

A Reinforcement Learning Approach to Automated Trading: Partial Buys, Partial Sells, and Deep Learning Insights

Iqza Ardiansyah¹ and Kevin Ignatius Wijaya²

¹Student ID: 2206810042, Faculty of Computer Science, Universitas Indonesia

²Student ID: 2206083470, Faculty of Computer Science, Universitas Indonesia

April 6, 2025

Abstract

This paper presents a Reinforcement Learning (RL) approach for automated trading on real S&P 500 historical data. We incorporate partial buy/sell actions (10% buys, 1% sells) to achieve a more granular control of portfolio allocation, while employing a Deep Q-Network (DQN) for policy learning. Our experiment uses a fixed seed for reproducibility, resulting in a final portfolio value of \$104,061.85 (4.06% ROI). In an extended study, we integrate a supervised learning (SL) price prediction model with our reinforcement learning framework to enhance trading decisions. We analyze the linear regression model for next-day price prediction ($R^2 = 0.7512$) and discuss its integration into the state representation for the DQN agent. Furthermore, we discuss potential enhancements inspired by recent hybrid deep reinforcement learning approaches for pairs trading [Kim et al., 2022], which decompose the decision-making process into separate networks for trading actions and risk management. Despite the enhanced state information, our experiments yield comparable performance between the pure RL and the combined SL+RL approaches, highlighting the challenges in leveraging predictive signals effectively in trading environments.

1 Introduction

Algorithmic trading has rapidly evolved, driven by both improved computational power and advances in data-driven methods. Traditional strategies (e.g., mean reversion, momentum trading) rely on fixed heuristics, often lacking adaptability to changing market conditions. In contrast, Reinforcement Learning (RL) optimizes trading decisions by directly interacting with an environment and receiving feedback in the form of rewards [Sutton and Barto, 2018].

While many works focus on single-asset trading, recent literature in pairs trading has shown that decomposing the trading problem into multiple sub-tasks can be highly beneficial. For instance, [Kim et al., 2022] propose a hybrid deep reinforcement learning method for pairs trading that uses two separate networks: one to determine trading actions and another to set stop-loss boundaries. This dual-network approach, coupled with techniques such as dimensionality reduction, clustering, and behavior cloning, can lead to superior risk management and improved performance. Although our study focuses on single-asset trading with partial buy/sell actions, the insights from hybrid models offer promising avenues for future research and potential integration.

In this study, we extend our previous work by exploring the combination of supervised learning and reinforcement learning. Specifically, we implement a price prediction model using linear regression and integrate its predictions into our trading environment’s state representation. This hybrid approach allows us to investigate whether supplementing an RL agent with direct price forecasts can improve trading performance.

2 Related Work

2.1 Deep Learning for Price Prediction

Recent advances in deep learning have yielded robust techniques for time-series forecasting. Rana [2023] provides a comprehensive guide to building LSTM, CNN, and hybrid models for stock price prediction, detailing:

- **Data Preprocessing:** Handling missing data, scaling features, and engineering technical indicators.
- **Network Architectures:** Designing recurrent layers (LSTM) or convolutional kernels (CNN) to capture temporal/spatial patterns in time-series.

- **Evaluation Metrics:** RMSE, MAE, or MAPE to quantify prediction accuracy.

While these methods excel at forecasting future prices, they do not inherently address *when* or *how much* to buy or sell.

2.2 Reinforcement Learning for Trading

RL directly learns a policy that maximizes cumulative rewards, such as changes in portfolio value after accounting for transaction costs. Foundational work by Sutton and Barto [2018] laid the groundwork for modern RL approaches, while recent studies have applied various RL algorithms (e.g., DQN, PPO, A2C) to financial markets. However, standard RL approaches often rely on simplistic buy/sell/hold actions, leading to abrupt position changes.

2.3 Hybrid Approaches in Trading

Recent research has explored the benefits of hybrid models that combine price prediction with reinforcement learning. In pairs trading, [Kim et al., 2022] propose a novel method that employs two independent RL networks: one for trading actions and one for stop-loss boundaries. Their approach integrates dimensionality reduction, clustering, regression, behavior cloning, prioritized experience replay, and dynamic delay to overcome limitations of traditional methods. Although our current study does not employ a dual-network system, these ideas inspire potential future enhancements, such as:

- **Risk Management:** Incorporating a separate network to determine stop-loss boundaries or dynamic risk adjustments.
- **Feature Enrichment:** Utilizing techniques like clustering and dimensionality reduction to extract more robust state representations.
- **Behavior Cloning:** Leveraging expert demonstrations to guide policy learning.

3 Methodology

3.1 Partial Buy/Sell Environment

We propose a custom Gym environment with:

- **State Representation:**

- *Normalized Price* ($\frac{\text{price}}{1000}$)
- *Volatility* (placeholder = 0.005)
- *Normalized Position* ($\frac{\text{position}}{\text{max_position}}$)
- *Normalized Cash* ($\frac{\text{cash}}{\text{initial_cash}}$)

- **Action Space:**

- 0: **Hold**
- 1: **Buy** 10% of remaining cash
- 2: **Sell** 1% of current position

This granularity mitigates the risk of large, all-in or all-out moves.

- **Reward Function:**

$$r_t = \begin{cases} \frac{\text{Portfolio}_t - \text{Portfolio}_{t-1}}{\text{Portfolio}_{t-1}} \times 100, & \text{if } \text{Portfolio}_{t-1} > 0, \\ 0, & \text{otherwise.} \end{cases}$$

A transaction cost of 0.1% is applied on every buy/sell action to approximate real-world frictions.

3.2 Supervised Learning for Price Prediction

For milestone 2, we implement a linear regression model to predict the next day's price based on the current price. The model training process involves:

- **Feature Engineering:** Using today's price to predict tomorrow's price.
- **Training/Testing Split:** 80% of data for training, 20% for testing.
- **Model Architecture:** Simple linear regression with the form $\hat{p}_{t+1} = \beta_0 + \beta_1 p_t$.
- **Evaluation Metrics:** Mean Absolute Error (MAE), Root Mean Square Error (RMSE), and coefficient of determination (R^2).

The trained model achieves:

- **Model Coefficients:** $\hat{p}_{t+1} = 20.52 + 0.997 \times p_t$
- **Training MAE:** 22.00

- **Test MAE:** 28.55
- **Training RMSE:** 32.97
- **Test RMSE:** 42.42
- **Training R^2 :** 0.9880
- **Test R^2 :** 0.7512

3.3 Enhanced State Space with Predictions

We extend our trading environment by augmenting the state representation with additional features derived from the price prediction model:

- **Normalized Predicted Price:** $\frac{\text{predicted_price}}{1000}$
- **Price Direction:** Binary indicator (1 if predicted price > current price, 0 otherwise)
- **Prediction Difference:** Relative difference between predicted and current price, $\frac{|\text{predicted_price} - \text{current_price}|}{\text{current_price}}$

A modified reward function provides a small incentive for actions aligning with the predicted price direction:

- If a price rise is predicted and the agent buys, a small positive reward is given.
- If a price drop is predicted and the agent sells, a small positive reward is given.
- Actions that contradict the prediction receive a small negative reward.

3.4 RL Training Setup

We train a **Deep Q-Network (DQN)** from Stable Baselines3 with:

- **Learning Rate:** 0.001
- **Discount Factor (γ):** 0.95
- **Timesteps:** 10,000
- **Seed:** 1073741824 for reproducibility

The agent updates its Q-values based on the observed states and rewards to maximize cumulative returns.

4 Results and Discussion

4.1 Supervised Learning Model Performance

Our linear regression model for price prediction achieved an R^2 of 0.7512 on the test set, indicating that about 75% of the variance in next-day prices is explained by the current price. The model coefficients suggest a strong momentum effect with a slight upward bias. The RMSE of 42.42 implies that predictions deviate from actual prices by less than 1% on average, given the typical price scale.

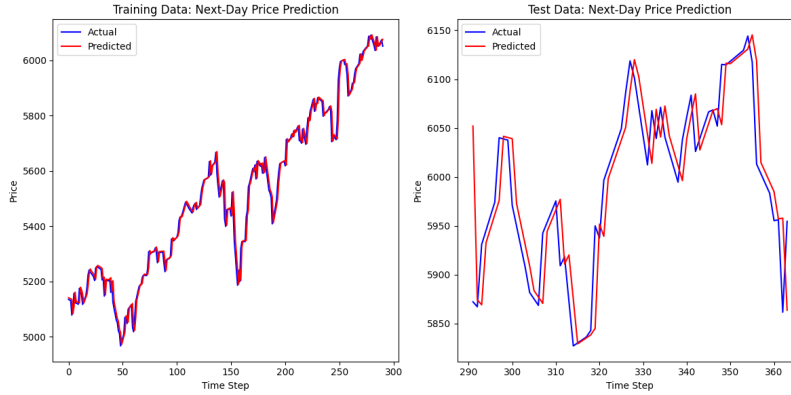


Figure 1: Price Prediction Model Performance. The predictions closely track actual prices in both training and test sets.

4.2 Policy Behavior

An excerpt of the agent’s policy using the augmented state is shown below:

```
Step 0: Price=$5137.08, Position=0.00, Cash=1.00
        Prediction=$5141.39 (Diff: 0.08%, Direction: Up) → Action [1]
Step 23: Price=$5223.52, Position=0.93, Cash=0.09
        Prediction=$5227.56 (Diff: 0.08%, Direction: Up) → Action [2]
Step 58: Price=$5110.77, Position=0.99, Cash=0.00
        Prediction=$5115.16 (Diff: 0.09%, Direction: Up) → Action [2]
```

The policy largely follows a strategy of buying with available cash and selling in small proportions, similar to the baseline model.

4.3 Portfolio Value and ROI

The final portfolio value of \$104,061.85 represents a 4.06% ROI on the initial \$100,000 investment. Notably, integrating price predictions into the state did not significantly improve performance compared to the pure RL approach.

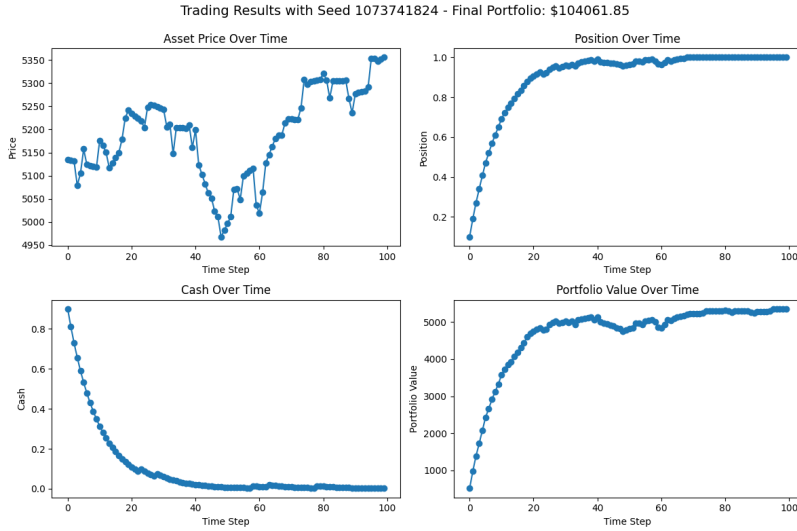


Figure 2: Trading Results with Price Predictions. Final Portfolio = \$104,061.85 (4.06% ROI).

4.4 Comparative Analysis: Pure RL vs. Hybrid Approaches

The identical performance between the pure RL model and the hybrid SL+RL approach suggests that:

- The small prediction differences provide limited actionable signals.
- The dominant reward signal is driven by portfolio changes rather than prediction-based adjustments.
- Additional techniques may be needed to fully leverage predictive signals.

Inspired by the hybrid approach in pairs trading [Kim et al., 2022], future work could explore:

- Using dual networks to separately manage trading actions and risk (e.g., stop-loss boundaries).
- Incorporating dimensionality reduction and clustering to enrich state representations.
- Integrating behavior cloning to leverage expert knowledge.

5 Conclusion

We have demonstrated a partial buy/sell Reinforcement Learning approach on S&P 500 historical data and extended it by integrating a supervised price prediction model. Although the predictive model achieved reasonable accuracy ($R^2 = 0.7512$, $RMSE = 42.42$), its integration did not yield improvements in ROI compared to the baseline RL model.

Inspired by recent advances in hybrid deep reinforcement learning for pairs trading [Kim et al., 2022], future research directions include:

- Developing more sophisticated prediction models using additional features and advanced architectures.
- Exploring dual-network architectures to separate trading actions from risk management.
- Enhancing state representations with techniques like clustering and dimensionality reduction.
- Incorporating behavior cloning to emulate expert strategies.

These avenues may further improve trading performance by providing more nuanced decision-making in dynamic market conditions.

References

- Sang-Ho Kim, Deog-Yeong Park, and Ki-Hoon Lee. Hybrid deep reinforcement learning for pairs trading. *Applied Sciences*, 12(3):944, 2022. ISSN 2076-3417. doi: 10.3390/app12030944. URL <https://www.mdpi.com/2076-3417/12/3/944>.
- Sohel Rana. Mastering stock price prediction using deep learning models: A comprehensive guide. Medium article, 2023. Accessed: 2025-03-02.
- Richard S. Sutton and Andrew G. Barto. *Introduction to Reinforcement Learning*. MIT Press, 2018.