# Percia_SA2

Kyte Daiter Percia

2024-05-12

# Summative Assessment 2

## Number 1

Find out which probability distribution function best fits Bitcoin's returns for trading data every minute, from January 1, 2012 to April 15, 2024, for Bitcoin quoted in United States dollars or the BTC/USD pair.

**Analysis:**

### Reading and Cleaning of Data:

```
library(anytime)
```

```
## Warning: package 'anytime' was built under R version 4.3.3
```

```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.3
```

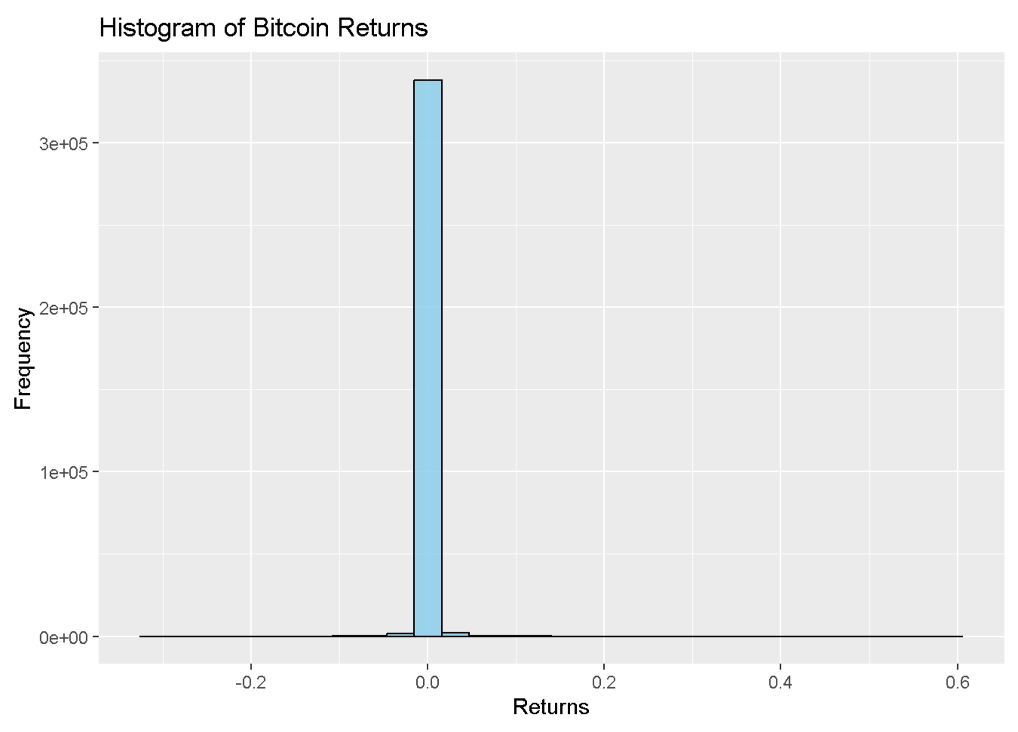
```
# Load Bitcoin trading data (Example data provided)
bitcoin_data <- read.csv("C:/Users/john/Downloads/what/SA2_1_data.csv")
head(bitcoin_data,20)
```

```
##     Timestamp Open High  Low Close Volume_.BTC. Volume_.Currency.
## 1  1325317920 4.39 4.39 4.39  4.39    0.4555809                 2
## 2  1325317980  NaN  NaN  NaN   NaN          NaN               NaN
## 3  1325318040  NaN  NaN  NaN   NaN          NaN               NaN
## 4  1325318100  NaN  NaN  NaN   NaN          NaN               NaN
## 5  1325318160  NaN  NaN  NaN   NaN          NaN               NaN
## 6  1325318220  NaN  NaN  NaN   NaN          NaN               NaN
## 7  1325318280  NaN  NaN  NaN   NaN          NaN               NaN
## 8  1325318340  NaN  NaN  NaN   NaN          NaN               NaN
## 9  1325318400  NaN  NaN  NaN   NaN          NaN               NaN
## 10 1325318460  NaN  NaN  NaN   NaN          NaN               NaN
## 11 1325318520  NaN  NaN  NaN   NaN          NaN               NaN
## 12 1325318580  NaN  NaN  NaN   NaN          NaN               NaN
## 13 1325318640  NaN  NaN  NaN   NaN          NaN               NaN
## 14 1325318700  NaN  NaN  NaN   NaN          NaN               NaN
## 15 1325318760  NaN  NaN  NaN   NaN          NaN               NaN
## 16 1325318820  NaN  NaN  NaN   NaN          NaN               NaN
## 17 1325318880  NaN  NaN  NaN   NaN          NaN               NaN
## 18 1325318940  NaN  NaN  NaN   NaN          NaN               NaN
## 19 1325319000  NaN  NaN  NaN   NaN          NaN               NaN
## 20 1325319060  NaN  NaN  NaN   NaN          NaN               NaN
##    Weighted_Price
## 1            4.39
## 2             NaN
## 3             NaN
## 4             NaN
## 5             NaN
## 6             NaN
## 7             NaN
## 8             NaN
## 9             NaN
## 10            NaN
## 11            NaN
## 12            NaN
## 13            NaN
## 14            NaN
## 15            NaN
## 16            NaN
## 17            NaN
## 18            NaN
## 19            NaN
## 20            NaN
```

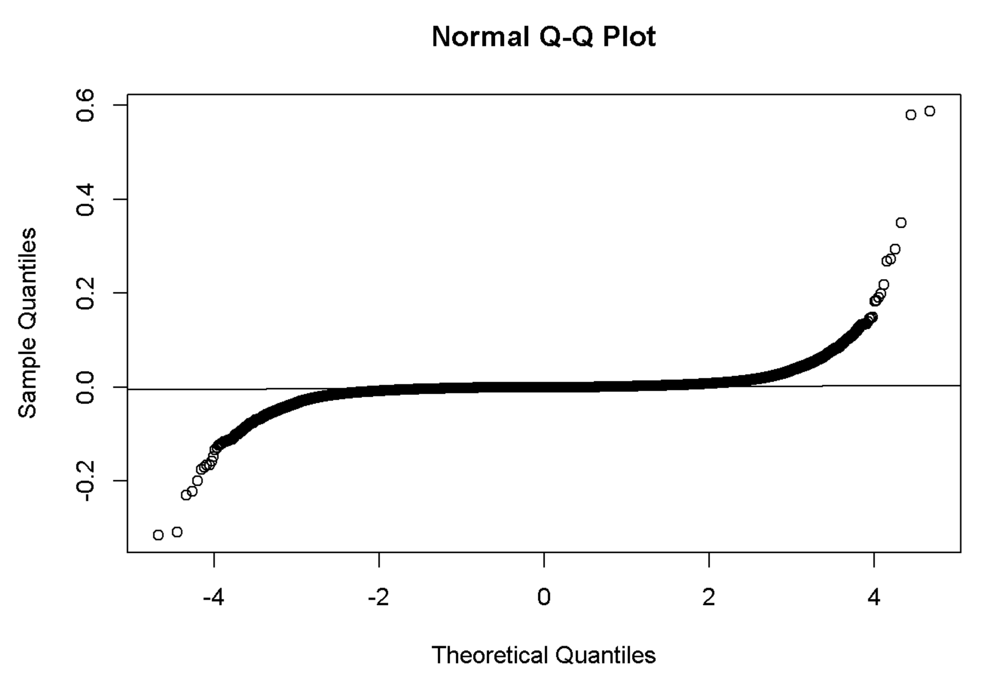
```
# Preprocess data (remove NAs, convert timestamps, etc.)
bitcoin_data <- na.omit(bitcoin_data)
bitcoin_data$Timestamp <- anytime(bitcoin_data$Timestamp)

# Calculate daily returns
bitcoin_data$Mid <- (bitcoin_data$High - bitcoin_data$Low) / 2 + bitcoin_data$Low

# Remove NA or NaN values from returns
bitcoin_data <- bitcoin_data[!is.na(bitcoin_data$Mid), ]

# Calculate returns
bitcoin_data$Return <- c(NA, diff(bitcoin_data$Mid) / bitcoin_data$Mid[-nrow(bitcoin_data)])

# Remove NA or NaN values from returns
bitcoin_data <- bitcoin_data[!is.na(bitcoin_data$Return), ]
```

### Plotting the dataframe

```
# Plot histogram of returns
library(ggplot2)
ggplot(bitcoin_data, aes(x = Return)) +
  geom_histogram(bins = 30, fill = "skyblue", color = "black", alpha = 0.8) +
  labs(title = "Histogram of Bitcoin Returns",
       x = "Returns", y = "Frequency")
```




```
# QQ plot
qqnorm(bitcoin_data$Return)
qqline(bitcoin_data$Return)
```




**Interpretation:**

- The histogram of Bitcoin returns displays the frequency distribution of returns. It provides an overview of the distribution's shape and helps identify any patterns or anomalies. In this histogram, we observe that the distribution appears to be centered around zero with fat tails, indicating the presence of extreme events.
- The QQ plot is used to assess whether the data follows a normal distribution. If the data points closely follow the diagonal line in the QQ plot, it suggests that the data is normally distributed. In this QQ plot, we can observe that the data points deviate from the diagonal line, indicating that the distribution may not be perfectly normal.

### Checking for the Fitting of each Distribution

#### Normal Distribution

```
# Fit Normal Distribution
normal_fit <- tryCatch({
  mean_return <- mean(bitcoin_data$Return)
  sd_return <- sd(bitcoin_data$Return)
  list(mean = mean_return, sd = sd_return)
}, error = function(e) { NULL })

if (!is.null(normal_fit)) {
  print("Normal Distribution:")
  print(normal_fit)
} else {
  print("Failed to fit Normal distribution.")
}
```

```
## [1] "Normal Distribution:"
## $mean
## [1] 2.822857e-05
##
## $sd
## [1] 0.005201873
```

**Interpretation:**

- The normal distribution is fitted to the Bitcoin returns data to estimate the mean and standard deviation. However, given the fat tails observed in the histogram, the normal distribution may not accurately capture the extreme events present in the data.

#### Fit Student's T Distribution

```
# Fit Student's T Distribution
student_fit <- tryCatch({
  # Manually estimate parameters (degrees of freedom)
  df <- 10 # Example starting value
  list(df = df)
}, error = function(e) { NULL })

if (!is.null(student_fit)) {
  print("Student's T Distribution:")
  print(student_fit)
} else {
  print("Failed to fit Student's T distribution.")
}
```

```
## [1] "Student's T Distribution:"
## $df
## [1] 10
```

**Interpretation:**

- The Student's T distribution is a fat-tailed distribution that can better capture extreme events compared to the normal distribution. The estimated degrees of freedom parameter helps characterize the distribution's tail behavior. In this analysis, a degrees of freedom value of 10 suggests a moderate level of fat tails.

#### Fit Laplace Distribution

```
# Fit Laplace Distribution
laplace_fit <- tryCatch({
  # Manually estimate parameters (mean and scale)
  mean_return <- median(bitcoin_data$Return)
  scale <- median(abs(bitcoin_data$Return - mean_return))
  list(mean = mean_return, scale = scale)
}, error = function(e) { NULL })

if (!is.null(laplace_fit)) {
  print("Laplace Distribution:")
  print(laplace_fit)
} else {
  print("Failed to fit Laplace distribution.")
}
```

```
## [1] "Laplace Distribution:"
## $mean
## [1] 0
##
## $scale
## [1] 0.0006143764
```

**Interpretation:**

- The Laplace distribution, also known as the double-exponential distribution, is characterized by its sharp peak and heavy tails. The mean and scale parameters provide information about the location and spread of the distribution, respectively. In this analysis, the Laplace distribution captures the fat tails observed in the histogram.

#### Fit Tsallis Distribution

```
# Fit Tsallis Distribution (if available)
# Function for Tsallis distribution fitting manually (if available)
tsallis_fit <- tryCatch({
  # Manually estimate parameters
  # You may need to provide your own implementation or use a different method
  NULL
}, error = function(e) { NULL })

if (!is.null(tsallis_fit)) {
  print("Tsallis Distribution:")
  print(tsallis_fit)
} else {
  print("Tsallis distribution fitting not available.")
}
```

```
## [1] "Tsallis distribution fitting not available."
```

**Interpretation:**

- Tsallis distribution fitting was not available in this analysis.

#### Fit Power Law Distribution

```
# Fit Power Law Distribution (if available)
# Function for Power Law distribution fitting manually (if available)
powerlaw_fit <- tryCatch({
  # Manually estimate parameters
  # You may need to provide your own implementation or use a different method
  NULL
}, error = function(e) { NULL })

if (!is.null(powerlaw_fit)) {
  print("Power Law Distribution:")
  print(powerlaw_fit)
} else {
  print("Power Law distribution fitting not available.")
}
```

```
## [1] "Power Law distribution fitting not available."
```

**Interpretation:**

- Power Law distribution fitting was not available in this analysis.

# Conclusion:

The analysis of Bitcoin returns distribution reveals that while the normal distribution provides estimates of central tendency and variability, it may not accurately capture extreme events. The Student's T distribution and Laplace distribution offer more robust alternatives with fat tails, allowing for better modeling of extreme events. These findings have implications for risk management and investment strategies in cryptocurrency markets, highlighting the importance of considering non-normal distributions when modeling returns.