# Forecasting daily Bitcoin volatility using GARCH models and intraday data

**Davis Vaughan**

This version was compiled on December 7, 2017

**GARCH modeling has been a topic of interest for many years, with much discussion around the forecasting performance of the models in the context of equities, but relatively few papers have applied the technique to crytocurrencies. The rapid growth of Bitcoin over the past few years, coupled with its highly volatile price, makes it a perfect candidate for analysis. This paper produces daily forecasts of variance through an aggregation of intraday forecasts, and compares the performance against measures of realized variance. Aggregation methods are found to be more accurate than 1-step ahead daily forecasts.**

GARCH | intraday | Bitcoin

## 1. Introduction

Forecasting daily volatility accurately is an important input into many applications in the financial industry. From a risk management perspective, Value at Risk (VaR) relies heavily on having an accurate measure of volatility. In fact, for the simple variance covariance method of estimating Value at Risk using a Normal distribution, the 95% estimate is simply $-1.65\sigma$. More generally, understanding future volatility allows a firm to hedge appropriately, and strategically place themselves for the times ahead.

Since the creation of GARCH modeling by Bollershev, Bollerslev (1986), there has been an explosion of literature extending the GARCH model to be more flexible, to include external regressors, and to attempt to capture asymmetric effects of returns. One of the most interesting papers, co-authored by Bollerslev himself, attempts to combat the argument that, while GARCH models often seem to fit well in-sample, they have poor forecasting performance, Andersen and Bollerslev (1998). He argues that the ex-post measure of variance commonly being used, the daily squared return, is a poor estimate of the true variance that one should measure performance against. Rather, one should use a summation of squared intraday returns aggregated to the daily level, termed *realized variance*, as a measure of that day's variance. Such an approach is one of the focuses of this paper.

The second focus is on an extension of Ñíguez's approach of forecasting monthly volatility of the EuroStoxx 50, Ñíguez (2008). There, he fits a GARCH model to daily data, forecasts the next month's worth of daily volatility, rolls the model forward one month, and repeats the process. The daily volatility forecasts for each month are then squared and summed to form a forecast of the next month's variance, similar in spirit to the concept of realized variance.

In this paper, Bitcoin volatility is forecasted using GARCH(1,1) models under the Normal and Student-t distributions. Models are fit using 5-minute data, and daily forecasts are created from aggregated 5-minute forecasts. These forecasts are compared against those generated from models fit to daily data. Model performance is evaluated using mean squared error (MSE), mean absolute percent error (MAPE), and $R^2$ coefficients from Minzer-Zarnowitz (MZ) regressions.

## 2. Volatility modeling

Let $r_{t_i} = log(P_{t_i}) - log(P_{t_{i-1}})$ represent the i-th 5-minute log return. The index t will refer to 5-minute sampling. Additionally, the j-th daily log return is denoted as $r_{d_j} = log(P_{d_j}) - log(P_{d_{j-1}})$. Because Bitcoin is continuously traded 24/7, there are 288 5-minute samples per day.

The data set used contains 5-minute prices for the Bitstamp exchange over the period from 2016-01-01 to 2017-10-20 gathered from the Kaggle competition, Bitcoin Historical Data. A link to the competition is provided in the references under Zielak (2017). Data after 2017-06-30 23:55:00 are considered out of sample. Figure 1 displays some descriptive plots of the 5-minute data along with the daily data. Notably, Bitcoin, like most other securities, exhibits the fact that volatility is not constant throughout time.

Volatility is modeled using GARCH(1,1) models, fitted at both the 5-minute and daily levels. Two distributions are used for returns, Normal and Student-t. For simplicity, the mean of the return process is set to 0. Additionally, after finding it insignificant, the constant in the GARCH model is set to 0. The resulting model for 5-minute sampling is,

$$\begin{aligned} r_{t_i} &= \epsilon_{t_i} \\ \epsilon_{t_i} &= \sigma_{t_i} z_{t_i} \\ \sigma_{t_i} &= \alpha \epsilon_{t_{i-1}} + \beta \sigma_{t_{i-1}} \end{aligned} \tag{1}$$

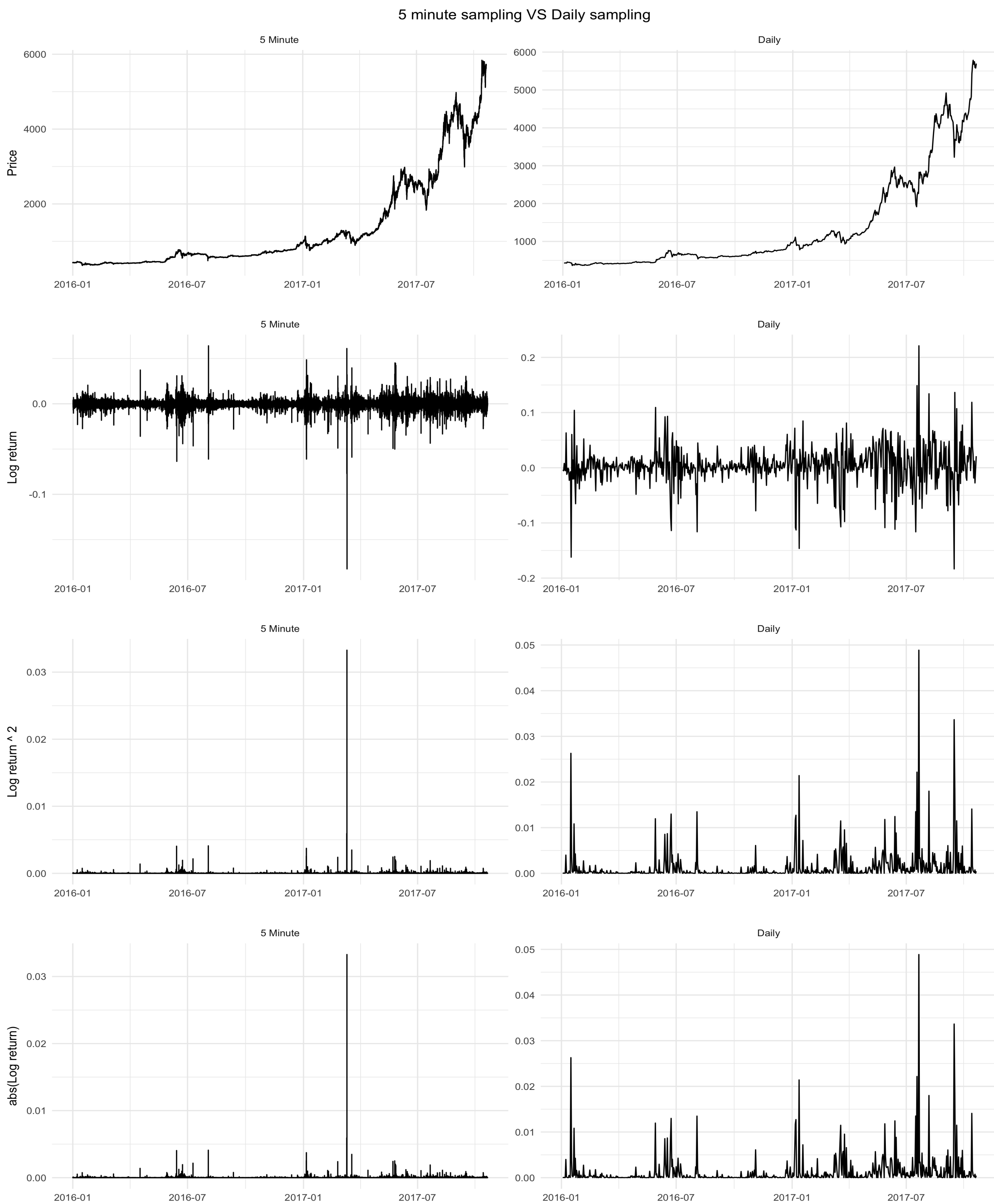


**Fig. 1.** Descriptive plots of 5-minute and daily returns

with $z_{t_i}$ represented as either $z_{t_i} \sim N(0,1)$ or $z_{t_i} \sim t_\nu$ depending on the underlying distribution. To differentiate the two distributions, GARCH(1,1)-N and GARCH(1,1)-T will denote GARCH(1,1) under the Normal and Student-t distributions respectively. The daily model is defined similarly, with $t_i$ replaced by $d_j$.

## 3. In sample estimation results

The results of the GARCH(1,1) in sample estimations are reported below in Table 1. Consistent with literature, $\alpha$ and $\beta$ estimates sum to a value close to 1, demonstrating the highly persistent nature of volatility at both the 5-minute and daily levels. Both sampling intervals produce a degrees of freedom estimate near 4, resulting from the heavy tailed nature of the returns. Not shown here are additional fits from GJR-GARCH models that attempted to capture any asymmetric properties of the distribution. Interestingly, none of the asymmetric parameters were statistically significant, suggesting that Bitcoin does not react more strongly to negative movements.

**Table 1. Estimation results of GARCH(1,1) in sample fits under different distributions and at different sampling intervals.**

| | Estimate | Robust Std Error | P-value |
|---|---|---|---|
| **Normal: 5-min** | | | |
| $\alpha$ | 0.048 | 0.001 | 0.000 |
| $\beta$ | 0.951 | 0.001 | 0.000 |
| **Student-t: 5-min** | | | |
| $\alpha$ | 0.108 | 0.003 | 0.000 |
| $\beta$ | 0.891 | 0.004 | 0.000 |
| $\nu$ | 3.927 | 0.030 | 0.000 |
| **Normal: daily** | | | |
| $\alpha$ | 0.086 | 0.027 | 0.001 |
| $\beta$ | 0.913 | 0.030 | 0.000 |
| **Student-t: daily** | | | |
| $\alpha$ | 0.142 | 0.027 | 0.000 |
| $\beta$ | 0.857 | 0.031 | 0.000 |
| $\nu$ | 3.619 | 0.318 | 0.000 |

## 4. Volatility forecasting and realized variance

The forecasting methods used can be broken down into two sections: the procedure used for including new information in the forecasts, and the performance measure used to validate against.

### 4.1. Forecasting methodology.

For daily sampling, the forecasting procedure is as follows:

1) An initial GARCH(1,1) model is fit using the first 546 daily returns.
2) A 1-step ahead forecast of the next day's variance is generated, denoted $\hat{h}_{d_{j+1}}$.
3) Using a moving window, the oldest day's return is dropped, the newest day's return is included, and the model is refit.
4) Steps 2 and 3 are repeated to generate 111 daily forecasts of variance.

For 5-minute sampling, the forecasting procedure is slightly more complicated to avoid any look ahead bias. The procedure is:

1) An initial GARCH(1,1) model is fit using the first 157535 5-minute returns. This ends the fit on 2017-06-30 23:55:00, the end of that day.
2) The next day's worth of 5-minute variance forecasts are generated recursively. This results in 288 5-minute forecasts for the next day (24 hours x 60 minutes / 5 minutes).
3) Those 5-minute forecasts are summed to generate a forecast of variance for that day, denoted $\hat{H}_{d_{j+1}} = \sum_{i=1}^{288} \hat{h}_{t_{i+288(j+1)}}$. The notation of $288(j+1)$ is used to ensure that the correct day's worth of 5-minute variance forecasts are summed.
4) Using a moving window, the oldest 288 5-minute returns are dropped, the newest 288 5-minute returns are included, and the model is refit. This effectively shifts the model 1 day forward.
5) Steps 2-4 are repeated to generate 111 daily forecasts of variance.

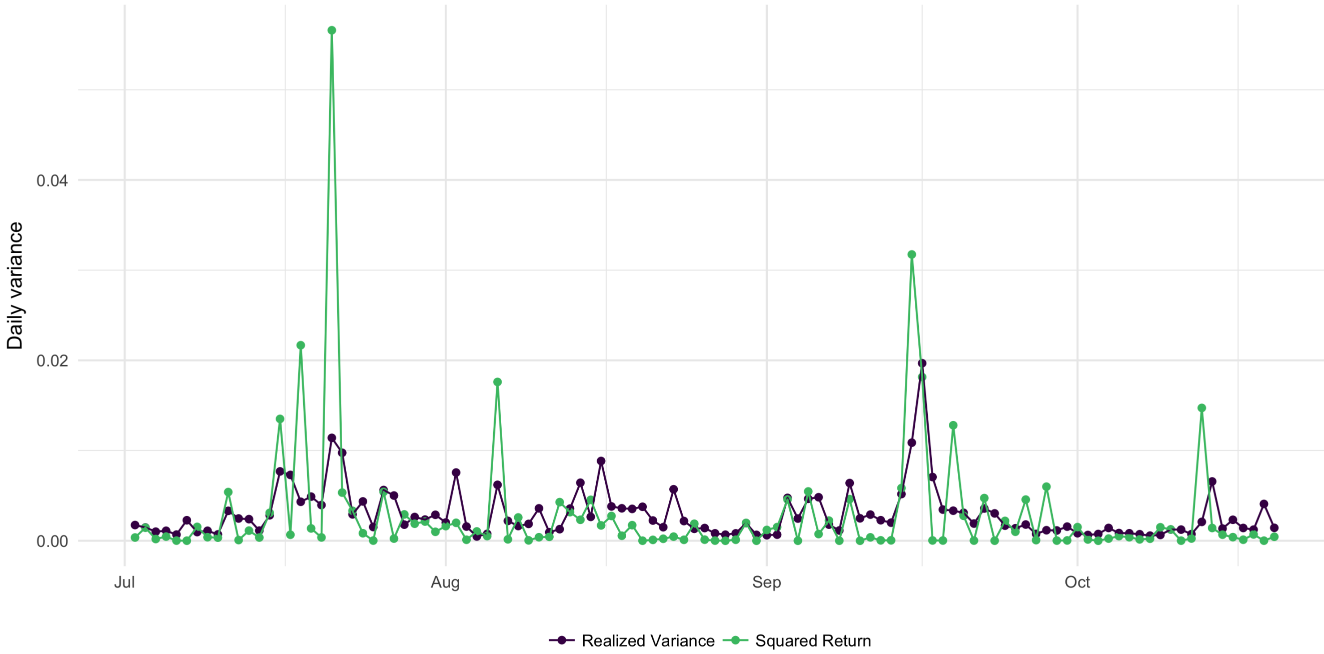


**Fig. 2.** Realized variance VS Squared return in the out of sample period. Squared return is very noisy, and not the best estimator to benchmark performance against.

Ideally, the aggregation approach allows the model to be more flexible, and take into account a finer level of detail. These approaches were repeated for both Normal and Student-t distributions.

To generate n-step ahead forecasts under the GARCH(1,1) model, the following recursive formula was used:

$$\hat{h}_{t+n} = (\alpha+\beta)^{n-1}\hat{h}_{t+1} \tag{2}$$

**4.2. Realized variance.** Equally as important as the forecast methodology is the proxy of variance that one measures performance against. Because volatility is unobservable, some estimate of the true volatility is required to calculate any kind of performance measure. A common measure of forecasting performance for daily variance is to use that day's squared return, $r^2_{d_{j+1}}$. However, as noted by Bollershev, while this is a consistent estimator of conditional variance, it is incredibly noisy, Bollerslev (1986). Bollershev proposes the use of intraday information to estimate the daily variance. The technique, termed *realized variance*, is adapted here as the sum of squared returns for the 288 5-minute returns in each day. Formally:

$$RV_{d_{j+1}} = \sum_{i=1}^{288} r^2_{t_{i+288(j+1)}}$$

Both the daily squared return and the realized variance will be used to generate performance metrics for the models, and their uses as benchmarks will be compared.

Figure 2 demonstrates the difference between the proxies of realized variance and the squared daily return. Squared return is incredibly noisy in comparison to realized variance, especially in periods of higher volatility, where the estimate can spike to unrealistically high amounts.

## 5. Forecasting results

The plot in Figure 3 shows the out of sample forecasting performance of the GARCH(1,1)-N models in comparision to the realized variance. The daily forecasts generated from aggregated 5-minute forecasts have more flexibility to adapt quickly to the movements in the level of variance compared to the GARCH(1,1)-N model fit to daily data, especially in transition periods from high to low volatility.

The extreme forecast from the aggregation model resulting in variance above 0.04 in mid July is of cause for concern, and discussion of this is addressed in the next section.

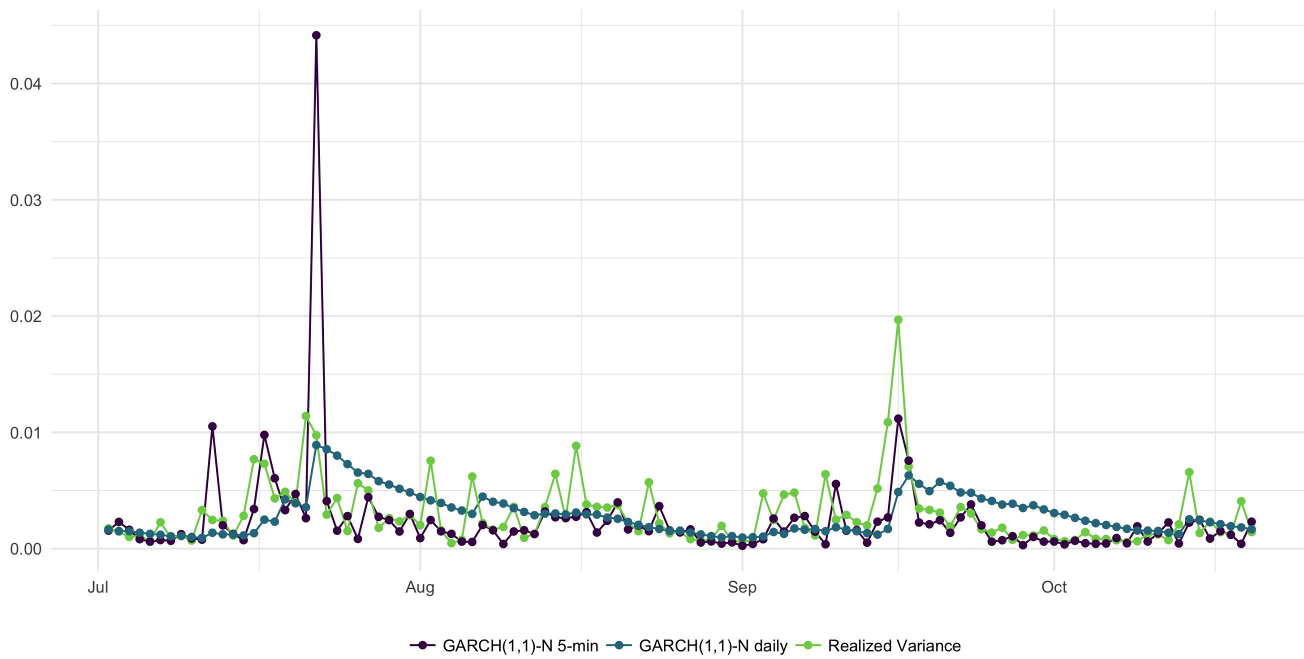


**Fig. 3.** Out of sample performance of GARCH(1,1) models under the Normal distribution. Daily forecasts of variance from aggregated 5-minute forecasts appear to be more flexible in adapting quickly to changes in the overall level of variance, especially in transition periods from high to low variance like mid September.

Figure 4 compares the Normal and Student-t forecasts for the 5-minute aggregation technique. Student-t forecasts tend to be higher in periods of high volatility, and slightly lower in periods of lower volatility, but overall they follow a similar track.

Table 2 displays a number of performance metrics from the GARCH(1,1) models under the Normal distribution. For brevity, Student-t results are not included as they are not very different from Normal. Included metrics are Mean Absolute Percent Error (MAPE), Root Mean Squared Error (RMSE), and the intercept, slope and $R^2$ from Minzer-Zarnowitz (MZ) regression of the corresponding proxy on the forecasts. For all metrics, the realized variance proxy implied the models forecasted better than when compared against the squared daily return proxy.

The first two columns correspond to the 5-minute aggregation method of daily forecasts, and the forecasts from daily models respectively. While the MAPE of the 5-minute method is lower than for daily, the RMSE is much higher. This is a direct result of the incredibly large forecast above 0.04 from the 5-minute method that can be seen in Figure 3. Column 3 contains the metrics calculated again for the 5-minute method, but with the removal of that one large forecast. After removing that single point, the RMSE drops below the daily method, the MZ slope coefficient jumps up to a value much closer to 1, and the MZ $R^2$ increases significantly. This finding prompted a further analysis to attempt to understand the outlier, outlined in the next section.

## 6. Applying a filter

The large variance forecast from the 5-minute model on 2017-07-21 results from a highly volatile period immediately before the end of the day on 2017-07-20. The 5-minute returns near the end of that day were highly out of the ordinary, which resulted in forecasts in the 5-minute model that were very high. The high persistance of the model predicted a very slow decay in the variance over the next day, resulting in the aggregate forecast of variance for that day that was much larger than the realized variance.

As a potential remedy to this, a filter was applied to the returns to remove outliers before running the GARCH models and making the forecasts. Specifically, values outside 5 standard deviations of the mean were replaced with the mean. Figure 5 displays the results of the forecasts before and after the filtration. The filtration “fixed” the large variance forecast, and otherwise the forecasts look very similar. Table 3 displays the additional performance metrics for the filtered Normal model. The performance is similar to the results from Table 2 with the 1 forecast outlier removed.

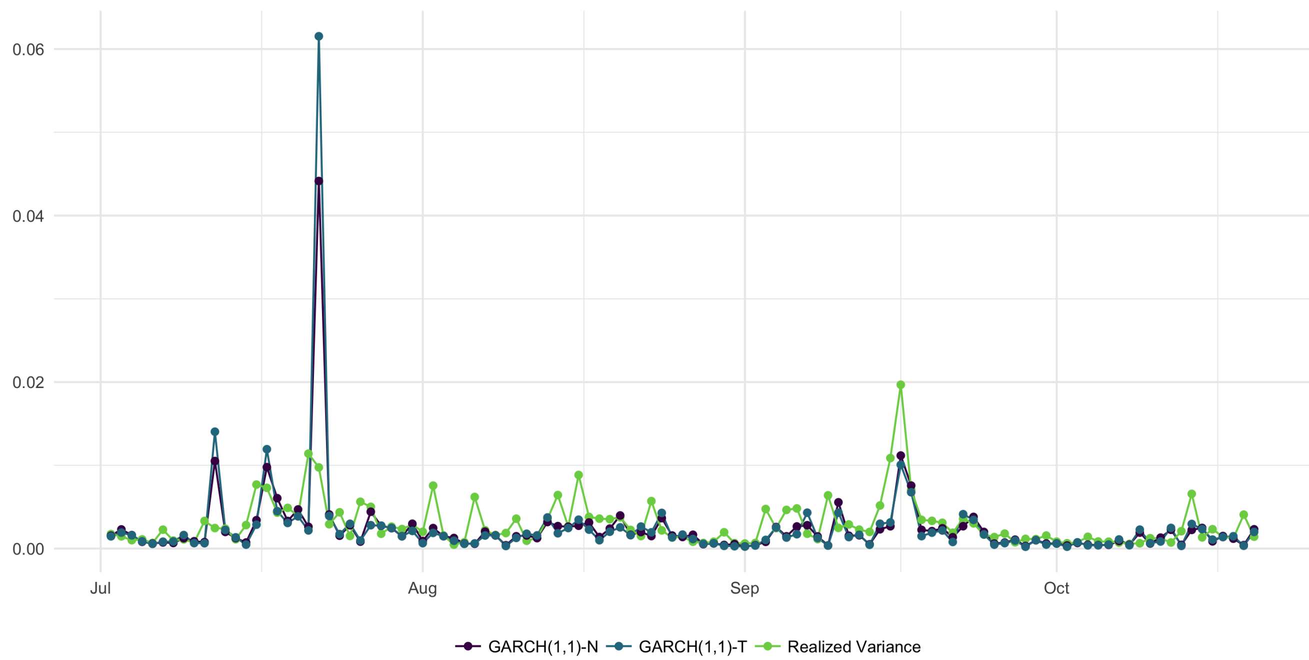


**Fig. 4.** Out of sample performance of Normal VS Student-t GARCH(1,1) models using the 5-minute aggregation technique. Student-t forecasts tend to swing more wildly, but overall there is not much difference.

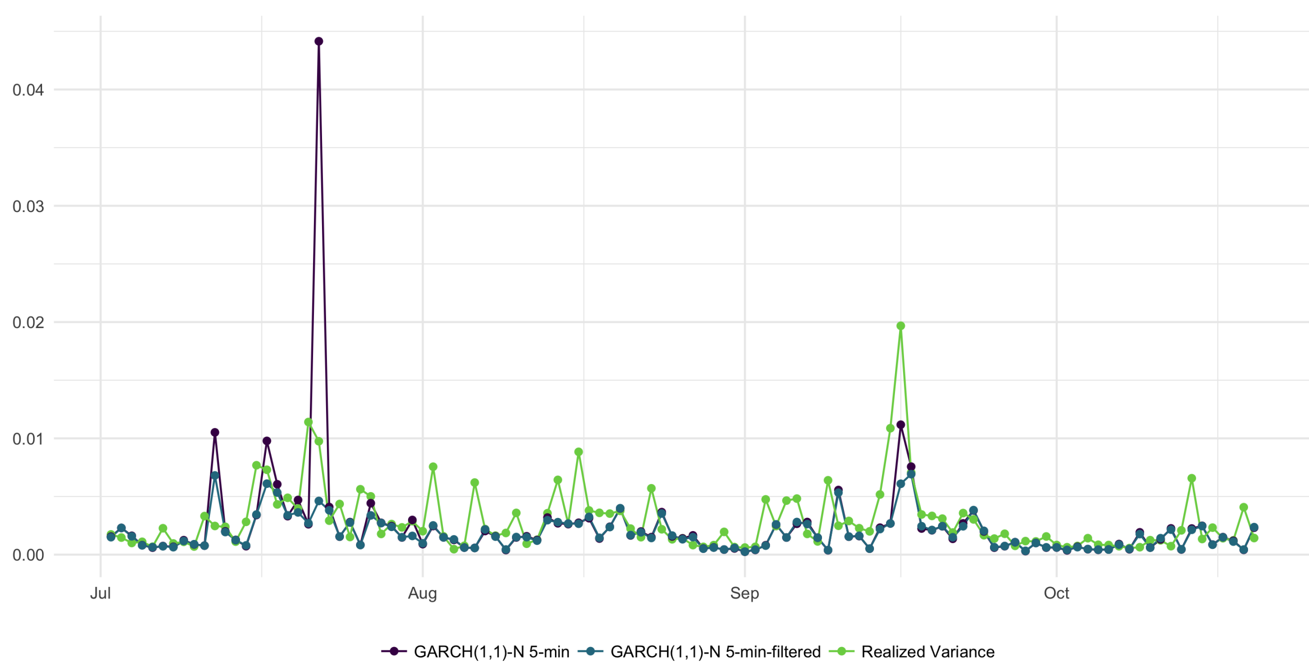


**Fig. 5.** Comparing filtered 5-minute variance forecasts with 5-minute variance forecasts. The filtration dampned the effects of the unusual 5-minute returns in July.

**Table 2. Out of sample performance of GARCH(1,1) Normal models. MAPE for 5-min is lower than for daily. Across the board, using RV as a proxy over $r^2$ gives more accurate results. Removing the 1 extreme forecast from the 5-min method results in a much higher MZ $R^2$, and a MZ slope much closer to 1.**

| | 5-Min | Daily | 5-Min No Outlier |
|---|---|---|---|
| **Proxy: RV** | | | |
| MAPE | 0.0278 | 0.0372 | 0.0249 |
| RMSE | 0.0041 | 0.0029 | 0.0024 |
| MZ Intercept | 0.0023 (0.00027) | 0.0017 (0.00049) | 0.0013 (0.00031) |
| MZ Slope | 0.2817 (0.05404) | 0.4415 (0.14418) | 0.7984 (0.113) |
| MZ $R^2$ | 0.1996 | 0.0792 | 0.3161 |
| **Proxy: $r^2$** | | | |
| MAPE | 0.2811 | 0.3247 | 0.2788 |
| RMSE | 0.0079 | 0.0071 | 0.0070 |
| MZ Intercept | 0.0025 (0.00076) | 0.0023 (0.00127) | 0.0016 (0.00096) |
| MZ Slope | 0.1610 (0.15063) | 0.1961 (0.37621) | 0.6067 (0.34722) |
| MZ $R^2$ | 0.0104 | 0.0025 | 0.0275 |

## 7. Conclusion

This paper analyzed the daily volatility of Bitcoin returns using GARCH modeling. Of note, two forecasting techniques were used, a traditional forecasting method where the model is fit to daily data to predict daily data, and a more granular model fit to 5-minute data where daily forecasts are generated from aggregating multi-step ahead 5-minute forecasts. The performance was measured against two proxies, the common, but noisy, daily squared return, and realized variance.

Forecasts generated from the 5-minute method were overall more accurate than traditional GARCH modeling forecasts, but care must be taken to deal with outliers. Models fit under the Normal distribution do not produce forecasts that are practically different from Student-t distributions. The realized variance proxy is a much more promising estimate of daily variance than the squared daily return. Looking only at the MZ regression $R^2$ values demonstrates this, where forecasts compared against daily squared returns present $R^2$ values near 0.

Future research could focus on creating models that better handled the large swings in short term volatility so that a filtration would not have to be applied. Additionally, other modeling periodicities could be tried to find an optimal forecasting granularity.

**Acknowledgments.** This work would not have been possible without the excellent rugarch R package. The paper formatting was handled completely by the pinp R package. Additionally, the tidyverse packages from RStudio were invaluable for data manipulation.

## References

Andersen, Bollerslev (1998). *Answering the skeptics: Yes, standard volatility models do provide accurate forecasts.* URL http://public.econ.duke.edu/~boller/Published_Papers/ier_98.pdf.

Bollerslev (1986). *Generalized Autoregressive Conditional Heteroskedasticity.* URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.468.2892&rep=rep1&type=pdf.

Zielak (2017). *Kaggle Data.* URL https://www.kaggle.com/mczielinski/bitcoin-historical-data.

Ñíguez TM (2008). *Predicting the monthly volatility of the EuroStoxx 50 using data sampled at different frequencies.* URL https://orff.uc3m.es/bitstream/handle/10016/11395/predicting_niguez_IFL_WP.pdf;jsessionid=129AA96FBE65A587C6B4A964D4682513?sequence=1.

**Table 3. Forecast performance of the filtered 5-minute model. Performance metrics are on par with the results from removing the extreme outlier from the non-filtered model.**

| | **5-Min Filtered** |
|---:|:---:|
| **Proxy: RV** | |
| MAPE | 0.0244 |
| RMSE | 0.0026 |
| MZ Intercept | 0.0010<br>(0.00037) |
| MZ Slope | 1.0567<br>(0.15876) |
| MZ $R^2$ | 0.2890 |
| **Proxy:** $r^2$ | |
| MAPE | 0.2666 |
| RMSE | 0.0069 |
| MZ Intercept | 0.0013<br>(0.00109) |
| MZ Slope | 0.8692<br>(0.46461) |
| MZ $R^2$ | 0.0311 |