Shahjalal University of Science and Technology
Department of Computer Science and Engineering

Reinforcement Learning for Anomaly Detection in Financial Time Series

Iqramul Islam

Reg. No.: 2017331054

4^{th} year, 1^{st} Semester

Rafat Ashraf Joy

Reg. No.: 2017331074

4^{th} year, 1^{st} Semester

Department of Computer Science and Engineering

Supervisor

Mohammad Shahidur Rahman

Professor

Department of Computer Science and Engineering

17th September, 2022

Reinforcement Learning for Anomaly Detection in Financial Time Series

A Thesis submitted to the Department of Computer Science and Engineering, Shahjalal University of Science and Technology, in partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering.

By

Iqramul Islam
Reg. No.: 2017331054
4^{th} year, 1^{st} Semester

Rafat Ashraf Joy
Reg. No.: 2017331074
4^{th} year, 1^{st} Semester

Department of Computer Science and Engineering

Supervisor

Mohammad Shahidur Rahman
Professor
Department of Computer Science and Engineering

17th September, 2022

Recommendation Letter from Thesis/Project Supervisor

The thesis/project entitled *Reinforcement Learning for Anomaly Detection in Financial Time Series* submitted by the students

1. Iqramul Islam
2. Rafat Ashraf Joy

is under my supervision. I, hereby, agree that the thesis/project can be submitted for examination.

Signature of the Supervisor:

Name of the Supervisor: Mohammad Shahidur Rahman

Date: 17th September, 2022

Certificate of Acceptance of the Thesis/Project

The thesis/project entitled *Reinforcement Learning for Anomaly Detection in Financial Time Series* submitted by the students

1. Iqramul Islam
2. Rafat Ashraf Joy

on 17th September, 2022, hereby, accepted as the partial fulfillment of the requirements for the award of their Bachelor Degrees.

Head of the Dept.	Chairman, Exam. Committee	Supervisor
Mohammad Abdullah Al Mumin	M. Jahirul Islam	Mohammad Shahidur Rahman
Professor and Head	Professor	Professor
Department of Computer Science and Engineering	Department of Computer Science and Engineering	Department of Computer Science and Engineering

Abstract

In this study, we aim to employ reinforcement learning for the purpose of detecting anomalies in financial time series data. The problem of identifying the data points that behave in an unexpected way compared to the rest of the data is called anomaly/outlier detection. For the domain of finance, anomaly detection is a significant tool as it provides actionable information. Particularly, we focus on finding potential anomalies in S&P 500 stocks data. Anomalies in stock market data indicate market manipulation. Although there are a myriad of conventional supervised and unsupervised anomaly detection methods, individual anomalies within a time series typically have varied profiles that support various anomaly assumptions. We plan to suggest a reinforcement learning-based model selection framework to take advantage of the strengths of various base models.

Keywords: Anomaly Detection, Time Series, Reinforcement Learning

Acknowledgements

Here goes Acknowledgements...

Contents

Abstract I

Acknowledgements II

Table of Contents III

List of Tables V

List of Figures VI

1 Introduction **1**

1.1 Problem Definition 1

2 Literature Review **3**

3 Data **7**

3.1 Injection of Outliers 7

4 Methodology **12**

4.1 Reinforcement Learning 12

4.2 Deep Q Learning 13

4.3 Markov Decision Process Formulation 14

5 Results and Discussion **15**

5.1 Isolation Forest 15

5.2 Clustering Based Local Outlier 16

5.3 One Class SVM 16

5.4 Local Outlier Factor 17

5.5 PCA 17

5.6 Histogram Based Outlier Detection 18

6 Conclusion and Future Work **19**

References **19**

Appendices **24**

A Title of Appendix A **25**

A.1 source-code to insert artificial anomaly 25

B Title of Appendix B **27**

List of Tables

3.1 Data Description 8

List of Figures

3.1 Daily values of a single company(Exxon Mobil) 8
3.2 Daily values of All Energy Sector Companies 8
3.3 Weekly values of a single company(Exxon Mobil) 9
3.4 Weekly values of All Energy Sector Companies 9
3.5 Real Chevron stock prices 10
3.6 Chevron stock prices After Injection of outliers 11

4.1 MDP formulation 14

5.1 Isolation Forest 15
5.2 Clustering based local outlier detection 16
5.3 One class SVM 16
5.4 Local Outlier Factor 17
5.5 PCA 17
5.6 Histogram Based Outlier Detection 18

Chapter 1

Introduction

There are myriads of applications of time series anomaly detection including fields as diverse as networking, security, finance, healthcare, machinery, astronomy, ecology, data center monitoring[1] and avionics. For example, a diseased condition of the heart of patient can be predicted via the observation of anomalies in ECG time series data. While the ECG of a normal and healthy heart conforms to the expected periodicity, the hearts with arrhythmia exhibits aberrations from the normal [2]. Data gathered from multiple sensors of a The detection outliers from light curves from astronomical data has been a topic of study by physicists [3]. Finding possible loss of confidential data may be facilitated by identifying irregularities in access patterns, as well as the frequency and pattern of employee email and other communications.

1.1 Problem Definition

There are different types of anomaly detection problems for time series data. The problems can be roughly divided into two categories. They are : 1. Identifying an anomalous time series from a set of time series 2. Distinguishing anomalous subsequences from a given time series and 3. Online anomaly detection which is crucial for agile reaction against the ups and downs of market.

The most common anomaly detection techniques are based on distance metrics. However, they miss the notions that are unique to time series data only such as seasonality and cycles. As a result, specialized algorithms are essential for this purpose which are better suited to this type of data.

The three types of time series problems are discussed in detail below.

- **Within a time series:**

    1. A significant difference in a data point's value from earlier data points indicates the occurrence of an event or point anomaly.
    2. Within a given time series, discords are abnormal subsequences that are "maximally distinct from the rest of the sequence [4]." This problem is remarkably common in datasets from anthropology, economics, video, astrophysics, physiology, and electrocardiograms(ECG). For instance, in the case of ECG data, all the individual points may be within the normal interval, however, a certain subsequence may not conform to the regular and periodic form.

       In formal definition, let Z = $\{\zeta(t)|1 \leq t \leq N\}$ where $\zeta(t)$ denotes the value of series Z, at time t. N is the length of the entire series Z. A subsequence can be represented as $\zeta_(p, w) = \{\zeta(p), ..., \zeta(p + w - 1)|1 \leq p \leq N - w + 1; 1 \leq w \leq N\}$.
    3. Often only the rate of change could be anomalous, despite the individual points may be abiding by the expectation. This phenomenon is known as 'rate anomaly'. For example, in the case of identifying credit card fraud, incidents of large number of transactions of small amount may result in an marked anomaly.

- **Between Different time series:**

- **Online Anomaly Detection:** There are some time series in which the nature of the data is altered as the time progresses. As a consequence, applying the model gained from old data is no longer applicable to the newer one. Therefore, it is required to apply the learning algorithm again in order to retrieve a new set of parameters. In an online anomaly detection problem, the anomaly is identified with respect to the most recent behavior of the time series.

Chapter 2

Literature Review

There are many algorithms and strategies for time series anomaly detection. These methods can be divided into five types. They are:

1. **Traditional Methods:** These methods are reliant on distance and density measures and can be implemented by simple and conventional machine learning algorithms. K nearest neighbours(KNN) [5], Isolation forest [6], One Class Support Vector Machine (OCSVM) [7], Local Outlier Factor (LOF) [8]. Although these methods are efficient in terms of time and computational resources, they do not exhibit state-of-the-art performance in real world data.

2. **Supervised Methods:** These methods extract features from the input data and construct a classification model to identify anomalies. Most of the methods in this category are based on deep learning. EGADS is a 'generic framework for time series anomaly detection' [9] developed by Yahoo. One downside of these methods is that they required a large volume of labeled data which is difficult to procure in some areas such as IOT sensors and other big data.

3. **Unsupervised Methods:** These methods do not need labeled data. In addition, they make the assumption that outliers have larger deviation from the normal distribution. Luminol [10] is python library developed by Linkedin. TwitterAD [11] which is a anomaly detection framework written in R, developed by Twitter. It utilizes statistical learning in its methods. DONUT [12] is an unsupervised anomaly detection system which uses variational auto

encoders. Microsoft incorporated SR-CNN [13] in their anomaly detection procedure, which is a widely used technique in computer vision. Siffer et al. used Extreme value theory for the purpose of identifying anomalies in high throughput streaming univariate and unimodal time series [14]. This approach does not make any assumption on the distribution of the data.

4. **Semi-supervised Methods:** These methods use only a portion of labeled data in the training phase. SNARE [15], a method developed by Mcglohon et al. uses label propagation for outlier identification in graph data. REPEN [16] is a method which works by learning representations of ultra high dimensional data. It requires only a handful of labeled anomalies to learn the representative features. Deep-SAD [17] is a semi-supervised technique developed by Ruff et al. deals with the problem from an information theoretic perspective. First, it uses an auto-encoder for model pre-training. A hyperparameter is required to regulate the balance of training data.

Machine learning based approaches learn a model from the labeled training data and classify new data between abnormal and normal classes. Classification and clustering - are the two ways to learn the model. Bayesian Networks [18], Support Vector Machines [19] are common machine learning algorithms to build anomaly detectors.

Until now, no study has been carried out on the prospect of utilizing reinforcement learning for anomaly detection in the financial time series data. However, reinforcement learning is a promising tool for this type of task, testified by the results of previous works. Huang et al. [20] applied value-based deep reinforcement learning coupled with DQN algorithm.

The study carried out by Golmohammadi et al. [21] employs prediction based Contextual anomaly detection for detecting time series anomaly, specifically detecting fraud in the securities market. The datasets used in this study were from different fields of SP 500 constituents. To mimic fraud, the researchers have induced artificial anomalies into the original, under the assumption that the original data was free from market manipulation. The authors devised and carried out a complete series of experiments to test the suggested approach on 5 distinct S&P sectors with daily and weekly frequency. Instead of utilizing a time series' history to forecast the next successive values, the authors use the behavior of comparable time series to predict the predicted values. However, one drawback of this study is that the proposed model yields higher number of false

positives compared to state-of-the art, thus reducing the precision.

Meta AAD [22] is an anomaly detection framework that learns a meta policy for query selection. First, the researchers filters out 'l' transferable features out of 'd' original features. Afterwards, the meta policy is trained with Proximal Policy Optimization (PPO) approach. The authors assessed the efficacy of this framework on 24 datasets. Meta AAD outperforms all of them and this method tends to even better if the number of queries is larger. This method extracts transferable meta features thus optimizing the meta policy on data streams.

The biggest drawback of this method is that it works for tabular data only, accordingly it is not yet fully applicable to detect time series and graph anomalies. In addition, the approach is not fully automated. It requires human(analyst) decision to finally detect the anomalies. The approach itself just selects probable query points that are supposed to be anomalous, so that the analyst's effort is maximized.

RLAD is a semi-supervised, time series anomaly detection algorithm developed by Wu et al. [23]. It combines reinforcement learning with active learning and label propagation to efficiently learn and adapt to anomalies in real-world time series data. The researchers discussed some traditional approaches (like supervised, unsupervised etc) used previously on anomaly detection and their limitations on detecting anomaly. Then they talked about reinforcement learning and active learning briefly which are used in this algorithm.
This study used two dataset Yahoo benchmark and KPI in their experiments. Finally the researchers compared the result with 7 semi-supervised and unsupervised anomaly detection algorithms. The results suggest that, RLAD outperformed all unsupervised and semi-supervised approaches.

PTAD [24] a generic policy based time series anomaly detector. It uses knowledge of deep reinforcement learning and based on A3C algorithm. The researchers discussed various state-of-the-art anomaly detection strategies and algorithms used previously on anomaly detection and their limitations. Then, they proposed their solutions. They have given a brief overview about reinforcement learning and formalized RL for time series anomaly detection. They compared the traits between value based and policy based DRL(Deep reinforcement Learning) superbly. Two dataset are used in the experiment(Yahoo benchmark and Numenta anomaly detection dataset). They used F1 score to compare with other models. Finally the researchers compared the result with 7 well known anomaly detectors. The result suggests that, though PTAD is not the best in every test

time series, it outperforms other time series anomaly techniques averagely. It also indicates that, RL based time series anomaly detectors achieve superior performance than other non-RL methods.

Chapter 3

Data

The financial data used in this study will be S&P constituents data - a reliable benchmark for risk analysis, financial forecasting and fraud detection. This data is publicly available and can be fetched by any stock market API. To simulate anomaly, we will inject outliers in that data. For generation of synthetic outliers, the approach followed by this paper [21] will be followed.

The data has both weekly and daily values of stocks roughly categorized into 5 sectors. These sectors are :

- Energy
- Financials
- Consumer Discretionary
- Information Technology
- Consumer Staples

All of the datasets are extracted from Thompson Reuters. The currency of the stock prices are in USD. It contains 40 years of historical data.

3.1 Injection of Outliers

S&P 500 data is considered to be free from anomalies by some reliable studies [25] [26]. A number of reasons pointed out by these studies are as follows:

S&P Sector	Number of time series	data points (weekly)	data points (daily)
Energy	44	63,000+	315,000+
Financials	83	117,000+	587,000+
Consumer Discretionary	85	111,000+	558,000+
Information Technology	66	80,000+	395,000+
Consumer Staples	40	64,000+	323,000+

Table 3.1: Data Description

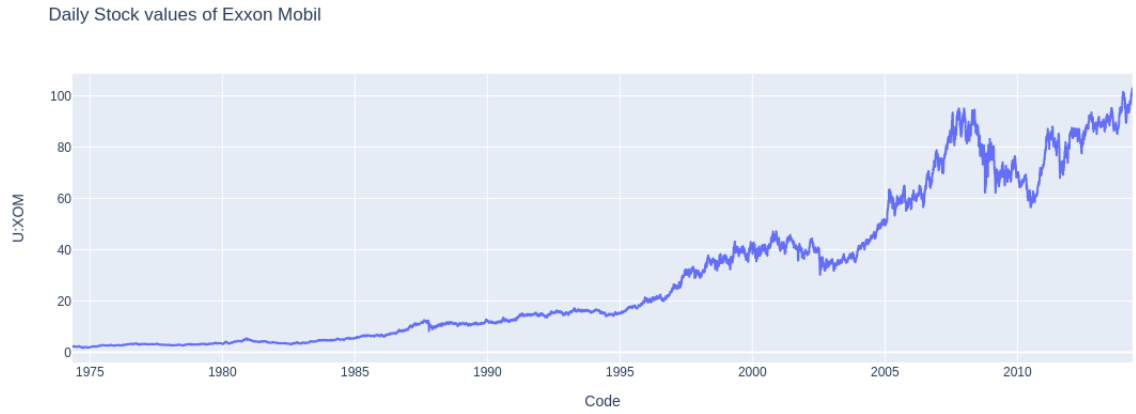

Figure 3.1: Daily values of a single company(Exxon Mobil)

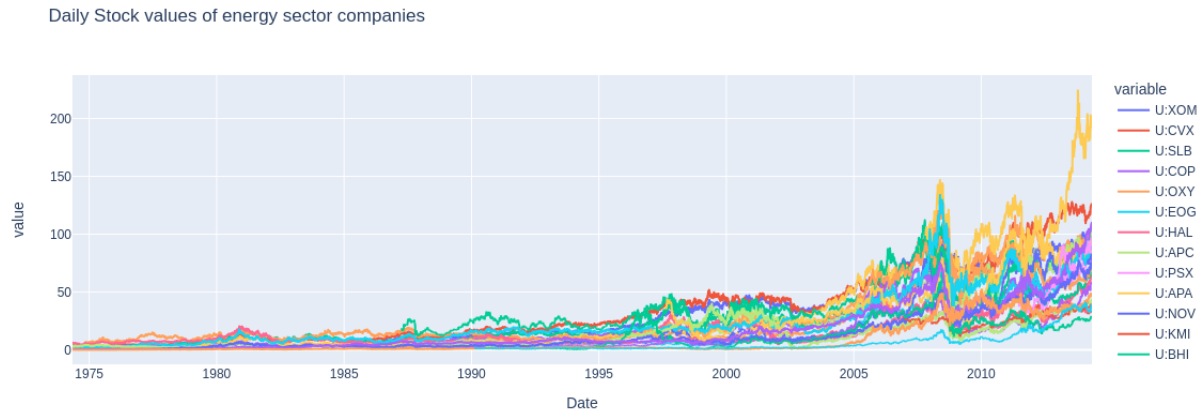

Figure 3.2: Daily values of All Energy Sector Companies

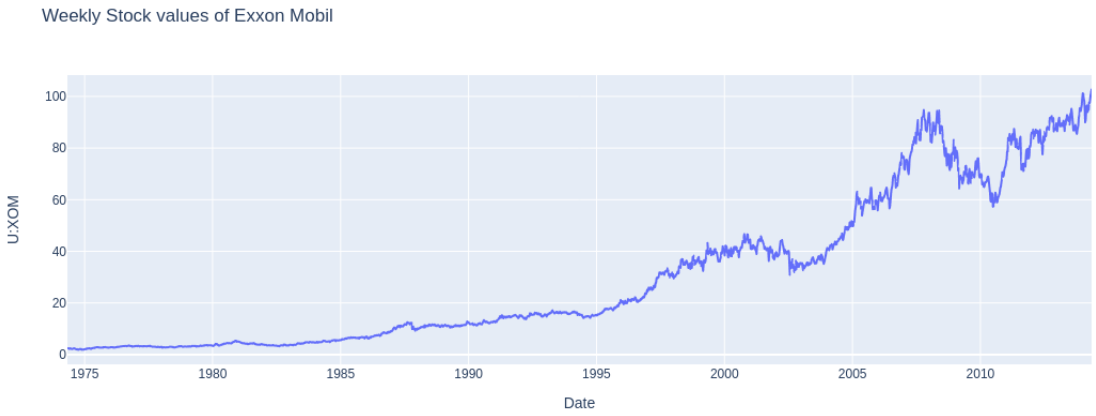

Figure 3.3: Weekly values of a single company(Exxon Mobil)

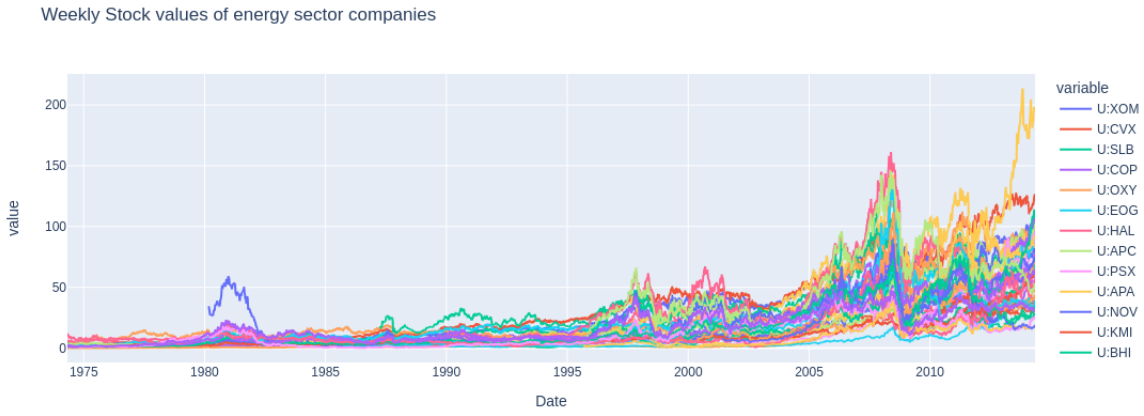

Figure 3.4: Weekly values of All Energy Sector Companies

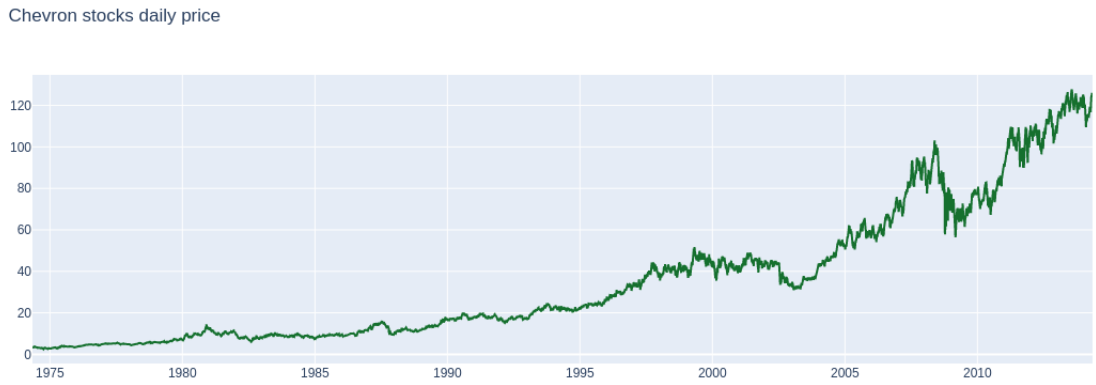

Figure 3.5: Real Chevron stock prices

- These stocks belong to the largest corporations in the United States. Hence, it is improbable that they would be manipulated by a small group of people.
- There are sellers and buyers of these stocks all year round. As a consequence, they are highly liquid, thus preventing a special group to take control over them.
- Both market analysts and regulatory bodies keep a close eye on it and strictly regulate it.

To generate artificial outliers, we followed a redesigned version of Tukey's method[27] for sub-sequences.

First, we randomly chose some data points to be altered(or anomalized). In those randomly chosen indices, we inserted values:

$$\varphi(i) = \mu + 3\sigma \tag{3.1}$$

where, $\varphi(i)$ is the inserted value at i^{th} index. μ and σ are respectively the mean and standard deviation of 160 observations both before and after i^{th} index. The source code to inject synthetic anomaly into the data has been given in Appendix A.

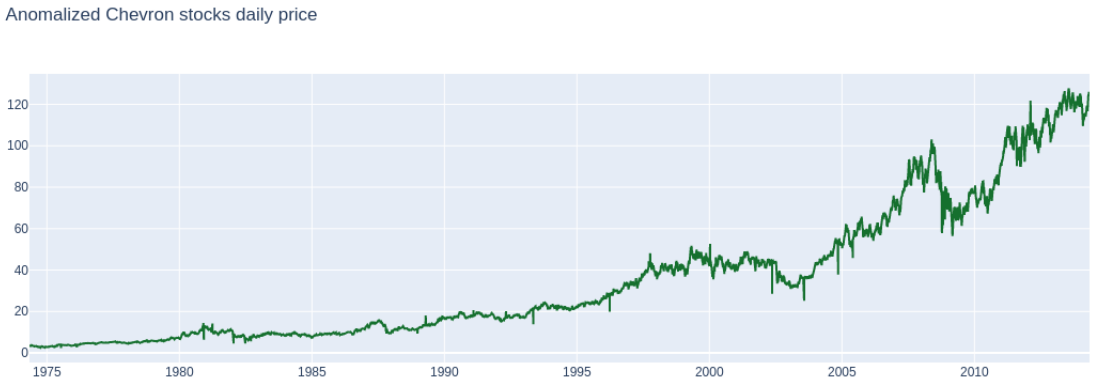

Figure 3.6: Chevron stock prices After Injection of outliers

Chapter 4

Methodology

4.1 Reinforcement Learning

Our anomaly detection issue is viewed as a Markov decision process (MDP), which can be stated as a tuple of $< S, A, P_a, R_a >$. S stands for the collection of environmental states. A is a series of actions taken by RL agent. P_a (s, s′) denotes the likelihood that action a will be taken at state s and result in state s'. R_a (s, s′) is called the immediate reward that an agent receives for doing an action a. y ϵ [0, 1] is called the discount factor. The value function is expressed as $V_\pi(s) = E[\sum_t^T]$. Beginning with state s, it reflects the anticipated reward return. The agent in MDPÂ attemptsÂ to learn a control policy $\pi : S \rightarrow A$ that optimizes the total future reward $\sum_t^T \gamma^t R_t$.

Model-based and model-free solutions to Markov decision process problems are frequently suggested solutions. Model-based approaches create a model of the environment. In order to interact with the environment, the agent must become familiar with it. Dynamic programming is the model-based algorithm that is used the most. It is a technique for resolving big issues by splitting them up into smaller issues. Model-free approaches just investigate the world, envisage the next scenario, and take the best actions he believes are appropriate. They do not attempt to fully grasp the environment. We exclusively discuss model-free algorithms in this study because they have more widespread applications and usages. Value-based algorithms and policy-based algorithms are the two groups of model-free algorithms that are suggested.

4.2 Deep Q Learning

A well-known value-based algorithm is Q learning. Agents are taught the action-value function Q(s, a) and can estimate how well an action will turn out for a given state. What determines the target value is:

$$target = R(s, a, s') + \gamma \, maxQ_k \, (s', a') \tag{4.1}$$

And the following updates the Q function:

$$Q_{k+1}(s, a) \leftarrow (1 - \alpha)Q_k(s, a) + \alpha target(s') \tag{4.2}$$

Unfortunately, when the action value function is approximated by a nonlinear function, such as neural networks, Q learning is unstable or even divergent. [28] Consequently, the Deep Q network method (DQN) that combines deep neural network and reinforcement learning that is proposed by $Deepmind$ is accepted in order to manage more difficult issues. It maintains the action value function training into a deep neural network for approximation. Experience replay [29] and target network are the crucial components of DQN.

The transitions of the tuple $< s, a, r, s' >$ are stored in experience replay. Where the state is defined by s, action is represented by a, and reward is defined by r and s' represents the next state after taking action.

During each iteration, the agent is trained by randomly selecting a small batch from it. It minimizes sample correlation and increases data effectiveness.

Target networks are made to handle problems with non-stationary target values. We construct a neural network to learn values for Q, but because the target values are so erratic, training is difficult to converge. In order to speed up the training process, another neural network with the exact same structure as the target network is created to fix target values temporarily.

4.3 Markov Decision Process Formulation

We formulate the selection of best anomaly detector as a markov decision process (MDP) as illustrated in Figure 4.1. The discount factor γ is set to 1.

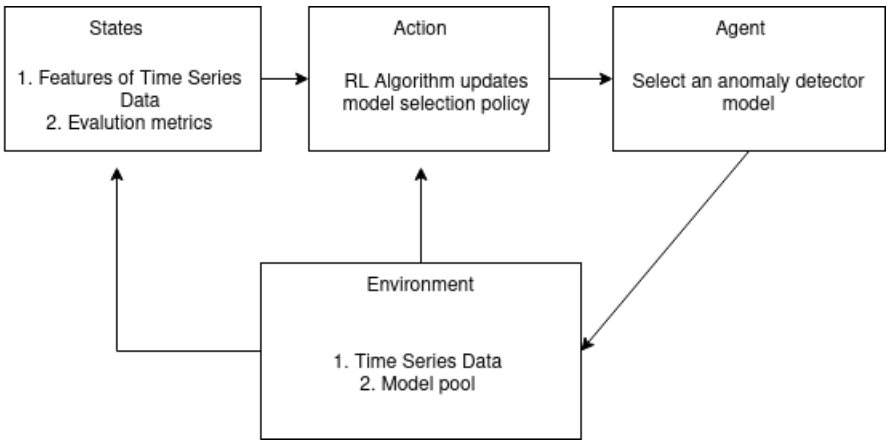

Figure 4.1: MDP formulation

- **States:** Every individual anomaly detector is considered as a state.
- **Action:** Among a list of candidate anomaly detectors, an action refers to selecting one of them.
- **Reward:** Based on a comparison between the predicted label and the actual label, the reward function is constructed.

$$Reward = \begin{cases} r_1 & \text{for True Positive } (TP) \\ r_2 & \text{for True Negative } (TN) \\ -r_3 & \text{for False Positive } (FP) \\ -r_4 & \text{for False Negative } (FN) \end{cases}$$

where, $r_1, r_2, r_3, r_4 > 0$

Chapter 5

Results and Discussion

5.1 Isolation Forest

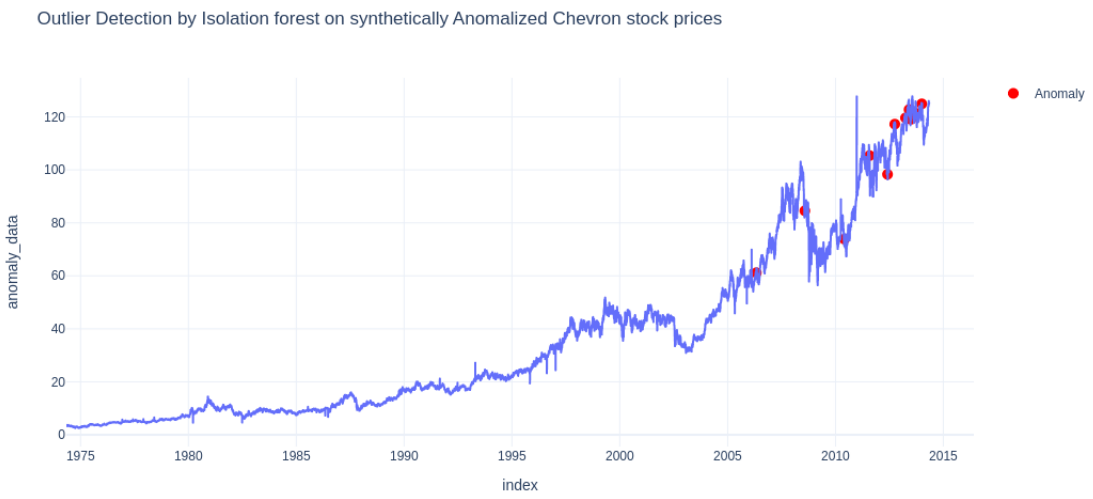

Figure 5.1: Isolation Forest

5.2 Clustering Based Local Outlier

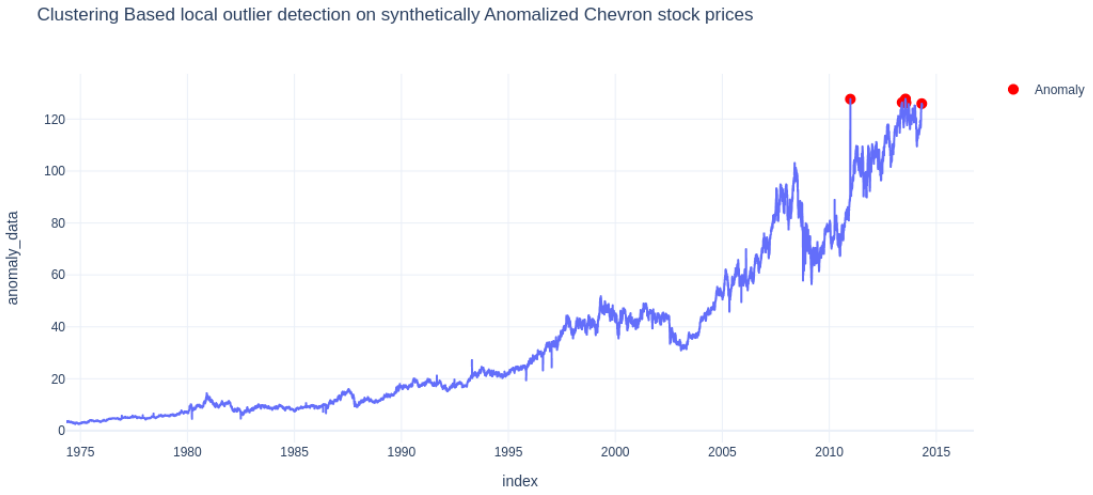

Figure 5.2: Clustering based local outlier detection

5.3 One Class SVM

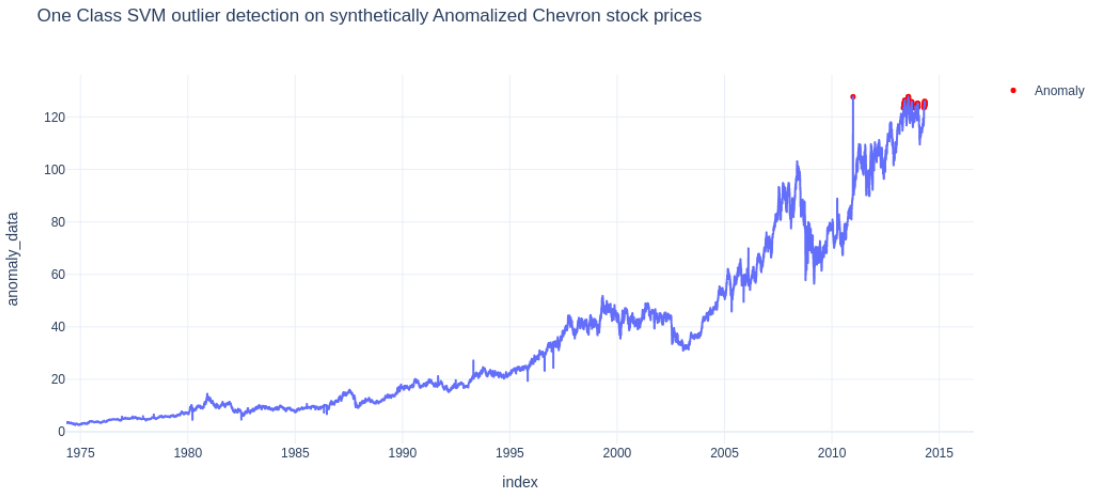

Figure 5.3: One class SVM

5.4 Local Outlier Factor

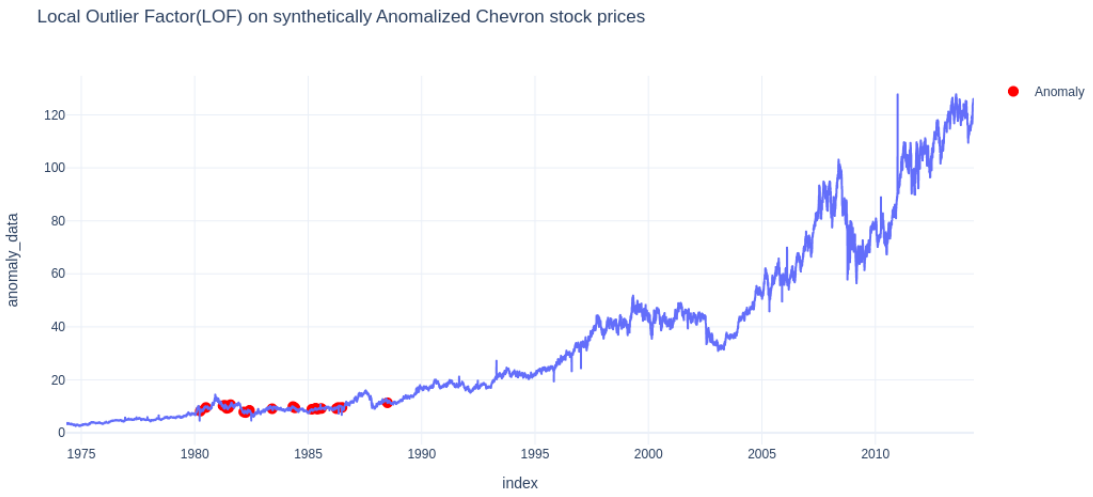

Figure 5.4: Local Outlier Factor

5.5 PCA

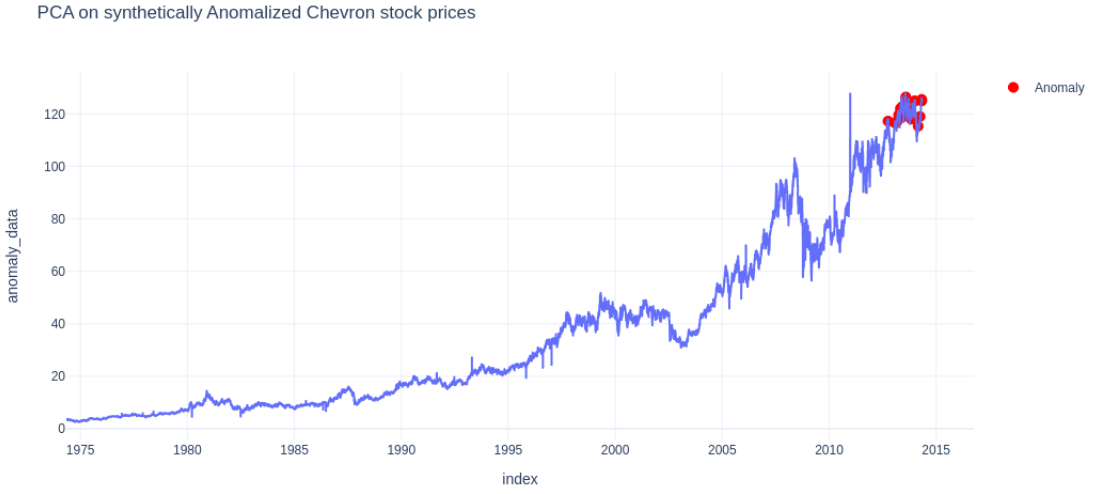

Figure 5.5: PCA

5.6 Histogram Based Outlier Detection

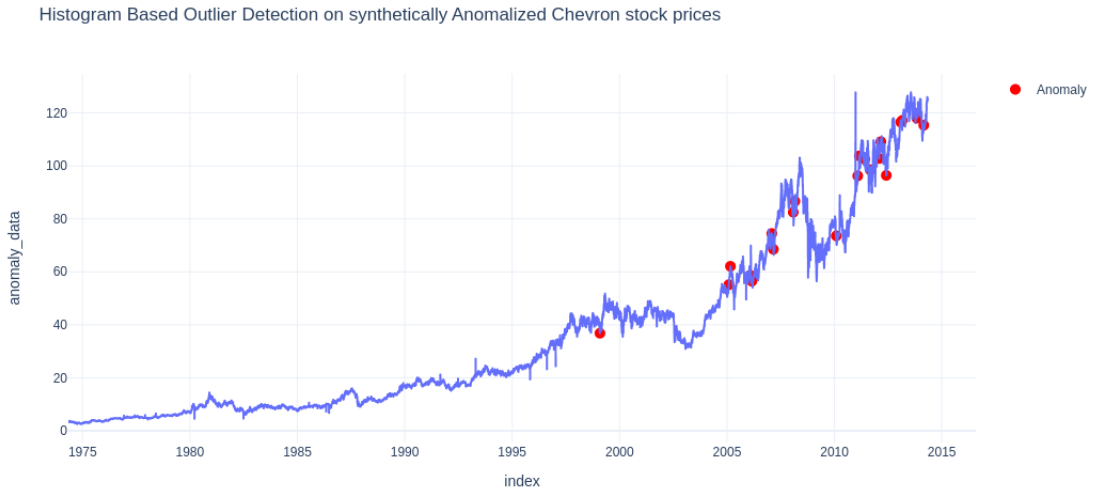

Figure 5.6: Histogram Based Outlier Detection

From the plots shown above, it can be noticed that most of the algorithms have jumbled the outliers either in the beginning of the time series(LOF) or in the last portion(PCA, One class SVM, Isolation forest). Histogram outlier detection algorithm performs the best as it captures the anomalies scattered uniformly all over the entire timespan.

Chapter 6

Conclusion and Future Work

So far, we have applied different conventional anomaly detection models on the synthetically anomalized dataset. It is evident that not all algorithms are suitable for detecting all types of anomalies. While a certain algorithm is better at identifying at a particular portion of the data, other algorithm is better at another section. In future, reinforcement learning will be used to decide which model to choose among a pool of 10 to 12 models.

References

[1] H. Ren, B. Xu, Y. Wang, C. Yi, C. Huang, X. Kou, T. Xing, M. Yang, J. Tong, and Q. Zhang, "Time-Series Anomaly Detection Service at Microsoft," *arXiv*, Jun. 2019.

[2] M. Thill, W. Konen, H. Wang, and T. Bäck, "Temporal convolutional autoencoder for unsupervised anomaly detection in time series," *Applied Soft Computing*, vol. 112, p. 107751, 2021.

[3] P. Protopapas, J. Giammarco, L. Faccioli, M. Struble, R. Dave, and C. Alcock, "Finding outlier light curves in catalogues of periodic variable stars," *Monthly Notices of the Royal Astronomical Society*, vol. 369, no. 2, pp. 677–696, 2006.

[4] E. Keogh, J. Lin, S.-H. Lee, and H. V. Herle, "Finding the most unusual time series subsequence: algorithms and applications," *Knowledge and Information Systems*, vol. 11, no. 1, pp. 1–27, 2007.

[5] F. Angiulli and C. Pizzuti, "Fast outlier detection in high dimensional spaces," in *European conference on principles of data mining and knowledge discovery*. Springer, 2002, pp. 15–27.

[6] L. F. Tony, T. K. Ming, and Z. Zhi-Hua, "Isolation-based anomaly detection," *ACM Transactions on Knowledge Discovery from Data (TKDD)*, vol. 6, no. 1, p. 3, 2012.

[7] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural computation*, vol. 13, no. 7, pp. 1443–1471, 2001.

[8] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "Lof: identifying density-based local outliers," in *Proceedings of the 2000 ACM SIGMOD international conference on Management of data*, 2000, pp. 93–104.

[9] N. Laptev, S. Amizadeh, and I. Flint, "Generic and scalable framework for automated time-series anomaly detection," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 1939–1947.

[10] linkedin, "luminol," Sep. 2022, [Online; accessed 16. Sep. 2022]. [Online]. Available: https://github.com/linkedin/luminol

[11] O. Vallis, J. Hochenbaum, and A. Kejariwal, "A novel technique for {Long-Term} anomaly detection in the cloud," in *6th USENIX workshop on hot topics in cloud computing (HotCloud 14)*, 2014.

[12] H. Xu, W. Chen, N. Zhao, Z. Li, J. Bu, Z. Li, Y. Liu, Y. Zhao, D. Pei, Y. Feng *et al.*, "Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 187–196.

[13] H. Ren, B. Xu, Y. Wang, C. Yi, C. Huang, X. Kou, T. Xing, M. Yang, J. Tong, and Q. Zhang, "Time-series anomaly detection service at microsoft," in *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, 2019, pp. 3009–3017.

[14] A. Siffer, P.-A. Fouque, A. Termier, and C. Largouet, "Anomaly detection in streams with extreme value theory," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2017, pp. 1067–1075.

[15] M. McGlohon, S. Bay, M. G. Anderle, D. M. Steier, and C. Faloutsos, "Snare: a link analytic system for graph labeling and risk detection," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2009, pp. 1265–1274.

[16] G. Pang, L. Cao, L. Chen, and H. Liu, "Learning representations of ultrahigh-dimensional data for random distance-based outlier detection," in *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, 2018, pp. 2041–2050.

[17] L. Ruff, R. A. Vandermeulen, N. Görnitz, A. Binder, E. Müller, K.-R. Müller, and M. Kloft, "Deep semi-supervised anomaly detection," *arXiv preprint arXiv:1906.02694*, 2019.

[18] K. Das and J. Schneider, "Detecting anomalous records in categorical datasets," in *KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*. New York, NY, USA: Association for Computing Machinery, Aug. 2007, pp. 220–229.

[19] J. Ma and S. Perkins, "Time-series novelty detection using one-class support vector machines," in *Proceedings of the International Joint Conference on Neural Networks, 2003.* IEEE, Jul. 2003, vol. 3, pp. 1741–1745vol.3.

[20] C. Huang, Y. Wu, Y. Zuo, K. Pei, and G. Min, "Towards Experienced Anomaly Detector Through Reinforcement Learning," *AAAI*, vol. 32, no. 1, Apr. 2018.

[21] K. Golmohammadi and O. R. Zaiane, "Time series contextual anomaly detection for detecting market manipulation in stock market," in *2015 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, Oct. 2015, pp. 1–10.

[22] D. Zha, K. Lai, M. Wan, and X. Hu, "Meta-aad: Active anomaly detection with deep reinforcement learning," in *20th IEEE International Conference on Data Mining, ICDM 2020, Sorrento, Italy, November 17-20, 2020*, C. Plant, H. Wang, A. Cuzzocrea, C. Zaniolo, and X. Wu, Eds. IEEE, 2020, pp. 771–780. [Online]. Available: https://doi.org/10.1109/ICDM50108.2020.00086

[23] T. Wu and J. Ortiz, "Rlad: Time series anomaly detection through reinforcement learning and active learning," *arXiv preprint arXiv:2104.00543*, 2021.

[24] M. Yu and S. Sun, "Policy-based reinforcement learning for time series anomaly detection," *Engineering Applications of Artificial Intelligence*, vol. 95, p. 103919, 2020.

[25] D. Enke and S. Thawornwong, "The use of data mining and neural networks for forecasting stock market returns," *Expert Systems with applications*, vol. 29, no. 4, pp. 927–940, 2005.

[26] H. Öğüt, M. M. Doğanay, and R. Aktaş, "Detecting stock-price manipulation in an emerging market: The case of turkey," *Expert Systems with Applications*, vol. 36, no. 9, pp. 11 944–11 949, 2009.

[27] "Exploratory Data Analysis," in *The Concise Encyclopedia of Statistics*. New York, NY, USA: Springer, New York, NY, 2008, pp. 192–194.

[28] Y. Li, "Deep reinforcement learning: An overview," *arXiv preprint arXiv:1701.07274*, 2017.

[29] L.-J. Lin, "Self-improving reactive agents based on reinforcement learning, planning and teaching," *Machine learning*, vol. 8, no. 3, pp. 293–321, 1992.

Appendices

Appendix A

Title of Appendix A

A.1 source-code to insert artificial anomaly

```
def anomaly_injector(
        givendf,anomaly_frac=0.02
    ):
    '''
    Returns
    1)the outlier injected dataframe
    2)the indices where outliers have been inserted
    '''

    new_arr = givendf['values'].to_numpy().copy()
    arr_min = new_arr.min()
    arr_max = new_arr.max()


    loc=np.mean(new_arr) # mean of distribution
    gamma=sum((val-loc)**3 for val in new_arr)/len(new_arr)  # gamma=(x_i-loc)^
                                            3/n
    nof_anomalies = int(len(new_arr) * anomaly_frac)
    idx_list = np.random.choice(a=len(new_arr), size=nof_anomalies, replace=
                                            False)
```

```
    for idx in idx_list:
        if np.round(np.random.random()):
            meanValue=np.mean(new_arr[idx-15:idx+15])
            sigma=np.std(new_arr[idx-15:idx+15]) # standard deviation
            new_arr[idx] = meanValue + 3*sigma

        else:
            sigma=np.std(new_arr[idx-160:idx+161]) # standard deviation
            meanValue=np.mean(new_arr[idx-160:idx+161])
            new_arr[idx] = meanValue - 3*sigma



    df = pd.DataFrame(
        {"time": givendf.index, "anomaly_data": new_arr}
    )
    return df,idx_list
```

Appendix B

Title of Appendix B

Appendix B goes here.....