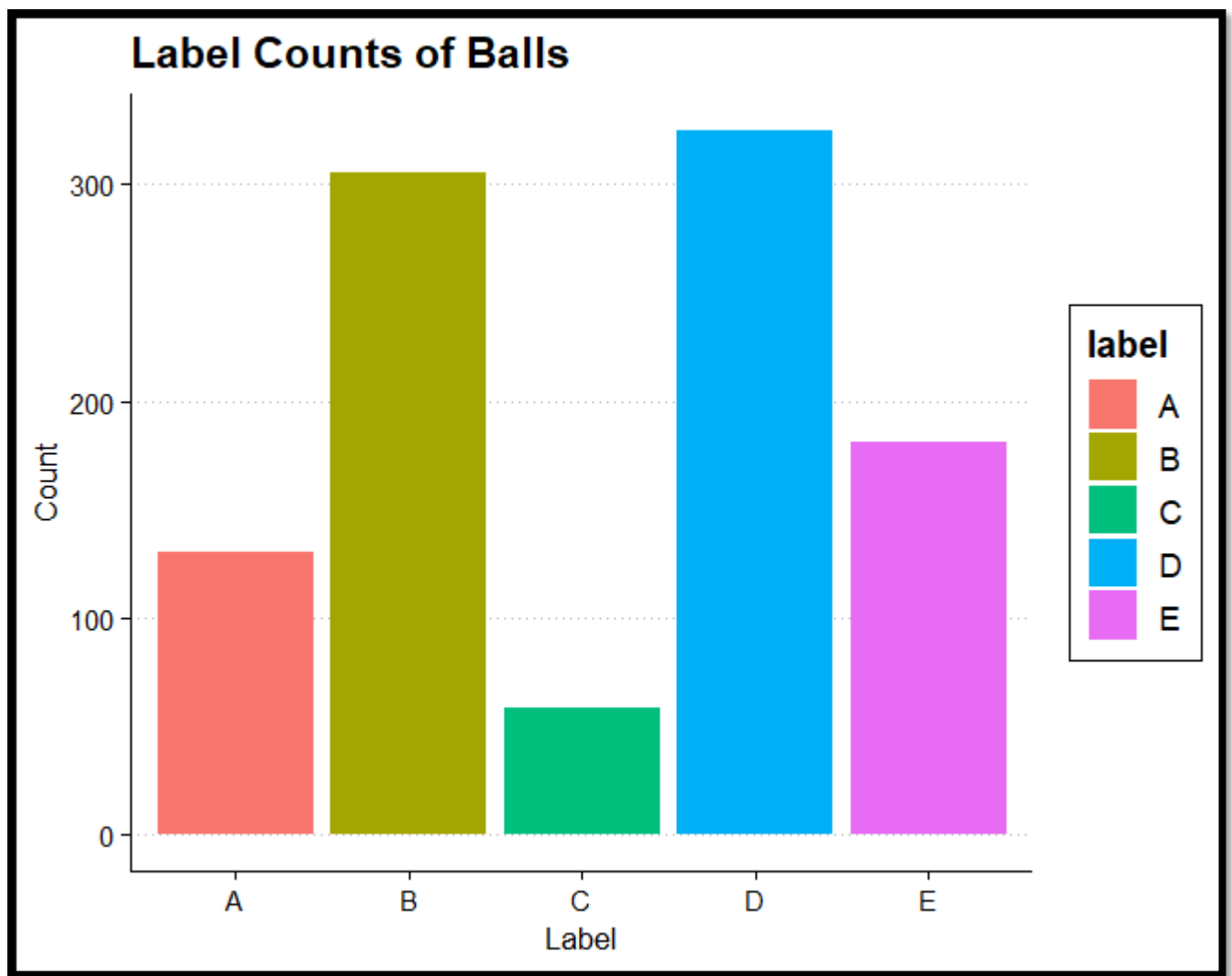
Part-1: - Ball Data

1.Problem 4 & 5: -

In this code, based on freq_color & freq_label data frames created bar plot in between color, frequency vs count of colors & labels respectively. I used ggplot from ggplot2 package and theme_clean() from ggthemes() package. Created manual color fill to achieve results required.



```
> colors_plot <- c("blue" = "blue", "green" = "green", "red" = "red", "yellow" = "yellow")
```



2.Problem 16: -

A factorial function in R. This function calculates the factorial of a non-negative integer `n`. Here's a breakdown of how the code works:

1. The base case checks if `n` is equal to 0. If `n` is 0, it returns 1 because the factorial of 0 is defined as 1.
2. The next conditional statement checks if `n` is less than 0. If `n` is negative, it returns -1. This is used to indicate that the input is not a valid non-negative integer.
3. If neither of the above conditions is met, the function recursively calculates the factorial by calling itself with the argument `n-1` and multiplying the result by `n`.

```
> factorial <- function(n){
+   if (n==0){
+     return(1)
+   } else if (n < 0){
+     return(-1)
+   } else {
+     return(n*factorial(n-1))
+   }
+ }
```

```
> factorial(0)
[1] 1
> factorial(3)
[1] 6
> factorial(5)
[1] 120
> factorial(-10)
[1] -1
```

Part-2: - Coin Flipping Data

3.Problem 18 & 19: -

Created a data frame `coin_outcomes` representing the possible outcomes of four coin flips, with each column representing the result of each flip. It also defines a function `calculate_probability` that calculates the probability of a particular outcome based on a given probability of getting heads (`heads_prob`). The code then applies this function to each row of the `coin_outcomes` data frame and adds the resulting probabilities as a new column.

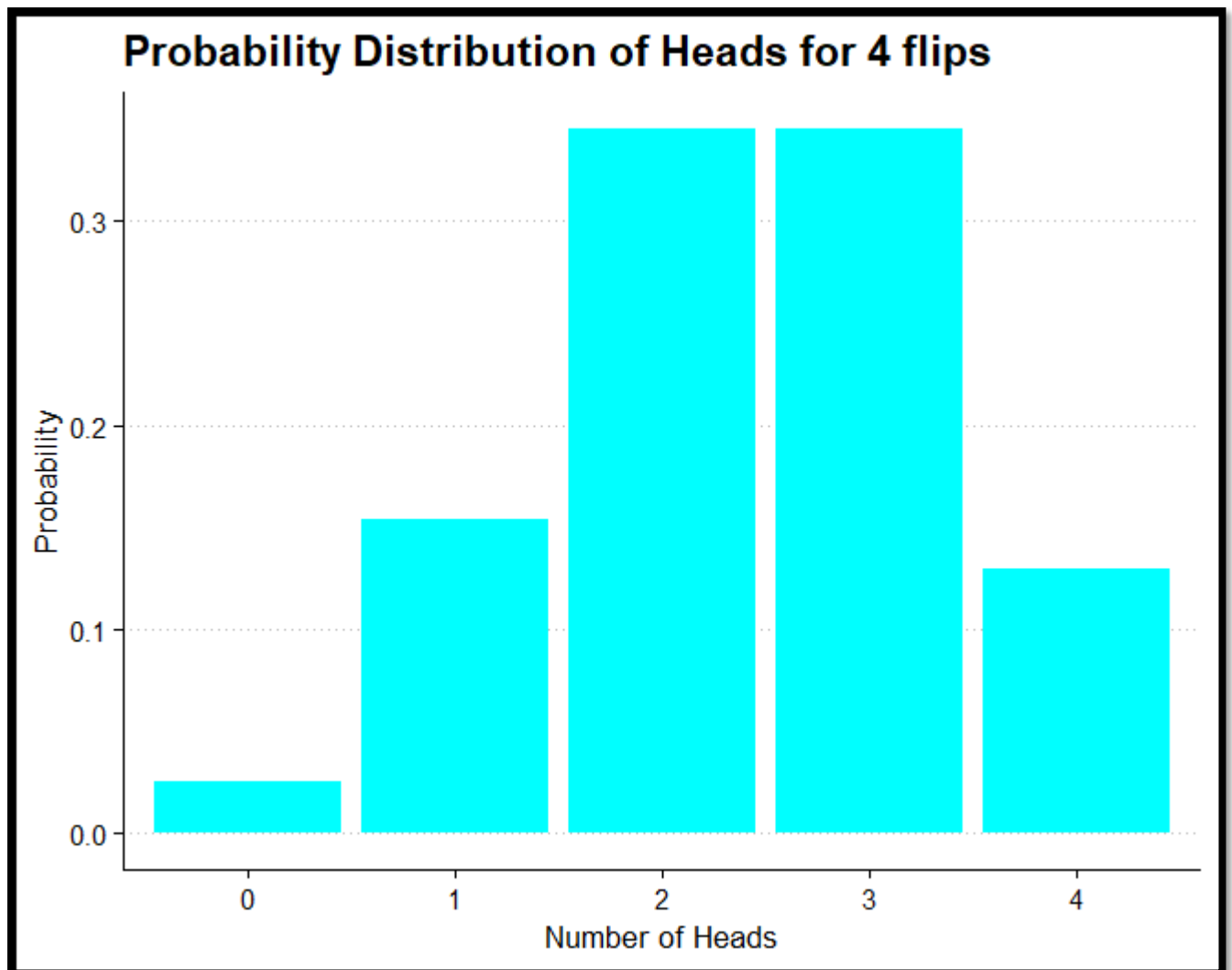
Here's what the code does step by step:

1. The `coin_outcomes` data frame is created with four columns (first, second, third, fourth), and each column is filled with "H" (heads) and "T" (tails) values in a way that simulates the possible outcomes of flipping a fair coin.
2. The `heads_prob` variable is set to 0.6, which represents the probability of getting heads in a single coin flip.
3. The `calculate_probability` function takes a row as input and calculates the probability of that specific outcome. It does this by iterating through each element in the row, and for each "H," it multiplies the `probability` by `heads_prob`, and for each "T," it multiplies the `probability` by `(1 - heads_prob)` (the probability of getting tails).
4. The `apply` function is used to apply the `calculate_probability` function to each row of the `coin_outcomes` data frame, and the resulting probabilities are stored in a new column called "probability."
5. The `coin_outcomes` data frame is printed, showing the original outcomes and the calculated probabilities for each row.

The resulting `coin_outcomes` data frame will have a new column "probability" that contains the calculated probabilities for each row, based on the given `heads_prob`. This code is simulating a simple probability calculation for coin flips with a known probability of getting heads.

4. Problem 24: -

Created a bar chart using summary_coins data frame.



Part-3: - Soccer Games

5.Problem 25: Probability of Winning All 10 Games

In this problem, we were tasked with finding the probability that a soccer team would win all 10 of their games, consisting of 5 home games and 5 away games. The given probabilities for winning each type of game were 75% for home games and 50% for away games.

To calculate this probability, we applied the multiplication rule for independent events. We found the probability of winning all 5 home games (0.75^5) and multiplied it by the probability of winning all 5 away games (0.5^5). The result was stored in the variable `prob25_result`.

```
> prob25_result <- (0.75^5)*(0.5^5)
> prob25_result
[1] 0.007415771
```

6. Problem 26: Probability of Winning More Than One Game

In this problem, the objective was to determine the probability that the soccer team would win more than one game out of the 10 they were about to play. To solve this, we employed the complement rule. Instead of directly calculating the probability of winning more than one game, we calculated the probability of not winning only zero or one game and then subtracted that from 1 to find the complement.

To find the probability of not winning zero or one game, we considered three scenarios:

1. Winning all 5 home games and losing all 5 away games ($0.25^5 * 0.5^5$).
2. Winning all 5 away games and losing all 5 home games ($0.5^5 * 0.25^5$).
3. Winning exactly one game (5 ways to win one game, each with a probability of 0.75 for a home game or 0.5 for an away game and 4 ways to lose the remaining 4 games).

We summed these probabilities and subtracted the result from 1, which yielded `prob26_result`.

```
> prob26_result <- 1 - ((0.25^5)*(0.5^5) + 0.75*(0.25^4)*(0.5^5) + (0.25^5)*
(0.5^4)*0.5)
> prob26_result
[1] 0.9998474
```

7.Problem 27: Counting Different Ways to Win 5 Out of 10 Games

In this problem, the task was to determine how many different ways the soccer team could win 5 out of the 10 games they were about to play. This was equivalent to finding the number of combinations of 10 games taken 5 at a time, disregarding the order in which they were won.

To calculate this, we utilized the binomial coefficient formula, specifically the `choose` function in R. We used `choose(10, 5)` to calculate the number of combinations of 10 items taken 5 at a time, which represents the number of different ways the team could win 5 games out of 10. This result was stored in the variable `prob27_result`.

```
> prob27_result <- choose(5,3)*choose(5,2)
> prob27_result
[1] 100
```

CONCLUSION

In summary, the provided code encompasses a comprehensive analysis of various scenarios, including ball data, coin flips, and soccer games. Here are the key takeaways:

1. Ball Data Analysis:

The code delves into the realm of probability and statistics concerning ball selection. It meticulously computes probabilities, conditional probabilities, and tabulates combinations, offering a profound understanding of how these concepts relate to real-life situations involving balls.

2. Coin Flip Probability:

The code adeptly calculates probabilities and visualizes the probability distribution for coin flips. It addresses specific probability challenges tied to coin outcomes, proving invaluable in fields where comprehending and visualizing probability distributions is pivotal.

3. Soccer Game Analysis:

The code provides valuable insights into a soccer team's performance. It quantifies the likelihood of distinct game outcomes, including winning all games, winning more than one game, and various methods of winning a specific number of games. These calculations yield crucial information for evaluating the team's performance and predicting outcomes in upcoming matches.

In conclusion, this code underscores the adaptability of probability and statistics in analyzing diverse scenarios, rendering it an invaluable instrument for decision-making and assessment across various contexts.

```
> test_file("project5_tests.R")  
[ FAIL 0 | WARN 0 | SKIP 0 | PASS 21 ]
```