Live Changepoint Detection and Portfolio Rebalancing

Literature Review of CUSUM-based methods
and an Interactive Simulation Toolbox

May 20, 2025

Angela Zhao* May 20, 2025
Department of Biostatistics, Johns Hopkins Bloomberg School of Public Health, USA

Abstract

In this study, we review the literature surrounding CUSUM-based changepoint detection. Starting in 1954, when Page first proposed the CUSUM test for i.i.d. processes, we work our way to the flexible "detect-and-reset" regimes used today. These principles are applied to create a complete workflow for monitoring live trading strategies. Our framework (i) recalibrates CUSUM after every alarm using a rolling post-event window, (ii) balances detection speed and false-alarms using warm-up and threshold heuristics, and (iii) supports AR- and GARCH-pre-whitening for volatility-clustering markets. An open-source Streamlit sandbox lets users stress-test these choices and visualizes the trade-off between hit-rate, detection delay, and time-averaged false-alarm rate. Monte-Carlo studies confirm that regular resets cap long-run false alarms, while GARCH residuals sharpen sensitivity when tuned correctly. Finally, a 2014-2025 U.S. equity case study shows CUSUM-triggered re- balancing can beat fixed-interval and buy-and-hold strategies during trending markets while exposing its limits in bear regimes. Together, the literature synthesis, methodology, and interactive tool supply a blueprint for continuous, resettable performance surveillance

*Johns Hopkins University, Advanced Probability IV / Adaptive Trading (2025). E-mail: `azhao@jh.edu`

1 Introduction

Online changepoint detection is a critical task in fields where rapid decision-making depends on real-time data, such as financial trading, network monitoring, and quality control. In particular, financial time series often exhibit sudden shifts in return patterns or volatility that can dramatically affect portfolio performance. Traditional batch or offline changepoint detection methods, while theoretically sound, are often too slow or inflexible for these settings.

Classic CUSUM methods are optimal under idealized assumptions—independent, identically distributed observations and known change parameters—but real-world data streams rarely meet these conditions. In particular, financial time series are noisy, autocorrelated, and heteroskedastic, with structural breaks occurring at unknown times. Moreover, applying CUSUM in practice requires tuning several parameters without closed-form guidance, which can significantly affect both sensitivity and specificity.

This paper provides a comprehensive literature review on the evolution of CUSUM changepoint detection methods, with a focus on recent developments that support online, adaptive, and resettable detection. We emphasize practical considerations, such as handling multiple change points, high-dimensional inputs, and volatility clustering. Additionally, we introduce an interactive Streamlit simulation toolbox that allows users to explore how CUSUM tuning affects performance in realistic settings. The app supports Monte Carlo simulation, AR and GARCH pre-whitening, and real-time performance metrics such as hit rate, lagged alarm detection delay (LADD), and time-averaged false alarm rate (TAFAR).

The remainder of the paper is structured as follows. Section 2 reviews the theoretical foundation and recent developments in CUSUM changepoint detection. Section 3 introduces the simulation engine, explains the data generation and CUSUM implementation, and presents illustrative output. Section 4 applies the proposed framework to real financial data and compares portfolio strategies that use or ignore online detection. Section 5 concludes with discussion and future research directions.

2 Literature review

The CUSUM method to detect a single change in the distribution of a process was first introduced by E.S. Page in 1954 [17]. Building on the ideas in sequential hypothesis testing, Page proposed to use a cumulative sum to accumulate evidence of a shift in parameter and signal an alarm when the sum exceeds a critical threshold. The method assumed independent, identically distributed (i.i.d.) observations with a known pre-change distribution and post-change shift. Decision rules are then derived while controlling false alarm rates. Under these assumptions, this CUSUM method is asymptotically optimal in the sense that it minimizes the detection delay for any given false-alarm frequency [11, 14]. Further theoretical advances extended CUSUM's optimality results to Bayesian and continuous-time settings [19, 21]. It is also important to note that for sufficiently large mean shifts, Crainiceanu and Vogelsang (2007) showed the CUSUM test loses power because the variance estimator inflates [6].

Traditional "offline" uses of CUSUM have limitations that motivated live/online extensions. Classic CUSUM methods assume a stationary regime until one abrupt change and require choosing a fixed threshold or stopping rule based on known pre-change properties. If the data exhibit gradual drifts, autocorrelation, or unknown change magnitudes, a fixed offline CUSUM may either give many false alarms or delay detection. Moreover, retrospective CUSUM tests only flag a change after analyzing the full data sequence, which is inadequate for real-time applications that demand prompt alerts.

To handle multiple change-points without inflating the false discovery rate in online settings, Basseville and Nikiforov first proposed restarting sequential tests post-detection in 1993 [2]. These principles were later formalized into detect-and-reset cycles and theoretical performance bounds for controlling the overall false alarm rate were derived [10, 22].

Modern literature for online CUSUM detection methodology focuses on relaxing model assumptions and improving adaptability. At its core, the CUSUM at time t can be viewed as the maximum, over all potential change-times k, of the log-likelihood ratio that the observations since k came from the post-change model. To address nonstationary environments where the parameters are unknown, Montes de Oca et al. (2010) proposed a nonparametric CUSUM for network traffic that divides time into slots and uses historical data in each slot to update the CUSUM decision rule, thereby accounting for slowly varying baseline rates [7]. Such methods address nonstationary environments by periodically "resetting" or adjusting the CUSUM to avoid accumulation of irrelevant long-term drift. The trade-off is that frequent resets or short reference windows can increase detection delay for small persistent changes, so tuning is required. Furthermore, when working with dependent, high-dimensional data, naive extensions often fall short and more specialized tests are needed. For example, Mei (2010) [13] and others have proposed procedures that monitor the maximum or average of many univariate CUSUM statistics to detect a change affecting a subset of components. More sophisticated frameworks use dimension reduction or sparsity assumptions (e.g. assuming only a few among many streams undergo change) to maintain detection power without excess false alarms. Overall, the literature seems to be moving from univariate i.i.d. CUMSUM to multivariate or even functional methods that can be applied to dependent data [12].

These methods were adapted across many fields of research, from their quality control origins to checking the stability of regression coefficients [3], to monitoring network traffic [7] and maternal blood glucose [20]. Financial markets are also a natural application for online changepoint detection. Below, we review some examples from the past decade. First, the CUSUM stopping rules have been used to detect entry and exit from trends with an application to high-frequency U.S. Treasury note data [4]. Another application could be high-frequency trading and surveillance to detect abnormal trading patterns as they occur. One technical report, for example, adapts an adaptive CUSUM to network packet data to detect bursts of activity [7]. In practice, one must carefully balance sensitivity and specificity – a CUSUM tuned too tightly will flag many benign fluctuations (false alarms), whereas too loose a threshold could miss subtle but important shifts. Furthermore, authors have used one-shot detectors to evaluate trading rules post hoc [5] and have discussed regime shifts in dynamic allocation but do not use real-time detection or resets [16]. However, in terms of monitoring portfolio strategies, few papers combine online detection, multiple breaks, strategy monitoring, and a reset and re-estimate framework. Overall, financial time series are noisy, heavy-tailed, and often have regime-dependent volatility. It is important to note that a CUSUM that works for a slow-moving index might trigger too often in a more volatile market, so adaptive calibration (or even self-learning thresholds) is needed for robust trading applications.

The last extension we will cover in this literature review is handling autocorrelated and heteroskedastic data. Classical CUSUM formulas impose independence and homoskedastic restrictions on the data and violations of these assumptions can lead to false detections. Two major strategies are used to handle these restrictions: (1) pre-whiten the data using a GARCH (Generalized Autoregressive Conditional Heteroskedasticity) filter to handle volatility clustering and (2) directly integrating GARCH modelling into the detection procedure. Pre-whitening involves filtering the data to remove serial correlation before applying a CUSUM test. Qiang and Ruggieri (2023) demonstrate the benefit of pre-whitening a strongly autocorrelated series [18]. Their results show that it greatly reduces the number of false alarms detected by a Bayesian changepoint model. However, pre-whitening can also remove some signal along with the noise, especially if the shift happens gradually. On the other hand, one can directly account for heteroskedasticity in the CUSUM procedure. If volatility shifts are not accounted for, a CUSUM meant to detect mean changes might instead trigger on a sudden increase in variance. A recent advance by Astill et al. (2023) [1] illustrates a solution: they replace the fixed variance in the CUSUM calculation with a real-time estimate of volatility, using a kernel-based estimator of the instantaneous variance. Moreover, Horváth et al. (2024) develop a CUSUM of quasi-likelihood scores to monitor for changes in GARCH(1,1) model parameters, such as a switch from a normal volatility regime to an explosive one [9]. They design custom boundary functions (with time-varying critical values) to achieve fast detection of a volatility regime change while controlling false alarms in a sequential manner.

3 Simulation tool

A closed form for determining the warmup length and critical value used in online CUSUM detectors have yet to be derived. Monte Carlo simulations remain a simple and effective way to identify parameters suitable to specific data and use cases. Here, we share a Streamlit application to visualize how modifying parameters affect the online detection performance. Please find the tool at https://adaptive-trading.streamlit.app/. Due to limitations with free Streamlit deployment and requirement-heavy implementation, the first time loading the app or running too many simulations may be slow.

Below, we describe the data generating mechanism and implementation details.

3.1 Simulating data streams

We implemented one-sided CUSUM statistics for detecting shifts in mean and shifts in variance. Artificial data was simulated in two steps. In the first step, the user specifies an event rate p and the baseline and post-shift means (or variances), $\mu_0, \mu_1, \sigma_0, \sigma_1$ respectively. We assume time is discrete. Event times, $\tau_1, \tau_2, ..., \tau_i, ...$, are drawn independently from a $Geometric(p)$ distribution, at which the mean and/or the variance flips with 50% chance. Denote by μ_i, σ_i the mean and variance of the data during the time interval $[\tau_i, \tau_{i+1}]$. In the second step, we simulate data between τ_i and τ_{i+1} by making independent draws at every timepoint from a Gaussian distribution with mean mu_i and variance σ_i. Figure 1 displays an example of the simulated data.

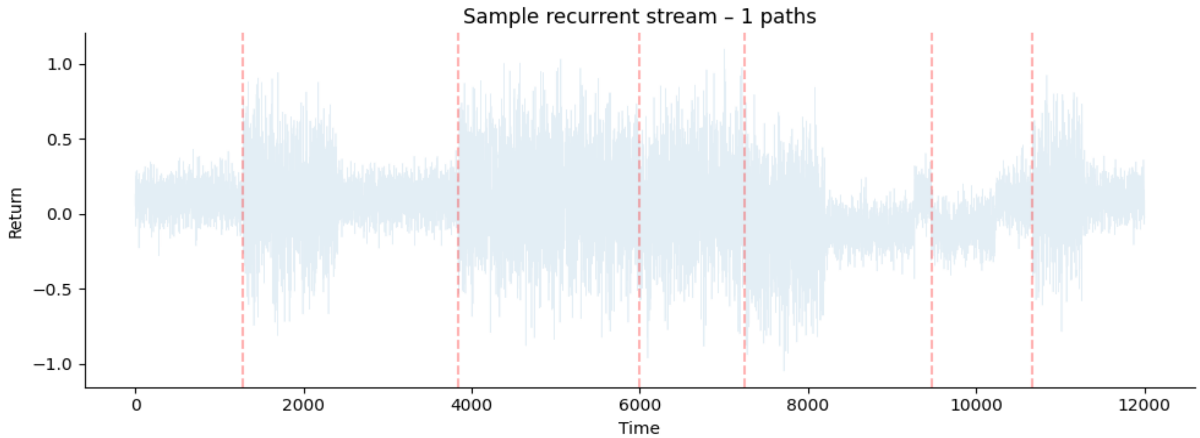

Figure 1: Simulated data using the procedure described in Section 3. Red dotted lines mark the timepoints at which either a downwards shift in mean or upwards shift in variance occur. The mean shifts between 0.1 and −0.1 and the standard deviation shifts between 0.1 and 0.3.

3.2 CUSUM formulation

The online CUSUM detector is formulated using the likelihood ration increment between a null and alternative model. For simplicity, we discuss the implementation of one-way detectors for downwards shifts in the mean and upwards shifts in the variance. However, this can easily be modified to one-way detectors for upwards shifts in the mean and downwards shifts in the variance simply by changing signs. Likewise, our implementation can be extended to two-way detectors.

Let x_t be the univariate data stream. After each alarm, the detector "blacks out" W observations for estimating the new initial parameters, (μ_0, σ_0). Assume $N(\mu_0, \sigma_0^2)$ is the null model and $N(\mu_0 + \delta_\mu, \sigma_0^2)$ is the alternative model.

The mean–down detector $(\mu\downarrow)$ is specified as follows:

$$k \;=\; \frac{\delta_\mu}{2}, \qquad g_t \;=\; \max\Big\{0,\; g_{t-1} + x_t - \mu_0 - k\Big\}, \quad g_0 = 0,$$

$$\tau_\mu \;=\; \inf\{t \geq 1 :\; g_t \geq h_\mu\}.$$

where x_t is the observed value at time t, μ_0 is the baseline mean, estimated from the previous W points, σ_0 is the baseline standard deviation (needed only for scaling) estimated from the previous W points, δ_μ is the minimum change we want to detect, $k = \delta_\mu/2$ is the one-sided CUSUM reference value, g_t is the one-sided CUSUM statistic for $\mu\downarrow$, h_μ is the critical value, and τ_μ is the stopping time when the alarm fires. After an alarm we reset $g_t \leftarrow 0$, skip the next W samples, and re-estimate (μ_0, σ_0).

Likewise, to specify a variance-up detector, we assume $N(\mu_0, \sigma_0^2)$ is the null model and $N(\mu_0, \sigma_1^2)$ is the alternative model. The variance–up detector $(\sigma\uparrow)$ is specified as follows:

$$s_t \;=\; \max\Big\{0,\; s_{t-1} + \ln\frac{\sigma_1}{\sigma_0} \;+\; \tfrac{1}{2}\Big[\tfrac{x_t - \mu_0}{\sigma_0}\Big]^2 \Big(1 - \sigma_0^2/\sigma_1^2\Big)\Big\}, \quad s_0 = 0,$$

$$\tau_\sigma \;=\; \inf\{t \ge 1:\; s_t \ge h_\sigma\}.$$

where $\sigma_1 = (1 + \delta_\sigma)\,\sigma_0$ is the post-shift standard deviation, δ_σ is the design shift in *fractional* units (0.4 $\Rightarrow$ 40% jump), s_t is the variance-up CUSUM statistic, h_σ : variance decision threshold, τ_σ is the first time s_t crosses h_σ, and the remaining definitions are as defined above.

Finally, CUSUM methods assume that data is i.i.d. However, in practice, data often violates stationarity assumptions. To accommodate this, we wrote wrappers that pre-whiten the data using AR(1) or GARCH(1,1) filters. The wrapper class can be further extended to Student-t filters to accommodate heavy-tailed data.

3.3 Evaluation metrics

We evaluate the performance of our CUSUM detectors using the hit rate, TAFAR, and LADD.

Let $\{\tau_j\}_{j\ge1}$ denote the (unobservable) change–points generated by the hazard rate p, and let a_j be the first alarm raised *after* τ_j. Assume there are a total of J true shifts in the simulation. Fix a tolerance window of length ℓ (here we set $\ell = \tau_j - \tau_{j-1}$, i.e. one change-period). Then, the hit rate is calculated by counting a *hit* whenever the detector fires within one period of the actual regime change, and average over all simulated shifts.

$$\text{Hit-rate} \;=\; \frac{1}{J}\sum_{j=1}^{J} \mathbf{1}\{\,0 < a_j - \tau_j \le \ell\,\},$$

Conditional on a hit, LADD averages the extra waiting time:

$$\text{LADD} \;=\; \mathbb{E}\big[\,a_j - \tau_j \;\big|\; 0 < a_j - \tau_j \le \ell\big].$$

The expectation is taken over all alarms that qualified as hits in the definition above. A smaller LADD means the detector reacts sooner once a true shift occurs.

Lastly, we define the indicator

$$F_t = \mathbf{1}\{\text{at least one false alarm is } \textit{open} \text{ at time } t\}.$$

Here an alarm is *false* if it fired when no change-point had occurred in the preceding ℓ observations. Over a horizon of T steps,

$$\text{TAFAR} \;=\; \frac{1}{T}\sum_{t=1}^{T} F_t.$$

where F_t flags whether time t is contaminated by a previous false alarm and T is the total length of the Monte-Carlo path. In summary, TAFAR measures the fraction of time the detector leaves the portfolio in a "false-alarm state." Lower TAFAR signals a cleaner detector. Clearly, there is a tradeoff between LADD and TAFAR. It is up to the user to identify what is an acceptable threshold for their usage. For example, if the user wishes to monitor daily portfolio returns, a TAFAR of 10^{-2} is more than sufficient, as that would indicate one false alarm approximately every 100 days. If, instead, the user wants to monitor portfolio returns at every minute, a TAFAR of 10^{-3} may be more appropriate.

3.4 Interpretation

In this section, we will analyze some results from the simulations. The bubble plot in Figure 2 illustrates the tradeoff frontier between LADD and TAFAR across multiple hyperparameter configurations for detecting downwards mean shifts. Each point represents a unique combination of detection threshold h and warmup window W, with bubble size representing hit rate. Several key patterns emerge. Increasing δ_μ makes the cumulative drift under a true shift steeper, so a fixed threshold h will be hit sooner *if* the change is at least δ_μ; but the detector becomes less sensitive to smaller moves. The threshold h balances TAFAR and LADD. In general, smaller h values generally reduce LADD (faster response) but increase TAFAR (more false positives). Larger W yield cleaner detection (lower TAFAR), likely due to more stable estimates of pre-change parameters, at the cost of slightly delayed reaction. The lower-right region (low TAFAR and low LADD) represents ideal configurations—highlighting a Pareto frontier of settings worth considering depending on the application's tolerance for false alarms. Interestingly, multiple configurations achieve comparable hit rates but differ substantially in LADD and TAFAR, emphasizing the need to jointly consider all three metrics during tuning.

Figure 3 shows a grid of heatmaps depicting hit rate across combinations of h and W, under different combinations of design shifts δ_μ and event rates p. This grid visualization is particularly useful for data-driven hyperparameter selection. Higher design shifts lead to stronger CUSUM signals and more forgiving hit rates across a wide range of parameters. Intermediate W (e.g., 100–150) seem to balance sensitivity and stability better than very small or very large windows. Figure 3 also shows a performance sweep of the detector as a function of h at $p = 0.0017$ and $W = 150$ for different δ_μ. Increasing h lowers TAFAR (cleaner detector) but increases LADD (slower alarms). Hit-rate curves are non-monotone, revealing a sweet spot ($h = 6$ when $\delta_\mu = 0.3$ for example, where sensitivity is maximised without excessive false alarms.

Figure 4 (left) compares the hit rate of the CUSUM mean-down detector across three pre-whitening settings: no pre-whitening ("raw"), AR(1) filtering, and GARCH(1,1) filtering, for fixed p and W. Across two values of h, we observe that GARCH filtering consistently yields the highest hit rate. The "raw" (unfiltered) series performs well too, particularly at lower h, but shows slight degradation as the threshold increases. AR filtering lowers hit rates. The right figure shows an interesting contrast. While GARCH filtering increases sensitivity (hit rate) and reaction time, it can also increase false alarms. Interestingly, these simulations were performed on i.i.d. data simulated using the procedure in 3.

4 Application to real stock data

We use stock data to illustrate one potential real-world application of online CUSUM detection. Closing prices of 10 stocks between January 1, 2014 and May 19, 2025 were retrieved from Yahoo Finance and live in the *data* folder in the Github repository. We compare the portfolio-wide returns of strategies that use changepoint detection against those that do not.

4.1 Portfolio strategies

We compare the cumulative returns of four different portfolio strategies, which we will refer to as buy-and-hold, fixed, CUSUM $\mu\downarrow$, and CUSUM $\sigma\uparrow$. The buy-and-hold strategy puts equal weight on each stock at the start of the backtesting horizon and makes zero trades.

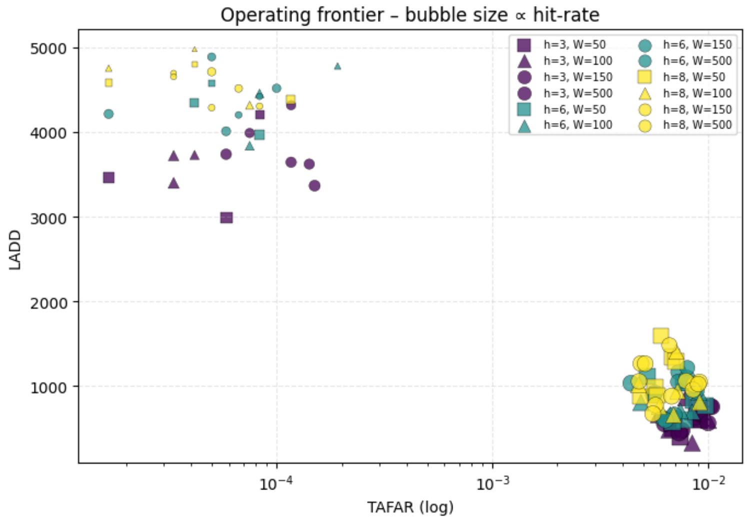

Figure 2: Bubble plot where each point is a Monte-Carlo average over 100 simulated paths with 12,000 time steps and event rate$p = 0.0017$. Marker shape encodes the warm-up window $W\{50, 100, 150\}$ (square, triangle, circle respectively) and colour encodes the CUSUM threshold $h\{3, 6, 8\}$. Lower-right is the desirable region (small LADD, small TAFAR). For this regime, $h = 3$ with $W = 50$ produces the fastest detections but at the cost of a higher false-alarm burden, whereas larger h and/or W shift the operating point upward or rightward.

The remaining use an aggressive momentum-based strategy that assigns all the weight to the N stocks with the highest momentum during a look-back window. The fixed strategy rebalances the portfolio at fixed intervals. The CUSUM $\mu \downarrow$ and $\sigma \uparrow$ strategies use the respective one-sided detectors defined in Section 3 and rebalances when an alarm sounds.

The user may specify the following quantities: the look-back window for the momentum strategy, the fixed window at which the fixed strategy rebalances, the number of stocks that are selected by the momentum strategy N, the minimum change they want to detect δ, the critical value h, the length of the warmup period W, select pre-whitening filters, and transaction costs.

4.2 Results suggest that informed portfolio rebalancing can increase returns during backtesting

Figures 5 shows an example of a "winning" strategy, where the CUSUM$\mu\downarrow$-based rebalancing strategy outperforms all other strategies, but in particular the fixed and buy-and-hold strategies. On the other hand, by simpling changing the horizon from 2014-2025 to 2018-2020 in Figure 6, all rebalancing strategies "lose" to buy-and-hold.

5 Discussion and Next Steps

The simulation framework confirms several findings from the literature: (1) Resetting after detection is critical to control long-run false alarm rates; (2) Sensitivity to shifts improves with larger target δ, but at the cost of missing smaller changes; (3) Pre-whitening the data—especially using GARCH(1,1) filters—can substantially improve hit rate, even if the

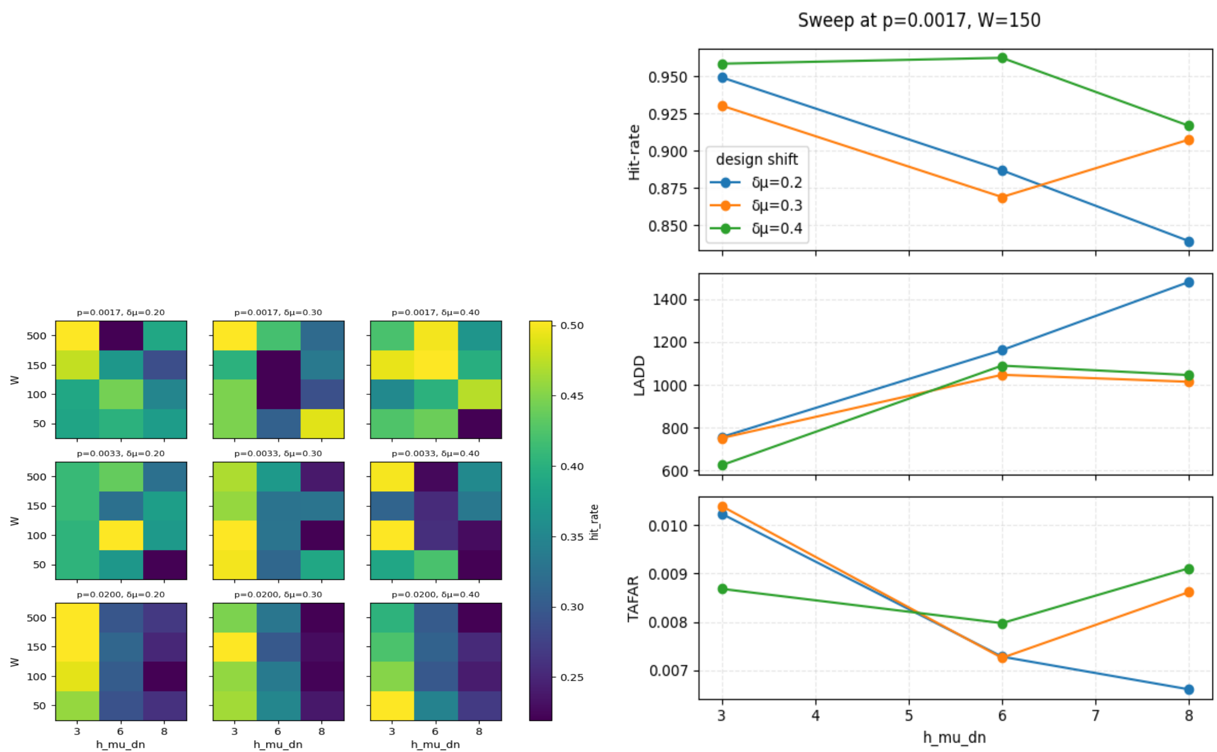

Figure 3: On the left, we have a grid of heatmaps that summarize the hit rates from 100 Monte-Carlo simulations for different parameters. Rows correspond to increasing event rates p and columns to increasing design shifts δ_μ. Within each panel, the horizontal axis is the threshold h and the vertical axis is W. Larger design shifts or lower hazard rates produce broad yellow bands i.e., robust operating regions—whereas small shifts combined with frequent changes require careful tuning. On the right, we have a performance sweep of the mean-down detector as a function of threshold h at $p = 0.0017$ and $W = 150$. The three panels report (top) hit rate, (middle) LADD, and (bottom) TAFAR for different design shifts δ_μ.

model is misspecified.

Notably, we show that GARCH filtering increases the responsiveness of the detector to true *volatility* shifts but also increases the rate of false alarms if not calibrated carefully. This points to an important tradeoff: volatility modeling can improve sensitivity but must be tempered by proper thresholding and warmup handling to avoid overfitting transient noise.

Although unintuitive, similar results have been shown in literature. Nelson (1992) proposed that ARCH models provide consistent estimates of the conditinal variance in high-frequency data settings even under model misspecification [15]. Dong et al. (2019) find that variance CUSUM detectors miss jumps on raw returns but fires cleanly on GARCH residuals [8]. However, Wong and Li (2009) warn about the implications of applying GARCH filters to data that do not inherently follow a GARCH process. Their study reveals that using GARCH models on non-GARCH data can lead to misleading inferences, particularly in risk management contexts [23]. It seems there is still room to explore the consequences

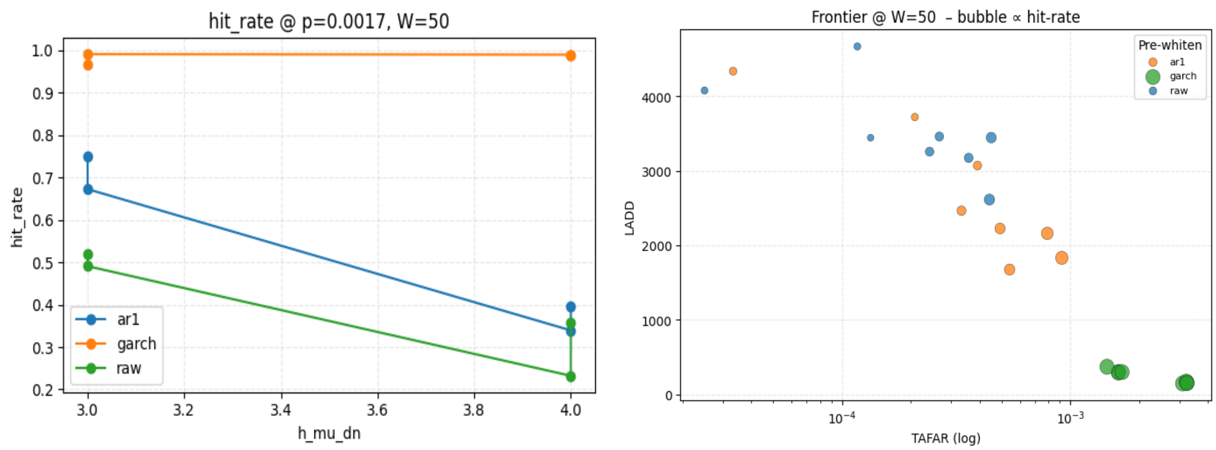

Figure 4: The left plot compares hit rate across pre-whitening methods (raw, AR(1), and GARCH(1,1) for variance-up detection. Hit rate was plotted as a function of CUSUM decision threshold $h\{3,4\}$, under fixed $p = 0.0017$ and $W = 50$. The right plot shows the results from the same simulations but with respect to LADD and TAFAR. Results were generated from 100 simulations.

and effectiveness of using GARCH to "boost" detection under possibly misspecified settings for different contexts and use cases.

Our current implementation assumes normally distributed innovations and models only univariate detectors. In practice, asset returns may exhibit skewness, kurtosis, or heavy tails, which would require more robust (e.g., Student-t) likelihood models or nonparametric approaches. Additionally, correlation among assets is not currently handled in the simulation but may be critical for detecting cross-asset regime shifts.

In the real-data example, we observe that CUSUM-based rebalancing strategies can outperform fixed strategies during certain market regimes—particularly during strong directional trends or rising volatility. However, during unstable or bearish periods, momentum-based rebalancing performs poorly, and buy-and-hold strategies dominate. This emphasizes the importance of context-aware tuning and suggests that hybrid or ensemble strategies may provide better long-term robustness. Our results also support the idea that daily monitoring (rather than high-frequency) may be a more feasible application for CUSUM-based rebalancing in practice, since the cost of false alarms is amortized over longer periods and transaction costs are less of a concern.

Future directions can (1) extend the detector to multivariate CUSUM using shared factor structures or maximum-statistics across assets; (2) develop adaptive thresholding techniques that learn or adjust the threshold h in response to recent detection performance or volatility; (3) incorporate more realistic innovation models (e.g., skewed or t-distributed residuals) and estimate changepoints under model uncertainty, and (4) explore hybrid architectures where CUSUM detectors feed into reinforcement learning or other policy optimizers for active trading strategies.

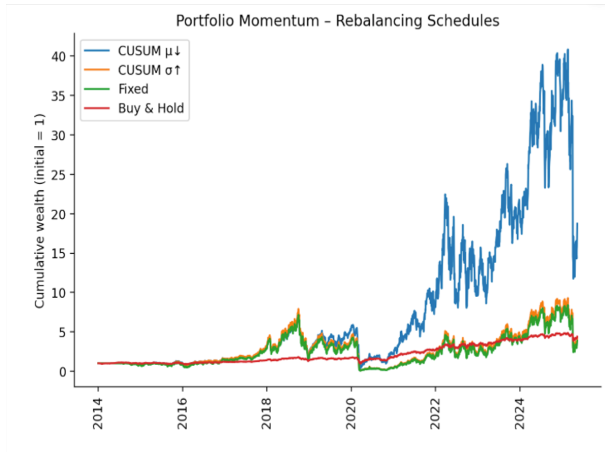

Figure 5: A "winning" example where a CUSUM-based rebalancing beats the fixed and buy-and-hold strategies defined in Section 4. User inputs are the following: 152 days momentum look-back, 21 days for the fixed rebalance interval, top-2 momentum assets, $\delta = 0.4$, $h = 8$, $W = 60$, GARCH pre-whitening, and 0.55% transaction costs. Stock data contains AAPL, GOOGL, AMZN, ORCL, XOM, CVX, COP, HES, and OXY closing prices from January 1, 2014 through May 19, 2025.

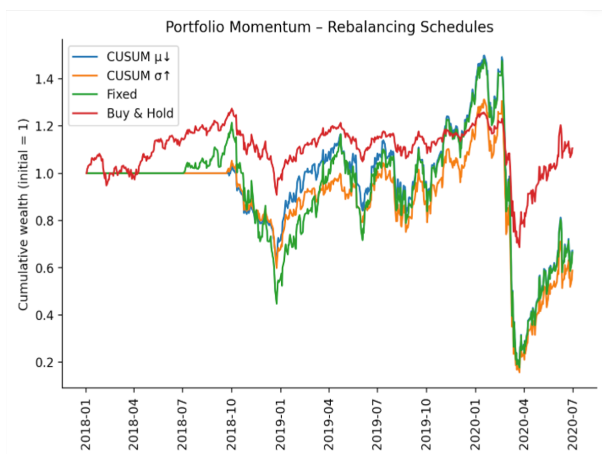

Figure 6: A "losing" example where a CUSUM-based rebalancing underperform compared to the buy-and-hold strategy defined in Section 4. User inputs are the following: 152 days momentum look-back, 21 days for the fixed rebalance interval, top-2 momentum assets, $\delta = 0.4$, $h = 8$, $W = 60$, GARCH pre-whitening, and 0.55% transaction costs. Stock data contains AAPL, GOOGL, AMZN, ORCL, XOM, CVX, COP, HES, and OXY closing prices from January 1, 2018 through May 19, 2020.

References

[1] Sam Astill, David I. Harvey, Stephen J. Leybourne, A. M. Robert Taylor, and Y. Zu. Cusum-based monitoring for explosive episodes in financial data in the presence of time-varying volatility. *Journal of Financial Econometrics*, 21(1):187–227, 2023.

[2] Michèle Basseville and Igor V. Nikiforov. *Detection of Abrupt Changes: Theory and Application*. Prentice Hall, 1993.

[3] R. L. Brown, J. Durbin, and J. M. Evans. Techniques for testing the constancy of regression relationships over time. *Journal of the Royal Statistical Society: Series B (Methodological)*, 37(2):149–192, 1975.

[4] Michael Carlisle, Olympia Hadjiliadis, and Ioannis Stamos. Trends and trades: A cusum-based high frequency trading strategy. AMS Meeting, UTEP, September 12-13, 2020, 2020.

[5] Peter Carr and Liuren Wu. Option profit and loss attribution and pricing: A new framework. *The Journal of Finance*, 75(4):1913–1963, 2020.

[6] Ciprian M. Crainiceanu and Timothy J. Vogelsang. Nonmonotonic power for tests of a mean shift in a time series. *Journal of Statistical Computation and Simulation*, 2007.

[7] Veronica Montes de Oca and George Kesidis. A cusum change-point detection algorithm for non-stationary sequences with application to data network surveillance. *Journal of Statistical Software*, 36(3):1–15, 2010.

[8] Manh Cuong Dong, Cathy W. S. Chen, Sangyeol Lee, and Songsak Sriboonchitta. Change-points from garch residuals using the cusum test. *Computational Economics*, 54(1):189–210, 2019.

[9] Lajos Horváth, Piotr Kokoszka, and Gergely Rácz. Sequential change-point detection in garch models. *arXiv preprint arXiv:2404.17885*, 2024.

[10] Tze Leung Lai. Sequential change-point detection in quality control and dynamical systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 57(4):613–658, 1995.

[11] Gary Lorden. Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42(6):1897–1908, 1971.

[12] Robert Malinas, Dogyoon Song, Benjamin D. Robinson, and Alfred O. Hero III. High-dimensional sequential change detection. *arXiv preprint arXiv:2502.05377v1*, 2025.

[13] Yajun Mei. Efficient scalable schemes for monitoring a large number of data streams. *Biometrika*, 97(2):419–433, 2010.

[14] G. V. Moustakides. Optimal stopping times for detecting changes in distributions. *The Annals of Statistics*, 14(4):1379–1387, 1986.

[15] Daniel B. Nelson. Filtering and forecasting with misspecified arch models i: Getting the right variance with the wrong model. *Journal of Econometrics*, 52(1-2):61–90, 1992.

[16] Peter Nystrup, Sebastian M. Lindstrøm, Sebastian H. Madsen, and Jesper Andreasen. Dynamic allocation or diversification: A regime-based approach to multiple assets. *Journal of Risk and Financial Management*, 13(12):311, 2020.

[17] E. S. Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

[18] Yue Qiang and Salvatore Ruggieri. Autocorrelation and parameter estimation in a bayesian change point model. *Mathematics*, 11(5):1082, 2023.

[19] Y. Ritov. Decision theoretic optimality of the cusum procedure. *The Annals of Statistics*, 18(3):1464–1469, 1990.

[20] Michal Shauly-Aharonov and Orit Barenholz-Goultschin. Real-time change-point detection algorithm with an application to glycemic control for diabetic pregnant women. *Methodology and Computing in Applied Probability*, 2019.

[21] A. N. Shiryaev. Minimax optimality of the method of cumulative sums (cusum) in the case of continuous time. *Russian Mathematical Surveys*, 51(4):750–751, 1996.

[22] Alexander G. Tartakovsky, Igor V. Nikiforov, and Michèle Basseville. Sequential analysis: Hypothesis testing and changepoint detection. *CRC Press*, 2014.

[23] Wing-Keung Wong and Li Li. The consequences of using garch filters for non-garch data. *Journal of Empirical Finance*, 16(5):811–815, 2009.