QxEAI: Assisting equity trading with quantum-like evolutionary algorithm

Lizhi Xin[1, †], Kevin Xin[1, *]

[1]XINVISIONQ, INC.

†Contributing author. E-mail: lizhi@xinvisionq.com;
*Corresponding author. E-mail: kevin@xinvisionq.com;

Abstract

Knowing when to buy or sell at the right time is crucial when it comes to trading on the stock market, but many times this is easier said than done. This paper proposes QxEAI, a quantum-like evolutionary algorithm, that produces action sequences to assist traders to buy or sell in the decision-making process. QxEAI utilizes the quantum principle of superposition and Darwinian natural selection to confront the dual uncertainty of the market and the traders' actions as well as the interactivity between the traders participating and the market. Using the Dow Jones Index as a medium, QxEAI is able to produce a forecast of the following week with odds of 80% by training the preceding four weeks of data.

Keywords: probabilistic forecasting, genetic programming, quantum-like evolutionary algorithm

1 Introduction

The equity market is known for being a volatile marketplace; there have been many attempts at formulating methods to predict equities' future trends, mostly being time series forecasting approaches [1]. There is the first systematic approach for time series forecasting; the Box-Jenkins model [2], which integrated the existing knowledge of the autoregressive and moving average methodologies, as well as many other mathematical methods [3-5]. Building off of the Box-Jenkins model, a wide range of AR methods have emerged, most notably the widely used auto-regressive integrated moving average (ARIMA) method [6].

When it comes to forecasting the stock market, it's universally acknowledged that it is very difficult, if not impossible to predict the exact absolute closing prices of the stocks traded. Therefore, mainstream traditional methods have broken down the approach into three steps: find a trend, locate an interval cycle, and treat external environment uncertainty as noise or completely ignore it [7-10].

Now the question: is it possible to forecast the future trend of the market accurately enough to be able to make the corresponding decisions of buy or sell?

The main challenge is that of a dual uncertainty: one that is the result of the interactions between the inherently uncertain market and the traders participating. Essentially the market itself is inherently uncertain, it acts as if it is in a random walk, no one knows when it will go up or down at any given time, this is the first uncertainty, but it's not just the market alone that comes into play, the traders and the corresponding actions (buy or sell) that they can take leads to another uncertainty. Since the market is uncertain, this clouds the trader's judgement of exactly what action to take (whether to buy or sell) and when they do so this in turn affects the trend of the market, because it is the "collaborative effort" of all the traders participating that determines the ever-changing states of the market. Basically, the uncertain nature of the market influences the uncertainty of the traders' actions which then in turn influences the uncertainty of the market and vice versa.

This work presents QxEAI, an algorithm that produces action sequences that utilizes a quantum-like evolutionary algorithm based on training a quantum-like logic decision tree to assist traders by providing them with corresponding actions of whether to buy or sell in response to the respective trend of the market (up or down). The contributions of this paper are 1) we propose a quantum-like evolutionary algorithm (QxEAI) that assists in equity trading, one which incorporates principles of evolution and principle of quantum superposition to tackle the uncertain market by producing a series of action sequences, and 2) we show using real world data from the Dow Jones, that this model is viable enough to do so by trading Dow Jones Futures.

In addition, our method has a few key advantages:

a) We are able to subtly model both the uncertainty of the market state (up or down) and traders' actions (buy or sell) all together under unified complex Hilbert space, instead of modeling them separately.

b) As the model learns through repetitive iterations, by evolution and natural selection, QxEAI optimizes strategies (formulated as logic decision trees) that guide traders to take actions with certain degree of beliefs.

c) Our approach does not assume uncertainty is "noise", something that traditional approaches seek to eliminate or reduce. By learning with the mindset that uncertainty is inherent, QxEAI is able to provide forecasts for data that have little or no regularities, a case where many other forecasting methods might not perform as well.

By utilizing both the principle of quantum superposition and evolution, QxEAI can find valuable information from raw data by providing traders with action sequences to guide them, thus QxEAI's role can be seen as the equivalent of an AI assistant trader. The structure of this paper as follows: Section 2 details the QxEAI methodology. Section 3 are the results. Section 4 is the discussion. Section 5 is the conclusion.

2 Methods

For equity trading, with the Dow Jones Index as a medium, the main elements are:

1. Data: the market can be modeled in the form of a time series, with each recorded closing price represented by a point in the data, described in (1).

$$\{(t_k, s_k)\}\ k = 1, \cdots, N \tag{1}$$

where t_k is time and s_k are the observed closing prices.

2. Market trend: at any given time, the trend of the market as reflected by deducing the change from the previous point which can be described as (2).

$$Q = \begin{cases} q_1 | \omega_1 \\ q_2 | \omega_2 \end{cases} \tag{2}$$

where q_1 is when the market is going up, ω_1 is the frequency of the market going up, q_2 is when the market is going down, ω_2 is the frequency of the market going down. At any given point, in order to deduce whether the market has gone up or down relative to the previous point, if point s_k is greater than s_{k-1} then the market has gone up, and if point s_k is less than s_{k-1} then the market has gone down.

3. Traders' actions: Regardless of what state the market is in, at any given time traders can take the actions of buy or sell, which can be described as (3):

$$A = \begin{cases} a_1 | p_1 \\ a_2 | p_2 \end{cases} \tag{3}$$

where a_1 is the trader will decide to buy, p_1 is the probability of that the trader will buy with degree of beliefs, a_2 is the trader will decide to sell, and p_2 is the probability of that the trader will sell with degree of beliefs.

4. Decision process: The process of taking a single action in correspondence of what state the market is in can be described as (4):

$$D | Q(q_1, q_2) \to A(a_1, a_2) \tag{4}$$

Where D is the decision process, Q is the trend of the market with q_1 and q_2 being whether the market has gone up or down respectively and A are the corresponding actions that the traders can take with a_1 being to buy and a_2 to sell.

5. Evaluation: For every trade made there are four possible outcomes.

   1) the market is going up and the trader buys.

2) the market is going up and the trader sells.

3) the market is going down and the trader buys.

4) the market is going down and the trader sells.

For any rational person the trader expects to profit, thus the maximum expected value is the most reasonable metric to be used for evaluation [11-12]. The trader profits when they buy or sell in correspondence with the trend of the market and will deficit if they buy or sell opposite of the market trend.

Now the main challenges faced are:

1. How to effectively model the market's inherent volatile nature (whether it will go up or down) plus the hesitation of the traders when deciding what action to take (whether to buy or sell) resulting in a dual uncertainty effect.

2. How to accurately describe the interactivity between the market's volatility and the traders' hesitant actions, as the two are intertwined because the market's unpredictability hampers the traders decisive decision-making ability and in turn the collective actions taken by all the traders influences and eventually determines the closing prices of the market.

The entire process of modeling this dual uncertainty is that the market is constantly in a state where there could be infinite possibilities that could happen, basically at any given time it is impossible to know whether the market will go up or down. But the inherent uncertainty of the market has no significant meaning without any traders' involvement, and it is with traders' participation that brings the infinite possibilities of whether to take the action of buy and sell, each with a certain degree of beliefs in doing so. The uncertainty of the market, coupled with the uncertainty of the actions that can be taken by the traders results in this infinite superposed spectrum of subjective and objective possibilities.

Thus, to deal with this overlapping dual uncertainty, the quantum principle of superposition can be utilized. The first postulate of quantum mechanics is "the state of an isolated physical system is represented, at a fixed time t, by a state vector $|\psi\rangle$ belonging to a Hilbert space $\mathcal{H}$ called the state space." Thus, when something is in a superposed state, all the possible states can be expressed by a state $|\psi\rangle$, which can be expressed as a linear combination of the states of the observable as (5). Essentially a superposed state is a state where all the possible states simultaneously exist until it is observed [13-14].

$$|\psi\rangle = c_1|\psi_1\rangle + c_2|\psi_2\rangle + \cdots + c_n|\psi_n\rangle \tag{5}$$

There are corresponding observed values of $o_1, o_2, \ldots, o_n$, and once the measurement happens [15] only one of these values o_n can be observed with a probability of $|c_n|^2$, as in (6).

$$|\psi\rangle \to |\psi_n\rangle \tag{6}$$

By superposing all the possible states of the market (either going up or down) and all the possible actions that the traders can take (either buy or sell) together according to the principle of quantum superposition we are able to postulate an effective model of both the potential states of the market and the collective possible actions taken by all the traders as in (7) and (8).

$$|Q\rangle = c_1|q_1\rangle + c_2|q_2\rangle \tag{7}$$

Where $|q_1\rangle$ denoting the market trending upwards, and $|q_2\rangle$ denoting the market trending downwards. $\omega_1 = |c_1|^2$ is the objective frequency of the increase; $\omega_2 = |c_2|^2$ is the objective frequency of the decrease.

$$|A\rangle = \mu_1|a_1\rangle + \mu_2|a_2\rangle \tag{8}$$

Where $|a_1\rangle$ denotes the buy action, and $|a_2\rangle$ denotes the sell action. $p_1 = |\mu_1|^2$ are the degree of beliefs to buy; $p_2 = |\mu_2|^2$ are the degree of beliefs to sell.

(8) represents the decision-maker's potential actions that can be taken (buy or sell) in a superposed state with infinite possibilities of buy or sell with varying degree of beliefs.

Given this, the question then becomes how does the trader decide which action to take: buy or sell?

Essentially when it comes to equity trading, the trader is faced with making decisions under incomplete information, which is just a form of decision-making under uncertainty [16]; they have to be able to sift out the most possible outcome of the market (whether it will go up or down) and take the best corresponding action (buy or sell) all with the highest degree of confidence (subjective degrees of belief) at split second notice. In plain English that just means know when the market will go up and buy and know when the market will go down and sell [17].

But in many cases, this is simpler said than done, the trader needs to have good strategies to guide them in taking the best actions. The second postulate of quantum mechanics is "all measurable quantities (observables) are described by Hermitian Linear operators". In this case, the observable for traders' actions is represented by an operator (matrix) ρ as in (9a), and the matrix just means the operation of transferring a state to another state. ρ in this context is just a 2x2 matrix and serves as a projection operator, one that projects the traders' undecided mind of whether to buy or sell to a final action taken of either buy or sell. Pure state in (9a) is a state with "quantum interference" which means that the trader can't decide yet whether to buy or sell as $\mu_1\mu_2^*|a_1\rangle\langle a_2| + \mu_1^*\mu_2|a_2\rangle\langle a_1|$ in (9a). Mixed state in (9b) is the state without "quantum interference" which means that the trader either buys with degree of beliefs p_1 or sells with degree of beliefs p_2.

Essentially the whole role or function that the projection operation is that it projects a result (buy or sell) with a probability p_n, and in the case of equity trading it projects whether to buy (a_1) or sell (a_2) with subjective probability as in (9c). In other words, the projection operator projects the traders' mind to an action that the trader will take, just like how Aristotle put it "from potentiality to actuality."

$$\text{Pure state:} \quad \rho = |A\rangle\langle A| = p_1|a_1\rangle\langle a_1| + p_2|a_2\rangle\langle a_2| + \mu_1\mu_2^*|a_1\rangle\langle a_2| + \mu_1^*\mu_2|a_2\rangle\langle a_1| \tag{9a}$$

$$\text{Mixed state: } \rho' = p_1|a_1\rangle\langle a_1| + p_2|a_2\rangle\langle a_2| \tag{9b}$$

$$\text{Decision process D:} \quad \rho \xrightarrow{\text{decision}} \rho' \tag{9c}$$

From ρ to ρ' is the projection operation highlighted, it projects the pre-action superposed mindset (buy and sell) to the actual decision the trader made (either buy or sell) with degree of beliefs. We also call the projection operator ρ the logic decision tree, and this entire projection of the decision process can be seen as a transformation from pure state to mixed state as in (9c).

The entire decision-making process can also be expressed in matrix form as in (10-11):

$$\rho = \begin{bmatrix} \rho_{11} & \rho_{12} \\ \rho_{21} & \rho_{22} \end{bmatrix} \xrightarrow{\text{diagonalization}} \begin{bmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{bmatrix} \xrightarrow{\text{normalization}} \rho' = \begin{bmatrix} p_1 & 0 \\ 0 & p_2 \end{bmatrix} = p_1|a_1\rangle\langle a_1| + p_2|a_2\rangle\langle a_2| \tag{10}$$

$$|a_1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |a_2\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix};\ |a_1\rangle\langle a_1| = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, |a_2\rangle\langle a_2| = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} \tag{11}$$

During the entire decision process, the trader has to take into account the state of the market, they can't just go in blindly and make a decision, thus the probability of the market (up or down) can be described by the objective frequency ω_1 and ω_2, essentially the market up-down is approximately 50-50 (hence the market is in a random walk). Now the question becomes how can the subjective probability of P_1 and P_2 of the traders in (9) be formulated?

Sometimes traders' will 100% firmly believe that the market will go up, other times they'll only have a 60% belief that the market will go up, but every time no matter what final action the traders' decide to take they will always have a degree of beliefs ranging from 0-100% to buy or 0-100% to sell, though that doesn't necessarily mean that they will always do so believing 50-50 buy/sell, it could be 60-40, 30-70, 80-20, etc.

Just having degree of beliefs to decide whether to buy or sell is not enough, therefore we need to optimize a dynamic degree of beliefs that are a group of strategies which will guide the traders' actions in the form of an action sequence to maximize their expected value, which is in line with the rational economic agent hypothesis that would be to make the most profit on every trade conducted.

Essentially this action sequence is the traders' actions taken at any given time coupled with their degree of beliefs. If the trader believes that the market will go up with 100% degree of beliefs and the market does indeed went up and the trader bought with that 100% degree of beliefs then that is the best-case scenario that the trader should be aiming for every time. Other times it could be that the trader bought with having an 80% degree of beliefs that the market will go up and 20% that the market will go down and the market did go up, then that is a satisfactory strategy, which is also what the trader should aim for.

Basically, the so-called trading strategies produced by the action sequence is to aid the trader in buying or selling in line with the trend of the market with the greatest corresponding degree of beliefs, i.e. to buy with 95% degree of beliefs that the market will go up when the market is trending upwards. Now the remaining problem is how to find, or balance the ability to find the most satisfactory strategy that allows the trader to buy or sell with the greatest degree of beliefs when the market is going up or down respectively to make the most profit every time.

For example, if the trader is able to buy with 100% of belief that the market will go up and does so when the market indeed goes up then this is the pinnacle of the action sequence strategies, the remaining job is to attempt to get to this level of excellence every time, or as close as possible. But this is easier said than done, the market's myriads of kaleidoscopic changes significantly hampers the ability to do so, essentially the best we can do is to find the most satisfactory ones to play our hand in this "game" of strategy, the cunning interactivity between the traders and the market, we turn to evolution and Genetic Programming (GP). And it is from these satisfactory strategies formulated by GP pushes us to constantly construct the most accurate action sequences to forecast the market's trend.

This is how we "deal" with this second challenge of the interactivity between the market and the traders by using GP, an algorithm that draws on the principles of Darwinian natural selection and evolution [18]. GP uses random crossover, selection, and mutation to formulate an executable program that solves problems accordingly [19-20]. First by randomly generating a certain number of individuals that comprise of a population, the algorithm obtains the fitness of each individual in the group and then by utilizing the principles of natural evolution for a number of generations it will optimize a most "satisfactory" solution to be used. The fittest ones that survive are the ones utilized, in line with the theory of natural evolution that states life has evolved through generations of selection, mutation, and crossover, the ones most adapted to the environment survive long enough to pass their genes off to the next generation [21]. The GP algorithm is shown in Algorithm 1.

Input:

- Historical dataset $\{(t_k, s_k), k = 0, \cdots, N\}$ (each sample consists of an equity's time and closing price);
- Setting:
  (1) Operation set F;
  (2) Dataset T;
  (3) Crossover Probability = 70%; Mutation probability = 5%.

Initialization:

- Population: randomly create 300 individuals.

Evolution:

Loop: for i = 0 to 80 generations.

a) Calculate fitness for each individual based on the historical dataset;
b) According to the quality of fitness:
   i. Selection: selecting parents.
   ii. Crossover: generate a new offspring using the roulette

algorithm based on crossover probability.

iii. Mutation: randomly modify the parent based on mutation probability.

Output:

- An individual of the best fitness.

Algorithm 1. GP Algorithm

The pure state (2x2 density matrix) ρ can be approximately constructed from eight basic quantum gates as (12).

$$\left\{\begin{array}{c} H=\frac{1}{\sqrt{2}}\begin{bmatrix}1 & 1\\ 1 & -1\end{bmatrix}\ X=\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}\ Y=\begin{bmatrix}0 & -i\\ i & 0\end{bmatrix}\ Z=\begin{bmatrix}1 & 0\\ 0 & -1\end{bmatrix} \\ S=\begin{bmatrix}1 & 0\\ 0 & i\end{bmatrix}\ D=\begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix}\ T=\begin{bmatrix}1 & 0\\ 0 & e^{i\pi/4}\end{bmatrix}\ I=\begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix} \end{array}\right\} \tag{12}$$

The operation set and the dataset for GP are as:

(1) Operation set $F = \{+,*,//\}$;

(2) Dataset $T = \{H, X, Y, Z, S, D, T, I\}$.

In order to find the most satisfactory set of strategies QxEAI optimizes the best actions by using logic trees (pure state ρ). The logic tree is essentially a decision tree that guides which strategies to take with corresponding actions. At any given point (time), the expected value under the current environment and the corresponding actions can be represented as (13c).

$$\rho_{market} = \omega_1|q_1\rangle\langle q_1| + \omega_2|q_2\rangle\langle q_2| \tag{13a}$$

$$\rho_{action} = p_1|a_1\rangle\langle a_1| + p_2|a_2\rangle\langle a_2| \tag{13b}$$

$$\rho_{market} \otimes \rho_{action} = \omega_1 p_1|\langle q_1||a_1\rangle|^2 + \omega_1 p_2|\langle q_1||a_2\rangle|^2 + \omega_2 p_1|\langle q_2||a_1\rangle|^2 + \omega_2 p_2|\langle q_2||a_2\rangle|^2 \tag{13c}$$

Where (13a) is the market observable operator; (13b) is the traders' actions observable operator; and (13c) is the composite system of the market and the traders' actions. In (13c), the first term means that the market is going up and the trader buys (this is a reward), the second term means that the market is going up and the trader sells (this is a punishment), the third term means that the market is going down and the trader buys (punishment), and the fourth term is that the market is going down and the trader sells (reward). According to economic theory, anytime someone makes a "decision", whether acting in their own self-interest or by other considerations of strategy there is always an expected value [22-24]. The expected value of each individual one is the possible scenarios of what the outcome could be paired with the state of what is being observed, as in (14):

$$EV_t = \begin{cases} \omega_1 p_1 s_{t,t-1}, \text{trend is up and trader believes so with } p_1 \\ -\omega_1 p_2 s_{t,t-1}, \text{trend is up and trader doesn't so with } p_2 \\ -\omega_2 p_1 s_{t,t-1}, \text{trend is down and trader doesn't with } p_1 \\ \omega_2 p_2 s_{t,t-1}, \text{trend is down and trader believes with } p_2 \end{cases} \tag{14}$$

Where $s_{t,t-1}$ is the absolute difference between the closing price of the current point and the previous point.

The fitness function of the logic tree is the sum of all the expected value of all the individual actions as (15):

$$fitness_{logicTree} = \sum_{t=1}^{n} EV_t \tag{15}$$

The purpose the fitness function is essentially an incentive system of reward and punishment. Before a decision is made and an action is chosen, there are the four potential outcomes that exist. Of course, only one can happen, basically one out of the four scenarios of the expected value in (14). Therefore, if the market is trending upwards and the belief is that it's trending upwards, that results in a reward. But if the market is trending downwards and the belief is that it's trending upwards, then a punishment is concurred. Vice versa with the other two scenarios. By learning historical data, the more rewards that are reaped then the more accurate chance there is of predicting the next trend. This also allows for no presumptions of the trend, the more times the right trend is "guessed" correctly the best strategies and actions are effectively evolved as a result. Generation after generation of evolution, the best strategy naturally arises, which is the ultimate goal of the fitness function. The best strategy that has evolved by natural selection is the one that can be utilized for future forecasting.

3 Results

For trading the Dow Jones Index Futures, QxEAI is no crystal ball that you can peer into and see the future closing prices. The best QxEAI can do is to assist a trader by providing an action sequence to help decide whether to buy or sell in response to if the market might go up or down in the future. In a sense, QxEAI is an AI trading assistant, a tool that doesn't actually make decisions and physically trade for a real-world human trader, but to provide important information and metrics to assist in making the best decisions possible.

3.1 Datasets

Four weeks of data from October 4th, 2024 to November 1st, 2024 were downloaded on November 2nd, 2024 and was used as training data. The data was trained consecutively on November 2nd, 2024 and November 3rd, 2024 and produced a forecast outcome for the week of November 4th, 2024 to November 8th, 2024. This ensures that QxEAI is able to produce its outcomes before actual trading data is recorded. After the forecasted trading week ends, the predicted results can then be compared to the actual recorded results at the end of the week, therefore guaranteeing a 100% real-world forecast with no possibility of knowing what has happened already beforehand. Following this strict procedure, QxEAI's assisting decision-making abilities are fully reflected.

3.2 Applications results

3.2.1 Train and Forecast

On both days, the training parameters were set the same, with 3 agents and a training frequency of 1000 repetitions. Once training concludes QxEAI obtains an optimized logic decision tree to produce forecasts. Both trainings produced 6 different possible outcomes each (total 12), which were all taken together and provided as reference metrics to make the final forecast "decision". Figure 1 shows the training results on November 2nd and Figure 2 shows the training results on November 3rd.

In both figures, the blue line is the recorded historical data, the closing prices of the Dow Jones Index from October 4th to November 1st, 2024. The yellow line are the most optimal calculated prices comprising of the fitted line of the data as produced by QxEAI through machine learning the historical data according to the logic tree utilized.

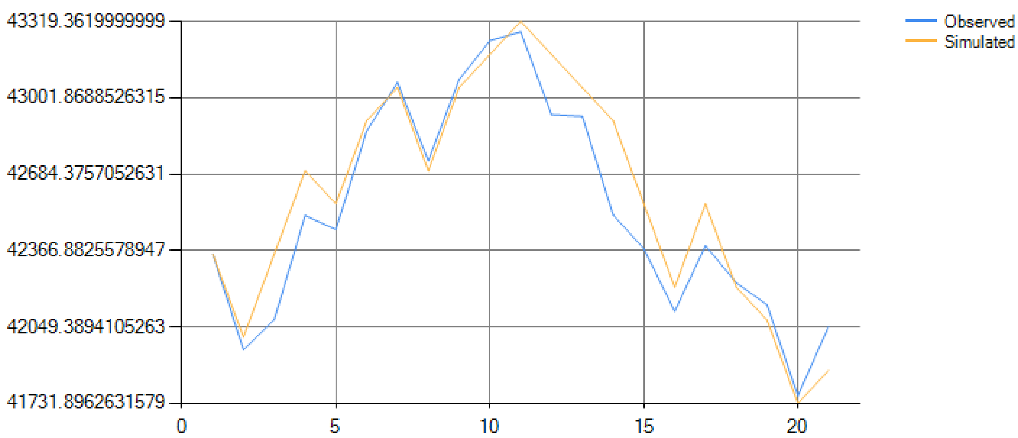

Figure 1: Results of first training on November 2^{nd}, 2024.

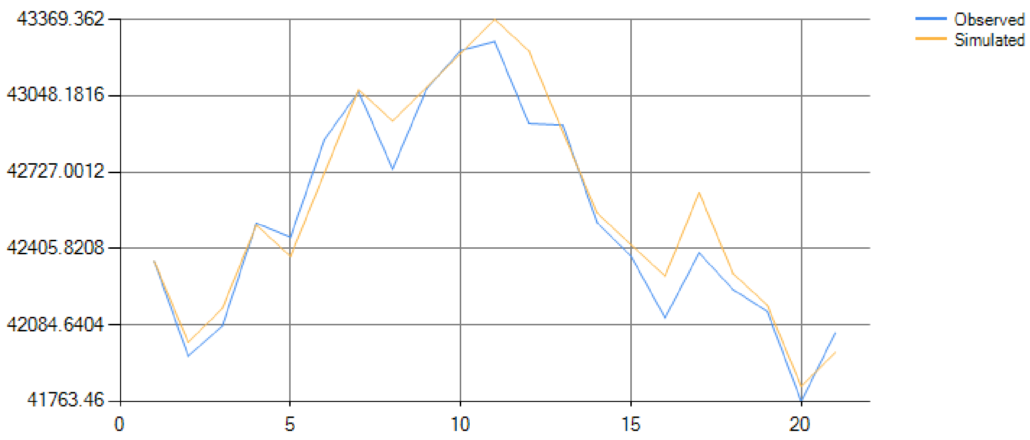

Figure 2: Results of second training on November 3rd, 2024.

Table 1 shows the 12 total possible forecast outcomes from the trainings on November 2nd, 2024 and November 3rd, 2024.

Date	Trend	Train 1 (Nov 2)						Train 2 (Nov 3)					
11/04/2024	1	1	1	0	0	0	1	1	0	0	0	1	1
11/05/2024	0	0	0	0	1	0	0	1	0	0	0	0	0
11/06/2024	0	0	0	0	0	0	0	0	0	0	0	0	0
11/07/2024	1	1	0	0	1	0	1	1	1	0	0	1	0
11/08/2024	0	0	0	1	0	0	0	0	0	1	0	0	1

Table 1 12 possible forecast outcomes training table

The actual observed trend of the market (up as 0 or down as 1) is displayed in the trend column, while the Train 1 and Train 2 columns show the predicted outcomes (whether the market will go up or down) from the days the data was trained on November 2nd and 3rd respectively. 0 represents the logic decision tree "believes" that the market will go up and 1 represents the logic decision tree "believes" that the market will go down, together they represent the action sequence calculated by QxEAI's logic trees; for example, 1 0 0 1 0 from the first possible forecast from training one represents the logic decision tree "believes" the future market will be Down, Up, Up, Down, Up.

3.2.2 Assisting Trading

The two groups of 6 forecasts produced from the two separate trainings are analyzed according to majority rules. Taking all 12 forecast outcomes into account, the one that is decided by majority rules is chosen as one action sequence to guide the final decision whether to buy or sell for each coming day from 11/04/2024 to 11/08/2024, as shown in Table 2.

Date	Trend of DJIA	Action sequence by QxEAI
11/04/2024	Down	Buy
11/05/2024	Up	Buy
11/06/2024	Up	Buy
11/07/2024	Down	Sell
11/08/2024	Up	Buy

Table 2 Final action sequenced provided by logic tree table

For the action sequences produced for the days of November 5th, 6th, and 8th, the majority are that the trend will go up, thus following majority rules the corresponding action to take is naturally to buy. For example, 10 out of the 12 outcomes produced by QxEAI for November 5th are that it believes that the market will go up and only 2 that it will go down. Therefore, it is most reasonable for QxEAI to believe that the market will go up, leading it to decide to buy on November 5th. The same applies for November 6th and 8th to buy, as the majority of the outcomes points to buy, leading QxEAI to do so.

However, for November 4th and 7th, the 12 trainings produced over the course of two days each produced 6 each, resulting in a tie of up and down, therefore there is no majority rule to decide whether to buy or sell. In this case, QxEAI will then have to randomly chose one or the other, here it chose to buy on the 4th and to sell on the 7th.

By taking a look at the actual results that were recorded for the following week of November 4th, 2024 to November 8th, 2024 and comparing it with the forecast results, QxEAI got 4 out of the 5 actions right, therefore its odds are 80%. If QxEAI chose to sell for both the 4th and 7th then it would have gotten them all right and the odds would've been 100%. If QxEAI chose opposite of what it ended up choosing this time, then the odds would've still been 80%. And if QxEAI had chosen the worst-case scenario of buy for both the 4th and 7th, the odds still would've been 60%. Thus, no matter what, whatever outcome QxEAI ends up with, best case 100%, worst case 60%, and an equal split of 80% odds, making the average of the possible odds still 80%.

Essentially, the purpose of downloading the four weeks of training data (10/04/2024-11/01/2024) on Saturday November 2nd and consequently training it on the day of and the following day is to produce a forecast outcome before the next week of trading (11/04/2024-11/08/2024) commences on Monday November 4th, therefore eliminating the chance of manual adjustment to account for the margin of error or deviation from the data results by means of a validation forecast where the results have already been recorded but is purposefully ignored and then later compared to the original data, as the forecast outcome for November 4th to 8th was produced before trading commenced that week.

Additionally, all train and forecast results produced by QxEAI are non-altered data, the fitting data produced from training and the forecast data have not been manually adjusted at all. The results included in this paper are the original produced outcomes as calculated by QxEAI with the respective proposed methodology of utilizing quantum superposition and Darwinian natural selection, and were not tampered with or manipulated for any other purposes to influence the results produced and conclusions reached other than for the purpose of academic research.

4 Discussion

Majority of traditional methods tend to focus on breaking down forecasting into three steps: 1) find a trend, 2) apply an interval cycle, and 3) eliminate external noise (uncertainty) as much as possible. If the said dataset can't be broken down into these three parts, then that dataset is treated as an unpredictable data series filled with random external noise that can't be reduced. Basically, traditional methods strive to reduce or completely eliminate external noise (uncertainty) by treating it as a bad thing to find a certain possible trend.

Compared to traditional methods, our methodology doesn't treat external noise (uncertainty) necessarily as a bad thing, we don't strive to reduce or eliminate noise and uncertainty but quite the contrary, we attempt to utilize uncertainty to find valuable information from the constantly changing environment to formulate a trend. Thus, we don't set out with the mindset of attempting to break up the data into trend, interval cycle, and noise, instead we seek to embrace uncertainty as a factor to help us find possible outcomes.

Now the challenge is how to deal with the dual uncertainty of the environment and possible actions that can be taken. Our methodology does this by using a unified complex Hilbert space to represent this dual uncertainty. The remaining problem is that how do we find the best one from the infinite possibilities arises. Our methodology does this by utilizing the economic principle of maximum expected value as the evaluation metric and using QxEAI to find the most satisfactory model by optimizing the best one from all the possible models, with the maximum expected value used as the fitness function.

QxEAI applies quantum superposition to construct all the possible solutions, while GP optimizes the best one by the fitness function.

While uncertainty is inherently part of the stock market, overly emphasizing on trying to eliminate the noise that uncertainty brings by all means overlooks the fact that without risk there is no reward. Just like the saying no pain, no gain, for the stock market the greater the risk the greater the reward. And it's this inherent uncertainty that brings the excitement of trading, because by doing so, we the traders determine the closing prices and fluctuations of the market but this in turn then clouds our judgement and hampers our decision-making ability in this constant unpredictable "game" of back and forth with the market that makes equity trading more exciting and fruitful when we reap the rewards associated with it.

5 Conclusion

We used a complex Hilbert space to describe the double existing uncertain nature and then used GP to optimize a set of satisfactory strategies in the form of an action sequence to guide traders to take actions for trading. We have shown by training four weeks of stock market data and then producing two groups of 6 possible forecast outcomes on the Saturday and Sunday of a weekend that we can produce a somewhat very accurate forecast for the upcoming week of trading. By training with 3 agents and a frequency of 1000 training repetitions each day, followed by producing 6 possible forecast outcomes each to evaluate all together with majority rules to "chose" the final action sequence to guide which actions (buy or sell) to take, QxEAI is able to produce a forecast with odds of 80% for the following week of trading.

Now of course, with any given model, ours included, again it is no crystal ball, one where you can peer into and clearly see what will happen in the future. Though using our methodology in this case, QxEAI was able to produce a forecast with 80% odds for this specific time, but if the market somehow turns on its head in the future, then in a case like that, this would hamper QxEAI's forecast significantly. However, the training parameters with QxEAI can be set and changed according to will, thus if the number of agents trained and the frequency of training repetitions are increased, QxEAI may avoid a situation like that from happening.

Building off of our previous work where we attempted to show that QxEAI could automate the time series forecasting process and produce probabilistic forecasts [25-26], further research will include increasing the number of agents trained and increasing the frequency of the training repetitions to see if more stable and accurate results can be produced. Hypothetically the more time that is used for training, i.e. more agents are used and the frequency of training repetitions are set, then the results produced should be more stable and accurate to some degree. Lastly, given the flexibility of our methodology, further research can include using QxEAI in the same way for any discrete time series to be a "AI machine assistant decision-maker".

Declarations

Data availability. The data that was used for the purposes of this study are publicly available for download from the Federal Reserve Economic Data website. The authors confirm that the data supporting the findings of this study are available within the article and its supplementary materials. All data that support the findings of this study are also available from the corresponding author, K.X., upon reasonable request.

Authors' contributions. All authors conducted the research and contributed to the development of the model. L.X. contributed to the research from the aspects of machine learning, decision theory, and wrote the code. K.X. wrote the manuscript and did data analysis. All authors reviewed the manuscript.

Conflict of interests. The authors declare that they may have relevant commercial interests in the sales and profits of the commercial tool they developed for their corporation XINVISIONQ, INC. that is based off the methodology presented in this paper but this did not, in any way influence the legitimacy and authenticity of the results and conclusions presented in this paper in accordance to the ethical requirements of academic research. The authors have no other conflicting academic interests to declare as the methodology formulated in this paper are the authors' original work.

Funding. This work received no funding.

References

[1] Jan G. De Gooijer, Rob J. Hyndman, 25 years of time series forecasting, International Journal of Forecasting, Volume 22, Issue 3, 2006, Pages 443-473.

[2] Box, G. E. P., & Jenkins, G. M. (1970). Time series analysis: Forecasting and control. San Francisco: Holden Day (revised ed. 1976).

[3] Holt, C. C. (1957). Forecasting seasonals and trends by exponentially weighted averages. O.N.R. Memorandum 52/1957, Carnegie Institute of Technology. Reprinted with discussion in 2004. International Journal of Forecasting, 20, 5–13.

[4] Winters, P. R. (1960). Forecasting sales by exponentially weighted moving averages. Management Science, 6, 324–342.

[5] Gardner Jr., E. S. (1985). Exponential smoothing: The state of the art. Journal of Forecasting, 4, 1–38.

[6] Hyndman, R. J., & Khandakar, Y. (2008). Automatic Time Series Forecasting: The forecast Package for R. Journal of Statistical Software, 27(3), 1–22.

[7] Alsharef, A., Aggarwal, K., Sonia et al. Review of ML and AutoML Solutions to Forecast Time-Series Data. Arch Computate Methods Eng 29, 5297–5311 (2022).

[8] David Salinas, Valentin Flunkert, Jan Gasthaus, Tim Januschowski, DeepAR: Probabilistic forecasting with autoregressive recurrent networks, International Journal of Forecasting, Volume 36, Issue 3, 2020, Pages 1181-1191.

[9] Wastergaard, G.; Erden, U.; Mateo, O.A.; Lampo, S.M.; Akinci, T.C.; Topsakal, O. Time Series Forecasting Utilizing Automated Machine Learning (AutoML): A Comparative Analysis Study on Diverse Datasets. Information 2024, 15, 39.

[10] Meisenbacher, S., Turowski, M., Phipps, K., Rätz, M., Müller, D., Hagenmeyer, V., & Mikut, R. (2022). Review of automated time series forecasting pipelines. WIREs Data Mining and Knowledge Discovery, 12(6), e1475.

[11] Administrative Behavior Herbert A. Simon. The Free Press, 1997.

[12] Matthias Klaes, Esther-Mirjam Sent; A Conceptual History of the Emergence of Bounded Rationality. History of Political Economy 1 March 2005; 37 (1): 27–59. doi: https://doi.org/10.1215/00182702-37-1-27

[13] Feynman, Richard P. The Feynman Lectures on Physics. Volume III, Quantum Mechanics. Basic Books, 2015.

[14] Dirac, P.A.M. The Principles of Quantum Mechanics; Oxford University Press: Oxford, UK, 1958.

[15] Von Neumann, J. Mathematical Foundations of Quantum Theory; Princeton University Press: Princeton, NJ, USA, 1932.

[16] Xin, L.; Xin, H. 2023 Decision-making under uncertainty—A quantum value operator approach. Int. J. Theor. Phys., 62, 48. (doi:10.1007/s10773-023-05308-w)

[17] Xin, L.; Xin, K.; Xin, H. 2023 On Laws of Thought—A Quantum-like Machine Learning Approach. Entropy, 25, 1213. (doi:10.3390/e25081213)

[18] Holland, J. Adaptation in Natural and Artificial System; University of Michigan Press: Ann Arbor, MI, USA, 1975.

[19] Koza, J.R. Genetic Programming, on the Programming of Computers by Means of Natural Selection; MIT Press: Cambridge, MA, USA, 1992.

[20] Koza, J.R. Genetic Programming II, Automatic Discovery of Reusable Programs; MIT Press: Cambridge, MA, USA, 1994.

[21] Darwin, Charles, 1809-1882. On the Origin of Species by Means of Natural Selection, or Preservation of Favoured Races in the Struggle for Life. London: John Murray, 1859.

[22] Von Neumann, J.; Morgenstern, O. Theory of Games and Economic Behavior; Princeton University Press: Princeton, NJ, USA, 1944.

[23] Smith, Adam. An Inquiry into the Nature and Causes of the Wealth of Nations. London: Harriman House, 2007.

[24] Mankiw, N Gregory. Principles of Microeconomics 10th Edition. Cengage Learning, 2023.

[25] Kevin Xin, Lizhi Xin. QxEAI - Automated probabilistic forecasting with Quantum-like evolutionary algorithm, 06 May 2024, PREPRINT (Version 1) available at Research Square [https://doi.org/10.21203/rs.3.rs-4366243/v1]

[26] Xin, Kevin and Lizhi Xin. “QxEAI: Quantum-like evolutionary algorithm for automated probabilistic forecasting.” (2024).