Stochastic Volatility and Call Option Pricing on the Cryptocurrency Exchange

Evan Azevedo

This project is an analysis of the price volatility of cryptocurrencies offered through the Coinbase exchange. In order to access data from Coinbase, we install the GDAX python api. This can be done from the command line with 'pip3 install gdax'.

1 Introduction

The Heston stochastic volatiltiy model describes the process of a stock price and its variance as a set of two stochastic differential equations:

$$dS_t = \mu S_t dt + \sqrt{v_t} S_t dW_t, \tag{1}$$

$$dv_t = k(\bar{v} - v_t)dt + \sigma\sqrt{v_t} dB_t, \tag{2}$$

with $dW_t dB_t = \rho dt$ for Brownian motions W_t, B_t. This models stock price and variance at time t, S_t and v_t, respectively. The variance approaches $\bar{v}$ at speed k, so for long time scales, $v = \bar{v}$. The parameters of this model $\theta = (v, \bar{v}, \rho, k, \sigma)$ are what we optimize with on a given data set using the Levenberg-Marquardt algrithm. The objective function is to minimize the residues by least squares method with respect to the value of a vanilla call option with price at maturity S_T and strike K.
The value of the call option lies in the right to the buyer of the option to exercise the option at maturity T to purchase the strike K worth of shares, and has value $C_i(K_i, T_i) = (K - S_T)1_{(S_T \geq K)}(S_T)$. From here, the value of a call option starting today can be inferred using Heston's pricing function $C(\theta; K, T)$ with characteristic function $\phi(\theta; u, T)$ derived in their paper. A problem here arises that call options effectively don't exist on the cryptocurrency exchange. The script sample_calls.py creates synthetic call options by sampling random closing prices over the time range of the specified maturity T with value $C(S_0, S_T) = S_T - S_0$.

The entire project is ran by calling the module main.py, and it allows the user to sample more data points of a user specified amount and maturity, and then optimize for the parameters over any existing data file. After the user chooses which data files to analyze, they can specify an amount to simulate investing into the Coinbase exchange. A portfolio of their investment is split among the coins they analyzed according to the ratio of the variances of their coins, less volatile coins recieving more investment. This portfolio is then saved into the directory potfolios/ and its value can be checked at any time with the script price_check.py.

2 Results

Calibrating the stochastic volatility results in huge variances, on the order of 100 for points sampled over the lifetime of the coin. Due to the speculative nature of the value of even estabilshed cryptocurrencies such as Bitcon, Etherium, this is somewhat to be expected. Other researchers have published their work using this model on call option data in the stock exchange and report variances typically in the range of 3-5. In short, my results are not very interpretable. To this effect, the paper by Cui et al. that I used as a reference in calibrating to my data specified that this method is best done on high amounts of data points - option calls - on the current market. It is also highly dependend on the choice of initial conditions. The parameters converge well with the current choice, but altering the inital guess at the parameters has huge effect on the output due to the huge steps the calibration must take to reach a minimia. This requires an exchange of option calls that does not exist for cryptocurrencies.

Other files this project includes are: data_info.py - gives some exploratory information about the Coinbase and Binance exchange markets, start_dates.py - calculates the first date that a coin was offered on the Coinbase exchange.

It is also important to note here that in sampling historic price data, the GDAX servers will reject the user when multiple inquiries have been made over a short amount of time, and this is the reason that the program sample_calls.py sleeps in between taking data points. This script in particular should not be run with $n > 250$ data points to be taken, as has not show stability in taking that many points.