Towards More Efficient Trading: Integration of Deep Q-Network and Algorithmic Trading Techniques

April 15, 2025

Abstract

The introduction of algorithmic trading in financial markets may be traced to the 1930s, initially designed to efficiently and accurately automate purchases of multiple financial instruments. Technological advances have been integrated into all aspects of human activity in modern times. For algorithmic trading, machine learning techniques have emerged, such as Q-Learning, a reinforcement learning algorithm known for its proficiency in making sequential decisions in uncertain environments. Its applicability is particularly relevant to market dynamics. This work presents an innovative integrated approach to optimizing trading strategies in financial markets, taking advantage of Q-learning, optimization techniques, and algorithmic trading. The proposed model seeks to maximize risk-adjusted returns, taking into account the strengths of each methodology. By combining the predictive capabilities of Q-learning, optimization techniques, and fast algorithmic trading automation, this model presents a solid framework for designing efficient and profitable trading strategies. The implemented methodology provides positive expected values with a successful ratio, after optimizing Q-Learning parameters for the backtesting strategy.

Data Source

The data used in this project is provided by TradingTech, a company known for its high-quality financial data. The data consists of tick data, which is the finest level of detail in trading records. Each tick represents a trade and includes information such as the price, a change in price, and the volume of the trade. This high-resolution data allows for a detailed analysis of market dynamics and the development of sophisticated trading strategies.

Tick Data Description

The tick data provided by TradingTech consists of several variables, each of which provides valuable information about each trade. Here is a description of each variable:

- **Date and Time:** The date and time of the trade, down to the millisecond. This is represented in the format DD/MM/YYYY, HH:MM:SS.mmm. For example, '31/05/2023,00:00:00.681' represents a trade that occurred on May 31, 2023, at 00:00:00.681.
- **Bid Price:** The highest price that a buyer is willing to pay for the asset.
- **Bid Volume:** The volume of futures that a buyer is willing to pay.
- **Ask Price:** The lowest price that a seller is willing to accept for the asset.
- **Ask Volume:** The volume of futures that a buyer is willing to pay.

This tick data provides a detailed view of the market dynamics, allowing for the development of sophisticated trading strategies. The high-resolution nature of the data, with timestamps down to the millisecond, allows for the analysis of market microstructure and the identification of short-term trading opportunities. However, the variables that are going to be used for the Analysis are:

- Date
- Time
- Bid Price
- Ask Price

The next image is a description of how Tick Data looks:

```
01/02/2022,00:00:00.580000000,1.12490,14,1.12500,38,E,
01/02/2022,00:00:00.586000000,1.12490,14,1.12500,48,E,
01/02/2022,00:00:00.597000000,1.12490,13,1.12500,48,E,
01/02/2022,00:00:00.609000000,1.12490,12,1.12500,48,E,
01/02/2022,00:00:00.645000000,1.12490,12,1.12500,50,E,
01/02/2022,00:00:01.138000000,1.12490,10,1.12500,50,E,
```

Exploratory Data Analysis (EDA)

Before applying machine learning techniques to the tick data, an exploratory data analysis (EDA) is conducted. EDA is a crucial step in any data analysis project as it allows us to understand the underlying structure of the data, identify outliers, detect anomalies, test assumptions, and check for missing or erroneous data.

The EDA process for this project involves several steps. First, the data is cleaned and preprocessed to handle any missing or erroneous data. This includes dealing with missing values and removing any outliers that could potentially skew the results of the analysis.

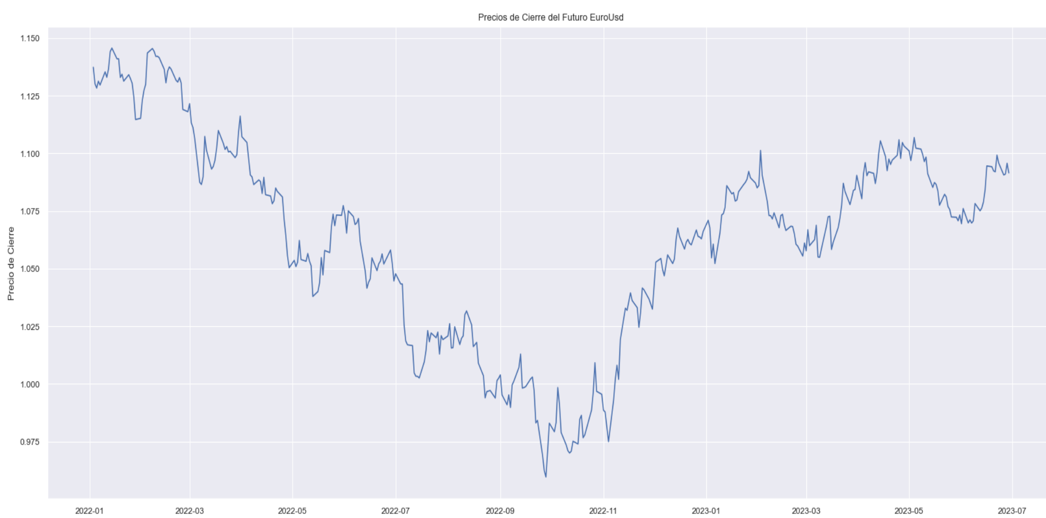
Next, various statistical measures are calculated to get a sense of the distribution and variability of the data. This includes measures such as the mean, median, standard deviation, and range of trading prices and volumes.

Finally, visualizations are created to help understand the relationships between different variables in the data. This includes scatter plots, histograms, and time series plots. These visualizations can provide valuable insights into the market dynamics and can help in the development of trading strategies.

The insights gained from the EDA process are then used to inform the feature engineering and model selection steps of the project.

Graph Analysis

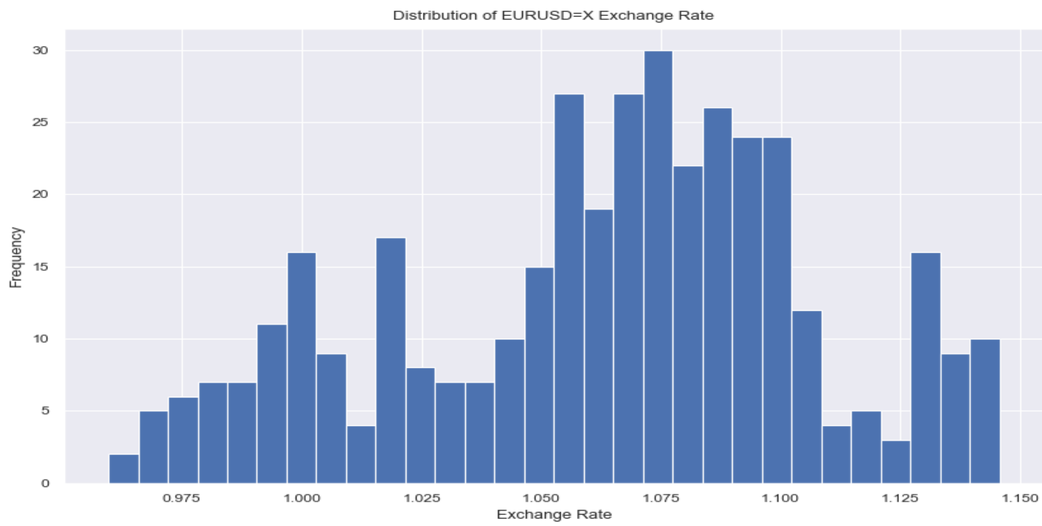
Graph 1: Close Prices



Interpretation: General trend - The chart shows a clear bearish trend for most of the analyzed period. This means that the euro has depreciated against the US dollar. Investors were willing to pay fewer dollars per euro.

Volatility: - Despite the general bearish trend, the price has experienced periods of volatility, with significant fluctuations in both directions. This suggests that the market has been sensitive to economic and political events that have affected the relationship between the euro and the dollar.

Graph 2: Distribution

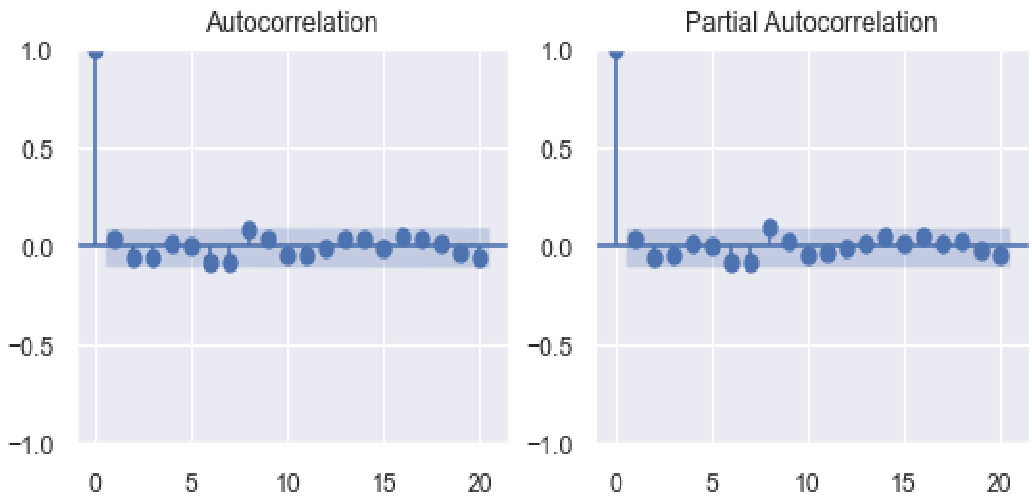


Interpretation: Tendencia general: - Bajista. El precio del euro frente al dólar ha ido disminuyendo a lo largo del período analizado, comenzando en 1,13 y llegando a 1,05 en marzo de 2023.

Niveles de soporte y resistencia:

- Soporte: Se identifican niveles de soporte en 1,04, 1,05 y 1,06. Estos niveles han actuado como "piso" en varias ocasiones, evitando que el precio caiga aún más. - Resistencia: Se observan niveles de resistencia en 1,08 y 1,10. Estos niveles han frenado el ascenso del precio en algunos momentos, actuando como "techo".

Graph 3: AutoCorrelation



Interpretation: Función de Autocorrelación (ACF):

- Retraso 0: La autocorrelación es siempre 1 en el retraso 0, ya que un valor siempre está perfectamente correlacionado consigo mismo. - Retrasos Posteriores: Los valores de autocorrelación en los retrasos posteriores son cercanos a cero y se mantienen dentro del intervalo de confianza (la banda azul sombreada).

Esto sugiere que no existe una correlación lineal significativa entre los valores de la serie temporal y sus valores pasados a partir del retraso 1. En otras palabras, el pasado de la serie no es útil para predecir su futuro de manera lineal.

Función de Autocorrelación Parcial (PACF):

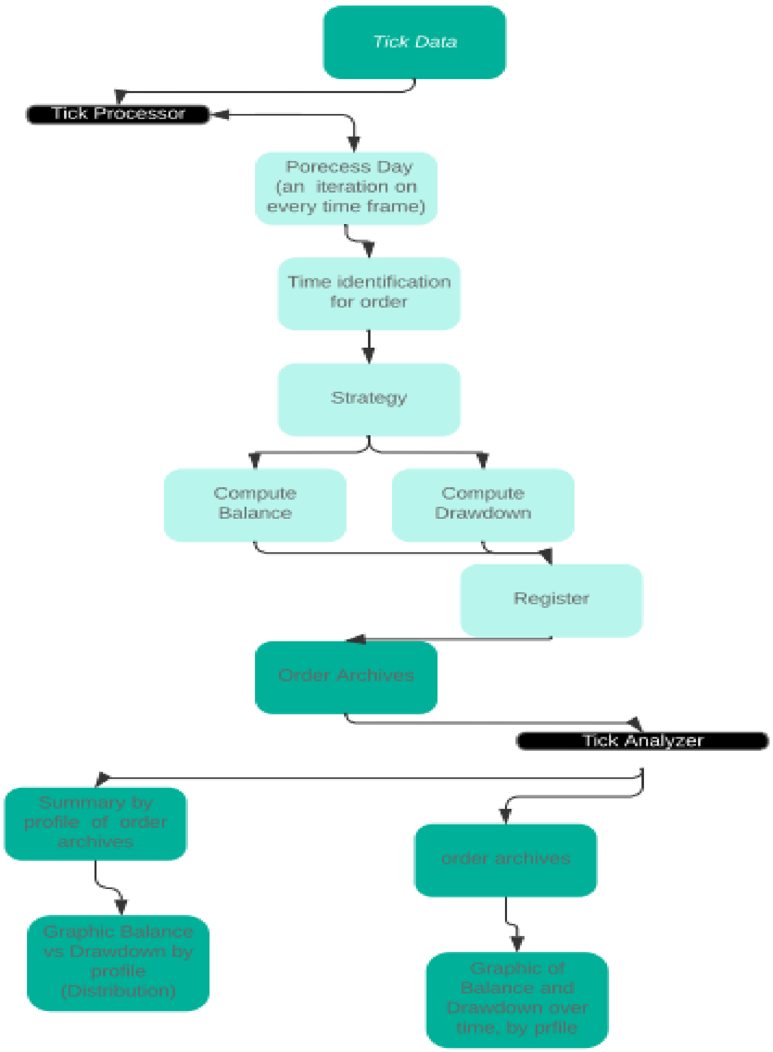
- Retraso 0: Similar a la ACF, la autocorrelación parcial es 1 en el retraso 0. - Retraso 1: Se observa un pico significativo en el retraso 1, lo que indica que el valor de la serie en un momento dado está significativamente correlacionado con el valor del período anterior, después de eliminar el efecto de los valores intermedios. - Retrasos Posteriores: Los valores de autocorrelación parcial caen dentro del intervalo de confianza a partir del retraso 2. Esto sugiere que, después de controlar por el retraso 1, no hay correlaciones parciales significativas con retrasos mayores.

Backtesting

As a guiding framework for our project, we have chosen to implement backtesting. Backtesting involves simulating a trading strategy on historical data to evaluate its performance and viability. The primary objective is to gain insights into how the strategy would have performed in the past, thereby providing a basis for assessing its potential effectiveness in future scenarios.

To achieve this, we will utilize the historical data at our disposal to execute the strategy and meticulously record the outcomes. This process will encompass several steps, including the selection of appropriate historical data, the formulation and coding of the strategy, the execution of the backtesting process, and the subsequent analysis of the results.

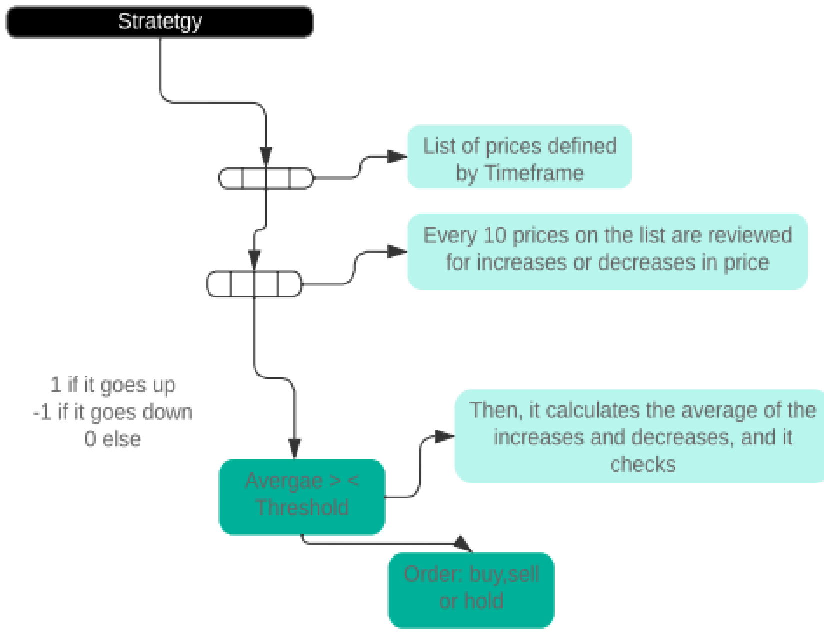
The analysis will focus on various performance of different Strategies. A crucial variable in our backtesting strategies is the timeframe. The timeframe dictates the intervals at which the strategy is applied. For each data file processed, the algorithm checks if the time specified by the timeframe has elapsed. If the designated timeframe has passed, the strategy is then invoked. This ensures that the strategy is evaluated at consistent and meaningful intervals, reflecting realistic trading conditions.



The Order Archives, produced in Tick Processor,are made in the following way:

- Column 0: **Execution ID**
- Column 1: **Threshold -** Value based on dictionary
- Column 2: **Decision -** Buy or Sell
- Column 3: **Volume of Buy or Sell -** 1
- Column 4: **Bid Price**
- Column 5: **Ask Price**
- Column 6: **Balance**

- Column 7: **Drawdown**



The next image is an example of an Order Archive:

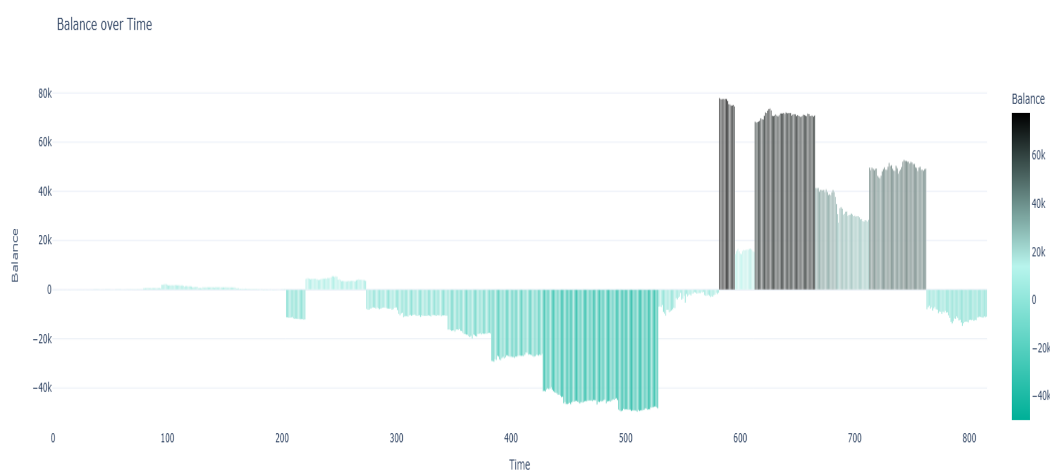
```
0:F:buy:1:1.051:1.0511:-9.999999999998899e-05:0:
1:D:sell:1:1.0513:1.0515:-2.220446049250313e-16:0:
2:C:buy:1:1.0512:1.0513:0.0:0:
3:H:buy:1:1.0521:1.0522:0.0008000000000001339:0:
4:A:buy:1:1.0525:1.0526:0.0015000000000000568:0:
5:C:sell:1:1.0507:1.0509:-0.0041999999999999815:0.005700000000000038:
6:D:sell:1:1.0493:1.0494:-0.006900000000000794:0.008400000000000851:
7:B:buy:1:1.0476:1.0477:-0.0086999999999993:0.0101999999999987:
8:G:buy:1:1.0459:1.046:-0.0121999999999989:0.013700000000000045:
9:D:buy:1:1.0458:1.0459:-0.0125999999999945:0.014100000000000001:
10:C:sell:1:1.0454:1.0455:-0.0142999999999757:0.0157999999999814:
11:C:buy:1:1.0482:1.0484:-0.0064999999999995:0.0157999999999814:
```

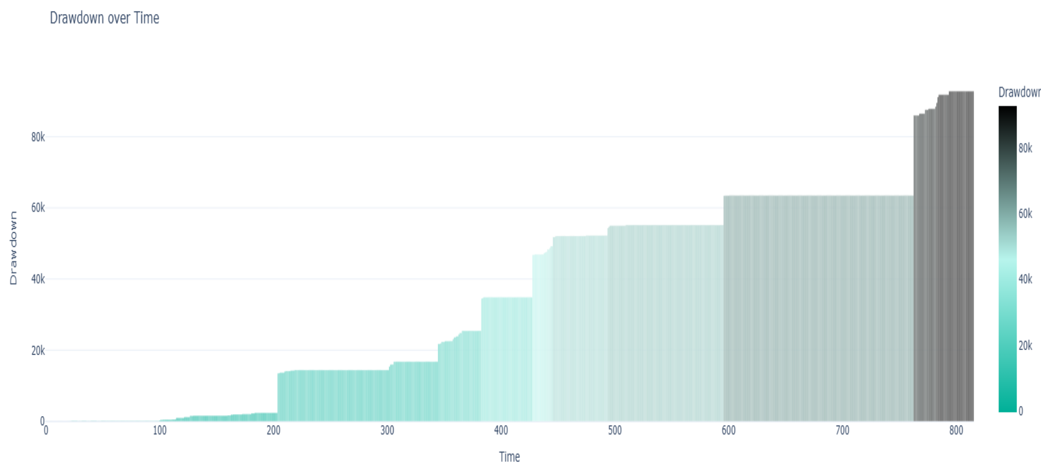
The challenge lies in processing and analyzing massive amounts of data in real-time while maintaining computational efficiency. This necessitates the implementation of robust algorithms capable of handling large-scale data structures without compromising performance. This is particularly crucial for strategies that rely on continuous data streams and require prompt calculations to generate timely signals. To effectively manage large-scale real-time data processing. For Tick Analyzer the data is distributed in two parts: Graph for a profile look and Graph for a general look over profile : Balance (X) vs Drawdown (Y) vs Profile (Z).

This is how a Order Archive looks:

```
35:B:buy:1:1.1274:1.1275:18.75000000000626:38.7499999999295
36:A:buy:1:1.1278:1.1279:37.5000000000042:38.7499999999295
37:B:sell:1:1.1282:1.1283:61.25000000001546:38.7499999999295
38:D:sell:1:1.128:1.1282:27.499999999958114:38.7499999999295
39:A:sell:1:1.1281:1.1283:28.750000000016264:38.7499999999295
40:A:buy:1:1.1279:1.128:46.2500000000878:38.7499999999295
41:A:sell:1:1.1282:1.1283:56.25000000001601:38.7499999999295
42:C:sell:1:1.1281:1.1283:26.250000000019313:38.7499999999295
43:E:sell:1:1.1283:1.1284:53.750000000016286:38.7499999999295
44:A:buy:1:1.1281:1.1282:52.50000000001642:38.7499999999295
45:C:sell:1:1.128:1.1281:49.999999999950084:38.7499999999295
46:B:buy:1:1.1283:1.1284:48.75000000001406:38.7499999999295
47:B:sell:1:1.1282:1.1283:46.25000000001434:38.7499999999295
0:I:buy:1:1.09:1.0901:45.000000000014474:38.7499999999295
1:C:buy:1:1.0904:1.0905:48.75000000001406:38.7499999999295
2:A:buy:1:1.0903:1.0904:45.000000000014474:38.7499999999295
3:I:sell:1:1.0895:1.0896:13.750000000009589:47.50000000000587
4:C:buy:1:1.0908:1.0909:45.000000000011696:47.50000000000587
5:C:sell:1:1.0904:1.0906:-3.7499999999829337:64.9999999999984
6:E:sell:1:1.0908:1.091:3.7500000000162403:64.9999999999984
7:B:sell:1:1.0914:1.0915:43.75000000001184:64.9999999999984
8:B:buy:1:1.0912:1.0913:42.50000000001197:64.9999999999984
```

The next graphs are some examples of a Drawdown, Balance vs Time for a single Profile



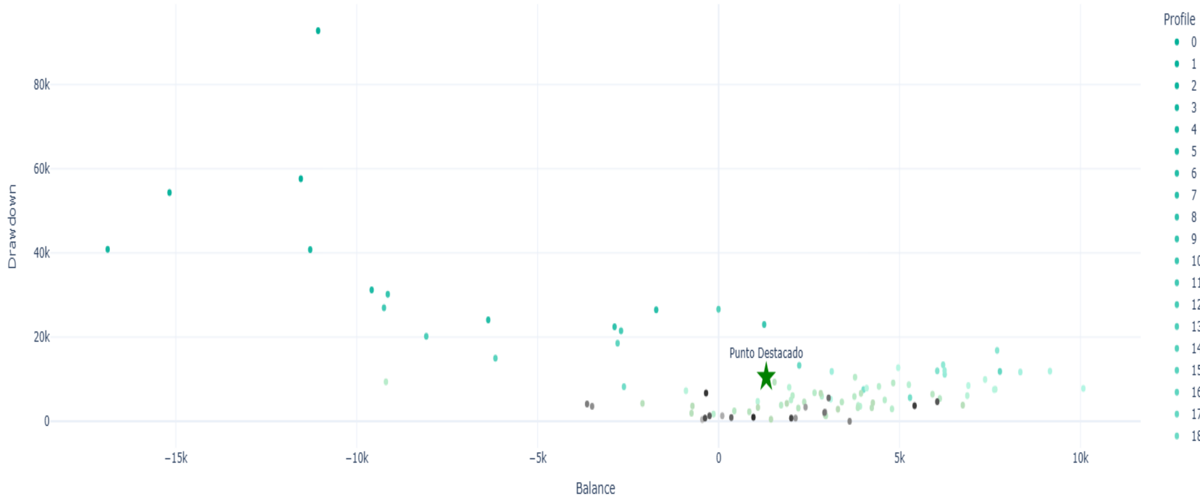


Following the backtesting module execution, a comprehensive analysis and evaluation of the results are crucial. This stage involves examining the performance of each profile generated during backtesting. It is anticipated that profiles will exhibit varying degrees of success, with some yielding positive balance and drawdown figures, while others may display a combination of both positive and negative outcomes. This variability in performance across profiles is a natural consequence of backtesting, as different profile configurations can lead to diverse levels of effectiveness in different market conditions. Taking into account that the most important parameter to choose the best profile is:

$$Balance/Drawdown = profitfactor$$

these are the best and the worst profile, based on profit factor:

- **Best profit factor :** Balance : 3621 / Drawdown : 10 = 362.1
- **Worst profit factor:** Balance : -11.06k / Drawdown : 92.79k = -0.1191



Centroid (Highlight Point):

Observation: The highlight point, representing the centroid of all profiles, is situated near the zero balance axis and in a region of relatively low drawdown.

Interpretation: The centroid indicates that, on average, the algorithm tends to operate with a balance close to zero and a moderate drawdown. This suggests that, considering all timeframes, the algorithm's strategy tends to be neutral in terms of net gain/loss, with a moderate risk exposure.

Conclusions:

Short Timeframe Strategy:

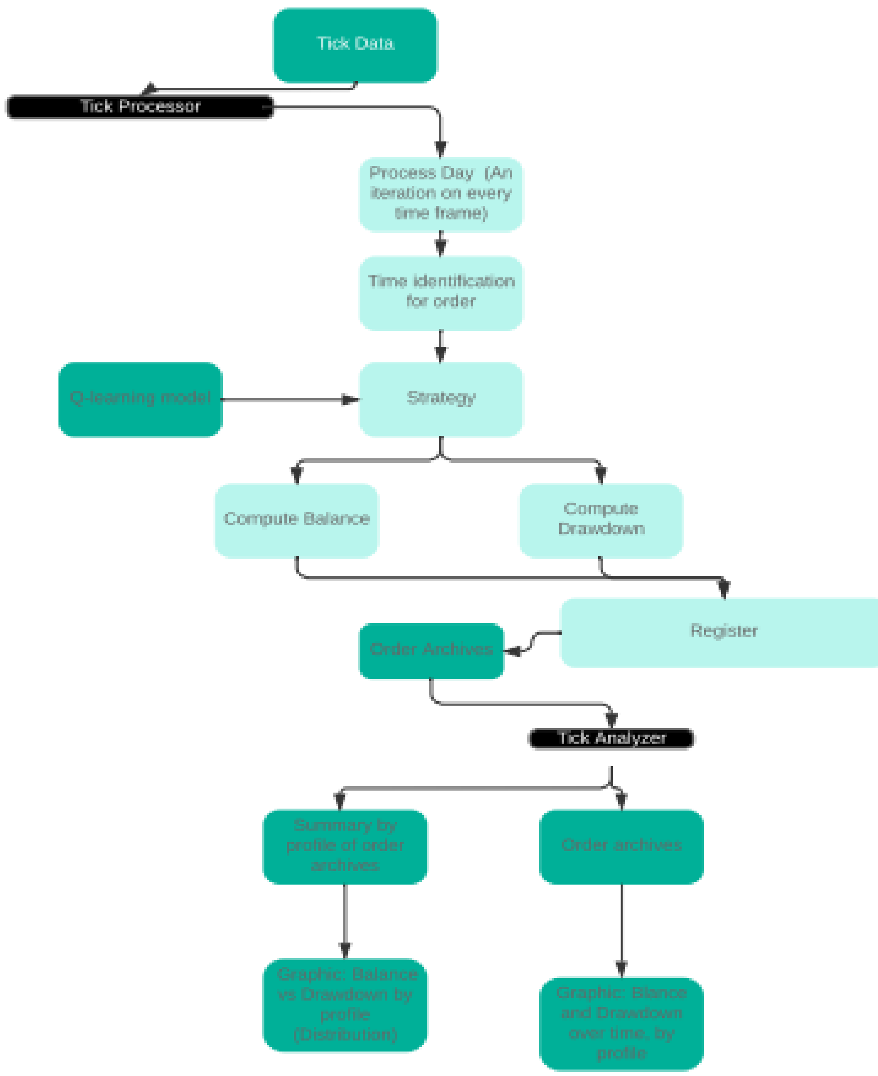
Short timeframes (purple/blue profiles) tend to be associated with negative results and higher drawdowns. This suggests that trading decisions based on short time intervals could be exposed to higher volatility and less reliable signals, resulting in greater cumulative losses.

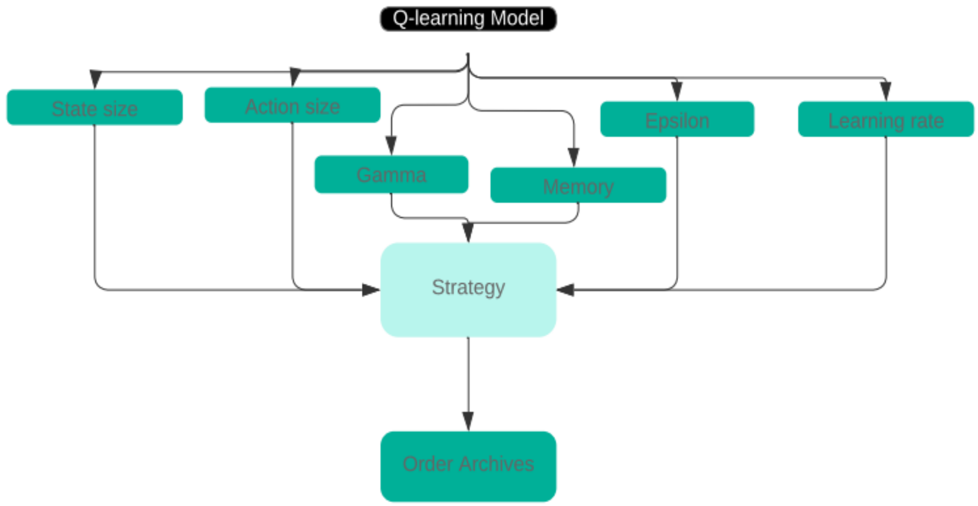
Long Timeframe Strategy:

Long timeframes (yellow/orange profiles) show a trend towards positive balances and more controlled drawdowns. This indicates that the algorithm's strategy is more effective in these intervals, likely due to better ability to identify and follow more stable market trends.

Q-learning implementation

The integration of Q-learning into the meticulously developed and reviewed backtesting process aims to enhance each aspect of machine analysis and ascertain the presence of any substantial disparities in outcomes with and without Q-learning. By incorporating Q-learning, a form of reinforcement learning, into the backtesting framework, the study endeavors to evaluate its impact on decision-making processes within the analyzed system. This deliberate inclusion serves as a methodological refinement, offering a systematic approach to discerning the efficacy of Q-learning in improving predictive accuracy and optimizing trading strategies. Through this integration, the study seeks to contribute to the ongoing discourse on the utilization of machine learning techniques in financial analysis, thereby enriching the understanding of their practical implications and potential benefits.





The Order Archives, produced in Tick Processor,are made in the following way:

- Column 0: **Execution ID**
- Column 1: **Threshold -** Value based on dictionary
- Column 2: **Decision -** Buy or Sell
- Column 3: **Volume of Buy or Sell -** 1
- Column 4: **Bid Price**
- Column 5: **Ask Price**
- Column 6: **Balance**
- Column 7: **Drawdown**

Reminding that profit factor :

$$Profitfactor : Balance/Drawdown$$

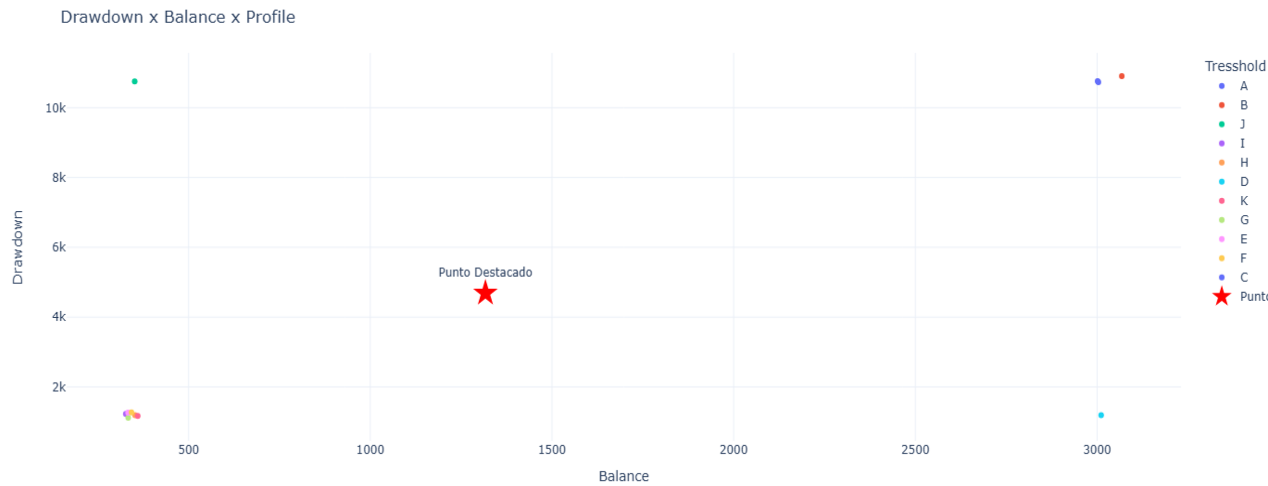
These are some aspects of The Backtesting modeule with the help of Q-learning model :

$$BestProfitfactor = 2.53$$

$$WorstProfitfactor = 0.03$$

with an estimated Profit factor, overall, of :

$$Estimated = 0.28092$$



General Distribution
Observation:

The profiles are distributed along the balance axis from approximately 0 to 10k and on the drawdown axis from 0 to over 10k.

Interpretation:

This dispersion indicates that the algorithm experiences a range of results in terms of positive balance and drawdown at all times, depending on the profile.

Centroid (Highlighted Point)
Observation:

The highlighted point, which represents the centroid of all profiles, is located near the zero balance axis and in a region of relatively low drawdown.

Interpretation:

The centroid indicates that, on average, the algorithm tends to operate with a balance close to zero and a moderate drawdown. This suggests that, considering all profiles, the algorithm's strategy tends to be neutral in terms of net gain/loss, with moderate risk exposure.

Conclusions

The majority of thresholds (profiles) tend to accumulate close to 0, for both drawdown and balance; This is striking and can be understood as a reliable management of risk and profit.

Conclusion

The integration of the Q-learning model has led to significant enhancements in the performance of the trading algorithm. Initially, the algorithm relied on static rules and traditional backtesting methods, which limited its ability to adapt to the dynamic nature of market fluctuations. The introduction of Q-learning, a reinforcement learning algorithm, has endowed the system with the capability to learn and adjust its trading strategies in real-time. This approach has enhanced the algorithm's responsiveness to emerging patterns in market data, thereby optimizing sequential decision-making in uncertain environments. Quantitative analysis of the strategy reveals an increase in the average profit factor and a reduction in relative drawdown, indicating an improved capacity to maximize risk-adjusted returns. Consequently, the integration of Q-learning has transformed the trading strategy into a more adaptable and efficient tool, enhancing its performance in volatile market conditions and improving its ability to consistently generate positive outcomes.

Thanks

This project wouldn‘t have been capable without the help of ’TRADING SOLUTIONS COMPANY S A S’ and specifically with the guide of Kevin Omar Dávila Castellar that provided a great guide for the costruction of the algoritm and gave the idea of the Strategy.