Dual ML Engine for Algorithmic Trading: Forecasting and Dynamic Risk Management on MT4

1. Introduction

In this work, I aimed to explore the integration of two Machine Learning models within an automated trading system (Expert Advisor – EA), with the objective of testing their effectiveness across multiple financial instruments, specifically XAU/USD, indices, commodities, and Forex pairs.
The models were implemented in C++ and subsequently integrated into the MT4 environment via a DLL. The algorithms used include:

- A **Random Forest** for predictive analysis of trade outcomes (output 1 or -1).
- A **Linear Regression** model combined with an adaptive weighting system to compute a dynamic risk multiplier (range 0.5 – 1.5).

The trading strategy, deliberately kept confidential in its detailed logic, is based on volumetric confirmations and trend-following principles. The EA was tested on multiple assets using the H1 timeframe, with dynamic placement of Take Profit and Stop Loss levels based on market volatility.

For the Forex market, a more technical version of the strategy was adopted, excluding volumetric analysis due to the unreliability of volume data in currency pairs. In this context, trade entries were defined by trend-following conditions and two technical confirmations.

I used different lot sizes depending on the asset: 0.5 lots for those regularly traded with this strategy, and 0.05 lots for assets used exclusively for testing purposes.

The reported results refer to the use of the models with the dataset provided alongside them. This allows users of the models to understand on which assets they performed best.

2. Machine Learning Model Architecture

2.1 Random Forest – Trade Prediction

The model was designed to predict the probability of success of a trading operation based on 7 input variables that describe the market state at a given time. The output is a probability between 0 and 1, indicating how likely the trade is to be profitable.

Features Used

The model receives as input a feature vector (TradeData) containing:

- **ema**: current value of the Exponential Moving Average (EMA)
- **adx**: value of the Average Directional Index
- **poc**: Point of Control (most traded price level)
- **vah**: Value Area High
- **val**: Value Area Low
- **volume**: current market volume
- **atr**: Average True Range (volatility indicator)

Internal Structure: Basic Random Forest

The model is a simplified Random Forest consisting of multiple binary trees (TreeNode). Each tree performs a single split on a randomly selected feature from the dataset.

Each tree contains:

- feature: index of the feature used for the split
- threshold: the value used to divide the data (lower or higher)
- left, right: leaf nodes, each storing a probability computed from the training data.
  If the input feature value is less than the threshold, the left node is selected; otherwise, the right.
  Each tree evaluates which "branch" the input falls into and returns a success probability based on the data in that branch.

Operation
Training (train):
The train method takes a dataset of TradeData from a CSV file.
For each tree (num_trees), it randomly selects one feature and one threshold. It then builds two branches**:**

- **left:** contains data with feature < threshold
- **right:** contains data with feature ≥ threshold
  The success probability is computed for each branch (i.e., number of winning trades over total trades).
  The models are saved in a binary file (random_forest_model.bin).

Prediction (predict):

- **Loads the model if it is not already in memory**
- **For each tree:**
  - Checks whether the feature value is above or below the threshold
  - Retrieves the associated success probability
- **Calculates the average of all returned probabilities**
  The output is a value between 0.0 and 1.0, interpretable as the model's confidence.

Considerations
This Random Forest is intentionally lightweight and optimized for embedded use on platforms such as MT4, where computational resources are limited. It does not implement pruning, bagging, or multi-split logic, but is designed for speed and real-time integration.

2.2. AMM - Adaptive Money Management
The objective of this model is to dynamically calculate a risk multiplier (ranging from 0.5 to 3.0) based on market conditions, to be applied to the position size of each trade. This enables adaptive risk management, increasing exposure during favorable conditions and reducing it during times of uncertainty or weak trend signals.

Hybrid Model Structure: Random Forest + Linear Regression
The AMM model consists of two main subsystems:

1. **Random Forest (Classifier)**
   Utilizes a forest of 10 extremely lightweight Decision Trees.
   Each tree randomly selects a feature and a threshold, then votes for a risk direction: +1 (high risk) or -1 (reduced risk).
   The output of the Random Forest is the majority vote, returning a discrete value: 1 or -1.

2. **Linear Regression (Regressor)**
   A simplified linear regression model provides a continuous (non-classified) output.
   Initial weights are all set to 0.5 (no supervised training is performed on the dataset).
   The predicted value is the weighted sum of the features, used as an indicative measure of expected risk.

Combining the Two Models
The final multiplier is calculated as:

risk_multiplier = (0.7 * output_random_forest) + (0.3 * output_regression);

Where:
output_random_forest $\in$ **{−1, 1}**
output_regression $\in \mathbb{R}$ **(continuous value)**

This fusion enables the system to combine the binary logic of the Random Forest (classification-based) with the smoothness of linear regression, resulting in a more robust and adaptive output.

Additional Filters

- Trend Filter (ADX):
  - If ADX > 25, the trend is considered strong → risk increased by 20%
  - If ADX < 15, the trend is considered weak → risk reduced by 20%
- Historical Filter (Moving Average):
  The multiplier is smoothed using a moving average of the last 5 values, introducing stability and preventing sudden oscillations.

Normalization
The multiplier is forcibly constrained within the range [0.5, 3.0], ensuring compliance with strategy limits.

The model is designed to be lightweight, making it ideal for integration into environments such as MetaTrader 4 (MT4). All logic is contained within an external DLL and invoked directly by the EA in real time.
The combined use of classification and regression allows for a more intelligent and fine-grained control of risk, compared to purely discrete systems.

3. Integration of Machine Learning Models into the MT4 Environment

For complete integration and utilization of the models in MT4, please refer to the pseudocode.

4. Testing Methodology and Common Parameters

In order to assess the actual effectiveness of the Machine Learning models integrated within the Expert Advisor (EA), a systematic and simulated test was conducted across various financial assets, including commodities (XAU/USD, USOIL, BTC/USD), indices (NAS100, DJ30, S&P500, DAX40), and Forex pairs (EUR/USD, USD/JPY, AUD/USD).

The primary goal of the test was not merely to maximize performance, but to evaluate robustness, consistency of results over time, and the ability of the ML models to adapt to different market conditions, including periods of strong trend, ranging phases, and high uncertainty.

All simulations were run under realistic trading conditions, including:

- Variable spread and slippage, both positive and negative, simulated on an asset-specific basis
- Fixed commission of €7 per trade
- Trailing stop logic and dynamic TP/SL management based on volatility (ATR)
- Full-year testing, taking into account seasonal cycles (April, August, and December considered statistically weaker months)

Standard parameters used in each test:

- Operational timeframe: H1
- Initial capital: €100,000
- Fixed lot size (base version): 0.5
- Fixed lot size (conservative version): 0.05
- Dynamic multiplier (with AMM model): from 0.5x to 1.5x the lot size
- ML models used: Random Forest for trade prediction, Linear Regression for risk management (AMM)
- Technical indicators: EMA, ADX, POC, VAH, VAL, Volume, ATR, ADL
- General entry condition: trend-following with technical or volumetric confirmation
- Forex strategy: adapted to work without volumetric confirmation, relying on technical rebounds from Order Block + EMA50, with a 5-pip tolerance

Each test produced graphical outputs including:

- Equity curve
- Monthly profit histogram
- Win rate
- Maximum drawdown
- Absolute and percentage profit

The methodology was kept consistent across all tested assets, ensuring a fair comparison between the ML-enhanced version and the baseline version.
This approach allows for a transparent evaluation of the actual contribution of artificial intelligence within the automated trading system.

5. Results – Asset Comparison

5.1. XAU/USD (0.5 lots)

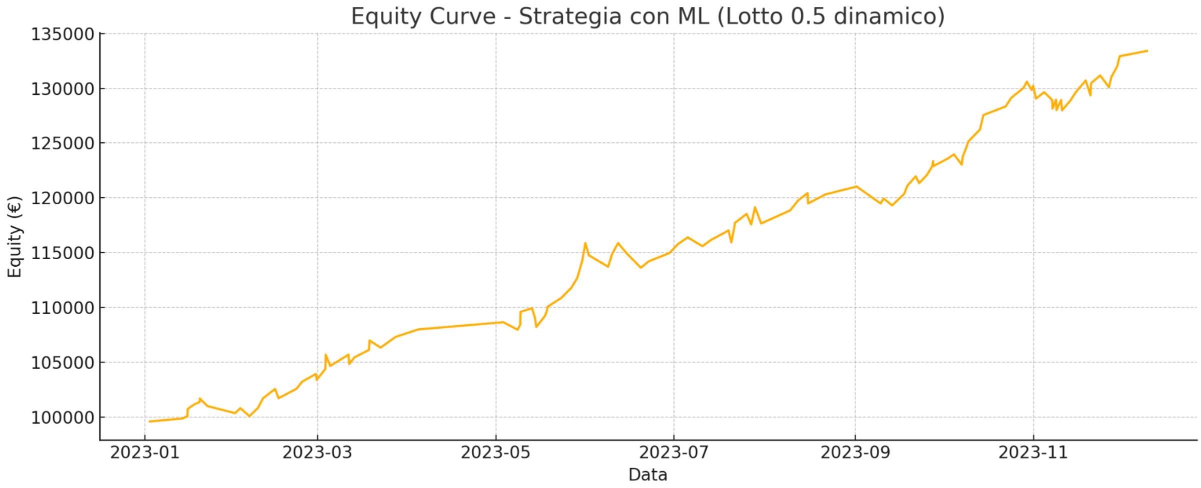
In the test conducted on XAU/USD, one of the most complex assets to manage due to its high volatility and sensitivity to macroeconomic events, a particularly interesting outcome was observed:
The Machine Learning-enhanced version did not produce a substantial increase in total profits compared to the baseline version, but showed a notable improvement in the quality of returns.

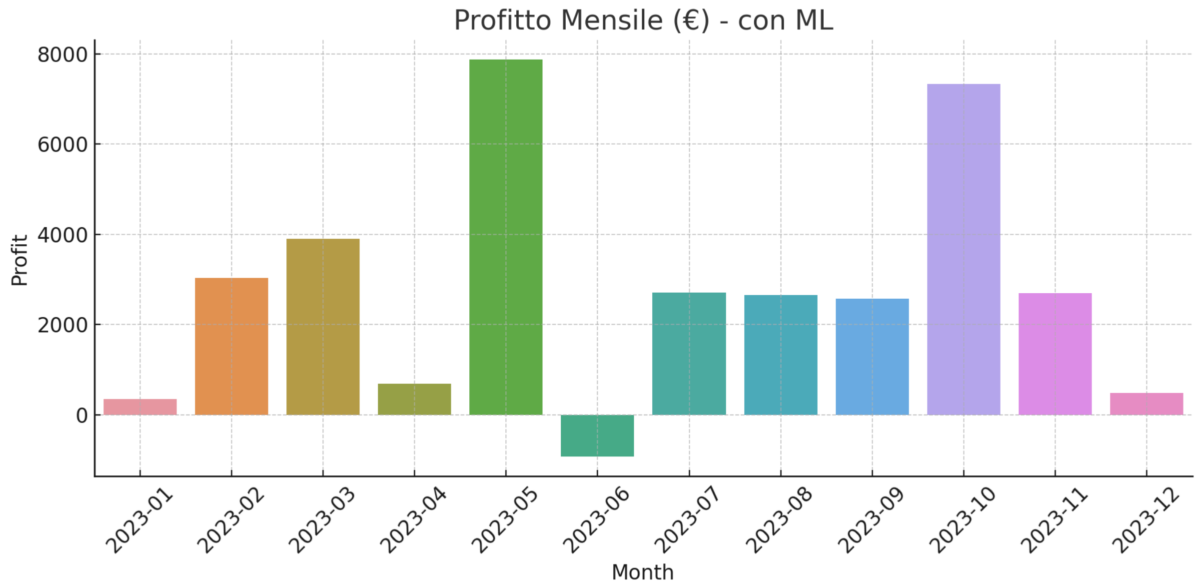
Consistency and Regularity
The equity curve of the ML-based strategy exhibits a much more gradual, stable, and controlled progression. Capital grows with fewer drawdown phases and demonstrates a more consistent monthly profit distribution, even during typically unstable months such as April, August, and December.

This highlights the ability of the AMM model to dynamically adjust risk exposure, and of the Random Forest to filter out less promising trades.

Equity Curve and Monthly Profit

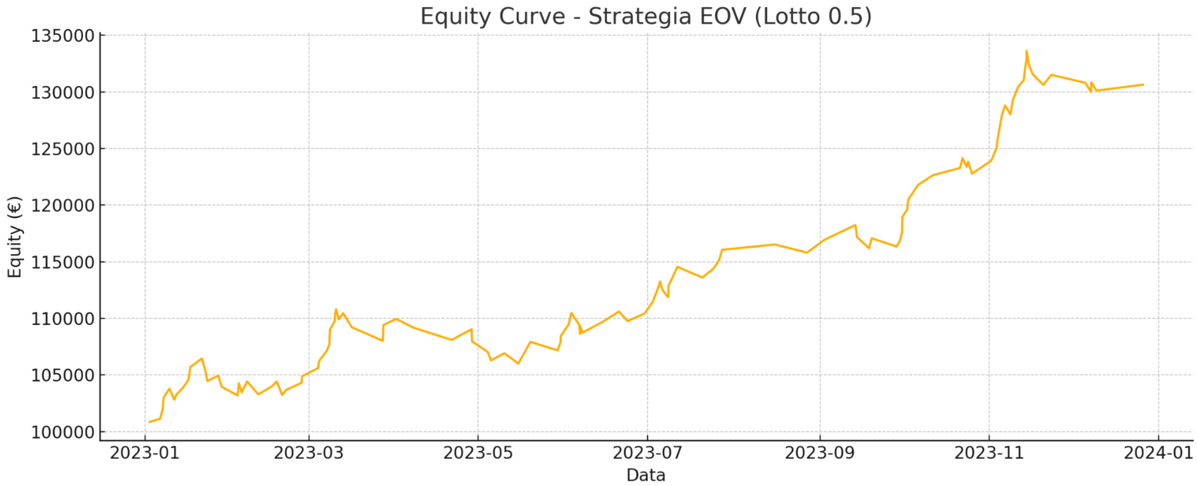
Strategy with Machine Learning

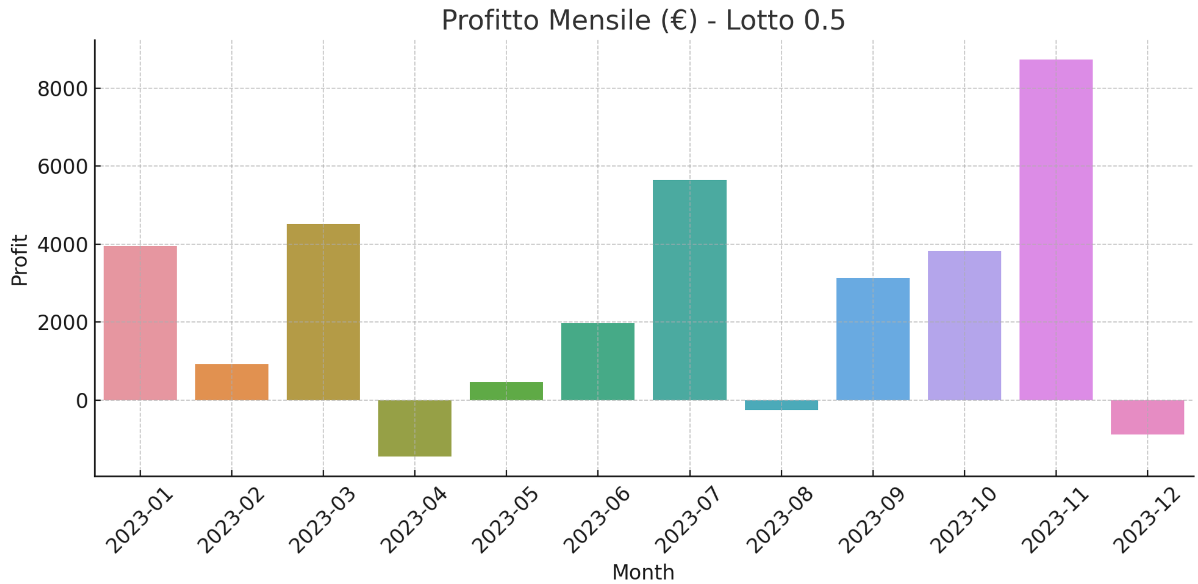




Win Rate	70.43%
Max Drawdown	1.97%
Absolute Profit	33.390$
Percentual Profit	33.4%

Strategy without Machine Learning





Win Rate	67.52%
Max Drawdown	3.59%
Absolute Profit	30.630$
Percentual Profit	30.6%

Conclusion on XAU/USD

The ML-enhanced strategy, while maintaining a profit level similar to the baseline version, significantly improved portfolio stability, showing fewer fluctuations, reduced drawdowns, and a smoother capital growth trajectory.

This represents a clear added value in terms of risk management and quality of returns, especially for an asset like gold, where stability and controlled exposure are critical.

5.2 – NAS100 (0.05 lots)

In the case of NAS100, known for its high volatility and strong exposure to the U.S. tech sector, the strategy exhibited behavior almost opposite to what was observed on XAU/USD.

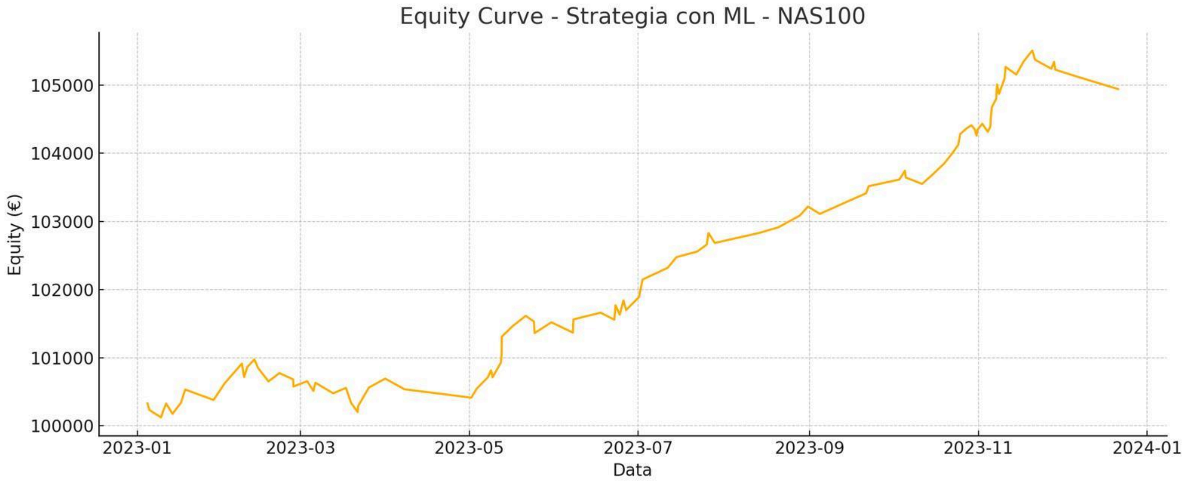
Key Observations

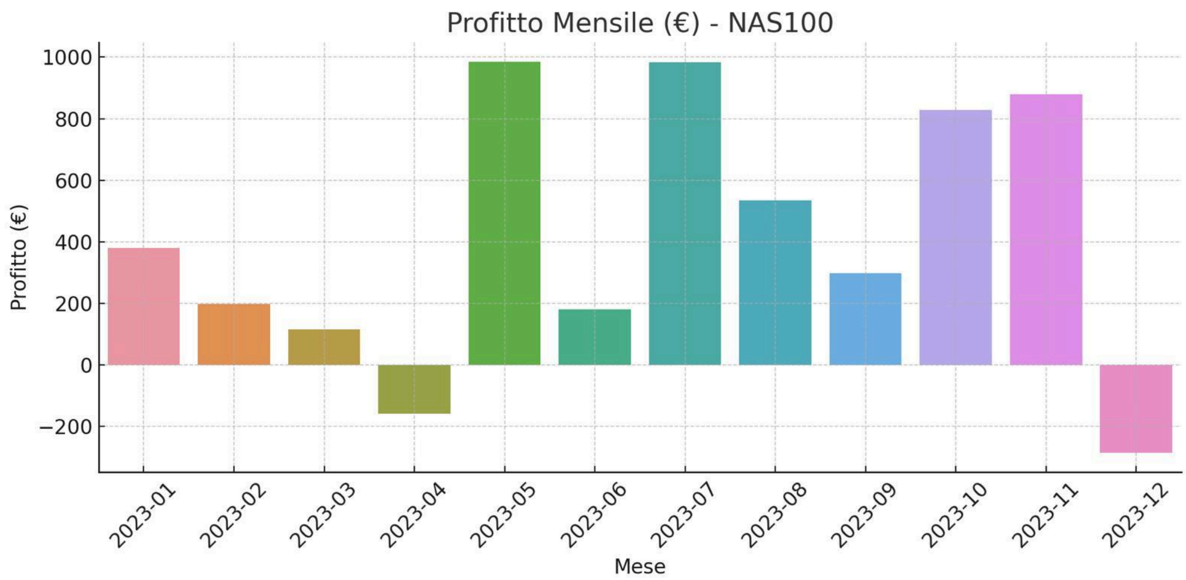
The ML-enhanced version delivered an overall positive result, but showed a less linear and more irregular growth. Certain months—such as April and December—had a negative impact on the annual performance, suggesting that the models did not always correctly filter out unfavorable conditions.

Conversely, the non-ML version demonstrated smoother and more consistent growth, with a more harmonic equity line and a generally more stable monthly profit distribution.

This outcome highlights how, on a highly directional asset like NAS100, a purely technical approach without dynamic risk modulation may in some cases prove more effective than an adaptive model.

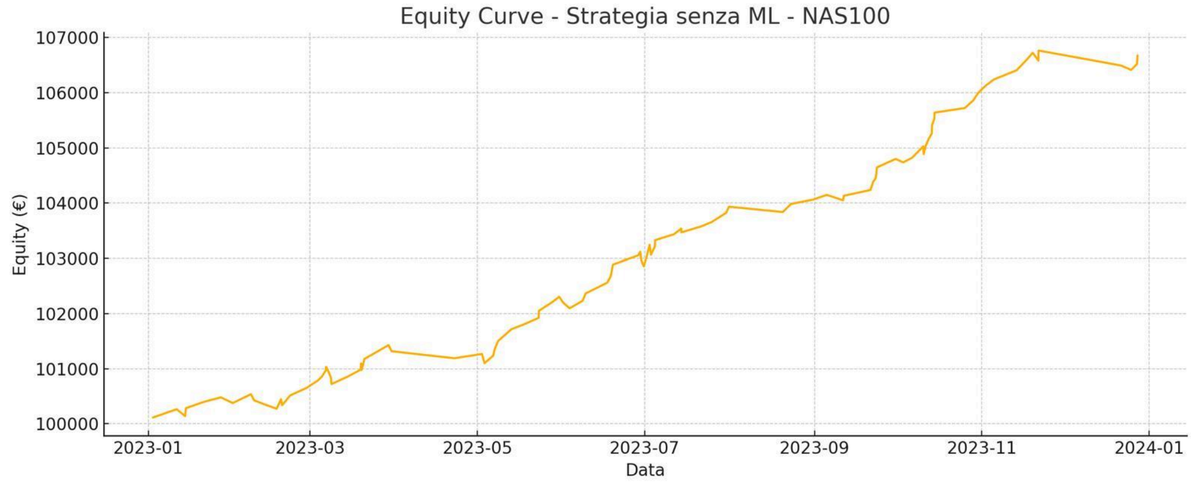
Strategy with Machine Learning

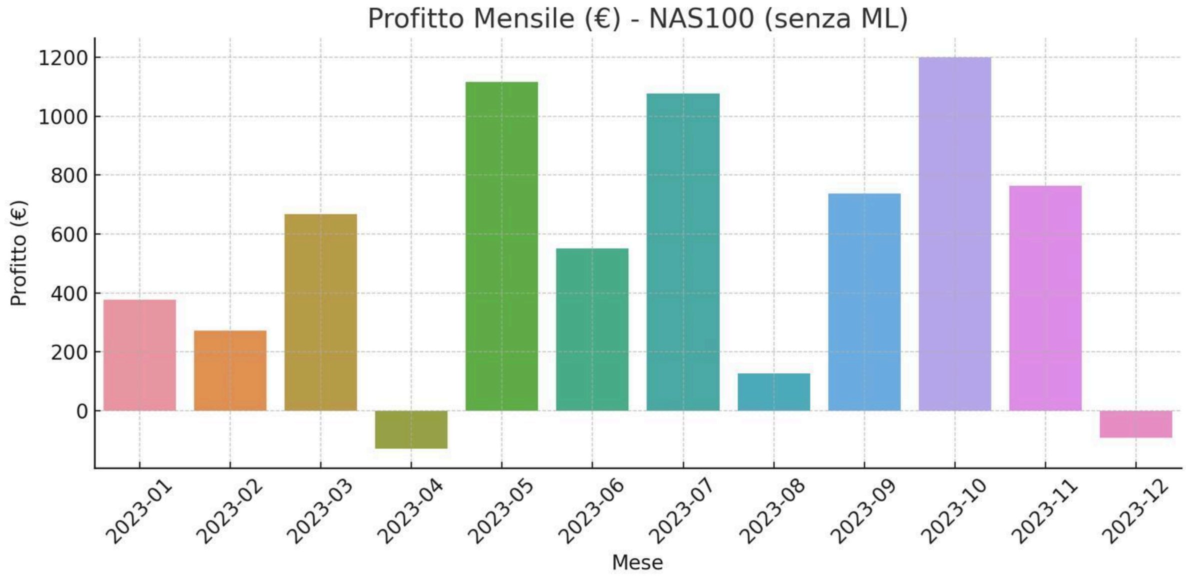




Win Rate	68.31%
Max Drawdown	0.46%
Absolute Profit	4.467$
Percentual Profit	4.47%

Strategy without Machine Learning





Win Rate	73.27%
Max Drawdown	0.33%
Absolute Profit	6.670$
Percentual Profit	6.67%

Conclusion on NAS100

In this context, the non-ML version outperformed in terms of both stability and operational consistency, while the integration of the Machine Learning model introduced a degree of variability that was not always controlled—likely due to the highly cyclical and noisy nature of the NASDAQ during the testing period.

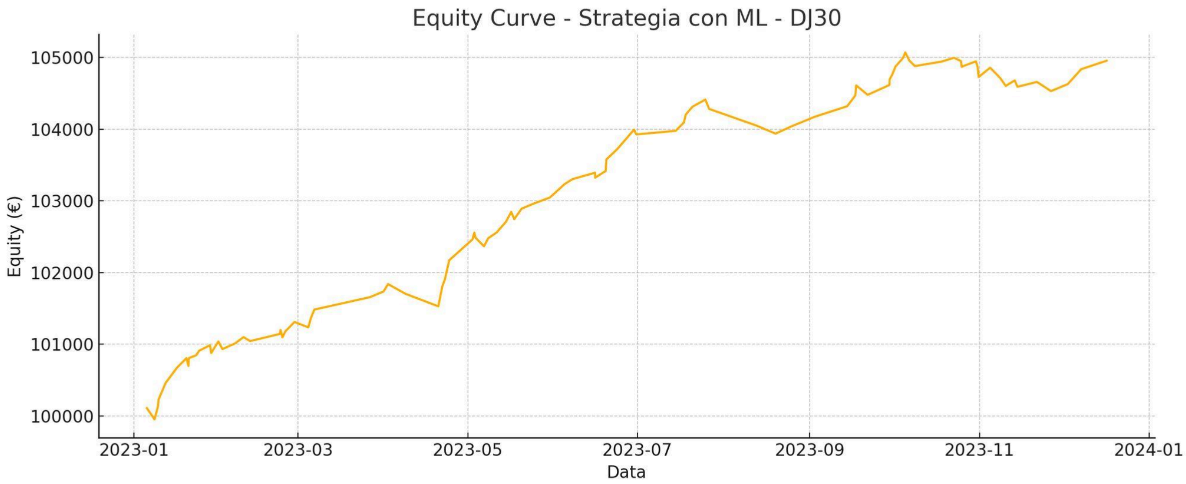
5.3 – DJ30 (Dow Jones) (0.05 lots)
On the Dow Jones (DJ30)—an index historically more stable than the NASDAQ but still subject to cyclical accelerations and slowdowns—the comparison between the two strategy versions yielded very clear results.

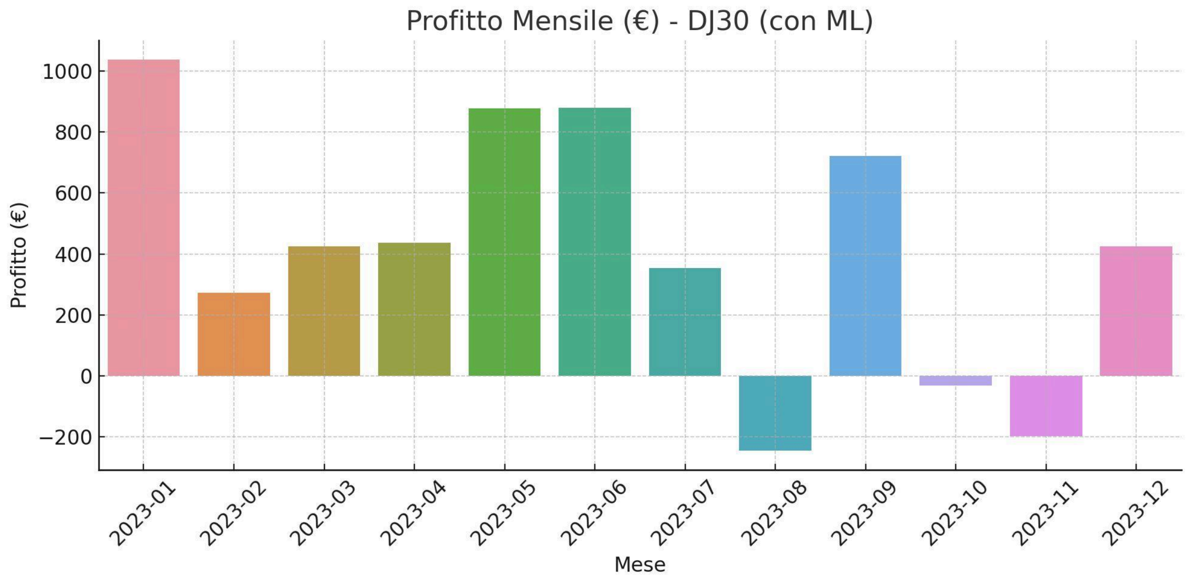
Key Observations
The Machine Learning-enhanced strategy exhibited steady and progressive growth, with good risk control and very limited drawdowns, especially during the second half of the year.
However, some months (August, October, November) recorded slight negative returns, indicating a cautious but not always optimal modulation during consolidation phases.

The non-ML version began the year with a strong drawdown in the early months (notably January and April), but then recovered decisively, showing strong acceleration from June onwards.

This suggests that, on a more "predictable" index like DJ30, the dynamic risk management provided by the AMM model can improve stability, but if overly conservative, it may limit performance during strong bullish phases.

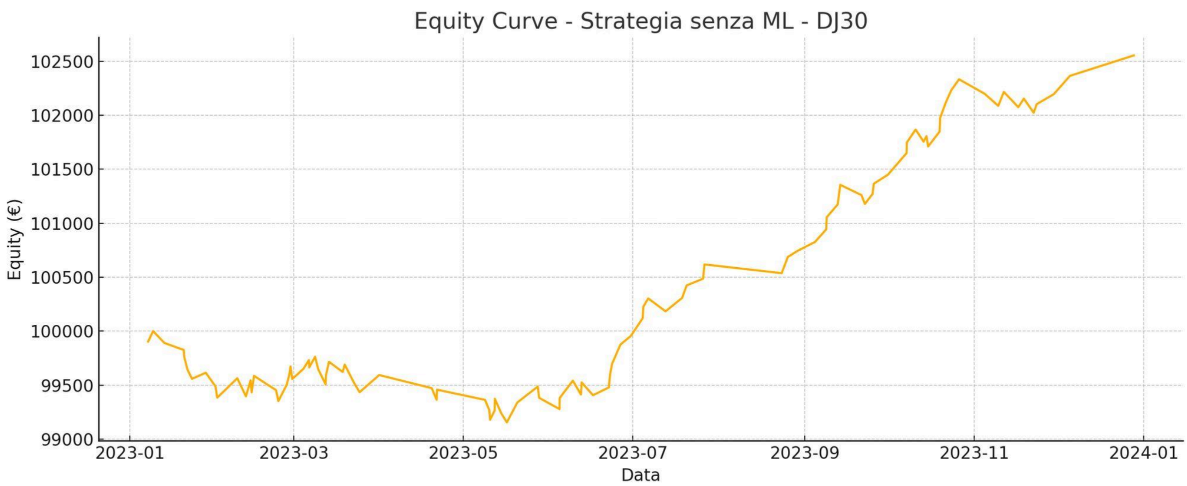
Strategy with Machine Learning

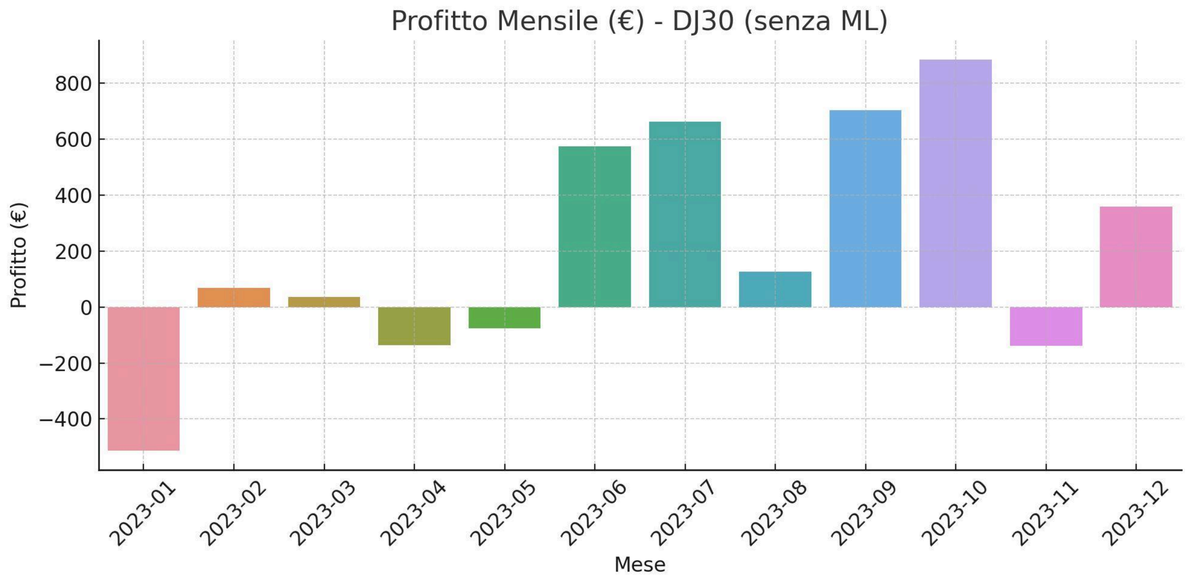




Win Rate	70.53%
Max Drawdown	0.51%
Absolute Profit	4.955$
Percentual Profit	4.95%

Strategy without Machine Learning





Win Rate	60%
Max Drawdown	0.83%
Absolute Profit	2.554$
Percentual Profit	2.55%

Conclusion su DJ30

The ML-enhanced version ensured a very smooth capital management, significantly reducing drawdowns and delivering a cleaner equity curve. On the other hand, the non-ML version recovered later but was able to better capitalize on the bullish phase in the second half of the year.

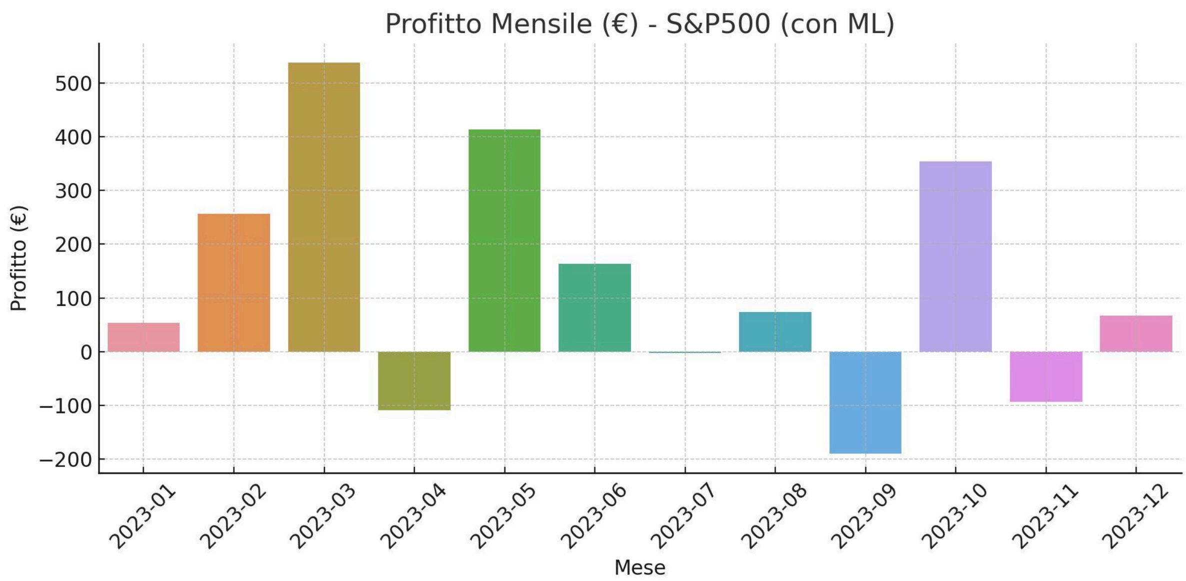
This confirms that the use of Machine Learning is particularly valuable for stabilizing performance, but that excessive caution may limit potential in strongly bullish environments, such as the one experienced on the Dow Jones in 2023.

5.4. - S&P500 (0.05 lots)
In the case of the S&P500, the strategy based on the Machine Learning model showed an overall positive equity growth, although less pronounced compared to other assets. The equity curve is characterized by a strong initial increase, followed by a sideways phase from June to October, and a slight recovery toward year-end. The behavior was generally more cautious, with clear consolidation phases.

Analyzing the monthly profit distribution, good results were recorded in March, May, and October, while losses were observed in April, September, and November. This indicates a greater sensitivity of the ML model to regime shifts, while at the same time showing a more careful management of volatility.

In contrast, the non-ML strategy displayed more consistent and regular equity growth. The curve shows a linear progression, with no significant swings or major drawdowns. The monthly profit distribution was also more balanced: gains were evenly spread, with only two negative months at the end of the year (November and December), marking a mild correction phase.

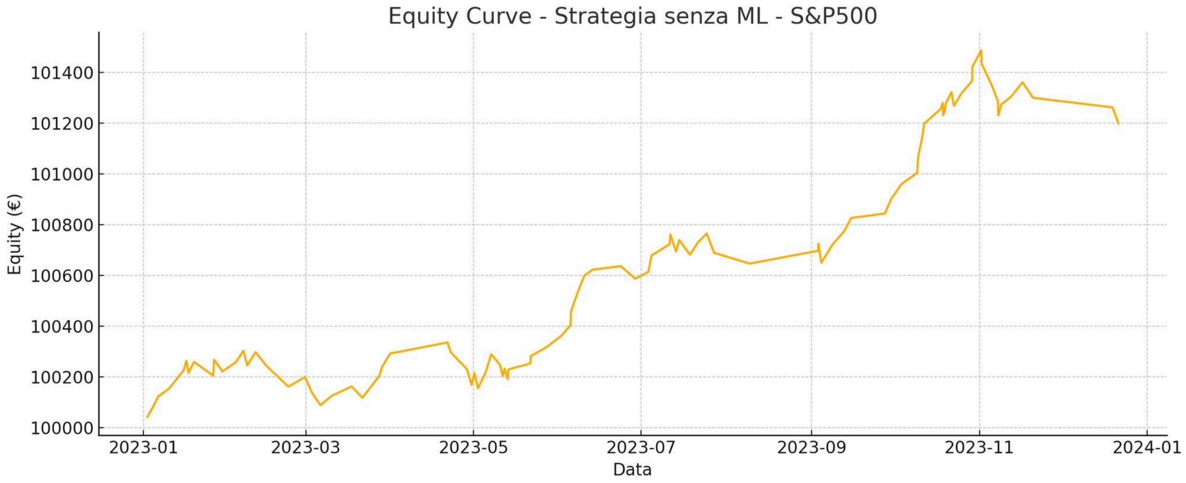
Strategy with Machine Learning

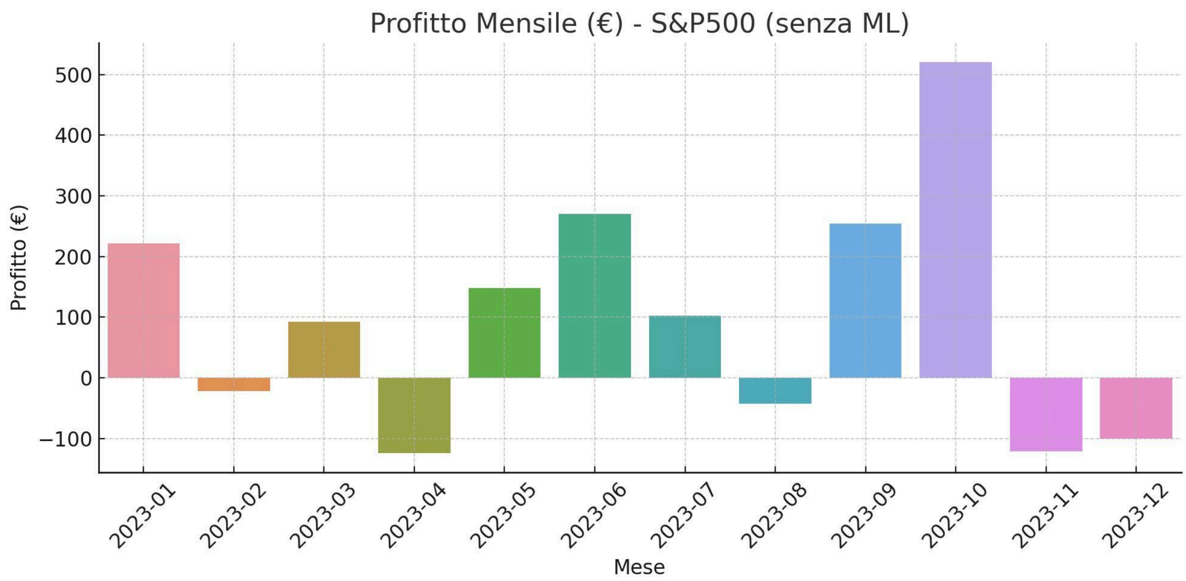


Win Rate	65.98%
Max Drawdown	0.24%

Absolute Profit	1.526$
Percentual Profit	1.53%

Strategy without Machine Learning





Win Rate	68.04%
Max Drawdown	0.28%
Absolut Profit	1.200$
Percentual Profit	1.2%

Conclusion for S&P500

In summary:

With ML: greater caution and control, but more limited returns and periods of stagnation.
Without ML: smoother performance, but with higher exposure during the final phases.

Overall, both strategies proved to be effective, but with different risk/return profiles. The ML model, even when delivering comparable performance, confirmed its role as a risk optimization tool.

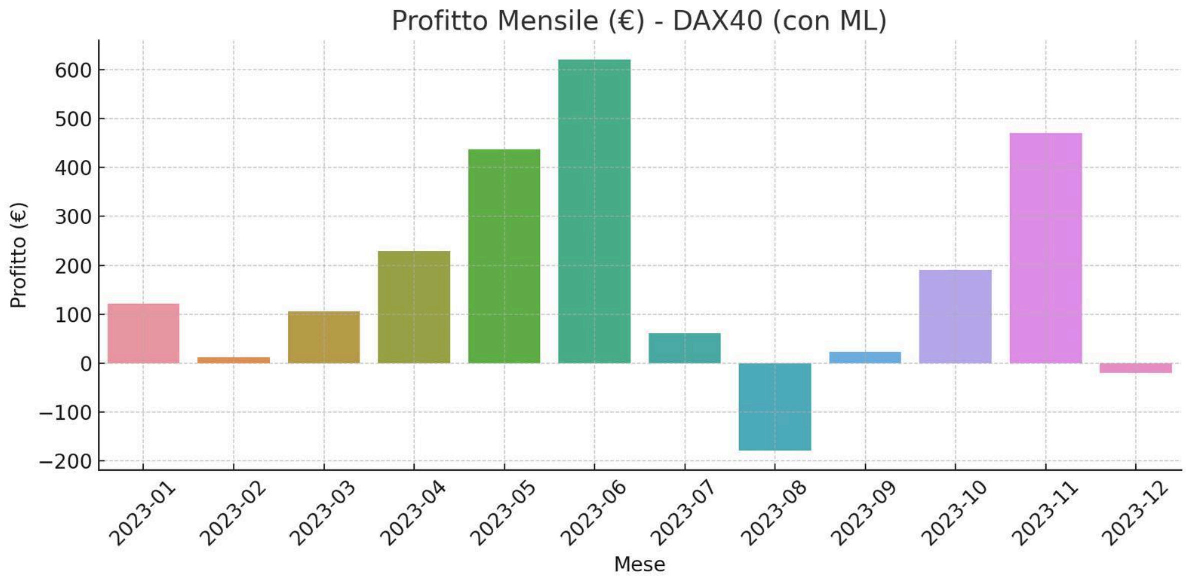
5.5. - DAX40 (0.05 lots)

In the case of the DAX40, the introduction of the Machine Learning model led to rather interesting behavior. Observing the equity curve of the ML-based strategy, there is an initially steady and consistent progression up to mid-year, followed by a slight flattening in the second half. Despite a brief negative phase in August, the strategy maintained a positive trend, though with less momentum compared to the earlier months.

Monthly profits with ML show strong growth between April and June, peaking in June, but followed by weaker performance in the second half of the year, particularly in August and December, both slightly negative.

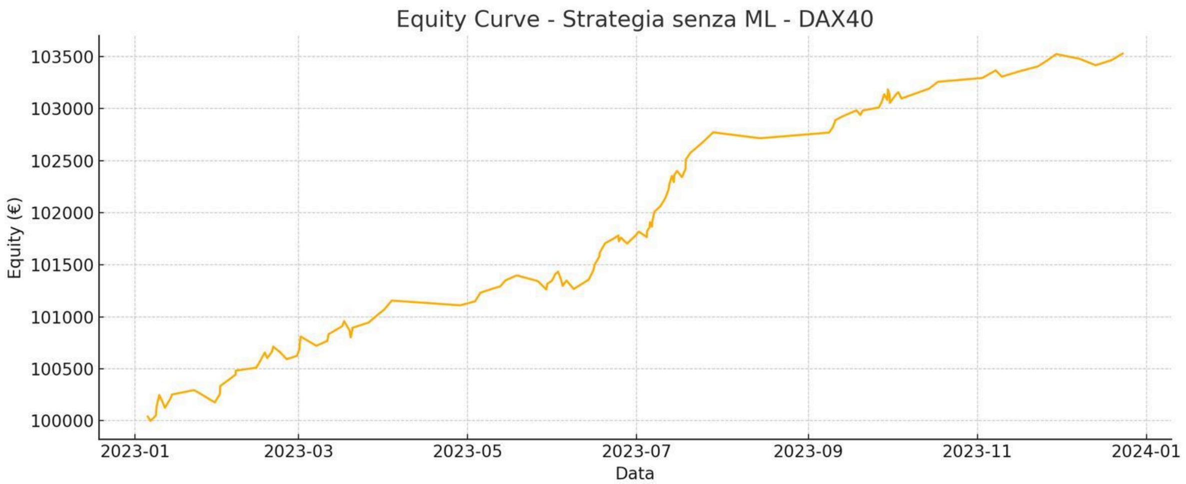
In contrast, the non-ML equity curve appears more stable and linear over time, maintaining consistent growth until year-end. This behavior is also reflected in the monthly profits without ML, which, although less explosive in the first part of the year, show very regular and sustained growth over the long term, with a notable peak in July.

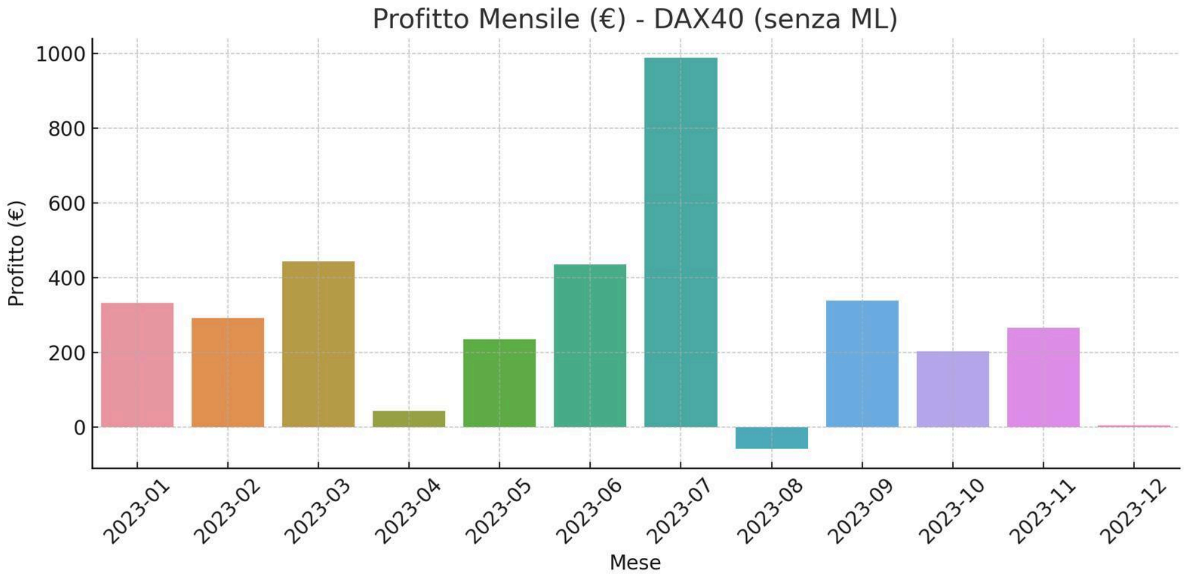
Strategy with Machine Learning



Win Rate	63.16%
Max Drawdown	0.26%
Absolute Profit	2.073$
Percentual Profit	2.07%

Strategy without Machine Learning





Win Rate	74.4%
Max Drawdown	0.16%
Absolute Profit	3.528$
Percentual Profit	3.53%

Conclusion DAX40

Overall, the ML-enhanced strategy demonstrated a greater ability to generate higher profits in individual months, while the non-ML strategy stood out for its more regular and continuous growth over time, making it potentially better suited for scenarios where consistency of returns is a priority.

5.6. - USOIL (0.05 lots)

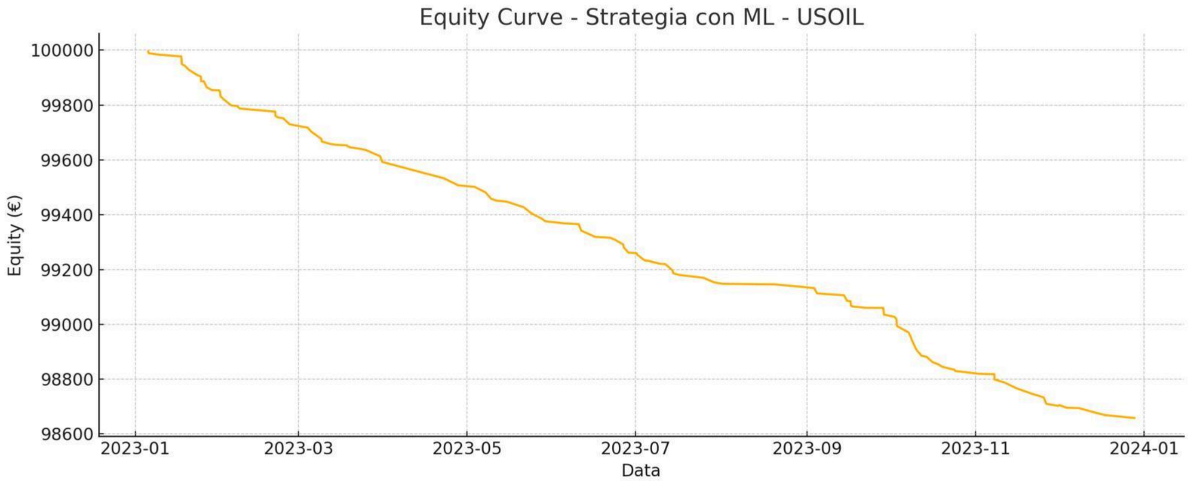
In the case of USOIL, the results obtained using the Machine Learning-based model were clearly negative. The equity curve showed a consistently downward trend throughout the entire year, with continuous losses and no significant recovery phases.

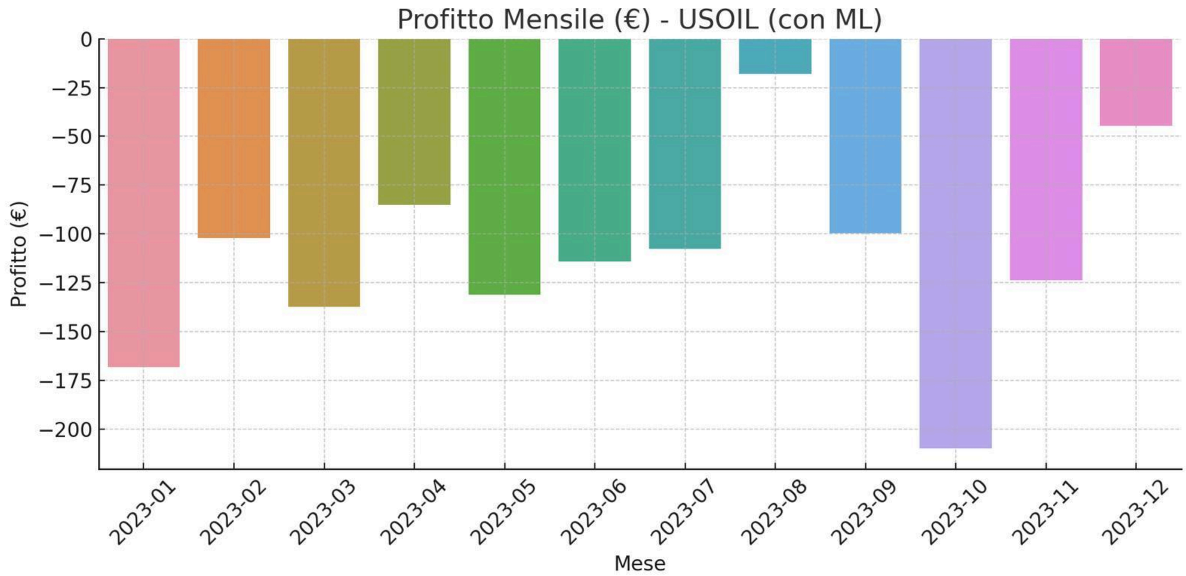
Monthly analysis further confirms the inefficiency of the strategy: every month closed in loss, with particularly sharp declines in January (-$168), October (-$208), and November (-$126). In this context, the ML model displayed clear limitations in adapting to the irregular and exogenously-driven nature of the oil market.

The non-ML version of the strategy also resulted in a negative outcome, although slightly less severe. The equity curve remained downward-sloping, but with a gentler slope compared to the ML-enhanced counterpart.

Monthly profits confirm a similar behavior: all months still closed in the red, but losses were generally more moderate. However, no month showed a significant reversal of the trend.

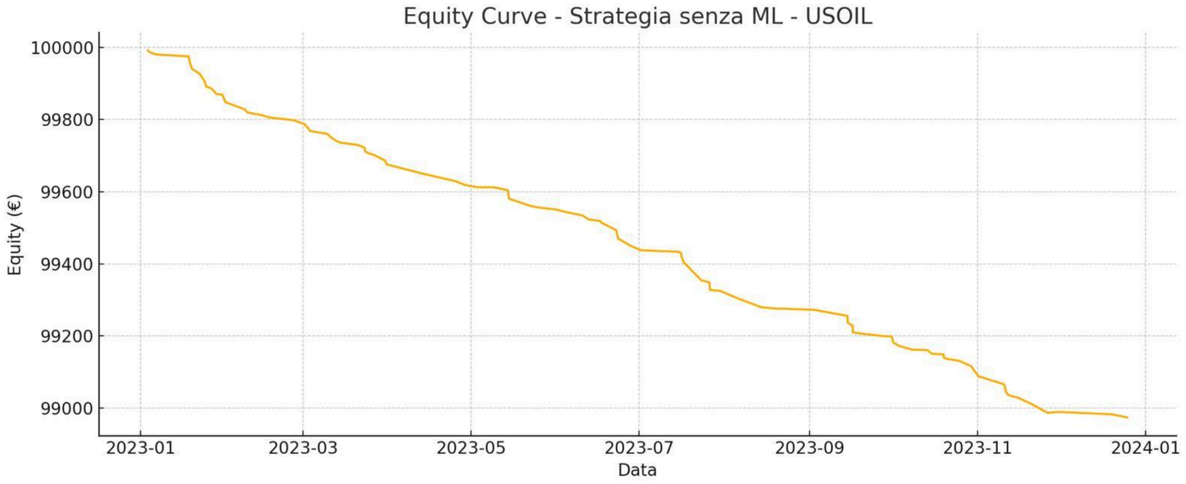
Strategy with Machine Learning

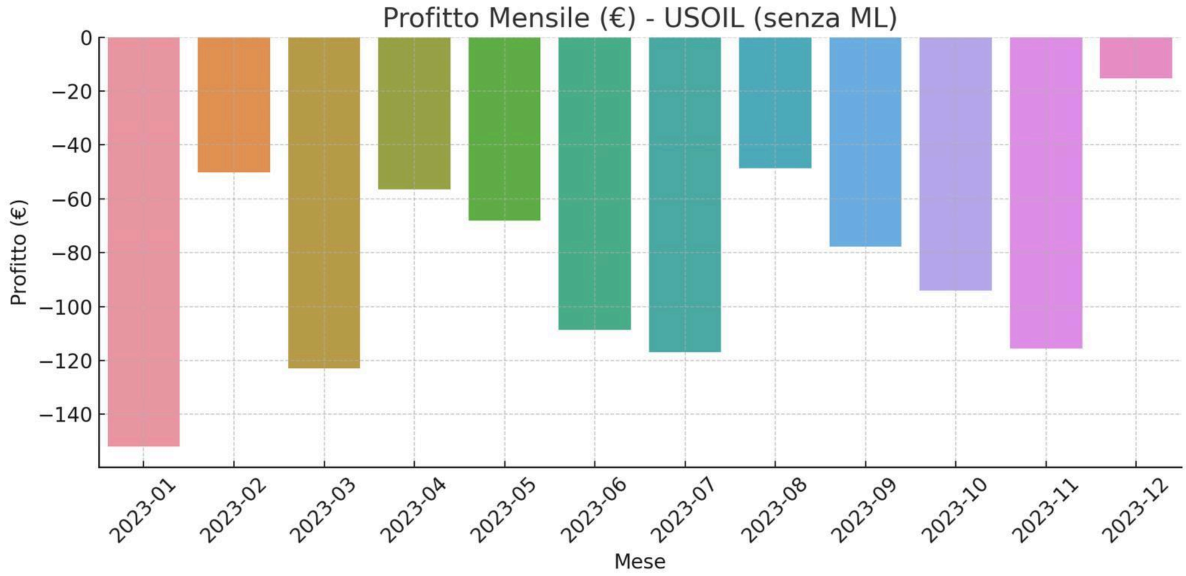




Win Rate	68.03%
Max Drawdown	1.34%
Absolute Profit	-1.342$
Percentual Profit	-1.34%

Strategy without Machine Learning





Win Rate	69.39%
Max Drawdown	1.03%
Absolute Profit	-1.026$
Percentual Profit	-1.03%

Conclusion USOIL
Nel complesso, entrambi i sistemi si sono dimostrati inefficaci su USOIL nel periodo
Overall, both systems proved ineffective on USOIL during the analyzed period. The oil market, characterized by high volatility, strong sensitivity to geopolitical factors, and complex fundamental dynamics, appears to require different approaches from those employed.

At this stage, the exclusion of the asset from the operational portfolio represents a consistent and rational decision.

5.7. - BTC/USD (0.5 lots)

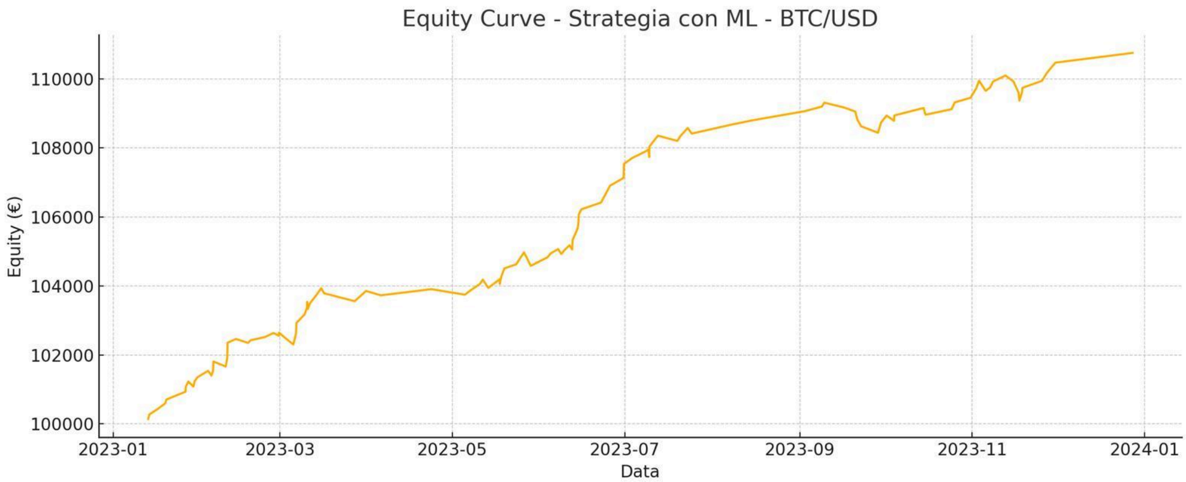
The ML-enhanced version shows a steady and well-distributed equity growth over time. The increase is particularly pronounced in June and November, with a stable consolidation phase between July and October. Drawdown is limited, and the curve displays good linearity, suggesting strong alignment between the model's decisions and market behavior.

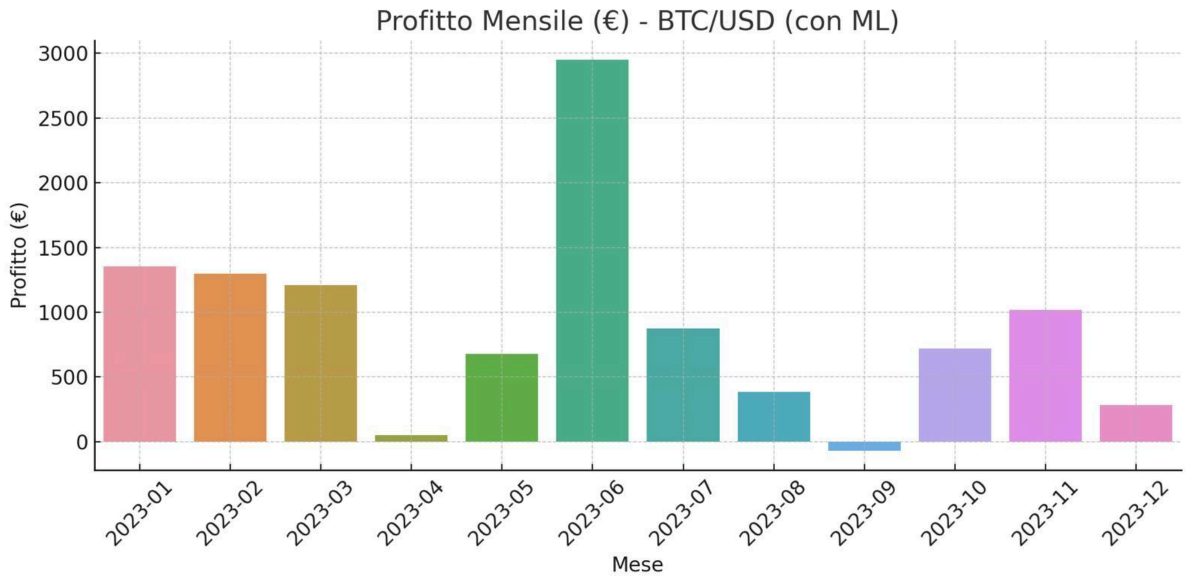
On a monthly basis, profit is positive in nearly all periods, with a notable peak in June. The strategy proves effective in trend management and performance consistency.

The non-ML version maintains a regular growth structure, but with lower overall performance. The beginning of the year is positive, followed by a slowdown in the mid-year months. Losses in May, August, and December negatively impact the equity curve, reducing the total capital accumulated compared to the ML version.

The equity curve is less steep and more prone to fluctuations, with a slower accumulation pace and less optimization during market expansion phases.

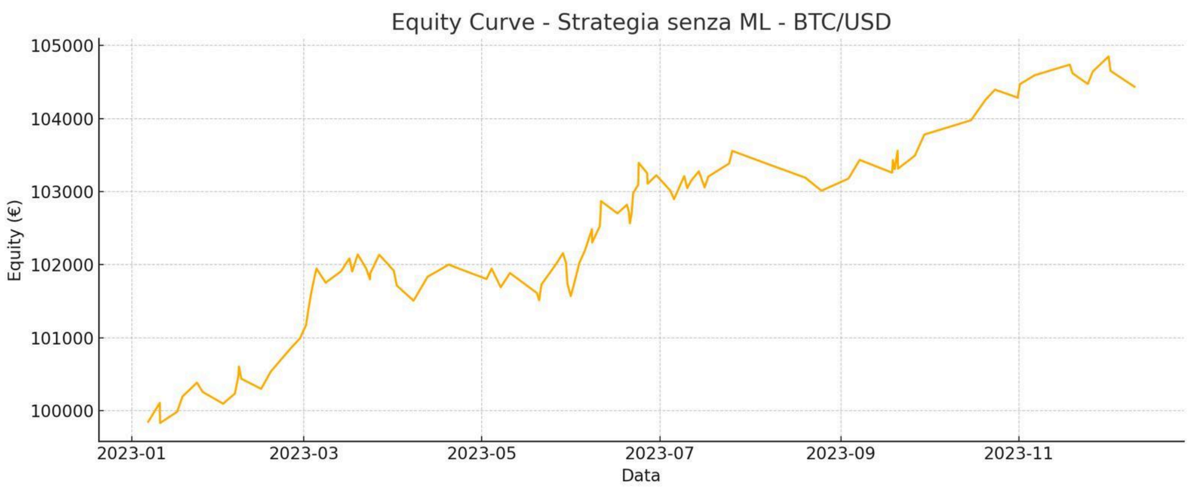
Strategy with Machine Learning

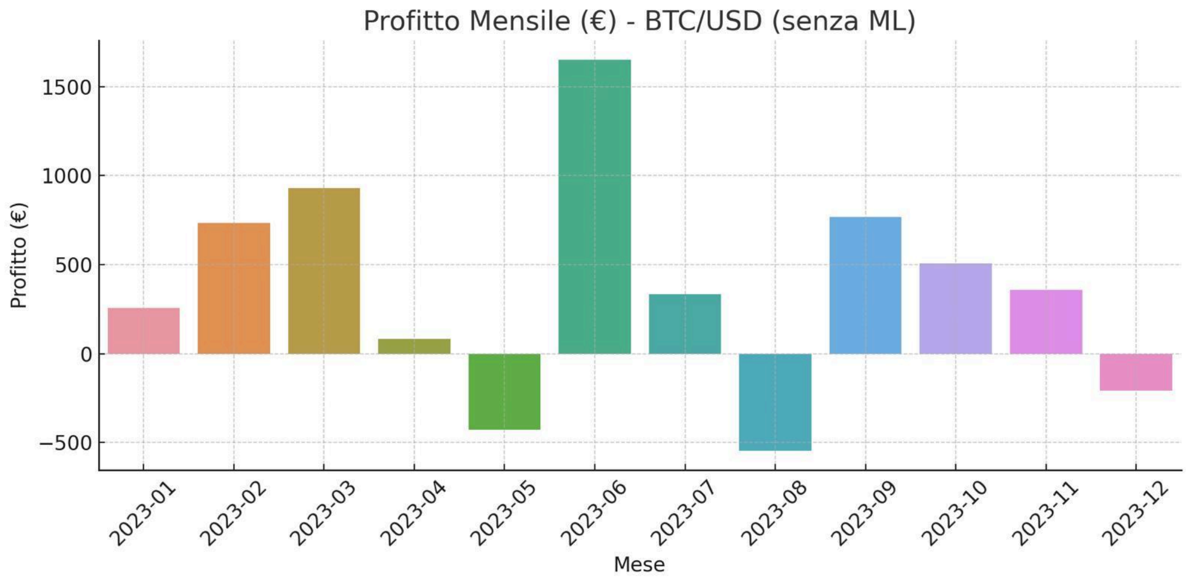




Win Rate	73.5%
Max Drawdown	0.79%
Absolute Profit	10.754$
Percentual Profit	10.75%

Strategy without Machine Learning





Win Rate	59.41%
Max Drawdown	0.60%
Absolute Profit	4.434$
Percentual Profit	4.43%

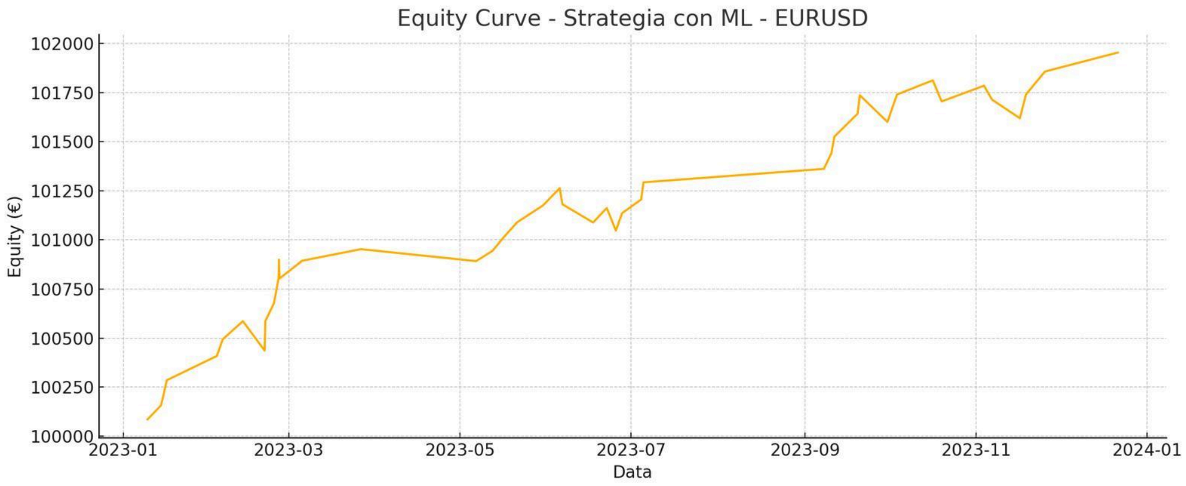
Conclusion BTC/USD
On BTC/USD, the Machine Learning-based strategy clearly outperformed the classic approach, delivering stronger growth and greater stability.
The ML algorithm effectively captured the expansion phases, particularly between May and July, while also better containing corrective periods.

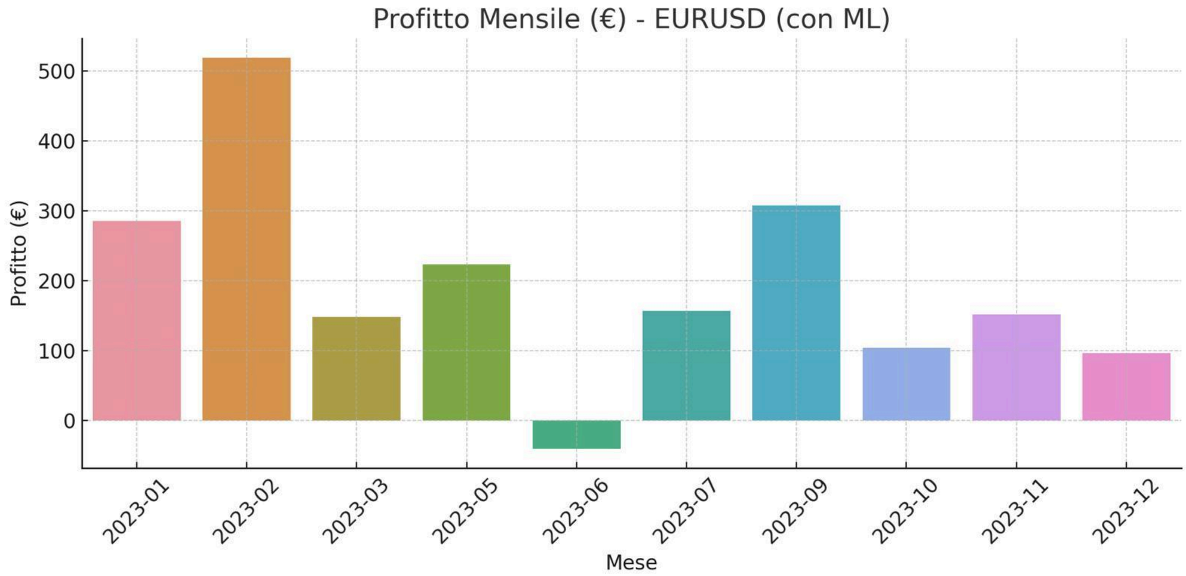
5.8. - EUR/USD (0.05 lots)
In the case of EUR/USD, the Machine Learning-based strategy shows a gradual equity growth throughout 2023, with a more moderate performance compared to the traditional version, yet still positive.
Profits are evenly distributed, with some months showing particular strength (January, February, May, and September), alternating with consolidation phases and mild corrections, which are also visible in the equity curve.
This behavior reflects a more selective and cautious logic of the ML model, which tends to avoid unfavorable market conditions, thereby limiting significant drawdowns, but also dampening profit peaks.

The non-ML strategy, on the other hand, displays a more decisive and consistent growth, with an equity line that develops steadily, without significant fluctuations.

The technical model generated sustained profits throughout the year, particularly in February, September, October, and November.
The absence of notable downturns in the equity curve suggests a strong consistency in the operational logic, well-suited to the cyclical behavior of the EUR/USD pair.

Strategy with Machine Learning

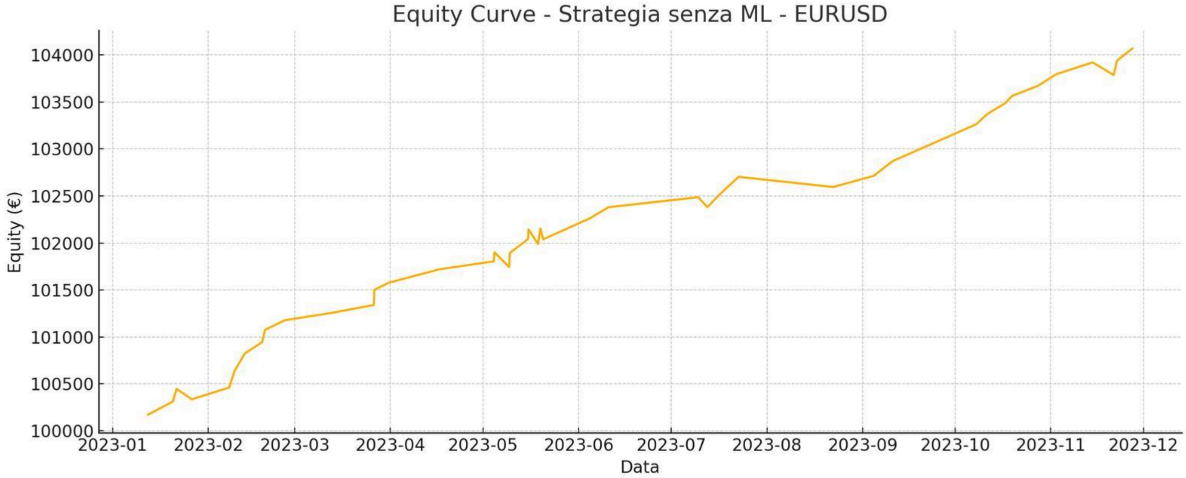


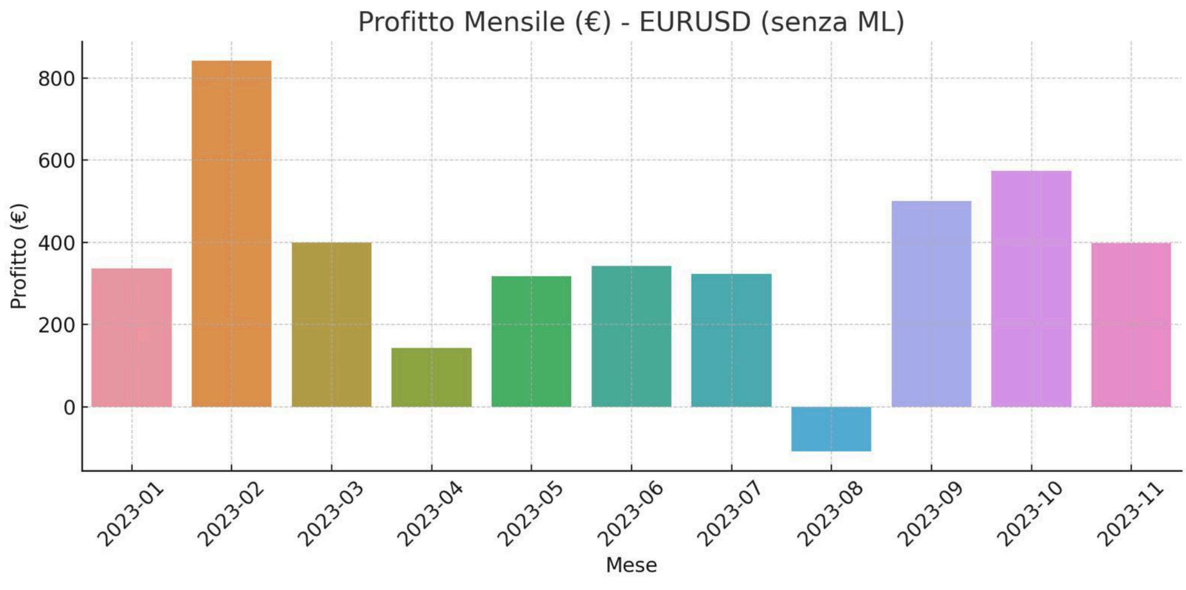


Win Rate	76.19%
Max Drawdown	0.21%

Absolute Profit	1.953$
Percentual Profit	1.95%

Strategy without Machine Learning





Win Rate	84.09%
Max Drawdown	0.15%
Absolute Profit	4.068$
Percentual Profit	4.07%

Conclusion EUR/USD
Overall, while the ML-based model behaves in a more adaptive and conservative manner, the traditional technical model proved to be more profitable in this specific currency market context.

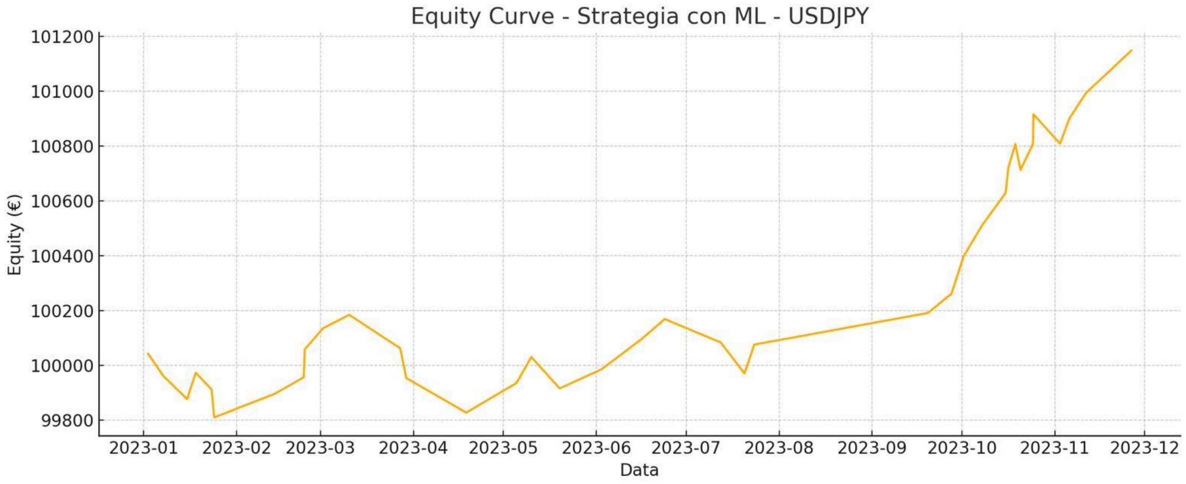
5.9. - USD/JPY (0.05 lots)
In the comparison between the strategy with and without Machine Learning (ML) on USD/JPY, significant differences emerge in terms of returns, consistency, and capital behavior.

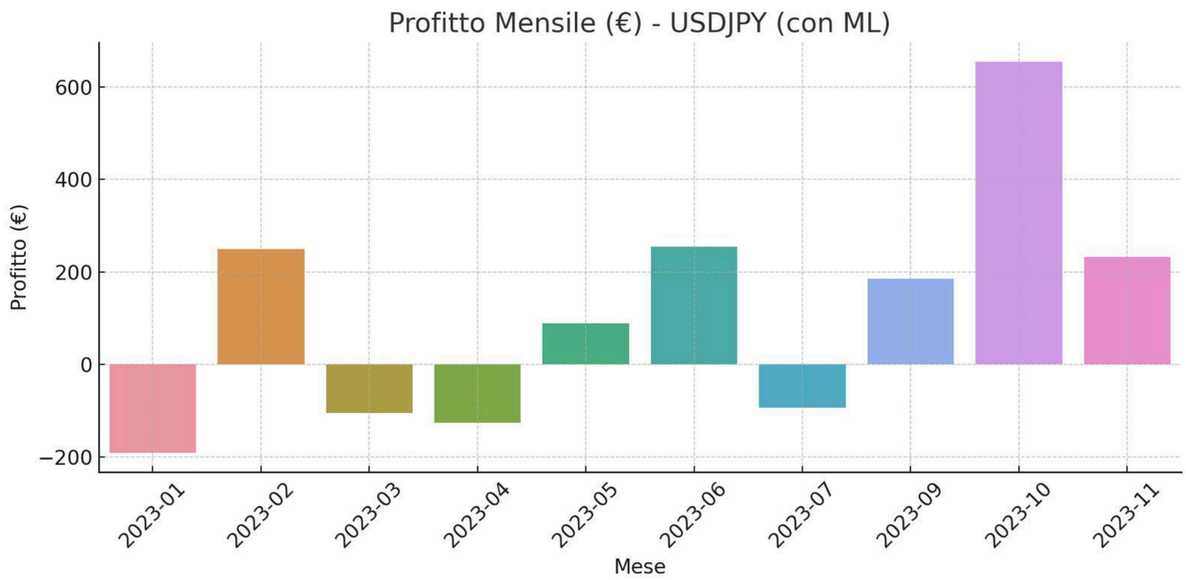
The non-ML strategy exhibits a steady equity growth without substantial corrections, delivering consistent returns throughout the months. Monthly profits are stable and well distributed, with notable peaks in May and September, and a generally positive trend. The equity curve shows a gradual and smooth ascent, free from sideways phases or significant drawdowns.

In contrast, the ML-based strategy features a first half of the year marked by sideways movement and frequent corrections, resulting in a slow and non-directional accumulation phase.
Starting in October, the model demonstrates a significant acceleration in returns, with a sharply rising equity curve and a maximum peak in November.
Monthly performance is more volatile compared to the non-ML version: early months show losses, followed by increasing profits concentrated in the latter part of the year.

This behavior suggests an initial learning or adaptation phase of the model, which eventually culminates in a strong profit surge during the final months.

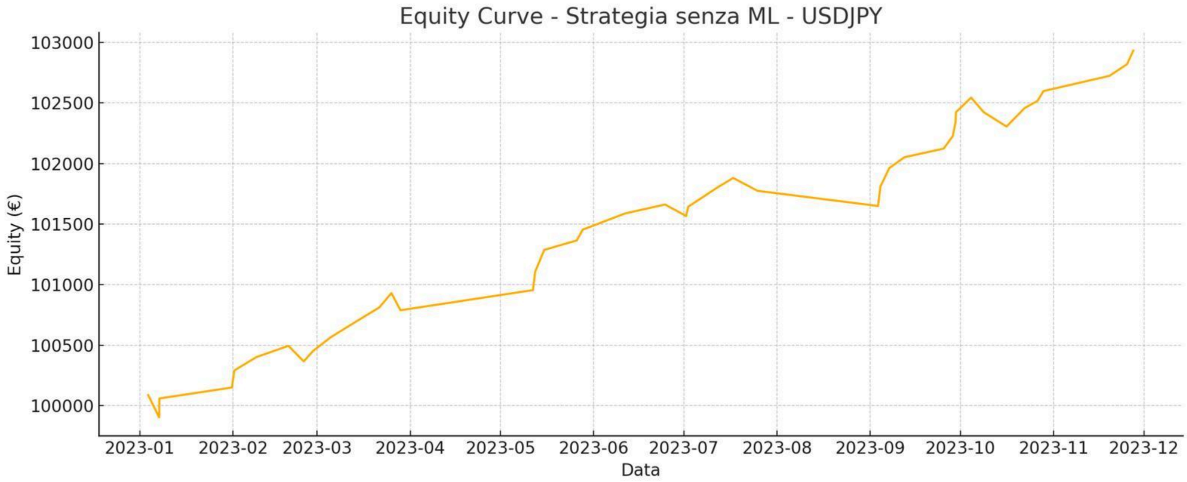
Strategy with Machine Learning

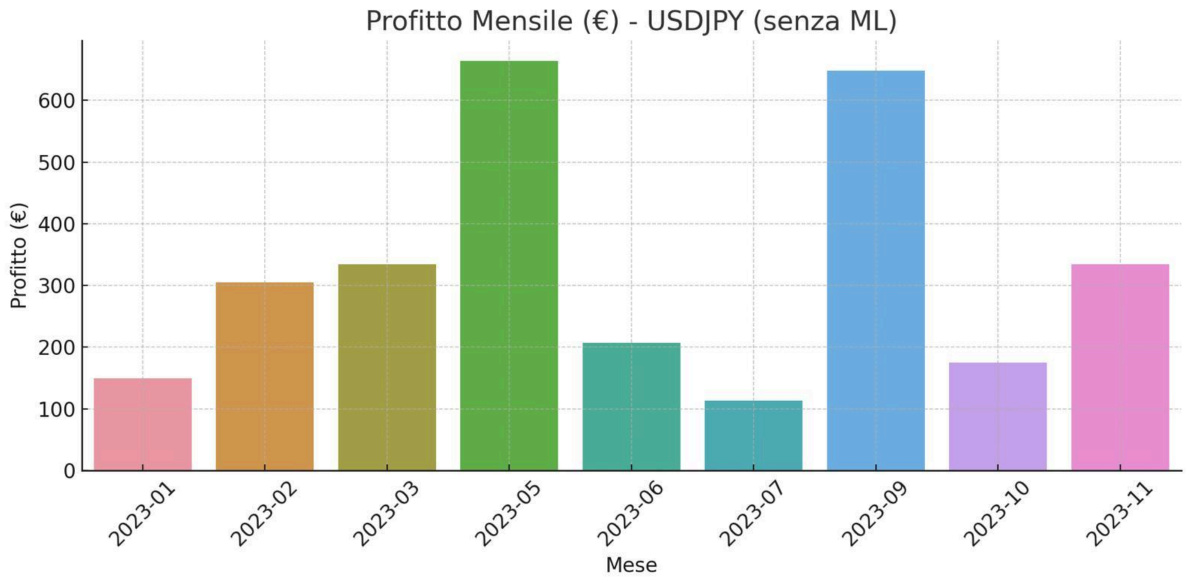




Win Rate	67.57%
Max Drawdown	0.35%
Absolute Profit	1.148$
Percentual Profit	1.15%

Strategy without Machine Learning





Win Rate	81.4%
Max Drawdown	0.23%
Absolute Profit	2.931$
Percentual Profit	2.93%

Conclusion USD/JPY

Overall, the ML-based model demonstrates greater reactivity in the final part of the year, but at the cost of a more turbulent initial phase.

The non-ML version, on the other hand, prioritizes stability and continuity from the outset, proving to be more robust, though less dynamic over the long term.

5.10. - AUD/USD (0.05 lots)

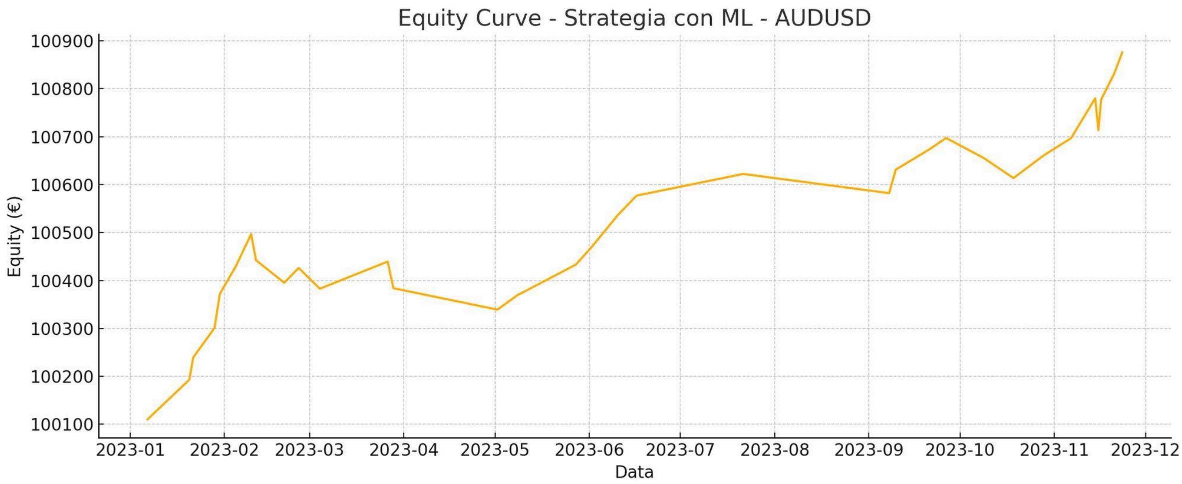
The Machine Learning-based strategy applied to AUD/USD showed a fluctuating but overall positive performance. The equity curve displays gradual growth, interspersed with consolidation phases and occasional intermediate corrections. Despite some oscillations in the first half of the year, the model maintained a generally bullish direction, closing the year with an equity increase.

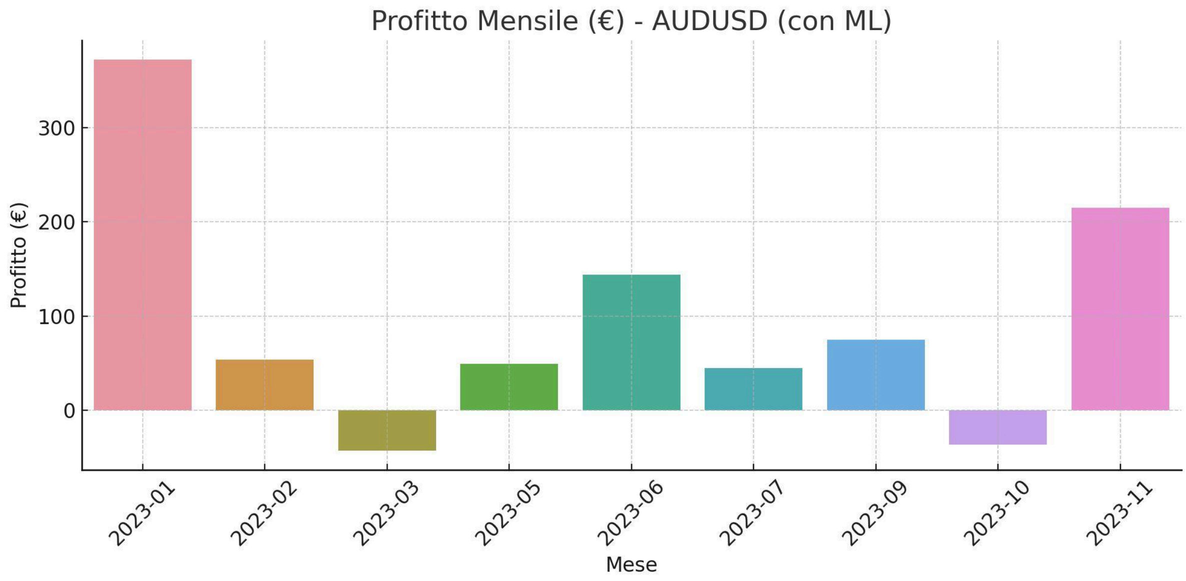
From a monthly perspective, 2023 opened with a very strong January (over $350), followed by a slowdown between February and April. March ended in the red, but the following months saw a progressive recovery, with performance consistently above zero—except in October. November marked another strong month for the strategy, contributing significantly to the final result.

When comparing this performance to the non-ML strategy, we note that it also delivered positive and consistent results. The equity curve is more regular, with fewer noticeable corrections. However, in terms of monthly profits, the traditional strategy tends to have a more stable distribution, with more months in profit and less volatility.

February, March, June, July, and October were particularly favorable months for the non-ML model.

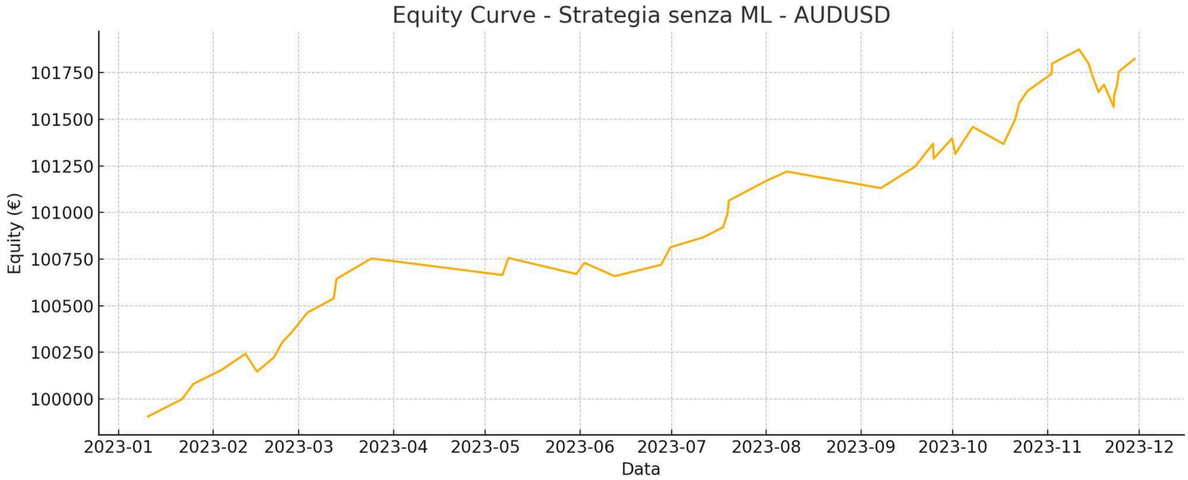
Strategy with Machine Learning

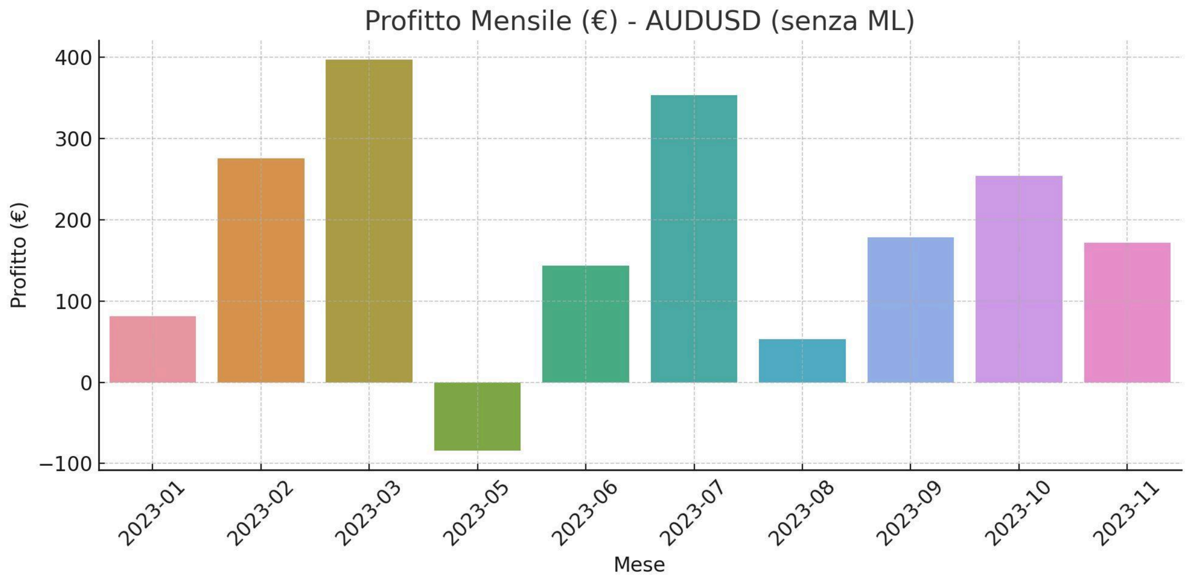




Win Rate	72.73%
Max Drawdown	0.16%
Absolute Profit	875.94$
Percentual Profit	0.88%

Strategy without Machine Learning





Win Rate	73.47%
Max Drawdown	0.30%
Absolute Profit	1.824$
Percentual Profit	1.82%

Conclusion AUD/USD
In summary, on AUD/USD, the Machine Learning model delivered good results, but with greater variability compared to the non-ML strategy, which demonstrated greater monthly stability.
Both strategies generated profit, but the ML version required more time to consolidate its advantage.

6. Comparative Analysis With/Without ML

The integration of Machine Learning (ML) into the trading strategy yielded mixed results depending on the target market, showing significant advantages in some cases and weaker performance in others compared to the traditional approach.

In high-volatility and strongly directional markets such as DJ30, DAX40, and BTC/USD, the ML-based model demonstrated a clear advantage.
In particular, on BTC/USD, the equity curve revealed robust and consistent growth

throughout the year, with significantly higher returns than the non-ML version. Similarly, on DJ30 and DAX40, the ML approach enabled better identification and exploitation of impulsive market phases, resulting in steeper equity curves and higher monthly profits overall.

In the case of the S&P500, the difference between the two strategies was more limited but still in favor of the ML model, which provided smoother progression and better handling of sideways phases.
On EUR/USD, the ML strategy also performed well, with a rising equity curve and a more diversified profit distribution, although it did not clearly outperform the non-ML version, which stood out for its continuity and monthly stability.

In more complex or noisy markets, such as USOIL and AUD/USD, the results of the ML model were more mixed.
On USOIL, the ML strategy showed a constant equity deterioration, significantly underperforming the base version. This suggests that the model failed to correctly interpret market signals or filter out the noise typical of this asset.
On AUD/USD, by contrast, the ML strategy achieved a positive return, though it was less stable than the non-ML version, which exhibited smoother growth and was less prone to drawdowns.

As for USD/JPY, the ML approach ended the year in positive territory, with a performance acceleration in the final months. However, the traditional version performed better overall, delivering a more consistent equity curve and stronger monthly profits, especially during the mid-year period.

Conclusion

The use of Machine Learning has proven particularly effective in markets with a strong directional component and in high-volatility environments, where the analysis of non-linear features and complex patterns can provide an informational edge.
However, in more stable or noisy markets, where stochastic factors dominate, the traditional strategy proved to be more reliable and performant.

This analysis highlights the importance of careful asset selection when applying ML models, and suggests the potential of hybrid systems, in which the activation of the model is conditional upon the presence of market conditions that are favorable to its predictive nature.

7. Conclusions and Future Developments

The project demonstrated the effectiveness of integrating Machine Learning models into an algorithmic trading strategy based on institutional logic.
The primary objective was to test the real impact of predictive models on the performance of an Expert Advisor (EA) in a multi-asset, multi-strategy environment, carefully evaluating the benefits and limitations compared to a traditional approach without AI.

Specifically, two models were employed: a Random Forest for trade prediction, and a risk allocation model (AMM) based on Linear Regression and Random Forest.
Both were developed in C++, integrated into MT4 via DLL, and trained on a structured dataset containing the following features: EMA, ADX, POC, VAH, VAL, volume, and ATR.
The dataset was internally created by combining proprietary indicators with historical market data.

The results presented in this report refer to these models trained on the specified dataset and may vary with retraining or on different time samples.

The purpose of the project is to make the model architectures and obtained results publicly available, promoting a transparent and replicable framework that can serve as a foundation for further advancements in AI-driven algorithmic trading.

All materials used in the project, including:

- the Volume Profile indicator (POC, VAH, VAL),
- the ML models (prediction and risk allocation),
- the datasets used for training,
  will be made available.

Developed and published by

[Marco Fedeli]
Quantitative Trader and AI Architect