VSN Sai Krishna Mohan
Kocherlakota
Title: Module 1 R Practice
Submission Date: 7th November 2023

INTRODUCTION:

In this project, a dataset of Bitcoin prices, including daily opening, closing, highest, and lowest values from 2013 to 2017, was utilized. The dataset was subjected to data cleaning, subsetting, and various data visualization and statistical analysis techniques using different R packages.

TASKS:

1. Data loading and cleaning

```
> bitcoin <- read_csv("Bitcoin.csv")
Rows: 1609 Columns: 6
>
> bitcoin <- clean_names(bitcoin)
> bitcoin
# A tibble: 1,609 × 6
      id date        open  high   low close
   <dbl> <chr>      <dbl> <dbl> <dbl> <dbl>
 1     1 22-09-2017 3628. 3758. 3554. 3631.
 2     2 21-09-2017 3901. 3916. 3614. 3631.
 3     3 20-09-2017 3916. 4031. 3858. 3906.
>
> bitcoin$date = dmy(bitcoin$date)
> bitcoin
# A tibble: 1,609 × 6
      id date        open  high   low close
   <dbl> <date>     <dbl> <dbl> <dbl> <dbl>
 1     1 2017-09-22 3628. 3758. 3554. 3631.
 2     2 2017-09-21 3901. 3916. 3614. 3631.
 3     3 2017-09-20 3916. 4031. 3858. 3906.
 4     4 2017-09-19 4074. 4094. 3869. 3925.
>
> bitcoin$year <- year(bitcoin$date)
> bitcoin
# A tibble: 1,609 × 7
      id date        open  high   low close  year
   <dbl> <date>     <dbl> <dbl> <dbl> <dbl> <dbl>
 1     1 2017-09-22 3628. 3758. 3554. 3631.  2017
 2     2 2017-09-21 3901. 3916. 3614. 3631.  2017
 3     3 2017-09-20 3916. 4031. 3858. 3906.  2017
```

The necessary packages have been installed and loaded in the R script. "tidyverse," "janitor," and "lubridate" packages were installed to facilitate functions such as `read_csv()` for reading data, data cleaning, and converting data variables from character to the year format, respectively.

2. Computing Statistical Equations

```
> mean_close<- mean(bitcoin$close) #mean of close
> mean_close
[1] 695.5634
>
> median_close<- median(bitcoin$close) #median of close
> median_close
[1] 447.53
>
> sd_close<- sd(bitcoin$close) #standard deviation of close
> sd_close
[1] 800.5576
>
> min_close<- min(bitcoin$close) #minimum close
> min_close
[1] 68.43
>
> max_close<- max(bitcoin$close) #maximum close
> max_close
[1] 4892.01
>
```

```
> summary_close<- summary(bitcoin$close) #summary statistics for close
> summary_close
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  68.43 260.89  447.53  695.56  702.03 4892.01
```

The code computes summary statistics for key variables, including mean, median, standard deviation, minimum, and maximum values. Additionally, it generates summary statistics specifically for the closing value of Bitcoin.

3. Subsetting Data

I. A new data frame, bitcoin_year,' is created. It first groups the data by year and then calculates the mean closing value for each year, storing the results in the "bitcoin_year" data frame. This will give you a summary of the annual average closing values of Bitcoin.

```
> bitcoin_year <- bitcoin %>% group_by(year) %>%
+   summarise(avg_rating = mean(close, na.rm = TRUE))
> bitcoin_year
```

II. Created another creates a new data frame, groups it by date, and duplicates the "close" variable without making any changes or transformations to the data.

```
> bitcoin_date <- bitcoin %>%
+   group_by(date) %>%
+   mutate(close = close)
> bitcoin_date
```

4. Data Visualizations

a. Line graph: -

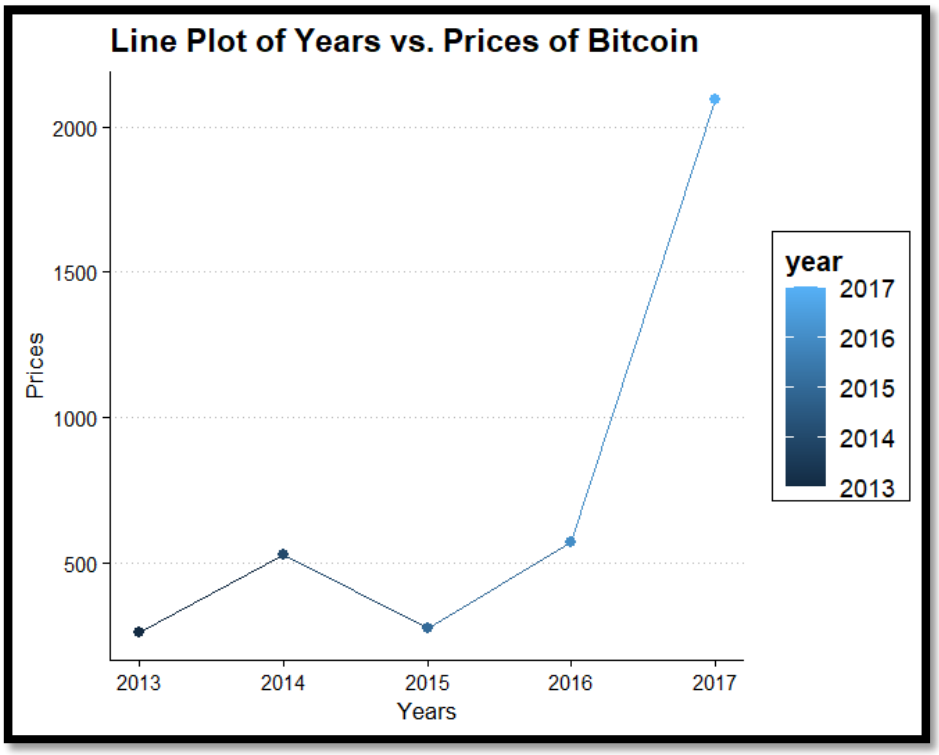

Figure 1: - Line Graph

```
> ggplot(bitcoin_year, aes(x = year, y = avg_rating, color = year)) +
+   geom_line() +
+   geom_point(size = 2) +
```

```
+   labs(x = "Years", y = "Prices") +
+   ggtitle("Line Plot of Years vs. Prices of Bitcoin") + theme_clean()
```

It visualizes the average Bitcoin prices over the years, with each year represented by a different color. The plot displays a line connecting data points and individual data points (dots). The x-axis represents years, the y-axis represents prices, and the plot has a title: "Line Plot of Years vs. Prices of Bitcoin." Additionally, it appears to use a custom theme called "theme_clean."

b. Scatter Plot: -

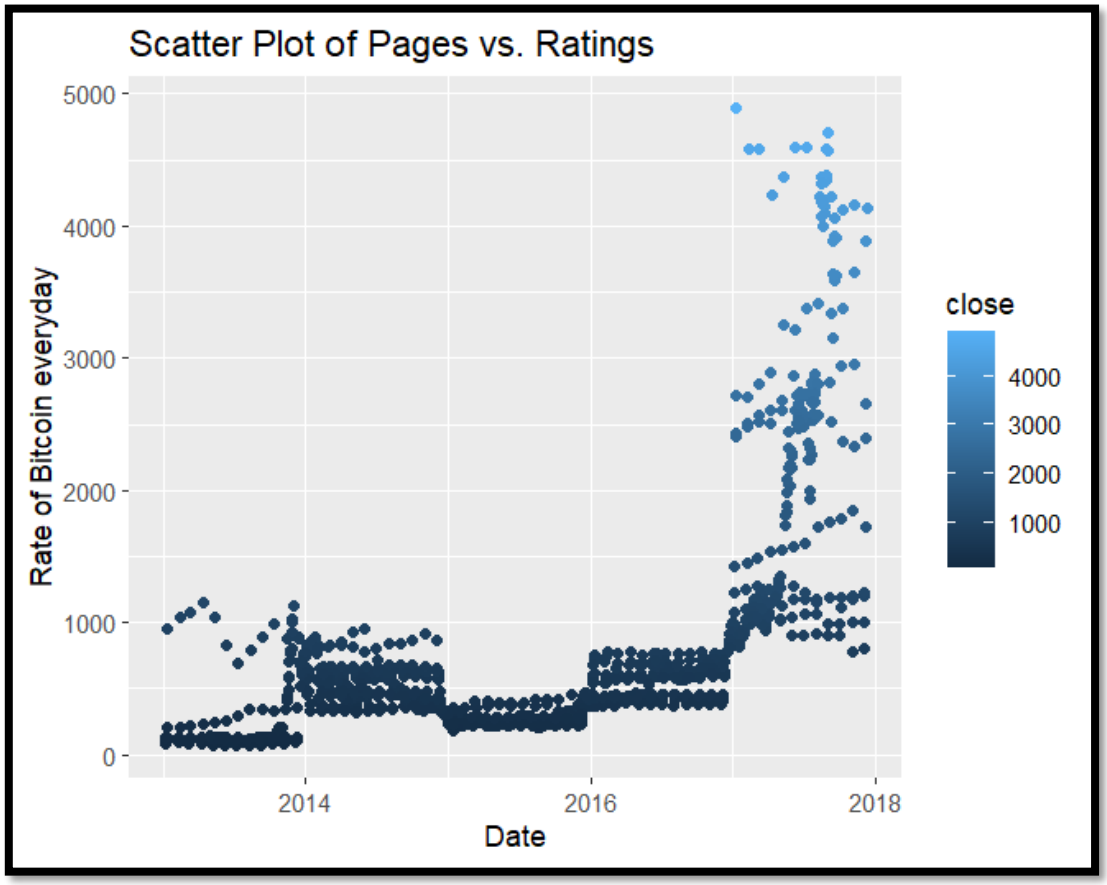

Figure 2: - Scatter Plot

```
> ggplot(bitcoin_date, aes(x = date, y = close, color = close)) +
+   geom_point() +
+   labs(x = "Date", y = "Rate of Bitcoin everyday") +
+   ggtitle("Scatter Plot of Pages vs. Ratings")
```

It visualizes the daily closing rates of Bitcoin over time. The x-axis represents dates, the y-axis represents the closing rates, and the color represents different closing rates. The plot is titled "Scatter Plot of Pages vs. Ratings."

CONCLUSION

The R script loads and preprocesses a Bitcoin dataset. It includes data cleaning, date format conversion, and year extraction. The script then calculates and visualizes key statistics related to Bitcoin's closing prices. It generates a line plot that displays the yearly average closing values and calculates summary statistics, such as mean, median, standard deviation, minimum, and maximum, for the "close" variable. This comprehensive analysis offers insights into Bitcoin's historical price trends and statistical characteristics.

CITATIONS

1. Holtz, Y. (n.d.). Add text labels with ggplot2. https://r-graph-gallery.com/275-add-text-labels-with-ggplot2.html
2. GeeksforGeeks. (2021, December 16). Calculate the average variance and standard deviation in R programming. https://www.geeksforgeeks.org/calculate-the-average-variance-and-standard-deviation-in-r-programming/