Risk Analytics - Practical 1

Winter semester 2024-2025, HEC, UNIL

Robin Michel, Faber Bickerstaffe, Antoine Magnin, Anastasia Pushkarev and Victorien Rodondi

2024-11-09

Part 1: Financial Returns and Normality

a) and b) Assessing Stationarity of Bitcoin Prices and Negative Log Returns

In this section, we assess the stationarity of the raw Bitcoin prices and their negative log returns (see Figure 1), as stationarity is crucial for time series modeling.

We first plotted the Bitcoin prices (see Figure 2) to visually inspect for trends or patterns. The plot showed a clear trend, suggesting non-stationarity. To confirm this, we applied the Augmented Dickey-Fuller (ADF) test (see Table 1), which resulted in a p-value of 0.3885, indicating that the raw Bitcoin prices are non-stationary.

To address this, we computed the negative log returns, a transformation commonly used in financial time series analysis to obtain a stationary series. Visual inspection of the negative log returns (see Figure 3) suggested stationarity. This was further confirmed by the ADF test (see Table 2), which gave a p-value of 0.01, leading us to reject the null hypothesis of non-stationarity and confirming that the negative log returns are stationary.

c) Assessing the Normality of Negative Log Returns

To evaluate whether the negative log returns of Bitcoin follow a normal distribution, we first examined the data visually using a histogram (see Figure 4) and a QQ-plot (see Figure 5). The histogram of the negative log returns suggests that the data may be approximately normally distributed, though some deviations from normality could exist. Additionally, the QQ-plot shows that the returns are normally distributed for non-extreme values, but there are noticeable deviations in the tails, indicating that the negative log returns may not follow a perfect normal distribution.

To formally test for normality, we applied the Anderson-Darling test (see Table 3), which gave a p-value less than 0.05. As a result, we reject the null hypothesis (H0) that the data is normally distributed. This confirms that, despite appearing somewhat normal in the central part of the distribution, the negative log returns are not normally distributed, especially due to the extreme values.

d) and e) Fitting a t-Distribution and Comparing Tails

Since the negative log returns deviate from normality, particularly in the extremes, we fit a t-distribution to the scaled data to check if it better captures these extreme values. A QQ-plot was generated to compare the negative log returns with the theoretical t-distribution (see Figure 6), which showed that the data fits the t-distribution quite well, including in the tails. For comparison, we also generated a QQ-plot for the normal distribution (see Figure 5), which demonstrated a poorer fit, particularly for extreme values. This suggests that the t-distribution, with its ability to model heavy tails, is a more appropriate fit for the data (see Figure 8).

Next, we compared the density plots of the normal and t-distributions. As expected, the t-distribution exhibited heavier tails than the normal distribution, meaning we should expect more extreme, unexpected events in a t-distribution (see Figure 9).

Since the Bitcoin data follows the t-distribution more closely, and the t-distribution has fatter tails than the normal distribution, this indicates that extreme values (large deviations from the mean) are more likely in the Bitcoin data than if it were normally distributed.

Part 2: Financial time series, heteroscedasticity and the random walk hypothesis

a) and b) Assessing Autocorrelation in Bitcoin Prices

Building on the previous stationarity analysis, we further explored the autocorrelation of the raw Bitcoin prices and the negative log returns. In part (a), we plotted the ACF (Autocorrelation Function) for both series (see Figures 9 and 10). As expected, the raw Bitcoin prices showed strong and persistent autocorrelation, consistent with their non-stationary nature. In contrast, the negative log returns displayed only minor, short-lived autocorrelation, aligning with their stationary behavior.

In part (b), we applied the Ljung-Box test (see Tables 4 and 5) to formally assess autocorrelation. The test confirmed significant autocorrelation in both series ($p\text{-value} < 0.05$). However, the negative log returns exhibited much weaker and short-term autocorrelation compared to the raw prices. This suggests that, while the log returns are stationary, they still contain some mild autocorrelation, typical of financial time series.

In summary, the raw series has strong autocorrelation and is non-stationary, whereas the negative log returns are stationary with only moderate, short-lived autocorrelation, making them better suited for modeling.

c) ARIMA Modeling

In this section, we proposed an ARIMA model for the negative log returns based on the analysis of the ACF and PACF plots (see Figures 10 and 11). The ACF plot showed a spike at lag 2, suggesting a moving average component of $q = 2$. Similarly, the PACF plot indicated a significant spike at lag 2, suggesting an autoregressive component of $p = 2$. Since the series is already stationary, we set $d = 0$. Based on this, we fit an ARIMA(2, 0, 2) model to the negative log returns. To confirm our model selection, we used the auto.arima() function, which automatically identifies the best ARIMA model. The function also suggested an ARIMA(2, 0, 2) model, reinforcing our choice.

Next, we assessed the residuals of the ARIMA model to evaluate its adequacy. The ACF of the residuals showed no significant autocorrelation (see Figure 12), indicating that the model successfully captured the underlying patterns in the data. The Ljung-Box test further confirmed this (see Table 6), as the p-value was greater than 0.05, suggesting no significant autocorrelation remaining in the residuals. However, the QQ-plot (see Figure 13) and Shapiro-Wilk test (see Table 7) indicated that the residuals do not follow a normal distribution ($p\text{-value} < 0.05$). Additionally, the plot of residuals over time (see Figure 14) showed signs of volatility clustering, implying that the variance is not constant, which indicates the presence of heteroscedasticity. This suggests that while the ARIMA model adequately captures the autocorrelation in the data, it fails to account for the changing variance and heavy tails in the residuals.

In conclusion, the ARIMA(2, 0, 2) model effectively captures the serial dependencies in the negative log returns but may not fully address the volatility present in the data. This indicates that a GARCH model, which accounts for time-varying volatility, could be a better fit for modeling the residuals.

d) GARCH Modeling

In this section, we fitted GARCH(1,1) models to the negative log returns using both a normal distribution and a standardized t-distribution for the residuals, aiming to capture the time-varying volatility (heteroscedasticity) of the residuals observed in the data.

Both models effectively captured most of the autocorrelation (see Figures 15 and 16), with only a small spike remaining at lag 1. The Ljung-Box test indicated some residual autocorrelation (p-value < 0.05) for both models (see Tables 8 and 9), suggesting minor dependencies left unmodeled. However, the QQ-plots revealed a notable difference (see Figures 17 and 18). The normal GARCH model showed clear deviations in the tails, indicating that the residuals deviate from a normal distribution and that the model did not handle extreme values well. The Shapiro-Wilk test (see Table 10) reject normality for the residuals of the normal GARCH model (p-value < 0.05), further confirming that the residuals do not align with a normal distribution and that the model is inadequate for capturing heavy tails. In contrast, the t-GARCH model's QQ-plot aligned better with the t-distribution quantiles, including in the tails, indicating that it better captured the heavy tails of the data. This confirms that the t-GARCH model more effectively handles extreme values compared to the normal GARCH model.

Despite the differences in tail behavior, the AIC/BIC values for both models were very close (AIC: -10.60 for normal vs. -10.75 for t-distribution), making it difficult to choose one model based on these criteria alone. However, the better fit of the t-GARCH model to the tail behavior suggests it is more appropriate for the data, as it captures the heavy tails more effectively than the normal GARCH model.

e) Two-Step ARIMA-GARCH Modeling

To address the potential issue of residual serial correlation when fitting a GARCH model directly on the negative log returns, we applied a two-step approach as suggested. First, we fitted an ARIMA(2,0,2) model to the negative log returns to capture any autocorrelation in the data. We then extracted the residuals from this ARIMA fit and applied a GARCH(1,1) model to these residuals.

The Box-Ljung test on the residuals of the final GARCH model (see Table 11) showed a non-significant result (p-value=0.9365), indicating that there was no significant autocorrelation left in the residuals. This confirms that the two-step ARIMA-GARCH approach effectively eliminated serial correlation, solving the issue of residual autocorrelation seen when fitting a GARCH model directly on the negative log returns, and leading to a better specified model.

f) Compare the Three Models

We compared the three models: ARIMA(2, 0, 2), GARCH(1, 1) with a normal distribution, and GARCH(1, 1) with a t-distribution, to find the best fit for the Bitcoin negative log returns.

The ARIMA model effectively captured the autocorrelation but struggled with the changing variance (heteroscedasticity), as evidenced by volatility clustering in the residuals. This indicates a violation of the homoscedasticity assumption, since ARIMA cannot model time-varying variance. Additionally, the ARIMA model failed to account for the heavy tails in the data, as its residuals showed deviations from a normal distribution in the extremes.

In contrast, the GARCH models handled the heteroscedasticity issue well, as they are specifically designed to model time-varying variance. The normal GARCH model captured most of the autocorrelation but did not fit the heavy tails effectively, showing deviations in the extremes. The t-GARCH model, however, provided a better fit for the tail behavior, as indicated by the alignment with the t-distribution quantiles in the QQ-plot.

Overall, the t-GARCH model is the most suitable, effectively handling both the heteroscedasticity and the heavy tails typical in financial time series, unlike the ARIMA model and the normal GARCH model.

Part 3: Dependence between time series

Appendices

Practical 1

Figures

Figure 1: Bitcoin Prices and Negative Log Returns Over Time on Common Scale

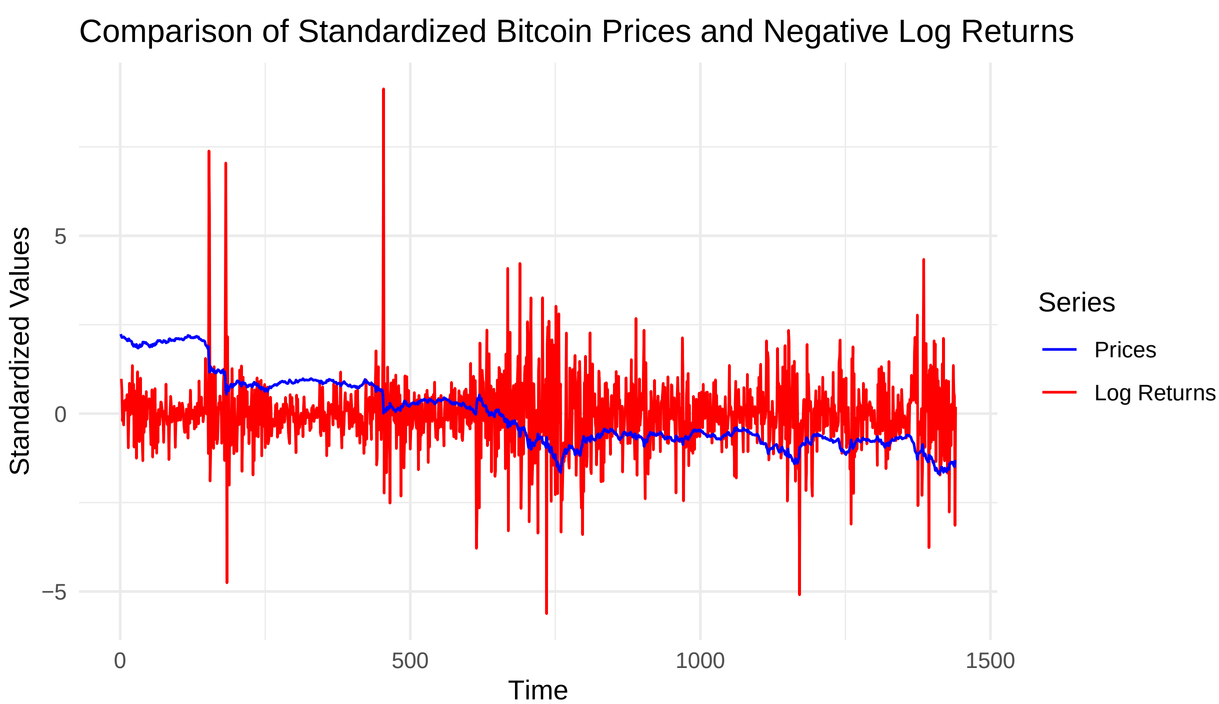


Figure 2: Bitcoin Prices Over Time

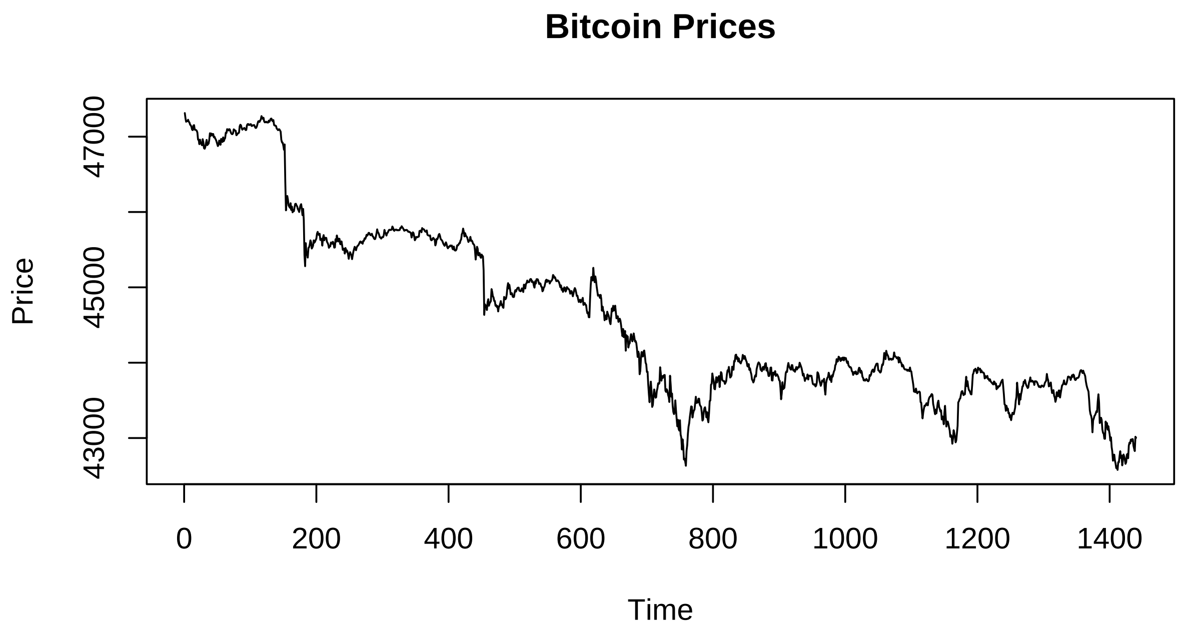


Figure 3: Negative Log Returns of Bitcoin Over Time

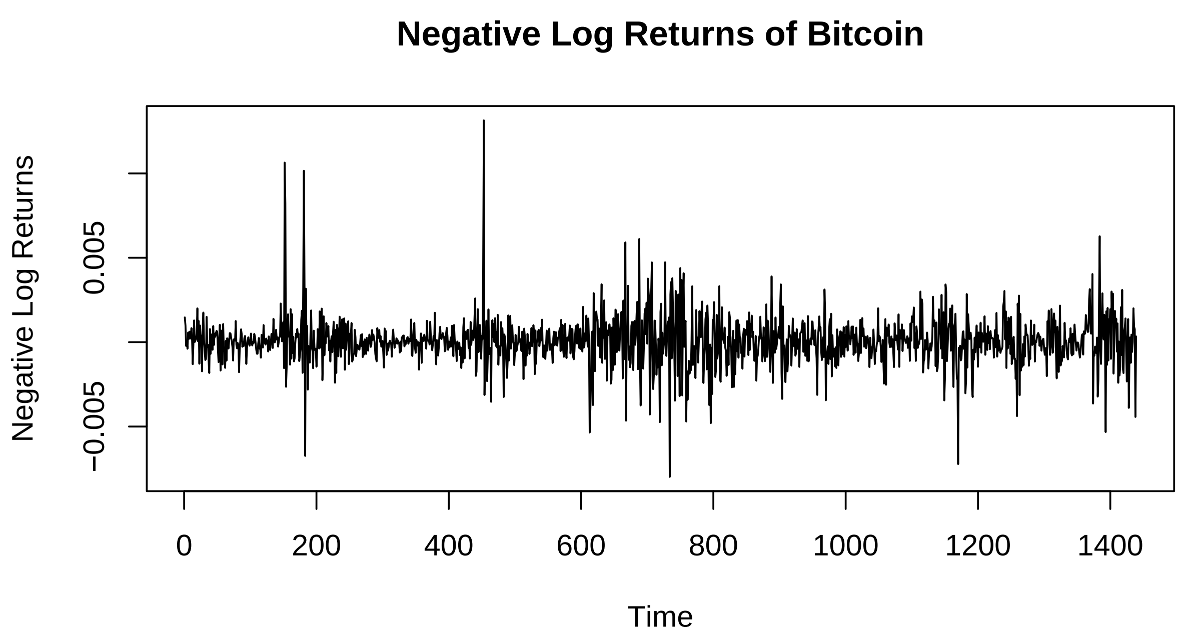


Figure 4: Histogram of Negative Log Returns

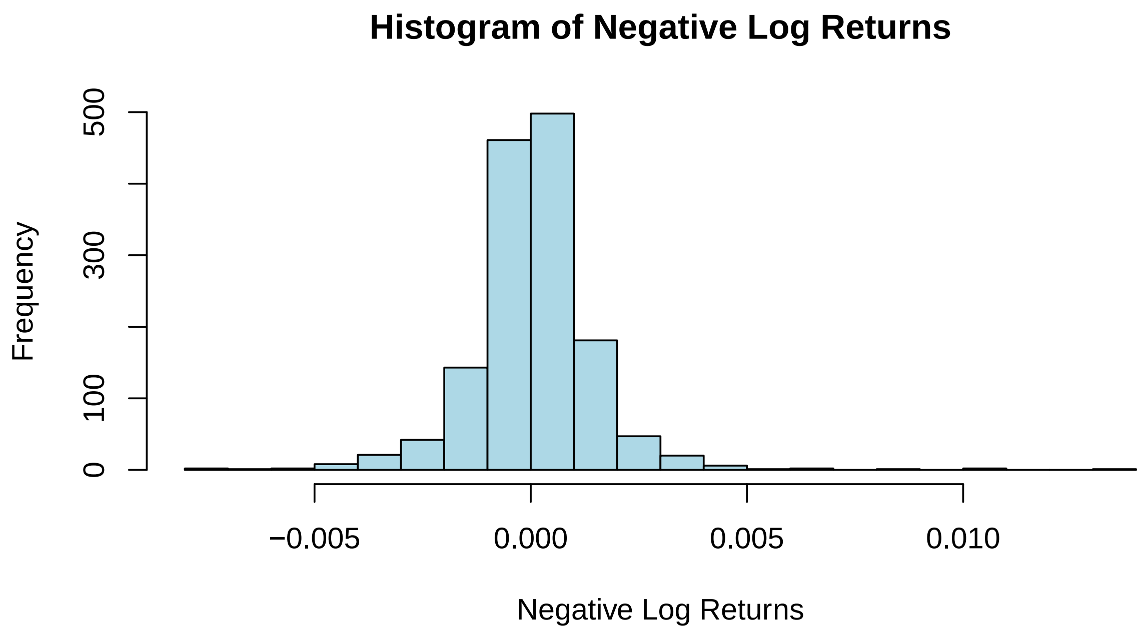


Figure 5: QQ-Plot of Negative Log Returns

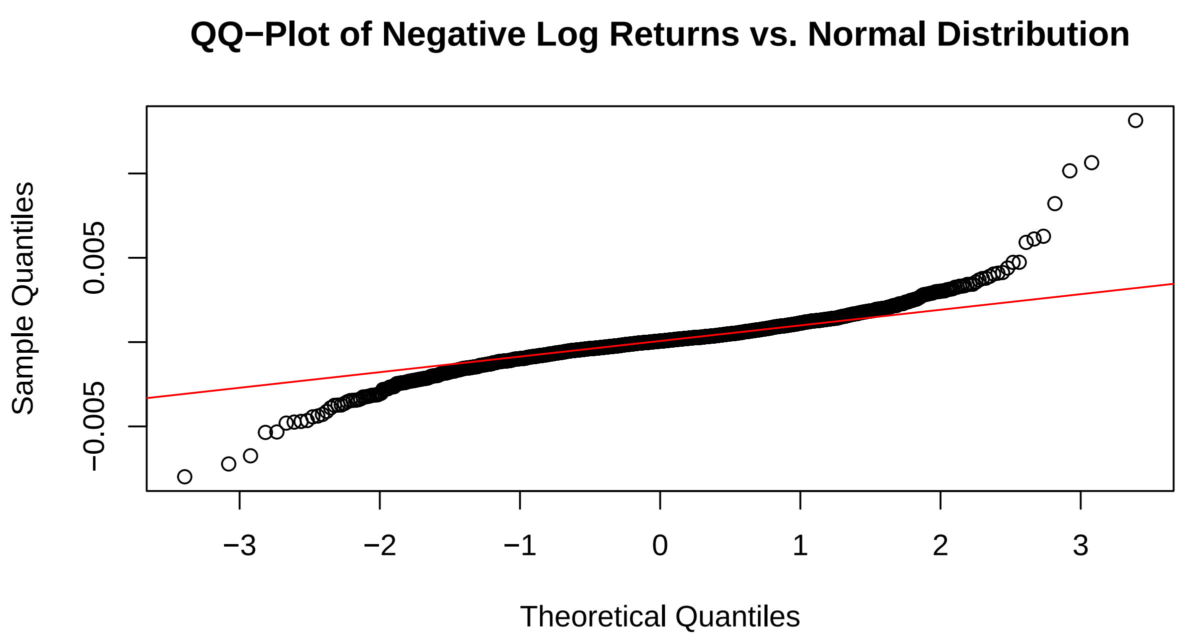


Figure 6: QQ-Plot of Negative Log Returns with t-Distribution

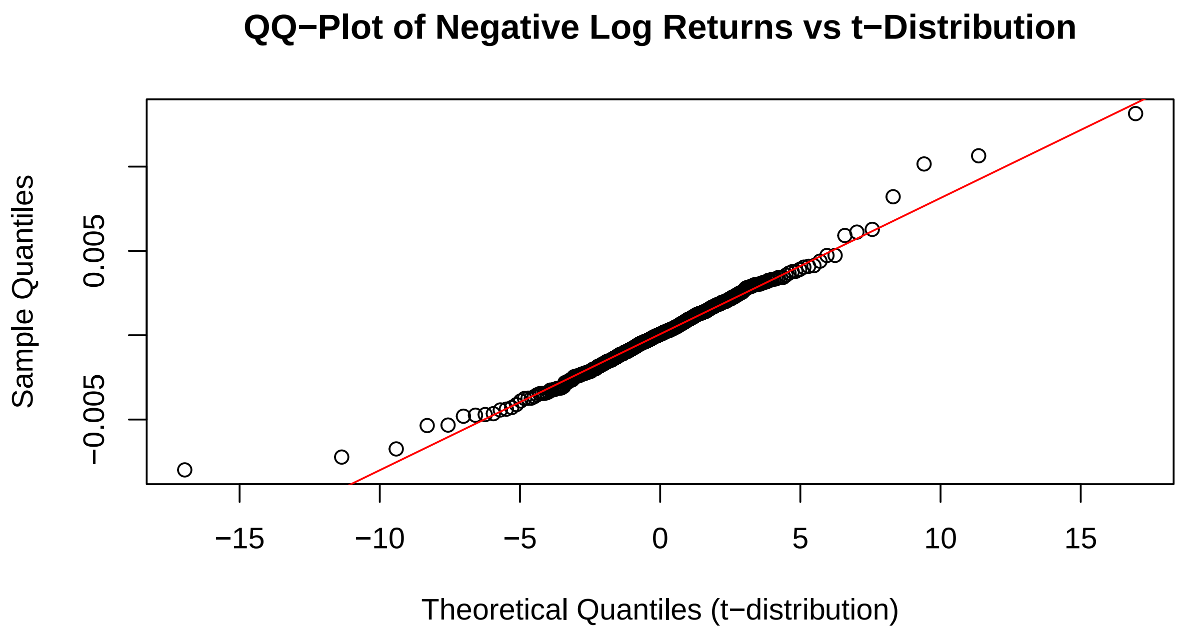


Figure 7: Histogram of Bitcoin Negative Log Returns with Fitted t and Normal Distribution

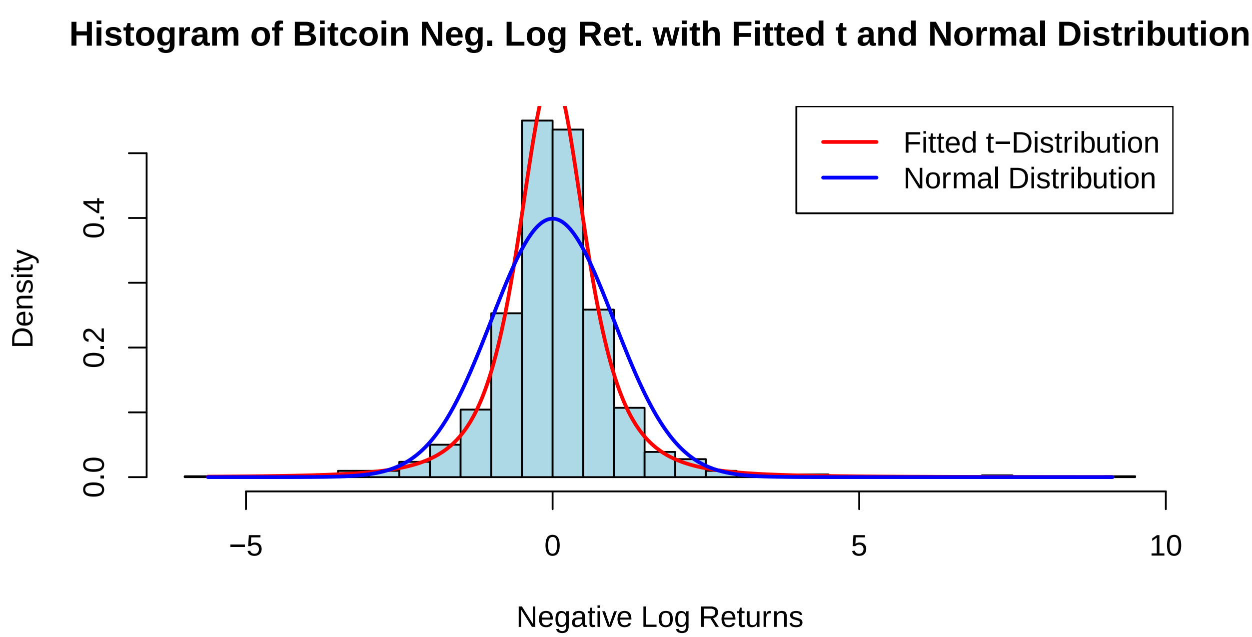


Figure 8: Density Comparison: Normal vs t-Distribution

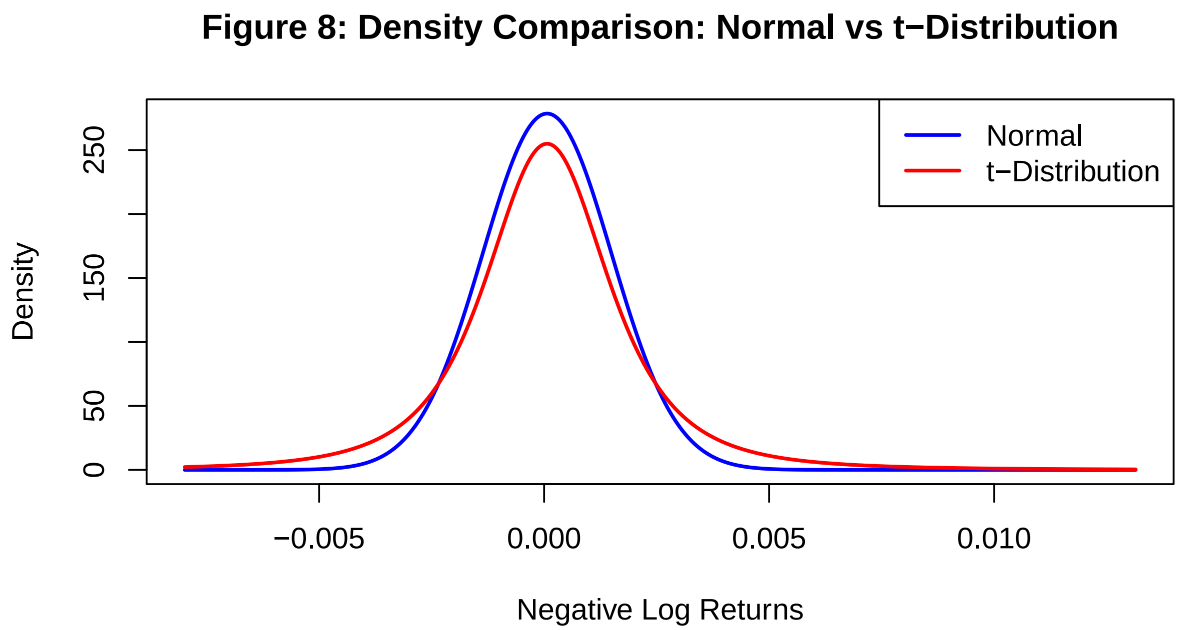


Figure 9: ACF of Bitcoin Prices

Figure 10: ACF of Negative Log Returns

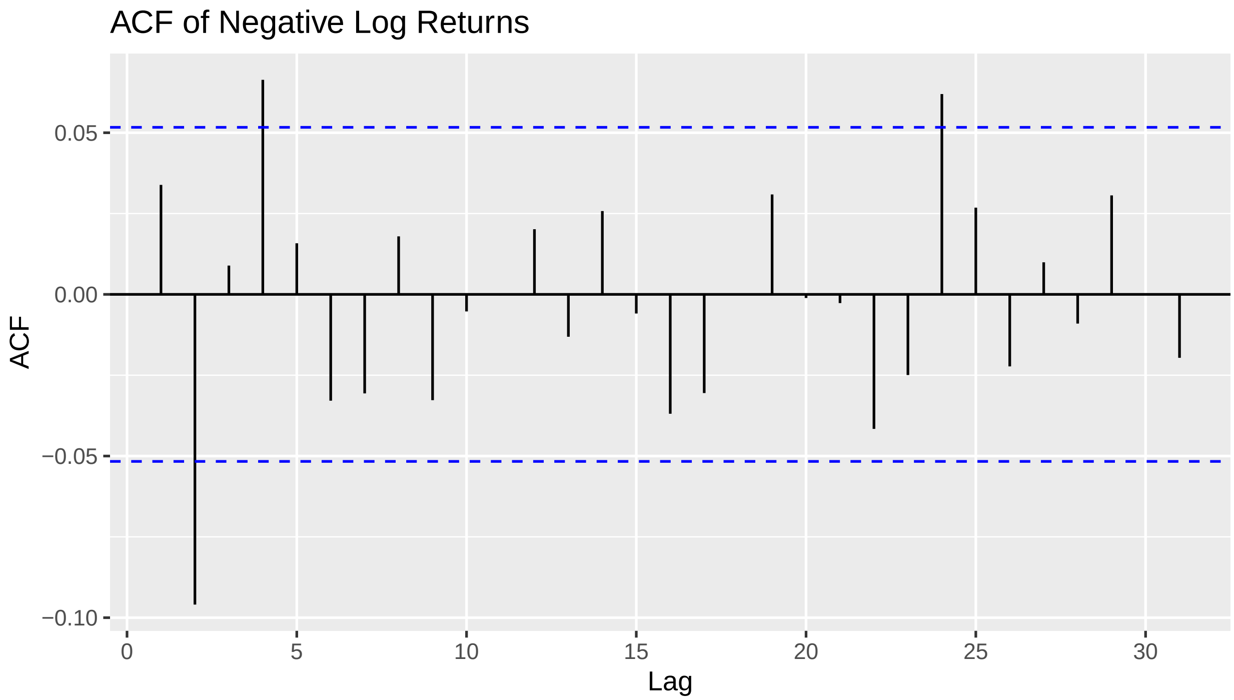


Figure 11: PACF of Negative Log Returns

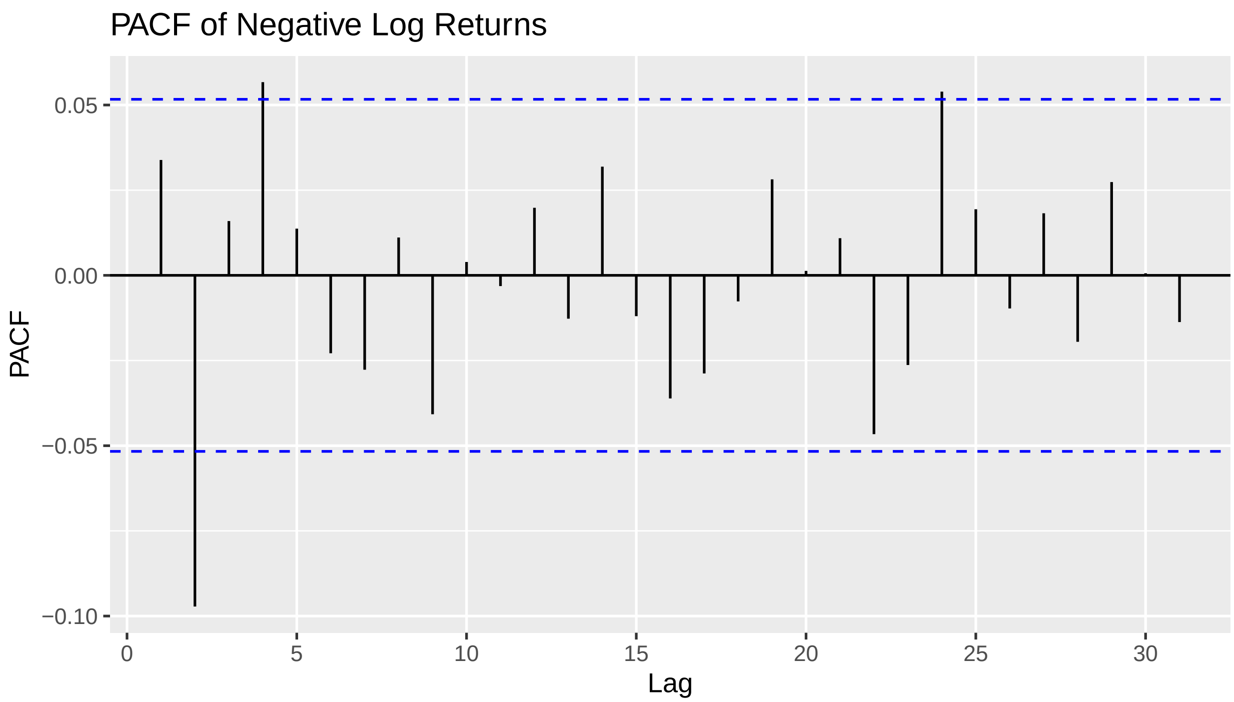


Figure 12: ACF of ARIMA(2, 0, 2) Residuals

```
##
## Call:
## arima(x = bitcoin_negative_log_returns, order = c(2, 0, 2))
##
## Coefficients:
##           ar1      ar2     ma1     ma2  intercept
##       -0.0520  -0.5415  0.0853  0.4479      1e-04
## s.e.   0.1717   0.1664  0.1824  0.1773      0e+00
##
## sigma^2 estimated as 2.022e-06:  log likelihood = 7391.82,  aic = -14771.65
```

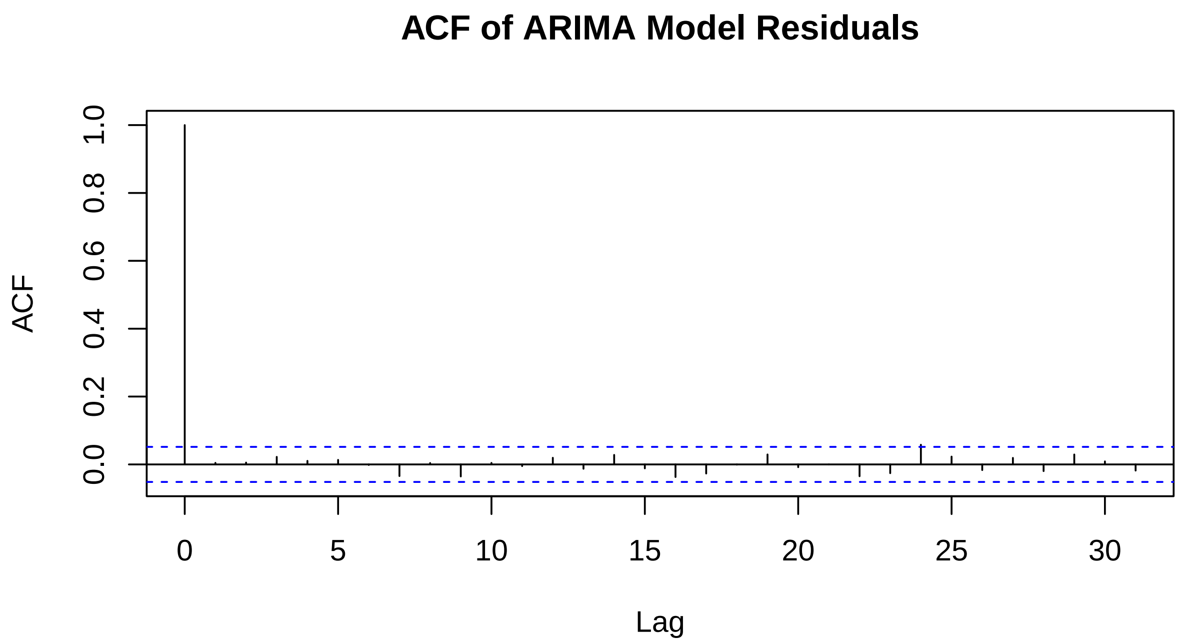


Figure 13: QQ-Plot of ARIMA(2, 0, 2) Residuals

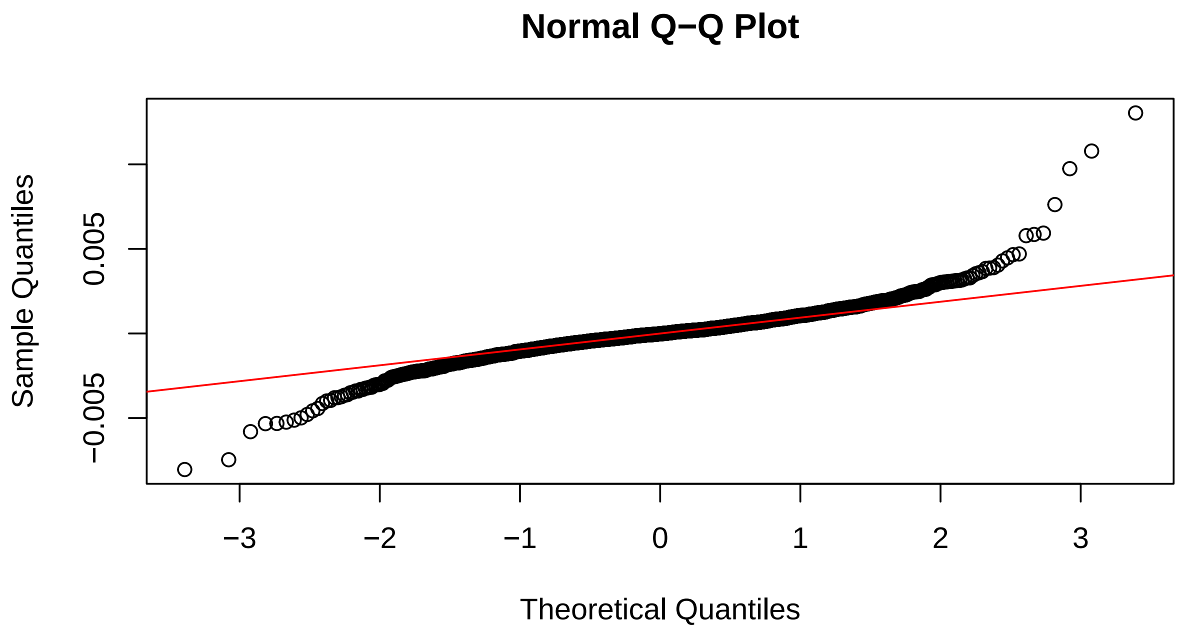


Figure 14: Residuals of ARIMA(2, 0, 2) Over Time

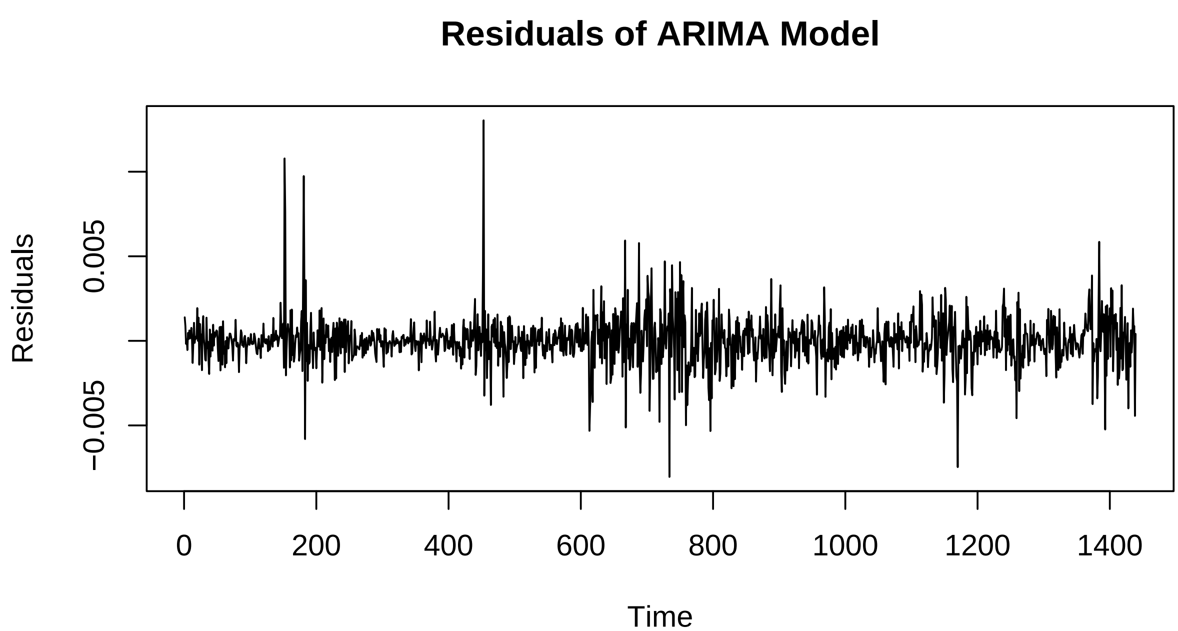


Figure 15: ACF of GARCH Normal(1, 1)

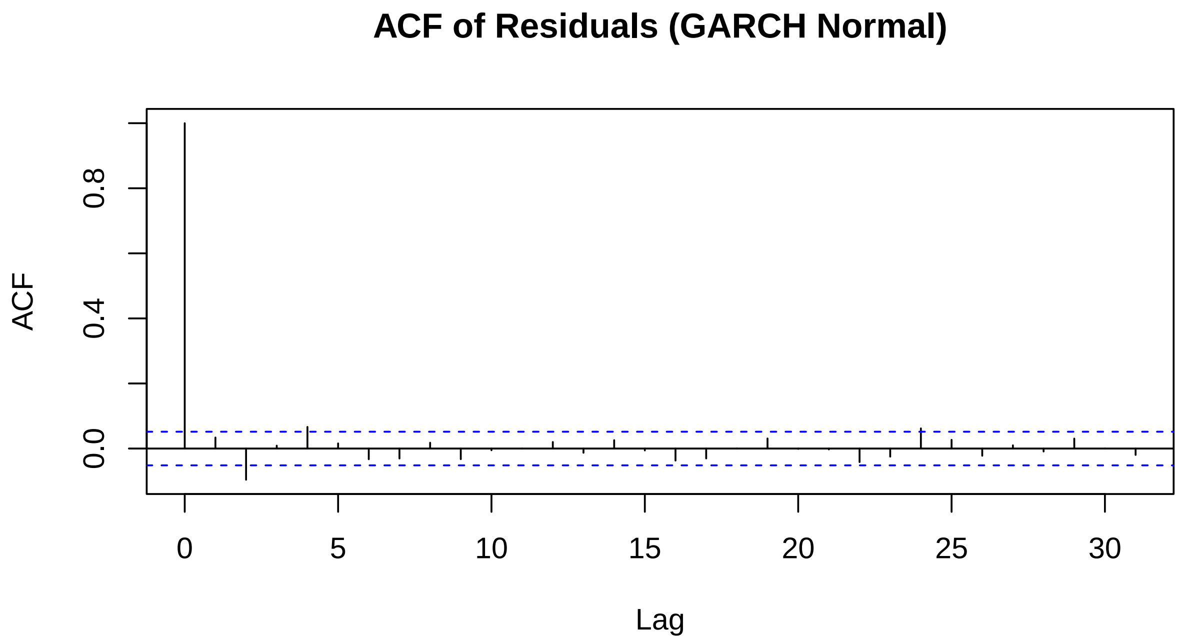


Figure 16: ACF of GARCH t-Distribution(1, 1)

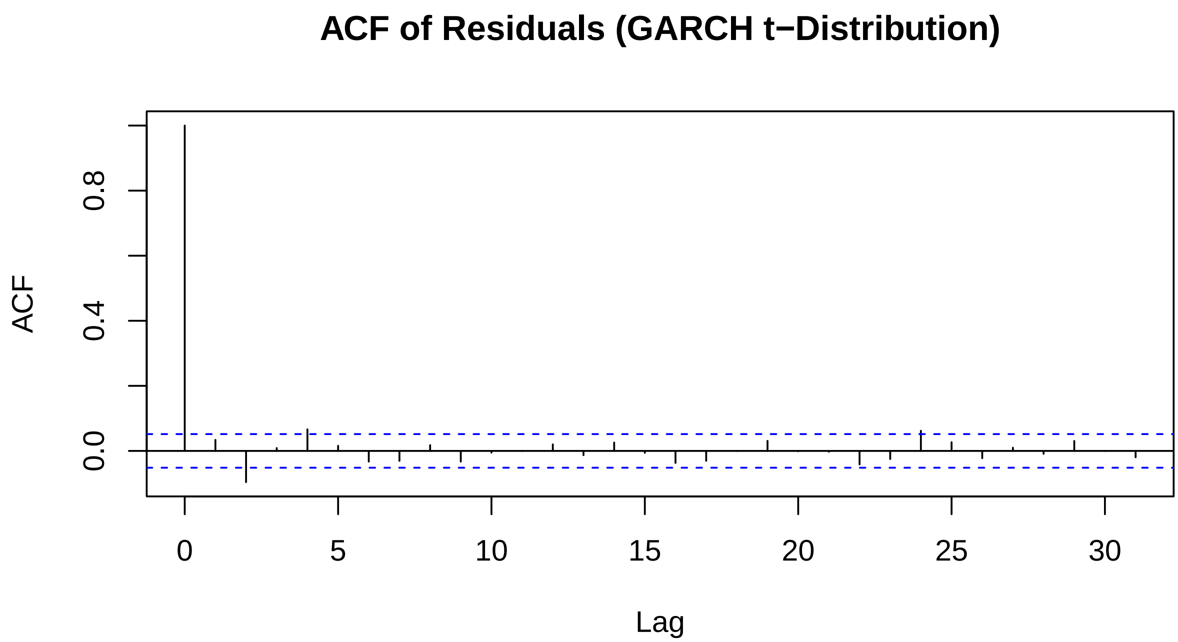


Figure 17: QQ-Plot of GARCH Normal(1, 1) Residuals

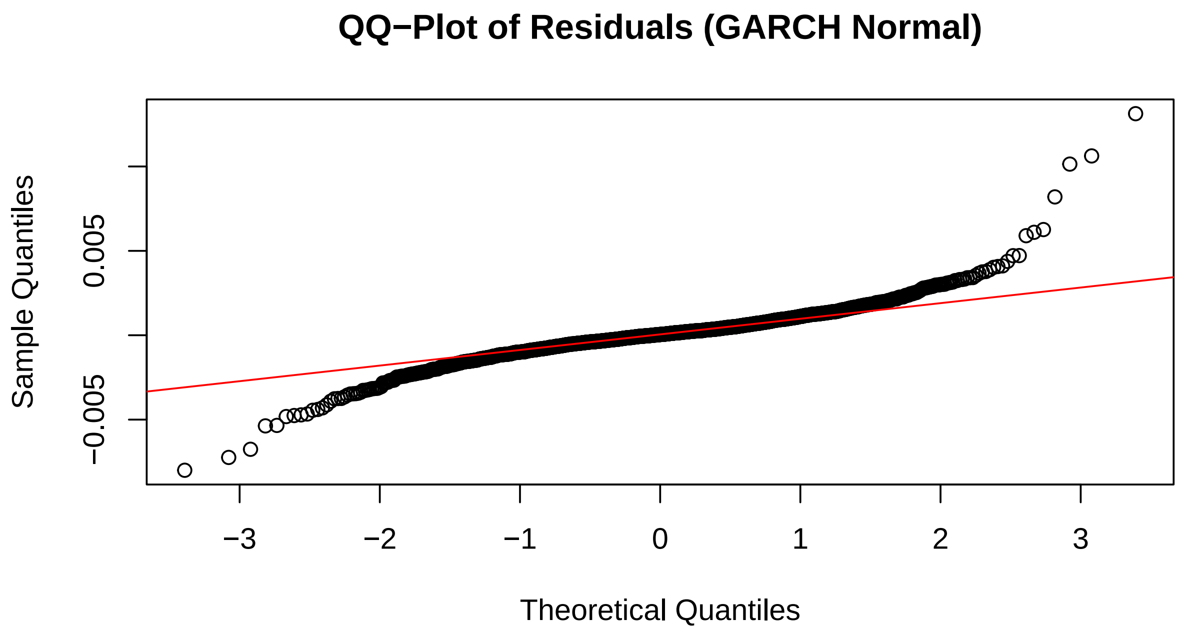
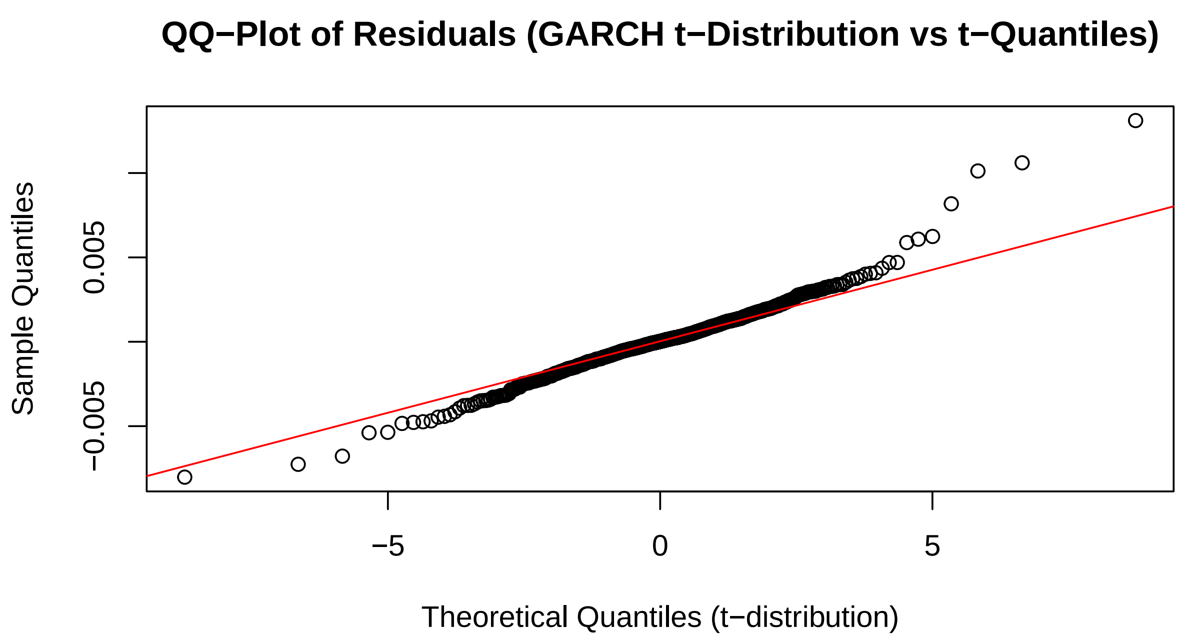


Figure 18: QQ-Plot of GARCH t-Distribution(1, 1) Residuals



Results tables

Table 1: Augmented Dickey-Fuller Test for Bitcoin Prices

```
##
##  Augmented Dickey-Fuller Test
##
## data:  bitcoin_prices
## Dickey-Fuller = -2.4484, Lag order = 11, p-value = 0.3885
## alternative hypothesis: stationary
```

Table 2: Augmented Dickey-Fuller Test for Negative Log Returns

```
##
##  Augmented Dickey-Fuller Test
##
## data:  bitcoin_negative_log_returns
## Dickey-Fuller = -11.035, Lag order = 11, p-value = 0.01
## alternative hypothesis: stationary
```

Table 3: Anderson-Darling Test for Normality of Negative Log Returns

```
##
##  Anderson-Darling normality test
##
## data:  bitcoin_negative_log_returns
## A = 26.277, p-value < 2.2e-16
```

Table 4: Ljung-Box Test for Autocorrelation in Bitcoin Prices

```
##
##  Box-Ljung test
##
## data:  bitcoin_prices
## X-squared = 26873, df = 20, p-value < 2.2e-16
```

Table 5: Ljung-Box Test for Autocorrelation in Negative Log Returns

```
##
##  Box-Ljung test
##
## data:  bitcoin_negative_log_returns
## X-squared = 33.356, df = 20, p-value = 0.03082
```

Table 6: Ljung-Box Test for ARIMA(2, 0, 2) Residuals

```
##
##  Box-Ljung test
##
## data:  residuals_arima
## X-squared = 11.355, df = 20, p-value = 0.9365
```

Table 7: Shapiro-Wilk Test for Normality of ARIMA(2, 0, 2) Residuals

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals_arima
## W = 0.89619, p-value < 2.2e-16
```

Table 8: Ljung-Box Test for GARCH Normal(1, 1) Residuals

```
##
##  Box-Ljung test
##
## data:  garch_normal_residuals
## X-squared = 33.356, df = 20, p-value = 0.03082
```

Table 9: Ljung-Box Test for GARCH t-Distribution(1, 1) Residuals

```
##
##  Box-Ljung test
##
## data:  garch_t_residuals
## X-squared = 33.356, df = 20, p-value = 0.03082
```

Table 10: Shapiro-Wilk Test for Normality of GARCH Normal(1, 1) Residuals

```
##
##  Shapiro-Wilk normality test
##
## data:  garch_normal_residuals
## W = 0.89321, p-value < 2.2e-16
```

Table 11: Ljung-Box Test for ARIMA-GARCH Residuals

```
##
##  Box-Ljung test
##
## data:  garch_residuals
## X-squared = 11.355, df = 20, p-value = 0.9365
```