

Backtesting Market Making Strategies in Crypto Landscapes

Master Degree Project in Informatics with a
specialization in Data Science

Second Cycle 15 credits

Spring term 2025

Student: Shivam Nayyar

Supervisor: Nikolaos Kourentzes

Examiner: Alexander Karlsson

Abstract

This research evaluates two market making models—one static (Roll-based) and one adaptive (GARCH-based, using dynamic predictions of conditional volatility)—within a custom-built simulation environment using high-frequency, real-world time-series data (cryptocurrency trades and order book snapshots). The study compares their performance by backtesting their trades signaled by them on an identical historical data stream of trades of a CLANKER, which is a cryptocurrency from Bybit. The adaptive GARCH model, which dynamically adjusted its conditional volatility predictions based on a GARCH(1,1) model derived from AR(1) residuals, exhibited significantly better risk management. Whereas, the Roll model's assumed homoskedasticity and used a constant spread for the bid and ask quotes.

The results displayed a nearly identical Sharpe ratios for both strategies; however, with the GARCH model yielding a significantly lower portfolio variance and inventory risk. This was because both strategies ended up accumulating net long positions throughout the backtest. However, GARCH model's inventory accumulation was much lower than Roll model (because of its tight spread causing more trades). While the simpler Roll model had a higher interaction rate (i.e., trade count), its lack of adaptivity to real time volatility led to higher risk exposure. These findings empirically suggest that incorporating dynamic, data-driven features enhances model robustness and risk control in processing high-velocity data streams, especially for high frequency trading and market making.

Table of Contents

1.	Introduction	1
1.1.	Problem definition	3
1.2.	Motivation for Research Question	4
2.	Background.....	5
2.1.	A Brief into Cryptocurrencies and Trading.....	5
2.2.	The Evolution of Market Making.....	6
2.3.	Preliminaries	7
2.3.1.	Roll Model	7
2.3.2.	ARMA Models.....	8
2.3.3.	Volatility Clustering and ARCH Models.....	9
2.3.4.	Asymmetric Information and Informed Trading.....	10
2.3.5.	Glosten and Milgrom (1985).....	10
2.4.	Market Making: Beyond Spot Trading.....	10
2.4.1.	Options.....	11
2.4.2.	Prediction Markets	11
3.	Method	13
3.1.	Backtesting Engine: A Custom Simulation Framework for High-Frequency Data	13
3.2.	Strategy	13
3.3.	Evaluation Framework: Sharpe Ratio	15
4.	Implementation	16
4.1.	Data.....	16
4.2.	Exploratory Data Analysis.....	16
4.3.	AR(1) Model	17
4.4.	Implementation of GARCH(1,1)	19
4.5.	Implementation of Roll Model	21
4.6.	The Backtest Simulation.....	21
5.	Results	22
5.1.	GARCH(1,1) Results	22
5.2.	Roll Model Results	23
5.3.	Comparative Model Performance Analysis	23
7.	Discussion.....	25
7.1.	Theoretical framework.....	25
7.2.	Limitations	26
7.3.	Implications for Practice and Recommendations	26

7.4. Ethical and societal aspects	28
8. Conclusion	28
References	29

1. Introduction

Predicting volatility—how much prices of financial instruments are likely to fluctuate in the near future given recent activity—is fundamentally important in financial markets because it directly quantifies short-term risk to a market participant. Accurately forecasting periods of high volatility enables market participants to adjust their strategies, such as modifying their biases to manage periods of increased risk, or identifying potential opportunities arising from larger expected price movements. An understanding of volatility, more specifically conditional volatility (more flexible and is calculated as new information arises), helps traders and investors better navigate uncertainty presented by any financial market.

Cryptocurrency markets not only exhibit extremely high volatility but also trade continuously 24/7 through two-way auctions, which imply that instead of just buyers consecutively bidding up prices (in a one way auction); in a two-way auction both buyers and sellers continuously offer competitive quotes to each other and trades occur when these quotes match. Modern day trade is generally conducted in an order book, which is a real-time list of buy and sell orders for an asset, sorted by price. It shows who wants to buy (bids), who wants to sell (asks), and how much at each price—basically a live snapshot of supply and demand. It is a simple, yet fundamental method to organize bids and asks, and match traders. In such dynamic environments and fast paced environments, market makers (or liquidity makers) become crucial. Market liquidity is the ability to buy or sell an asset quickly at the current prices near market equilibrium, without causing a large price change.

A market maker's role is to maintain market liquidity by continuously placing and maintaining buy and sell orders (especially near a calculated mean or mid price, which is always changing) which traders (or liquidity takers) can trade against at any given time. The need for a market maker arises fundamentally because, at any given moment, buyers and sellers may not perfectly match in price or timing—market makers fill this gap by continuously quoting both sides and facilitating either side of trade at any moment. In their absence, markets and traders experience severe absence of liquidity, as evidenced by Foucault, Pagano, and Röell (2023). Because of this reason, it becomes essential for them to accurately predict how much prices might suddenly fluctuate (i.e. conditional volatility) and jump around so they can adjust the prices they offer for buying and selling. This helps them avoid losing money when the market becomes unpredictable and ensures they can continue to offer prices to others so as to trade smoothly.

A position is a directional bet that a market participant takes on the price of an asset. They can either a long position, where they earn money when an asset's price appreciates, or contrarily a short position, where they gain profit when an asset's price depreciates. Their basket of positions (also referred as inventories) is also called as their portfolio, whose value (and profit/loss) can be measured at discrete times during and at the end of strategy.

In context of cryptocurrencies, market makers are generally programmable algorithms which provide liquidity by consistently quoting buy and sell prices of cryptocurrencies in both futures and spot markets, across several trading venues and orderbooks. Their implicit intention is to generate a small profit from the bid-ask spread, or the difference between the prices at which they buy and sell securities. They undertake small buy and sell positions continuously (against liquidity takers) and try to maintain a neutral inventory in the long run. This means that they don't tend to have a long-term bias, or forecast, of where price will be over an extended period of time. And because of the same reason, they avoid having large one-sided positions for too long.

A central problem that the academics face is that statistical and econometric methods work in financial market work well under assumptions, and not so often on real world data. For example, as argued by Hasbrouck (2006, pp. 40–41) linear time series models often fail to fully capture the nonlinear dynamics and structural breaks commonly observed in actual market behavior. These models typically assume covariance stationarity (when a series' mean and variance is stationary but autocovariance depends on it's lag and not the time where it's measured), and homoskedasticity—assumptions that are frequently violated in high-frequency financial data. To back the claim further, Bacoyannis et al. (2018) from JP Morgan argue that the 'data modeling culture of quantitative researcher assumes that financial markets can be approximated by simple models, and often end up missing the essential properties of an environment and may end up giving a false sense of certainty. However, for a market maker, the primary challenge often lies less in achieving model fidelity to an underlying "true" value, and more in strategically utilizing model outputs—such as conditional volatility and expected value estimates—to make robust trading decisions within the observed, dynamic market environment and operational constraints. The focus thus shifts towards how effectively a strategy can leverage imperfect models to navigate actual market volatility and manage risk.

It's necessary to experiment such models on actual financial data because models with too many assumptions can become abstracted for real world application. However, as mentioned by Stoikov et. al (2024, pp. 5), it's difficult to obtain accurate historical orderbook data, especially for cryptocurrencies, as it is often gated for institutional traders due to the expensive costs for the same. This research paper attempts to bridge those two major gaps by exploring and backtesting two alternative, yet simple strategies by which market makers can determine their bid and ask prices in the cryptocurrency market. The key objective of this paper is to test the robustness of such strategies by mimicking real-life financial environments using real data and a custom built backtesting engine, in order to obtain and evaluate the performance of each.

This study examines one of the strategies which quotes prices dynamically, using market volatility, with the implementation of GARCH(1,1) model, alongside another strategy which employs a static approach to calculating spreads, founded the Roll model. As a foundational market making model which likely won't be used as a proper market maker, it is first important to understand the different experiences offered by a static spread, vs. a dynamic

spread strategy, with regards to profitability, and more importantly, minimizing directional risk by keeping the net position close to 0, over the period of the backtest. GARCH, which is formulated by predicting conditional variance based on known past shock values; as it is one of the easiest ways to introduce oneself to algorithmic market making. Similarly, the Roll model serves as an important benchmark to calculate a static spread by inferring it from the negative autocorrelation between trade prices, as explained later. The main goal from the backtests will be to find the strategy with the lowest variance in net position over the course of the backtest. Managing inventory risk is prioritized before profitability, since a larger inventory does not guarantee profitability. This is due to the stochastic nature of financial markets, which stems from the inherent randomness and unpredictability of asset price movements.

1.1. Problem definition

In spite of the advanced development of market making, the basic issue remains the same: successful liquidity provision and profitable bid-ask spread capture alongside managing the extensive risks associated with volatile, high-frequency markets such as cryptocurrency. This is academically important by empirically testing and contrasting results of market-making strategies within a custom backtesting engine.

It specifically investigates the performance differences between a static spread-setting approach (Roll model) and a dynamic, volatility-adaptive method (GARCH model). This comparative analysis contributes to the understanding of algorithmic trading effectiveness, the practical application of market microstructure theories, and provides a methodological framework for potential liquidity provision in high-frequency contexts of crypto and other financial instruments such as Options and Prediction markets.

Studying crypto market making also presents significant challenges for researchers. Data across exchanges is often fragmented, inconsistent in format, and incomplete, with many platforms and exchanges charging substantial fees for access to high-quality, granular historical data. This fragmentation makes standardization and validation difficult. Unlike traditional markets, the absence of a widely accepted benchmark dataset hinders reproducibility and cross-study comparison. Moreover, as is obvious by limited research work on the topic, most high frequency trading is done behind closed doors of algo trading companies that have no incentive to leak their source of income by publishing academic research on profitable trading strategies. It is also evidenced by CBS (2011), where all major high frequency trading firms in the US denied entry interview for their short documentary, and had to settle with interviewing a smaller high frequency trading firm.

Even though it does not tackle all the problems listed above, this empirical study is vital for building knowledge, identifying actionable steps, key performance drivers, and finding about the direction to choose for more advanced strategies in these unique and complex digital asset

environments. The present study compares two different market-making strategies for the CLANKER token on Bybit Futures, on 28th and 29th April, 2025. CLANKER is a crypto token whose trading data which was analyzed for this study and Bybit Futures is the platform on which its trading takes place. This token was chosen since it was newly listed on Bybit on the 28th and was unlikely to be contested with high-frequency market makers in the very beginning. The order book in the initial days of a smaller token listed on Bybit Futures is generally illiquid, and provides the best shot at market making for smaller players with limited infrastructure and capital. Furthermore, the market cap of the token at listing was quite low, at around \$30 million— another justification for its illiquidity.

The primary aim of this study is to empirically compare a dynamic GARCH-based spread strategy and a static Roll model-based spread strategy to determine which of the two strikes a superior trade-off between returns and risk management. To achieve this, the study will:

1. Create and custom-build a backtesting engine designed for high-frequency, order-book-level simulations and historical execution of limit orders.
2. Implement and test both the GARCH(1,1) and Roll model as market-making strategies within this engine on the CLANKER/USDT futures pair from Bybit's historical order book and trading data.
3. Evaluate their performance using Sharpe ratio (a measure to find risk-adjusted return, is properly discussed later), portfolio value variance, and other metrics including terminal portfolio value and profit and loss, total trades executed, and average inventory, with the overall goal being to evaluate risk-adjusted return and portfolio value variance as a priority, and the rest of the metrics secondarily.

This leads to our central research question: How do the GARCH and Roll market-making strategies differ in their performance with the primarily Sharpe Ratio, portfolio value variance and secondarily retrieved metrics? From a purely statistical perspective, the study first tries to predict the expected value of an asset using AR(1) and then subsequently attempts to predict its variance, in order to potentially test the predictions under real trading conditions and present its financial results.

1.2. Motivation for Research Question

The M6 Competition (Makridakis et al., 2023) gathered 200+ teams around the world to make forecasts and investing decisions with a key objective to assess risk-adjusted profitability of a diverse range of participants and discover whether or not they are able to consistently generate superior profits, relative to market returns. Importantly, the top performers were a mix of distinct teams, teams using pure time series models and others employing machine learning techniques. It verifies that fact model sophistication by itself is not the only path to achieve good investing and forecasting results—good time series methods with proper feature selection can also suffice in the same. In fact, Hypothesis 10 (p. 19-20) at the M6 competition shows that there is no evidence to prove that the better forecasting teams employed more

sophistication than the top investment teams, leading to their better predictive outcomes. This is why, this experiment relies on simple time series modeling since it can always suffice, given the right data and features in financial markets.

In Hypothesis 6 (p. 15-17) at the M6 competition, they show that higher returns are not linked with low risk. Henceforth, measuring purely profits (or returns) for our strategy is not a sound approach as they do not have less risk. In fact, it is shown that high returns can be achieved with varying levels of risk. For example, many participants with high returns, but took very high risk to achieve the same, whereas top performing participants got moderately good returns with low risk. In order to address this problem, it is essential to utilize a metric that simultaneously takes both return and risk into account and balance them. Therefore, they use Information ratio, to measure risk-adjusted returns and neither purely returns, nor purely risk. It is a variant of the Sharpe ratio (Sharpe, 1964) was one of the first methods to define a risk-adjusted return and is the primary method that will be used by our study.

2. Background

Both empirical trading practices and scholarly research have been significantly impacted by the transition of market making from traditional physical trading environments to complex algorithmic platforms. The development has made it possible to create more accurate models that explain market dynamics, particularly the complexities of information asymmetry and inventory risk management. Early models were surpassed by advanced statistical architectures that better captured the complexity and volatility of price dynamics as markets moved toward increased electronic integration and data richness. This combination of theoretical abstractions and practical application is demonstrated by the models examined in this study, including GARCH volatility modeling and Roll's spread modeling, which are crucial instruments for creating and evaluating modern market-making strategies in complex, high-frequency contexts like cryptocurrency trading.

2.1. A Brief into Cryptocurrencies and Trading

Since the introduction of Bitcoin in 2009, the financial markets and payments industry underwent phenomenal change. The individual or group, Satoshi Nakamoto, initially designed Bitcoin, the very first decentralized digital currency, in their whitepaper titled "Bitcoin: A Peer-to-Peer Electronic Cash System" (Nakamoto, 2008). The blockchain system, a public ledger that records all transactions without the need for trust or intermediaries, was presented for the very first time in this whitepaper.

In 2016, BitMEX introduced the perpetual XBTUSD leveraged swap, a groundbreaking crypto derivative allowing traders to speculate on Bitcoin's price with up to 100x leverage and no expiration date, revolutionizing the cryptocurrency trading landscape (BitMEX, 2016). This innovation, followed by similar offerings from exchanges like Binance, Bybit, and OKX,

significantly increased market liquidity and volatility, transforming cryptocurrency market dynamics by enabling high-risk, high-reward trading strategies and fostering greater speculative activity.

Cryptocurrency trading is carried out under two main modes: on-chain and centralized exchange (CEX) trading. On-chain trading is conducted directly on blockchain networks via smart contracts, which uphold transparency, decentralization, and user control of funds; however, it comes with high costs and slow transaction times, mainly due to network congestion and gas fees (transaction fees paid to network operators). Liquidity pools are usually a core mechanism used in trading on chain, where the counterparty is generally a smart contract, designed pool consisting both currencies involved in trading. On the contrary, CEX trading is conducted off-chain on exchanges such as Binance or Coinbase, where users deposit their funds into the custody of the exchange. They manage trade execution internally, facilitating quicker and often cheaper transactions, yet require users to entrust a third party with their information and funds. Unlike on-chain liquidity pools, the heart of trading on exchanges is their high frequency centralized Limit Order Books (LOBs). LOBs maintain a highly dynamic list of buy and sell orders (bids and asks) at various price levels, for price discovery and immediately matching trades between users. This form of trading supports greater liquidity, tighter bid-ask spreads, and enables advanced order types like stop orders, take-profit orders, etc. It is the preferred mechanism for professional and high-frequency traders. Most LOBs have historically existed on centralized exchanges, however, dYdX (Antonio Juliano, 2017) and HyperLiquid have changed that.

2.2. The Evolution of Market Making

Market making began on pit trading in conventional stock and commodity exchanges, such as the New York Stock Exchange (NYSE) and the Chicago Mercantile Exchange (CME). In those open-outcry systems, traders would physically gather around in circular "pit is" where they would shout out bids and offers. Market makers, who were once referred to as specialists or locals, would stand prepared to complete buy or sell orders at particular prices, profiting from the bid-ask spread while also providing ongoing liquidity to traders and investors betting on the direction of the market. This traditional approach to market making formed the foundation for modern electronic systems, where algorithms have replaced traders shouting but still perform the same role: ensuring market liquidity and efficiency.

Academic research into market making and its intricacies began to take shape in the latter half of the 20th century. A major early contribution came from Bagehot (1971). This work was crucial in identifying the fundamental challenge of asymmetric information faced by market makers, positing that they incur losses when trading with informed participants and must recoup these losses from uninformed traders, or liquidity traders.

Research on market making models began in the mid-20th century, with foundational models by Kyle (1985) and Glosten and Milgrom (1985) formalizing how market makers set prices under asymmetric information. These models showed how market makers balance inventory risk and adverse selection while providing liquidity.

Over time, literature expanded to include algorithmic and high-frequency market making, especially with the rise of electronic markets since the 2000s. The more recent studies focus on automated market makers in crypto, inventory control, and optimal quoting strategies under volatility, reflecting the evolving complexity and competitiveness of modern markets. For example, a foundational approach to optimal market making was introduced by Avellaneda and Stoikov (2008), which models inventory and price risk in a limit order book environment.

Market making now has very much become a speed and technological arms race between competing firms, with everyone trying to shave latency off by milliseconds to get an advantage. This has led to massive investments in ultra-low latency infrastructure, co-location with the big exchanges, FPGAs (Field Programmable Gate Arrays), and microwave or laser communication links. All of this is aimed at executing trades microseconds, and even nanoseconds faster than competitors (Nahar et al., 2024).

One of the very recent studies by Stoikov et al. (2024) backtested and implemented crypto market making strategies using 1 minute candlestick data (discrete time series consisting of open, high, low and close of prices). However, candlestick data is hardly granular enough to conduct high frequency market making in fast paced markets like crypto, and is often too slow to appropriately compete with professional market makers which compete aggressively and get the spread down to the minimum possible. As discussed by Hasbrouck (2006) throughout his book, the importance of utilizing live trading and order book data is crucial for being successful in market making environments. Therefore, even though not directly used to retrieve signals from historical trading data and granular 100 ms order book snapshots are indeed utilized in the backtesting engine to at least get a closer simulation to a real market making environment.

2.3. Preliminaries

2.3.1. Roll Model

In the past before the electronic trading regime, quoted spreads were difficult to obtain and researchers before 1990s usually only had access to transaction prices of each market. Therefore, there was an implicit need to calculate transaction costs and since brokerage costs were variable, there was a need to model the spread. Roll (1984) came up with a simple model to estimate the spread as 2 times the square root of first-order autocovariance on the entire price series, since the prices expected to have a negative serial autocorrelation because the trade prices randomly simply either hit a bid, or an ask price. Although mostly unused now because it is too simplified and unrealistic for practical market making purposes as signed orderflow and orderbook data is easily accessible in crypto directly from crypto

exchanges; it served as an important starting point in market microstructure analysis because it linked market costs directly to visible price negative serial correlation. Therefore, providing the first generally used technique to estimate the unknown bid-ask spread using just transaction prices. The basic Roll model is used in the research as the base model to retrieve a fixed spread, to model the for backtesting our market making sample. The classic Roll formula used to calculate half spread is the following:

$$C = \sqrt{-\gamma_1}$$

Where, γ_1 is the first-order auto covariance of trade prices, and C represents the distance from the mid-price to either the bid or the ask. The negative sign is assigned as the bid-ask bounce makes the γ_1 negative. Since the bid and ask are equidistant from the mid-price, the full spread is $2C$. The square root of the negative auto covariance is taken to convert the value back to price unit is.

2.3.2. ARMA Models

AR models are a class of time series models that predict future values based on a linear combination of past values. They assume that past price movements contain enough similarity to be informative about future values, which aligns well with the microstructure of financial markets where short-term dependencies often do exist. MA (Moving Average) models, on the other hand, model future values as a function of past forecast errors, making them well-suited for capturing short-lived shocks or noise in price movements.

As implemented in this research, this makes AR and MA processes (Box and Jenkins, 2015) a natural starting point for modeling price dynamics, since they directly capture the first-order serial dependencies introduced by the bid-ask bounce phenomenon. This phenomenon causes prices to often alternate due to trade direction, introducing negative autocorrelation that these models can effectively detect. While not sophisticated enough for full-scale market making strategies, they offer a useful baseline for understanding short-term price behavior and initiating research.

Using the serial dependencies, ARMA models the future conditional expectation of the time series (in our case, the forecasted mid price — introduced in Section 3.2) and assume a constant variance structure. However, since financial markets time series are heteroskedastic in nature, and constant variance usually never holds. This is why, as introduced in the next section, ARCH models are used to model the leftover time-varying conditional variance from the ARMA model residuals, given that those residuals don't have any autocorrelation left and are effectively white noise.

2.3.3. Volatility Clustering and ARCH Models

Volatility in time series refers to the size of data fluctuations; whereas, volatility clustering describes the common pattern where large changes tend to follow large changes, and small changes follow small changes. For data science, this means the time series' variance is not constant and requires specialized approaches to capture these dynamic shifts in variability.

Volatility clustering is one of the bigger empirical characteristics of financial time series, in which periods of high price volatility are followed by periods of high volatility and, analogously, for quiet periods to follow quiet periods (Mandelbrot, 1963). Such persistence in volatility first, discovered more than half a century ago, is also loosely described as "volatility begets volatility." It is one such fact that is not explained by traditional models that expect constant variance. To address this, Engle (1982) suggested the Autoregressive Conditional Heteroskedasticity (ARCH) model, which explicitly models time-varying conditional variance, extended by Bollerslev (1986) to the GARCH(p,q) model, which is the main model used to backtest our market making strategy (with p=1 and q=1). Volatility clustering in general, is extremely important to understand for market makers and other market participants because it has immediate implications for risk measurement, derivatives pricing, and the dynamic updating of trading strategies in order to respond to ever changing market conditions.

The GARCH model, particularly the GARCH(1,1) specification, is often favored over a pure ARCH(q) model because it can capture long memory and persistence in volatility more parsimoniously. By incorporating a lagged conditional variance term ($\beta \cdot \sigma_{t-1}^2$), GARCH(1,1) can often represent the volatility dynamics with fewer parameters than would be required by a high-order ARCH model that relies solely on many lags of past squared shocks. The following is the notation for calculating GARCH(1,1) conditional variance (as used in the strategy):

$$\sigma_t^2 = \omega + \alpha \cdot \varepsilon_{t-1}^2 + \beta \cdot \sigma_{t-1}^2$$

Where, σ_t^2 is the conditional variance, ω is a constant (long-run variance), α captures the impact of recent shocks, β reflects the persistence of past variance and ε_{t-1} is the model error at time $t - 1$.

Contextually, GARCH(1,1) functions very similar to an ARMA(1,1) because both use an autoregressive component (based on the previous period's value of what's being modeled – predicted variance for GARCH, the series level for ARMA) and a moving average component (based on the previous period's shock/error term, which is squared residual for GARCH, and residual for ARMA) to forecast the next period's value. The only difference is in what they're trying to predict — variance for GARCH and expected value for ARMA.

2.3.4. Asymmetric Information and Informed Trading

A market consists of both informed and uninformed traders, as discussed by O'Hara (1998). Informed traders, or insiders, have asymmetric (superior) information that are predictive of future returns. This leads uninformed traders (people with no superior information) to carry higher risk in their positions as they do not possess information with much predictive power. Market makers, who are always at risk of dealing with informed participants with asymmetric information, must use a positive bid-ask even without a profit expectation to recover losses from informed trades (Copeland and Galai, 1983). This positive bid-ask spread, which is supposed to be higher than normal when informed trading is suspected, enables them to retrieve a higher margins to recover the losses faced by informed trades.

2.3.5. Glosten and Milgrom (1985)

The Glosten-Milgrom (1985) model is a foundational work in market microstructure that explains how bid-ask spreads emerge due to asymmetric information between informed and uninformed traders. In the model, a single dealer sets prices while facing market buys or sells orders from traders at random, some of whom may have private information about the asset's true value (V_{low} or V_{high}). The asset has a binary terminal payoff (e.g., high or low value), and the dealer updates beliefs based on the sequence of trades. This setup shows how order flow can reveal critical information, leading the dealer to widen spreads to protect against informed trading, if any exists, modeled by parameter μ . The model was a breakthrough in showing that prices can reflect private information through Bayesian learning.

However, it is of limited practical use today in financial markets as it assumes one single dealer, does not account for competition, and simplifies actual markets, which trade through continuous auctions, specifically crypto and stocks, where there is no terminal value. Unlike the model's binary payoff structure, such markets feature continuous, open-ended trading with no definite final result, which makes the model's assumptions less relevant to contemporary trading environments.

2.4. Market Making: Beyond Spot Trading

Market making is not just restricted to the trading of simple financial securities and cryptocurrencies. It's principles are applicable to a wide range of financial products, from simple to more complex. This section discusses two of the further applications of market making: Options and Prediction Markets trading. Their market making is very different and has more nuances from what this research paper has covered so far because they can require distinct methodologies for estimating an asset's expected value or fair price.

2.4.1. Options

Market making is not only highly prevalent in spot and futures market, it is also widespread in the options market. As described by Hull (2010), a call option gives the buyer, the right (but not the obligation) to buy an asset at the price on expiry; and a put option gives the buyer the right to sell an asset at the strike price on expiry, from the respective option writers. The core concept may sound simple, however, under most speculative use cases (such as volatility betting), they are extensively traded and change hands between traders and are not just meant to be used for hedging, or holding on till expiry. And naturally, this arises the need for high-frequency market makers in options which in turn demands for strong pricing models to calculate real-time fair value for any given options. The extremely popular Black-Scholes model (Black and Scholes, 1973) is the most well known for the same, which provides a simple framework to calculate a fair price for an options contract which builds as important groundwork for more complex options pricing models. A key assumption of their model is constant volatility, which is never true in real markets. This is why our volatility forecast serves as a very important input to retrieve the appropriate options price. And on the other hand for a trader, implied volatility (IV) and respective greeks, which are calculated using market prices, help to understand the risk profile of an option.

Crypto options trading began in 2016, with the launch of Deribit. Although not as popular as perpetual futures, they are a growing segment in the crypto trading landscape with extensive growth in volume over the years (**CoinGlass, n.d.**). However, due to the added complexity of options trading relative to the usual spot and futures trading, they have been mentioned for informational purposes and their market making will be out of scope in this research.

2.4.2. Prediction Markets

Similar to conventional financial markets, prediction markets are operated on a counterparty basis, where the participants wager against one another as to the outcome of the future event. Prediction markets primarily use a limit order book (LOB) protocol for trade execution. An essential distinction with prediction markets is their event-contingent closure. When the eventual resolution of the event arrives, trading on the market ceases and all contracts are closed at a binary price, typically 0 for 'No' events and 100 for 'Yes' events. This settlement process ensures that it is a zero-sum game, where one party's winnings equal the losses of the other. Contract prices are thus constrained between 0 and 100 and are probabilistic indicators. The prevailing market-implied probability is generally the midpoint of the best offer and bid prices in the LOB, especially in one of the most famous protocols, Polymarket. Market efficiency, and the correctness of this implied probability, or the midpoint, is assumed to be derived from the behavior of informed traders. These agents, who are motivated by potential arbitrage gains, trade actively to realign perceived mispricings and therefore participate in the price discovery process. With sufficient liquidity and an adequate number of sophisticated traders, market prices will tend to move towards a correct representation of publicly available information.

Market makers (MMs) play a crucial role in prediction markets as they supply continuous liquidity and thus enable traders to sell or buy contracts at any time without incurring significant slippage, particularly in less liquid markets. By simultaneously placing buy and sell orders close to the prevailing price, MMs narrow the bid-ask spread as in spot markets, which enhances the price efficiency and makes them better reflect true probabilities. Their presence reduces the threat of illiquidity, allows for smoother price discovery, and helps to keep the market stable, especially as event resolution approaches. Without market makers, prediction markets can suffer from wide spreads and weak price signals that can ruin their function to provide good odds for future events. Glosten and Milgrom (1985), which could not be used in our research, resembles much closer to market make prediction markets since they do have binary terminal payoffs, which spot and futures do not.

3. Method

3.1. Backtesting Engine: A Custom Simulation Framework for High-Frequency Data

The ground backtesting engine had to meet very specific requirements like controlling for latency and permitting tick by tick trading data, which was difficult to find in existing backtesting libraries that were open source. Easier alternatives, e.g. the usage of Pine Script® by TradingView might have been possible due to the availability of second-based charting, easy strategy backtesting and their easy to write programming language. But it was not chosen due to the lack of availability of a historical order book and very expensive trade-by-trade data. However, it must be noted still possible to create and backtest very rough, limited, and simple market making strategies in TradingView if appropriate paid subscriptions have been paid for.

For these reasons, the backtesting engine was created from scratch in python. It was able to support orderflow data, order book, and user-defined limit orders. There are 3 main components of the engine: Broker, Trade Simulator, and the Core Backtesting For Loop, explained as follows:

1. Broker: The Broker component manages the simulated trading account, tracking cash, positions, and open limit orders, while also applying transaction fees for executed trades. It is also the class where the Strategy resides.
2. Trade Simulator: This component is responsible for matching and filling incoming historical market trades against the Broker's existing limit orders on price-time priority.
3. Core Backtesting Loop: This loop iterates through historical trades. For each trade, it first updates the order book to the state it was in a defined latency period before that trade occurred, allowing us to account for latency (though none was specified, since we used the rather slow 100 ms order book updates anyways). These decisions are then processed by the Broker and Trade Simulator, and portfolio metrics are recorded.

All these components (or classes) respectively work together to provide a smooth and efficient backtesting Python framework tailored to market making.

3.2. Strategy

The backtesting strategy is done at 100 millisecond intervals because that's the granularity of the orderbooks updates in the data. Return calculations in the supporting code are based on log returns (due to their symmetric nature), which are quite close to simple returns as denoted throughout the paper to make it easier to read and understand the mathematical notation.

Under the strategy, M_t is the true value of an asset at time t . It is defined as the following, which is a common proxy to define it:

$$M_t = \frac{Best\ Ask_t + Best\ Bid_t}{2}$$

The Best Ask is the lowest price at which a seller is willing to sell, and conversely, a Best Bid is the highest price at which a buyer is willing to buy; where Best Ask > Best Bid.

In our strategy, the central reference for quoting is a 'Predicted Mid', denoted M_{t+1} . This is derived by first forecasting the next period's mid-price return, let's call it r_{t+1} , using an Autoregressive (AR) model applied to recent historical mid-price returns. The mid-price return at time t is defined as:

$$r_t = M_t - M_{t-1}$$

Our AR(1) model then forecasts the next return as:

$$r_{t+1} = \alpha + \beta \cdot r_t$$

Where α is assumed to be zero as the long-term mean of these short-term returns is expected to be negligible, and β is the AR(1) coefficient. The 'Predicted Mid' for time $t + 1$ is then calculated by adding this predicted return to the current mid-price M_t :

$$M_{t+1} = M_t + r_{t+1}$$

Our bid-ask quotations are set symmetrically around this M_{t+1} with the variable C , representing the half-spread:

$$\begin{aligned} Ask_t &= M_{t+1} + C_t \\ Bid_t &= M_{t+1} - C_t \end{aligned}$$

The single bid order and single ask order will be set and updated every 100 ms, matching the orderbook interval, with each with quantity of one, which has been chosen arbitrarily to retain simplicity. C will be constant under the base Roll model, which is set by the square root of the negative first-order autocovariance of trade prices. Statistically, it will perform best under the assumption of homoskedasticity, which is never true under real financial markets— also the reason why it was chosen as the benchmark model due to its simplicity.

The GARCH(1,1) model on the other hand uses the predicted standard deviation (or conditional volatility) per time step t , to set half spread C based on the past shock value and past model prediction, as it assumes symmetric variance prediction. This will perform best under the assumption of heteroskedasticity, especially significant volatility clustering takes place.

The backtesting for loop then runs through every past trade. For each trade, it first synchronizes the order book by applying all orderbook updates that occurred up to the timestamp of that trade. This ensures the strategy operates and sends limit orders on the correct market state. Trade Simulator then conducts a virtual simulation, checking if the incoming market orders crosses any of the strategy's resting limit orders. If a match occurs, it records these as fills and determines the updated orders with any remaining quantities. Separately at the end of each trade, the portfolio history list logs key performance and state metrics (like portfolio value, position, and quoted spread) at each trade's timestamp, providing a snapshot of the strategy's status just before the current market trade is processed against its orders.

3.3. Evaluation Framework: Sharpe Ratio

Developed by Sharpe (1964), the Sharpe ratio is one of the most popular financial ratios to capture risk adjusted returns of a portfolio, which means how much return a portfolio (a basket of a single or several positions for each strategy) generates relative to the risk taken. For example, if two trading strategies generate equal profits on average per trade but the variance of those results is much greater in one of the strategies, it implies that it undertook significantly greater financial risk to achieve those results. A high profit is desirable for every trader (or market maker), however, the risk taken to achieve those results is what makes one better than the other.

The Sharpe ratio is defined as the following:

$$S = \frac{R_p - R_f}{\sigma_p}$$

Where, R_p is the expected (or mean) portfolio return, R_f is the risk free-rate which is generally considered as the rate of return on government bonds (as they have close to 0 risk of loss or default) or other risk-free securities, and σ_p is the standard deviation of the portfolio returns (risk). For our paper, R_f is assumed to be 0 due to the unavailability of a consensus risk-free rate in cryptocurrencies and its very negligible role in our highly short-term backtesting. Strategies with higher Sharpe ratios are considered better, because they deliver higher returns per unit of risk and reflect a more optimal balance between risk and reward.

The second most important evaluation metric for our research is the variance of the portfolio value across backtest, is perfectly correlated with σ_p and is picked out and viewed at separately to measure the risk taken by a portfolio.

4. Implementation

4.1. Data

Bybit provides historical trades data for all their spot trading and contract pairs. For this study, data for the CLANKER/USDT pair from Bybit was utilized, specifically covering the period April 28-29, 2025. We were able to retrieve top 500 levels of both bids and asks, which included an initial snapshot followed by orderbook updates typically every 100 ms (given that a new order was pushed to the order book). Historical trades were retrieved down to the millisecond, timestamps, initially provided in a format (seconds with decimal precision), were converted to milliseconds (by multiplying by 1000) and then rounded to one decimal place for processing; each trade record included timestamp, price, size, and tick direction. In the backtesting environment, these order book updates were processed sequentially to reconstruct the book state immediately preceding each historical trade. The granular trade and order book data are directly used to run the backtest and the market-making engine, ensuring a testing environment that is very close, if not perfect, to the sequence in which trades took place.

4.2. Exploratory Data Analysis

For 28th and 29th April, 2025, 29435 total discrete trades on CLANKER took place with a non-zero arrival time and 62% of the successive trades happening above 100 ms. It is a good crypto token and time to attempt market making on, since this implies limited high frequency trading. This means that we will miss about 38% of the trades and the consequent changes in the mids, but it is a reasonable trade off as it is a better position to compete than other tokens with high volume and high market making competition, with the median order arrival time being 0.22 seconds (as in Figure 4.2.1), much higher than our 100 ms observation span.

After resampling and forward-filling missing values for all 100 ms intervals, approximately 1,339,906 rows were obtained. In 97% of cases, the midpoint did not change consecutively due to the token's low trading volume and frequency. Therefore, our main challenge is to make appropriate market-making decisions during the 3% of the time when the mid does shift unexpectedly. With a 100 ms observation/reaction cycle, the strategy cannot react to multiple trades or mid-price changes occurring within the same 100 ms window—it only observes the market at discrete 100 ms intervals. Since we also lack data with higher time resolution, any mid-price changes within <0.1 seconds are considered an acceptable information loss.

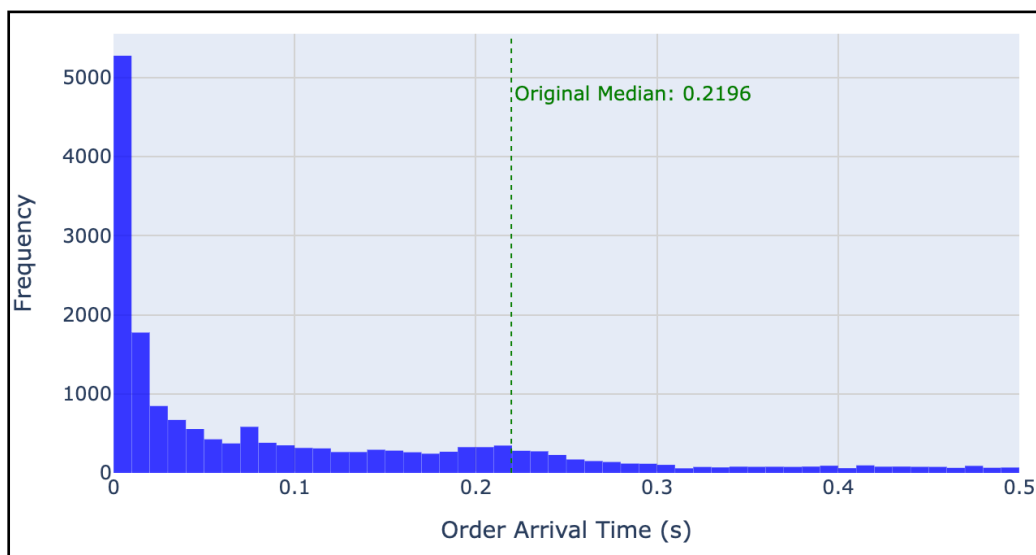


Figure 4.2.1: Order Arrival Histogram CLANKER for 28th and 29th April, 2025

4.3. AR(1) Model

An important aspect of the analysis concerns the autocorrelation structure of returns. It was found that only the first-order autocorrelation of the 100 ms mid returns was rather insignificant and small, at -0.025 (see Figure 4.3.2), which makes a lot of sense since 97% of the time the mid stays constant, causing the mid return to be 0. However, it just happens to be that CLANKER is not popular (due to low volume traded) and does not have fast successive trades, with a average order arrival time of 4.5 second. Popular tokens trade much more than that and for many, predicting on a 100 ms interval will be ideal for them.

Even though we failed to find any significant autocorrelation on 100 ms mid returns, the most logical starting point to model returns is usually the AR(1) model. It works especially well with trade prices due to the bid-ask bounce described by Roll (1984), especially when trades would occur on average of a 100 ms gap (better than the current average order arrival time), one would expect statistically significant negative serial autocorrelation. Assuming that trades take place successively on an average of 100 ms on future pairs, Hasbrouck (2006, p. 30) verifies that AR(p) beyond $p=1$ of trade price returns would be zero. This is because first-order autocorrelation represents bid-ask bounce effect, but beyond that persistence in autocorrelation implies predictability, which simply cannot sustain as market participants would quickly exploit it, and the predictability will vanish. This falls phenomenon falls under the weak form of Efficient Market Hypothesis, as first described by Fama (1970).

The regression coefficient of AR(1) was statistically significant, and the results are provided below. The ARMA mixed model was not used because the residual had little to no autocorrelation remaining and further lags was not chosen as the coefficients were close to

zero. Finally, as mentioned prior, the AR(1) model's residuals will be used in our main volatility model, GARCH.

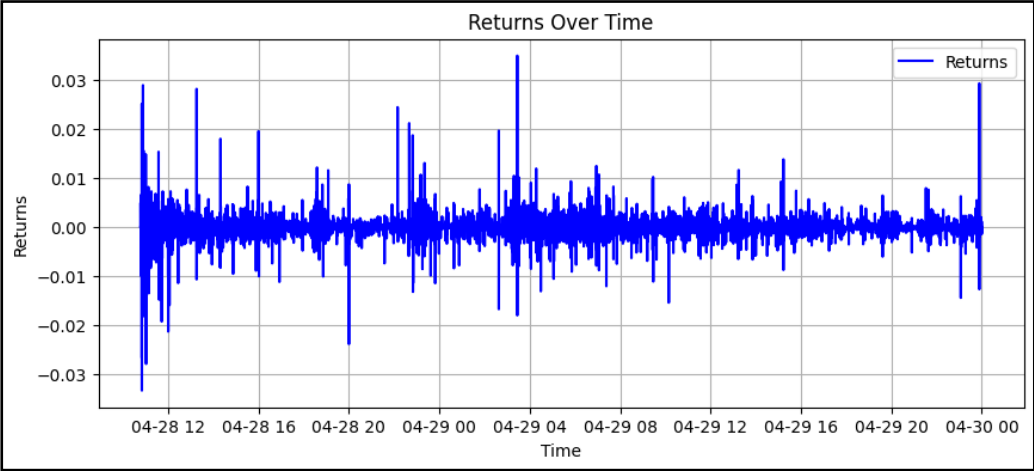


Figure 4.3.1: CLANKER Returns

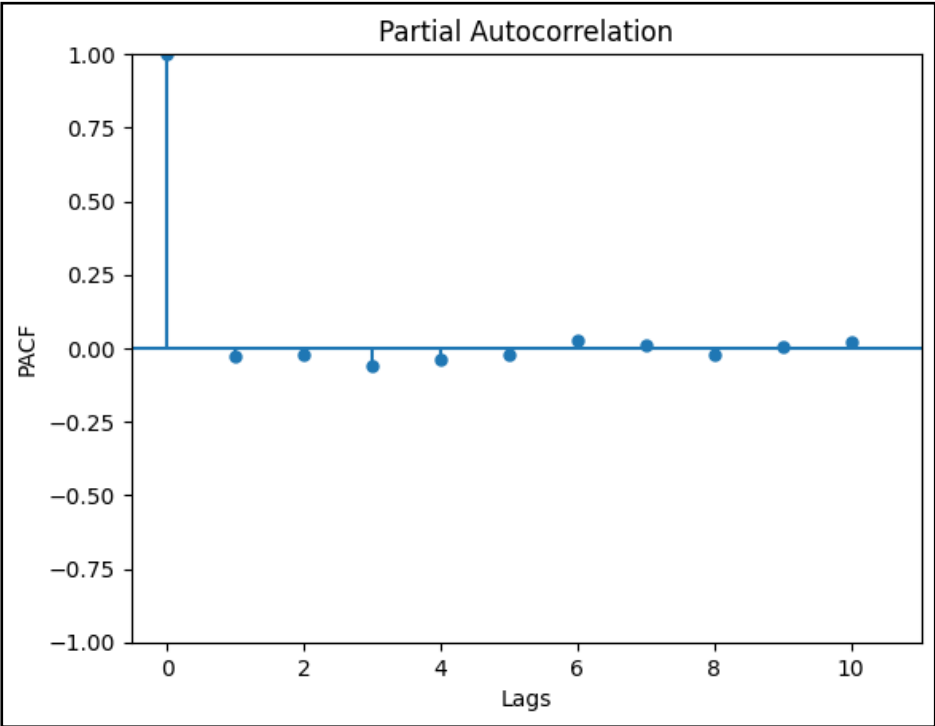


Figure 4.3.2: CLANKER Mid Returns PACF (10 lags)

4.4. Implementation of GARCH(1,1)

GARCH(1,1) by Bollerslev (1986), is used in this research because it is a widely accepted baseline for modeling financial volatility. The GARCH(1,1) model can capture the salient features of volatility dynamics (such as clustering and persistence) with a minimal number of parameters. Hansen and Lunde (2005) provided evidence that ARCH-based models performed no better than GARCH(1,1) to predict conditional volatility of exchange rate data, therefore, it justifies the usage of GARCH in our context.

ARCH was avoided because it would have required too many lags to capture the volatility clustering, and it's inefficiency in capturing the persistence of financial volatility. This inefficiency itself is the reason why Bollerslev, T. (1986) created GARCH to begin with, because by introducing a lagged conditional variance term directly into the variance equation, the GARCH model provides a far more reasonable and flexible way to model the long-term memory of volatility without needing an impractically large number of ARCH terms, which often complicates estimation and risks producing invalid negative variance forecasts.

We applied GARCH to the entire AR(1) residuals of CLANKER token. The model fit very well, with β having a very high coefficient 0.93 (p-value <0.001), implying a very strong connection with the last prediction, and as discussed earlier, 97% of the time, the mid-return is 0. The alpha coefficient also stands at a reasonable 0.05 (p-value <0.001) which means that a shock on ϵ_{t-1}^2 contributes 5% of the predicted conditional volatility at time t. To confirm the direct correlation between absolute mid return changes and volatility predictions, we then visually graph both variables and find their correlation coefficient. For this mini-analysis, absolute mid return changes are chosen instead of usual mid returns since they capture the magnitude and ignore the direction of the return. It is done to ensure whether or not GARCH model's predictions of volatility are actually aligned with the realized historical volatility. We find the autocorrelation of both series to be 0.3, which is strong enough to confirm their obvious relationship. This implies that the spread in the strategy will be quoted appropriately with respect to volatility predictions. The side by side graphs underneath, the same can be seen and visually verified (Figure 4.4.1 and Figure 4.4.2).

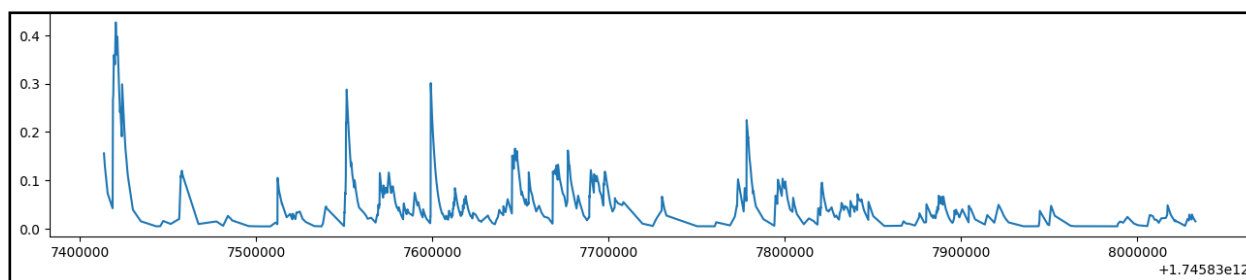


Figure 4.4.1: Volatility Predictions over first 2000 trades

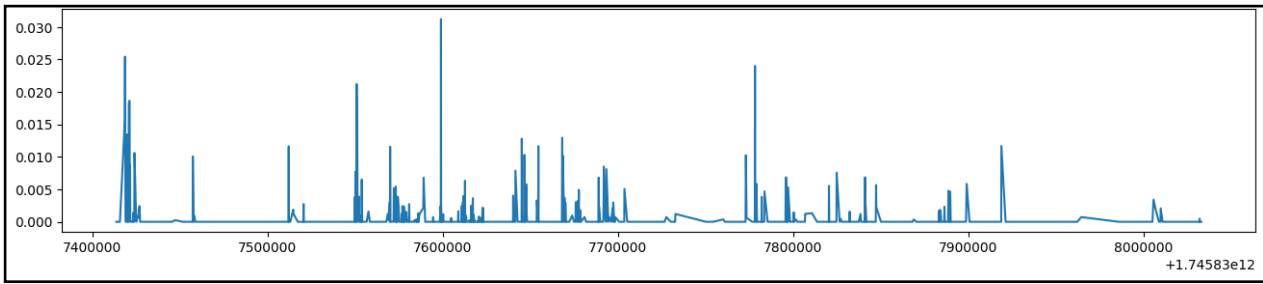


Figure 4.4.2: Absolute mid returns over first 2000 trades

Also, it can be noticed from the absolute mid returns graph above that they are very sporadic in nature and generally appear in clusters, as discussed by Cont, R. (2001). This volatility clustering is what the GARCH model is meant to take advantage of the fact that little to no volatility is followed by less volatility, and high volatility is followed by higher volatility as well.

It is also observed that after a spike, the volatility prediction decays overtime after a spike. This happens because the 0.85 beta coefficient keeps dampening the predicted level over time when realized volatility (or absolute mid returns) dries down, which aligns with the mean reverting properties of a well-estimated ARMA (Box and Jenkins, 2015), and it's close relative GARCH. So, it is very clear that the spread under GARCH increases when a large volatility move takes place and slowly decreases as no further absolute returns come in. Looking at Figure 4.4.3, many timestamps are missing since in those periods, either nothing was quoted since C fell below minimum quoting threshold of 0.038, or no trading took place in the period for the spread to be recorded. The bid and ask history of GARCH strategy for the first 2000 trades in the backtest, where a bid at a certain time is represented as a green dot, and an ask is represented as a red time.



Figure 4.4.3: GARCH(1,1) Strategy Bid and Ask history Over First 2000 Trades

4.5. Implementation of Roll Model

The Roll model takes advantage of the negative serial correlation in trade price change seen, the "bid-ask bounce," to measure the effective spread as enunciated earlier. The first-order autocorrelation of price change for the CLANKER/USDT data was -0.22. Applying Roll's formula, where C is the negative square root of the first-order auto covariance, to the dataset yielded a constant half-spread C of 0.038. The above value was then used consistently by the underlying market-making strategy to determine its bid and ask quotes symmetrically around the expected mid-price. The fixed spread approach provides a baseline to assess the dynamic GARCH-based strategy.

4.6. The Backtest Simulation

To evaluate the effectiveness of the developed algorithmic models (e.g., the GARCH-based and Roll-based), a simulation-based evaluation framework was employed. This framework functions as a data-driven environment, designed to rigorously assess how each model would have performed when applied to a specific sequence of historical trades. It is to be noted that historical trades are different from the trades executed by the market maker in the backtest (which are only considered as executed when historical trade prices match with bid and ask prices of our market making strategy)—the historical trades are the real trades that took place during 28th and 29th April, 2025 for the CLANKER token.

The methodology involves processing time-ordered historical trading data, which is the historical trades, event by event. At each step (for each historical trade), the midpoint is retrieved from the snapshot of the orderbook right before the historical trade took place, and the M_{t+1} is calculated, as prescribed in Section 3.2.

Afterwards, in case of GARCH strategy, the specific conditional volatility predictions derived from GARCH conditional volatility prediction. If the half spread C prediction exceeds the minimum of 0.038, and then proceeds to set Ask_t and Bid_t symmetrically around the M_{t+1} (as explained in Section 3.2). If the minimum threshold does not exceed, neither Bid_t nor Ask_t are set. This was done to ensure that the quotes don't get too close to each other, since that would lead to orders with an unrealistically low spread.

Roll Strategy also calculates M_{t+1} in the same way; however, no predicted conditional volatility is required since the Roll model always sets the Bid_t and Ask_t with $C = 0.038$. Do recall that t are discrete times with an interval of 100 ms.

The consequences of these buy and sell actions are then simulated based on predefined interaction rules within the historical data environment, which asks the following question: Given that a historical trade took place, if the strategy's bid and ask orders are present in the simulated orderbook at time t , would they match the real historical trade? This design

decision is meant to simulate what would have happened in case my orders in the specific sequence were actually present at time t on 28th and 29th, April, 2025.

If a buy order is matched with the historical trade, then the position (or inventory) increases as +1. Similarly if a sell order is fulfilled, then position decreases as -1. Throughout all historical trades in the backtest, the cumulative total of the position and cash is updated at each matched trade. When a buy occurs, cash decreases by the trade price (since we pay cash for buying 1 quantity). When a sell occurs, cash increases by the trade price (since we receive cash for selling 1 quantity). This simulates realistic capital flow, enabling accurate tracking of Portfolio Value for at every step of the backtest (i.e. each historical trade) as:

$$\text{Portfolio Value} = \text{cash} + (\text{position} \cdot \text{trade price})$$

After getting the portfolio value, it is easy to retrieve the returns for each step of the backtest, by just differencing the portfolio value with the previous step's portfolio value. And finally, at the end of this simulation, these recordings of portfolio value and profit/loss per step of the backtest allows us to calculate our key performance indicators (KPIs) of Sharpe Ratio and Portfolio Value alongside the other variables such the mean of CLANKER position/inventory for each strategy.

Both GARCH and Roll Strategies start with an initial cash of 10,000 with no assumption of a trading fees. And as explained earlier, in both versions, both the bids and asks will be set at 100 ms intervals at a singular quantity, 1 for bid and ask each, and the backtesting engine checks whether or not a match took place with historical data. There is no need to include further latency in the backtesting code because 100 ms is already suitable for the purposes of market making.

5. Results

5.1. GARCH(1,1) Results

The simulation started with an initial capital of \$10,000. The strategy finished with a final portfolio value of \$9,782.36, which is a net loss of \$217.64. The final portfolio had a cash balance of \$6,096.12 and a holding of approximately 90.04 units of the asset. Over the whole backtesting duration, the GARCH strategy placed a total of 4,575 trades. A complete record of portfolio value, inventory, as well as the quoted spreads at every incidence of trading, was thoroughly recorded from over 120,000 historical trades. The average inventory throughout the backtest turned out to be 67.

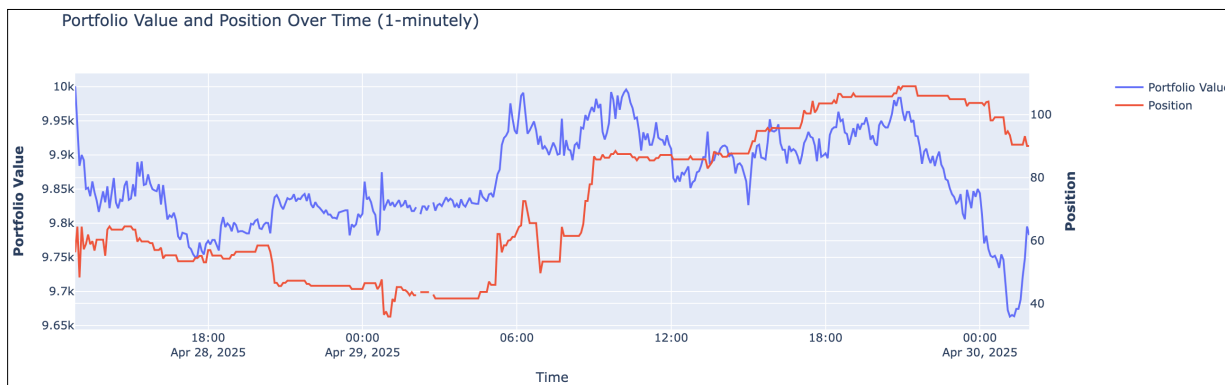


Figure 5.1.1: GARCH(1,1) Strategy Portfolio Value and Position over Sample Period

5.2. Roll Model Results

Starting with capital of \$10,000, the simulation finished at terminal portfolio value of \$9,134.40, corresponding to a net loss of \$865.60 (3 times higher than GARCH). The final inventory of the strategy was at approximately 461.37 unit is of the asset and the average was 252. During the backtesting period, the Roll strategy executed a far greater number of trades at 32,465 trades.

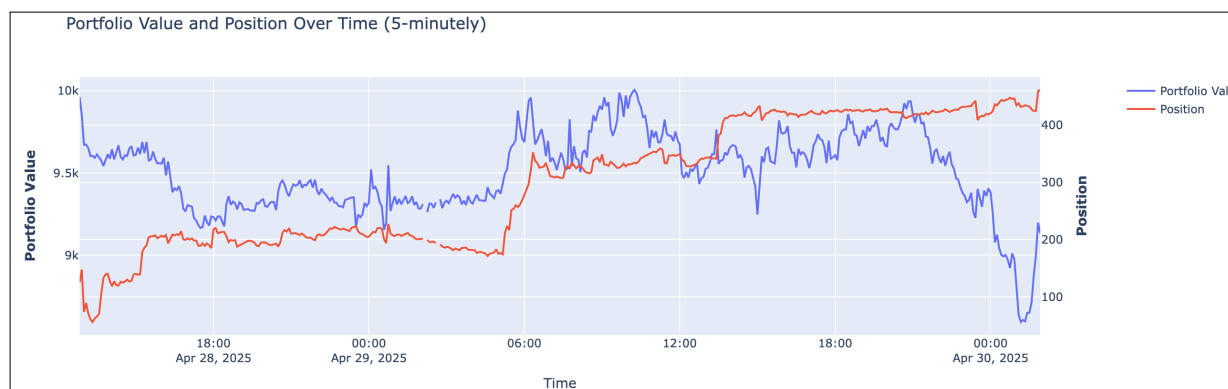


Figure 5.1.2: Roll Strategy Portfolio Value and Position over Sample Period

5.3. Comparative Model Performance Analysis

The total executed trades is about 7 times higher for Roll, and that is reasonable since it was always quoting the same tight spread all the time, causing it to match a lot more frequently with the historical trades. The portfolio values of both strategies look highly correlated from the graphs (Figure 5.1.1 and 5.1.2). However, the variance on the portfolio value of the Roll Strategy, which is 62513, is 13 times higher than that of GARCH strategy, 4765. This is because the Roll strategy had a very strong rise in position size over time and therefore,

caused the portfolio value to be more reactive to price changes throughout the backtest, as inventory (or position) is a huge factor in the calculation of portfolio value, as seen in the Portfolio Value formula in Section 4.6.

A very unexpected and surprising result of the backtest was that the Sharpe ratio of both strategies was exactly the same up to 5 decimals, as seen in Table 1. This is generally not supposed to happen since different strategies are supposed to provide different risk-adjusted returns, however, in our backtest and conditions, that wasn't the case. Despite utilizing distinct methodologies for determining their operational parameters (static spread for the Roll model, dynamic spread for the GARCH model), both algorithms exhibited nearly the same risk-adjusted performance.

This outcome is primarily attributable to the fact that both models, predominantly acquired and held long/buy positions in the same underlying asset (CLANKER) during the identical evaluation period. This is visible in the Figure 5.1.1 and 5.1.2 as we can easily see that positions for both strategies stayed highly positive the entire time. Also, as discussed earlier, both portfolio values are also highly correlated. This caused the mean returns and standard deviation of returns be proportionally similar for both strategies, as seen in Table 1 below. As such, their cumulative financial results were largely governed by the price series trajectory of this common asset, and caused both strategies to have very similar risk-return profile (i.e. standard deviation-mean return) across the backtest. While the models differed in their interaction frequency (order matching) and internal state management (inventory), these operational distinctions were less influential on the final risk-adjusted metrics than the shared position to the price changes of the single asset under evaluation.

Despite similar risk-adjusted return, both strategies can still be differentiated using their average position (or inventory) across the backtest. Going to the basics, the main problem with market making is the inventory accumulation — as more one-sided inventory that is accumulated (like long/buy positions in our backtest), the greater is the risk of the strategy in question (because of the risk of a sudden fall in asset price). This is explained by the average inventory, which is at 62 for GARCH and 252 for Roll. At about 4 times greater, it explains how much greater the inventory risk was, taken by Roll. Beyond that, whereas Roll quoted the entire time, GARCH only quoted about 40-50% of the time since spread went below min threshold quite a lot and often for long periods. As price of CLANKER went down by the end of the day, both strategies lost money; but GARCH strategy's loss was more reasonable, whereas Roll's loss was about 3 times higher. This is important because, hypothetically, if the asset were to rapidly drop in price if the backtest continued, the Roll Strategy would face the highest risk and subsequent loss.

Table 1: Performance Metrics of Both Models

	Roll Model	GARCH Model
Sharpe Ratio	-0.000563	-0.000562
Mean Returns	-0.0072	-0.0018
Standard Deviation of Returns	12.78	3.21
Portfolio Value Variance	62513	4765
Total Trades	32465	4575
Final Portfolio Value	9134	9782
Final Profit and Loss	-866	-218
Average Inventory	252	68

7. Discussion

7.1. Theoretical framework

It is difficult to find papers that had backtested market making strategies on real data since they are generally done by proprietary firms for a profit motive and do not have the incentive to disclose neither the data, nor the strategy or the results. Theoretical frameworks, such as Chakraborty and Kearns (2011), indicate that model performance (profitability) is negatively impacted by stale inventory (continuous long positions in CLANKER throughout the backtest) and strong trends in price. This means that holding inventory in the same direction as large price changes take place (as in our case, where price went down), can have a negative impact on profitability. This is because predicting prices is a very difficult task, as shown by Makridakis et al. (2023), where a majority of the contestants in the forecasting competition showed overconfidence in their forecasted quantiles of prices and underestimated the true randomness of price movements in their respective stocks. Our empirical evaluation of the Roll model, which accumulated significant inventory during a data trend, showed higher portfolio value variance (i.e, higher risk of being wrong), aligning with this framework.

Simulation studies like Xiong et al. (2015) suggest that models incorporating dynamic features derived from real-time data (such as volatility or order flow characteristics) achieve better performance. Our GARCH model, which utilized a dynamic volatility as a feature, yielded superior risk metrics (i.e. portfolio value variance and lower inventory accumulation) compared to the static Roll model. This supports the value of adaptive models and suggests that incorporating additional dynamic features, as identified by Xiong et al., could further enhance our model's predictive capabilities and overall performance.

7.2. Limitations

One critical observation and limitation found while running the experiment was that when quotes were really tight and the spread was very small: like in case the fixed full spread of 0.0745 by the Roll model, or when recent volatility shrank (in GARCH model) and spread reached near the minimum C , these orders were often much more aggressive and landing well within the actual closest bids and asks at the time. For instance, if the strategy's bid price is substantially above the market's best bid, it implies an overpricing (or buying too high) relative to what the market's best buyer was willing to pay. More broadly, the prevailing market bid-ask spread of the actual orderbook data reflects the collective, real-time assessment of short-term volatility and risk by all active market makers. If a strategy consistently generates spreads narrower than this observed market spread, it suggests that the strategy's underlying model (GARCH or Roll) is underestimating market's true variance. This divergence indicates a potential miscalibration, where the strategy consistently underestimates true market risk and fails to capture the real consensus on fair pricing. The backtest would've been more realistic if our strategy's orders were closer to the actual bids and asks quoted by all market participants. This is further validated by Glosten and Milgrom (1985) and O'Hara (1998) which show that market makers widen spreads to protect themselves against informed traders and appropriately manage variance in either direction.

From a data science perspective, this discrepancy is crucial because it highlights a potential flaw in the model's ability to generate realistic, actionable outputs (i.e., variance predictions) based on historical data. Backtesting, as an evaluation framework, relies on the assumption that the simulated actions (i.e., placing realistic quotes) would have plausibly occurred. If the model's outputs (predicted variance) consistently deviate from observed market realities (variance predicted by real historical bids/asks), it compromises the validity and predictive power of the entire simulation. This suggests either the features driving the quote generation (e.g., volatility forecasts) are either miscalibrated, or the model translating those features into quotes is misspecified.

7.3. Implications for Practice and Recommendations

This research offers practical insights for cryptocurrency market makers by empirically comparing dynamic (GARCH-based) and static (Roll-model) spread strategies. The findings indicate that while a static strategy like Roll can achieve significantly higher trade volumes—advantageous if maker rebates are present (additional monetary benefits received by market makers for their presence)—it incurs substantially greater portfolio variance and inventory risk. Conversely, the GARCH strategy, by adapting spreads to volatility, demonstrates superior risk management. This highlights a critical trade-off for practitioners. The developed backtesting engine itself serves as a useful framework for such evaluations. Furthermore, while this study assumed zero trading fees and considered latency implicitly addressed by

100ms data granularity, it underscores the general necessity for practitioners to incorporate realistic trading fees and explicit trade latencies in their own strategy evaluations. This is because these factors critically impact profitability, the reliability of backtest results and applicability of models in real life. Furthermore, quotes should not be set too aggressive with respect to market quotes as that might underestimate future volatility, and neither should they be too far away (as they may underestimate volatility).

From a data science standpoint, the empirical finding that the GARCH model alone was insufficient for a profitable market making. This suggests that its feature set—primarily derived from the serial autocorrelation of historical price volatility—lacked the necessary predictive power to accurately predict variance. This highlights the importance of incorporating richer data sources. The work of Glosten and Milgrom (1985) implies that features engineered from real time order flow data enhances the model's ability to predict and mitigate adverse selection risk. Similarly, Hasbrouck (2006) indicates that market depth data can be transformed into meaningful features, enabling the development of more robust and predictive strategy.

Using a single predictive model like GARCH for volatility forecasting, as in inventory management research (e.g., Kourentzes et al., 2019), may not optimize overall decision-making. Kourentzes et al. showed that tailoring forecasting models to inventory-specific costs and objectives improved outcomes, even if traditional accuracy metrics dipped. Similarly, in market making, a volatility forecast's value lies in its integration into a broader strategy accounting for inventory risk, transaction costs, competition, and portfolio goals; rather than evaluating the volatility model in isolation. Our findings, where the GARCH strategy showed better risk control (akin to better inventory performance) but no clear sign of profitability, align with this principle of considering the larger decision context beyond isolated forecast accuracy.

The final recommendation is based on the Sharpe ratio of both strategies being nearly identical. This finding is important for practitioners because it underscores that sophisticated dynamic strategies may not always yield superior risk-adjusted returns if the dominant factor influencing outcomes is the market exposure, which is how sensitive the portfolio is to price movements of a single asset. While the GARCH model demonstrated better internal risk control (e.g., lower inventory variance), its ultimate risk-adjusted performance was constrained by the same underlying asset trend as the simpler Roll model. Practitioners must therefore carefully consider whether the added complexity and potential data requirements of dynamic models provide sufficient benefit beyond managing common market risk, especially in scenarios where a strong directional trend in the traded asset is the primary determinant of profit/loss. It highlights the need to evaluate strategies not just on their internal mechanics but also in the context of the comprehensive market exposures (or the overall positions) they inherently take.

7.4. Ethical and societal aspects

The research utilizes aggregated, publicly available market data (Bybit CLANKER/USDT trades and order book snapshots), which inherently minimizes direct personal privacy concerns. However, the paper acknowledges broader issues of data fragmentation and inconsistency in cryptocurrency markets, an ethical consideration as the reliability of any data-driven strategy relies heavily on data quality and representativeness. If the described GARCH or Roll strategies were to be implemented on real financial or crypto market, significant ethical diligence would be required. The paper itself highlights their foundational nature and potential for miscalibration—such as quoting spreads far from market consensus or accumulating excessive inventory risk (as seen with the Roll model). Deploying such systems without robust risk controls and further refinement can lead to financial losses for users. Societally, while effective market making—examined in this research—is key to maintaining liquid and efficient markets, high-frequency trading more broadly raises concerns about fair access and the influence of algorithms on market behavior. This is important to mention because it links back to the infamous Flash Crash of 2010 (Kirilenko et al., 2017), where a mutual fund’s automated and poorly designed selling program sold \$4.1 billion of securities and triggered a market crash in the US markets on May 6th, 2010. Our study seeks to deepen understanding and offer an evaluative framework, but applying its insights in practice must involve careful consideration of potential effects on market fairness and stability to avoid negative broader market situations as mentioned above.

8. Conclusion

This research empirically compared two algorithmic market-making models—a static-parameter Roll model and an adaptive GARCH model using a dynamic volatility feature—within a custom-built, high-frequency simulation environment using real-world cryptocurrency order book and trade data. The evaluation revealed that while the simpler Roll model achieved a significantly higher interaction rate (trade count), the adaptive GARCH model demonstrated superior risk management, evidenced by substantially lower variance in its cumulative performance metric (portfolio value) and more effective control of a critical internal state variable (inventory). Specifically, the Roll model's portfolio value variance was 13 times higher, and its inventory standard deviation was 6 times greater, leading to larger negative performance deviations.

These findings highlight, from a data science perspective, the benefits of adaptive models that incorporate dynamic features for improved robustness and risk control in managing complex, time-series data. While neither model achieved positive cumulative outcomes, the GARCH model's adaptive approach to parameter setting based on a volatility feature provided a more stable and risk-averse performance profile, suggesting that incorporating dynamic, data-driven features is crucial for developing more effective market-making algorithms.

References

- Juliano, A. (2017). dydx: A standard for decentralized derivatives. <https://coinprika.com/storage/cdn/whitepapers/448405.pdf>
- Foucault, T., Pagano, M., & Röell, A. (2023). Market liquidity: theory, evidence, and policy. Oxford University Press. <https://doi.org/10.1093/oso/9780197542064.003.0002>
- Kyle, A. S. (1985). Continuous auctions and insider trading. *Econometrica: Journal of the Econometric Society*, 1315-1335. <https://doi.org/10.2307/1913210>
- Avellaneda, M., & Stoikov, S. (2008). High-frequency trading in a limit order book. *Quantitative Finance*, 8(3), 217-224. <https://doi.org/10.1080/14697680701381228>
- Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. <https://bitcoin.org/bitcoin.pdf>
- BitMEX. (2016, May 13). Announcing the launch of the perpetual XBTUSD leveraged swap. BitMEX Blog. <https://blog.bitmex.com/announcing-the-launch-of-the-perpetual-xbtusd-leveraged-swap/>
- Roll, R. (1984). A simple implicit measure of the effective bid-ask spread in an efficient market. *The Journal of Finance*, 39(4), 1127–1139. <https://doi.org/10.1111/j.1540-6261.1984.tb03897.x>
- Box, G. E., Jenkins, G. M., Reinsel, G. C., & Ljung, G. M. (2015). Time series analysis: forecasting and control. John Wiley & Sons. <https://doi.org/10.1111/jtsa.12194>
- Hull, J. (2010). *Options, Futures, and Other Derivatives*. India: Pearson Education.
- Sharpe, W. F. (1964). Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3), 425-442. <https://doi.org/10.2307/2977928>
- Xiong, Y., Yamada, T., & Terano, T. (2015, December). Comparison of different market making strategies for high frequency traders. In 2015 Winter Simulation Conference (WSC) (pp. 324-335). IEEE. <https://doi.org/10.1109/wsc.2015.7408175>
- Bitcoin Options open interest, Bitcoin options trading volume, Cryptocurrency options data | CoinGlass. (n.d.). Coinglass. <https://www.coinglass.com/options>

Black, F., & Scholes, M. (1973). The pricing of options and corporate liabilities. *Journal of political economy*, 81(3), 637-654.

Bollerslev, T. (1986). Generalized Autoregressive Conditional Heteroskedasticity. *Journal of Econometrics*, 31(3), 307–327. [https://doi.org/10.1016/0304-4076\(86\)90063-1](https://doi.org/10.1016/0304-4076(86)90063-1)

Engle, R. F. (1982). Autoregressive conditional heteroscedasticity with estimates of the variance of United Kingdom inflation. *Econometrica*, 50(4), 987-1007. <https://doi.org/10.2307/1912773>

O'Hara, M. (1998). *Market microstructure theory*. John Wiley & Sons.

Mandelbrot, B. (1963). The variation of certain speculative prices. *The Journal of Business*, 36(4), 394-419. <https://doi.org/10.1086/294632>

Cont, R. (2001). Empirical properties of asset returns: stylized facts and statistical issues. *Quantitative Finance*, 1(2), 223–236. <https://doi.org/10.1080/713665670>

Stoikov, S., Zhuang, E., Chen, H., Zhang, Q., Li, S., Wang, S., & Shan, C. (2024). Market Making in Crypto. Available at SSRN. <https://doi.org/10.2139/ssrn.5066176>

CBS. (2011, June 6). Wall Street: The speed traders. YouTube. <https://youtu.be/I5nAovTXu6I>

Hasbrouck, J. (2007). *Empirical market microstructure: The institutions, economics, and econometrics of securities trading*. Oxford University Press. <https://doi.org/10.1093/oso/9780195301649.001.0001>

Glosten, L. R., & Milgrom, P. R. (1985). Bid, ask and transaction prices in a specialist market with heterogeneously informed traders. *Journal of financial economics*, 14(1), 71-100. [https://doi.org/10.1016/0304-405x\(85\)90044-3](https://doi.org/10.1016/0304-405x(85)90044-3)

Fama, E. F. (1970). Efficient capital markets. *Journal of finance*, 25(2), 383-417.

Chakraborty, T., & Kearns, M. (2011, June). Market making and mean reversion. In *Proceedings of the 12th ACM conference on Electronic commerce* (pp. 307-314). <https://doi.org/10.1145/1993574.1993622>

Kirilenko, A., Kyle, A. S., Samadi, M., & Tuzun, T. (2017). The flash crash: High-frequency trading in an electronic market. *The Journal of Finance*, 72(3), 967-998. <https://doi.org/10.2139/ssrn.1686004>

Kourentzes, N., Trapero, J. R., & Barrow, D. K. (2020). Optimising forecasting models for inventory planning. *International Journal of Production Economics*, 225, 107597. <https://doi.org/10.2139/ssrn.3363117>