SEC 1-SA2 DELA ROSA, R

Github Link:
https://github.com/rddelarosa/APM1110/blob/main/SA2/SEC_1-SA2_DELA-ROSA%2C-R.md?plain=1

Roland Dela Rosa

May 16, 2025

1. Find out which probability distribution function best fits Bitcoin's returns for trading data every minute, from January 1, 2012 to April 15, 2025, for Bitcoin quoted in United States dollars or the BTC/USD pair.

```
btc_data <- read.csv("btcusd_1-min_data.csv")

btc_data$Timestamp <- as.POSIXct(btc_data$Timestamp, origin = "1970-01-01", tz = "UTC")

# Filter data up to April 15, 2025
end_date <- as.POSIXct("2025-04-15 23:59:59", tz = "UTC")
btc_data <- subset(btc_data, Timestamp <= end_date)

btc_returns <- btc_data[, c("Timestamp", "Close")]
btc_returns$SimpleReturn <- c(NA, diff(btc_returns$Close) / head(btc_returns$Close, -1))

# Remove NA rows (first return is NA)
btc_returns <- na.omit(btc_returns)

# Define function for KS test
perform_ks_test <- function(sample_data, sim_data) {
  ks_res <- ks.test(sample_data, sim_data)

  return(list(D = ks_res$statistic, p_value = ks_res$p.value))
}

# Prepare data for testing
returns <- btc_returns$SimpleReturn
n <- length(returns)


# Normal distribution
test_normal <- rnorm(n, mean = mean(returns), sd = sd(returns))
ks_normal <- perform_ks_test(returns, test_normal)
```

```
## Warning in ks.test.default(sample_data, sim_data): p-value will be approximate
## in the presence of ties
```

```r
# Student's t-distribution (df = n-1)
test_student_t <- rt(n, df = n - 1)
ks_student_t <- perform_ks_test(returns, test_student_t)
```

```
## Warning in ks.test.default(sample_data, sim_data): p-value will be approximate
## in the presence of ties
```

```r
# Laplace distribution (mean & sd)
test_laplace <- rlaplace(n, m = mean(returns), s = sd(returns))
ks_laplace <- perform_ks_test(returns, test_laplace)
```

```
## Warning in ks.test.default(sample_data, sim_data): p-value will be approximate
## in the presence of ties
```

```r
# Uniform distribution (min and max from data)
sim_uniform <- runif(n, min = min(returns), max = max(returns))
ks_uniform <- perform_ks_test(returns, sim_uniform)
```

```
## Warning in ks.test.default(sample_data, sim_data): p-value will be approximate
## in the presence of ties
```

```r
# Exponential distribution (make data positive)
returns_pos <- returns - min(returns) + 1e-6
sim_exponential <- rexp(n, rate = 1 / mean(returns_pos))
ks_exponential <- perform_ks_test(returns_pos, sim_exponential)
```

```
## Warning in ks.test.default(sample_data, sim_data): p-value will be approximate
## in the presence of ties
```

```r
# 11. Aggregate results
ks_results <- data.frame(
  Distribution = c("Normal", "Student's t", "Laplace",
                   "Uniform", "Exponential"),
  D_Statistic = c(ks_normal$D, ks_student_t$D, ks_laplace$D,    ks_uniform$D, ks_exponential$D))

ks_results <- ks_results[order(ks_results$D_Statistic), ]

print(ks_results)
```

```
##   Distribution D_Statistic
## 3      Laplace   0.2034688
## 1       Normal   0.2151366
## 2  Student's t   0.4943319
## 5  Exponential   0.6229917
## 4      Uniform   0.6402629
```

Based on the Kolmogorov-Smirnov tests performed data, the Laplace distribution provides the best fit among the tested distributions. This conclusion is drawn from the Laplace distribution having the lowest D-statistic (0.203), indicating the smallest maximum distance between the empirical and theoretical cumulative distribution functions. Other distributions show moderate fit but with higher D-statistics, suggesting they capture the data characteristics less accurately. The Student's t, Uniform, and Exponential distributions exhibit significantly higher D-statistics, reflecting poor alignment with the data's distribution.

```r
dist_results <- data.frame(
  Distribution = c("Laplace", "Normal", "Student's t", "Uniform", "Exponential"),
  Reason = c(
    "Lowest D-statistic; best fit to heavy tails and peaks",
    "Assumes lighter tails; underestimates extreme returns",
    "Poor fit; higher deviation from empirical data",
    "Uniform spread unsuitable for observed return patterns",
    "Not suitable; data has symmetric heavy tails, not skewed"
  )
)

# Print the table with kable
print(dist_results, caption = "Distribution Fit Comparison Based on D-Statistic")
```

```
##   Distribution                                                Reason
## 1      Laplace    Lowest D-statistic; best fit to heavy tails and peaks
## 2       Normal    Assumes lighter tails; underestimates extreme returns
## 3  Student's t           Poor fit; higher deviation from empirical data
## 4      Uniform   Uniform spread unsuitable for observed return patterns
## 5  Exponential Not suitable; data has symmetric heavy tails, not skewed
```

2. Test using Shapiro-Wilk normality test the Ethereum returns for trading data every five minutes, from August 7, 2015 to April 15, 2025.

```r
# Load packages
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```r
library(data.table)
```

```
##
## Attaching package: 'data.table'

## The following objects are masked from 'package:lubridate':
##
##     hour, isoweek, mday, minute, month, quarter, second, wday, week,
##     yday, year

## The following objects are masked from 'package:dplyr':
##
##     between, first, last
```

```r
eth_data <- fread("ETHUSD_1m_Binance.csv")

eth_data <- eth_data %>%
  mutate(datetime = as.POSIXct(`Open time`, origin = "1970-01-01", tz = "UTC")) %>%
  filter(datetime >= as.POSIXct("2015-08-07 00:00:00", tz = "UTC"),
         datetime <= as.POSIXct("2025-04-15 23:59:59", tz = "UTC")) %>%
  arrange(datetime)

eth_5min <- eth_data %>%
  mutate(datetime_5min = floor_date(datetime, "5 minutes")) %>%
  group_by(datetime_5min) %>%
  summarize(Close = last(Close), .groups = "drop") %>%
  arrange(datetime_5min)

eth_5min <- eth_5min %>%
  mutate(log_return = log(Close / lag(Close))) %>%
  na.omit()

set.seed(123)
sample_returns <- sample(eth_5min$log_return, size = 5000)

shapiro_test <- shapiro.test(sample_returns)
cat("Shapiro-Wilk test p-value:", shapiro_test$p.value, "\n")
```

```
## Shapiro-Wilk test p-value: 2.115809e-64
```

```r
if (shapiro_test$p.value < 0.05) {
  cat("Reject normality based on Shapiro-Wilk test\n")
} else {
  cat("Fail to reject normality based on Shapiro-Wilk test\n")
}
```

```
## Reject normality based on Shapiro-Wilk test
```

The Shapiro-Wilk test yielded a p-value which is far below the conventional significance level of 0.05. This extremely small p-value indicates strong evidence against the null hypothesis of normality. Therefore, we reject the assumption that the Ethereum 5-minute log returns follow a normal distribution. This means the returns data exhibit statistically significant deviations from normality, such as skewness, heavy tails. Consequently, using models or statistical methods that assume normality may not be appropriate for this dataset without applying transformations or considering alternative distributions.