**Exploring Foreign Exchange Reinforcement Learning Methods**

Team 2 - Final Project Paper

Kaleigh Jaron, Mike Smith, Atul Lanka, [Alex Bergmann](mailto:alexanderbergmann2021@u.northwestern.edu)

**1 Abstract**

This paper describes the implementation of a Deep Reinforcement Learning based system that is capable of automatically trading on the Foreign Exchange Market.

**1.1 FX Trading**

Foreign Exchange Trading is the process of trading currencies for a profit on the Foreign Exchange, a financial marketplace that determines the exchange rate for every country's currency denomination.This provides a platform for buying, selling, exchanging, and speculation of currencies. These currencies are traded in pairs, which means the worth of one of the currencies in that pair has a direct relationship on the worth of the other. Put simply, this decides how much of a particular country’s currency another country can purchase, and vice versa.

**1.2 Reinforcement Learning**

Reinforcement Learning is a subfield of ML that seeks to automatically learn the process of optimal sequential decision making. Deep RL refers to combining Reinforcement Learning with Deep Neural Networks, which provide a practical and effective approach to improving several aspects of the traditional Reinforcement Learning agent learning process.

**1.3 Reinforcement Learning in Finance**

Reinforcement Learning can be a useful tool to learn how to behave in an environment by interacting with it, which is accomplished primarily by performing actions and receiving feedback. In applying Reinforcement Learning to FX trading, the goal is to interact with the historical data and trading environment and in a way that allows an RL agent to learn a profitable trading strategy through trial and error. In other terms, build an RL agent capable of selecting the actions, or decisions on which currency pairs to buy, hold or sell in a way that maximizes the expected cumulative reward or total profit.

Developing a good investment strategy is a major objective within the area of financial trading, which makes RL an ideal method to provide better assistance to make trading decisions on the FX market. The model(s) outlined in the paper utilizes historical data from OANDA. This is a FX trading dataset which contains the historical exchange rate for the Euro-Dollar(EUR\_USD) currency pair between a timeframe starting at 1/2/2017 and ending at 3/9/2021 at 10 minute intervals and includes information associated with each pair such as volume, open/close price and high and low pricing. The timestamp for each row was encoded into cos/sin representation for better interoperability.

With a better understanding of the relationship between currencies and particularly, which factors lead to fluctuations in terms of profit or loss in regard to the value of a certain currency pair. Participants in the FX marketplace can be more confident attempting to target specific currencies or pursuing a certain trading strategy that matches the objectives of the organization or individual. A successful FX Trading Reinforcement Learning system is of significant practical benefit, as automating the discovery of investment strategies eliminates the need for manual building of financial models and reduces uncertainty.

**2 Introduction**

The foreign exchange market is among the largest of financial markets in the world,

reaching $6.6 trillion per day in April 2019 according to the 2019 Triennial Central Bank Survey

of Foreign Exchange and Over the Counter derivatives markets (Thiruchelvam). Accordingly, there is significant interest in the application of artificial intelligence to trading strategy for foreign exchange. This paper surveys the existing research on applications of one type of artificial intelligence, reinforcement learning, to foreign exchange trading, then examines several versions of such a model under varied parameters to test which approach is most effective.

**3 Literature Review**

Foreign exchange as an application for artificial intelligence has a long history. There are two broad categories of related work. First, there are natural language processing approaches typified by Khadjeh Nassirtoussi et. al. (2015) which seek to mine headline or other text as a predictor of foreign exchange rate movements. These are outside the scope of this paper but may be of interest for those pursuing a holistic view. The second are various modeling approaches to trading data. Hu et. al. (2021) and Islam et. al. (2020) offer helpful surveys of various branches of machine learning application to foreign exchange trading and their evolution, particularly focusing on developments in the last five years.

Among these, reinforcement learning is the focus of this paper. Reinforcement learning has been considered as an avenue to explore foreign exchange trading for many years, dating at least to Dempster’s work in the late 1990s and early 2000s, itself building on binary string work popularized by Holland in the classic Adaptation in Natural and Artificial Systems (1975). Dempster predates the renaissance of machine learning in the 2010s; at the time (2002) he and his partners were able to demonstrate theoretical success using a high-frequency foreign exchange trading framework combining a genetic algorithm with reinforcement learning, but only in a hypothetical arena without transaction costs. To his credit, Dempster acknowledged this shortcoming and outlined additional scope for improvement such as adapting to a risk-adjusted return approach and building in stop-loss downside protection. He also highlighted the recurrent reinforcement learning approach developed by Moody (1998) as a possible improvement, later expanded by Gold (2003). Bates (2003), working with Dempster and Romahi, took an interesting angle, using evolutionary reinforcement learning to analyze whether order book and order flow data might be used to reverse engineer trading strategy. Hryshko and Downs (2004) developed further incremental steps but noted that to be truly useful, a real-time system would need to be developed. A few years later, in 2006, Dempster and Leemans had more success using adaptive reinforcement learning, based on recurrent reinforcement learning, in conjunction with a risk management overlay and a dynamic utility optimization layer. In 2009, Corrêa et. al. explored the application of hierarchical neuro-fuzzy logic to the reinforcement learning foreign exchange model, successfully detecting long-term strategies and enabling profitability with fewer trades. For more on fuzzy logic and foreign exchange, see Yu et. al. (2007) for a discussion of its application to foreign exchange rates forecasting.

More recent developments in the field focus on refinement of the methodology. For example, Carapuço et. al. (2018) used a Q-learning algorithm combined with just a three-layer ReLU neural network and novel state and reward signals to generate yearly average profit of about 13% -19% on Euro/USD data from 2010-2017. Tsai and Wang (2019) use a statistical arbitrage policy and a heat-map view of a Gramian Angular Field, comparing execution between Deep Q Learning and Proximal Policy Optimization. Rundo (2019) combined deep learning with reinforcement learning to forecast short-term trends in foreign exchange, achieving accuracy of about 85% and high-frequency trading return on investment of a whopping 98% with nearly 16% less investment. Lee (2019), used his Chaotic Oscillatory Multi-Agent-based Neuro-Computing System (“COSMOS”) to combine a supervised learning forecaster with a reinforcement learning trader without data overtraining and deadlock which bedevil recurrent neural networks using sigmoid or gaussian-based activation functions. The COSMOS implementation scope is massive, with 39 financial signals for real-time trading of 129 products including cryptocurrency, foreign exchange, commodities, and financial indices, generating 1% prediction error and average monthly returns ranging from 8-13%. Zeng and Khushi (2020), while not focusing on a trading model, proposed an innovative approach to forecasting using a combination of three models including a noise-removing one and a recurrent neural network. The denoising aspect in particular has interesting implications for incremental improvement in trading models.

Also of interest is recent work in reinforcement learning on stock market trading, in many ways analogous to foreign exchange trading. Si et. al. (2017) explored deep reinforcement learning’s application to multi-objective modeling of stock index future trading. Kang et. al. (2018) focused on portfolio management with a standalone deep reinforcement learning model. Zhu et. al. (2018) combined reinforcement learning with box normalization for superior trading results. Shin et. al. (2019) introduced a combination of CNN, LSTM, and reinforcement learning using stock trading charts, generating probabilities for buy, sell, and hold. Finally, Li et. al. (2020) compared DQN, Double DQN, and Duelling DQN methodologies in stock price prediction, showing DQN to be superior.

**4 Methods**

In designing this project it was determined there would be 3 areas to explore, the custom environment, the neural network, and the reinforcement learning model. The custom environment would include aspects such as the actions the agent could take, the method of introducing rewards, and the way observation was output to the neural network. The neural network would include the architecture of the model’s decision making process, and would explore the benefits of models such as a dense neural network, convolutional neural network, and Long Short Term Memory (LSTM) networks. LSTMs are very powerful in sequence prediction problems because they're able to store past information and have feedback connections. Finally, the modifications to the DQN reinforcement learning model would explore how the policy, window size, and dueling architecture modified the results.

The data being used in this experimental process consisted of foreign exchange data collected between January 2017 and March 2021 in increments of 10 minutes, 1 hour, and 4 hours for the EURO/USD currency pair. To test the functionality of our models simulated data manually generated from a repeating sine wave was also used.

The environment was designed in a way that only one trade could be going on at any given time. Every step, the agent could make the decision to enter a short or a long trade, wait, or exit an existing trade. If the action selected by the agent was invalid in the current state, the environment would default to waiting until the next step. This would happen, for example, if the agent tried to enter a new trade when another trade was still active. Once a trade was exited, the episode would end. Below is a list of the efforts made to modify the environment in the hopes of improving performance.

| **Environment Comparisons Made** | |
| --- | --- |
| An action space of 0/1 indicating going short/long at every step. In this scenario, waiting only took place within a trade. 1 trade needed to be active at all times. | An action space of 0-3. 0 = sell all units, 1= wait, 2 = enter long trade, 3 = enter short trade. |
| Rewarding directly based off of the unrealised profit / loss from that step. A $50 profit would mean a reward of 50. | Rewarding in uniform amounts. If the model made profit that turn, reward +1. If the model lost money that turn, reward - 1. |
| Since we are executing one position at a time, we teach the agent to follow the proper sequence of closing a position and generating profit/loss by ignoring any invalid sequence of actions [Example: long order followed by a short order] | Ignoring but also penalizing the agent - reward = -1 for selecting an invalid sequences of actions in an effort to accelerate learning accurate decision making |
| Presenting observation data of a small time frame, such as 10 minute increments | Presenting observation data of a large time frame, up to 24 hour increments |
| Presenting a small number of rows of data per observation (1-10) | Presenting a large number of rows of data per observation (24-100) |
| Presenting raw numeric data in each observation | Presenting normalised and scaled numeric data in each observation |
| Presenting simple Open, Close, High, Low, Volume data in the observation | Presenting indicators in the observation such as exponential moving average, simple moving average, bollinger bands, RSI, etc. |
| Presenting numeric data in the observation | Presenting an image of a generated OHLC chart in the observation |

The neural networks designed for this project took in the input size of the observation space and multiplied it by the window size before flattening it. From there, we explored a variety of layer structures using the Keras platform, including dense neural networks, 1D and 2D convolutional neural networks, and LSTM. The exact model structures used can be viewed in the appendix.

Using Keras reinforcement learning, we also explored the following avenues of modifying the reinforcement learning process.

| **Model Comparisons Made** | |
| --- | --- |
| Using a small window size (1). This only allowed the model to look at 1 step at a time. | Using a large window size (10). This allowed the model to look at multiple steps at a time. |
| Using the Epsilon Greedy Policy | Using the Boltzmann Q Policy |
| Using a normal DQN architecture | Using a dueling DQN architecture |
| Using a smaller learning rate (~1e-4) | Using a larger learning rate (~1e-2) |
| Updating the target model every turn | Using soft updates to update the target model gradually |

**5 Results**

Part of the reason we explored so many avenues for improvement is because of the lack of tangible results being produced. Unfortunately, despite generating and comparing a plethora of different combinations of environments, networks, and reinforcement learning structures - as outlined in the comparison tables above - we were unable to produce meaningful results. Studying the sequence of actions, they often appear random and not in the intended order, even after several episodes of training. This leads us to believe that there is a possibility of a fundamental issue in the code regarding how the data is being processed between the neural network and reinforcement learning model. All efforts depicted in the methods above resulted in actions that continued to appear random and did not improve the reward over time, even with running the model with data in a simple sine wave configuration. The silver lining in all this is that the rewards/profits were effectively +/- 1% range of our starting balance. So while we were not able to generate significant profits, our model performed conservatively and generated minimal losses. Below is a table of some of the graphs generated during training.

| DQN RL Model  (Larger image in appendix image 6) | Dueling DQN RL Model  (Larger image in appendix image 14) |
| --- | --- |
| 1D CNN Model  (Larger image in appendix image 9) | 2D CNN Model  (Larger image in appendix image 12) |
| Epsilon Greedy Policy  (Larger image in appendix image 6) | Boltzmann Q Policy  (Larger image in appendix image 15) |

As can be seen in the above graphs meaningful changes are taking place between different modifications to the model. Even though none of the models are resulting in obvious proffits, the 2D CNN, for example, consistently produces both higher profits and less loss then the 1D CNN model. A similar comparison can be made between the standard DQN method vs the dueling DQN architecture, with the dueling DQN being more successful. Once the bug in the code preventing the model from learning correctly is overcome, these will be areas to further explore.

**6 Conclusion**

We acknowledge that the domain of finance and trading is random and uncertain, and not being able to meet our expectation and create a profitable strategy through reinforcement learning is not surprising - there are no guarantees in trading. However, there are several improvements and methods we still need to explore in order to break through the barrier of randomness. Apart from our continued efforts to refine our current LSTM model, future considerations can include using genetic algorithms in tandem with reinforcement learning, natural language processing to gauge general sentiments from articles or social media and additional optimization layers that can behave more adaptively to the market or data trends itself.

**7 Appendix**

| **Figure 1**  LSTM Model Summary |  |
| --- | --- |
| **Figure 2**  LSTM Profit Plot |  |
| **Figure 3**  LSTM Reward Plot |  |
| **Figure 5**  DNN Model Summary |  |
| **Figure 6**  DNN Profit Plot |  |
| **Figure 7**  DNN Reward Plot |  |
| **Figure 8**  1D Convolutional Neural Network Summary |  |
| **Figure 9**  1D CNN Profit Plot |  |
| **Figure 10**  1D CNN Reward Plot |  |
| **Figure 11**  2D Convolutional Neural Network Summary |  |
| **Figure 12**  2D CNN Profit Plot |  |
| **Figure 13**  2D CNN Profit Plot |  |
| **Figure 14**  DNN with average dueling activated Profit Plot |  |
| **Figure 15**  DNN with Boltzmann Q Policy |  |

**8 References**

Bates, R.G, M.A.H Dempster, and Y.S Romahi. “Evolutionary Reinforcement Learning in FX Order Book and Order Flow Analysis.” In 2003 IEEE International Conference on Computational Intelligence for Financial Engineering, 2003. Proceedings, 355–62. IEEE, 2003. https://doi.org/10.1109/CIFER.2003.1196282.

Carapuço, João, Rui Neves, and Nuno Horta. “Reinforcement Learning Applied to Forex Trading.” Applied Soft Computing 73 (2018): 783–94. https://doi.org/10.1016/j.asoc.2018.09.017.

Corrêa, Marcelo F, Marley Vellasco, Karla Figueiredo, and Pedro Vellasco. “Trading Strategy in Foreign Exchange Market Using Reinforcement Learning Hierarchical Neuro-Fuzzy Systems.” In Advances in Neuro-Information Processing, 461–68. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009. <https://doi.org/10.1007/978-3-642-03040-6_56>.

Dempster, M.A.H, and V Leemans. “An Automated FX Trading System Using Adaptive Reinforcement Learning.” Expert Systems with Applications 30, no. 3 (2006): 543–52. <https://doi.org/10.1016/j.eswa.2005.10.012>.

Dempster, M.A.H, T.W Payne, Y Romahi, and G.W.P Thompson. “Computational Learning Techniques for Intraday FX Trading Using Popular Technical Indicators.” IEEE Transactions on Neural Networks 12, no. 4 (2001): 744–54. <https://doi.org/10.1109/72.935088>.

Dempster, M. A. H, and Y. S Romahi. “Intraday FX Trading: An Evolutionary Reinforcement Learning Approach.” In Lecture Notes in Computer Science, 347–58. Berlin: Springer, 2002.

Gold, C. “FX trading via recurrent reinforcement learning.” Proceedings of the IEEE International Conference on Computational Intelligence in Financial Engineering 2003, op.cit., (2003): 363–371.

Holland, J. H. “Adaptation in natural and artificial systems.” University of Michigan Press, Ann Arbour, MI, 1975.

Hryshko, A, and T Downs. “System for Foreign Exchange Trading Using Genetic Algorithms and Reinforcement Learning.” International Journal of Systems Science 35, no. 13-14 (2004): 763–74. <https://doi.org/10.1080/00207720412331303697>.

Hu, Zexin, Yiqi Zhao, and Matloob Khushi. “A Survey of Forex and Stock Price Prediction Using Deep Learning.” Applied System Innovation 4, no. 1 (2021): 9–. https://doi.org/10.3390/asi4010009.

Islam, Md. Saiful, Emam Hossain, Abdur Rahman, Mohammad Shahadat Hossain, and Karl Andersson. “A Review on Recent Advancements in FOREX Currency Prediction.” Algorithms 13, no. 8 (2020): 186–. <https://doi.org/10.3390/a13080186>.

Kang, Qinma, Huizhuo Zhou, and Yunfan Kang. “An Asynchronous Advantage Actor-Critic Reinforcement Learning Method for Stock Selection and Portfolio Management.” In Proceedings of the 2nd International Conference on Big Data Research, 141–45. ACM, 2018. https://doi.org/10.1145/3291801.3291831.

Khadjeh Nassirtoussi, Arman, Saeed Aghabozorgi, Teh Ying Wah, and David Chek Ling Ngo. “Text Mining of News-Headlines for FOREX Market Prediction: A Multi-Layer Dimension Reduction Algorithm with Semantics and Sentiment.” Expert Systems with Applications 42, no. 1 (2015): 306–24. <https://doi.org/10.1016/j.eswa.2014.08.004>.

Lee, Raymond S.T. “COSMOS Trader – Chaotic Neuro-Oscillatory Multiagent Financial Prediction and Trading System.” The Journal of Finance and Data Science 5, no. 2 (2019): 61–82. https://doi.org/10.1016/j.jfds.2019.01.001.

Li, Yuming, Pin Ni, and Victor Chang. “Application of Deep Reinforcement Learning in Stock Trading Strategies and Stock Forecasting.” Computing 102, no. 6 (2020): 1305–22. https://doi.org/10.1007/s00607-019-00773-w.

Moody, John, Lizhong Wu, Yuansong Liao, and Matthew Saffell. “Performance Functions and Reinforcement Learning for Trading Systems and Portfolios.” Journal of Forecasting 17, no. 5‐6 (1998): 441–70. https://doi.org/10.1002/(SICI)1099-131X(1998090)17:5/63.0.CO;2-#.

Rundo, Francesco. “Deep LSTM with Reinforcement Learning Layer for Financial Trend Prediction in FX High Frequency Trading Systems.” Applied Sciences 9, no. 20 (2019): 4460–. <https://doi.org/10.3390/app9204460>.

Shin, H.-G., I. Ra, and Y.-H. Choi. “A Deep Multimodal Reinforcement Learning System Combined with CNN and LSTM for Stock Trading.” In Proceedings of the 2019 International Conference on Information and Communication Technology Convergence (ICTC), Jeju Island, Korea, 16–18 October 2019.

Si, Weiyu, Jinke Li, Peng Ding, and Ruonan Rao. “A Multi-Objective Deep Reinforcement Learning Approach for Stock Index Future’s Intraday Trading.” In 2017 10th International Symposium on Computational Intelligence and Design (ISCID), 2:431–36. IEEE, 2017. https://doi.org/10.1109/ISCID.2017.210.

Thiruchelvam, Sharon. “BIS Triennial OTC Survey Round-Up.” Global Investor, 2019.

Tsai, Yun-Cheng, and Chun-Chieh Wang. “Deep Reinforcement Learning for Foreign Exchange Trading,” 2019.

Yu, Lean, Kin Keung Lai, and Shouyang Wang. “Neural-Network-Based Fuzzy Group Forecasting with Application to Foreign Exchange Rates Prediction.” In Computational Science – ICCS 2007, 423–30. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007. https://doi.org/10.1007/978-3-540-72586-2\_61.

Zeng, Zhiwen, and Matloob Khushi. “Wavelet Denoising and Attention-Based RNN- ARIMA Model to Predict Forex Price.” In 2020 International Joint Conference on Neural Networks (IJCNN), 1–7. IEEE, 2020. <https://doi.org/10.1109/IJCNN48605.2020.9206832>.

Zhu, Yingying, Hui Yang, Jianmin Jiang, and Qiang Huang. “An Adaptive Box-Normalization Stock Index Trading Strategy Based on Reinforcement Learning.” In Neural Information Processing, 335–46. Cham: Springer International Publishing, 2018. https://doi.org/10.1007/978-3-030-04182-3\_30.