Table Of Content :

1. Load libraries
2. Import datasets
3. Check dimensions and types
4. Data Cleaning
5. Data Exploration
6. Data Visualization
7. Conclusions and Insights

1. Load libraries :

```
library(ggplot2)
library(dplyr)
library(tidyr)
library(lubridate)
library(corrplot)
```

2. Read the dataset files :

```
btc_1min <- read.csv("/kaggle/input/bitcoin-historical-data/btcusd_1-min_data.csv")
btc_1h <- read.csv("/kaggle/input/bitcoin-historical-data/btcusd_1-h_data.csv")
```

3. Give brief about data :

```
head(btc_1min)
```

```
  Timestamp   Open   High    Low  Close Volume Weekday
1  1325317920 4.58000 4.58000 4.58000 4.58000 0.00000       7
2  1325317980 4.58000 4.58000 4.58000 4.58000 0.00000       1
3  1325318040 4.58000 4.58000 4.58000 4.58000 0.00000       1
4  1325318100 4.58000 4.58000 4.58000 4.58000 0.00000       1
5  1325318160 4.58000 4.58000 4.58000 4.58000 0.00000       1
6  1325318220 4.58000 4.58000 4.58000 4.58000 0.00000       1
```

```
head(btc_1h)
```

```
  Timestamp   Open   High    Low  Close Volume Weekday
1  1325317920 4.58000 4.58000 4.58000 4.58000 0.00000       7
2  1325319600 4.57333 4.57333 4.56333 4.56333 2.84400       1
3  1325323200 4.56333 4.57333 4.56333 4.57333 1.12000       1
4  1325326800 4.57333 4.57333 4.57333 4.57333 0.00000       1
5  1325330400 4.57333 4.57333 4.57333 4.57333 0.00000       1
6  1325334000 4.57333 4.57333 4.57333 4.57333 0.00000       1
```

4. Data Exploration :

```
str(btc_1min)
```

```
'data.frame':	2627997 obs. of  7 variables:
 $ Timestamp: int  1325317920 1325317980 1325318040 1325318100 1325318160 1325318220 ...
 $ Open     : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ High     : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ Low      : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ Close    : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ Volume   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Weekday  : int  7 1 1 1 1 1 1 1 1 1 ...
```

```
str(btc_1h)
```

```
'data.frame':	43829 obs. of  7 variables:
 $ Timestamp: int  1325317920 1325319600 1325323200 1325326800 1325330400 1325334000 ...
 $ Open     : num  4.58 4.57 4.56 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ High     : num  4.58 4.57 4.57 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ Low      : num  4.58 4.56 4.56 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ Close    : num  4.58 4.56 4.57 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ Volume   : num  0 2.84 1.12 0 0 0 0 0 0 0 ...
 $ Weekday  : int  7 1 1 1 1 1 1 1 1 1 ...
```

Metadata :

- Timestamp :
  - The number data this record (Unix TS (1970-01-01))
- Open :
  - The price in Bitcoin at the start of the trading day
- High :
  - The highest price in Bitcoin during the trading day.
- Low :
  - The lowest price in Bitcoin during the trading day.
- Close :
  - The last price in Bitcoin during the trading day.
- Volume :
  - The quantity of Bitcoin sold on the current day.
- Weekday (Integer):
  - The trading day (starts in 1 (Sunday), 2 (Monday) to 7 (Saturday))
- Volume :
  - The trading volume in USD for the day. It differs from most Market data datasets

Metadata :

```
str(btc_1min)
```

```
'data.frame':	2627997 obs. of  7 variables:
 $ Timestamp: int  1325317920 1325317980 1325318040 1325318100 1325318160 1325318220 ...
 $ Open     : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ High     : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ Low      : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ Close    : num  4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 4.58 ...
 $ Volume   : num  0 0 0 0 0 0 0 0 0 0 ...
 $ Weekday  : int  7 1 1 1 1 1 1 1 1 1 ...
```

```
str(btc_1h)
```

```
'data.frame':	43829 obs. of  7 variables:
 $ Timestamp: int  1325317920 1325319600 1325323200 1325326800 1325330400 1325334000 ...
 $ Open     : num  4.58 4.57 4.56 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ High     : num  4.58 4.57 4.57 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ Low      : num  4.58 4.56 4.56 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ Close    : num  4.58 4.56 4.57 4.57 4.57 4.57 4.57 4.57 4.57 4.57 ...
 $ Volume   : num  0 2.84 1.12 0 0 0 0 0 0 0 ...
 $ Weekday  : int  7 1 1 1 1 1 1 1 1 1 ...
```

5. Data Cleaning :

```
colSums(is.na(btc_1min))
```

```
Timestamp      Open      High       Low     Close    Volume   Weekday
        0         0         0         0         0         0         0
```

```
colSums(is.na(btc_1h))
```

```
Timestamp      Open      High       Low     Close    Volume   Weekday
        0         0         0         0         0         0         0
```

```
colSums(btc_1min == 0)
```

```
Timestamp      Open      High       Low     Close    Volume   Weekday
        0         0         0         0         0   1627         0
```

```
colSums(btc_1h == 0)
```

```
Timestamp      Open      High       Low     Close    Volume   Weekday
        0         0         0         0         0      0         0
```

```
btc_1min <- unique(btc_1min)
btc_1h <- unique(btc_1min)
```

```
dim(btc_1min)
```

```
[1] 2627997       7
```

```
dim(btc_1h)
```

```
[1] 2627997       7
```

```
btc_1min$Volume[btc_1min$Volume == 0] <- mean(btc_1min$Volume, na.rm = TRUE)
btc_1h$Volume[btc_1h$Volume == 0] <- mean(btc_1h$Volume, na.rm = TRUE)
```

```
colSums(btc_1min == 0)
```

```
Timestamp      Open      High       Low     Close    Volume   Weekday
        0         0         0         0         0         0         0
```

```
colSums(btc_1h == 0)
```

```
Timestamp      Open      High       Low     Close    Volume   Weekday
        0         0         0         0         0         0         0
```

6.Eda :

```
# Set the theme for all plots
theme_set(theme_minimal(base_size = 14))

# Create a function to plot histograms for all numeric columns
plot_histograms <- function(data, dataset_name) {
  numeric_cols <- sapply(data, is.numeric)
  for (col in names(data)[numeric_cols]) {
    p <- ggplot(data, aligned = TRUE)
    print(p)
  }
}

# Apply to both datasets
cat("Plotting histograms for 1-minute data...\n")
plot_histograms(btc_1min, "1-min")

cat("Plotting histograms for 1-hour data...\n")
plot_histograms(btc_1h, "1-hour")
```

```
Plotting histograms for 1-minute data...
Plotting histograms for 1-hour data...
```

7. Conclusions and Insights :

All Questions 10% Answered

- What is the overall price trend in Bitcoin data?
- What is the relationship between volume and price?
- Are there any seasonal patterns in Bitcoin prices?
- Is volume data reliable or should we be cautious?
- What is the correlation between different price features?
- Are there any outliers that might affect our analysis?
- What is the price range and volatility?
- Which day of the week is most volatile?

BITCOIN INVESTMENT SIMULATION

Contact

Email: ahmedali.tn@gmail.com

Phone: +216 50 964 084

Linkedin

Ahmed ALI

Kaggle

Ahmed ALI SALLEM

Discord

Ahmedali#1445

Ahmedali.net

Export data file

```
write.csv(btc_1min, "/kaggle/input/bitcoin-historical-data/btcusd_1-min_data.csv", row.names = FALSE)
```

```
[1] TRUE
```