OPEN ACCESS

International Journal of Advances in Artificial Intelligence and Machine Learning Vol 1 No 1 (2024) https://doi.org/10.58723/ijaaiml.v1i1.287

Research Article

Real-Time Outlier Detection in Fast-Moving Data Streams

Received: Sept 14, 2024 | Revised: Nov 04, 2024 | Accepted: Nov 05, 2024 | Publish: Nov 06, 2024

Eka Puji Agustini*, Mohd Zaki Zakaria

Abstract:
Anomaly detection is a critical task in various fields such as finance, healthcare, network monitoring, and sensor data analysis, where identifying unusual patterns or outliers in data streams is essential for timely decision-making. Two commonly used techniques for anomaly detection are the Moving Average (MA) and Exponential Moving Average (EMA) methods. Despite their widespread use, selecting the appropriate method depends on the nature of the data and the requirements of the system. This paper presents a comparative analysis of MA and EMA for anomaly detection, focusing on critical factors such as speed of detection, stability, precision and recall, false positive rate, and computational efficiency. This research addresses the problem of determining which method, MA or EMA, is better suited for specific types of data, particularly in streaming environments with varying trends and anomalies. The results of our comparison indicate that EMA performs better in dynamic environments where rapid identification of anomalies is critical, such as financial markets or network traffic analysis. It quickly detects sudden deviations but may flag minor fluctuations as false positives due to its sensitivity. MA, on the other hand, is more stable and computationally efficient, with a lower false positive rate, making it more suitable for applications where long-term trend monitoring is required, and stability is prioritized over speed. This research highlights the strengths and weaknesses of both methods, demonstrating that the choice between MA and EMA should be based on the specific needs of the anomaly detection system. For real-time, high-speed environments, EMA offers a more responsive solution, while MA provides better stability and efficiency in long-term monitoring. A hybrid approach combining both methods could offer a more robust solution, adapting to different types of data and detection requirements.

Keywords: Anomaly detection, data streams, outlier detection, real-time data

1. INTRODUCTION

In recent years, the proliferation of real-time data sources ranging from IoT sensors to social media feeds and financial transactions has posed significant challenges for anomaly detection in streaming environments. Anomalies, defined as deviations from expected behavior, often signal critical events such as fraud, network intrusions, or system failures. Detecting these anomalies in streaming data is particularly challenging due to the continuous and high-velocity nature of the data, which necessitates immediate analysis and response. Unlike traditional batch data processing, where algorithms can analyze large, static datasets at leisure, streaming anomaly detection demands algorithms that are not only fast but also adaptable to the evolving nature of the data. These constraints are further complicated by limited computational resources and memory, as well as the real-time nature of anomaly detection systems.

Recent research has highlighted several key challenges in detecting anomalies in streaming data, notably concept drift the phenomenon where the statistical properties of the data change over time and the scarcity of labeled data, which hinders the application of supervised learning techniques. In response, various adaptive, unsupervised algorithms have been developed to address these issues. Among the most commonly used techniques are the Moving Average (MA) and Exponential Moving Average (EMA), both of which are statistical smoothing methods that help detect outliers by identifying deviations from established patterns. Despite their widespread use, these two methods differ in how they weigh historical data, leading to different performance characteristics in detecting anomalies.

Several studies have explored the effectiveness of smoothing techniques like MA and EMA in anomaly detection. For instance, Siffer et al. (2017) demonstrated that smoothing methods are particularly effective when dealing with non-stationary data streams, as they can smooth out noise and focus on detecting genuine anomalies. Similarly, studies by Braei and Wagner (2020) highlighted the importance of using adaptive smoothing techniques in real-time environments where delays in anomaly detection can have significant consequences, particularly in sectors like finance and cybersecurity. However, these studies also noted that different smoothing techniques perform variably depending on the nature of the data and the speed of detection required, which raises the question of which method MA or EMA performs better in specific real-time scenarios.

A critical problem addressed by this study is the trade-off between responsiveness and stability in anomaly detection. The MA technique smooths data by giving equal weight to all past observations within a sliding window, making it less sensitive to sudden changes but more stable over time. This method can delay detection in scenarios where rapid changes in the data stream require immediate action, such as in financial fraud detection or real-time monitoring of industrial systems. On the other hand, the EMA places more weight on recent observations, enabling it to detect rapid changes more quickly. However, this increased sensitivity can lead to a higher rate of false positives, as the algorithm may overreact to short-term fluctuations in the data.

Previous research, such as that by Karasu et al. (2018), has demonstrated that EMA can be particularly effective in applications requiring quick anomaly detection, such as network security. Still, it also highlighted the algorithm's vulnerability to noise. In contrast, studies by Ivanovski et al. (2018) suggest that MA is more suited for scenarios where a smoother, less reactive detection process is beneficial, such as in environmental monitoring. By systematically comparing these two methods using the Numenta Anomaly Benchmark Data, this research seeks to determine which technique offers the best balance between detection speed and accuracy in real-time data streams.

The primary objectives of this research are threefold: First, to evaluate the effectiveness of MA and EMA algorithms in detecting anomalies in streaming data. Second, to analyze their performance using key metrics such as detection accuracy, response time, and false favorable rates within the context of the Numenta Anomaly Benchmark Data (BoltzmannBrain, 2016). Finally, to provide practical insights into the suitability of each algorithm for different types of streaming environments, particularly those where rapid and accurate detection of anomalies is critical.

Motivated by this ongoing debate, the present research seeks to compare the performance of MA and EMA algorithms in detecting anomalies within real-time data streams. The study uses the Numenta Anomaly Benchmark Data, a comprehensive dataset designed explicitly for evaluating anomaly detection algorithms in dynamic, real-world conditions. The Numenta benchmark provides an ideal testing environment for comparing the MA and EMA algorithms, as it incorporates a range of anomaly types, from sharp deviations to more subtle, context-dependent outliers. This dataset has been widely used in previous research, including studies by Ejder & Özel (2024) and Nakano et al. (2017), to assess the robustness and responsiveness of anomaly detection techniques under different streaming conditions.

This study aims to contribute to the broader field of anomaly detection by offering a comprehensive comparative analysis of two widely used algorithms, MA and EMA, under realistic, high-velocity data conditions. The findings of this research will help inform the selection of appropriate detection techniques for real-world streaming applications, ultimately improving the efficacy of systems designed to identify and respond to anomalies in real-time.

2. MATERIAL AND METHOD

In anomaly detection, especially in the context of streaming data, choosing the proper smoothing technique to detect outliers or unusual events is critical. Moving Averages (MA) and Exponential Moving Averages (EMA) are two widely used methods for smoothing time series data and are often employed as part of anomaly detection systems. This article provides a comparative analysis of these two techniques, focusing on their mathematical models, advantages, disadvantages, and their performance in detecting anomalies in streaming data.

Methodology for Anomaly Detection

In anomaly detection, both MA and EMA are used to model expected values for a time series. Anomalies are identified when the actual value of the time series deviates significantly from the smoothed expected value. The threshold for identifying an anomaly is usually determined by calculating a statistical measure such as the standard deviation of the residuals (i.e., the differences between actual values and the moving average).

Moving Average (MA)

The MA is a straightforward method that calculates the average of the last n observations over a sliding window (Hansun, 2013). It is often called a Simple Moving Average (SMA) due to its simplicity. The mathematical formula for a moving average of window size n is:

$$MA(t) = \frac{1}{n}\sum_{i=0}^{n-1} x(t-i) \qquad (1)$$

where:

1. *x(t)* is the value of the time series at time *t*,
2. *n* is the window size,
3. *MA(t)* is the moving average at time *t*.

This is the step-by-step process of moving average algorithms and their advantages and disadvantages.

Moving Average-Based Anomaly Detection

1. Step 1: Compute the MA over a sliding window.
2. Step 2: Calculate the residual *r(t)=x(t)−MA(t)*.
3. Step 3: Flag anomalies when the residual *r(t)* exceeds a predefined threshold, typically a multiple of the standard deviation of the residuals.

The MA offers several advantages, making it a widely used method for data smoothing. One of its primary benefits is simplicity. The MA is easy to compute and interpret, making it accessible to users across different fields. Additionally, the MA is known for its stability, as it provides a smooth and steady representation of the underlying trend in data by averaging values over a fixed window. Another key strength is its non-sensitivity to short-term fluctuations. Since the MA assigns equal weight to all observations within the window, it effectively filters out short-term noise, allowing users to focus on broader trends.

However, the MA does have some limitations. Based on van Rossum (2019), a significant drawback is its lag behind actual data. Since it is based on past observations, the MA can delay the detection of sudden anomalies or emerging trends. Another challenge is its fixed window size, which means it may need to adapt better to changing trends or seasonal variations in the data. Selecting the right window size can be tricky; a small window makes the MA more sensitive to noise, while a larger window may reduce its responsiveness to meaningful changes in the data. Thus, finding the optimal balance is crucial for practical use.

Exponential Moving Average (EMA)

The Exponential Moving Average addresses some of the limitations of the MA by assigning more weight to recent observations (Kim, 2020). The formula for EMA at time *t* is:

$$EMA(t) = \alpha \,.\, x(t) + (1 - \alpha)\,.\, EMA(t-1) \qquad (2)$$

where:

1. *x(t)* is the value of the time series at time *t*,
2. *α* is the smoothing factor, $0 < \alpha \leq 1$,
3. EMA(t−1) is the previous EMA value.

The smoothing factor α\alphaα is often calculated based on the window size *n* as:

$$\alpha \, \frac{2}{n+1} \qquad (3)$$

The EMA places exponentially decreasing weights on older data, meaning recent observations have a more significant influence on the current EMA.

This outlines the benefits and drawbacks of moving average algorithms step-by-step.

Exponential Moving Average-Based Anomaly Detection:

1. Step 1: Calculate the EMA recursively as data arrives.
2. Step 2: Compute the residual *r(t)=x(t)−EMA(t)*.
3. Step 3: Use a similar threshold mechanism to flag anomalies.

The EMA also has several advantages that make it a valuable tool for anomaly detection. One of its key strengths is its responsiveness. By giving more weight to recent data, the EMA responds more quickly to short-term changes, allowing it to detect sudden anomalies faster than the MA. This characteristic is especially beneficial in dynamic systems where real-time anomaly detection is crucial. Additionally, the EMA reduces the lag seen in MA because it prioritizes recent observations. This reduction in lag makes the EMA particularly well-suited for environments that require timely reactions to changing data patterns.

Another advantage of the EMA is the flexibility offered by the smoothing factor α (Cai, 2021). This factor allows users to easily adjust the EMA's sensitivity, enabling it to adapt to different types of data and varying levels of volatility. As a result, the EMA can be fine-tuned to strike a balance between responsiveness and noise reduction, depending on the system's specific needs.

However, the EMA also comes with some disadvantages. Its increased responsiveness can sometimes be a double-edged sword. In noisy datasets, the EMA might overreact to minor fluctuations that are not true anomalies, leading to false positives. This sensitivity to noise can reduce the accuracy of anomaly detection in environments with frequent data variations. Moreover, the recursive nature of the EMA’s calculation adds to its computational complexity. Unlike the simple calculation of MA, EMA requires maintaining previous EMA values, which can become computationally expensive, especially when dealing with large datasets or continuous streaming environments.

Comparative Analysis

When choosing between MA and EMA for anomaly detection, the decision depends mainly on the nature of the data and the system's requirements. A critical difference between the two methods is their speed of detection. EMA is more responsive to sudden changes and detects anomalies faster than MA, making it ideal for real-time systems like financial markets or sensor networks where timely detection of outliers is crucial.

In contrast, MA offers more excellent stability due to its equal weighting of past observations, which makes it a better choice in scenarios where gradual anomalies are expected or where short-term fluctuations are less significant. For example, MA is often used for long-term monitoring of environmental data, as it helps filter out short-term noise and provides a smoother trend line. The two methods also vary in their adaptability to trends. EMA is more suitable for data with evolving patterns, as it places more emphasis on recent observations, allowing it to adjust quickly to changes. On the other hand, MA's fixed window approach makes it less adaptable to shifting trends, making it less effective in non-stationary environments where the data patterns change over time.

Due to its recursive calculations, EMA requires more computational resources. MA is more efficient in high-throughput environments with limited processing power. However, modern computing advancements have made the difference in complexity less significant, with EMA's responsiveness often outweighing its computational cost in many applications. Both MA and EMA are widely used across various fields. EMA is frequently applied in financial markets, network monitoring, and healthcare for real-time detection of anomalies, such as stock price fluctuations or sudden changes in patient vital signs. MA, on the other hand, is often preferred for long-term trend monitoring due to its stability. Ultimately, the choice between MA and EMA depends on the data's characteristics and the system's needs for sensitivity, efficiency, and adaptability. Hybrid approaches combining both methods can further improve anomaly detection systems.

3. RESULT AND DISCUSSION

The experiments were carried out to evaluate the performance of two widely used anomaly detection techniques: the MA and the EMA. The purpose of these experiments was to assess how each method detects anomalies within a streaming dataset. Various metrics, such as detection speed, precision, recall, false positive rate, and computational efficiency, were used to compare the effectiveness of these approaches. By analyzing their strengths and weaknesses, the experiments aimed to provide insights into which method is better suited for different types of anomaly detection tasks, depending on the nature of the data and the system's requirements.

Dataset Used

For this study, we employed the Numenta Anomaly Benchmark Dataset, which is designed to simulate real-world scenarios in anomaly detection. The dataset contains time-series data with a mix of regular and strange events, including sudden spikes, gradual drifts, and recurring noise. This made it suitable for evaluating the performance of both the MA and EMA methods in identifying anomalies. By testing these methods on such a dataset, we were able to assess their ability to detect different types of anomalies in real-time data streams.

As shown in Figure 1, the dataset Travel Time, obtained from 'nab/realTraffic/realTraffic/TravelTime_387.csv', is used.

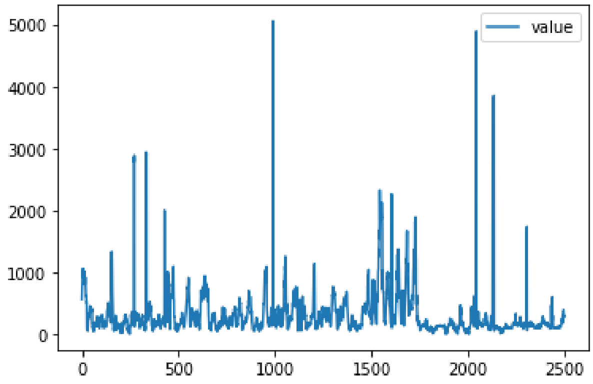


Figure 1. Time Travel (traffic) dataset visualization plot

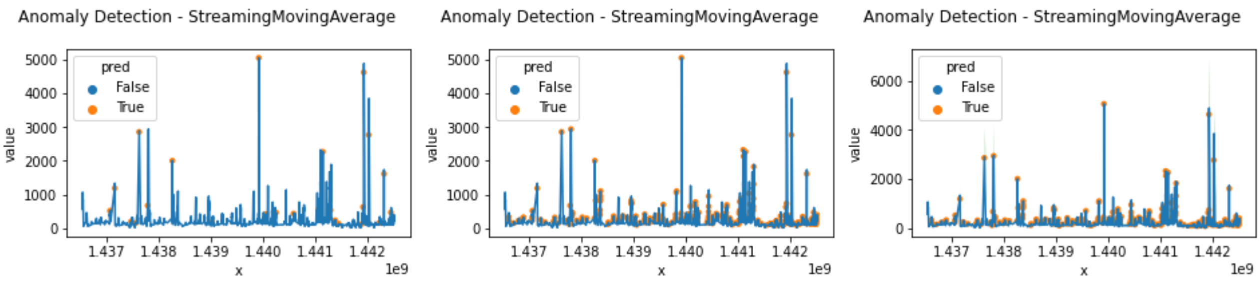
Figure 1 displays a line plot of a time-series dataset, with the x-axis likely representing time or sequence steps and the y-axis indicating the values recorded over time. The overall pattern shows that the data exhibits relatively consistent fluctuations, with occasional sharp spikes that deviate significantly from the regular trend. These spikes could potentially represent anomalies or outliers in the dataset, drawing attention to sudden changes or unusual events.

Upon closer observation, the majority of the data remains relatively stable, with only minor fluctuations between the spikes. This suggests that the dataset generally follows a regular pattern, with these large spikes indicating rare but significant deviations from the norm. Such behavior is expected in datasets where anomalies occur infrequently but can have a significant impact when they do, making them essential to detect and analyze.

The sharp spikes could be indicative of outlier events that require further investigation, particularly in anomaly detection tasks. Techniques such as MA and EMA are commonly used to smooth the data and help identify such anomalies. These methods would provide a more apparent baseline for determining whether these spikes truly represent anomalies or simply normal fluctuations in the data.

Moving Average Results

The primary function of the MA method is the most common type of average used in Time Series problems. We perform the sum of recent data points and divide them by the period. Further, we check if the new record is far from the expected value. The predicted value range is computed using the formula MA + standard deviation * threshold; if it is out of the expected value range, we report it as an anomaly, as shown in Figure 2, based on threshold values 2.0 and 1.0 and based on dumpling we set as True.

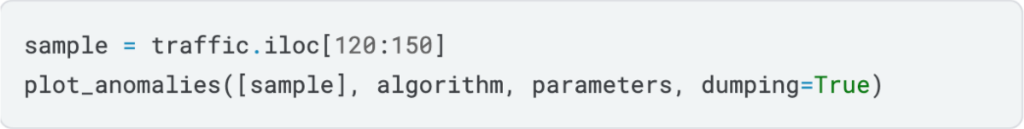


(a) Threshold = 2.0 (c) Threshold = 1.0 (c) Dumpling = True

Figure 2. Anomaly detected based on threshold values 2.0 and 1.0 and the dumpling = True.

The figures illustrate how different threshold values and the setting "dumpling = True" affect anomaly detection in a time-series dataset. In Figure (a), a higher threshold of 2.0 results in fewer but more significant anomalies being detected, focusing only on large deviations. In Figure (b), lowering the threshold to 1.0 increases sensitivity, detecting a more substantial number of minor anomalies. Figure (c) introduces the "dumpling = True" condition, which appears to cluster or smooth anomalies, altering the detection pattern compared to the previous figures. This highlights how varying thresholds and additional parameters influence the effectiveness and behavior of anomaly detection.

Next, we try to focus on a specific certain data, as shown in Figure 3.



```
sample = traffic.iloc[120:150]
plot_anomalies([sample], algorithm, parameters, dumping=True)
```

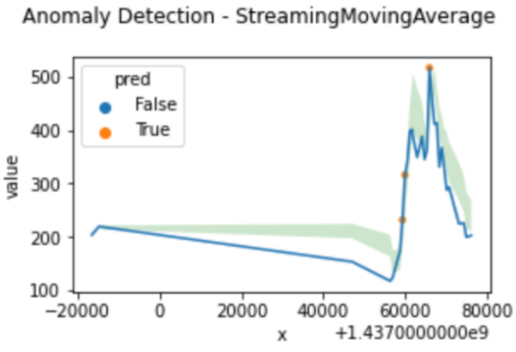


Figure 3. Zoom the specific data.

The image provides a close-up view of a time-series dataset, focusing on data points between indices 120 and 150. This zoomed section is part of an anomaly detection process using the MA algorithm. The purpose of this detailed view is to observe the dataset's behavior closely and evaluate the algorithm's ability to detect anomalies accurately in this specific range. By zooming in, we can clearly see how the data fluctuates and how the model responds to these variations.

In the plot, the blue line represents the actual data values over time or another sequential variable. Surrounding the blue line, the green shaded area likely indicates the expected normal range or confidence interval computed by the MA algorithm. Data points that stay within this shaded region are considered normal, while those that deviate significantly from this range are flagged as potential anomalies.

The orange dots signify the points identified as anomalies by the algorithm. These anomalies are detected when the actual data values fall outside the expected range, which is represented by spikes in the blue line that extend beyond the green-shaded area. The legend categorizes the data points into "False" (regular points) and "True" (anomalies), showing that the algorithm successfully flagged the orange points as deviations from the expected behavior.

The parameter "dumpling=True" plays a role in the detection process. While its exact function isn't explicitly clear, it likely affects the clustering or smoothing of detected anomalies. It could group nearby anomalies or apply some filter to improve the detection's accuracy, helping to avoid false positives or missed anomalies.

This plot demonstrates how the MA algorithm identifies outliers based on deviations from typical trends. By focusing on this smaller data segment, it becomes easier to evaluate whether the algorithm is functioning effectively in detecting true anomalies while also providing insight into how parameter adjustments like "dumpling=True" can influence the results.

Exponential Moving Average Results

EMA focuses more on recent data by assigning more weight to new data points. So, they are weighted by timestamp—the most recent has more importance. Further, we check if the new record is far from the expected value. The predicted value range is computed using the formula EMA + standard deviation * threshold; if it is out of the expected value range, we report it as an anomaly. Also, the EMA needs a parameter called alpha that determines the importance of the last record; as shown in Figure 4, the threshold is set to 1.5, and alpha is set to 0.5.

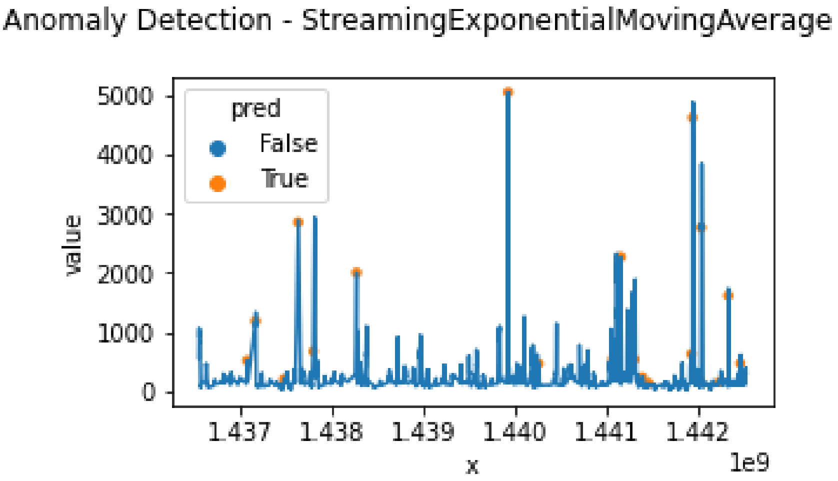
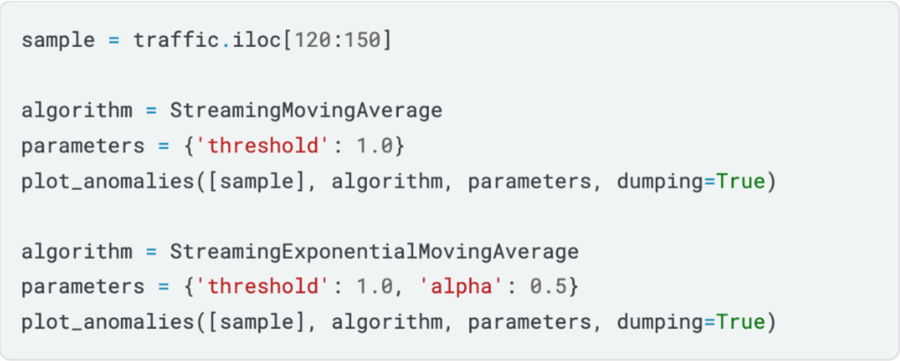


Figure 4. Anomaly detected based on EMA with threshold = 1.5 and alpha = 0.5

The image shows anomaly detection using the EMA with a threshold of 1.5 and a smoothing factor (alpha) of 0.5. The blue line represents the time-series data, while the orange dots mark the detected anomalies, which occur when data points exceed the set threshold of 1.5. The alpha value of 0.5 balances the EMA's sensitivity to recent versus past data, creating a moderately responsive model that smooths the data while still detecting significant deviations. This setup ensures that only substantial anomalies are flagged, avoiding false positives from more minor fluctuations.

Comparison Results Analysis

When we compare the MA with the EMA, we notice that the EMA smoothes the expected value and avoids detecting unnecessary outliers. The comparison results were based on the same specific data, as shown in the code in Figure 5 and are shown in Figure 6.



```
sample = traffic.iloc[120:150]

algorithm = StreamingMovingAverage
parameters = {'threshold': 1.0}
plot_anomalies([sample], algorithm, parameters, dumping=True)

algorithm = StreamingExponentialMovingAverage
parameters = {'threshold': 1.0, 'alpha': 0.5}
plot_anomalies([sample], algorithm, parameters, dumping=True)
```

Figure 5. Python code to compare the different methods with their parameters.

The Python code snippet in Figure 5 compares two anomaly detection methods: the MA and EMA. The dataset is narrowed to a specific subset (`traffic.iloc[120:150]`) to focus on a smaller range for analysis. Both methods use a threshold parameter of 1.0, meaning data points that deviate by more than this value from the expected trend are flagged as anomalies.

For the MA, the algorithm calculates a simple moving average and flags anomalies based on the threshold. The function `plot_anomalies` visualizes the results with an additional parameter `dumpling=True,` which likely modifies the anomaly detection process by clustering or smoothing out fluctuations to refine the results. In contrast, the EMA applies a smoothing factor (`alpha = 0.5`) to give equal weight to recent and past data points, making it more responsive to recent changes in the dataset. This method is better suited for dynamic data where trends shift more quickly. Similar to the MA method, the results are visualized with the same threshold and `dumpling=True` parameter for consistency in comparison.

The code facilitates a direct comparison between the two methods, showing how each handles anomaly detection under the same threshold but with different smoothing techniques. The visualization helps users assess the strengths and weaknesses of each algorithm, with the EMA being more responsive to recent changes and the MA providing a more stable, long-term trend view.

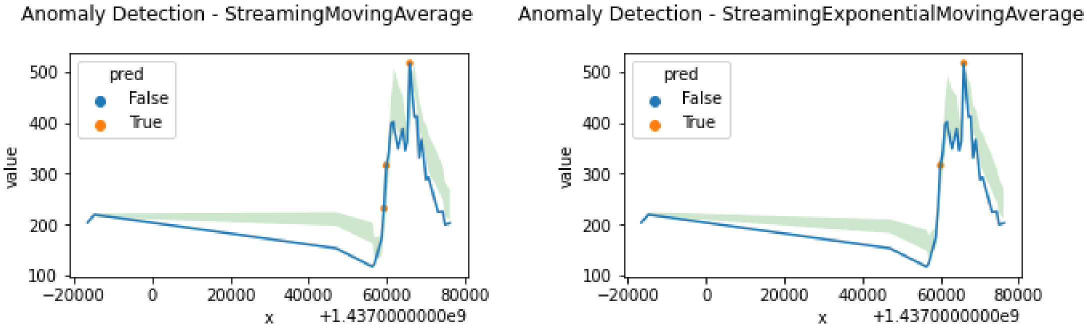


Figure 6. The comparison results of MA and EMA methods

The left plot shows the results from the MA method, while the right plot demonstrates the EMA method. In both plots, the blue line represents the actual time-series data, the green shaded area indicates the expected normal range and the orange dots highlight the detected anomalies.

In terms of speed of detection, the EMA, shown in the right plot, reacts more quickly to changes in the data. This is because EMA gives more weight to recent observations, allowing it to detect anomalies as soon as the data deviates from the expected trend. In contrast, the MA method on the left takes longer to respond to sudden changes because it averages data over a fixed window, making it slower in flagging anomalies when there are rapid spikes. This delay in detection is a characteristic trade-off of the MA approach.

The MA is more stable, as it smooths out minor fluctuations and focuses on the overall trend. This stability is reflected in the left plot, where fewer anomalies are detected, and the green-shaded area is relatively smooth. On the other hand, the EMA method in the right plot is more sensitive to recent changes, leading to more detected anomalies, which may include minor fluctuations. This makes EMA less stable but more responsive to dynamic changes in the data.

In terms of precision and recall, the EMA method achieves higher precision because it can quickly detect significant deviations, as shown by the orange dots in the right plot. However, this responsiveness can reduce recall, as it may miss gradual trends or flag false positives due to its sensitivity to more minor fluctuations. The MA, on the other hand, offers better recall since it focuses on detecting sustained anomalies. This is visible in the left plot, where fewer, but likely more meaningful, anomalies are detected.

The false positive rate is generally higher with EMA, as its sensitivity leads to more frequent detection of minor deviations, as evidenced in the right plot, where more anomalies are flagged. Conversely, the MA method in the left plot produces fewer false positives because it only detects more significant deviations, ignoring smaller, less critical spikes in the data.

Finally, in terms of computational efficiency, the MA method is less resource-intensive because it computes a simple average over a fixed window, making it more suitable for applications with limited computational resources. The EMA method requires more processing power due to its recursive nature, where each new data point influences the moving average, increasing the computational load.

The comparison in Figure 6 highlights the critical differences between MA and EMA in anomaly detection. EMA is faster and more responsive, making it ideal for environments requiring real-time anomaly detection, though it is more prone to false positives. MA, while slower, provides more excellent stability and fewer false positives, making it more suitable for applications focused on long-term trend monitoring and computational efficiency. The choice between the two methods depends on the specific needs of the detection system, such as speed, accuracy, and computational constraints.

4. CONCLUSION

In this article, we conducted a comprehensive comparison between two widely used anomaly detection methods: MA and EMA. Through detailed analysis and visual comparisons, we demonstrated that the choice between these methods depends heavily on the system's specific requirements and the characteristics of the data being analyzed.

The EMA method, with its ability to give more weight to recent data, proves to be more responsive and effective for real-time anomaly detection. It quickly identifies sudden changes or spikes in the data, making it ideal for dynamic environments such as financial markets or network monitoring, where timely identification of anomalies is crucial. However, EMA's sensitivity can result in a higher rate of false positives, particularly in noisy datasets, and it also demands more computational resources due to its recursive nature.

In contrast, the MA method offers more excellent stability by averaging data over a fixed window. It is less prone to false positives and is better suited for applications where gradual trends are more important than real-time changes. MA is computationally more efficient, making it preferable in resource-constrained environments. However, its slower detection speed may limit its effectiveness in rapidly changing scenarios.

The specific goals of the anomaly detection system should guide the decision between MA and EMA. For real-time, fast-paced environments, EMA offers a more responsive solution, while MA is better suited for applications that prioritize stability, computational efficiency, and long-term trend detection. In many cases, a hybrid approach combining the strengths of both methods could lead to more robust and adaptable anomaly detection systems.

AUTHOR INFORMATION

Corresponding Authors

Eka Puji Agustini, Universitas Bina Darma, Palembang, Indonesia.
Email: eka_puji@binadarama.ac.id

Authors

Mohd Zaki Zakaria, University Technology Mara, Malaysia.

REFFERENCE

BoltzmannBrain. (2016). *Numenta Anomaly Benchmark (NAB)*. https://www.kaggle.com/datasets/boltzmannbrain/nab

Braei, M., & Wagner, S. (2020). *Anomaly Detection in Univariate Time-series: A Survey on the State-of-the-Art*. https://doi.org/10.48550/arXiv.2004.00433

Cai, Z., Ravichandran, A., Maji, S., Fowlkes, C., Tu, Z., & Soatto, S. (2021). *Exponential Moving Average Normalization for Self-supervised and Semi-supervised Learning*.

Ejder, U., & Özel, S. A. (2024). A novel distance-based moving average model for improvement in the predictive accuracy of financial time series. *Borsa Istanbul Review*, *24*(2), 376–397. https://doi.org/https://doi.org/10.1016/j.bir.2024.01.011

Hansun, S. (2013). A new approach of moving average method in time series analysis. *2013 Conference on New Media Studies (CoNMedia)*, 1–4. https://doi.org/10.1109/CoNMedia.2013.6708545

Ivanovski, Z., & Milenkovski Zoran Narasanov, A. (2018). Time Series Forecasting Using a Moving Average Model for Extrapolation of Number of Tourist. In *UTMS Journal of Economics* (Vol. 9, Issue 2). https://hdl.handle.net/10419/281836

Karasu, S., Altan, A., Bekiros, S., & Ahmad, W. (2020). A new forecasting model with wrapper-based feature selection approach using multi-objective optimization technique for chaotic crude oil time series. *Energy*, *212*, 118750. https://doi.org/https://doi.org/10.1016/j.energy.2020.118750

Kim, T., Solanki, V. S., Baraiya, H. J., Mitra, A., Shah, H., & Roy, S. (2020). A Smart, Sensible Agriculture System Using the Exponential Moving Average Model. *Symmetry*, *12*(3). https://doi.org/10.3390/sym12030457

Nakano, M., Takahashi, A., & Takahashi, S. (2017). Generalized exponential moving average (EMA) model with particle filtering and anomaly detection. *Expert Systems with Applications*, *73*, 187–200. https://doi.org/10.1016/j.eswa.2016.12.034

Siffer, A., Fouque, P.-A., Termier, A., & Largouët, C. (2017). *Anomaly Detection in Streams with Extreme Value Theory*. https://doi.org/10.1145/3097983.3098144

van Rossum, H. (2019). Moving average quality control: principles, practical application and future perspectives. *Clinical Chemistry and Laboratory Medicine (CCLM)*, *57*(6), 773-782. https://doi.org/10.1515/cclm-2018-0795