



Time Evolution of Pilot Waves in Finance

by

© Amir Gholizad

A thesis submitted to the School of Graduate Studies in partial fulfillment of the requirements for the degree of Master of Science.

Department of Physics and Physical Oceanography
Memorial University

January 2025

St. John's, Newfoundland and Labrador, Canada

Abstract

This thesis explores the time evolution of quantum potentials in financial time series by developing novel technical indicators based on these quantum potentials. It starts with an introduction to de Broglie-Bohm's pilot wave theory and its application in financial market analysis, detailing the derivation of quantum potential and quantum force from the probability density function (PDF) of logarithmic price changes. The research employs the sliding window technique to observe the temporal evolution of these quantum potentials and quantum forces and their corresponding features, while also examining correlations with traditional technical and statistical indicators.

A significant part of the study involves comparing real financial markets to a theoretical Gaussian system defined by random walk theory, revealing notable differences in the number of equilibrium states and the shape of constraining walls. Strong correlations are found between the new features-particularly the length and depth of quantum potentials-and other indicators like standard deviation.

Ultimately, this research advances the use of the pilot wave model in finance by introducing the "widening" behavior of quantum potentials and providing fully functional indicators for future research.

Acknowledgements

I would like to express my deepest gratitude to my supervisors, Dr. James P. F. LeBlanc and Dr. Emmanuel Haven, for granting me the freedom to choose this project and for their invaluable guidance throughout my two-year program.

I am also thankful to the Department of Physics and Physical Oceanography and the School of Graduate Studies at Memorial University of Newfoundland for their financial support.

Finally, I extend my heartfelt thanks to Mr. Vahid Sheigani, a brilliant physics student, whose exceptional knowledge of financial markets greatly supported my work.

Table of contents

Title page	i
Abstract	ii
Acknowledgements	iii
Table of contents	iv
List of tables	vii
List of figures	viii
List of symbols	xi
List of abbreviations	xii
1 Introduction	1
1.1 New approaches to financial markets	4
1.2 Our model	6
2 Mathematical and computational tools	9
2.1 Estimating probability density functions	9
2.1.1 Gaussian density estimation (GDE)	10
2.1.2 Histogram	10

2.1.3	Kernel density estimation (KDE)	12
2.1.4	Choosing the right bandwidth	14
2.2	Symmetric derivatives	17
2.3	Pilot wave theory	18
2.3.1	Quantum potential	20
2.3.2	Quantum force	20
2.3.3	Bohmian mechanics in finance	21
2.4	Sliding window technique	23
3	Model and results	27
3.1	Price time series	27
3.1.1	Data access	29
3.2	Estimating probability density functions	30
3.2.1	GDE	32
3.2.2	KDE	32
3.3	Quantum potentials	33
3.4	Market forces	36
3.5	Multi-time frame analysis	38
3.6	Features	45
3.6.1	Time evolution of features	45
3.6.2	Analysis	52
4	Conclusion	55
4.1	Deviations of real markets from random walk theory	56
4.2	Correlations between technical, statistical, and quantum-based indicators	56
4.3	The future	57

5	Appendix A: Code Snippets	58
5.1	Download and pre-processing historical data	58
5.1.1	yahoo!finance API	58
5.1.2	Log-returns	59
5.1.3	Numerical differentiation	59
5.2	Creating density, quantum potentials, and quantum forces	60
5.2.1	GDE	60
5.2.2	KDE	61
5.2.3	Quantum potential and quantum force	61
5.3	Features (technical indicators)	63
5.3.1	Potential depth	63
5.3.2	Potential length (width)	63
5.3.3	Local maximum points	64
5.3.4	Local minimum points	64
5.3.5	Market force at $q = \mu$	65
	Bibliography	67

List of tables

3.1	Tick data sample, Microsoft.	28
3.2	Daily bar data sample, Microsoft.	29
3.3	Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of Tesla Inc., window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5	50
3.4	Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of Bitcoin, window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5	50
3.5	Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of S&P500 ETF Trust, window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5	51
3.6	Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of Johnson&Johnson, window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5	51

List of figures

1.1	Schematic representation of our developed model	8
2.1	Histogram of daily log-returns of AAPL from 28/09/2013 to 23/09/2023.	11
2.2	Kernel estimates showing individual kernels.	13
2.3	Kernel estimates showing individual kernels with smaller window width $b = 0.004$. Compare with Fig. 2.2 where $b = 0.008$	13
2.4	Compariosn of Gaussian (GDE) and kernel (KDE) estimates of daily log-returns of AAPL from 28/09/2013 to 23/09/2023.	17
2.5	Sliding Window Technique for Making Technical Indicators (Example: Simple Moving Average with Window Size 3)	25
3.1	Daily close prices of (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023	30
3.2	Distribution of log-returns of daily close prices of (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023	31
3.3	Gaussian density estimation (GDE) and kernel density estimation (KDE) of log-returns of daily close prices for (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023	33
3.4	GDE and KDE quantum potentials of log-returns of daily close prices for (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023	35

3.5	GDE VS KDE market force comparison, generated from log-returns of daily close prices for (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023	37
3.6	Gaussian and kernel density estimations for the log-returns of Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)	38
3.7	Gaussian and kernel density estimations for the log-returns of S&P500 ETF Trust in multiple time frames (from 28/09/2013 to 23/09/2023)	39
3.8	Quantum potentials of Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)	40
3.9	Quantum potentials of S&P500 ETF Trust in multiple time frames (from 28/09/2013 to 23/09/2023)	41
3.10	Market forces of Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)	42
3.11	Market forces of S&P500 ETF Trust in multiple time frames (from 28/09/2013 to 23/09/2023)	43
3.12	Kurtosis and number of equilibrium points of S&P500 ETF Trust and Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)	44
3.13	Daily close price and indicators based on market potential and market force of Tesla Inc. from 28/09/2021 to 23/09/2023 (window size = 30 days)	46
3.14	Daily close price and indicators based on market potential and market force of Bitcoin (USD). from 28/09/2021 to 23/09/2023 (window size = 30 days)	47
3.15	Daily close price and indicators based on market potential and market force of S&P500 ETF Trust. from 28/09/2021 to 23/09/2023 (window size = 30 days)	48
3.16	Daily close price and indicators based on market potential and market force of Johnson&Johnson. from 28/09/2021 to 23/09/2023 (window size = 30 days)	49

3.17	The correlations between Length and Depth of Potential and STDEV (normalized to [0,1], window size = 30 days)	52
3.18	Comparison of two Potentials with different STDEV (σ) values (Ex- ample: TSLA, window size = 14 days)	53

List of symbols

σ	Standard deviation
μ	Mean, average, or expected value
κ	Kurtosis or tailedness of a distribution
\hbar	Reduced Planck constant = $6.582 \times 10^{-16} eV.s$
$\mathcal{N}(\mu, \sigma^2)$	Normal or Gaussian distribution with mean μ and variance σ^2

List of abbreviations

KURT	Kurtosis
SKEW	Skewness
QF	Quantum force
QP	Quantum potential
STDEV	Standard deviation
MSE	Mean squared error
SMA	Simple moving average
RSI	Relative strength index
Log-return	Logarithmic rate of return
EMH	Efficient market hypothesis
PDE	Partial differential equation
PDF	Probability density function
GBM	Geometric Brownian motion
MISE	Mean integrated squared error
i.i.d	Independent, identically distributed
MACD	Moving average convergence/divergance

Chapter 1

Introduction

In March 1900 a French mathematician, Louis Bachelier, presented the first formalization of a random walk theory in his doctoral thesis. He believed the movements of the stock exchange are influenced by countless factors, both natural and artificial, making it impossible to mathematically predict market fluctuations with complete accuracy. Although conflicting viewpoints coexist, complicating the use of probability theory for understanding price movements, it's still feasible to conduct a mathematical analysis of the market's static condition at a specific point by determining the likelihood of certain price variations. His work aimed to explore and develop a mathematical formula for understanding these probabilities, building on theoretical insights related to stock exchange operations. He laid the foundation for the absolute Brownian motion model in finance by determining the probability of prices through the introduction of what is now known as the Chapman-Kolmogorov equation and by recognizing that a Wiener process satisfies the diffusion equation. This led to the estimation of a normal or Gaussian probability density function for price values, which was later found to be inaccurate. Despite several errors, Bachelier's thesis remains a pioneering work in finance[1, 2]. It is regrettable that his work went unnoticed for so many years until Paul Samuelson, an American economist, brought it to light.

In 1947, Samuelson released a book derived from his doctoral work at Harvard[3]. Unlike Bachelier's thesis, Samuelson's publication was revolutionary. His book, along with a later textbook[4] that eventually became the best-selling economics text ever, enabled others to recognize the significance of Bachelier's contributions, nearly fifty

years after they were made. Paul Samuelson expanded upon Louis Bachelier’s pioneering work on financial modeling by introducing a more sophisticated framework that better captured the dynamics of asset prices. While Bachelier’s model utilized absolute Brownian motion with normally distributed returns, Samuelson’s model introduced the concept of geometric Brownian motion (GBM), which assumes that asset prices follow a log-normal distribution. This innovation accounted for the fact that prices cannot fall below zero and reflected the compounding nature of returns, which was a key limitation in Bachelier’s original approach. His model laid the foundation for modern financial theory[5].

The second correction to Bachelier’s model was made by a physicist named Maury Osborne[6]. He proposed that the prices of common stocks and the value of money can be considered as a collection of decisions in statistical equilibrium, resembling an ensemble of particles in statistical mechanics. Assuming that two prices for a security at times t , $t + \tau$ are $p(t)$ and $p(t + \tau)$ where τ is the time step, if we define the price change (price log-return) as

$$q(t) = \ln(p(t + \tau)) - \ln(p(t)), \quad (1.1)$$

then the steady-state distribution function of q is $f(q) = \exp(-q^2/2\sigma^2\tau)/\sqrt{2\pi\sigma^2\tau}$, which is the same as the probability distribution for a particle undergoing geometric Brownian motion, where σ represents the dispersion after one unit of time. In other words, Osborne believed it is the “rate of return” of prices, not the price itself that exhibits a geometric Brownian motion[7].

Another major correction to the stochastic model of financial time series was introduced by the renowned mathematician, Benoit Mandelbrot[8]. In order to understand the significance of Mandelbrot’s work we need to first learn a bit about a kind of distributions that are known as “fat-tailed” distributions. Normal and fat-tailed distributions are two types of probability distributions that describe the likelihood of different outcomes in a dataset, or in our case the frequency of price returns. Normal distribution, often referred to as the Gaussian distribution, is symmetric and bell-shaped. It is characterized by a single peak at the mean, with the tails of the distribution approaching zero as they move away from the mean. Meanwhile, a fat-tailed distribution has heavier tails than the normal distribution, meaning that it has a higher probability of extreme outcomes (values far from the mean) than would be expected

in a normal distribution. This results in a distribution that appears more spread out and has more probability mass in the tails. Also, the tails of a normal distribution decrease exponentially, meaning that the probability of observing values further from the mean decreases rapidly while the tails of a fat-tailed distribution decrease more slowly than in a normal distribution, indicating that extreme values are more likely to occur in the dataset. In 1963, Mandelbrot proposed to replace the Gaussian distribution of price returns with “stable Paretian” probability distributions, which were first introduced by Paul Lévy[9]. Stable Paretian or “Lévy stable” distributions include a broader range of probability density functions (PDFs), including those with fat tails. Mandelbrot argued that this family of distributions does a better job in explaining the sharp peaks and extreme events observed in empirical price changes compared to the Gaussian distribution[8].

Building on the GBM model of Bachelier and Osborne, the American mathematician, Edward Thorp, pioneered the concept of “hedging” for the first time[10]. Thorp’s methods led to the development of additional strategies involving convertible securities (financial instruments) similar to options, which can be exchanged for another type of security, such as bonds or preferred shares that can be converted into common stock.

A few years later, inspired by Thorp’s groundbreaking work, Myron Scholes and Fischer Black introduced their famous option pricing model. This model allowed traders to determine the fair price of an option contract, assuming a fixed volatility (σ) for the underlying asset. Black and Scholes formulated a partial differential equation (PDE), which, when solved, provides the fair price of an option contract. This well-known equation is referred to as the Black-Scholes PDE[11].

The Black-Scholes model is based on the principles of random walk theory and geometric Brownian motion. Although the model is built on the assumption of normally distributed returns, empirical data often exhibit fat tails, indicating very high kurtosis values. This discrepancy means that if a model is over-fitted to historical data, it may not produce reliable predictions. To address this, researchers such as Lisa Borland[12] have sought to enhance the Black-Scholes model using non-Gaussian distributions, leading to progressively better estimates. Since the 1950s, there has been a growing interest among mathematicians and physicists in modeling stock markets, leading to the development of new financial models, or the refinement of existing ones[1].

1.1 New approaches to financial markets

Within the past couple of decades, several new approaches to financial time series have emerged, with those explored by physicists falling under the field known as “econophysics”. Econophysics is an interdisciplinary field referred to as a science and methodology that studies financial markets and economic events as physical systems. It uses theories and techniques initially created by physicists to address economic problems and covers study areas such as climate economics, economic networks, emergent phenomena, financial time-series, image processing, adaptive behavior, market microstructure, risk management, social dynamics, quantum social science, and wealth and income inequality. The term econophysics was coined by Eugene Stanley, an American physicist, in the mid-1990s, and the field has grown as a branch of both physics and economics. One of the newest models employed by econophysicists is David Bohm’s causal interpretation of quantum mechanics, well known as pilot wave theory[13, 14, 15].

Pilot wave theory¹ presents a quasi-deterministic framework for interpreting quantum phenomena within quantum mechanics. This theory, formulated by Louis de Broglie and David Bohm, provides a causal explanation for the behavior of particles at the quantum level. At the heart of pilot wave theory is the idea that particles have well-defined positions and velocities at all times, guided by an underlying wave function. Unlike the conventional Copenhagen interpretation, which views quantum mechanics as inherently probabilistic and wave functions as representing mere probabilities, pilot wave theory treats the wave function as a real physical entity that directs the motion of particles. In pilot wave theory, the wave function is considered a guiding field that influences the trajectory of particles. This field, often referred to as the “pilot wave,” propagates through space and time, determining multiple paths that particles might take. The wave function evolves according to Schrödinger’s equation, while particles follow an ensemble of possible paths governed by the guidance equation[14].

In 2007 O. Choustova[16] and again in 2015 F. Tahmasebi et al. [17] presented a new approach for studying fat-tailed distributions of log-returns. These studies applied de Broglie-Bohm’s pilot wave theory for studying financial markets, finding that price return fluctuations were influenced by the entanglement between current and

¹Also known as de Broglie-Bohm theory.

prior-day prices. Using the Bohmian quantum approach, Tahmasebi et al. showed that the quantum potential confined price returns in a scale-invariant manner, indicating that the extent of confinement varies with the time scale of the observations. They attributed this confinement of log-returns to the entanglement between the current and past states of the market. This perspective directly contradicts the efficient market hypothesis (EMH), which assumes that market data consists of independent and identically distributed (i.i.d) random variables. Analysis of oil and gold markets revealed short-term effectiveness of quantum potential, while long-term influences resembled white noise.

Two years later in another study Chen Shen and Emmanuel Haven[18] estimated real and quantum potentials from financial commodities by analyzing the log returns of six common commodities. By comparing quantum and classical potentials, they demonstrated differing information, suggesting quantum potentials may reflect social and psychological factors, while classical potentials indicate market conditions. They also estimated the Pearson correlation between the potential forces, indicating a medium or weak interaction effect. Recent studies have underscored the importance of quantum potentials in risk management. For instance, a research by Hossein Khaksar et al. introduces the concept of quantum risk and analyzes portfolio optimization using quantum potentials. The results demonstrate that quantum risk exhibits a power-law behavior over time, similar to standard deviation but with a different (smaller) exponent[19].

There are many emerging works being done in the field of quantum finance. Some of them demonstrate that Hamiltonians and operators, traditionally used in physics, can effectively model financial systems, particularly in analyzing the dynamics of information exchange between traders and portfolio management[20].

Other works extend microeconomics by introducing a statistical approach akin to the transition from classical to statistical mechanics, modeling market prices and quantities as stochastic processes and proposing a specific model to analyze the dynamics of these market variables through an “action functional”[21].

One of the oldest examples of employing physical interpretations in finance is a research in 1985 by Albert S. Kyle that presents a dynamic model of insider trading using sequential auctions to explore the informational content of prices, market liquidity, and the value of private information, showing that in a continuous trading

limit, prices follow geometric Brownian motion, market depth remains constant, and all private information is eventually reflected in prices[22].

In a very recent research, David Orrell develops a quantum model for financial time series, where energy levels correspond to transaction counts, addressing limitations of classical oscillators by incorporating quantum features like non-zero volatility and capturing the non-Gaussian nature of financial statistics, accurately predicting phenomena such as the square-root law of price impact[23].

1.2 Our model

This project builds upon the foundational works of Choustova[16], Shen et al.[18], and Tahmasebi et al.[17] by investigating the characteristics of newly identified quantum potentials in finance. Our primary objective is to explore the time evolution of quantum potentials and quantum forces using numerical methods rather than relying on analytical solutions.

In the next chapter, we establish the mathematical and methodological foundations of our research by adapting de Broglie-Bohm’s pilot wave theory to suit our purposes. We will guide you through the methods used in this study, from estimating the probability density functions (PDFs) of price changes to computationally deriving the quantum potentials and creating the corresponding technical indicators.

Chapter 3 outlines the entire process of extracting, developing, and analyzing the time evolution of new technical indicators based on the properties of quantum potentials and quantum forces. After estimating the PDFs of price changes for a selected set of assets, and utilizing other computational techniques such as numerical differentiation, we construct steady-state quantum potentials and quantum forces. These physical entities reveal significant differences between real markets and GBM models, such as the variation in the number of equilibrium states and the shape of the constraining walls of quantum potentials.

By varying the time frame τ^2 we observe notable differences in the structure of quantum potentials, quantum forces, and the overall system. This experiment shows that, over longer time frames (e.g., monthly versus daily), the market tends to behave

² τ may also be referred to as the time step, time scale, or resolution, depending on the community (traders, financial analysts, or scientists) and the relevant literature.

more like a normal system.

To study the time evolution of quantum potentials and quantum forces, we assign specific features to them and employ a sliding window technique to develop new technical indicators. These indicators enable us to monitor the evolution of features and their associated entities with each step τ . By examining the Pearson correlations between the new indicators and several well-known technical and statistical indicators, we identify a strong correlation with standard deviation, which we attribute to volatility and market risk, referring to it as the “widening” behavior of quantum potentials. These findings will be discussed in more detail in Chapter 4.

Appendix A provides open-source Python scripts and a set of fully functional technical indicators with adjustable window sizes for the reader’s use. Figure 1.1 provides a detailed schematic representation of the model we have developed. This figure illustrates the key components and their interconnections, offering a comprehensive overview of the structure and functionality of our designed model.

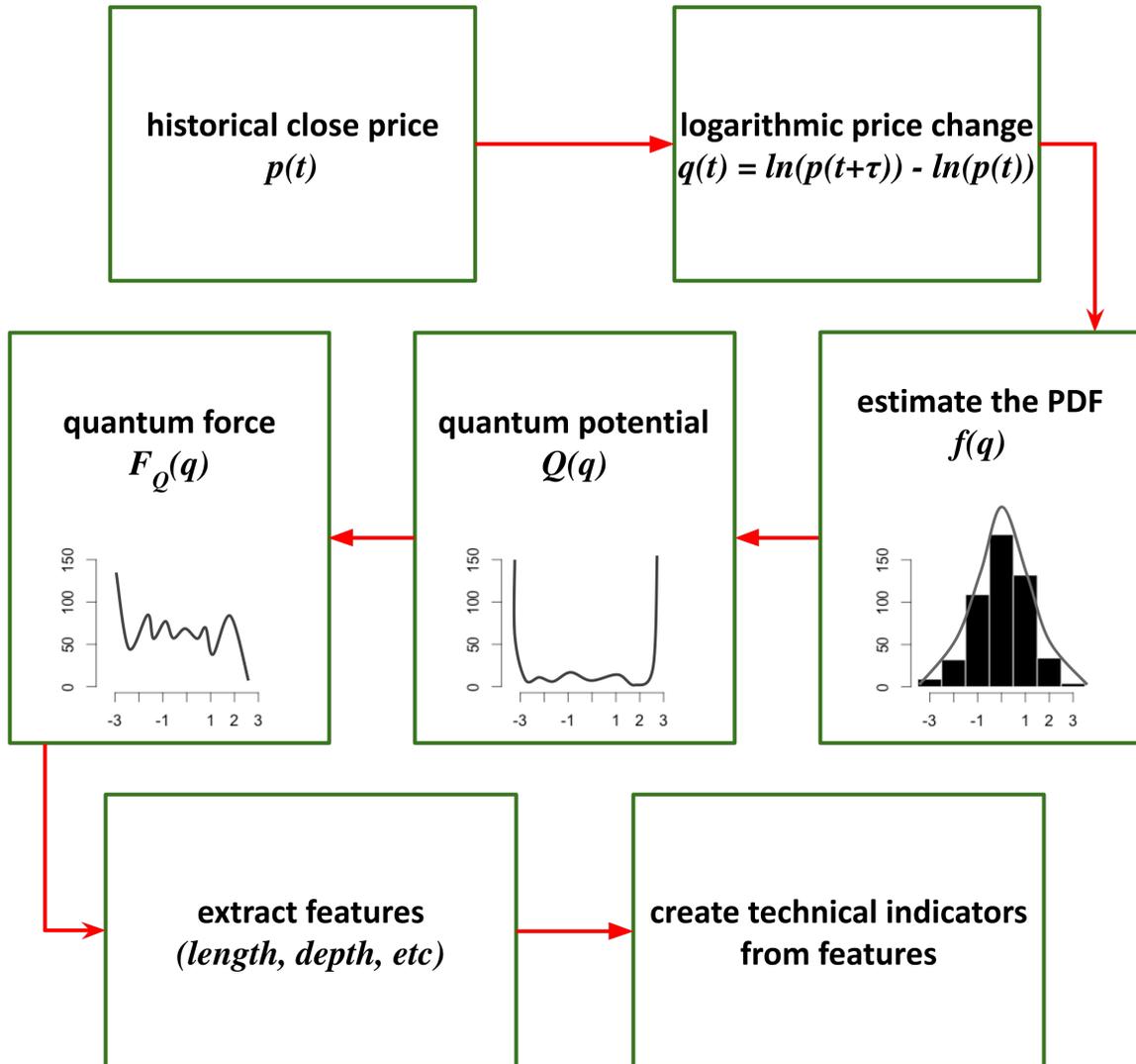


Figure 1.1: Schematic representation of our developed model

Chapter 2

Mathematical and computational tools

This chapter delves into the mathematical and computational tools employed throughout this project. We start by presenting the mathematical and computational techniques used to estimate the probability density functions governing distributions (i.e., histograms) of logarithmic price returns. Moving on, we introduce de Broglie-Bohm's pilot wave theory, an alternative interpretation of quantum mechanics. Next, we explore the concept of quantum potentials and demonstrate how a financial time series can be modeled within this framework, drawing on the works of F. Tahmasebi et al.[17] and C. Shen et al.[18] Finally, we present our own model and the computational methods used to study the time evolution of quantum potentials and quantum forces.

2.1 Estimating probability density functions

Probability density functions (PDFs) offer high-resolution statistical insights into a data set. Consider a random variable Q . The probability associated with Q taking values within an interval can be expressed by the PDF f , satisfying the relation[24]

$$P(a < Q < c) = \int_a^c f(q) dq \quad \forall a < c.$$

Now, suppose we have a set of n independent, identically distributed (i.i.d.) random variables Q_1, Q_2, \dots, Q_n . These random variables could be the logarithmic price changes of a financial time series obtained from Eq. 1.1. The PDF of these random variables can be estimated using both parametric and non-parametric methods. Estimating the density function for this data set offers several computational advantages, including differentiability and increased precision. This section will explore the mathematical procedures for estimating the best-fit PDF for a given data set.

2.1.1 Gaussian density estimation (GDE)

A Gaussian fit is a parametric approach and one of the simplest methods for estimating a PDF. Given a dataset of i.i.d. data with mean μ and variance σ^2 , the underlying density \hat{f} can be estimated by first computing the sample mean μ and variance σ^2 from the data, and then using these estimates in the formula for the normal distribution as

$$\text{GDE} : \hat{f}_G(q) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(q-\mu)^2}{2\sigma^2}\right). \quad (2.1)$$

This technique, known as ‘‘Gaussian density estimation’’ or GDE, fits a normal distribution $\mathcal{N}(\mu, \sigma^2)$ to any data set. We will apply this method to model the market as a normal system, where price changes are simulated as following geometric Brownian motion. Note that when estimating PDF of log-returns, μ represents the average log-return, and σ is the standard deviation.

2.1.2 Histogram

The histogram is the oldest and most widely used non-parametric density estimator. It provides a realistic approximation of distributions, as it involves minimal smoothing. The histogram is defined as[24]

$$\hat{f}(q) = \frac{N_i}{nb},$$

where N_i is the number of Q_i in the same bin as q and n is the total number of observations. The bins of the histogram are defined as the intervals $[q_0 + mb, q_0 +$

$(m + 1)b$), where q_0 and b are the origin and bin width and $m \in \mathbb{Z}$.

Selecting the right origin and bin width are important for constructing histograms. Note that increasing the number of bins results in a less smoothed data representation, and vice versa. Figure 2.1 shows an example histogram of daily log-returns of AAPL.

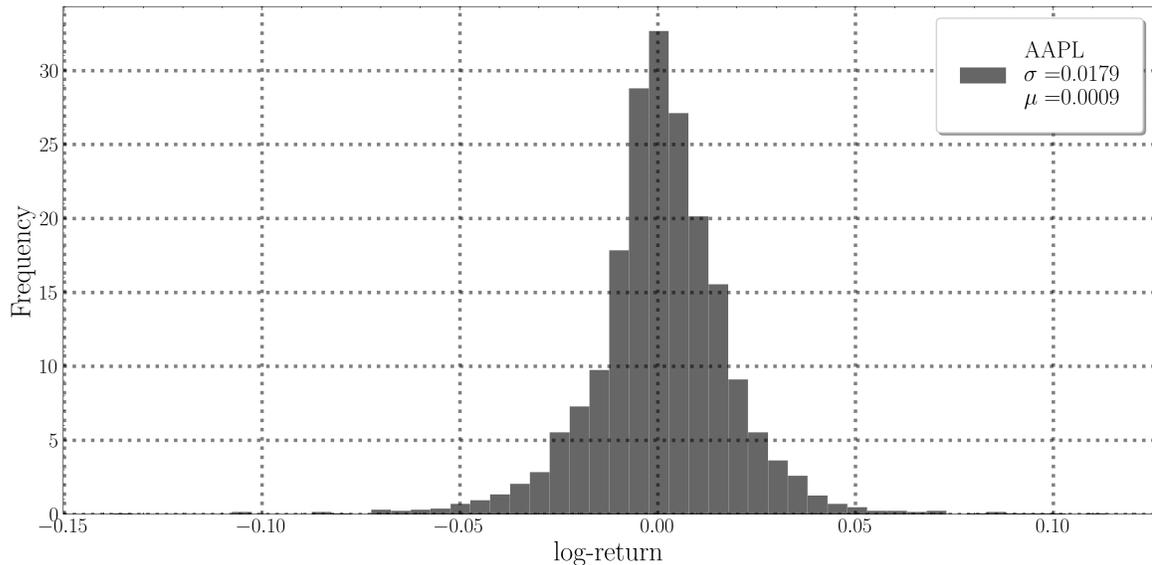


Figure 2.1: Histogram of daily log-returns of AAPL from 28/09/2013 to 23/09/2023.

By defining the probability density f of a random variable Q as

$$f(q) = \lim_{b \rightarrow 0} \frac{1}{2b} P(q - b < q < q + b),$$

for a given b , the probability $P(q - b < q < q + b)$ can be estimated by the proportion of sample points within the interval $(q - b, q + b)$. Therefore, a straightforward estimator \hat{f} of the density can be defined by selecting a small value for b as below[24]

$$\hat{f}(q) = \frac{N_P}{2bn},$$

where N_P is the number of Q_i falling within $(q - b, q + b)$. This estimator is referred to as the “naive estimator”. We can also define the weight function w as

$$w(q) = \begin{cases} \frac{1}{2} & \text{if } |q| < 1 \\ 0 & \text{otherwise} \end{cases}. \quad (2.2)$$

Using this definition, the naive estimator can be expressed as[24]

$$\hat{f}(q) = \frac{1}{n} \sum_{i=1}^n \frac{1}{b} w\left(\frac{q - Q_i}{b}\right).$$

By using each data point as the center, this method eliminates the need to specify a position q_0 . However, selecting the appropriate value for b remains necessary.

Histograms and Naive estimates are well known for their “stepwise” look which doesn’t allow differentiation. To address this issue, as well as for additional technical reasons discussed later, we will explore a more advanced estimation method in the following section.

2.1.3 Kernel density estimation (KDE)

We can improve the naive estimator to address some of the previously mentioned challenges. Initially, we can substitute the weight function w with a kernel function that meets the requirement

$$\int_{-\infty}^{\infty} K(q) dq = 1. \tag{2.3}$$

The kernel K can be any probability density function, but in this case, we will use $\mathcal{N}(0, 1)$, which represents a normal distribution with mean $\mu = 0$ and variance $\sigma^2 = 1$. Referring to the naive estimator definition, the kernel estimator using kernel K is given by[24]

$$\hat{f}(q) = \frac{1}{nb} \sum_{i=1}^n K\left(\frac{q - Q_i}{b}\right). \tag{2.4}$$

Here, b denotes the window width, also known as the “smoothing parameter” or “bandwidth”. Unlike the naive estimator, which sums “boxes” centered on the observations, the kernel estimator sums individual “probability densities” centered on each data point. The kernel function K defines the shape of these small PDFs, while the window width b controls their size. Figure 2.2 demonstrates the KDE method, displaying both the individual probability densities and the estimate \hat{f} obtained by summing them.

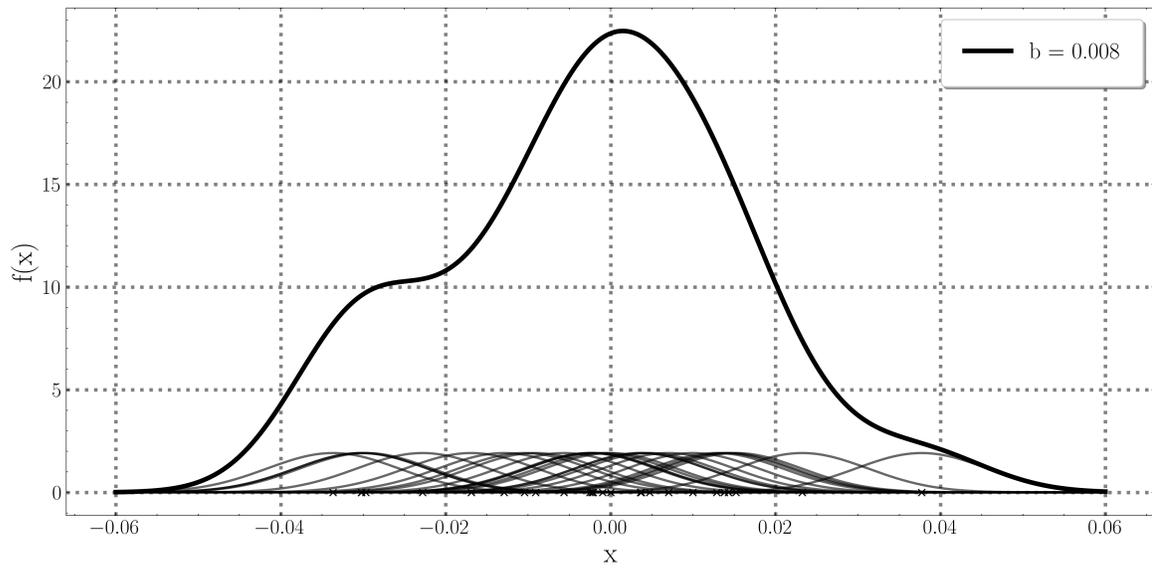


Figure 2.2: Kernel estimates showing individual kernels.

Figure 2.3 illustrates the effect of varying the window width. As b decreases towards zero, the small PDFs become sharply peaked, while a larger b produces flatter PDFs, leading to a loss of detail.

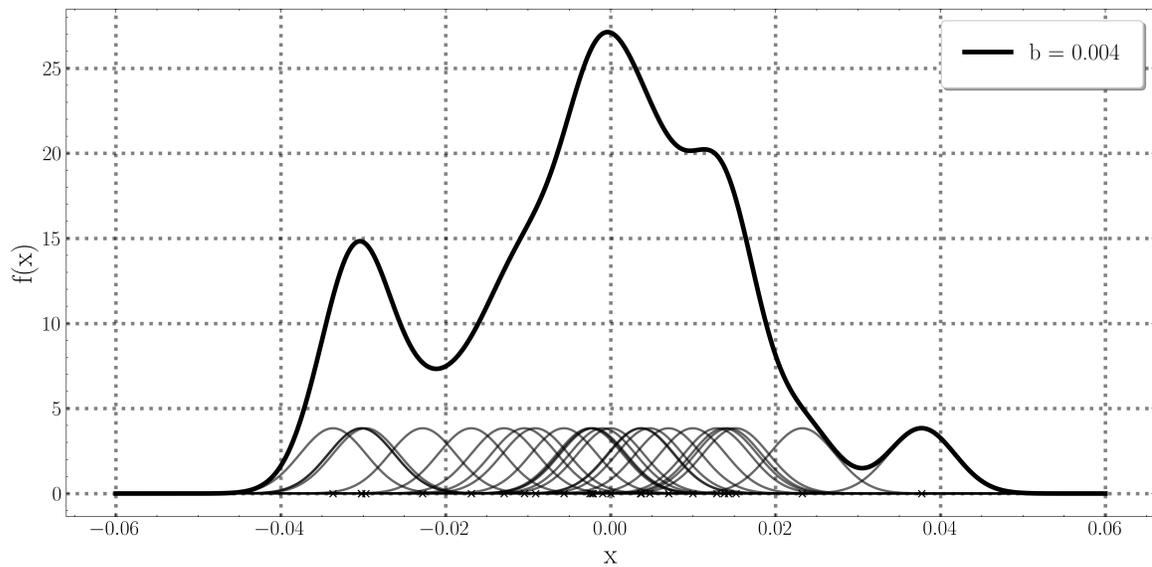


Figure 2.3: Kernel estimates showing individual kernels with smaller window width $b = 0.004$. Compare with Fig. 2.2 where $b = 0.008$

The kernel estimator \hat{f} has key properties based on its definition. If the kernel K is

non-negative and meets the normalization condition (i.e., it acts as a probability density function), then \hat{f} will also be a valid probability density function. Furthermore, \hat{f} will reflect the continuity and smoothness of K ; for instance, a normal density kernel results in a smooth estimator with derivatives of all orders. Despite its widespread use, the kernel estimator faces challenges with long-tailed distributions. A fixed window width can add noise to the tails, and excessive smoothing to reduce this noise may mask important features of the main distribution. Adaptive methods to address this issue will be explored in the following section.

2.1.4 Choosing the right bandwidth

Several methods exist to quantify the error between the density estimator \hat{f} and the true density f . A widely recognized metric for assessing this at a specific point is the mean squared error (MSE), which is defined as

$$\text{MSE}_q(\hat{f}) = \text{E}[(\hat{f}(q) - f(q))^2]. \quad (2.5)$$

This can be further broken down into[24]

$$\text{MSE}_q(\hat{f}) = (\text{E}[\hat{f}(q)] - f(q))^2 + \text{Var}[\hat{f}(q)], \quad (2.6)$$

which represents the sum of the squared bias ($\text{E}[\hat{f}(q)] - f(q) = \text{bias}_q(\hat{f}, f)$) and the variance at point q . Using a kernel defined as $\hat{f}(q) = \frac{1}{n} \sum_{i=1}^n w(Q_i, q)$, with a weight function set to

$$w(y, q) = \frac{1}{b} K\left(\frac{q - y}{b}\right),$$

the mean and the variance will look like

$$\text{E}[\hat{f}(q)] = \int \frac{1}{b} K\left(\frac{q - y}{b}\right) f(y) dy,$$

$$n \text{Var}[\hat{f}(q)] = \int \frac{1}{b^2} K\left(\frac{q - y}{b}\right)^2 f(y) dy - \left(\frac{1}{b} \int K\left(\frac{q - y}{b}\right) f(y) dy\right)^2.$$

It is evident that by adjusting the degree of smoothing, one can trade off between reducing bias and increasing variance, or vice versa.

To measure the error across the entire dataset, we use the mean integrated squared error (MISE) method[25]. The MISE is defined as

$$\text{MISE}(\hat{f}) = \mathbb{E} \left[\int (\hat{f}(q) - f(q))^2 dq \right]. \quad (2.7)$$

By moving the expectation inside the integral and substituting the integrand with Eq. 2.6, we get

$$\begin{aligned} \text{MISE}(\hat{f}) &= \int \mathbb{E}[(\hat{f}(q) - f(q))^2] dq \\ &= \int \text{MSE}_q(\hat{f}) dq, \end{aligned}$$

or more explicitly as

$$\text{MISE}(\hat{f}) = \int (\mathbb{E}[\hat{f}(q)] - f(q))^2 dq + \int \text{Var}[\hat{f}(q)] dq, \quad (2.8)$$

which expresses the MISE as the sum of the integrated squared bias and the integrated variance. This method provides a comprehensive evaluation of the estimator's accuracy by integrating the squared difference between the estimated density $\hat{f}(q)$ and the true density $f(q)$ over the entire range of q .

Based on the detailed mathematical derivations by Silverman, 1986[24], Equation 2.8 can be transformed and minimized to determine the optimal bandwidth in terms of the expected density f and the kernel function K . Result of this minimization is[24]

$$b_{\text{opt}} = k_2^{-2/5} \left\{ \int K(y)^2 dy \right\}^{1/5} \left\{ \int f''(q)^2 dq \right\}^{-1/5} n^{-1/5}, \quad (2.9)$$

where k_2 is the second moment of the kernel function ($k_2 = \int y^2 K(y) dy$).

Selecting the appropriate amount of smoothing is crucial in density estimation and is particularly important for our project. It is essential to remember that the choice of smoothing parameter should always align with the intended use of the density estimate. In our model, which involves differentiating estimated density functions, it is imperative to find a bandwidth that minimizes additional noise without leading to

overfitting. To achieve this, we plotted several curves and selected the estimate that best aligned with our prior expectations of the density. This trial-and-error approach is satisfactory for many applications. Ultimately, we opted to use the ‘‘Silverman’s rule of thumb’’ method, which provides appropriately smoothed densities by adjusting the bandwidth based on the sample size n and standard deviation σ . This method is also endorsed by C. Shen and E. Haven in their work, where they employed it to estimate the optimal bandwidth for the amplitude of a pilot wave[18].

Silverman’s rule of thumb

Silverman’s rule of thumb[24] uses a Gaussian distribution to assign a value to the term $\int f''(q)^2 dq$ in the expression 2.9 for the ideal window width. Doing so and by using a normal kernel with $\mu = 0$ and $\sigma^2 = 1$

$$K(y) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{y^2}{2}\right), \quad (2.10)$$

the window width will be[24]

$$b_{\text{opt}} = \left(\frac{4}{3}\right)^{1/5} \sigma n^{-1/5} = 1.06\sigma n^{-1/5}. \quad (2.11)$$

In our model, we simply used the standard deviation (σ) of our log-return data set and then substituted it into Eq. 2.11 to get the right bandwidth. Substituting Eq. 2.10 and Eq. 2.11 into Eq. 2.4 yields the final form of the KDE

$$KDE : \hat{f}_b(q) = \frac{1}{\sigma\sqrt{2\pi}} \left(\frac{3}{4n^4}\right)^{1/5} \sum_{i=1}^n \exp\left(-\left(\frac{3n}{4}\right)^{2/5} \frac{(q - Q_i)^2}{2\sigma^2}\right), \quad (2.12)$$

which is the density estimation formula that we used in our project. The Q_i data points represent the individual log-returns, q is a uniform grid that spans the entire range of the dataset, and σ denotes the standard deviation of the log-returns. The use of Eq. 2.12 produces a well-smoothed curve that more accurately represents the underlying PDF of the dataset compared to a simple Gaussian fit (see Fig. 2.4 for comparison).

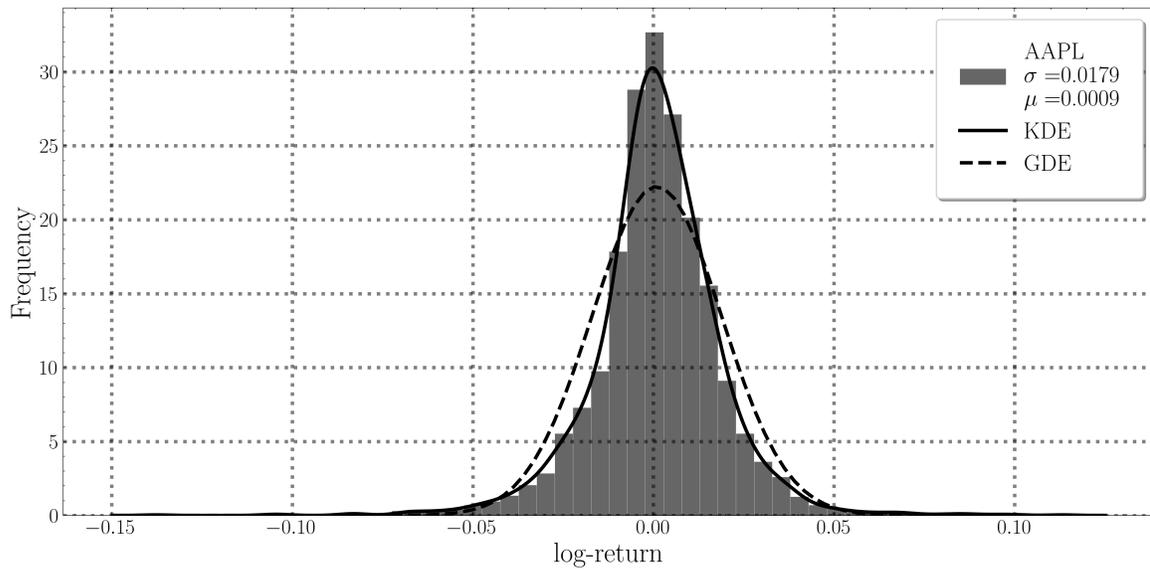


Figure 2.4: Comparison of Gaussian (GDE) and kernel (KDE) estimates of daily log-returns of AAPL from 28/09/2013 to 23/09/2023.

2.2 Symmetric derivatives

Symmetric derivatives are a concept in mathematics, particularly in the study of differential equations and calculus of variations. The purpose of using this approach in our work was to provide a method for finding derivatives without requiring the analytical form of a PDF, such as those estimated using KDE¹. The symmetric derivative of a function f at a point q is defined as

$$f'(q) = \lim_{h \rightarrow 0} \frac{f(q+h) - f(q-h)}{2h},$$

where h is a small increment. This definition differs from the standard derivative in that it uses the average rate of change between $q+h$ and $q-h$ rather than just the rate of change at q .

Given a function f , the second order symmetric derivative at a point x is defined by

$$f''(q) = \lim_{h \rightarrow 0} \frac{f(q+h) - 2f(q) + f(q-h)}{h^2}.$$

¹In this research, we treated histograms as non-continuous, while considering KDE and GDE estimates of PDFs as continuous and differentiable functions.

This definition represents the average concavity of the function over the interval $[q - h, q + h]$. The first symmetric derivative is called Lebesgue's derivative and the second one is Schwarz's derivative[26, 27, 28].

For the purpose of this project and in order to be able to numerically differentiate estimated PDFs, we need to design an algorithm that takes the derivative of any given function made of uniform q and f arrays. The designed algorithms for 1st and 2nd order derivatives that are suitable for our work look like below

$$f'|_{q=q_i} = \frac{\partial f}{\partial q}|_{q=q_i} = \frac{f(q_i + \overline{\delta q}) - f(q_i - \overline{\delta q})}{2\overline{\delta q}}, \quad (2.13)$$

and

$$f''|_{q=q_i} = \frac{\partial^2 f}{\partial q^2}|_{q=q_i} = \frac{f(q_i + \overline{\delta q}) - 2f(q_i) + f(q_i - \overline{\delta q})}{(\overline{\delta q})^2}, \quad (2.14)$$

where $\overline{\delta q}$ is the average spacing between every two grid points, which for a uniform grid is constant, in other words

$$\overline{\delta q} = \delta q_{ij} = q_i - q_j \quad \forall i, j \in [1, n], \quad i \neq j.$$

The algorithm that we designed (see Appendix A, [subsection 5.1.3](#)) takes a 2D array containing q and f values, checks the uniformity of data based on condition above, and computes the 1st or 2nd derivative of the given array².

2.3 Pilot wave theory

In Pilot Wave theory, the wave function acts as a guiding field which influences the paths of particles. The wave function evolves according to the Schrödinger equation, and particles follow deterministic trajectories based on the guidance equation. For a single object in 1 dimensional space with a position representation q , the pilot or

²This method disregards the first and last data points. In our use case, since we employ density estimation methods, the curves remain smooth at the endpoints without any significant jumps or discontinuities. As a result, omitting these points does not lead to any loss of critical information.

guiding wave function ψ looks like[14, 29]

$$\psi(q, t) = R(q, t) \exp\left(i \frac{S(q, t)}{\hbar}\right), \quad (2.15)$$

where S and R are the phase and amplitude of the wave function. $S(q, t)$ is also the solution of the Hamilton-Jacobi equation in classical mechanics and it is called Hamilton's principal function[29].

The wave function 2.15 is the solution of time-dependent Schrödinger partial differential equation

$$i\hbar \frac{\partial \psi}{\partial t} = -\frac{\hbar^2}{2m} \frac{\partial^2 \psi}{\partial q^2} + V(q, t)\psi(q, t), \quad (2.16)$$

where \hbar is the Planck constant, $V(q, t)$ represents the real potential, and m is mass. Substituting Eq. 2.15 into Eq. 2.16 yields a system of real and imaginary field equations. After separating real and imaginary parts we obtain ³

$$\frac{\partial R^2}{\partial t} + \frac{1}{m} \frac{\partial}{\partial q} \left(R^2 \frac{\partial S}{\partial q} \right) = 0, \quad (2.17)$$

$$\frac{\partial S}{\partial t} + \frac{1}{2m} \left(\frac{\partial S}{\partial q} \right)^2 + \left(V - \frac{\hbar^2}{2mR} \frac{\partial^2 R}{\partial q^2} \right) = 0, \quad (2.18)$$

where R and S are the real valued amplitude and phase parameters of the wave function. In classical events when $\frac{\hbar^2}{2m} \rightarrow 0$, the term $S(q)$ will be the solution of classic Hamilton-Jacobi equation. If we consider an ensemble of particle trajectories, i.e. sequences of log-returns, which are solutions of the equations of motion, if all of these trajectories are normal to any given surface of constant $S(q)$, then they are normal to all surfaces of constant $S(q)$, and $\nabla S(q)/m$ will represent the velocity, $v(q)$, for any particle passing the point q . In classical mechanics, these paths converge into a single trajectory, where Newton's laws of motion apply.

³Refer to [30, 29] for more detailed derivations.

2.3.1 Quantum potential

Equation 2.17 illustrates how the amplitude $R^2 = |\psi|^2$ of the quantum field evolves over time. Knowing initial state of the wave function ($\psi(q, 0)$) is necessary for determining $R(q, t)$ and $S(q, t)$, though we will not cover the analytical solutions of Hamilton-Jacobi equation in this project.

Equation 2.18 is the Hamilton-Jacobi equation in quantum framework. The last term in Eq. 2.18 is called the quantum potential and it is defined as

$$Q(q, t) = \frac{\hbar^2}{2m} \frac{\nabla^2 R}{R}. \quad (2.19)$$

Note that in classical limits the quantum potential vanishes and we achieve the classic Hamilton-Jacobi equation.

Based on Bohm, 1952[14], it is also convenient to write $R^2(q) = f(q)$, or $R = f^{\frac{1}{2}}$ where $f(q)$ is the true or estimated probability density. This is because the amplitude of a pilot wave is equivalent to $R^2 = |\psi|^2$, meaning that R^2 can be regarded as a probability density. Using this analogy we used the probability density of log-returns as R^2 . To fit the PDF in Eq. 2.19 and simplify our calculations, we substituted R with \sqrt{f} and assumed $\frac{\hbar^2}{2m} = 1$. Equation 2.19 was then simplified as below:

$$Q(q) = \frac{1}{\sqrt{f(q)}} \frac{\partial^2 \sqrt{f(q)}}{\partial q^2}. \quad (2.20)$$

Equation 2.20 will be used frequently and we might refer to it as quantum potential or market potential.

2.3.2 Quantum force

Rearranging Eq. 2.18 we can obtain

$$\partial S / \partial t + (\nabla S)^2 / 2m = -(-Q + V). \quad (2.21)$$

Given the classical and quantum field functions, we can determine the force acting on a particle. With the knowledge of particle's initial position and momentum, this allows us to calculate its complete trajectory. Taking the gradient of Eq. 2.21 from

both sides gives

$$(\partial/\partial t + (1/m)\nabla S \cdot \nabla) \nabla S = -\nabla(-Q + V), \quad (2.22)$$

and by substituting $\nabla S/m = \dot{q}$ [29] we get the following equation of motion for a particle under the influence of both the classical potential $V(q)$ and the quantum potential $Q(q)$

$$\frac{d}{dt}(m\dot{q}) = -\nabla(-Q + V), \quad (2.23)$$

where

$$d/dt = \partial/\partial t + \dot{q} \cdot \nabla \quad (2.24)$$

describes the rate of change over time relative to a point moving along with the particle.

Equation 2.23 takes the form of Newton's second law, indicating that the particle is influenced by both a quantum force $\vec{F}_Q = -\nabla Q$ and other classical forces $\vec{F}_V = -\nabla V$ [29]. This indicates that the force experienced by an object is equal to the negative gradient of the potential energy function. In simpler terms, it means that the force points in the direction of the steepest decrease in potential energy. From now on, we shall refer to \vec{F}_Q as quantum force or market force. Equation 2.23 will now take the form

$$\frac{d}{dt}(m\dot{q}) = F_V - F_Q, \quad (2.25)$$

which differs somewhat from David Bohm's explanation⁴. The decision to take the negative sign outside of the quantum potential and quantum force is rooted in the interpretations of our model, which will be detailed in the following section.

2.3.3 Bohmian mechanics in finance

In this section, we will explore how to model a financial time series within the framework of Bohmian Mechanics. The majority of the assumptions and general concepts are drawn from the works of Tahmasebi et al[17]. and Shen et al[18].

Consider a financial asset such as TSLA. When we observe the market tomorrow,

⁴In Bohm's paper (see [14, 15]), the quantum potential is defined as $Q(q, t) = -\frac{\hbar^2}{2m} \frac{\nabla^2 R}{R}$, differing from our version of the quantum potential by a negative sign.

the return on this asset, compared to the previous day, could vary by a fraction. However, suppose that tomorrow's return is somehow influenced by the returns of the past N days (N is called the “window” size). In this case, if we construct the probability density function (PDF) of the log-returns over the past N days, as demonstrated in section 2.1, tomorrow's log-return is likely to follow the same distribution. In other words, similar to the concept in Bohmian Mechanics, there exists an ensemble of log-return configurations described by a probability distribution $P = R^2 = f(q)$, where f represents the PDF (or its estimate) of the log-returns. There are several reasons to why we use the quantum pilot wave to address a financial asset and not a classical wave. Some studies suggest that the pilot wave function can be viewed as a wave function carrying information [30, 13]. Additionally, as we will see in the next section, the quantum force derived from a pilot wave is independent of the wave's amplitude and instead depends on the field itself and its characteristics, such as mean and standard deviation. Also, an essential characteristic of a quantum potential is its role in giving rise to non-locality. This non-locality arises from the fact that the pilot wave, which directs the motion of particles, is defined in configuration space, a space that encompasses all possible positions of every particle in the system. Consequently, if we extend our model to include multiple financial assets, represented by an expanded pilot wave function $\psi(q_1, q_2, \dots, q_M, t)$, even a minor change in the price of one asset can immediately influence the trend of another asset elsewhere[31]. This is because the trajectories of the assets are governed by the wave function's global structure. Such behavior is frequently observed in economics; for example, during the global financial crisis of 2008, the collapse of the housing market in the United States triggered a worldwide economic downturn. These observations lead to the logical conclusion that the pilot wave is, in fact, an information wave. Based on these interpretations, we can construct the quantum potential and quantum force for the log-returns. For Gaussian estimation of log-returns, where $R^2(q, t) = f_G(q, t)$, the quantum potential and quantum force can be derived analytically using Equations 2.1 and 2.20. Therefore, the quantum potential $Q_G(q, t)$ and the market force $F_{Q_G}(q, t)$ for a Normal distribution $\mathcal{N}(\mu, \sigma^2)$ are given by

$$Q_G(q, t) = \frac{(q - \mu)^2}{4\sigma^4} - \frac{1}{2\sigma^2}, \quad (2.26)$$

and

$$F_{QG}(q, t) = -\frac{\partial Q_G}{\partial q} = -\frac{q - \mu}{2\sigma^4}, \quad (2.27)$$

where q represents the logarithmic price return at time t , and μ and σ are the mean and standard deviation of the log-returns over the past N days. For kernel estimation of log-returns, numerical methods, such as Schwarz's derivative, provide an efficient way to calculate the corresponding potential and force. A more detailed algorithm will be presented in the next chapter. Examining Q_G reveals a parabolic shape, indicating the presence of an equilibrium state where the system's quantum potential energy is minimized. For a normal distribution, this equilibrium point corresponds to the mean, μ . Additionally, the market force F_{QG} is linear, taking a zero value precisely at this mean. This suggests that the expected log-return corresponds to a state of equilibrium in the system, where the energy is at its lowest possible level, and no net quantum force (in this case market force) is acting on it. Essentially, this is the point where the system is most stable, with forces balanced and energy minimized. Further exploration and interpretation of these findings are discussed in Chapter 4. These characteristics and interpretations are a result of moving the negative sign outside the quantum potential. Without this minor modification, stable equilibrium points and constraining walls for quantum potentials would not exist. Additionally, in Chapter 3, we will explore how this method aids in extracting and interpreting the features of quantum potentials and quantum forces. Note that the primary aim of this research has been to investigate the characteristics of quantum potentials and quantum forces within financial time series. The general framework of Bohmian mechanics remains undeveloped for financial markets, with numerous undefined parameters and constants (such as the classical potential V , \hbar , and m) that would need specification to establish a comprehensive Bohmian mechanics theory for financial time series. At this stage, it is more practical to accept a few necessary simplifications to achieve the study's objectives.

2.4 Sliding window technique

A time series can represent financial data or physical metrics, such as temperature or pressure, for a machine over a period of time. One approach to studying these metrics is to apply mathematical or statistical calculations (mean, standard deviation,

etc.) to the entire month’s data, yielding a single, fixed result. Alternatively, we can analyze a small time window, such as the first week, and calculate the metrics for that period. We then move the window forward by one day, gather another set of metrics, and repeat this process until we reach the end of the month. By examining these results, we can create a new time series showing the evolution of the metrics over the month. This technique, which enables the study of a system’s time evolution using any set of metrics, is known as the “sliding window technique”. The metrics that are used to smooth data or identify trends and patterns are called “technical indicators”.

To demonstrate how technical indicators and the sliding window technique work, we’ll use one of the basic indicators: the simple moving average (SMA). In financial analysis, the SMA represents the unweighted average of the last k data points. Considering a sequence of price data $P = [p_1, p_2, \dots, p_N]$, the SMA indicator calculates the average across a fixed number of consecutive points, known as the window size k . For the SMA at position i , the window $W_i = [p_i, p_{i+1}, \dots, p_{i+k-1}]$ represents the values included in the calculation. Mathematically, the SMA for each window is computed as

$$\text{SMA}_{i+1}^k(p) = \frac{1}{k} \sum_{j=i}^{i+k-1} p_j. \quad (2.28)$$

To efficiently compute the SMA across the sequence, we utilize the sliding window technique, which reduces redundant calculations as the window shifts. For each new window W_{i+1} , we adjust the previous sum by subtracting the element p_i that is leaving the window and adding the new element p_{i+k} that is entering it. This allows us to compute the SMA as follows

$$\text{SMA}_{i+1}^k(p) = \text{SMA}_i^k(p) - \frac{p_i}{k} + \frac{p_{i+k}}{k}. \quad (2.29)$$

By applying this incremental update, the SMA for each new window is calculated in constant time, $O(1)$, rather than recalculating the full sum for each window, which would be $O(k)$. This technique demonstrates how sliding windows enhance computational efficiency, particularly in time series analysis and real-time applications where quick updates to the moving average are crucial.

For N sequence of price data and the window size $k = 3$, the SMA indicator will

look like

$$\text{SMA}_1^3(p), \text{SMA}_2^3(p), \dots, \text{SMA}_{N-3}^3(p), \text{SMA}_{N-2}^3(p),$$

where the last window is

$$\text{SMA}_{N-2}^3(p) = \frac{1}{3} \sum_{j=N-2}^N p_j.$$

A visual representation of this indicator and the sliding window technique is shown in Fig. 2.5.

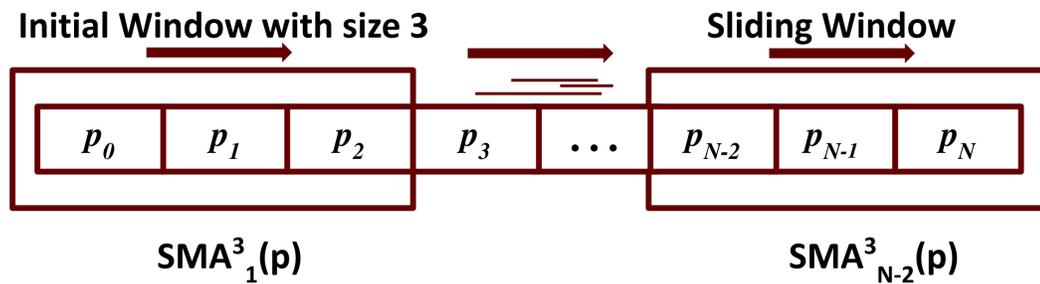


Figure 2.5: Sliding Window Technique for Making Technical Indicators (Example: Simple Moving Average with Window Size 3)

Throughout the following chapter we will select a number of famous indicators, in addition to SMA, in order to study the time evolution of closing prices, just as represented on Fig. 2.5. These set of indicators include: relative strength index (RSI) and moving average convergence/divergence (MACD). The Relative Strength Index (RSI) gauges the speed and magnitude of price changes, evaluating the recent gains of a security against its losses over a given period, usually 14 days. The RSI ranges from 0 to 100. Based on empirical facts in technical analysis, values above 70 often signal that the security may be overbought and could face a price decline, whereas values below 30 suggest that it might be oversold and potentially due for a price increase. The Moving Average Convergence Divergence (MACD) is another key trend-following tool in technical analysis that helps identify shifts in a trend's strength, direction, and duration. The MACD consists of two lines: the MACD line, which is the difference between the 12-day and 26-day exponential moving averages (EMAs), and the signal line, which is the 9-day EMA of the MACD line. A cross of the MACD line above the

signal line indicates a upward price movement, while a cross below signals a downward movement.

In what follows we will calculate a few statistical measures such as mean, standard deviation, skewness, and kurtosis associated with the daily log-returns within the same window as for technical indicators. The most exciting part of the next chapter is when we create our own indicators using the features extracted from market potentials and market forces. Finally we will analyze the correlations of our new indicators with other technical and statistical ones and show the results in a correlation matrix. Given the insights from the correlation matrix and the understanding that the selected technical and statistical indicators capture essential market characteristics (such as volatility, trend, and reversal) we can uncover the role of quantum potentials, quantum forces, and their associated indicators in studying financial markets.

Chapter 3

Model and results

This chapter outlines our methodology and procedures for generating quantum potentials of log-returns and analyzing their time evolution using the sliding window technique. Each section details the methods and algorithms utilized to achieve this objective. After thoroughly analyzing the results and extracting features from these new structures, we will present our conclusions and highlight the key points of this project.

3.1 Price time series

The initial step in studying a financial asset involves obtaining its real-time or historical price time series. Financial markets consist of various assets, including stocks, options, futures contracts, bonds, and currencies. A broker facilitates the trading of these assets. When a buyer and a seller agree on a price, the transaction is recorded in the broker's system, and the asset and money are transferred between traders. These transactions are updated within fractions of a second and are visible to all, providing real-time information on the asset's value. Since these data are recorded continuously, they form a time series. Each component of a time series provides a detailed record of every transaction (or "tick"), including the price, volume, and timestamp of each trade. If this data is not filtered, it is referred to as "tick data".

Tick data offers the highest resolution of financial information, ensuring that every single trade is captured in the dataset. This unprocessed and raw form of data

allows traders and analysts to apply their own filtering methodologies. Due to the challenges of recording and maintaining such detailed and extensive datasets, tick data is typically expensive to obtain and requires substantial computing resources for pre-processing. A sample of this type of raw data is shown on Table 3.1. These are the information of every single trade within a single day (12/05/2023) on Microsoft’s stock shares. Each row presents information about a single trade that occurred at a

2023-05-12 04:00:00.004,309.77,200,1,310.41,200,1
2023-05-12 04:00:00.156,309.8,200,1,310.41,200,1
2023-05-12 04:00:00.696,309.8,200,1,310.8,200,8
2023-05-12 04:00:01.008,309.8,200,1,310.37,200,1
2023-05-12 04:00:02.518,309.8,100,1,310.26,100,1
2023-05-12 04:00:02.518,309.8,100,1,310.06,100,1
.
.
.
2023-05-12 19:59:15.384,308.29,400,7,308.37,700,8
2023-05-12 19:59:35.601,308.29,400,7,308.37,600,8

Table 3.1: Tick data sample, Microsoft.

specific point in time. The first two columns indicate the date and time of each trade, while the remaining columns contain ambiguous data that requires classification and filtering.

To effectively employ computational techniques, it is crucial to filter the tick data, extract important information, and arrange these extractions in a consistent format. Ideally, the data should be structured in a form commonly referred to as “bars”. Bars are created by filtering and sampling the raw tick data according to time, volume, or other specific criteria. While several bar construction techniques are widely used in finance, we’ll focus on the most straightforward type known as “time bars”. These bars are generated by sampling data at consistent time intervals, such as once per minute. Generally, the gathered data usually includes information such as[32]:

- date or timestamp,
- opening and closing price,
- highest and lowest price,

- volume-weighted average price (VWAP),
- and other related information such as traded volume, etc.

Date	Open	High	Low	Close	VWAP
2023-05-12	310.549988	310.649994	306.600006	308.970001	306.454865

Table 3.2: Daily bar data sample, Microsoft.

Table 3.2 shows the filtered and structured daily bar data for the same day and asset as presented in Table 3.1. By comparing Table 3.2 with Table 3.1, it is clear that bar data provides a more organized and manageable format, though it comes at some cost. Firstly, time bars naturally oversample information during periods of low activity and undersample during periods of high activity, as the number of recorded activities in a high-activity minute differs from that in a low-activity period. Additionally, time-sampled series often display poor statistical properties, such as serial correlation[32]. For more simplicity in our work, from now on, we rely only on the closing prices of daily time bars.

3.1.1 Data access

Tick and bar data can be accessed through various Application Programming Interfaces (APIs). These APIs may be provided either for free or for a fee, depending on the provider and the quality of the data. To access daily bar data for various securities, we used a free Python library called `yfinance`, which connects to the **yahoo!finance** API (see Appendix A, subsection 5.1.1). We selected Tesla and Bitcoin for their high volatility, while S&P500 ETF Trust and Johnson&Johnson were chosen for their stability, exhibiting low volatility. Figure 3.1 illustrates the historical daily closing prices for the selected securities over nearly two years. The horizontal axis represents time t , while the vertical axis denotes the price p . Given the daily time frame, we set the time step τ to 1 day. Although intraday data, such as hourly or minute intervals, can be accessed through various APIs, we opted for the daily format due to its widespread use among traders and researchers. As discussed, the function $p(t)$ will form the basis of our model. In the following sections, we will demonstrate how to estimate the probability density of price changes using Eq. 1.1 and curve estimation methods like Gaussian density estimation (GDE) and kernel density estimation

(KDE) (see [subsection 2.1.1](#) and [subsection 2.1.3](#)).

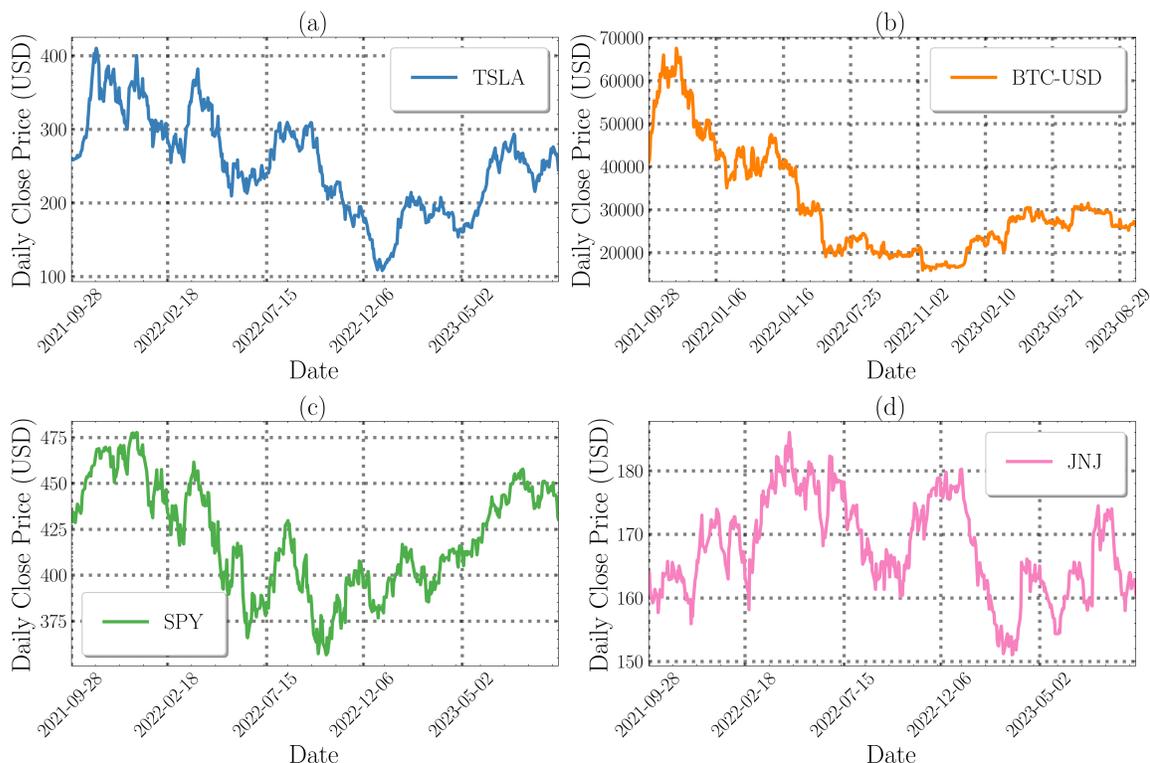


Figure 3.1: Daily close prices of (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023

3.2 Estimating probability density functions

After obtaining the daily close prices, we calculate the price changes of all four time series and create their histograms as initial representations of the distributions. To do this we use Eq. 1.1 and determine the logarithmic price change for each day compared to the previous day, then we organize this data into bins to construct our histograms. Figure 3.2 shows the obtained histograms of log-returns for the chosen financial assets. As the plots clearly show, the width of each histogram is directly related to the standard deviation σ of log-returns.

The data related to a histogram is typically not differentiable. A histogram represents discrete data by counting occurrences within specified bins or intervals. The resulting bar heights indicate the frequency of data points within each bin, forming

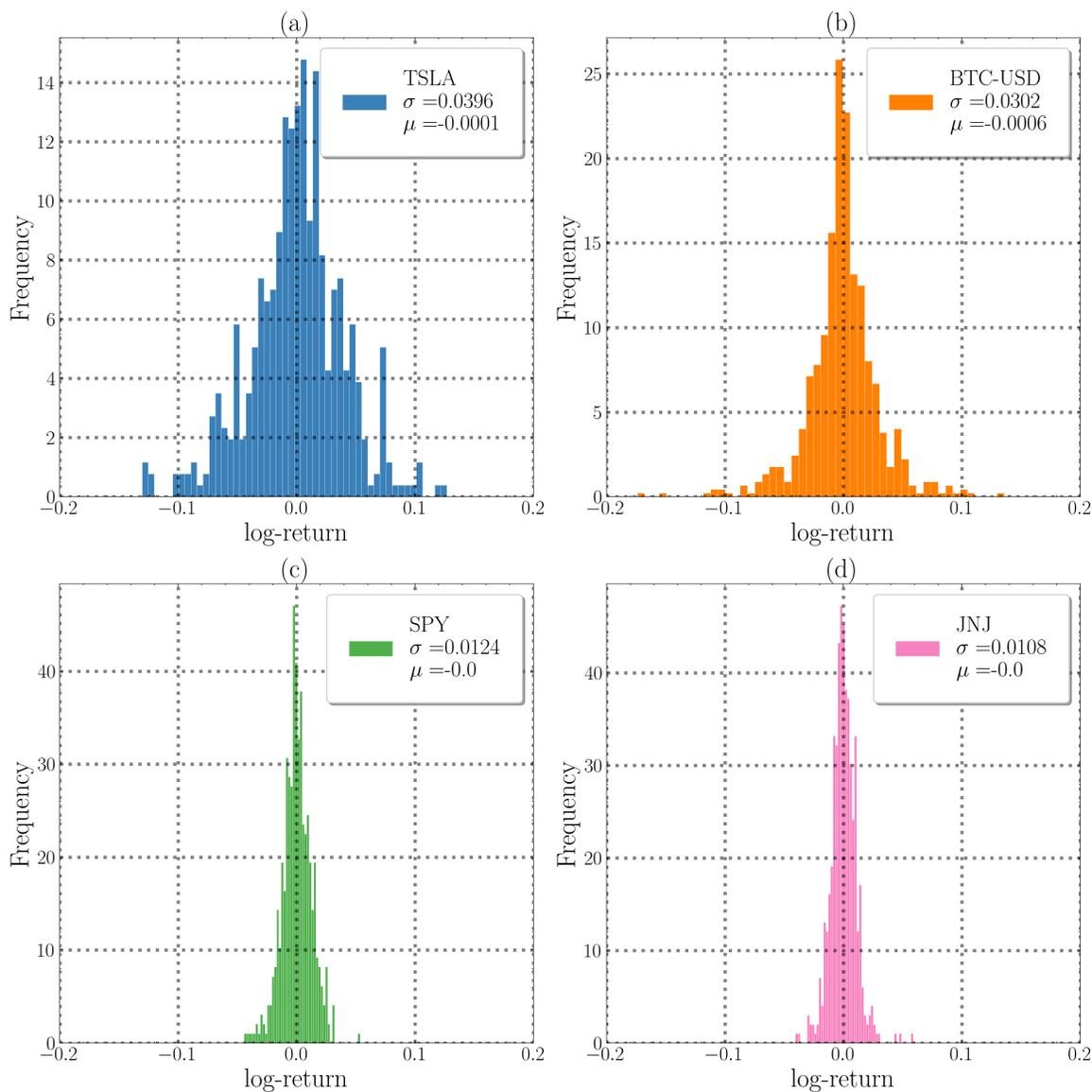


Figure 3.2: Distribution of log-returns of daily close prices of (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023

a step-like function. Differentiability requires a smooth, continuous function where derivatives exist at every point within its domain.

As we discussed in previous chapter, we used two methods for finding the associated PDFs with the histograms of log-returns.

3.2.1 GDE

To estimate a Gaussian distribution for the log-returns, we follow the approach outlined in [subsection 2.1.1](#). We developed a function that performs this task by accepting the log-returns of each time series as input. This function outputs a `dataframe` object containing the values of q_{uniform} and f_G . The q_{uniform} represents uniformly generated data points based on the original log-returns q . By using these uniform log-returns along with the parameters `mu` and `sigma`, which are derived from the original q data, we apply them to a Gaussian function (defined as Eq. 2.1) and get the result in a `dataframe` format (see the source code in Appendix A, [subsection 5.2.1](#) for more detail).

3.2.2 KDE

Following from the description of KDE method in [subsection 2.1.3](#), to perform kernel estimation on the log-returns, we utilized the `seaborn.kdeplot()` function, which takes an array of log-returns as input, generates $\mathcal{N}(0, 1)$ distributions at each q point, and sums them to form the final distribution curve. We specified `"silverman"` for the `bw_method` and set `common_norm` to `True`. These parameters ensure a smooth and normalized probability density function (PDF). Once the data points for the estimated PDF are generated, they can be stored in a `pandas` dataframe for future use (see the source code in Appendix A, [subsection 5.2.2](#) for more detail).

Figure 3.3 shows the estimated PDFs of the original histograms presented in Fig. 3.2. These continuous curves, plotted on a uniform grid along the log-return axis, facilitate numerical differentiation. The differences between the Gaussian estimate (dashed curve) and the kernel estimate (solid curve) are clear. Notably, the kernel estimate responds better to extreme events, as seen in the tails of the TSLA data. Additionally, the KDE method more accurately captures the highs and lows without over-smoothing the data, unlike the GDE. These plots effectively highlight the differences between fat-tailed distributions and Gaussian ones. They provide enough evidence for why log-returns cannot be assumed to follow a normal distribution. These differences have also been pointed out by other researchers such as Benoit Mandelbrot[8]. In the next section, we will further explore the differences between these two types of PDFs by analyzing their quantum potentials and quantum forces.

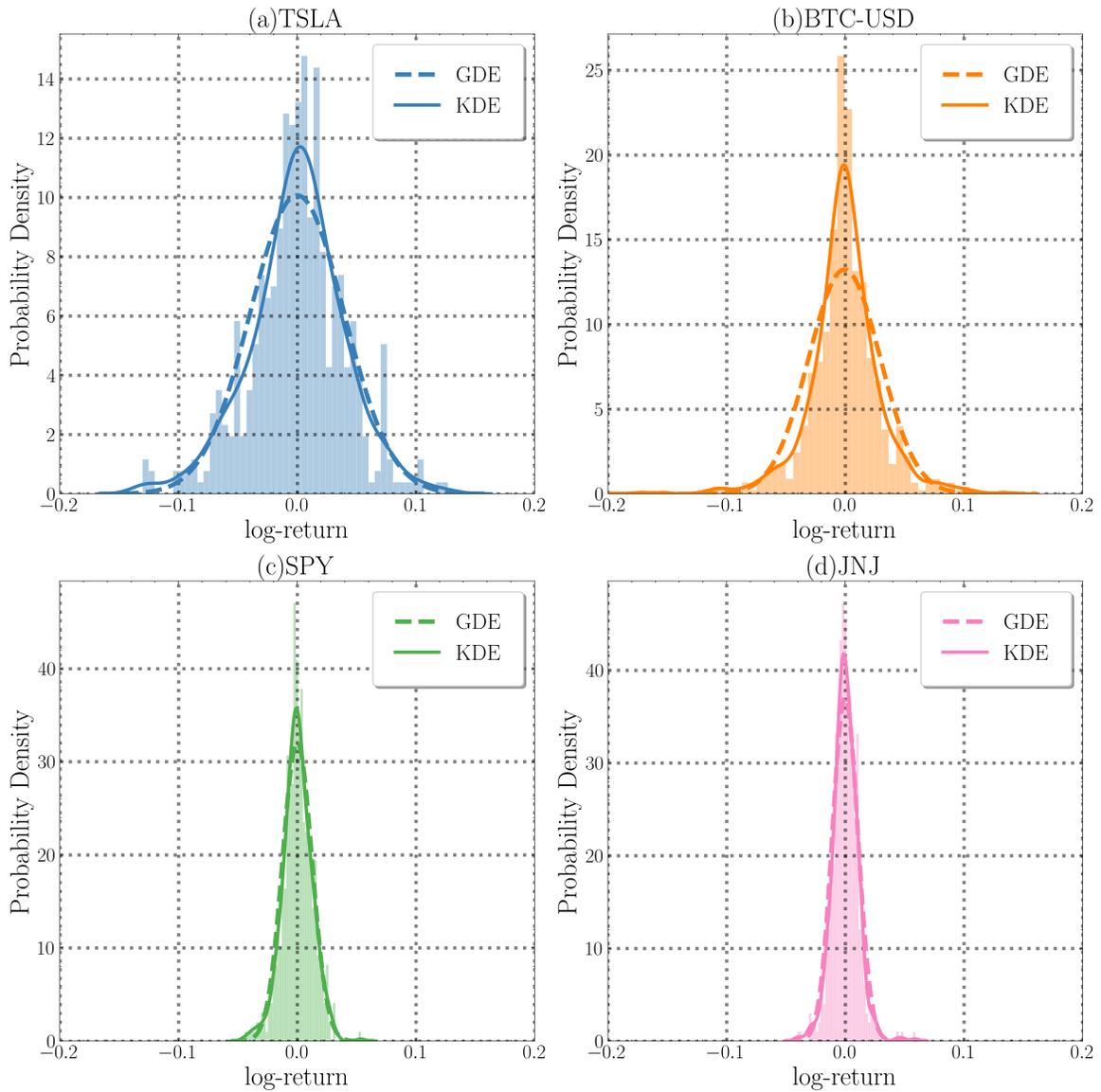


Figure 3.3: Gaussian density estimation (GDE) and kernel density estimation (KDE) of log-returns of daily close prices for (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023

3.3 Quantum potentials

In this section, we demonstrate how the quantum potentials of log-returns are constructed using pilot wave theory and other numerical methods discussed in earlier sections. As outlined in [subsection 2.3.1](#) and [subsection 2.3.3](#), quantum potentials

represent the likelihood of certain log-returns occurring based on energy terms. According to the reformulated definition of quantum potentials, to obtain the quantum potential of an estimated PDF, we simply need to substitute the estimated f functions from Fig. 3.3 into the reshaped format of quantum potentials as Eq. 2.20. We can create a function to handle the numerical derivatives of f , following the approach described in section 2.2. For a distribution estimated using a Gaussian function, we can utilize the analytical form of the PDF to develop a Python function based on Eq. 2.26 and apply it to the q_{uniform} data points (for more details, see the source code in Appendix A, subsection 5.2.1 and subsection 5.2.2).

Figure 3.4 shows quantum potentials related to GDE and KDE fits of daily log-returns for the same collection of symbols. As before, the dashed and solid lines show the potential made from Gaussian and kernel density estimates. As we can see, the quantum potentials made from KDE are very different than Gaussian ones. The Gaussian quantum potential is simply a parabola with a minimum at average log-return (μ). Derived from Eq. 2.26 we get

$$Q_G|_{q=\mu} = \frac{(\mu - \mu)^2}{4\sigma^4} - \frac{1}{2\sigma^2} = -\frac{1}{2\sigma^2},$$

which is the expected value of a Gaussian quantum potential at point $q = \mu$.

The domain of the quantum potentials driven from KDE distributions, appearing as walls, are related to the range of motion of every asset at the moment, or well know as “volatility”. Any log-return outside these walls are very unlikely to happen. These plots clearly illustrate why almost no one anticipates massive returns from the market. For instance, if the walls of a quantum potential are situated at -0.05 and 0.05, they set the boundaries for an investor’s risk and potential return, meaning it is unrealistic to expect a return of 0.1 or a loss of 0.15.

Unlike Gaussian potentials, there are more than one local minimum points in non-Gaussian quantum potentials, indicating multiple equilibrium states. In other words, in a real market, traders don’t want the market to stay still. This creates other equilibrium states other than average return that are really important. Whenever an asset falls in one of its equilibrium points, it is unlikely to leave that state unless there is a significant market movement. In such a scenario, the asset is likely to shift to the nearest equilibrium state in the direction of the market. Unlike the Gaussian model, which predicts a stable, converging market with expected return of μ , a more

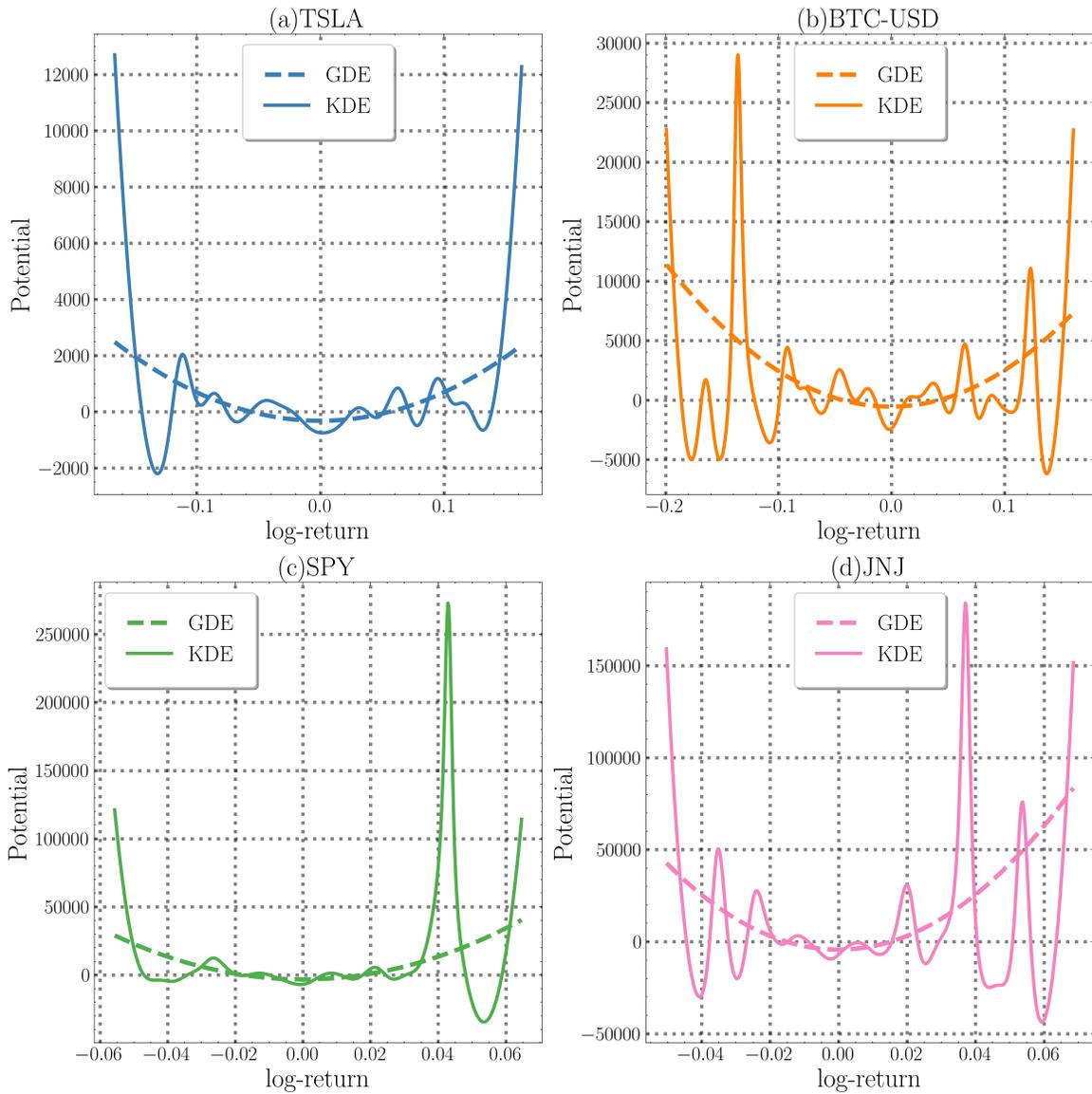


Figure 3.4: GDE and KDE quantum potentials of log-returns of daily close prices for (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023

realistic model of financial markets suggests a dynamic system with multiple equilibrium states, which are not apparent in density functions. These multiple equilibrium points could provide profitable opportunities for investors. This property of quantum potentials contradicts the principles of the “efficient market hypothesis” or “EMH”. It is important to note that these equilibrium points should not be mistaken for any noise arising from our numerical methods. Readers can easily compare the numerical

and analytical derivatives of the Gaussian PDF and observe a very high degree of agreement. In other words, no noise was detected from our numerical methods.

Another key characteristic of KDE quantum potentials is their ability to emphasize data in the tails. The deep minima located at the ends of the quantum potentials represent extreme events in fat-tailed distributions. Due to their significant depth, these minima serve as prominent equilibrium points. Thus, overlooking these states would be a critical error in any financial model. We will further explore these behaviors in the following sections.

3.4 Market forces

As described in [subsection 2.3.2](#), the quantum force F_Q is defined as the negative gradient of the potential, indicating that the force directs the system towards the steepest decrease in potential energy. This implies that the force is zero at local minima of the potential, which correspond to equilibrium states. The market force drives the market toward these equilibrium states. From the expression for the quantum force of a Gaussian system in [Eq. 2.27](#), we can observe that

$$F_{QG}|_{q=\mu} = -\frac{\mu - \mu}{2\sigma^4} = 0,$$

indicating that the market force for a Gaussian quantum potential is a flat line, with a value of zero at the average log-return (μ). This further suggests that a Gaussian model aligns with EMH, predicting a single expected return for market participants. In a fair game (normal system), the dynamics of the market push players towards a single equilibrium state, which is the average return, meaning no one can outperform the average. Creating a python function that constructs market forces is quite easy. All we did was to feed the `dataframes` containing the uniform log-returns and quantum potential values to our first order symmetric derivative function and multiply it by -1 (see the source code in [Appendix A, subsection 5.3.5](#) for more detail).

[Figure 3.5](#) illustrates the market forces derived from the quantum potentials shown in [Fig. 3.4](#). These plots clearly demonstrate that a real market is more dynamic and complex than a normal system. It is evident that multiple forces influence the market. Unlike the Gaussian model, which results in a single equilibrium point with zero force,

in a more realistic model, the force crosses the zero line multiple times, indicating multiple minima in the quantum potential plots. An interesting feature to consider is the value of the quantum force at $q = \mu$. Comparing this with the expected value from Gaussian model ($F_Q(\mu) = 0$) could provide insights into the fairness of the market and measure deviations from the Gaussian model. Specifically, if $|F_Q(\mu)| \gg 0$, it suggests that the market is unlikely to yield no returns.

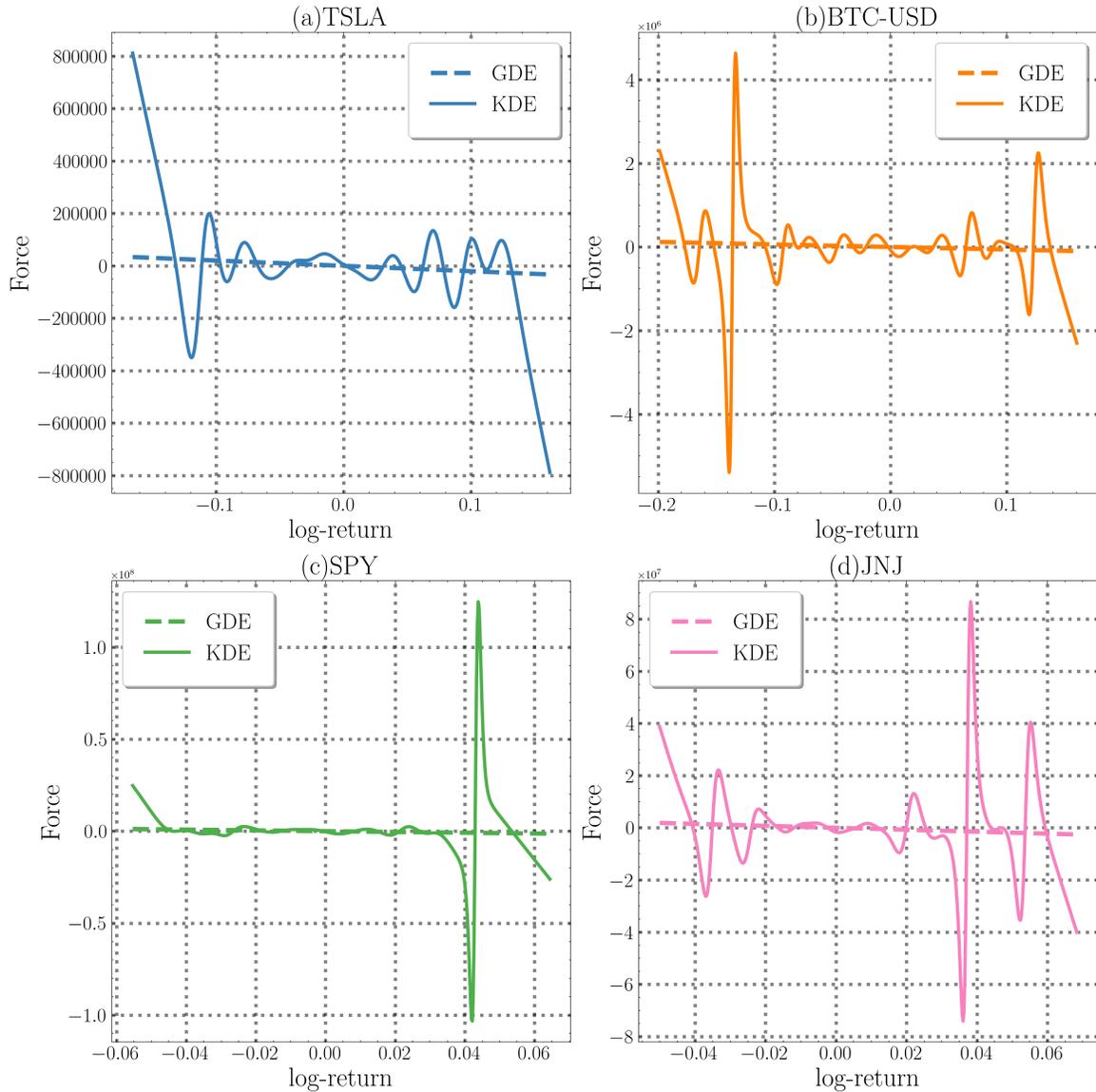


Figure 3.5: GDE VS KDE market force comparison, generated from log-returns of daily close prices for (a) Tesla, (b) Bitcoin, (c) S&P500 ETF Trust, and (d) Johnson&Johnson from 28/09/2021 to 23/09/2023

3.5 Multi-time frame analysis

Studying markets in various time frames provides valuable insights into both the short-term and long-term behavior of an asset. Financial markets usually start to

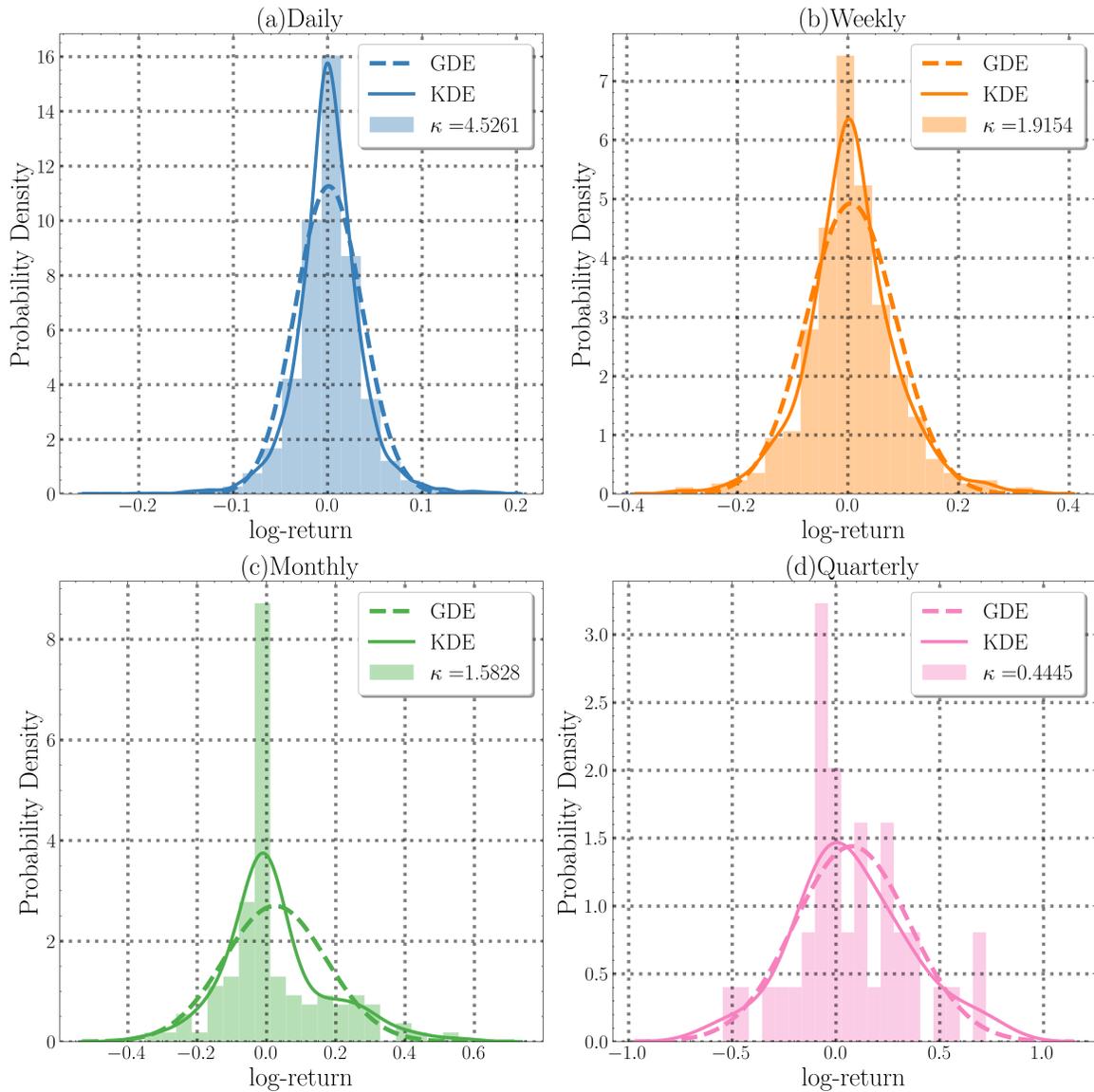


Figure 3.6: Gaussian and kernel density estimations for the log-returns of Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)

act like a Gaussian system when studied in longer time frames. In this section, we examine the quantum potential and quantum force of log-returns across daily, weekly, monthly, and quarterly time frames to analyze in detail the deviations of our selected

assets from normal systems. We select Tesla and S&P500 ETF Trust for this purpose.

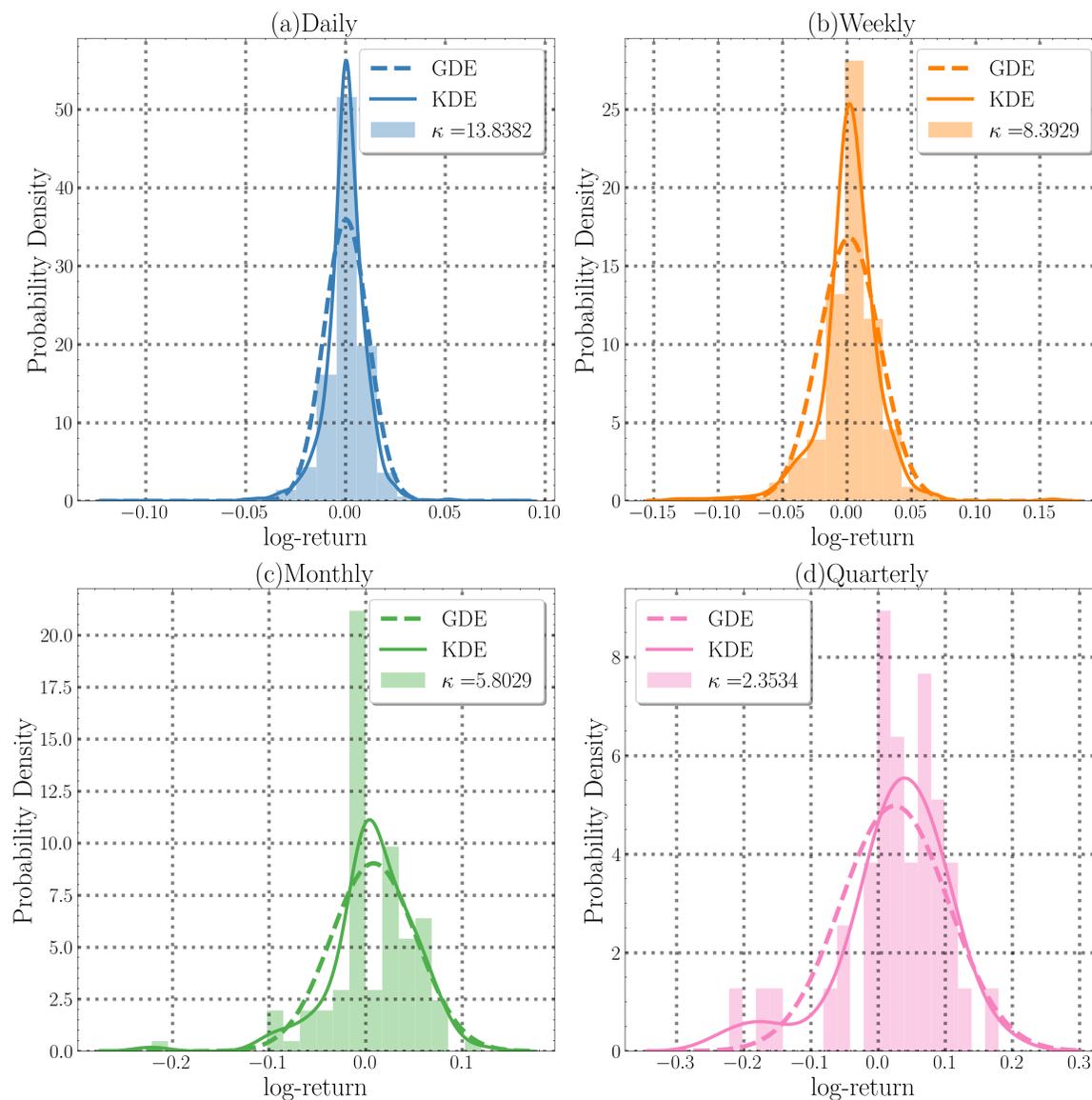


Figure 3.7: Gaussian and kernel density estimations for the log-returns of S&P500 ETF Trust in multiple time frames (from 28/09/2013 to 23/09/2023)

The initial observation of probability density functions across various time frames in Fig. 3.6 and Fig. 3.7 indicates that the kurtosis κ tends to decrease as the time frame lengthens¹ (note the values of κ for different time frames). In statistical terms,

¹Note that the size of the datasets shrinks for longer time frames. When performing multi-time frame analysis for a fixed time window, a reduction in granularity and dataset size can pose challenges. But in technical analysis, it is widely recognized that gaining long-term insights often requires sacrificing precision, and vice versa.

this means that the “tailedness” of the distributions diminishes over greater time frames, implying that extreme values or outliers become less frequent. As a result, the kernel estimates of these distributions begin to resemble Gaussian distributions more closely, characterized by their bell-shaped curve and lack of heavy tails. This

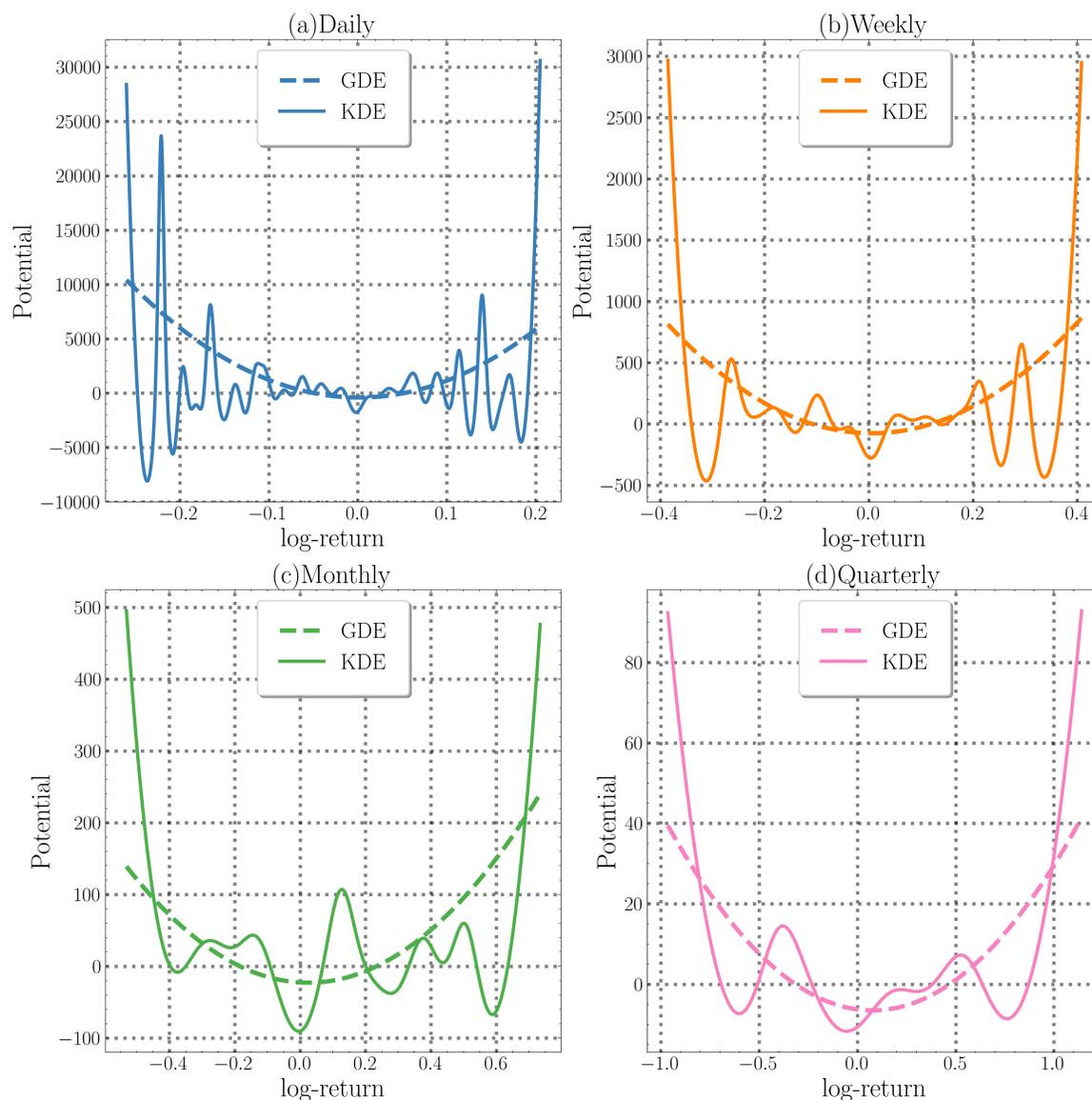


Figure 3.8: Quantum potentials of Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)

trend suggests that longer time frames provide more stable and predictable data patterns, where random fluctuations and outliers exert less influence on the overall distribution. These figures also show that the estimated PDFs become increasingly

skewed towards positive log-returns over longer time frames. This trend suggests the potential for greater returns for long-term investors. However, it is important to note that this may not apply to every asset, especially those that have historically proven to be unprofitable.

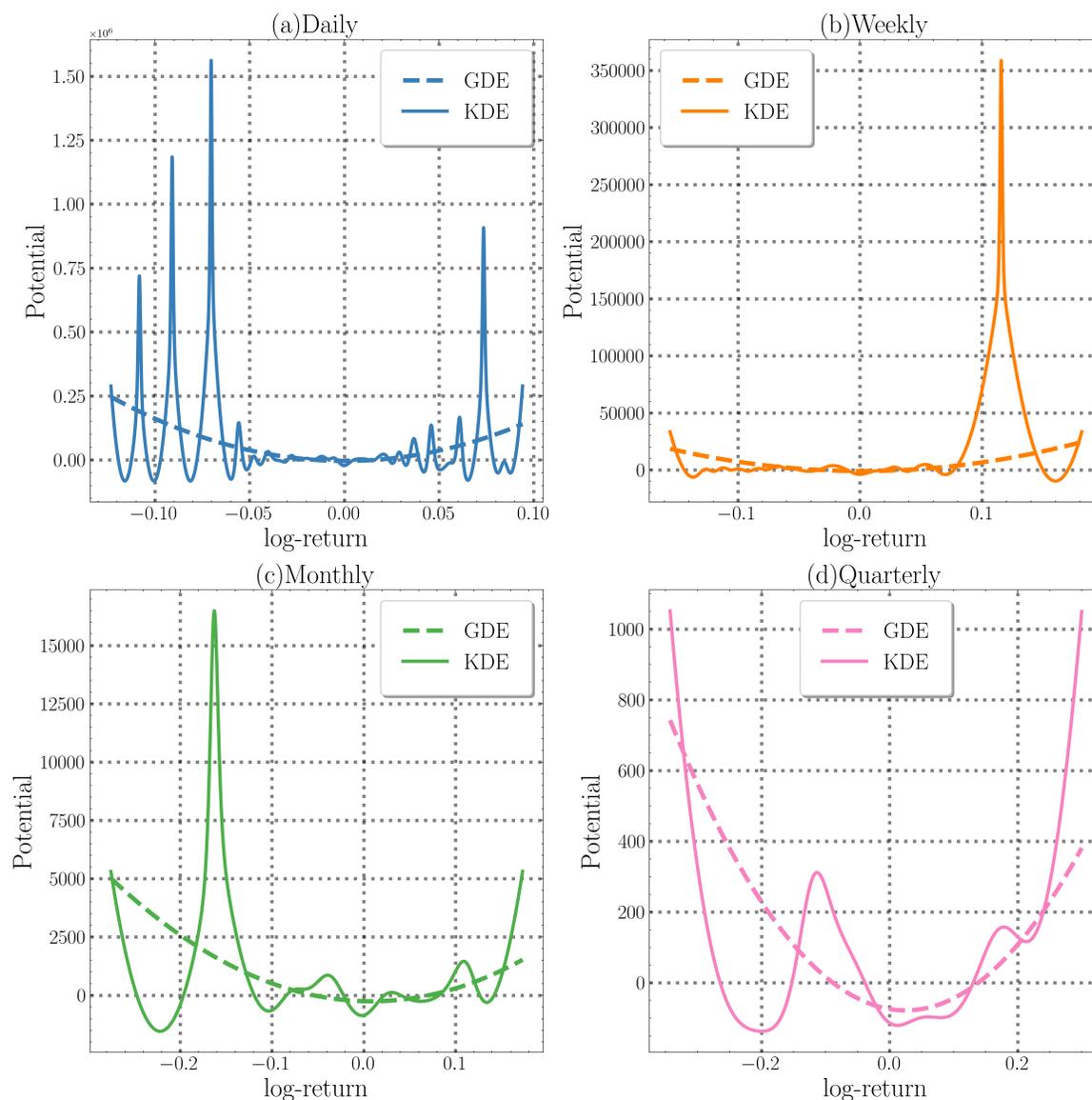


Figure 3.9: Quantum potentials of S&P500 ETF Trust in multiple time frames (from 28/09/2013 to 23/09/2023)

Figures 3.8 and 3.9 illustrate the structural differences in quantum potentials across various time frames. Similar to the probability density functions, the quantum potentials derived from KDE method tend to resemble those formed from Gaussian

distributions. Notably, for larger time frames, the quantum potentials exhibit a simpler structure. This indicates that, over the long term, quantum potentials converge to a fewer number of equilibrium points that are more dominant. Based on these two

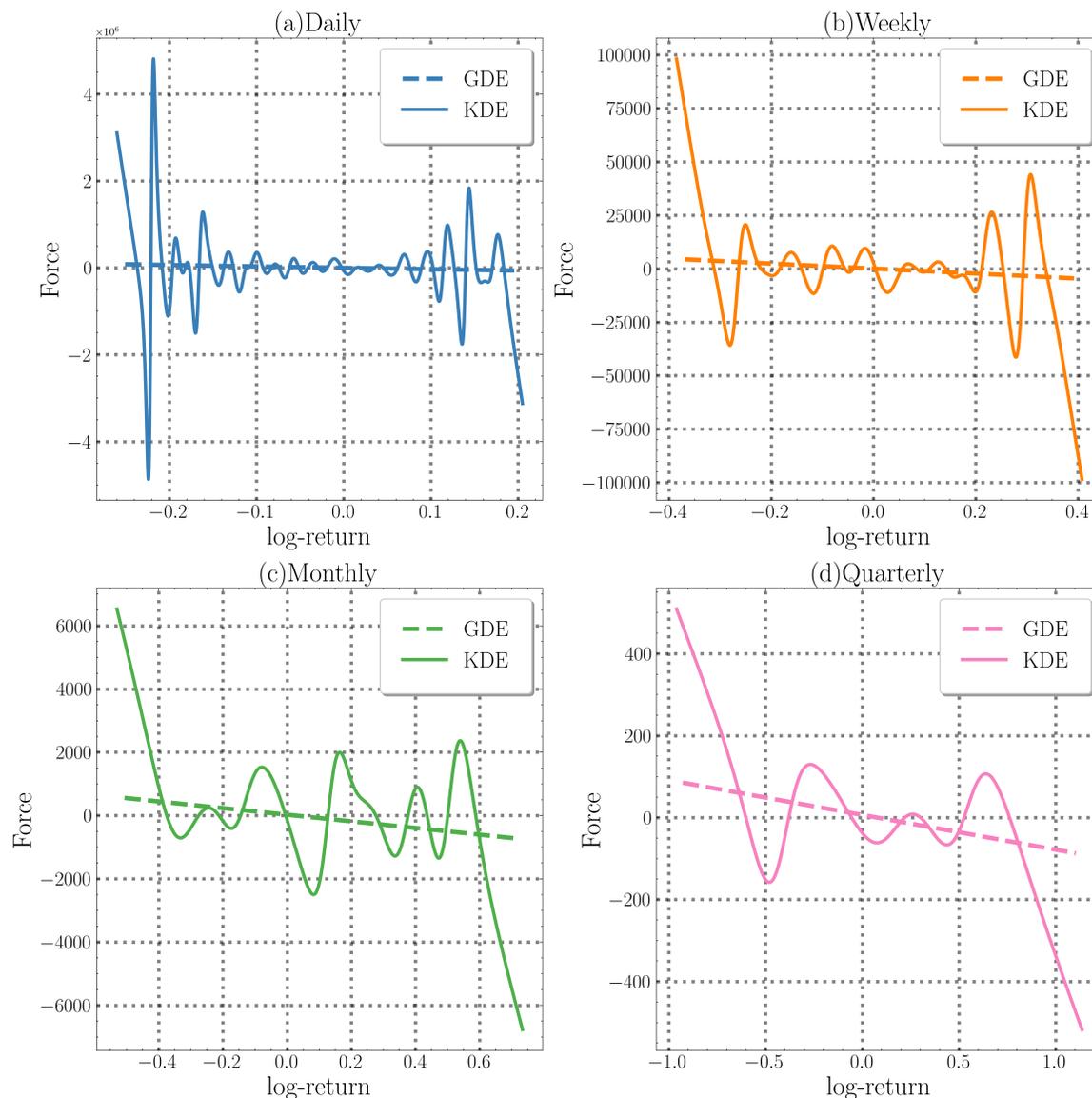


Figure 3.10: Market forces of Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)

figures, the dominant equilibrium states in longer time frames, which are associated with weak returns (i.e., those in the middle of the plots), become deeper and more stable as the time frame extends. This may be related to the tendency of long-term investors to seek small but consistent returns. Another point to note is the widening

of the constraining walls. Observing both axes of the plots, it is evident that the quantum potentials are becoming shallower and wider. We will refer to this as the “widening” behavior and will discuss it further in the following sections.

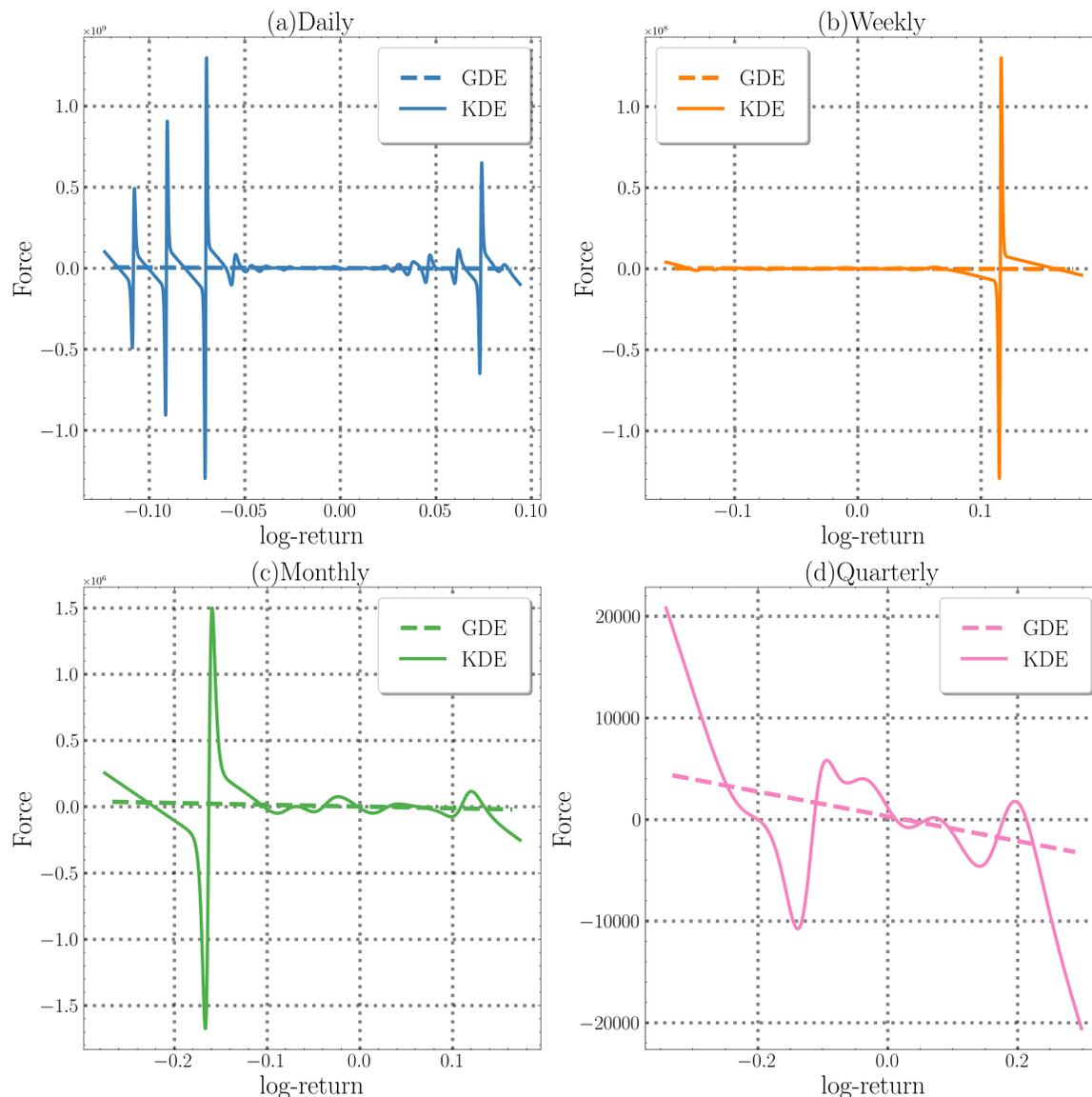


Figure 3.11: Market forces of S&P500 ETF Trust in multiple time frames (from 28/09/2013 to 23/09/2023)

The changes we observed in the structures of probability density functions and quantum potentials are also evident when examining market forces across multiple time frames. As shown in Fig. 3.10 and Fig. 3.11, it is clear that over longer time frames, market forces exhibit simpler but more dominant structures. This indicates

that minor sources of force tend to converge into a smaller number of more influential ones.

In this section, we examined the behavior of our new tools over both short and long-term periods. We observed that over longer time frames, particularly with quarterly price changes, the quantum potentials and quantum forces of real markets become simpler in structure, displaying fewer but more definite features. Most of our findings are consistent with Tahmasebi's original research[17]. That study suggested that quantum potentials tend to resemble Gaussian white noise ($\mathcal{N}(\mu, \sigma^2)$) as we analyze larger time scales. It was proposed that short-term price fluctuations are primarily influenced by market supply and demand, whereas long-term trends are shaped by political and natural factors[17].

Figure 3.12 illustrates the behavior of kurtosis of distributions of log-returns across various time frames and the number of local critical points on the potential graphs. It is evident from these figures that as the time frame increases and kurtosis values decrease, the number of local minima and maxima in the potential graphs also declines, thereby confirming our earlier conclusions.

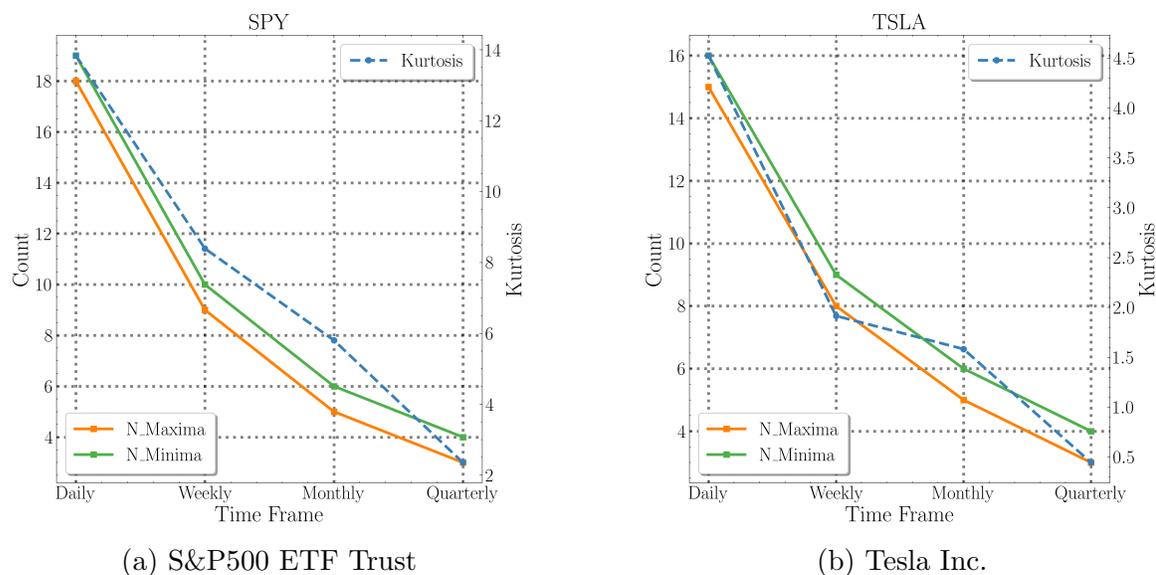


Figure 3.12: Kurtosis and number of equilibrium points of S&P500 ETF Trust and Tesla Inc. in multiple time frames (from 28/09/2013 to 23/09/2023)

3.6 Features

By observing Figures 3.4 and 3.5 we can detect a couple of unique features of quantum potentials and market forces that may carry some information about the current state of the market. We selected and measured a number of features and using the sliding window technique, created new technical indicators based on them. These are the features that were selected and examined:

- length (width) of the quantum potential
- depth of the quantum potential
- number of local minima (stable equilibrium points)
- number of local maxima (unstable equilibrium points)
- market force at the theoretical equilibrium point ($q = \mu$).

To determine the width of the quantum potentials, we measured the distance between the first and last data points in the log-return grid. The depth of the quantum potentials was defined as the difference between their maximum and minimum values. For calculating the remaining three features, we applied numerical and algorithmic techniques².

3.6.1 Time evolution of features

The sliding window technique is commonly used in time series analysis and financial data processing to calculate technical indicators. This method involves computing indicators over a fixed subset of data points (a “window”) that moves or slides through the entire dataset. We used this technique to make our own new indicators that reveal the time evolution of quantum potentials and market forces. We applied this method to our selected financial assets and examined the correlations between newly made indicators with other famous statistical and technical indicators.

Figures 3.13 to 3.16 display daily close price and time evolution of the newly made indicators based on quantum potentials and quantum forces of four selected assets with a sliding 30 days window.

²Refer to subsections 5.3.1, 5.3.2, 5.3.3, 5.3.4, and 5.3.5 in Appendix A for more detail.

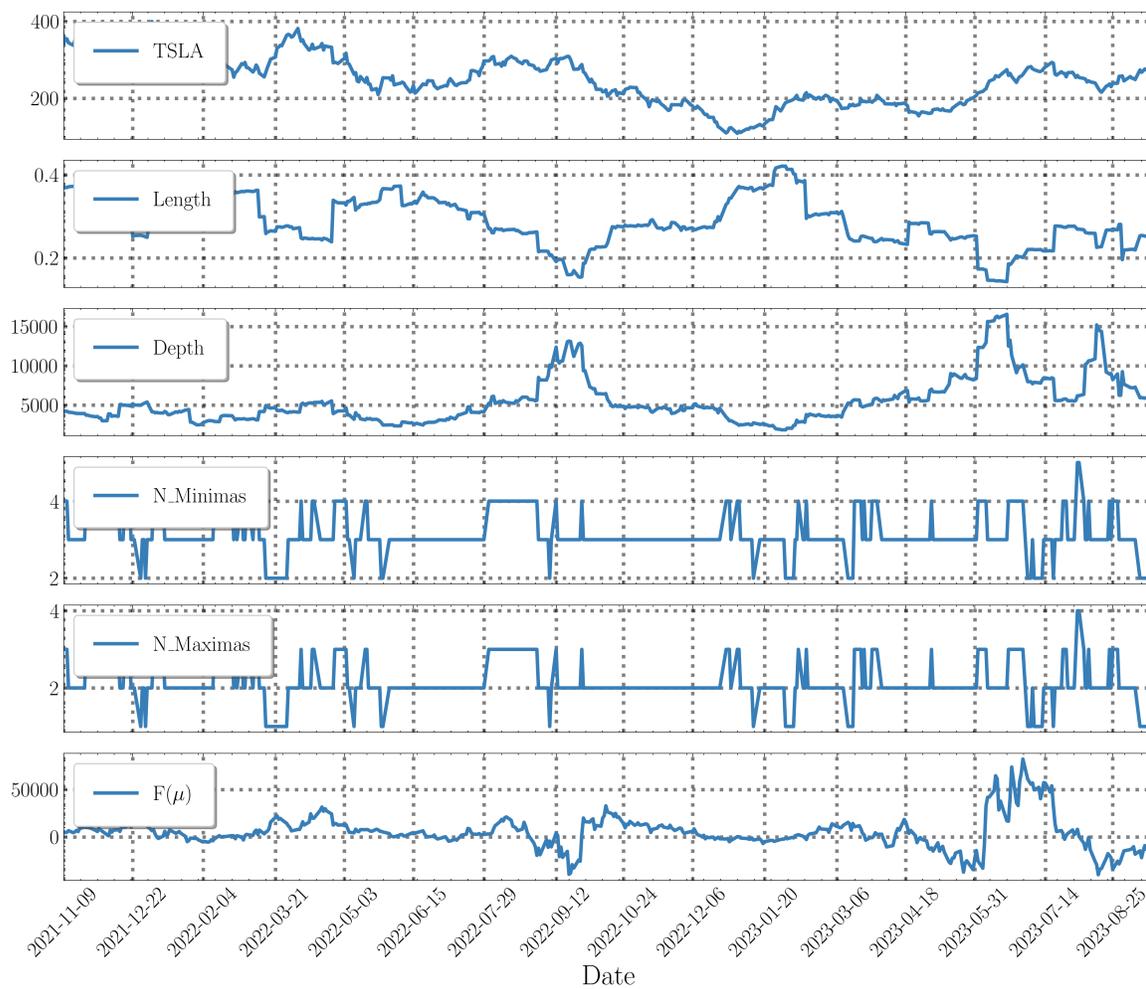


Figure 3.13: Daily close price and indicators based on market potential and market force of Tesla Inc. from 28/09/2021 to 23/09/2023 (window size = 30 days)

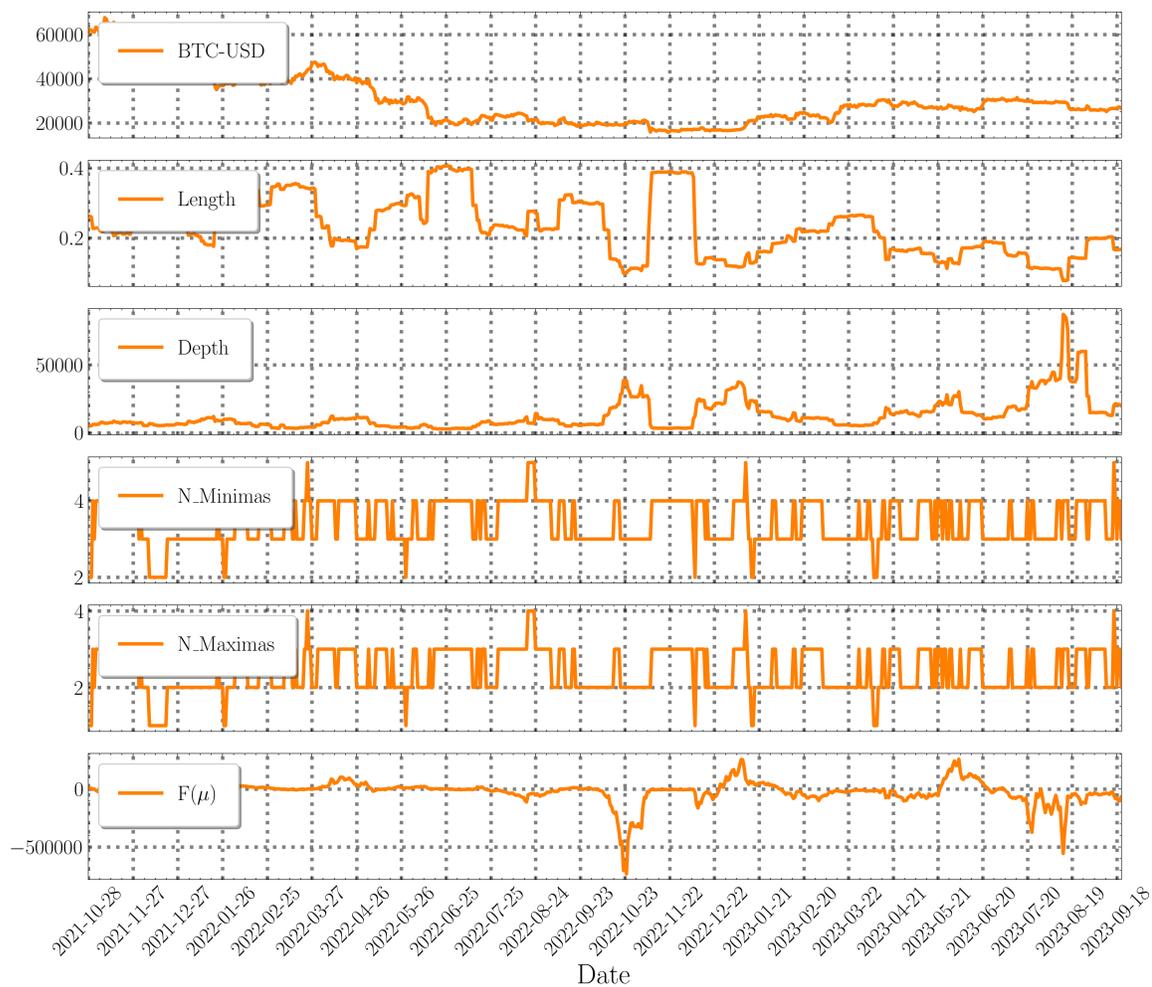


Figure 3.14: Daily close price and indicators based on market potential and market force of Bitcoin (USD). from 28/09/2021 to 23/09/2023 (window size = 30 days)

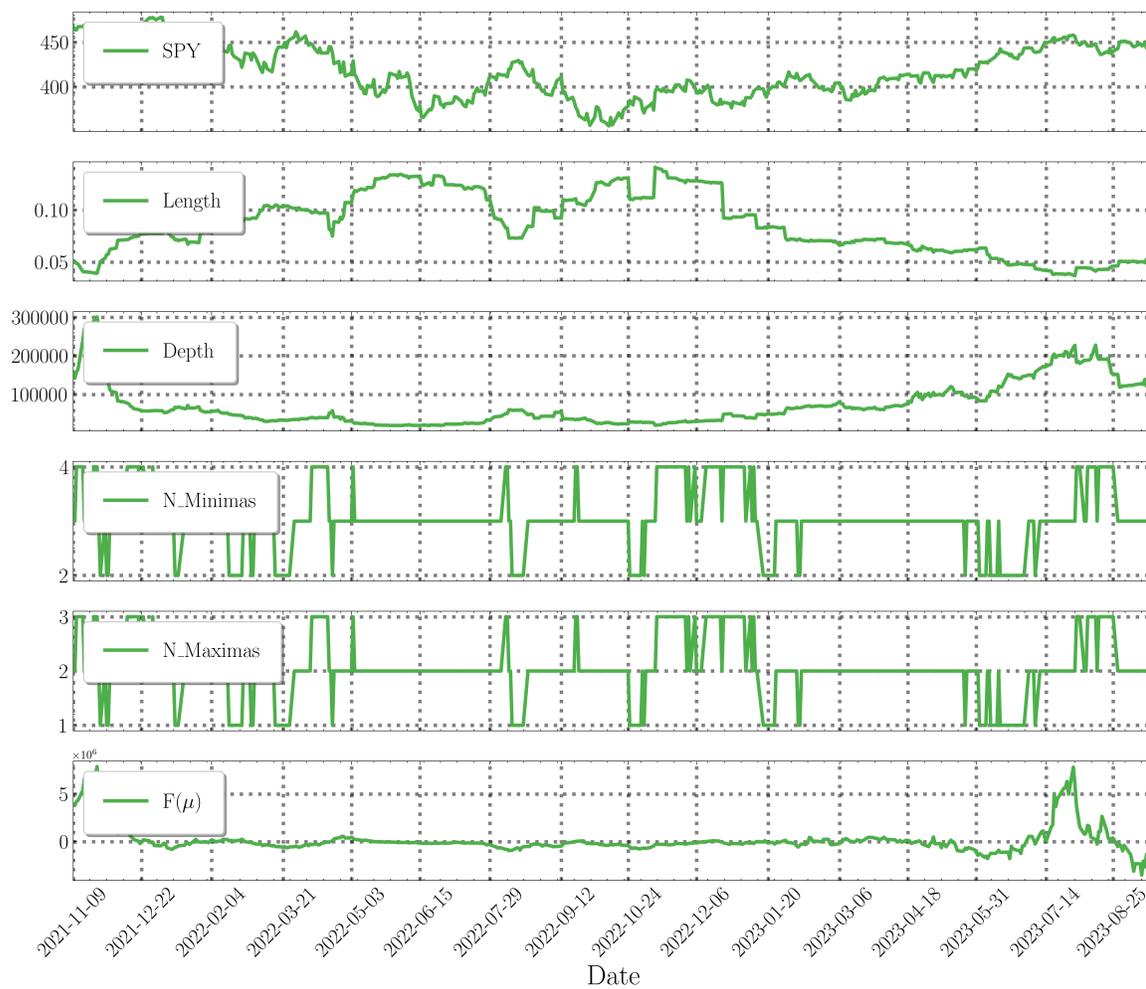


Figure 3.15: Daily close price and indicators based on market potential and market force of S&P500 ETF Trust. from 28/09/2021 to 23/09/2023 (window size = 30 days)

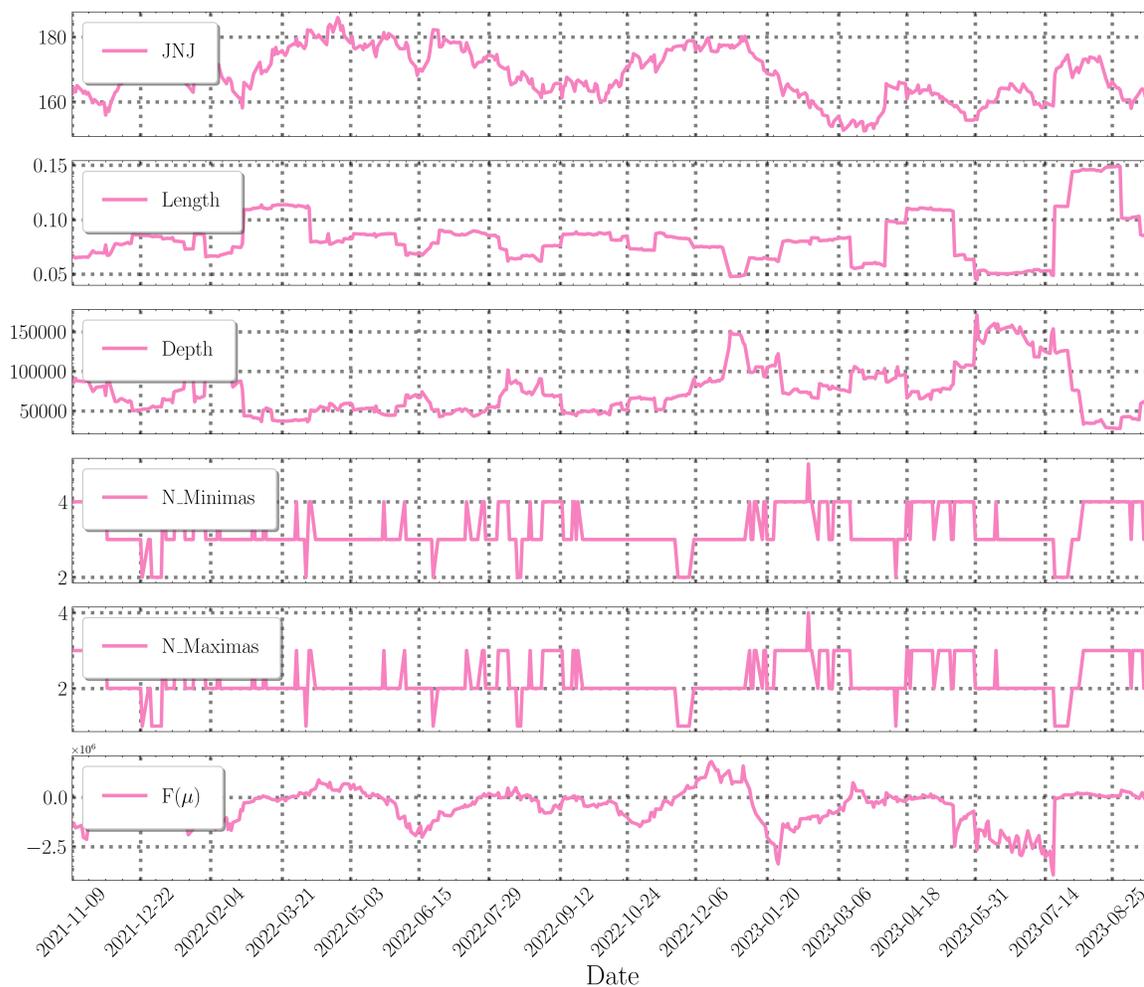


Figure 3.16: Daily close price and indicators based on market potential and market force of Johnson&Johnson. from 28/09/2021 to 23/09/2023 (window size = 30 days)

Tables 3.3 to 3.6 present the Pearson correlation coefficients between new indicators and several well-known statistical and technical indicators. Correlations above 50% are highlighted in green. The Pearson correlation coefficient, denoted by ρ , quantifies the linear correlation between two variables. This coefficient ranges from -1 to 1, where values close to 1 or -1 indicate strong correlation, and values close to 0 indicate weak or no linear correlation.

The Pearson correlation coefficient between two sets of random variables, X and Y , is defined as

$$\rho_{X,Y} = \frac{\text{cov}(X, Y)}{\sigma_X \sigma_Y},$$

where σ_X and σ_Y are the standard deviations of X and Y , respectively, and $\text{cov}(X, Y)$ represents their covariance. Covariance itself is calculated as

$$\text{cov}(X, Y) = E[(X - E[X])(Y - E[Y])],$$

where $E[X]$ and $E[Y]$ are the expected values of X and Y . We used the functionality of the `pandas` library in Python to efficiently compute these correlations.

	p	q	SMA	RSI	MACD	MEAN	STDEV	SKEW	KURT	Length	Depth	N_Min	N_Max	$F(\mu)$
p	1.00	0.08	0.88	0.50	-0.12	0.40	-0.12	0.09	0.17	0.01	0.04	0.16	0.16	0.21
q	0.08	1.00	-0.10	0.23	0.21	0.15	0.01	-0.05	-0.03	-0.00	0.03	-0.09	-0.09	0.02
SMA	0.88	-0.10	1.00	0.09	-0.42	-0.00	-0.02	0.06	0.18	0.12	-0.08	0.19	0.19	0.11
RSI	0.50	0.23	0.09	1.00	0.25	0.89	-0.41	-0.06	0.18	-0.32	0.43	0.06	0.06	0.30
MACD	-0.12	0.21	-0.42	0.25	1.00	0.16	0.09	0.20	-0.34	-0.06	0.03	-0.22	-0.22	-0.07
MEAN	0.40	0.15	-0.00	0.89	0.16	1.00	-0.28	-0.03	0.14	-0.21	0.27	0.06	0.06	0.34
STDEV	-0.12	0.01	-0.02	-0.41	0.09	-0.28	1.00	0.00	-0.16	0.91	-0.88	-0.07	-0.07	-0.08
SKEW	0.09	-0.05	0.06	-0.06	0.20	-0.03	0.00	1.00	-0.31	-0.08	-0.11	-0.09	-0.09	-0.15
KURT	0.17	-0.03	0.18	0.18	-0.34	0.14	-0.16	-0.31	1.00	0.22	0.11	0.25	0.25	-0.29
Length	0.01	-0.00	0.12	-0.32	-0.06	-0.21	0.91	-0.08	0.22	1.00	-0.84	0.07	0.07	-0.17
Depth	0.04	0.03	-0.08	0.43	0.03	0.27	-0.88	-0.11	0.11	-0.84	1.00	0.02	0.02	0.03
N_Min	0.16	-0.09	0.19	0.06	-0.22	0.06	-0.07	-0.09	0.25	0.07	0.02	1.00	1.00	0.00
N_Max	0.16	-0.09	0.19	0.06	-0.22	0.06	-0.07	-0.09	0.25	0.07	0.02	1.00	1.00	0.00
$F(\mu)$	0.21	0.02	0.11	0.30	-0.07	0.34	-0.08	-0.15	-0.29	-0.17	0.03	0.00	0.00	1.00

Table 3.3: Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of Tesla Inc., window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5

	p	q	SMA	RSI	MACD	MEAN	STDEV	SKEW	KURT	Length	Depth	N_Min	N_Max	$F(\mu)$
p	1.00	0.04	0.97	0.15	-0.21	0.06	0.14	0.04	-0.35	0.03	-0.22	-0.08	-0.08	0.13
q	0.04	1.00	-0.07	0.23	0.16	0.19	-0.02	0.14	-0.03	-0.02	0.01	-0.01	-0.01	0.01
SMA	0.97	-0.07	1.00	-0.04	-0.34	-0.13	0.18	-0.10	-0.32	0.07	-0.23	-0.07	-0.07	0.14
RSI	0.15	0.23	-0.04	1.00	0.20	0.86	-0.30	0.69	-0.32	-0.34	0.07	-0.12	-0.12	-0.04
MACD	-0.21	0.16	-0.34	0.20	1.00	0.18	0.08	0.20	0.06	0.10	-0.01	-0.17	-0.17	0.03
MEAN	0.06	0.19	-0.13	0.86	0.18	1.00	-0.37	0.71	-0.23	-0.39	0.13	-0.09	-0.09	-0.07
STDEV	0.14	-0.02	0.18	-0.30	0.08	-0.37	1.00	-0.18	0.08	0.97	-0.78	0.08	0.08	0.28
SKEW	0.04	0.14	-0.10	0.69	0.20	0.71	-0.18	1.00	-0.47	-0.23	-0.09	-0.16	-0.16	-0.10
KURT	-0.35	-0.03	-0.32	-0.32	0.06	-0.23	0.08	-0.47	1.00	0.28	0.21	0.24	0.24	-0.02
Length	0.03	-0.02	0.07	-0.34	0.10	-0.39	0.97	-0.23	0.28	1.00	-0.72	0.16	0.16	0.27
Depth	-0.22	0.01	-0.23	0.07	-0.01	0.13	-0.78	-0.09	0.21	-0.72	1.00	-0.01	-0.01	-0.40
N_Min	-0.08	-0.01	-0.07	-0.12	-0.17	-0.09	0.08	-0.16	0.24	0.16	-0.01	1.00	1.00	0.02
N_Max	-0.08	-0.01	-0.07	-0.12	-0.17	-0.09	0.08	-0.16	0.24	0.16	-0.01	1.00	1.00	0.02
$F(\mu)$	0.13	0.01	0.14	-0.04	0.03	-0.07	0.28	-0.10	-0.02	0.27	-0.40	0.02	0.02	1.00

Table 3.4: Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of Bitcoin, window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5

	p	q	SMA	RSI	MACD	MEAN	STDEV	SKEW	KURT	Length	Depth	N_Min	N_Max	$F(\mu)$
p	1.00	0.09	0.88	0.57	0.02	0.41	-0.63	-0.10	-0.12	-0.64	0.58	-0.05	-0.05	0.32
q	0.09	1.00	-0.08	0.22	0.18	0.16	-0.01	0.08	-0.07	-0.01	0.00	-0.02	-0.02	-0.02
SMA	0.88	-0.08	1.00	0.18	-0.34	0.00	-0.52	-0.28	-0.08	-0.53	0.48	0.01	0.01	0.31
RSI	0.57	0.22	0.18	1.00	0.34	0.80	-0.65	0.20	0.01	-0.63	0.68	-0.02	-0.02	0.41
MACD	0.02	0.18	-0.34	0.34	1.00	0.27	0.21	0.20	-0.29	0.15	-0.17	-0.24	-0.24	-0.19
MEAN	0.41	0.16	0.00	0.80	0.27	1.00	-0.46	0.35	0.06	-0.43	0.40	0.03	0.03	0.16
STDEV	-0.63	-0.01	-0.52	-0.65	0.21	-0.46	1.00	0.03	0.04	0.98	-0.85	0.02	0.02	-0.34
SKEW	-0.10	0.08	-0.28	0.20	0.20	0.35	0.03	1.00	0.05	0.08	-0.08	0.11	0.11	-0.17
KURT	-0.12	-0.07	-0.08	0.01	-0.29	0.06	0.04	0.05	1.00	0.20	0.05	0.32	0.32	0.20
Length	-0.64	-0.01	-0.53	-0.63	0.15	-0.43	0.98	0.08	0.20	1.00	-0.84	0.11	0.11	-0.32
Depth	0.58	0.00	0.48	0.68	-0.17	0.40	-0.85	-0.08	0.05	-0.84	1.00	0.06	0.06	0.61
N_Min	-0.05	-0.02	0.01	-0.02	-0.24	0.03	0.02	0.11	0.32	0.11	0.06	1.00	1.00	0.22
N_Max	-0.05	-0.02	0.01	-0.02	-0.24	0.03	0.02	0.11	0.32	0.11	0.06	1.00	1.00	0.22
$F(\mu)$	0.32	-0.02	0.31	0.41	-0.19	0.16	-0.34	-0.17	0.20	-0.32	0.61	0.22	0.22	1.00

Table 3.5: Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of S&P500 ETF Trust, window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5

	p	q	SMA	RSI	MACD	MEAN	STDEV	SKEW	KURT	Length	Depth	N_Min	N_Max	$F(\mu)$
p	1.00	0.12	0.77	0.69	0.15	0.55	0.33	0.17	-0.03	0.20	-0.35	-0.31	-0.31	0.39
q	0.12	1.00	-0.09	0.26	0.18	0.18	0.05	0.13	0.05	0.03	0.01	-0.12	-0.12	0.02
SMA	0.77	-0.09	1.00	0.12	-0.36	0.01	0.16	-0.13	-0.18	0.02	-0.35	-0.09	-0.09	0.28
RSI	0.69	0.26	0.12	1.00	0.47	0.88	0.29	0.41	0.24	0.30	-0.08	-0.35	-0.35	0.26
MACD	0.15	0.18	-0.36	0.47	1.00	0.32	0.12	0.30	0.05	0.07	0.04	-0.33	-0.33	0.08
MEAN	0.55	0.18	0.01	0.88	0.32	1.00	0.34	0.41	0.25	0.37	-0.13	-0.25	-0.25	0.27
STDEV	0.33	0.05	0.16	0.29	0.12	0.34	1.00	0.47	0.48	0.94	-0.82	0.06	0.06	0.47
SKEW	0.17	0.13	-0.13	0.41	0.30	0.41	0.47	1.00	0.39	0.47	-0.12	-0.16	-0.16	0.08
KURT	-0.03	0.05	-0.18	0.24	0.05	0.25	0.48	0.39	1.00	0.70	-0.04	0.12	0.12	0.23
Length	0.20	0.03	0.02	0.30	0.07	0.37	0.94	0.47	0.70	1.00	-0.66	0.17	0.17	0.43
Depth	-0.35	0.01	-0.35	-0.08	0.04	-0.13	-0.82	-0.12	-0.04	-0.66	1.00	-0.13	-0.13	-0.45
N_Min	-0.31	-0.12	-0.09	-0.35	-0.33	-0.25	0.06	-0.16	0.12	0.17	-0.13	1.00	1.00	-0.12
N_Max	-0.31	-0.12	-0.09	-0.35	-0.33	-0.25	0.06	-0.16	0.12	0.17	-0.13	1.00	1.00	-0.12
$F(\mu)$	0.39	0.02	0.28	0.26	0.08	0.27	0.47	0.08	0.23	0.43	-0.45	-0.12	-0.12	1.00

Table 3.6: Correlation matrix of indicators (normalized to $[0,1]$) with daily close prices (p) and log-returns (q) of Johnson&Johnson, window size = 30 days, green numbers represent correlations greater than 0.5 or less than -0.5

3.6.2 Analysis

Looking at the new indicators (e.g., Figures 3.13-3.16 and Tables 3.3-3.6), we can identify a couple of general behaviors that consistently appear regardless of the asset or time series.

Correlations between length and depth of quantum potentials and STDEV

The correlation matrices reveal a strong positive correlation between the standard deviation and the length of the quantum potentials, and a strong negative correlation between the depth of the quantum potential and the other two. Figure 3.17 displays this correlation for all four assets in more detail.

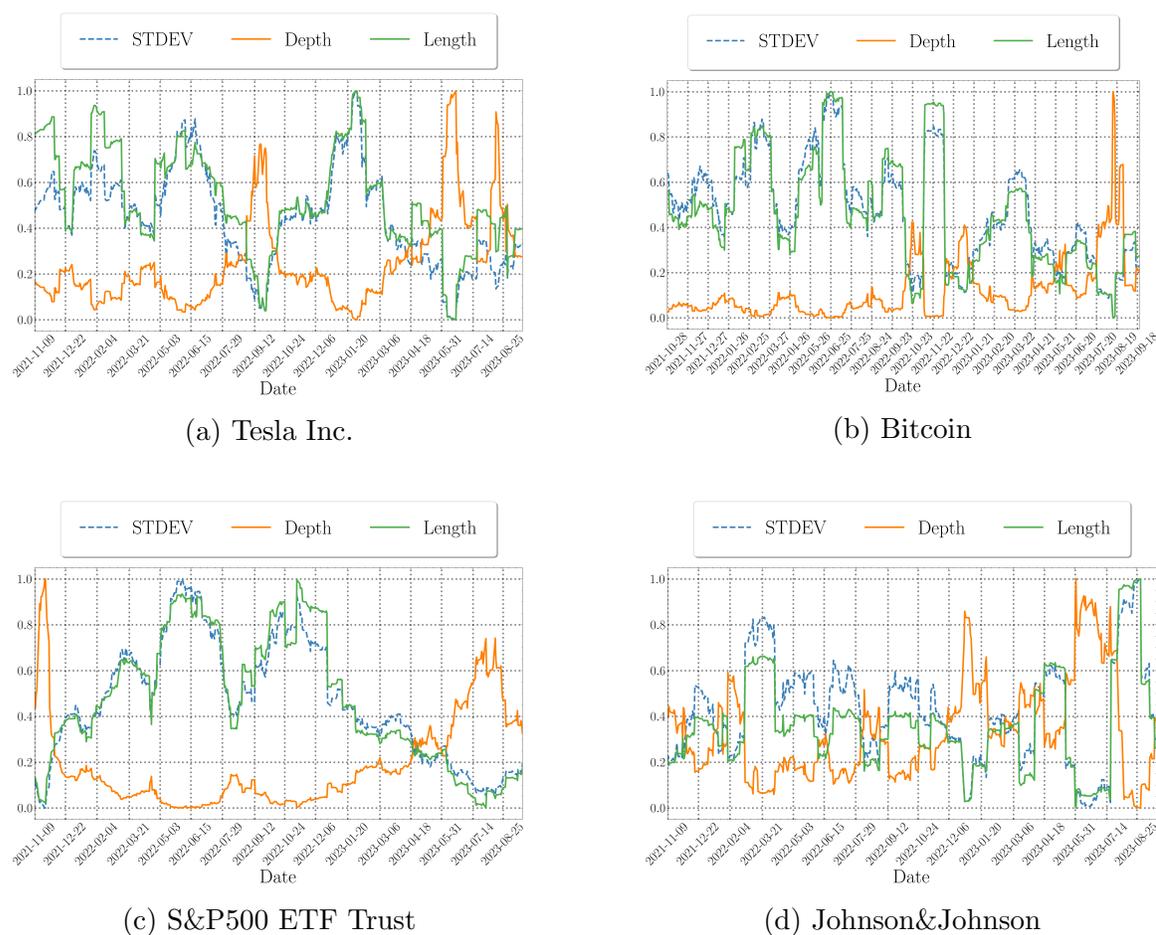


Figure 3.17: The correlations between Length and Depth of Potential and STDEV (normalized to $[0,1]$, window size = 30 days)

As we can see the depth and length of quantum potentials are positive and negatively correlated with standard deviation. Based on this fact, for a volatile market, the well shaped quantum potentials must be shallower and wider. Figure 3.18 compares two quantum potentials from Tesla Inc. within the most and least volatile 14 day windows. This suggests that in a volatile market, the likelihood of prices breaking through the boundaries of the shallower potential is higher, indicating increased risk. Conversely, in a low-volatility and trending market, price changes are confined within a specific range.

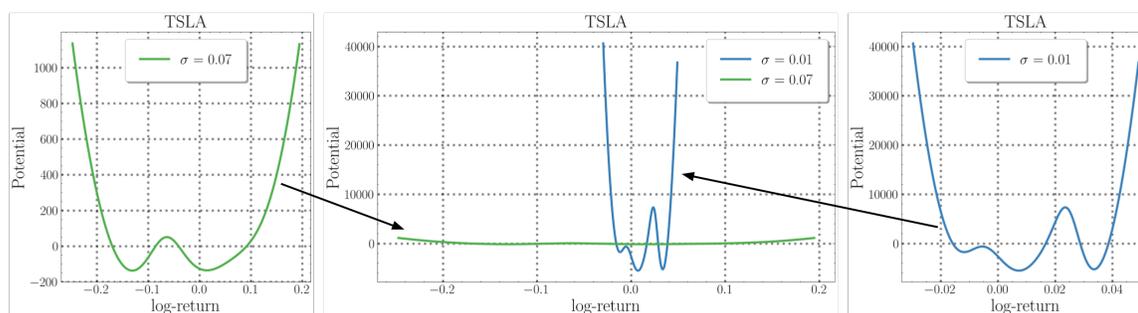


Figure 3.18: Comparison of two Potentials with different STDEV (σ) values (Example: TSLA, window size = 14 days)

Number of minima and number of maxima

As discussed before, the local minima in quantum potentials represent the dominating log-returns for the current state of any asset within a specified time frame (window). This means that instead of just one equilibrium state for a normally distributed system, we get multiple dominating equilibrium points. The system can fall into any of these points and will need the push from the market to jump to the other ones. Number of maxima is and should be always one less than number of minima to maintain the structure of the local quantum potential wells.

Force at theoretical equilibrium

In section 3.4, we discussed that the market force for a Gaussian system vanishes at the average log-return, expressed as $F_{QG}(\mu) = 0$. We computed the market force at the average log-return for each sliding window using the KDE-fitted data. We hypothesized that this feature might indicate a system's divergence from a Gaussian one. In a Gaussian system, regardless of the statistical values (mean, STDEV, skewness, kurtosis, etc.), the system forces individual players towards a converging point—the average point, the global equilibrium. However, a more realistic model suggests that the system tends to deviate from this equilibrium. This deviation varies over time and could signify the failure of the efficient market hypothesis (EMH). According to EMH, if markets are efficient, price changes should follow a random walk, implying that future price changes are independent of past price changes. The random walk theory often leads to the assumption that price changes are normally distributed, particularly in the weak form of EMH. However, in practice, financial returns frequently exhibit fat tails, deviating from the normal distribution. The value of market force at the average point could serve as another indicator of the validity of EMH at any given moment.

Chapter 4

Conclusion

This work was built upon two original studies on quantum potentials of log-returns of financial time series[17, 18]. According to F. Tahmasebi et al. and Shen et al., the quantum potentials of financial time series are due to the entanglement between an asset's price and its previous day's price. Based on this original model, we set to study the time evolution of quantum potentials. Quantum potentials are made using an estimated or defined probability density function of logarithmic price returns (see Eq. 2.20). To estimate the probability density function of log-returns, we used Gaussian and kernel density estimates. Having created quantum potentials, we were then prepared to concentrate on the two primary objectives of this research. These goals were:

- Investigating deviations of financial markets from normal (Gaussian) Systems,
- Studying time evolution of quantum potential and quantum force based indicators and finding any correlations with other technical and statistical indicators.

As we advanced through the project, we obtained intriguing results in both domains. The following sections of this chapter will explore these achievements in greater detail and discuss the potential future directions for this project.

4.1 Deviations of real markets from random walk theory

The earliest models of financial markets treated them as fair games, similar to a coin toss. According to the EMH, a fair game implies that each player has an equal chance of winning under the same conditions, with outcomes often modeled by a normal distribution when influenced by many independent factors, as suggested by the central limit theorem. This results in a distribution of outcomes approximating a normal distribution over time. In processes following a normal distribution, it makes sense to envision a single minimum point in potential, aligning with the expected value, implying that no one emerges as the ultimate winner in a fair game. In systems governed by conservative forces, like gravity or electrostatic forces, the force acting on an object relates to the system's potential energy. At equilibrium, where the force is zero, the potential energy is at a minimum (stable equilibrium) or maximum (unstable equilibrium), indicating how the system will respond to changes in configuration. In a fair game, market forces drive every player to an equilibrium state, which is the expected value of log-returns. The concepts of market force, market potential, and equilibrium states are interconnected through potential energy, where equilibrium corresponds to points of minimum or maximum potential energy. Differences between price trends and a Wiener process reveal distinct characteristics in quantum potential, including multiple local minima and unstable equilibrium points, indicating a complex and dynamic market force.

4.2 Correlations between technical, statistical, and quantum-based indicators

A very surprising finding is the strong correlation between moving standard deviation and the length and depth of quantum potentials. The results suggest that in a highly volatile market, the quantum potentials tend to be shallower and wider. This implies a higher likelihood of the price escaping beyond the constraining walls. There might be additional technical or statistical indicators that could correlate with our quantum-based indicators. The quest to identify an indicator based on pilot wave theory that captures market trends is ongoing. Our goal is to develop a structure that

effectively maps two of the most important characteristics of any market: trend and volatility.

4.3 The future

One of the primary objectives in designing indicators is to develop predictive machine learning models and trading strategies. To effectively design a model for financial purposes, it is crucial to access unfiltered raw tick data. This necessity arises because you must filter this data yourself to ensure accuracy and relevance. The basic log-return formula shown in Eq. 1.1 could be replaced with more advanced techniques, such as fractional differentiation, as proposed by Marcos López de Prado[32]. Additionally, using dollar or volume bars is often considered a better alternative to time bars. These initial data pre-processing steps are intensive and require substantial computational resources. Following this, the process involves creating quantum potentials and indicators for the newly processed data. Subsequently, a machine learning model is designed, which includes determining the feature importance of each indicator and fine-tuning the model to optimize its predictive capabilities.

Chapter 5

Appendix A: Code Snippets

This chapter lists all the Python algorithms, functions, and code snippets used throughout the project.

5.1 Download and pre-processing historical data

5.1.1 [yahoo!](#)finance API

```
1 class yahoodata:
2     """
3     Imports histroical data from yahoo finance for the given
4     parameters:
5
6     ticker: symbol
7     period1&2: starting and ending dates
8     time_frame: time frame ("1d": daily, "1mo":monthly)
9
10    The getdata() method gives the entire historical data (OHLCV)
11    """
12
13    def __init__(self, ticker, period1, period2, time_frame):
14        self.ticker = ticker
15        self.period1 = period1
16        self.period2 = period2
17        self.time_frame = time_frame
```

```

18 def getdata(self):
19     history = yf.download(
20         tickers=self.ticker,
21         start=self.period1,
22         end=self.period2,
23         interval=self.time_frame,
24     )
25     return history.dropna().reset_index().set_index("Date")

```

Listing 5.1: A class that stores historical financial data

5.1.2 Log-returns

```

1 def log_returns(historical_data):
2     """
3     Historical data must be a numpy array of prices (Open, High, Low,
4     or Close)!
5     """
6     returns = np.log(historical_data) - np.log(historical_data.shift
7     (1))
8     return returns

```

Listing 5.2: A function that calculates log-returns of a given time series

5.1.3 Numerical differentiation

```

1 def symmdiff(q, f, firstDer=False, secondDer=True):
2     """
3
4     Takes the 2nd and first derivatives of f(q) with respect to q (i.
5     e.  $d^2f/dq^2$  and  $df/dq$ ).
6     Arguments:
7         f: the function f(q)
8         q: the base q
9     """
10
11     # Calculating the average grid spacing dq
12     dq = (q - q.shift(1)).mean()
13
14     # calculating the second derivative of f(q) with respect to q
15     if secondDer and not firstDer:

```

```

16         d2f = (f.shift(1) + f.shift(-1) - (f * 2)) / (dq**2)
17         return d2f # D2 is 2 variables shorter than q => len(q) =
len(D2) + 2
18
19     # calculating the first derivative of f(q) with respect to q
20     elif firstDer and not secondDer:
21         dif = (f.shift(-1) - f.shift(1)) / (dq * 2)
22         return dif # D1 is 1 variable shorter than q => len(q) =
len(D1) + 1

```

Listing 5.3: A function that finds the first or 2nd symmetrical derivative of any 2 dimensional dataset.

5.2 Creating density, quantum potentials, and quantum forces

5.2.1 GDE

```

1 def gaussian_fit(returns):
2     """
3     Calculates the gaussian function for a given dataset:
4     returns: original return values for obtaining
5             mean and standard deviation
6     """
7     mu = returns.mean()
8     sigma = returns.std()
9
10    q_uniform = density_function(
11    returns, size=len(returns), common_norm=True, bw_method="
silverman", clf=True).index.values
12
13    a = 1 / (sigma * np.sqrt(2 * np.pi))
14    power = (-1 * ((q_uniform - mu) ** 2)) / (2 * (sigma**2))
15    f = a * np.exp(power)
16
17    return pd.DataFrame(
18        {"Rephurbished Returns (q)": q_uniform, "GDE": f}
19    ).set_index("Rephurbished Returns (q)")

```

Listing 5.4: A function that fits a Gaussian density function to a given distribution

5.2.2 KDE

```

1 def density_function(
2     returns, size, common_norm=True, bw_method="silverman", clf=True,
3     *args, **kwargs
4 ):
5     """
6     Constructs the pdf, f(q), of a given distribution,
7     q (price returns) using Kernel Density Estimation method.
8     Arguments:
9         q: the grid (price returns)
10        N: the size of the reconstructed data
11        bw_method: bandwidth (smoothing factor)
12                of Kernel Density Estimator.
13                Default is set to 'silverman'.
14    """
15    graph_data = sns.kdeplot(
16        data=returns,
17        bw_method=bw_method,
18        common_norm=True,
19        gridsize=size,
20        *args,
21        **kwargs,
22    )
23    q = graph_data.get_lines()[0].get_xdata()
24    f = graph_data.get_lines()[0].get_ydata()
25    if clf:
26        plt.clf()
27    return pd.DataFrame(
28        {"Rephurbished Returns (q)": q, "KDE": f}
29    ).set_index("Rephurbished Returns (q)")

```

Listing 5.5: A function that estimates the density function of a given distribution using KDE method

5.2.3 Quantum potential and quantum force

```

1 def potential(pdf):
2     """
3     Calculates the quantum potential of a given distribution, f(q)
4     The theoretical definition of quantum potential is:
5         
$$U = 1/f(q) * d^2f/dq^2$$


```

```

6     """
7     q = pdf.index.to_series()
8
9     quantum_potential = symmdiff(q, np.sqrt(pdf)) / np.sqrt(pdf)
10    return pd.DataFrame(
11        {"q": q.to_numpy().squeeze(), "f(q)": quantum_potential.
12         to_numpy().squeeze()}
13    ).set_index("q")

```

Listing 5.6: A function that gives the quantum potential of a given quantum field amplitude/probability density

```

1 def market_force(returns, grid_size=100, dist_type=None):
2     """
3     Calculates the market force of a Q-Potential
4     The theoretical definition of the market force is:
5         F = -dU/dq
6     Arguments:
7         returns: rate of returns
8         grid_size: size of the rephurbished returns (set to 100
9     by default).
10    """
11    if dist_type == "GDE":
12        dens_temp = gaussian_fit(returns)
13    elif dist_type == "KDE":
14        dens_temp = density_function(returns=returns, size=grid_size)
15    elif dist_type == None:
16        dens_temp = density_function(returns=returns, size=grid_size)
17    else:
18        return None
19
20    pot_temp = potential(pdf=dens_temp)
21
22    MF = -symmdiff(
23        q=pot_temp.index.to_series(), f=pot_temp, firstDer=True,
24        secondDer=False
25    )
26
27    return MF

```

Listing 5.7: A function that computes the market force of any given log-returns (with GDE and KDE options).

5.3 Features (technical indicators)

5.3.1 Potential depth

```

1 def potential_depth(returns, grid_size=100):
2     """
3     Calculates the depth of a Q-Potential for given rates of returns
4     Arguments:
5         returns: rate of returns
6         grid_size: size of the rephurbished returns (set to 100
by default).
7     """
8     dens_temp = density_function(returns=returns, size=grid_size)
9     pot_temp = potential(dens_temp)
10
11    maxHeight = pot_temp.max()[0]
12    minDepth = pot_temp.min()[0]
13    return maxHeight - minDepth

```

Listing 5.8: A function that measures the depth of a quantum potential.

5.3.2 Potential length (width)

```

1 def potential_width(returns, grid_size=100):
2     """
3     Calculates the width of a Q-Potential for given rates of returns
4     Arguments:
5         returns: rate of returns
6         grid_size: size of the rephurbished returns (set to 100
by default).
7     """
8     dens_temp = density_function(returns=returns, size=grid_size)
9
10    maxWidth = dens_temp.index.to_series().to_numpy()[-2]
11    minWidth = dens_temp.index.to_series().to_numpy()[1]
12
13    return maxWidth - minWidth

```

Listing 5.9: A function that measures the width (length) of constraining walls of quantum potentials.

5.3.3 Local maximum points

```

1 def local_maxima(returns, grid_size=100):
2     """
3     Finds the local maxima of a Q-Potential for given rates of
4     returns
5     Arguments:
6         returns: rate of returns
7         grid_size: size of the rephurbished returns (set to 100
8         by default).
9     """
10    dens_temp = density_function(returns=returns, size=grid_size)
11    pot_temp = potential(pdf=dens_temp)
12
13    localMax = pot_temp.shift(-1).rolling(window=3, axis=0).apply(np.
14    argmax)
15
16    Maxima = pot_temp[localMax == 1].dropna()
17
18    return Maxima

```

Listing 5.10: A function that finds local maximum points known as unstable equilibrium points.

```

1 def maxima_count(returns, grid_size=100):
2     """
3     Finds the counts of the local maxima of a Q-Potential for given
4     rates of returns
5     Arguments:
6         returns: rate of returns
7         grid_size: size of the rephurbished returns (set to 100
8         by default).
9     """
10    Maxima = local_maxima(returns=returns, grid_size=grid_size)
11    return len(Maxima)

```

Listing 5.11: A function that counts the number of local maxima.

5.3.4 Local minimum points

```

1 def local_minima(returns, grid_size=100):
2     """

```

```

3     Finds the local minima of a Q-Potential for given rates of
returns
4     Arguments:
5         returns: rate of returns
6         grid_size: size of the rephurbished returns (set to 100
by default).
7     """
8     dens_temp = density_function(returns=returns, size=grid_size)
9     pot_temp = potential(pdf=dens_temp)
10
11    localMin = pot_temp.shift(-1).rolling(window=3, axis=0).apply(np.
argmin)
12
13    Minima = pot_temp[localMin == 1].dropna()
14
15    return Minima

```

Listing 5.12: A function that finds local minimum points known as stable equilibrium points.

```

1 def minima_count(returns, grid_size=100):
2     """
3     Finds the counts of the local minima of a Q-Potential for given
rates of returns
4     Arguments:
5         returns: rate of returns
6         grid_size: size of the rephurbished returns (set to 100
by default).
7     """
8     Minima = local_minima(returns=returns, grid_size=grid_size)
9     return len(Minima)

```

Listing 5.13: A function that counts the number of local minima.

5.3.5 Market force at $q = \mu$

```

1 def force_at_mu(returns, grid_size=100):
2     """
3     Calculates the market force of a Q-Potential at avrage log-return
(or very close to average log-return)
4     Mathematically:
5         F(q=mu) = -dU/dq | q=mu

```

```
6     Arguments:
7         returns: rate of returns
8         grid_size: size of the rephurbished returns (set to 100
9         by default).
10        """
11    MF = market_force(returns=returns, grid_size=grid_size)
12    return MF.iloc[abs(MF.index.to_series()).argmin()].to_numpy().
13    squeeze()
```

Listing 5.14: A function that finds the value of market force at average log-return.

Bibliography

- [1] R. N. Mantegna and H. E. Stanley. *Introduction to Econophysics: Correlations and Complexity in Finance*. (Cambridge University Press, Cambridge, 1999).
- [2] L. Bachelier. Théorie De La Spéculation. *Annales Scientifiques de l'École Normale Supérieure*, **17**:21–86, 1 1900.
- [3] P. A. Samuelson. *Foundations of Economic Analysis*. (Harvard University Press, Cambridge, 1983).
- [4] P. A. Samuelson. *Economics: An Introductory Analysis*. (McGraw-Hill, New York, 1948).
- [5] P. A. Samuelson. Mathematics of Speculative Price. *SIAM Review*, **15**:1–42, 1973.
- [6] J. O. Weatherall. *The Physics of Wall Street, New York*. (Houghton Mifflin Harcourt, 2013).
- [7] M. F. M. Osborne. Brownian Motion in the Stock Market. *Operations Research*, **7**:145–173, 1959.
- [8] B. Mandelbrot. The Variation of Certain Speculative Prices. *The Journal of Business*, **36**:394, 1963.
- [9] P. P. Lévy. *Calcul Des Probabilités*. (Gauthier-Villars, Paris, 1925).
- [10] E. O. Thorp and S. T. Kassouf. *Beat the Market*. (Random House, New York, 1967).
- [11] F. Black and M. Scholes. The Pricing of Options and Corporate Liabilities. *Journal of Political Economy*, **81**:637–654, 1973.
- [12] L. Borland. Option Pricing Formulas Based on a Non-Gaussian Stock Price Model. *Physical Review Letters*, **89**:098701, 2002.
- [13] D. Bohm and B. J. Hiley. *The Undivided Universe*. (Routledge, New York, 1993).

- [14] D. Bohm. A Suggested Interpretation of the Quantum Theory in Terms of “Hidden” Variables. I. *Physical Review*, **85**:166–179, 1952.
- [15] D. Bohm. A Suggested Interpretation of the Quantum Theory in Terms of “Hidden” Variables. II. *Physical Review*, **85**:180–193, 1952.
- [16] O. Choustova. Quantum Modeling of Nonlinear Dynamics of Stock Prices: Bohmian Approach. *Theoretical and Mathematical Physics*, **152**:1213–1222, 2007.
- [17] F. Tahmasebi et al. Financial Market Images: A Practical Approach Owing to the Secret Quantum Potential. *Europhysics Letters*, **109**:30001, 2015.
- [18] C. Shen and E. Haven. Using Empirical Data to Estimate Potential Functions in Commodity Markets: Some Initial Results. *International Journal of Theoretical Physics*, **56**:4092–4104, 2017.
- [19] H. Khaksar et al. Using the Quantum Potential in Elementary Portfolio Management: Some Initial Ideas. *Entropy*, **23**:180, 2021.
- [20] F. Bagarello and E. Haven. Toward a Formalization of a Two Traders Market with Information Exchange. *Physica Scripta*, **90**:015203, 2014.
- [21] B. E. Baaquie. Statistical Microeconomics. *Physica A Statistical Mechanics and its Applications*, **392**:4400–4416, 2013.
- [22] A. S. Kyle. Continuous Auctions and Insider Trading. *Econometrica*, **53**:1315–1335, 1985.
- [23] D. Orrell. A Quantum Oscillator Model of Stock Markets. *Quantum Economics and Finance*, **1**:74–81, 2024.
- [24] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. (CRC Press, Boca Raton, 1998).
- [25] Murray M. Rosenblatt. Remarks on Some Nonparametric Estimates of a Density Function. *The Annals of Mathematical Statistics*, **27**:832–837, 1956.
- [26] B. S. Thomson. *Symmetric Properties of Real Functions*. (CRC Press, Boca Raton, 1994).
- [27] P. R. Mercer. *More Calculus of a Single Variable*. (Springer, New York, 2014).
- [28] R. J. LeVeque. *Finite Difference Methods for Ordinary and Partial Differential Equations*. (SIAM, Philadelphia, 2007).
- [29] P. R. Holland. *The Quantum Theory of Motion*. (Cambridge University Press, Cambridge, 1993).

- [30] E. Haven and A. Y. Khrennikov. *Quantum Social Science*. (Cambridge University Press, New York, 2013).
- [31] A. Y. Khrennikov. *Ubiquitous Quantum Structure: From Psychology to Finance*. (Springer-Verlag, Heidelberg, 2010).
- [32] M. L. De Prado. *Advances in Financial Machine Learning*. (John Wiley & Sons, 2018).