Advances in Educational Technology and Psychology (2021) 5: 81-92
Clausius Scientific Press, Canada
DOI: 10.23977/aetp.2021.55013
ISSN 2371-9400

Analysis and Forecast of Exchange Rate Based on Reinforcement Learning

Shifeng Wu*, Pan-yun Mao

Hunan University of Humanities, Science and Technology, Loudi, China

fengtree@126.com

**corresponding authors*

Keywords: Exchange rate forecasting, Deep learning, Reinforcement learning, Python, big data

Abstract: For individuals, exchange rate changes affect individuals' consumption of foreign goods and services; for government enterprises, exchange rate forecasting is the basis for corporate foreign exchange risk measurement and strategic adjustment. Therefore, exchange rate forecast analysis has importance research value. Based on the specific situation of exchange rate trading under big data, this paper analyzes and predicts the RMB exchange rate. At the same time, by using multiple reinforcement learning models, it is predicted when and what kind of buy-and-sell strategy can finally achieve effective returns. Finally, the model results of the model are compared between multiple currency exchanges, and the model is compared and demonstrated. The validity and adaptability ensure the accuracy of the model. The results of this paper model can be used as a reference for individuals and even the government for exchange rate changes.

1. Introduction

As we all know, the exchange rate is affected by many different factors [1]. Since ancient times, the exchange rate has been affecting the economic trend of a country and even the whole world. Similarly, the exchange rate is closely related to the life of ordinary people. Therefore, we need to analyze and forecast the exchange rate. The importance of the exchange rate is reflected in all aspects. For a country, the exchange rate is an important variable that affects the external and internal equilibrium of a country's economy. According to the exchange rate, we can accurately and clearly judge our country's comprehensive economic strength and therefore formulate the correct macroeconomic strategy; For enterprises and investors, the correct exchange rate forecast can give them some effective suggestions, so as to formulate the correct strategy, avoid investment risk and improve the rate of return.

For the big data analysis and forecast of exchange rate, at present, the forecast of exchange rate change trend is mainly based on the historical data of exchange rate in the past. By collecting a large

number of data, the corresponding mathematical model is established to realize the exchange rate forecast. At present, there are many factors that affect the exchange rate [2-4] and there are unstructured and uncertain factors, including national policies, public opinion and other accidental factors at the national level, or the dramatic changes of the exchange rate caused by the sudden increase of the import and export volume, It is difficult to grasp the accidental jump of the exchange rate market and the long-term effect caused by it only by the technical level such as analysis and prediction. Therefore, the big data analysis and prediction of exchange rate, in terms of the current research status and technology, can only be relatively accurate to fit the real situation, but can not be completely correct, but the research results still have a certain reference value.

So far, the analysis and prediction of exchange rate between various currencies has not been studied systematically and well[5-7], and most of the researches are based on time series, but the exchange rate is not only affected by time. Therefore, this paper analyzes the main indicators that affect the exchange rate changes, and uses deep learning model and reinforcement learning model to predict the exchange rate changes and trading strategies among various currencies, aiming to create a reference model that can provide people with the exchange rate changes and trading strategies.

The main structure of this paper is as follows .The second chapter introduces the DQN，Double DQN model. The third chapter shows the results of the model, and then analyzes the results of the algorithm to predict the exchange rate changes. The fourth chapter is a general summary of this paper.

2. Related Algorithms

Reinforcement learning algorithm [8] consists of four main components: subject, behavior, environment and reward. Reinforcement learning maximizes the reward value by taking different behaviors in the same state, calculating losses and adjusting parameters. The environment in the exchange rate forecast refers to the environment in which the exchange rate changes according to certain factors; Action refers to the behavior strategy made according to the environment. In the exchange rate, it is shown as selling exchange rate, buying exchange rate, and keeping unchanged. There are many options for reward, which means that according to the results of the behavior, the money earned can be used as reward in the exchange rate forecast. Reinforcement learning is a kind of unsupervised learning. Reinforcement learning will make behavior randomly in a state without any labels, and then get the result. Through the feedback of the result, adjust the previous behavior. Through continuous adjustment, the algorithm can learn what kind of behavior to choose in a certain state, and get the best results. Therefore, reinforcement learning can be used to make the best decision strategy for buying and selling exchange rate, so as to achieve the purpose of judging and forecasting the future exchange rate.

Reinforcement learning algorithms can be divided into the following categories: 1. Based on strategy, the purpose is to find the best strategy; 2. Based on the value, the goal is to find the maximum sum of rewards; 3. Based on behavior, the goal is to take the best action at every step.

Common reinforcement learning algorithms are mainly divided into the following: Q-learning, DQN, double DQN, dueling DQN, SARSA and so on. The purpose of exchange rate forecasting is to find the strategy to maximize the return, so we choose the value-based reinforcement learning model in the later models, which are mainly: DQN, double DQN, dueling DQN.

Q-learning algorithm [9] is the use of tables to record in a given state, take an action, what can be

rewarded, while recording the state value and reward and the corresponding action. This table is initialized to 0, and then the table will be automatically updated every step

$$Q(s,a) = R(s,a) + \gamma * \max\{Q(\tilde{s},\tilde{a})\} \quad (3.1)$$

where s represents the current state, a represents the current action, $\tilde{s}$ represents the next state, $\tilde{a}$ represents the next action. γ is the greed factor to determine the importance of future rewards,$0 < \gamma < 1$. Q represents the maximum benefit that can be gained by adopting behavior a at the current state of s, R is the immediate gain, and future gains depend on the actions of the next stage.

Like the Q-learning algorithm, the same value-based algorithm, the DQN algorithm has a wider range of uses than it does. Because in the Q-learning algorithm, the Q table is used to store the Q value for each state action, but once the dimension becomes large, the Q table is used very large. So you can replace the Q value in the Q table with a function, so you can combine deep learning with intensive learning, which is the DQN algorithm [10].

Compared to the Q-learning algorithm,DQN uses two networks of the same structure to get the Q value. In the Q-learning algorithm, there is a problem with updating the parameters of the neural network using data in a certain order, so the DQN algorithm uses empirical playback, i.e. using a Memory to store the experienced data, and each time the parameters are updated, a portion of the data is extracted from Memory for updates to break the correlation between the data [11].

The overall flow of the algorithm is as follows:

(1) First initialize a Memory D with an initial capacity of N.

(2) Initialize a Q neural network and randomly initialize the weights.

(3) Loops through multiple batches

(4) Initialize a state s,loop events, repeat steps 5 to10

(5) Select an action ε a with probability, or choose$a = \arg\max Q(s,a')$

(6)Perform action a,observe reward reward and new status s', save <s, a, reward, s'> in Memory D

(7) Randomly select a minibatch's <s, a, reward,s'> from Memory D

(8) Determine whether the target is terminated, using the following formula:

$$y_j = \begin{cases} r_j, & \text{end} \\ r_j + \gamma \max_{a'} Q(s_{j+1}, a'; \theta^-), & \text{no end} \end{cases} \quad (3.2)$$

(9) updates the network parameters $(y_j - Q(s_j, a_j; \theta))^2$as loss.

(10) Update the parameters of another network every C step, i.e. synchronize the two networks.

Through the update iteration of the two networks, the parameters are updated so that the same behavior can be obtained in a similar state, and then the strategy of maximum benefit can be planned through the Q-learning method. However, the DQN algorithm has its downsides, so it is optimized by the Double DQN algorithm, the11.

In the DQN algorithm, the formula for calculating the target value of Q is:

$$y_j = r_j + \gamma \max_{a'} Q(s_{j+1}, a'; \bar{\theta}) \tag{3.3}$$

In the Double DQN algorithm, we calculate the formula for calculating the Q target value:

$$y_j = r_j + \gamma Q(s_{j+1}, \operatorname{argmax}_a Q(s_{j+1}, a; \theta); \bar{\theta}) \tag{3.4}$$

These two formulas mean that in the DQN algorithm, we calculate the maximum value by using a validation $Q(s,a)$ network and then a target network. $Q(s',a')$ In Double DQN, when calculating, you enter $Q(s',a')$it s' into the authentication network to get, $Q(s',a')$then select the one, $a''=\operatorname{argmax}Q(s',a')$then enter it into the s' target network, $Q(s',a'')$and use it $Q(s',a'')$as a label for updates. That is, in DQN we select the action and the a' calculation uses the same $Q(s',a')$network parameter, which can lead to the selection of an overestimation, resulting in an overly optimistic estimate of the value. To avoid this, the selection behavior and measurement behavior are coupled in Double DQN, and the target network is obtained by using the behavior that verifies the state s' of the a'' network selection. $Q(s',a'')$

Also optimizing the DQN algorithm is the Dueling DQN algorithm. Because many times,the value of Q has little to do with the action, and sometimes taking any action in one state has no effect on the result. For example, left and right in a game does not change the final result of the game. So in the Dueling DQN algorithm, the Q value is divided into the state V value and the action A value, and a Q value function is determined by the state function V and the advantage function A. That is:

$$Q(s,a;\theta,\alpha,\beta) = V(s;\theta,\alpha) + A(s,a;\theta,\beta) \tag{3.5}$$

The state function V is only related to the state, and the advantage function A and the action are also state-related. At the same time, the state function S V and A have their own parameters α and β. In fact, only the structural part of the Q network needs to be modified in the code section so that the Q value in Dueling DQN is the V-value.

3. Analysis of the Results of the Intensive Learning Exchange Rate Decision

3.1. Strengthen the Construction of the Learning Model

Divide the exchange rate data into training and test sets in a 3:1 scale, and then start using the enhanced learning model, the most basic DQN model is built as follows:

(1) Create an environment class that conforms to exchange rate changes, which needs to be initialized, and the data that needs to be entered in the initialization function (the closing price of the exchange rate selected in this paper) is required, and the historical data days history_t(default 90 days). At the same time, the environment class needs to have a reset function to reset the environment. The reset function is primarily used to initialize parameters such as total closing profit,which records total returns so far, positions for the list of purchases,which record the exchange rate at which it has been bought and not sold, and the current yield position_value, indicating how much gain will be earned in a single sell. All of these parameters are initialized to 0, i.e. the environment is reset. At the same time, the environment class also needs a running function, which is mainly used to get a state value in the environment, what action can be performed to gain what benefits, that is, for the environment, perform an action every day, and then return the proceeds to save.

(2) The model structure used here is 2 layers linear layer and 2 relu activation functions delineated, and finally 3 values are obtained through a layer of linear functions,3 values correspond to the results of 3 behaviors in one state.

(3) Define the parameters needed to initialize each model training process. The main parameters used in the enhanced learningmodel a reepoch_num, step_max,memory_size, batch_size, epsilon, epsilon_decreas,epsilon_min,start-reduce-epsilon, update_q_freq gamma,show_log_freq。Where epoch_num represents how many rounds of loop;step_max represents the maximum number of steps, i.e. how many steps can be taken in a round of epoch; memory_size represents the size of the memory pool;batch_size represents the size of a batch of data; and epsilon represents the initial value of epsilon, which makes the learning process of intensive learning gradually decrease from broad to deep epsilon,which reduces randomness of behavior, changes the way you learn, and reinforces learning to learn more deeply than more broadly;epsilon_decrease represents how much epsilon is reduced at a time; epsilon_min represents the minimum value of epsilon; starts. Reduce epsilon represents the first few steps to start on epsilon To reduce;train_freq represents the frequency with which training is updated once;update_q_freq represents the frequency of synchronizing another network;gamma represents the discount of reward, which indicates how much of the current reward represents future results;show_log_freq represents how many rounds of epoch output the result once.

(4) After defining the parameters, start training the model. Construct 2 models of the same structure Q_train and Q_eval,cycle through multiple rounds, in each round, initialize the reset environment, take a step in the current state, let the Q_eval network generate all possible behavior values, and select the behavior that maximizes the benefits, and then save the corresponding observations, namely the new state, rewards, and corresponding behaviors, to memory memory, disrupt memory, and take batch_size The data in memory, the loss loss is calculated by formula, the parameters are optimized by the RMSProp optimizer, and then the next step is taken. Synchronize parameters in 2 networks at the same time every update_q_freq steps. Until the iteration to the last round, the training is complete.

The difference between the Double DQN model writing steps and the above is in the calculation of the Q value step, we can modify the corresponding source code according to the formula, the rest of the place remains unchanged.

In the Dueling DQN model, the main difference is the model structure, in Dueling DQN, the model structure used in this paper is to modify a 3-tier linear layer in the DQN to 2 3-tier linear layers, calculating V and A values, respectively, and finally adding up 2 results to return Q value, which remains the same as the DQN model elsewhere.

3.2. Model Results are Compared and Analyzed

The results of several different currency exchange data models are as follows:

As shown in Figure 3.1 (left) below, the results of training on the DQN model using RMB/USD exchange rate data from 1990 to 2018 show that the total benefit of the training set is 9 and the total benefit of the test set is 9. As shown in Figure 3.1 (right) below, the results of training on the Double DQN model using RMB/USD exchange rate data from 1990 to 2018 show that the total benefit of the training set is 0 and the total benefit of the test set is 69.

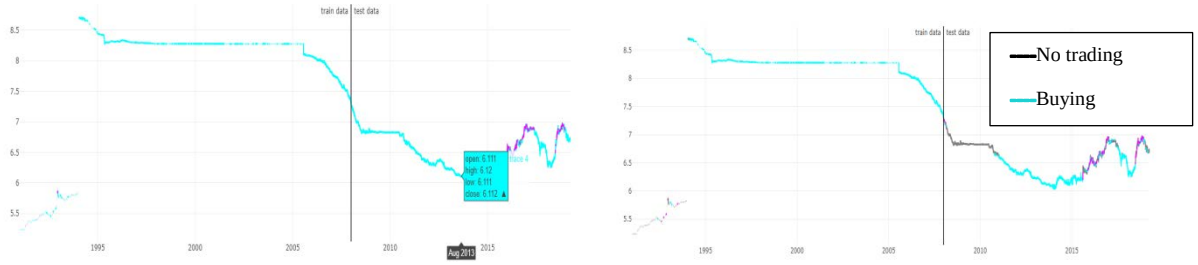

Figure 3.1: RMB-USD exchange rate data :left is DQN results,right is DOUBLE DQN results.

As shown in Figure 3.2 below, the results of training on the Dueling DQN model using RMB/USD exchange rate data from 1990 to 2018 show that the total benefit of the training set is 3and the total benefit of the test set is 65.

Figure 3.2: RMB-USD exchange rate data Dueling DQN results

As shown in Figure 3.3 (left), the results of training on the DQN model using the People's Currency to Japanese Currency Exchange Rate data from 2003 to 2018 show that the total benefit of the training set is 19 and the total benefit of the test set is 26. As shown in Figure 3.3 (right), the results of training on the Double DQN model using RMB/JDR data from 2003 to 2018 are 21 for the training set and 116 for the test set.

Figure 3.3: RMB/Japanese yen exchange rate data :left is DQN results,right is Double DQN

As shown in Figure 3.4 below, the results of training on the Dueling DQN model using RMB/JR exchange rate data from 2003 to 2018 show that the total benefit of the training set is 70 and the total benefit of the test set is 25.

Figure 3.4: RMB/JD exchange rate data Dueling DQN results

As shown in Figure 3.5 below, the results of training on the DQN model using the Canadian currency exchange rate data from 2000 to 2018 show that the total benefit of the training set is 21 and the total benefit of the test set is -4.

Figure 3.5: Canadian exchange rate data DQN results

As shown in Figure 3.6 below, the results of training on the Double DQN model using Canadian currency exchange rate data from 2000 to 2018 show that the total benefit of the training set is 10 and the total benefit of the test set is 10.

Figure 3.6: Double DQN Results for Canadian Currency Exchange Rate Data

As shown in Figure 3.7 below, the results of training on the Dueling DQN model using the Canadian currency exchange rate data from 2000 to 2018 are 2 for the training set and 2 for the test set.

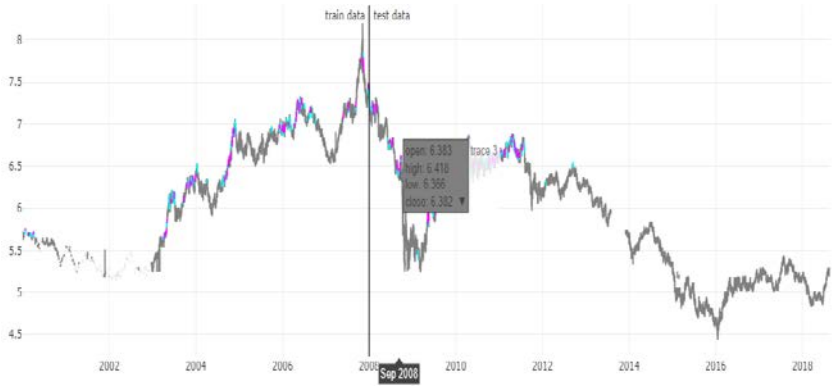

Figure 3.7: Dueling DQN Results for Canada vs. RMB Exchange Rate Data

As shown in Figure 3.8 below, the results of training on the DQN model using offshore RMB/JN exchange rate data from 2010 to 2018 are 44 for the training set and -21 for the test set.

Figure3.8. Offshore RMB/Japanese currency exchange rate data DQN results

Figure 3.9 shows the results of training on the Double DQN model using offshore RMB/ Japanese currency exchange rate data from 2010 to 2018, with a total training set return of 23 and a test set total return of 3.

Figure 3.9: Double DQN Results for Offshore RMB/Japanese Dollar Exchange Rate Data

Figure 3.10 shows the results of training on the Dueling DQN model using offshore RMB/ Japanese exchange rate data from 2010 to 2018, with a total training set return of 13 and a test set total gain of 2.

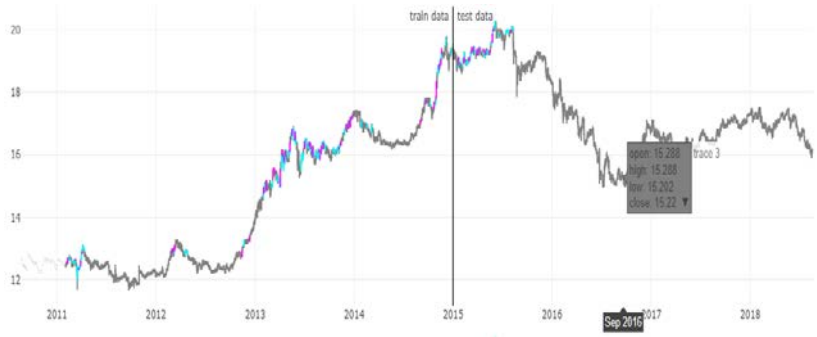

Figure 3.10: Offshore RMB/Japanese currency exchange rate data Dueling DQN results

As shown in Figure 3.11 below, the results of training on the DQN model using euro-to-offshore RMB exchange rate data data from 2010 to 2018show that the total benefit of the training set is 20 and the total benefit of the test set is 12.

Figure 3.11. EUR/OFFSHORE RMB exchange rate dataDQNresults

As shown in Figure 3.12 below, the results of training on the Double DQN model using euro-to-offshore RMB exchange rate data from 2010 to 2018 are 8 for the training set and 8 for the test set.

Figure 3.12: Double DQN results for EUR/OFFSHORE RMB exchange rate data

As shown in Figure 3.13 below, the results of training on the Dueling DQN model using euro-to-offshore RMB exchange rate data from 2010 to 2018 are 10 for the training set and 6for the test set.

Figure 3.13. EUR/OFFSHORE RMB exchange rate data Dueling DQN results

By comparing the performance of the enhanced learning model in many different exchange rate data, test sets and training sets can basically achieve cumulative gains, poor performance on DQN, the reason may be that there is an overestimation problem, the so-called DQN overestimation, that is, DQN $\max_{a'} Q(s_{j+1}, a'; \theta^{-})$ in the calculation, for the Q value is obtained by using the max operation, and then all batch Q values are averaged and then updated network, and normal network operation is the first to all batch Q The value is taken evenly and the maximum value is taken to update the network because $E(\max(X_1, X_2)) \geq \max(E(X_1), E(X_2))$, resulting in an overestimation problem. Exchange rate data therefore performs better on Double DQN and Dueling DQN, which reinforce the learning model. At the same time, in order to verify the applicability of this paper's enhanced learning model, this paper uses Shanghai gold data from 2002 to 2018 as the data of the enhanced model for training, and the results of the model are as follows:

Train on the Double DQN model using gold data from 2002 to 2018 with a total training set benefit of 60and a test set total benefit of 9, as shown in Figure 3.14

Figure 3.14: Gold Data Double DQN Results

Train on the Dueling DQN model using gold data from 2002 to 2018 with a total training set benefit of 115 and a test set total benefit of 76, as shown in Figure 3.15

Figure 3.15: Gold Data Dueling DQN Results

As can be seen from the results diagram, the enhanced learning model can still plan some reasonable strategies on multiple similar data so that we can achieve the benefits. In the case of data limitations, the use of a well-adjusted parameters of the intensive learning model can be used to some extent as a reference for our future exchange rate buy and sell strategy.

4. Conclusions

This paper makes use of a variety of popular deep learning algorithms, by selecting good key indicators, to build a model. Through past historical exchange rate data, a deep learning model can be established to predict the future trend of exchange rate changes, and to a certain extent, the accuracy and availability of the model can be guaranteed. Use enhanced learning algorithms: DQN, Double DQN,Dueling DQN to fit different data, generate strategies, reap benefits, and confirm the validity of the model with model results. Through the above two models, it is possible for some people to make comprehensive decisions of labor and algorithms by combining the strategies obtained from the intensive learning model with the results provided by the deep learning model in exchange rate-related activities, thus providing effective help.

References

[1] Yu Zhuhao.Three important questions about the RMB exchange rate..Hebei: Hebei Institute of Economics and Social Sciences, 2008. 05

[2] Shen Ling, Zhang Chunxiu.Analysis of the causes and effects of RMB exchange rate appreciation. .Shijiazhuang: Huaxin College, Shijiazhuang School of Economics, 2006

[3] Huang Qi. Based on the small wave analysis and the exchange rate prediction combination model that supports vector machine SVM Hunan: Hunan University, 2009

[4] Ouyang Liang.Exchange rate prediction combination model based on wavelet analysis and neural network Hunan: Hunan University,2008

[5] Ye Deqian,Yang Sakura, Kim Dae-bing. The system design of intensive learning algorithms based on neural network integration Hebei: Yan shan University,2006

[6]Hui Xiaofeng,Liu Hongsheng, Hu Wei, He Danqing. RMB exchange rate forecast based on the time series GARCH model Harbin University of Technology,2003

[7]Alex Graves, Abdel-rahman Mohamed, Geoffrey E. Hinton. Speech recognition with deep recurrent neural networks[R]. In Proc: ICASSP, 2013.

[8]Guo Xian,Fang Yongquan. In-depth intensive learning: Getting started with the principles Electronic Industry Press,2018.01

[9]Matthew Hausknecht, Risto Miikkulainen, Peter Stone. A neuro-evolution approach to general atari game playing[R]. US:cs.utexas.edu,2013.

[10]David Silver. Deep Reinforcement Learning [R]. US:cs. LG,2015.

[11]Hado van Hasselt, Arthur Guez, David Silver. Deep Reinforcement Learning with Double Q-learning [R]. US:cs LG2015.12.