Master Project

submitted to the faculty of The University of Utah in partial fulfillment of the requirements for the degree of Master of Statistics Department of Mathematics

Change Point Detection with Applications

Author:
Han Ambrose

Supervisory Committee:
Lajos Horváth (Committee Chair)
Fernando Guevara Vasquez
Jingyi Zhu

March, 2021

Contents

CHANGE POINT DETECTION WITH APPLICATIONS

Abstract. Change points detection has been widely used to detect anomolies or abrupt changes in time series data. Such abrupt changes may represent transitions that occur between states. Detection of change points is useful in modeling and prediction of time series and is found in application many areas. This paper examines several methods that detect change points in time series. The methods examined include segmentation using CUSUM and weighed CUSUM with independent errors and dependent errors. Further, we applied this method to detect changes in financial data.

Date: March, 2021.

1. Introduction

It is unlikely that the behavior of time series would be the same for a longer periods. Due to governmental inventions, different economic condition, natural disasters the behaviour of the time series might change but the time is usually unknown. It is important to check the stability of our model. Neglecting changes and use our data as a stationary sequence could cause misleading conclusions and false predictions. We wish to investigate how to segment a data set.

Detecting changes in the data is also in finding if there is any changes in the mean of the dataset. We established a null hypothesis that the mean is constant or no changes detected. If we reject the null hypothesis, it means that the mean changes and the data is non-stationary. Every time we find a change, we cut dataset into segments and repeat the process. What we get is a returns of locations of changes where anomalies are found. This is done using the CUSUM process.

This paper is structured as introducing segmentation using CUSUM process with independent errors. However, this method is not complete and it includes more change points than it should. We then evaluated the the effectiveness of CUSUM method and provided improvement with weighted CUSUM to exclude false change points. We covered both cases when the variance is known and when variance is unknown. It is easy to set up the variance in the simulation but in application, it is extremely unlikely to know the variance. Hence, in practice this is replaced by using sample variance.

We also examined assumptions on the error terms, in order to use sample variance or long run variance estimator. We assume the error term from the models are uncorrelated and follows a GARCH process. In this case, we use the sample variance instead of the long run covariance estimator.

2. The CUSUM method

We start with a simple example. Let $X_1, X_2, \ldots, X_N$ be independent normal random defined as

$$X_i = \left\{ \begin{array}{ll} 2+\epsilon_i, & \text{if } \ 1 \le i \le \lfloor N/3 \rfloor, \\ 1+\epsilon_i, & \text{if } \ \lfloor N/3 \rfloor + 1 \le i \le \lfloor 2N/3 \rfloor \\ \epsilon_i, & \text{if } \ \lfloor 2N/3 \rfloor + 1 \le i \le N, \end{array} \right\} \tag{1}$$

where $\lfloor x \rfloor$ denotes the integer part of x. We assume that $\epsilon_1, \epsilon_2, \ldots, \epsilon_N$ are independent and identically distributed normal random variables with mean 0 and variance σ^2. According to our model we start with mean 2, this changes to 1 at $\lfloor N/3 \rfloor + 1$ and to 0 at $\lfloor 2N/3 \rfloor + 1$. So we have exactly two changes at $\lfloor N/3 \rfloor + 1$ and $\lfloor 2N/3 \rfloor + 1$. We wish to estimate the number of changes in the sequence. The testing and estimation method based on the CUSUM sequence

$$T(k) = \sum_{i=1}^{k} X_i - \frac{k}{N} \sum_{i=1}^{N} X_i, \quad 1 \le k \le N.$$

We reject the no change in the mean null hypothesis if

$$T_N = \frac{1}{\sigma} N^{-1/2} \max_{1 \le k \le N} |T(k)|,$$

where σ is a scaling parameter. If the observations are independent, then σ^2 is usually the variance of the observations while in the dependent case σ^2 is the long run variance. If the no change null hypothesis is true, then under minor conditions on the X_i's (cf. Aue and Horváth 2013 and Horváth and Rice 2014)

$$T_N \xrightarrow{\mathcal{D}} \sup_{0 \le t \le 1} |B(t)|,$$

where $\{B(t), 0 \le t \le 1\}$ is a Brownian bridge. The convergence in distribution is denoted by $\xrightarrow{\mathcal{D}}$.

Let $c(\alpha)$ the critical value for the supremum of the absolute value of a Brownian bridge, i.e.

$$P\left\{ \sup_{0 \le t \le 1} |B(t)| \ge c(\alpha) \right\} = \alpha.$$

The process $\{B(t), 0 \le t \le 1\}$ is a Brownian bridge if it is Gaussian, i.e. for all $t_1, t_2, \ldots, t_L$ the distribution of the vector $(B(t_1), B(t_2), \ldots, B(t_L))$ is L dimensional normal and $EB(t) = 0$, $EB(t)B(s) = (\min(t,s) - ts)$. These critical values $c(\alpha)$ are widely available, since these are the asymptotic values for the classical Kolmogorov–Smirnov statistic.

Figures 1–5 show the graphs of the absolute value of some random generated CUSUM sequences in the model (1) when the sample size is $N = 100$ and the variance is chosen as $\sigma^2 = .1$. In this case the times of changes are $k_1 = 33$ and $k_2 = 66$. Figures 1–5 also identify the times of changes as well. In all cases the graphs show that the sequences are dominated by the expected values of the random variables. The value of T_N is stable in all cases but the location of the maximum varied widely. On In Figures 1–3 we obtained 61, 35 and 65 as the estimators for one of the possible changes. These values are close to the true values of the times of the changes in the mean. However, Figures 4 and 5 indicate a completely different pattern. The estimated times of changes are 59 and 48 which are too far from the true values of k_1 and k_2. The functions in Figures 1–5 exhibits the pattern that the flat part of the mean of the observations dominate the random part. By definition, if the maximum is reached at two or more different

values, we take the smallest value as the time of change, so we "push" our estimator close to k_1. But we find values which are not close to any of the real times of changes.

FIGURE 1. A realization of the absolute value of the CUSUM process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.

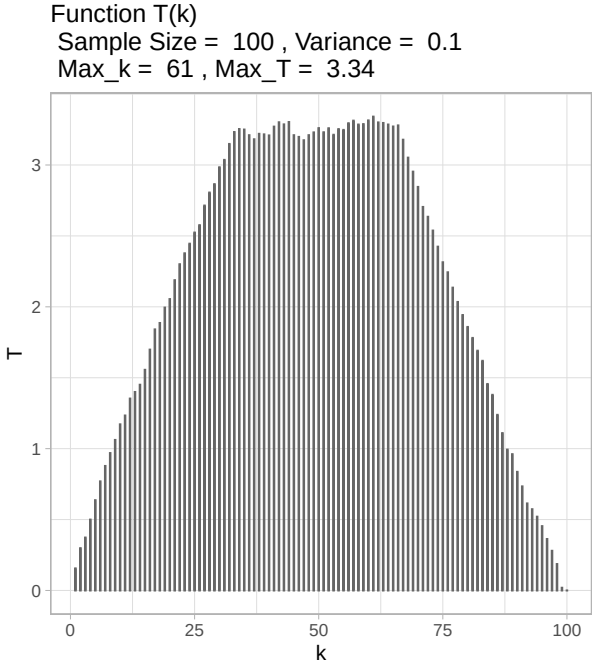


FIGURE 2. A realization of the absolute value of the CUSUM process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.

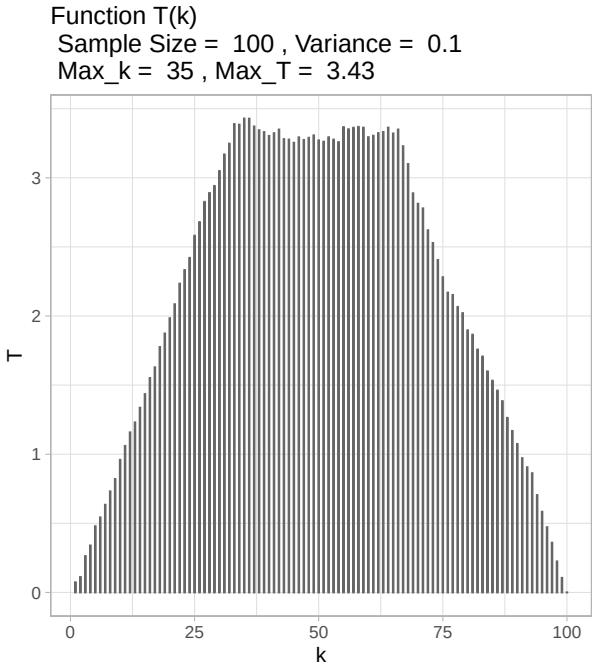


FIGURE 3. A realization of the absolute value of the CUSUM process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.

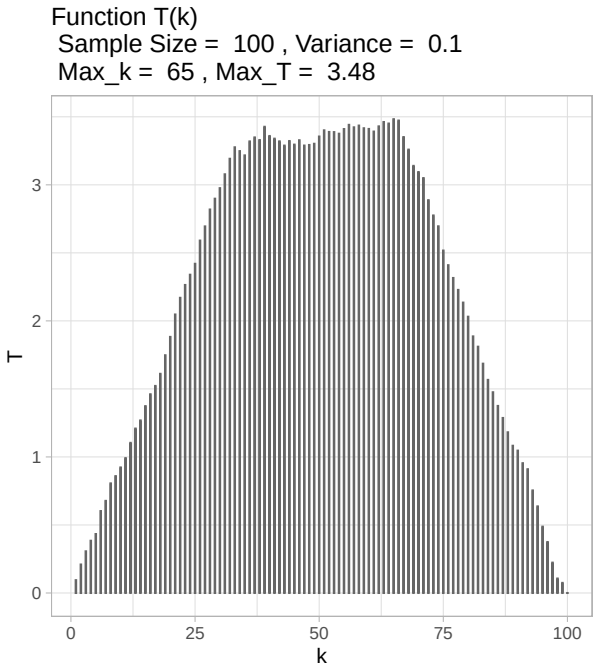


FIGURE 4. A realization of the absolute value of the CUSUM process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.

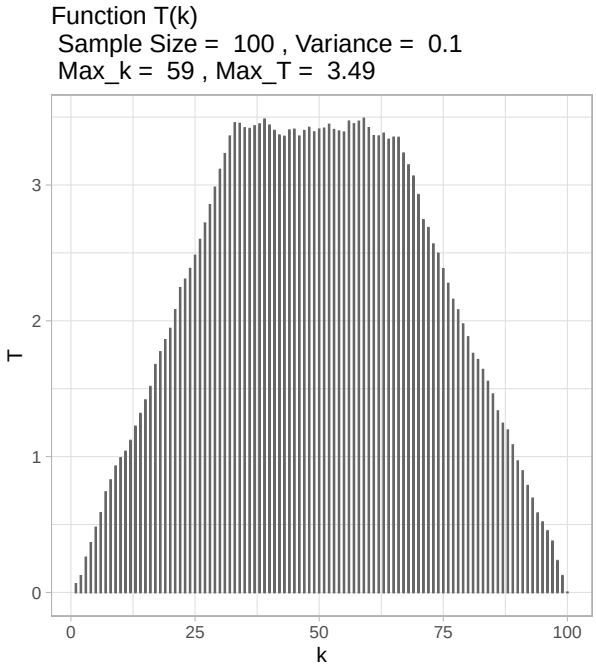
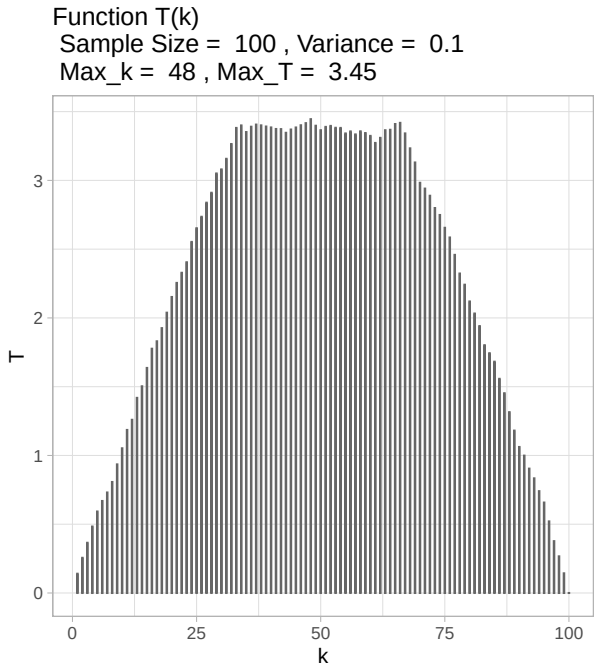


FIGURE 5. A realization of the absolute value of the CUSUM process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.



To find all the changes we suggest the binary segmentation procedure: we look at the function

$$T^{(1)}(k, X_1, X_2, \ldots, X_N)) = \frac{1}{N^{1/2}} \left| \sum_{i=1}^{k} X_i - \frac{k}{N} \sum_{i=1}^{N} X_i \right|$$

and we check if

$$\max_{1 \le k \le N} T(k, X_1, X_2, \ldots, X_N)) \ge \sigma c(\alpha).$$

If this inequality is true then the sequence changed its mean and the location of the maximum is our estimator for the time of change. This is denoted by $\bar{k}_1$. If the maximum is reached at more than one point, take the smallest one. Cut the data into to two subsets $X_1, X_2, \ldots, X_{\bar{k}_1}$ and $X_{\bar{k}_1+1}, X_{\bar{k}_1+2}, \ldots, X_N$ and again we compute the CUSUM sequence from each subset

$$T^{(1,1)}(k, X_1, X_2, \ldots, X_{\bar{k}_1}) = \frac{1}{\bar{k}_1^{1/2}} \left| \sum_{i=1}^{k} X_i - \frac{k}{\bar{k}_1} \sum_{i=1}^{\bar{k}_1} X_i \right|.$$

If the maximum of this sequence is less than $c(\alpha)$, then there is no change in this subset and nothing more is done with this subsequence. If it is above the critical value we found a change, and the location of the maximum $\bar{k}_{1,1}$. Now we cut the subset $X_1, X_2, \ldots, X_{\bar{k}_1}$ into two subsets at $\bar{k}_{1,1}$ and continue the segmentation on each subset.
From $X_{\bar{k}_1+1}, X_{\bar{k}_1+2}, \ldots, X_N$ we compute

$$T^{(1,2)}(k, X_{\bar{k}_1+1}, X_{\bar{k}_1+2}, \ldots, X_N) = \frac{1}{(N - \bar{k}_1)^{1/2}} \left| \sum_{i=\bar{k}_1+1}^{k} X_i - \frac{k - \bar{k}_1 + 1}{N - \bar{k}_1} \sum_{i=\bar{k}_1+1}^{N} X_i \right|.$$

If the maximum of this sequence is less than $\sigma c(\alpha)$, then there is no change in this subset and nothing more is done with this subsequence. If it is above the critical value we found a change, and the location of the maximum $\bar{k}_{1,2}$. We cut $X_{\bar{k}_1+1}, X_{\bar{k}_1+2}, \ldots, X_N$ into two subsets at $\bar{k}_{1,2}$

and we continue the segmentation.
Hence we have the times of changes $\bar{k}_1$, $\bar{k}_{1,1}$, $\bar{k}_{1,2}, \ldots$. If there is no change, the location of the time of change is zero (this is when we stop at the first step). We repeated this experiment M times and we got M vectors of the times of changes.
At this point we divide the data into two subsets. We call them left and right. We compute the max CUSUM for the left side. We create a vector: If the max CUSUM is below the critical level, the first coordinate is 1 and the second is 0. If the max CUSUM is above the critical level, the first coordinate is 0 and the second coordinate is the location of the maximum. If this is repeated $M = 200$ times, we have the vectors $v_{\text{left},i} = (v_{\text{left},i,1}, v_{\text{left},i,2})^\top, 1 \leq i \leq M$. Since we expect two changes we organized the outcome of the simulations in the following way: We compute

$$\bar{v}_{\text{left},1} = \frac{1}{M} \sum_{i=1}^{M} v_{\text{left},i,1}$$

and

$$\bar{v}_{\text{left},2} = \frac{1}{M - M\bar{v}_{\text{left},1}} \sum_{i=1}^{M} v_{\text{left},i,2}$$

With $\bar{v}_{\text{left},1}$ we know the probability if in the second step there is a change on the left side, and with $\bar{v}_{\text{left},2}$ is the location of the change if change is found. We need to do this on the right side of the first change as well.
We computed the histogram of the times of changes and we also obtained a smooth estimator for the distribution of the times of changes. The results are shown in Figures 9–11. In all cases we observe that the chance that the maximum is reached on the "flat" part of the CUSUM curve does not go to zero when the sample size N gets larger. Essentially, the distribution of the time of change is the following: the estimators concentrate around k_1 and k_2, as it is expected, but a uniform distribution between k_1 and k_2 is also part of the limiting behaviour. This holds for all sample sizes N, if σ^2 is small. In this case the "drift term", i.e. the expected value of $T(k)$ dominates the behaviour of $|T(k)|, 1 \leq k \leq N$. Figure 12 shows the frequency how many times we find k_1 and k_2 in the first step.

FIGURE 6. The distribution of the estimated times of changes.

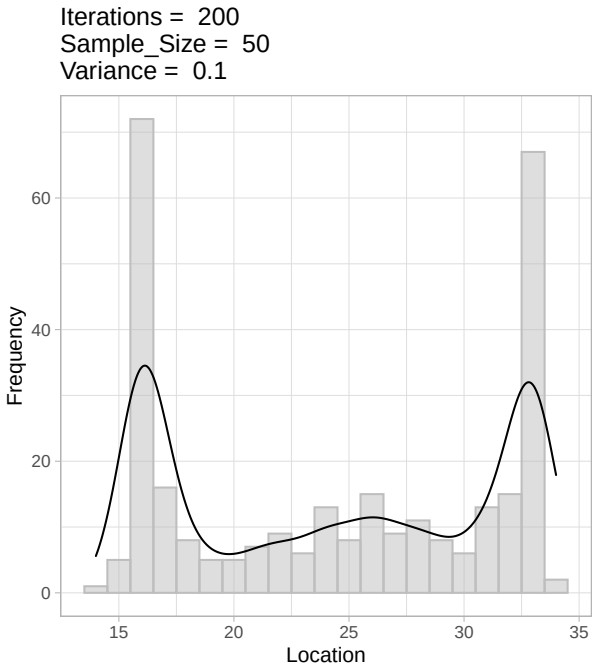


FIGURE 7. The distribution of the estimated times of changes.

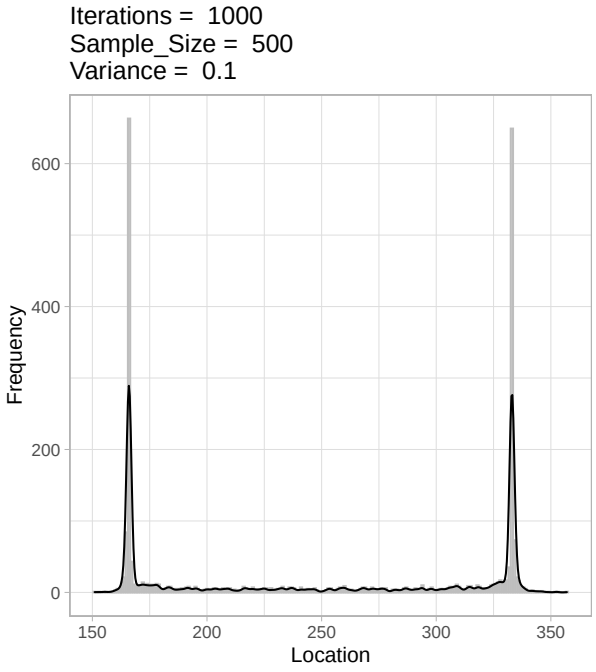


FIGURE 8. The distribution of the estimated times of changes.

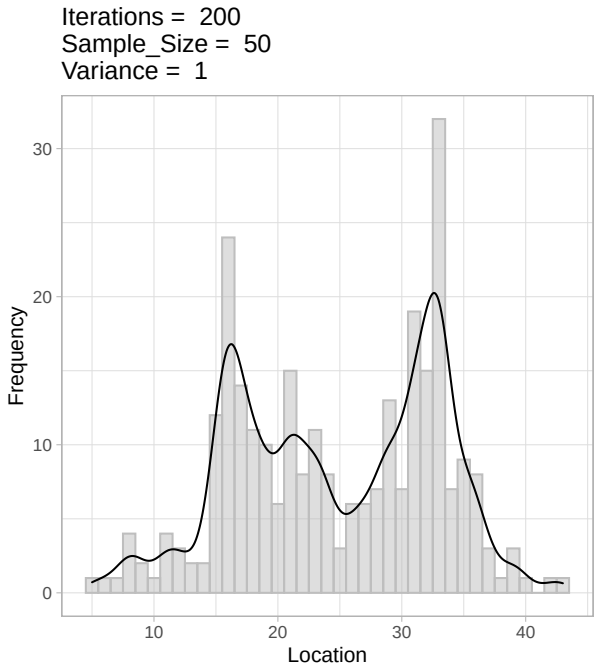


FIGURE 9. The distribution of the estimated times of changes.

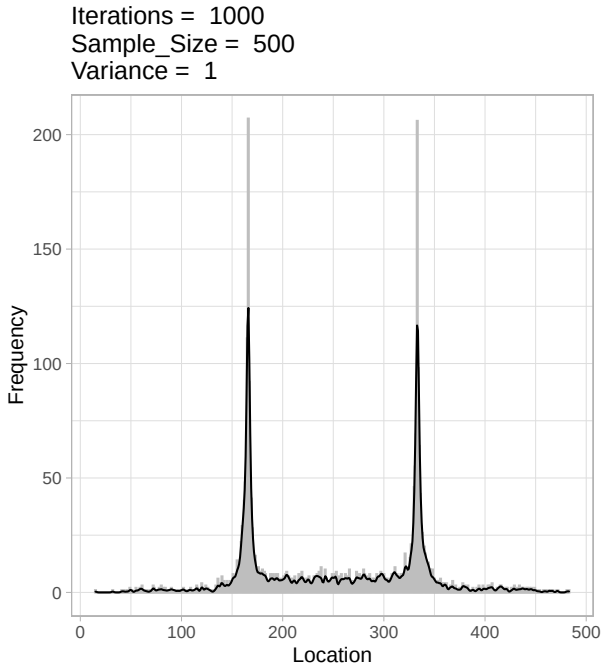


FIGURE 10. The distribution of the estimated times of changes.

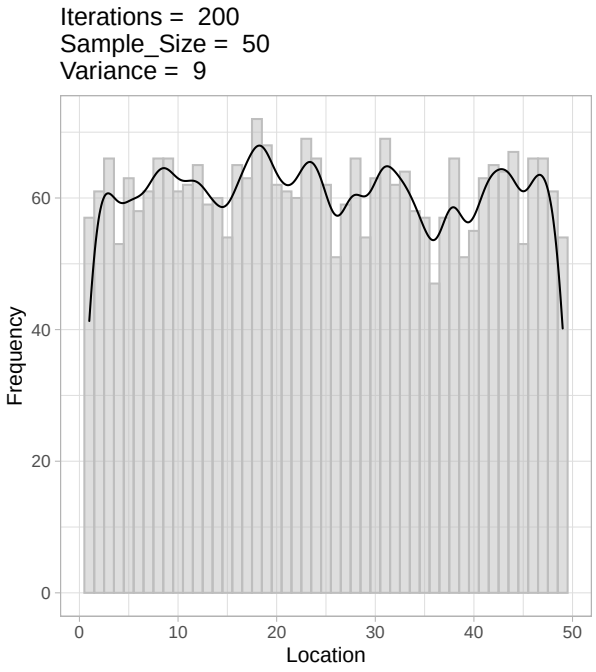


FIGURE 11. The distribution of the estimated times of changes.

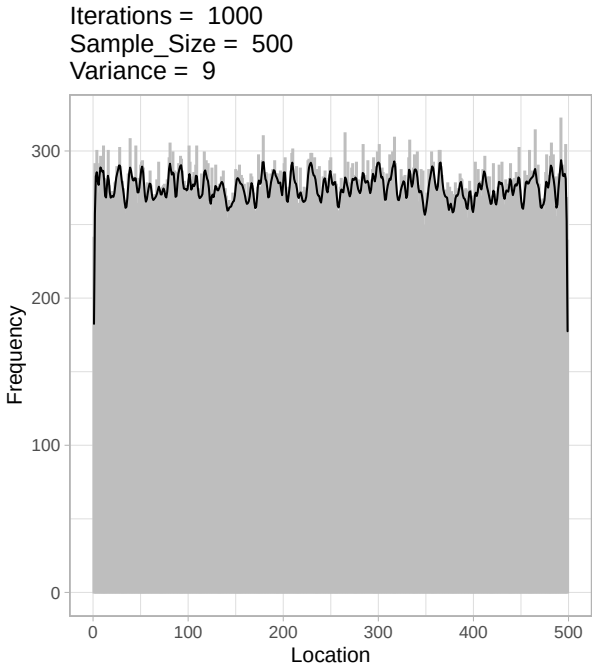
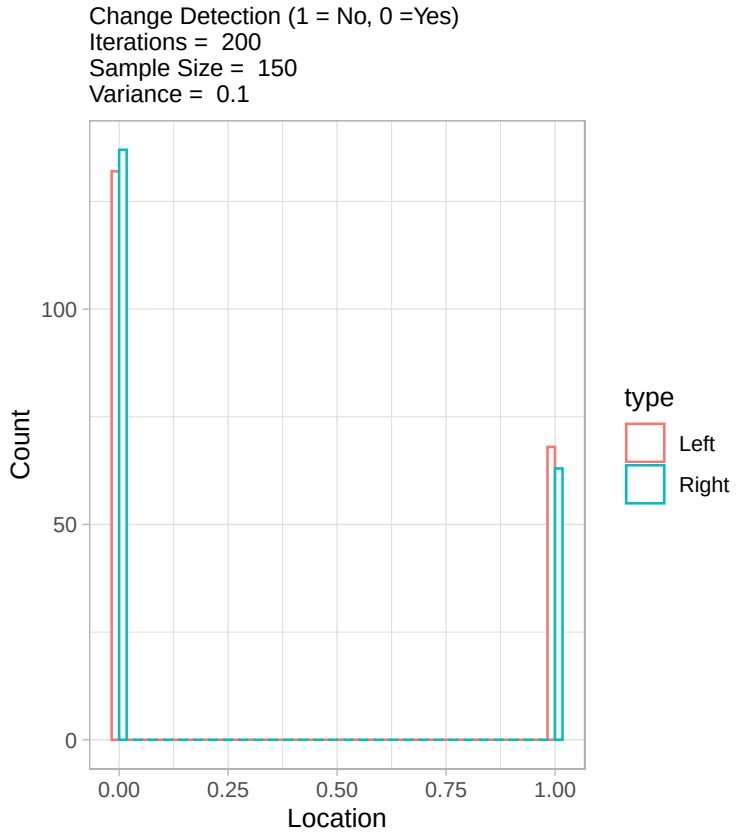


FIGURE 12. The distribution of the first and second estimated times of changes.



Figures 9–11 confirm that we can find a "change point" which are not really of the change points, essentially several points between $\lfloor N/3 \rfloor$ and $\lfloor 2N/3 \rfloor$ were declared to be change points but no change occurred there. Hence the CUSUM finds more change points than we have in the data. Figures 1–5 show the sample path of a typical CUSUM sequence and they prove that the issue is the flat part of $|T(k), 1 \leq k \leq N$. The probability that the change occurs between $\lfloor N/3 \rfloor$ and $\lfloor 2N/3$ looks uniform and it is not negligible even for large sample sizes.

One suggestion is to use the weighted maximum:

$$T_N(\kappa) = \frac{1}{\sigma} \max_{1 \leq k < N} \frac{N^{-1/2}}{((k/N)(1-(k/N)))^{\kappa}} \left| \sum_{i=1}^{k} X_i - \frac{k}{N} \sum_{i=1}^{N} X_i \right|,$$

$0 \leq \kappa \leq 1/2$, where σ is scaling as before. It is shown in Csörgő and Horváth (1997) that if $\kappa = 1/2$ (standardization) is used, $T_N(1/2)$ has an extreme value limit with unusual centralization and normalization and the rate of convergence is extremely slow. The limit distribution is not known, not even for large N except when $\kappa = 0$ and $\kappa = 1/2$. It is shown in Csörgő and Horváth (1997) that the rate of convergence is slow when $\kappa = 1/2$ and much better when $\kappa < 1/2$. Our example shows that $\kappa = 0$ introduces points in the binary segmentation which are not change points. However, Horváth et al. (1997) proves that this cannot happen when $\kappa > 0$. Hence we are interested in the behaviour of $T_N(kappa)$ when $0 < \kappa < 1/2$ is used in the binary segmentation.

There is no formula for $T_N(\kappa)$ nor for its limit

$$T(\kappa) = \sup_{0<t<1} \frac{1}{(t(1-t))^\kappa}|B(t)|,$$

where $B(t)$ is a Brownian bridge. To do the testing step, we need to find $c_\kappa(\alpha)$ such that

$$P\left\{\sup_{0<t<1} \frac{|B(t)|}{(t(1-t))^\kappa} \le c_\kappa(\alpha)\right\} = 1-\alpha,$$

If $\epsilon_1, \epsilon_2, \ldots, \epsilon_N$ are iid standard normal random variables, then

$$T_N(\kappa) = \max_{1\le k<N} \frac{N^{-1/2}}{((k/N)(1-k/N))^\kappa}\left|\sum_{i=1}^{k}\epsilon_i - \frac{k}{N}\sum_{i=1}^{N}\epsilon_i\right| \xrightarrow{\mathcal{D}} \sup_{0<t<1}\frac{|B(t)|}{(t(1-t))^\kappa}. \tag{2}$$

Now we can use simulations to get $c_\kappa(\alpha)$. Note that in this case

$$T_N \stackrel{\mathcal{D}}{=} \max_{1\le k<N} \frac{1}{((k/N)(1-k/N))^\kappa}|B(k/N)|,$$

so T_N can be considered as discreet version of $|B(t)|$. Using the modulus of continuity of the Brownian bridge one can show (cf. Csörgő and Horváth (1997) that

$$|P\{T_N \le x\} - P\{T \le x\}| = O_P\left(N^{-1/2+\kappa}(\log N)^{1/2}\right).$$

We simulated $M = 500$ independent copies of $T_N(\kappa)$, and computed the empirical distribution of the generated copies of $T_N(\kappa)$ when $N = 500$. Thus we got $c_\kappa(\alpha)$ for $\alpha = 0.0, 0.1, 0.05, 0.001$ and $\kappa = 0, 0.1, 0.2, \ldots, 0.49$. The results are given in Table 1 The simulation result for $\kappa = 0$ is compared to the known values of $\kappa_0(\alpha)$.
Figures 13–14 display some of the empirical distribution functions used in the simulations of the critical values of $T(\kappa)$.

TABLE 1. Selected critical values for $T(\kappa)$.

α	Empirical Critical Values	Theoretical Critical Values
$\kappa = 0$		
0.01	1.604	1.628
0.05	1.329	1.358
0.1	1.190	1.224
$\kappa = 0.1$		
0.01	1.843	
0.05	1.552	
0.1	1.410	
$\kappa = 0.3$		
0.01	2.617	
0.05	2.146	
0.1	1.947	
$\kappa = 0.45$		
0.01	3.260	
0.05	2.755	
0.1	2.596	

FIGURE 13. The empirical distribution of $T(0)$ based on Monte Carlo simulations.

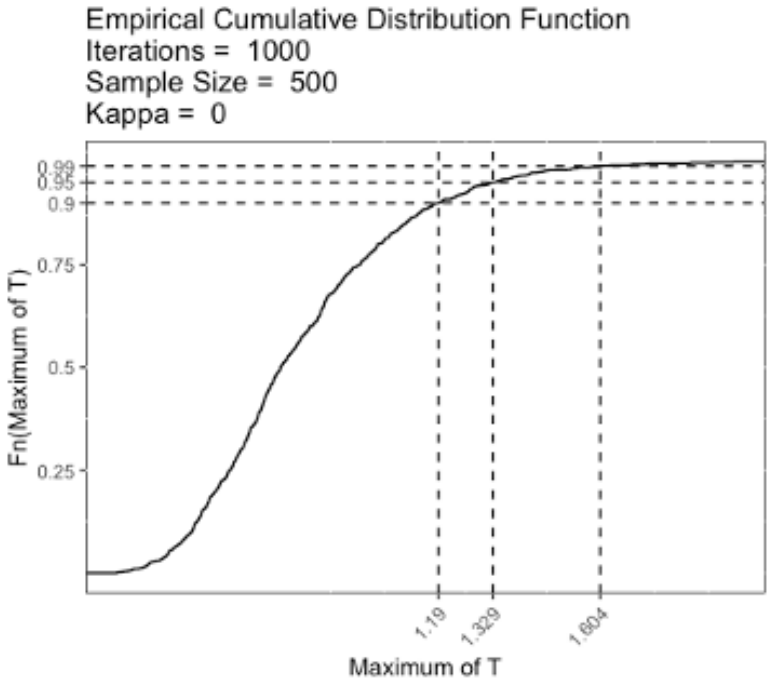


FIGURE 14. The empirical distribution of $T(0.1)$ based on Monte Carlo simulations.

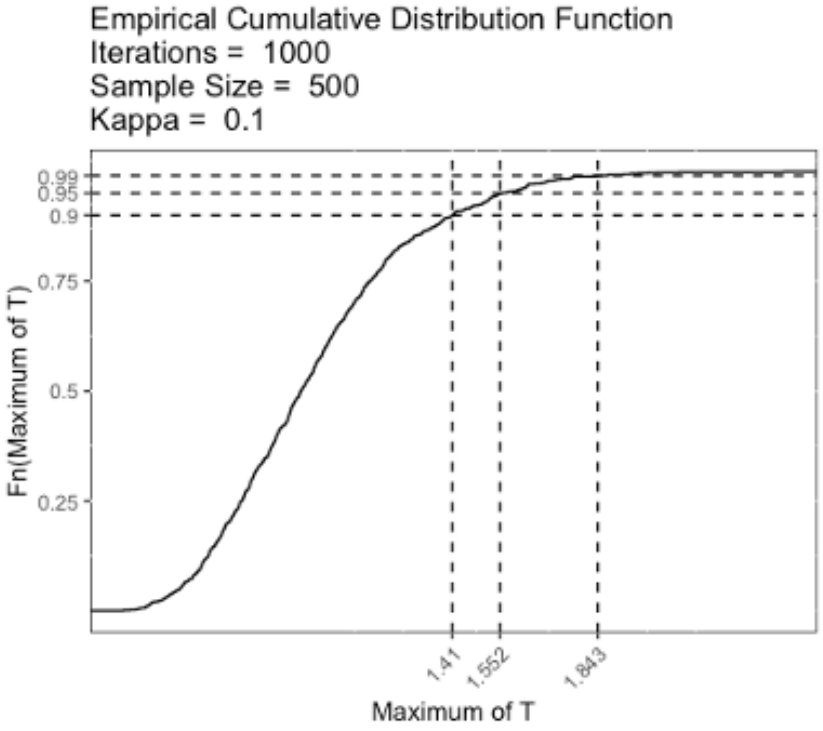


FIGURE 15. The empirical distribution of $T(0.3)$ based on Monte Carlo simulations.

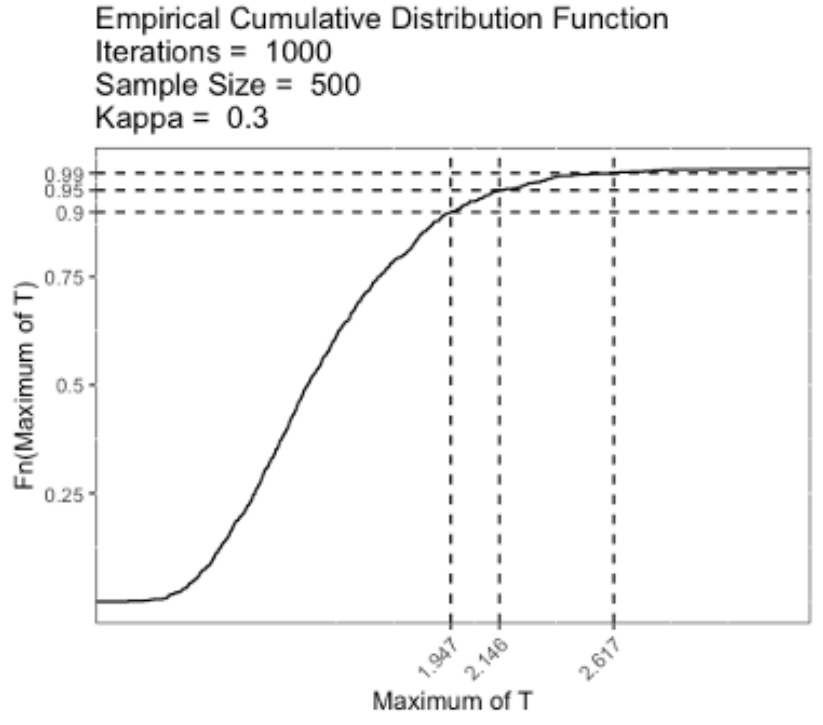
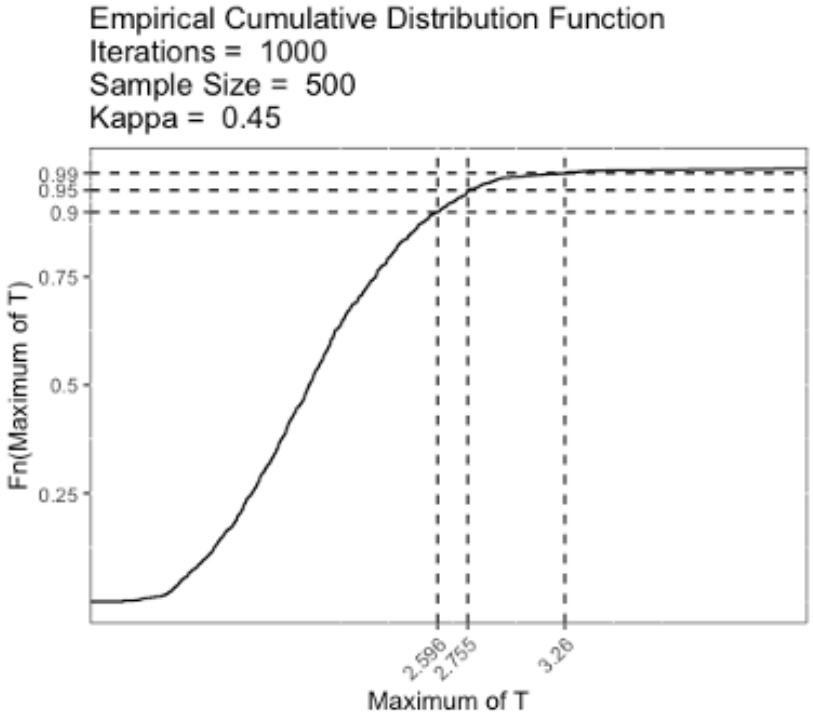


FIGURE 16. The empirical distribution of $T(0.45)$ based on Monte Carlo simulations.



3. Change point estimation based on weighted CUSUM statistics with independent errors

We used the model of (1) and the errors are still independent identically distributed normal random variables with zero mean and variance σ^2. For comparison in Figures 17 and 18 we display a typical sample path of the CUSUM process when weight function is used. The shapes of the realizations confirms the theoretical result that the "flat" of the unweighted CUSUM disappears. Using larger weight, the largest value of the weighted CUSUM is closer to and sharper at the change point.

FIGURE 17. A realization of the absolute value of the weighted CUSUM $\kappa = .1$ process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.

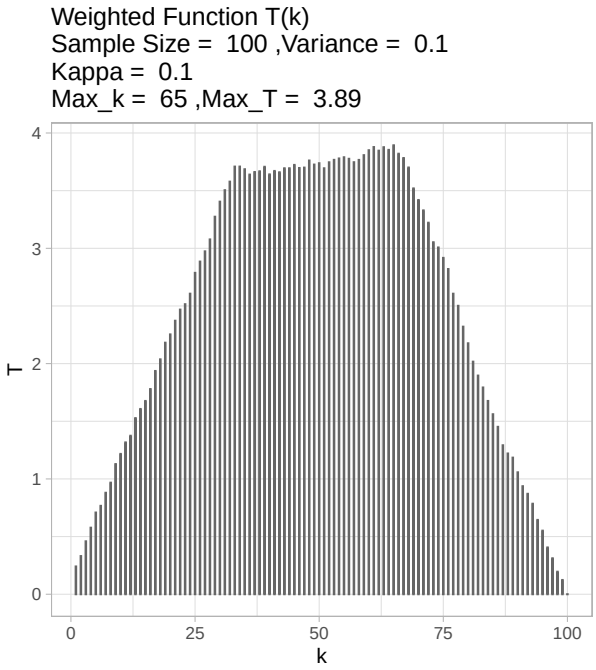
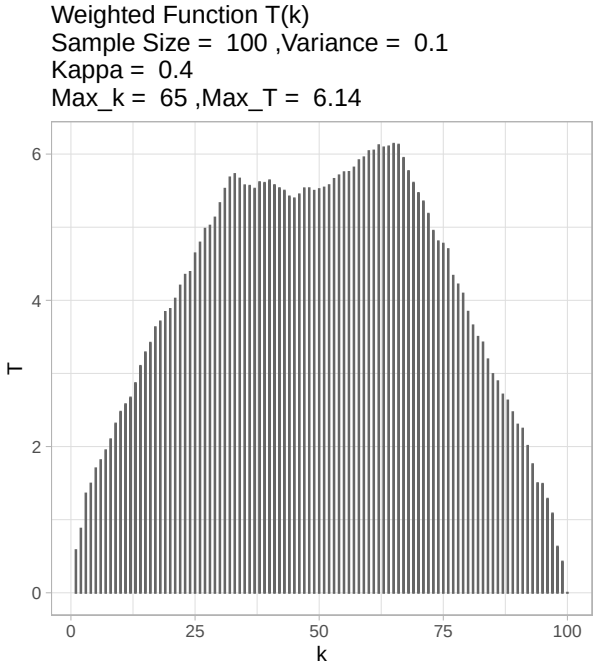


FIGURE 18. A realization of the absolute value of the CUSUM $\kappa = .4$ process in the model (1) with $N = 100$ and $\sigma^2 = 0.1$.



We repeated our experiments but now we used exponential (λ) observations. For the sake of comparison, $\lambda = \sigma^2$, where σ^2 is the variance in the normal case. First we used Monte Carlo

simulations to get the exact critical when exponential distribution is used. The results are in Table together with values obtained from the limit distribution. The results provide a good fit even in case of small sample sizes.

TABLE 2. Selected critical values for $T(\kappa)$ in case of exponential observations compared to values from the limit distribution.

$\kappa = 0$		
0.01	1.604	1.698
0.05	1.329	1.332
0.1	1.190	1.212
$\kappa = 0.1$		
0.01	1.843	1.895
0.05	1.552	1.602
0.1	1.410	1.421
$\kappa = 0.3$		
0.01	2.617	2.548
0.05	2.146	2.132
0.1	1.947	1.933
$\kappa = 0.45$		
0.01	3.260	3.877
0.05	2.755	3.067
0.1	2.596	2.733

We also repeated our experiments when there is exactly to changes in the observations. Figures 19–21 show the distribution of the estimated time of change found first. The results indicate a clear pattern. If $\kappa = 0$, i.e. no weight is used, our estimates are not change points with relatively large probabilities, and they can be anywhere between the true times of changes. If $\kappa = 0.4$, then there are changes found at the end of the data. It is shown in the figure below that if there is no change in the data, then the location of the maximum is at the beginning or at the end of the data with probability $1/2$ if $\kappa = 0$. Due to the heavy weight when $\kappa = 0.4$ is used we find artificial changes close to the end.

FIGURE 19. The distribution of the time of change found in the first step in case of exponential $\lambda = 0.1$ observations when $\kappa = 0$.

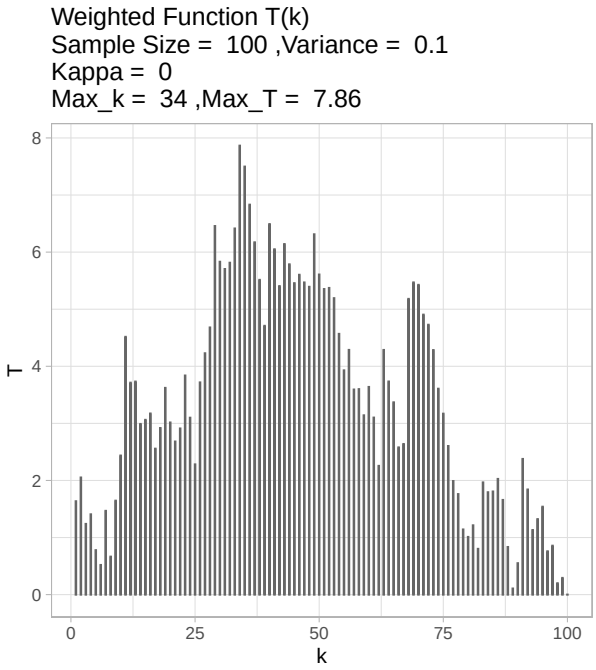


FIGURE 20. The distribution of the time of change found in the first step in case of exponential $\lambda = 0.1$ observations when $\kappa = 0.1$.

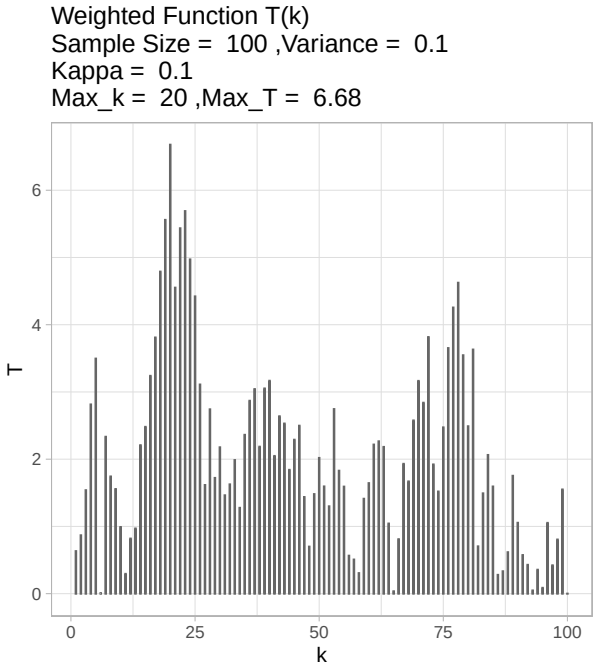
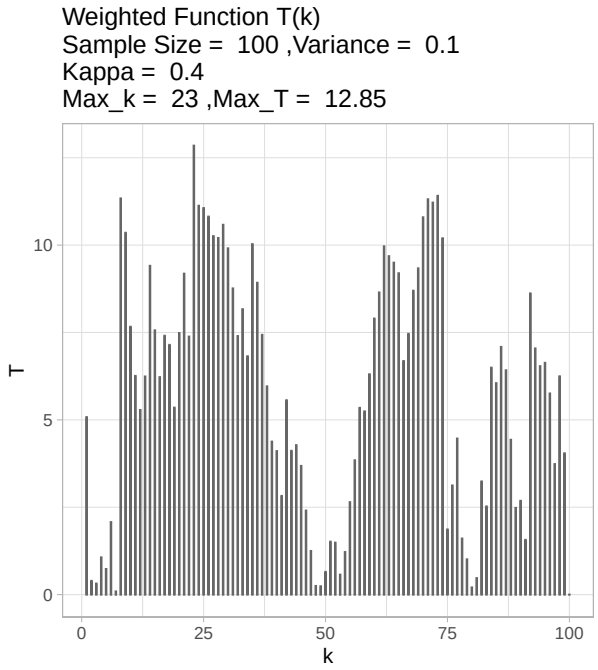


FIGURE 21. The distribution of the time of change found in the first step in case of exponential $\lambda = 0.1$ observations when $\kappa = 0.4$.



In the simulations we assumed that the variance is known which is extremely unlikely in application. In case of independent observations the usual estimator for the unknown variance is the sample variance

$$S_N^2 = \frac{1}{N-1}\sum_{i=1}^{N}(X_i - \bar{X}_N)^2,$$

where

$$\bar{X}_N = \frac{1}{N} = \sum_{i=1}^{N} X_i$$

is the sample mean. Using the weak law of large numbers (cf. DasGupta 2008)

$$S_N^2 \xrightarrow{P} \sigma^2,$$

where $\sigma^2 = \mathrm{var}(X_1)$ is the variance of the observations. Under the alternative

$$S_N^2 \xrightarrow{P} \tau^2,$$

where $\tau^2 > 0$. This means that the estimation of σ^2 does not change the validity of the test under the null nor under the alternative. We want to know that the estimation of σ^2 is only needed in the first step of the procedure, when we test if there is a change in the mean. We have that

$$\frac{1}{S_N^{1/2}} \max_{1\le k\le N} \left(\frac{N^2}{k(N-k)}\right)^{\kappa} \left|\sum_{i=1}^{k}\epsilon_i - \frac{k}{N}\sum_{i=1}^{N}\epsilon_i\right| \xrightarrow{\mathcal{D}} \sup_{0<t<1} \frac{1}{(t(1-t))^{\kappa}}|B(t)|$$

under the null hypothesis and

$$\frac{1}{S_N^{1/2}} \max_{1\le k\le N} \left(\frac{N^2}{k(N-k)}\right)^{\kappa} \left|\sum_{i=1}^{k}\epsilon_i - \frac{k}{N}\sum_{i=1}^{N}\epsilon_i\right| \xrightarrow{P} \infty$$

under the at least one change in the mean alternative. If this null hypothesis is rejected, the location of the maximum does not depend on the estimator of the variance.

4. Change point estimation based on $T_N(\kappa)$ with dependent errors

If $\epsilon_i, -\infty < i < \infty$ is a stationary sequence with $E\epsilon_i = 0$ and

$$\lim_{N\to\infty}\left(\frac{1}{N^{1/2}}\sum_{i=1}^{N}\epsilon_i\right)^2 = \sigma^2 > 0.$$

It is shown in the literature that under minor assumption on the dependence in the stationary sequence,

$$\frac{1}{\sigma N^{1/2}}\max_{1\le k\le N}\left(\frac{N^2}{k(N-k)}\right)^{\kappa}\left|\sum_{i=1}^{k}\epsilon_i - \frac{k}{N}\sum_{i=1}^{N}\epsilon_i\right| \xrightarrow{\mathcal{D}} \sup_{0<t<1}\frac{1}{(t(1-t))^{\kappa}}|B(t)|,$$

where $0 \le \kappa < 1/2$ and $\{B(t), 0 \le t \le 1\}$ is a Brownian bridge, as before. In the dependent case σ^2 is not the variance of the individual observations, but the long run variance of the stationary sequence. Using the sample variance defined before, we might under or overestimate the long run variance. The standard estimator for the long run variance is the kernel estimator. First we define the sample correlations:

$$\hat{\gamma}_\ell = \begin{cases} \dfrac{1}{N-\ell}\displaystyle\sum_{i=1}^{N-\ell}(X_i - \bar{X}_N)(X_{i+\ell} - \bar{X}_N), & \text{if} \quad \ell \ge 0, \\ \dfrac{1}{N-|\ell|}\displaystyle\sum_{i=-(\ell-1)}^{N}(X_i - \bar{X}_N)(X_{i+\ell} - \bar{X}_N), & \text{if} \quad \ell < 0. \end{cases}$$

Now the long run variance estimator is

$$\hat{\sigma}_N^2 = \sum_{\ell=-(N-1)}^{N-1} K\left(\frac{\ell}{h}\right)\hat{\gamma}_\ell,$$

where $K(t)$ is the kernel and h is the smoothing parameter (window). It is usually assumed that

$$\text{the first two derivatives of } \ K \ \text{ are bounded}$$

and

$$K(0) = 1.$$

The smoothing parameter is a function of the sample size N and

$$h = h(N) \to \infty \quad \text{and} \quad h/N \to 0.$$

If the null hypothesis holds, then

$$\hat{\sigma}_N^2 \xrightarrow{P} \sigma^2.$$

Hence under the null hypothesis

$$\frac{1}{\hat{\sigma}_N N^{1/2}}\max_{1\le k\le N}\left(\frac{N^2}{k(N-k)}\right)^{\kappa}\left|\sum_{i=1}^{k}\epsilon_i - \frac{k}{N}\sum_{i=1}^{N}\epsilon_i\right| \xrightarrow{\mathcal{D}} \sup_{0<t<1}\frac{1}{(t(1-t))^{\kappa}}|B(t)|.$$

However, the behaviour of σ_N^2 is different from the independent case since under the alternative

$$\frac{\hat{\sigma}_N^2}{h} \xrightarrow{P} \tau^2 > 0.$$

This means that the power of the test will be seriously reduced if h is too large. It is known that if the errors are from an ARMA(p, q), then the errors are correlated. The ARMA(p, q)

sequence is the stationary solution of

$$\epsilon_i = \sum_{\ell=1}^{p} \alpha_i y_{i-\ell} + \eta_i + \sum_{\ell=1}^{q} \eta_{i-\ell}, \quad -\infty < i < \infty,$$

where $\eta_i, -\infty < i < \infty$ are uncorrelated variables with $E\eta_i = 0$ and $E\eta_i^2 = E\eta_j^2, -\infty < i, j < \infty$. The stationary solution can be represented as a linear process of the innovations $\eta_i, -\infty < i < \infty$. In principle, if we know that the errors are from ARMA(p, q) processes, than simpler estimator using the properties of ARMA(p, q) sequences. However, this methods would require the knowledge of several parameters of ARMA(p, q). Also, to have a reasonable fit with ARMA(p, q) we might need relatively large p and q. So we might prefer to kernel long run variance estimator discussed above.

In financial applications linear processes, including ARMA(p, q) sequences, are rarely used. Usually volatility processes are preferred. The most popular one is GARCH (generalized autoregressive conditionally heteroscedastic) process. If $\epsilon_i, -\infty < i < \infty$ is a GARCH(1,1) sequence, it is the solution of

$$\epsilon_i = \sigma_i \eta_i \quad \text{and} \quad \sigma_i^2 = \omega + \alpha \epsilon_{i-1}^2 + \beta \sigma_{i-1}^2, \quad -\infty < i\infty,$$

where $\epsilon_i, -\infty < i < \infty$ are independent and identically distributed random variables with $E\eta_i = 0$, $E\eta_i^2 = 1$, $\omega > 0, \alpha \geq 0, \beta \geq 0$. The condition $\omega > 0$ yields that ϵ_i is not degenerate. The GARCH (1,1) equation can be solve explicitly, and the stationary solution is

$$\sigma_k^2 = \omega \left(1 + \sum_{j=1}^{\infty} \prod_{i=1}^{j} (\alpha \eta_{k-i}^2 + \beta) \right), \quad -\infty < k < \infty.$$

By the independence of the η_j's

$$\begin{aligned} E\sigma_k^2 &= \sum_{j=1}^{\infty} E \left(\prod_{i=1}^{j} (\alpha \eta_{k-i}^2 + \beta) \right) \\ &= \sum_{j=1}^{\infty} \prod_{i=1}^{j} E(\alpha \eta_{k-i}^2 + \beta) \\ &= \sum_{j=1}^{\infty} (E(\alpha \eta_0^2 + \beta))^j, \end{aligned}$$

so $E\sigma_0^2 < \infty$ if and only if $E(\alpha \eta_0^2 + \beta) = \alpha + \beta < 1$. In this case, due to stationarity

$$\begin{aligned} E\sigma_i^2 &= \omega + \alpha E\epsilon_{i-1}^2 + \beta E\sigma_{i-1}^2 \\ &= \omega + \alpha E\eta_{i-1}^2 E\sigma_{i-1}^2 + \beta E\sigma_{i-1}^2 \\ &= \omega + (\alpha + \beta) E\sigma_{i-1}^2 \end{aligned}$$

so

$$E\sigma_0^2 = \frac{\omega}{1 - \alpha - \beta}.$$

The number of the moments even in case of standard normal random variables, the number of moments of σ_i^2 depends on α and β. It is easy to see that

$$E\sigma_i^{2\nu} < \infty$$

if and only if

$$E \left(\sum_{j=1}^{\infty} \prod_{i=1}^{j} (\alpha \eta_{k-i}^2 + \beta) \right)^{\nu}.$$

If $\nu > 1$, then by Minkowski's inequality,

$$\left(E\left(\sum_{j=1}^{\infty} \prod_{i=1}^{j} (\alpha\eta_{k-i}^2 + \beta) \right)^{\nu} \right)^{1/\nu} \leq \sum_{j=1}^{\infty} \prod_{i=1}^{j} \left(E(\alpha\eta_{k-i}^2 + \beta)^{\nu} \right)^{1/\nu}$$
$$= \sum_{j=1}^{\infty} \left(E(\alpha\eta_0^2 + \beta)^{\nu} \right)^{j/\nu}.$$

If $E(\alpha\eta_0^2 + \beta)^{\nu} < 1$, then $E\sigma_0^{2\nu} < \infty$. Also,

$$\sum_{j=1}^{\infty} \prod_{i=1}^{j} E(\alpha\eta_{k-i}^2 + \beta)^{\nu} \leq E\left(\sum_{j=1}^{\infty} \prod_{i=1}^{j} (\alpha\eta_{k-i}^2 + \beta) \right)^{\nu}$$

and therefore $E\sigma_0^{2\nu} < \infty$ if and only if $E(\alpha\eta_0^2 + \beta)^{\nu} < 1$.

If $\alpha + \beta < 1$, not only the stationary solution exists but ϵ_i has a finite second moment. It follows from the recursion that for $i > j$,

$$E\epsilon_i\epsilon_j = E\eta_i\sigma_i\eta_j\sigma_j = E\eta_i E\sigma_i\eta_j\sigma_j = 0,$$

by the assumed independence of the η_i's and $E\eta_i = 0$. This means that ϵ_i is an uncorrelated sequence and there is no need to use the long run variance estimator since in case of uncorrelated the sample variance works, i.e.

$$S_N^2 \xrightarrow{P} \sigma^2$$

and

$$\sigma^2 = E\epsilon_i^2.$$

So in case of a GARCH(1,1),

$$\frac{1}{S_N^{1/2}} \max_{1\leq k\leq N} \left(\frac{N^2}{k(N-k)} \right)^{\kappa} \left| \sum_{i=1}^{k} \epsilon_i - \frac{k}{N} \sum_{i=1}^{N} \epsilon_i \right| \xrightarrow{\mathcal{D}} \sup_{0<t<1} \frac{1}{(t(1-t))^{\kappa}} |B(t)|$$

under the null hypothesis and

$$\frac{1}{S_N^{1/2}} \max_{1\leq k\leq N} \left(\frac{N^2}{k(N-k)} \right)^{\kappa} \left| \sum_{i=1}^{k} \epsilon_i - \frac{k}{N} \sum_{i=1}^{N} \epsilon_i \right| \xrightarrow{P} \infty,$$

where S_N^2 is the sample variance. Also, the power is not reduced as in case when the long run variance estimator is used.
Since the GARCH observations are uncorrelated, usually the squares of the log returns are analysed, especially if the investigation is based on sample correlations.

5. Applications

Natural application of the procedure presented is with financial and economic data.The application of these tests were performed on two datasets, gold in sterling and GameStop stock.

Background on volatility of gold dataset:

The beginning of 2016 was rather turbulent for gold. Safe-haven bids initially drove prices higher because safe-haven demand was strong. During the last quarter, prices kept falling due to the interest rate hike in December 2016. Early fears that the Federal Reserve would boost interest rates quickly reversed themselves as it became clear that the economy was too fragile to support aggressive tightening of monetary policy.

Background on volatility of GameStop stock:

In 2020, GameStop shares were worth a few dollars. By January 28th 2021 they peaked at over $480. The surge in value was seemingly down to amateur online traders who were on a mission to take on short-selling professionals. The frenzy around GameStop has rattled Wall Street, forced some online brokers to restrict trading and even raised concerns at the US Treasury.

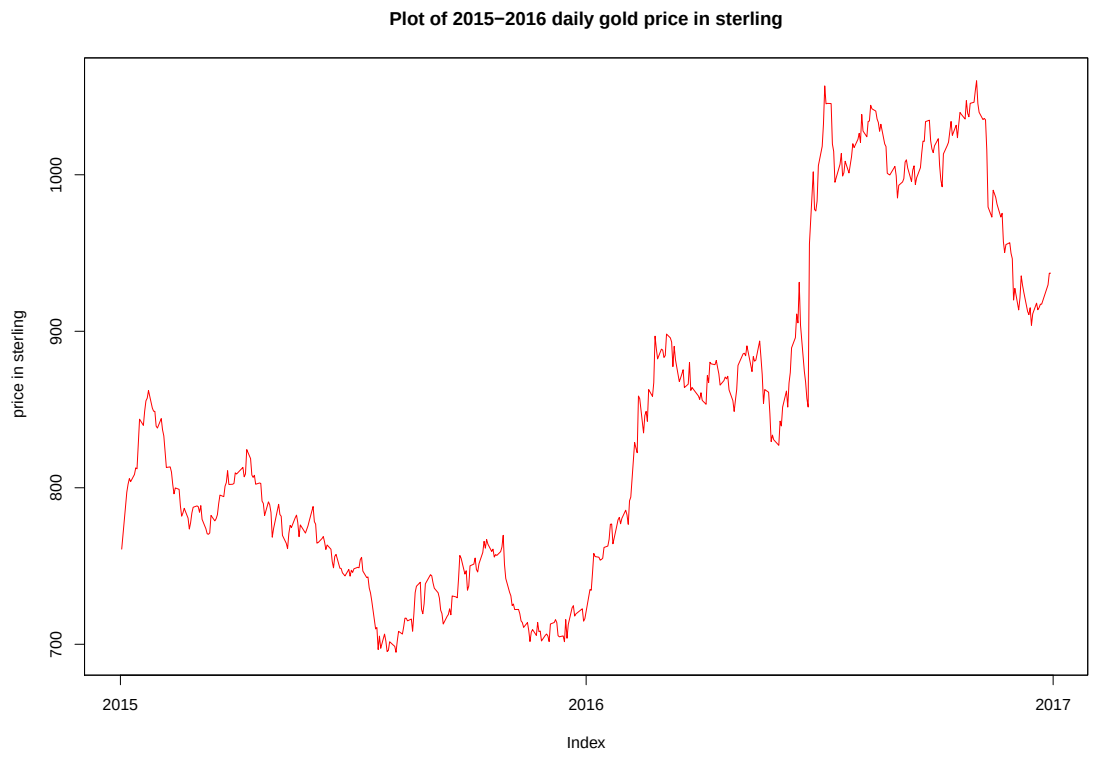
5.1. **Weighted CUSUM with GARCH assumption on Gold Data.**

We collected gold price in sterling from 01/01/2015 – 31/12/2016.
For more information, data was collected from https://www.bankofengland.co.uk/boeapps/database/

5.1.1. *Time Series Analysis of the Data.*

Using the same dataset as the above and looking at the basic statistics of the time series of gold price in sterling, we observe that the mean value is not zero and the variance is very high. This indicates that the time series is non-stationary with varying mean and variance. Thus, to stationarize the process, we study the log return of the price.

FIGURE 22



5.1.2. *Time Series Analysis of Log Return.*

As mentioned, in order to apply the test properly we need to convert the daily price into daily log returns to meet the stationarity assumption. We do so by taking the ratio of the daily price at time $i+1$ and the daily return at time i in order to calculate the ratio.

Let $X_1, X_2, ..., X_n$ be the gold prices. The return value is

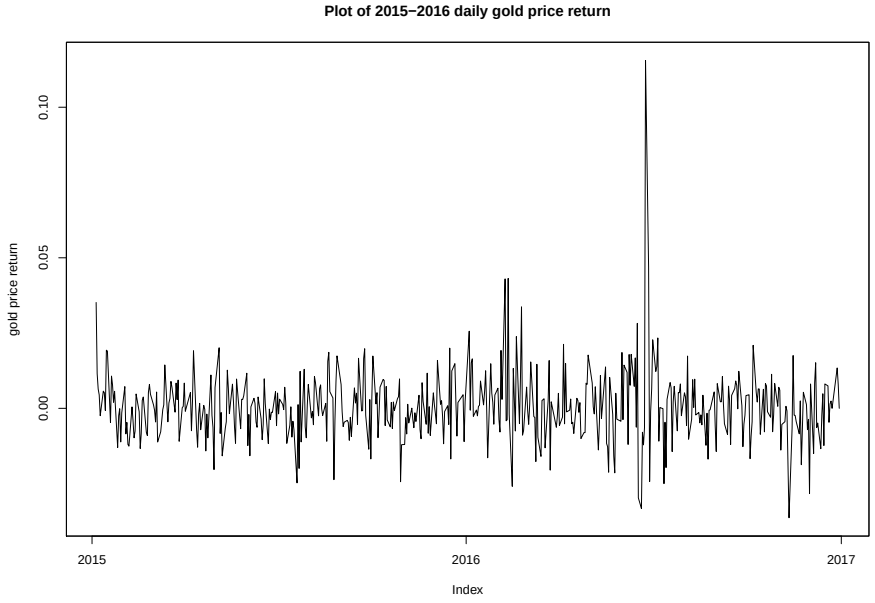
$$\frac{X_{i+1}}{X_i}$$

Then, we take the log of this ratio to get the return

$$Y_i = log\left(\frac{X_{i+1}}{X_i}\right) = log(X_{i+1}) - log(X_i)$$

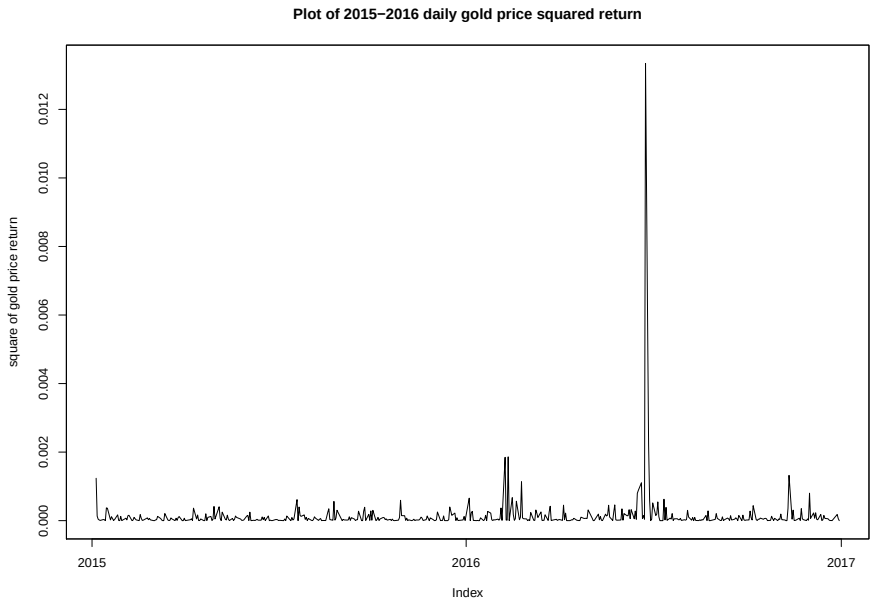
From Figure 23 we observe that the returns vary along the zero line with the largest log return of gold prices observed around beginning, mid and end of 2016, which shows signs of volatility.

FIGURE 23



To better identify its volatility, we also use the square of log return. Peaks are clearly shown in Figure 24 below.

FIGURE 24



5.1.3. *Change Point for Gold Data.*
Based on the volatility of the dataset, we are interested in finding if there is any change point, or if its mean has changed. Change point could be more subtle. Looking at Figures 23 – 24 alone could give us an idea of where the change should be but not quite clear.

The CUSUM is

$$\frac{1}{S_{N-1}} \frac{N^{-1/2}}{((k/N)(1-(k/N)))^{\kappa}} \left| \sum_{i=1}^{k} Y_i - \frac{k}{N-1} \sum_{i=1}^{N-1} Y_i \right|$$

with the sample mean and sample variance as follow

$$\bar{Y}_{N-1} = \frac{1}{N-1} \sum_{i=1}^{N-1} Y_i$$

$$S^2_{N-1} = \frac{1}{N-2} \sum_{i=1}^{N-1} (Y_i - \bar{Y}_{N-1})^2$$

We reject if

$$\max_{1 \le k < N-1} \frac{1}{S_{N-1}} \frac{N^{-1/2}}{((k/N)(1-(k/N)))^{\kappa}} \left| \sum_{i=1}^{k} Y_i - \frac{k}{N-1} \sum_{i=1}^{N-1} Y_i \right| \ge c(\kappa, \alpha)$$

Figures 25–27 are absolute value of the CUSUM process for Gold Data. More peaks are clearly shown, beginning and end also show more volatility.

Using Table 1, we fail to reject the hypothesis for cases when $\kappa = 0$ and when $\kappa = 0.3$. In the case $\kappa = 0.45$, we are close to find a change point and reject the null hypothesis with critical value $= 2.5$ at $\alpha = 0.1$. The higher κ, the more likely the model is weighted and change points are found.

FIGURE 25. A realization of the absolute value of the CUSUM process

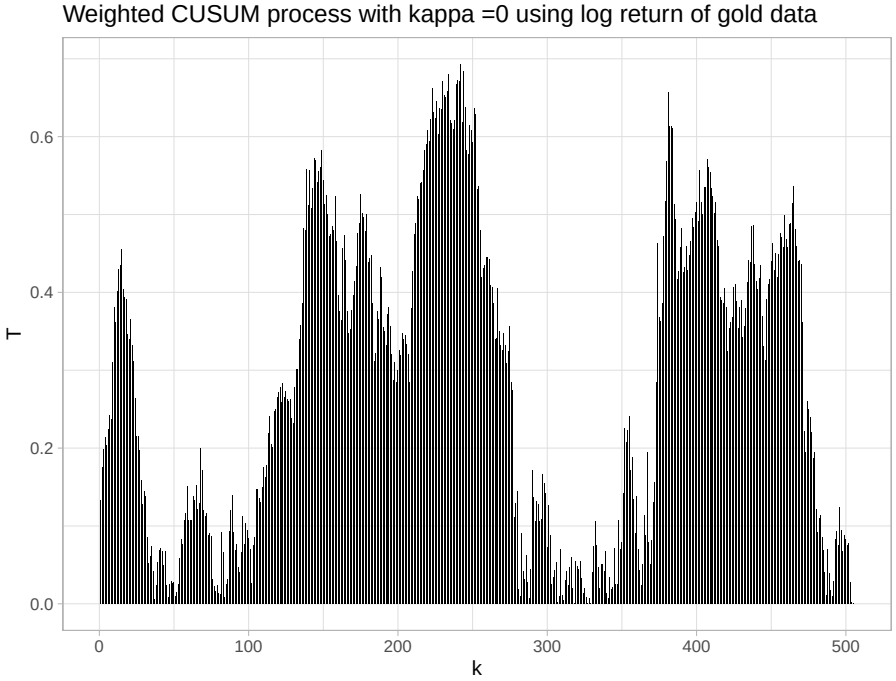


FIGURE 26. A realization of the absolute value of the CUSUM process

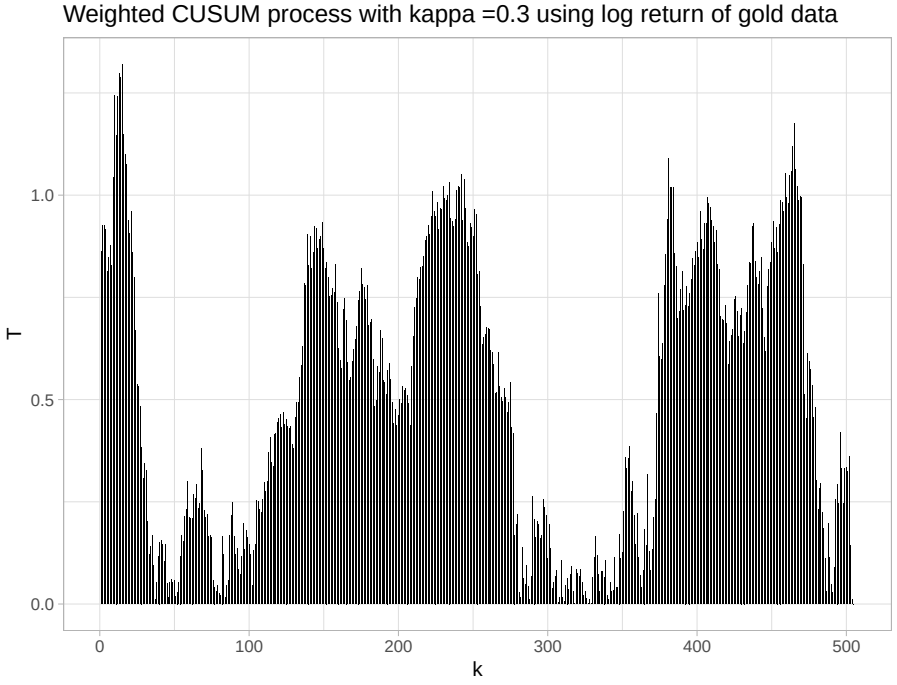
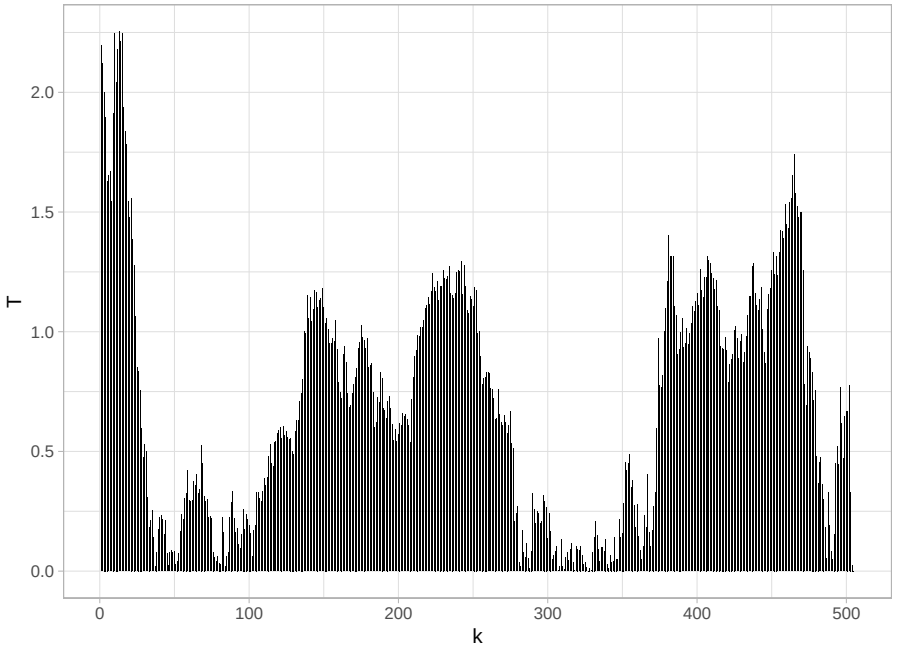


FIGURE 27. A realization of the absolute value of the CUSUM process



We could extend this research to monthly and quarterly return in the hope that more change points were better identified as price movement is more different versus daily prices tend to change gradually. Daily returns could have mean zero even during more volatile periods.

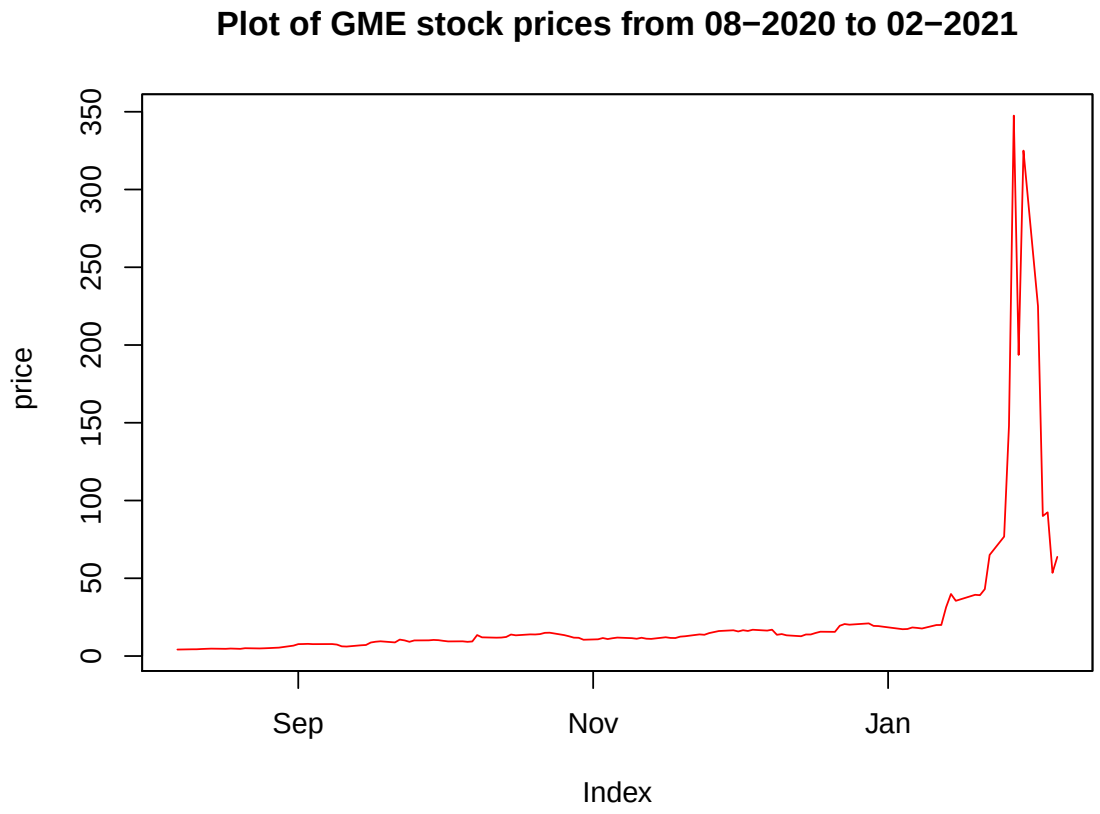
5.2. **Weighted CUSUM with GARCH assumption on GameStop Stock Data.**

5.2.1. *Time Series Analysis of the Data.*

We are still interested in daily return but this time we pick a stock that is very volatile to test change points. We collected daily GameStop stock prices in the past 6 months from August 2010 to February 2021 during the short squeeze period. For more information, the data was collected from https://finance.yahoo.com/quote/GME/history?p=GME

Figure 28 shows volatility by the end of January and the beginning February when there were a surge and drop in stock prices due to GameStop frenzy. Again, we have to use log return to stantionarize the time series data.

FIGURE 28



5.2.2. *Time Series Analysis of Log Return.*

Figures 29 and 30 shows the log return and the square of log return of GME data. As expected, we see some big swings at the end, but also some small swings in the beginning of the data.

FIGURE 29

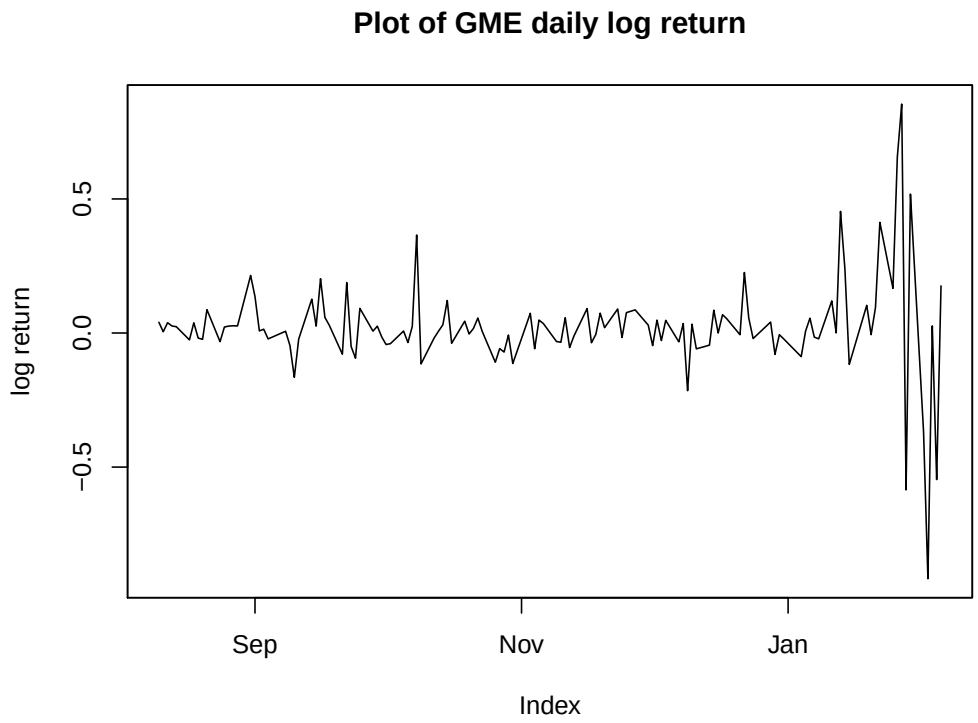
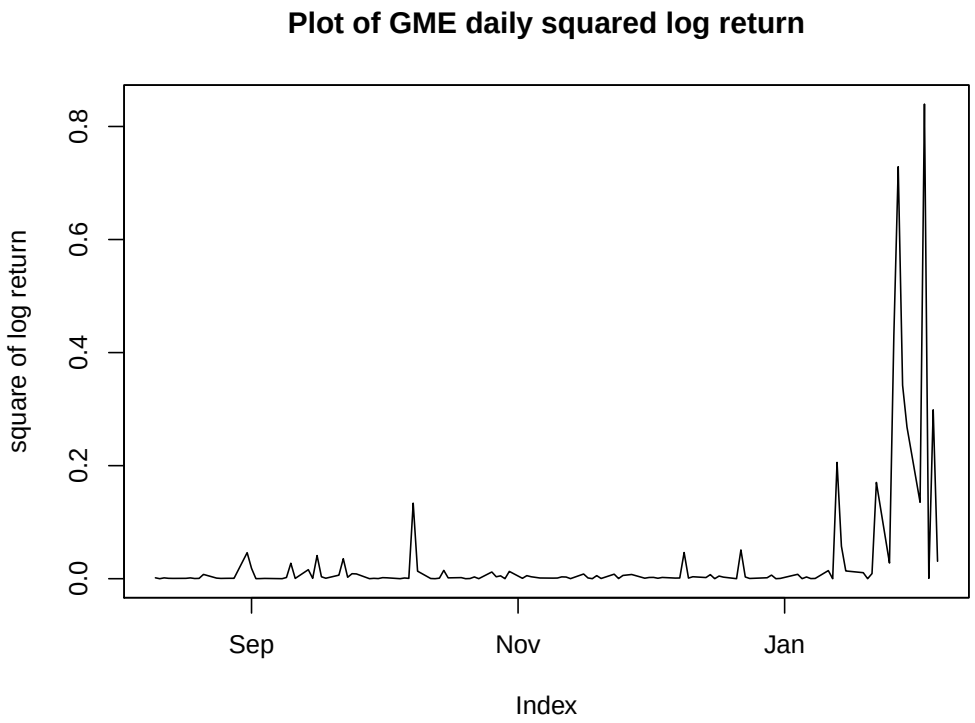

Plot of GME daily log return
log return
0.5
0.0
−0.5
Sep
Nov
Jan
Index

FIGURE 30



5.2.3. *Change Point for GME Data.*

Figures 31–33 are absolute value of the CUSUM process for Gold Data. More peaks are clearly shown, some small peaks at the beginning and large peaks at the end in February 2020 when the GameStop frenzy happened.

Using Table 1, we fail to reject the hypothesis for case when $\kappa = 0$. In the cases of $\kappa = 0.3$ and $\kappa = 0.45$, we are able to reject the null hypothesis with critical values of 1.9 and 2.5 at $\alpha = 0.1$. The higher κ, the more likely the model is weighted and change points are found.

FIGURE 31. A realization of the absolute value of the CUSUM process

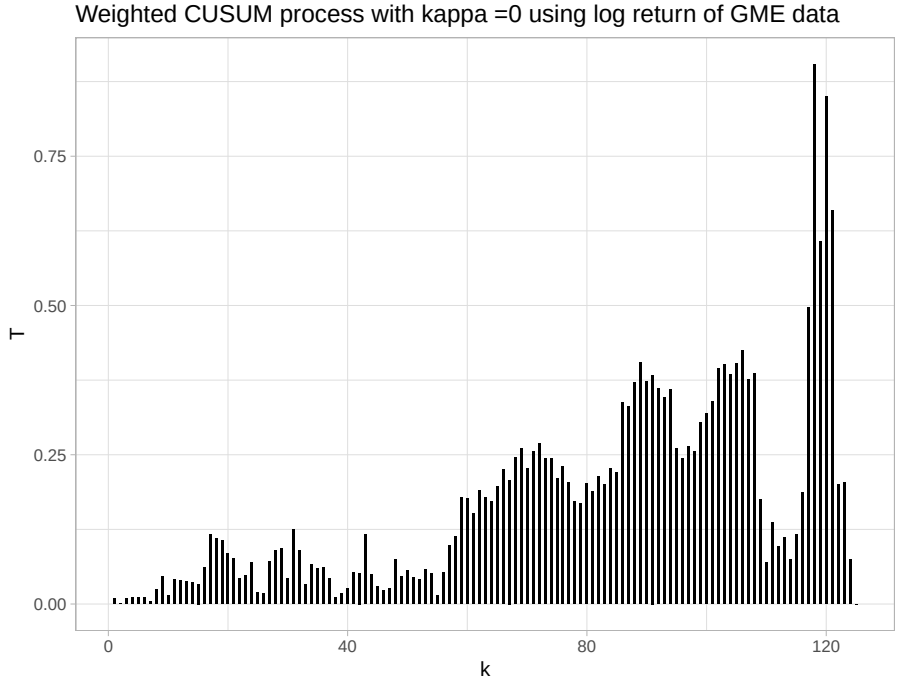


FIGURE 32. A realization of the absolute value of the CUSUM process

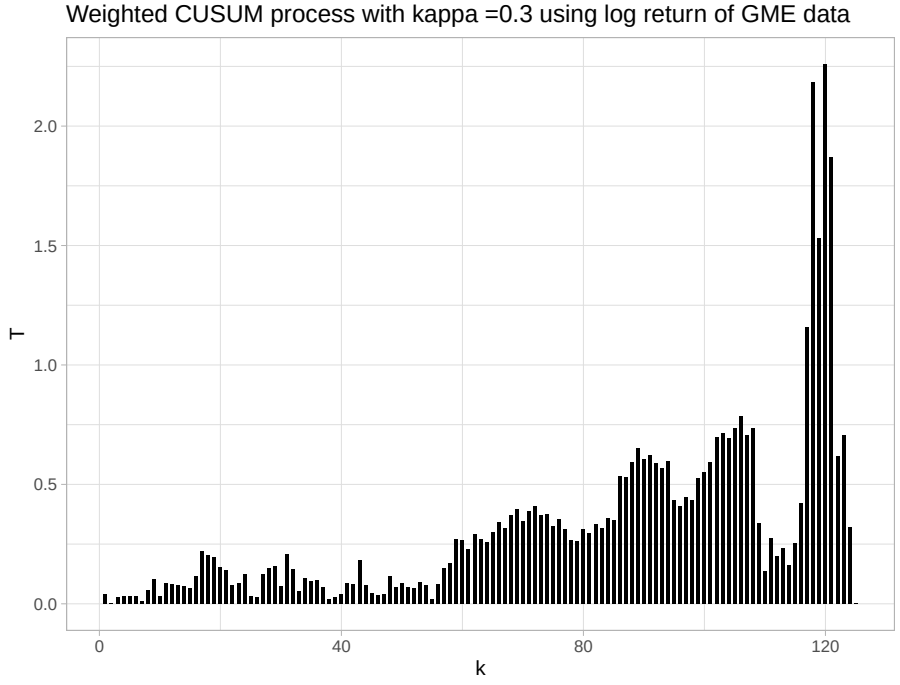
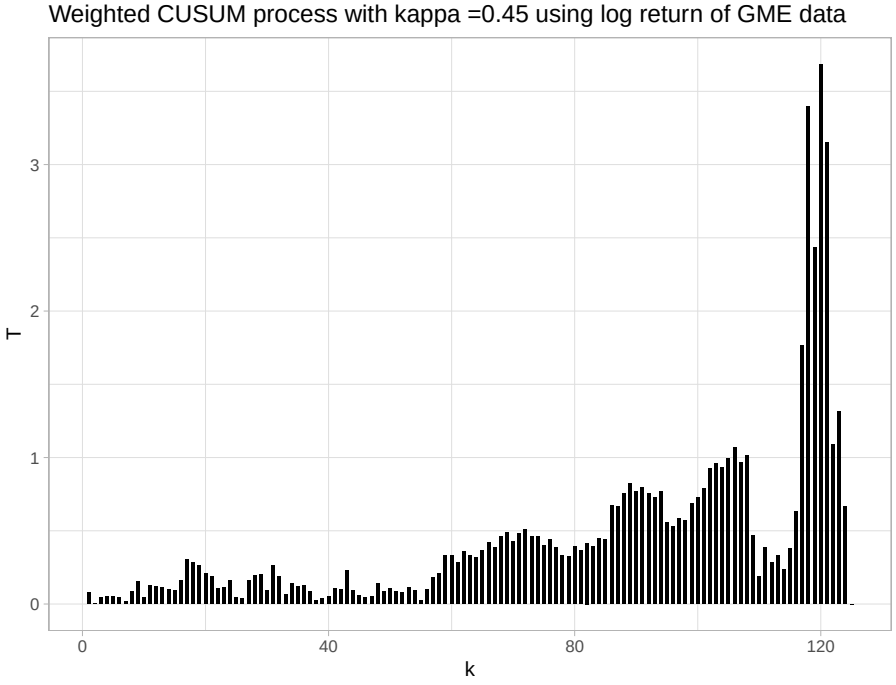


FIGURE 33. A realization of the absolute value of the CUSUM process



Since GameStop stock were more volatile, null hypothesis were rejected with higher confidence level and change points were easily identified

6. Conclusion

This research is limited to the assumption that the error term is uncorrelated and follows the GARCH process. If the assumption hold true, the error term is to be stationary. Then, the only problem that we are testing is whether or not the mean change. CUSUM analysis is an effective and powerful statistical tool for determining if and when a change in a data set has occurred. However, weighted CUMSUM method should be used for better accuracy. The higher the weight, the more likely it is to detect change. This allows us find important or abrupt changes in the dataset and avoid false positives.

By using the empirical to estimate the critical values, the tool also provides a confidence level that indicates the likelihood of the change. This technique works well with large changes because they produce an obvious change. The larger the volatility of the dataset, the more likely we are able to reject the null hypothesis and find change points. Perhaps this works well with volatile dataset. However, the application so far only focus on daily dataset where changes might not be drastic. The research could also be extended towards monthly, quarterly and yearly dataset to see the effectiveness of certain data types.

In terms of financial application, this detects anomalous patterns early and reliably, such methods form a foundation for active risk management, especially with unstable economy and past economic crises, including the latest and still-ongoing global financial meltdown and recession that started in 2008–2009 and pandemic provides graphic evidence of the importance of efficient methods for continuous financial surveillance.

7. Appendices

7.1. **Codes for CUSUM.**

Below is R codes for CUSUM

```
library(dplyr)
library(ggplot2)

#function that return the change point given dataframe
seq_change <- function(data) {
  df2 <- data %>% mutate(cumsum = cumsum(data$x),
                         ID = row_number(), #ID is reset every sequence for calcul
                         total =sum(data$x)) %>%
                  mutate(length_df = length(ID)) %>%
                  mutate(frac = ID/length(ID)) %>%
                  mutate(temp1 = frac * total) %>%
                  mutate(absolute = abs(cumsum - temp1)) %>%
                  mutate(T = absolute / (sqrt(length(ID))))
  #print(df2)
  #maxplot <- ggplot(df2, aes(x = k, y = T)) + geom_col()
  #print(maxplot)

  #Find the max value of T and return k where the max value is, this is the cuto
  #if it is tie return the smaller k
  maxk <- df2$k[max.col(t(df2$T), ties.method = "first")]
  maxT <- df2$T[max.col(t(df2$T), ties.method = "first")]

  return (list(maxT=maxT,maxk=maxk))
}


recurs_seq <- function(data){
  change_point <- seq_change(data)
  #print(length(data$k))
  #if (change_point$maxT >= 1.23/length(data$k)) { # Kolmogorov critical values
  if (change_point$maxT >= 1.23) { # Kolmogorov critical values
    #record the change point location
    locations <<- c(locations,change_point$maxk)
    left_cut <- subset(data, k<=change_point$maxk) %>% select(k, x)
    #print(left_cut)
    recurs_seq(left_cut)
    right_cut <- subset(data, k>change_point$maxk) %>% select(k, x)
    #print(right_cut)
    recurs_seq(right_cut)
  }
}
```

```
#####---------- Running the expirement m times ----------
# array that stores the location of change
locations <- c()

N <- 50
var <- 0.1
sd = sqrt(var)
N1 <- floor(N/3)
N2 <- floor(2*N/3)-floor(N/3)
N3 <- N - floor(2*N/3)

iter = 200
for (i in 1:iter){
  print(paste("i = ", i))
  #generate n independent normal random variables
  #first sequence
  x1 <- rnorm(N1, mean = 2, sd = sd)
  #second sequence
  x2 <- rnorm(N2, mean = 1, sd = sd)
  #third sequence
  x3 <- rnorm(N3, mean = 0, sd = sd)
  #combine 3 sequences
  x<- c(x1,x2,x3)
  #create a dataframe
  df <- data.frame(x) %>% mutate(k = row_number())

  recurs_seq(df)
}

locations.freq = table(locations)
loc_df <- as.data.frame(locations)

iterations <- paste("Iterations = ", iter)
variables_num <- paste("Sample_Size = ", N)
variance <- paste("Variance = ", var)
ggplot(loc_df,aes(x = locations)) + geom_histogram(aes(x = locations,y = ..count..),
  geom_density(aes(x = locations, y = ..count..), bw = 1, adjust=1, color = 'black')
  ggtitle(paste0(iterations,"\n",variables_num,"\n",variance)) +
  xlab("Location") + ylab("Frequency") + theme_light()
```

7.2. **Codes for CUSUM with Two Coordinators.**

Below is R Codes for CUSUM with Two Coordinators

```
library(tidyverse)
library(dplyr)
library(ggplot2)

#function that return the change point given dataframe
seq_change <- function(data) {
  df2 <- data %>% mutate(cumsum = cumsum(data$x),
                         ID = row_number(), #ID is reset every sequence for calc
                         total =sum(data$x)) %>%
    mutate(length_df = length(ID)) %>%
    mutate(frac = ID/length(ID)) %>%
    mutate(temp1 = frac * total) %>%
    mutate(absolute = abs(cumsum - temp1)) %>%
    mutate(T = absolute / (sqrt(length(ID))))

  #print(df2[1:5,])
  #maxplot <- ggplot(df2, aes(x = k, y = T)) + geom_col()

  #Find the max value of T and return k where the max value is, this is the cuto
  #if it is tie return the smaller k
  maxk <- df2$k[max.col(t(df2$T), ties.method = "first")]
  maxT <- df2$T[max.col(t(df2$T), ties.method = "first")]
  #print(paste("maxT",maxT))
  #print(paste("maxk",maxk))

  return (list(maxT=maxT,maxk=maxk))
}

coordinates <- function(segment) {
  if (segment$maxT >= 1.23) { # Kolmogorov critical values

    vector <- data.frame(0,segment$maxk)
  }else {
    vector <- data.frame(1,0)
  }
  names(vector) <- c("first_coord", "sec_coord")

  return (vector)
}

all_left_vectors <- data.frame(matrix(ncol = 2, nrow = 0))
names(all_left_vectors) <- c("first_coord", "sec_coord")

all_right_vectors <- data.frame(matrix(ncol = 2, nrow = 0))
```

```
names(all_right_vectors) <- c("first_coord", "sec_coord")

N <- 150
var <- 0.1
sd <- sqrt(var)
N1 <- floor(N/3)
N2 <- floor(2*N/3)-floor(N/3)
N3 <- N - floor(2*N/3)

iter = 200
for (i in 1:iter){
  print(paste("i = ", i))
  #generate n independent normal random variables
  #first sequence
  x1 <- rnorm(N1, mean = 2, sd = sd)
  #second sequence
  x2 <- rnorm(N2, mean = 1, sd = sd)
  #third sequence
  x3 <- rnorm(N3, mean = 0, sd = sd)
  #combine 3 sequences
  x<- c(x1,x2,x3)
  #create a dataframe
  data <- data.frame(x) %>% mutate(k = row_number())

  #print(data[1:5,])

  #print('First cusum')
  first_cusum <- seq_change(data)
  left_cut <- subset(data, k<=first_cusum$maxk) %>% select(k, x)
  #print('Second cumsum')
  #print(left_cut)
  second_cusum_left <- seq_change(left_cut)
  #print(second_cusum_left$maxk)
  #print(second_cusum_left$maxT)
  left_vector <- coordinates(second_cusum_left)
  all_left_vectors <<- rbind(all_left_vectors, left_vector)

  right_cut <- subset(data, k>first_cusum$maxk) %>% select(k, x)
  second_cusum_right <- seq_change(right_cut)
  right_vector <- coordinates(second_cusum_right)
  all_right_vectors <<- rbind(all_right_vectors, right_vector)
}

result <- function(all_vectors) {
  prob_change <- mean(all_vectors$first_coord)
  sec_loc_change <- sum(all_vectors$sec_coord)/(nrow(all_vectors)-nrow(all_vectors)*
```

```
  return (list(1-prob_change, round(sec_loc_change,2)))
}

r_left <- result(all_left_vectors)
r_right <- result(all_right_vectors)

left_title <- paste("Point of Change (Left) = ", r_left[2], ", Prob = ", r_left[
right_title <- paste("Point of Change (Right) = ", r_right[2], ", Prob = ", r_rig

#ggplot(all_left_vectors,aes(x = sec_coord)) + geom_histogram()
#ggplot(all_right_vectors,aes(x = sec_coord)) + geom_histogram()
#as.data.frame(table(all_left_vectors$sec_coord))

iterations <- paste("Iterations = ", iter)
variables_num <- paste("Sample Size = ", N)
variance <- paste("Variance = ", var)

d = data.frame(x = c(all_left_vectors$sec_coord, all_right_vectors$sec_coord),
               type=rep(c("Left", "Right"), c(length(all_left_vectors$sec_coord)
ggplot(d, aes(x=x, color=type)) + geom_histogram(fill="white",position="dodge")+
  ggtitle(paste0("Location of Change Points", "\n", iterations,"\n",variables_num
  xlab("Location") + ylab("Count") + theme_light() + theme(plot.title = element_t

# ggplot(d, aes(x = x, color=type)) +
#   geom_histogram(aes(y = (..count..)/sum(..count..)), fill="white",position="d
#   scale_y_continuous(labels = percent)

p = data.frame(x = c(all_left_vectors$first_coord, all_right_vectors$first_coord
               type=rep(c("Left", "Right"), c(length(all_left_vectors$first_coor
ggplot(p, aes(x=x, color=type)) + geom_histogram(fill="white",position="dodge")+
  ggtitle(paste0("Change Detection (1 = No, 0 = Yes)", "\n", iterations,"\n",varia
  xlab("Location") + ylab("Count") + theme_light() + theme(plot.title = element_t
```

7.3. **Codes for Weighted CUSUM.**

Below is R codes for Weighted CUSUM

```
library(dplyr)
library(ggplot2)

seq_change <- function(data, kappa) {
  df2 <- data %>% mutate(cumsum = cumsum(data$x),
                         ID = row_number(), #ID is reset every sequence for calculat
                         total =sum(data$x)) %>%
    mutate(length_df = length(ID)) %>%
    mutate(frac = ID/length(ID)) %>%
    mutate(temp1 = frac * total) %>%
    mutate(absolute = abs(cumsum - temp1)) %>%
    mutate(T1 = absolute / (sqrt(length(ID)))) %>%
    mutate(T = T1/((frac * (1-frac))^kappa)) # kappa = 0.1, 0.45

  df2$T[is.na(df2$T)] <- 0

  #print(df2)
  #print(df2[1:5,])

  #Find the max value of T and return k where the max value is, this is the cutoff p
  #if it is tie return the smaller k
  maxk <- df2$k[max.col(t(df2$T), ties.method = "first")]
  maxT <- df2$T[max.col(t(df2$T), ties.method = "first")]

  return (list(maxT=maxT,maxk=maxk))
}

N <- 500
var <- 1
sd <- sqrt(var)
kappa = 0

max_k <- c()
max_T <- c()

iter = 1000
for (i in 1:iter){

  print(paste("i = ", i))

  x <- rnorm(N, mean = 0, sd = 1)
  #create a dataframe
  df <- data.frame(x) %>% mutate(k = row_number())
```

```
  change_point <- seq_change(df, kappa)
  max_k <<- c(max_k,change_point$maxk)
  max_T <<- c(max_T,change_point$maxT)
}

iterations <- paste("Iterations = ", iter)
sample_size <- paste("Sample Size = ", N)
variance <- paste("Variance = ", var)
kp <- paste("Kappa = ", kappa)

max_T <- as.data.frame(max_T)

#hist(max_T$max_T)

max_T.q <- round(quantile(max_T$max_T, probs = c(0.99, 0.95, 0.90)),3)

ggplot(max_T, aes(max_T)) + stat_ecdf(geom = "step")+
  ggtitle(paste0("Empirical Cumulative Distribution Function ","\n",iterations,"\
  #geom_vline(aes(xintercept=max_T.q),linetype = "dashed")+
  geom_hline(yintercept=c(0.99, 0.95, 0.90), linetype='dashed')+
  geom_vline(xintercept=max_T.q,linetype = "dashed")+
  scale_x_continuous(breaks = max_T.q,labels = max_T.q)+
  scale_y_continuous(breaks = c(0.25, 0.50, 0.75,0.99, 0.95, 0.90),labels = c(0.2
  labs(x= "Maximum of T",y = "Fn(Maximum of T)")+
  theme(axis.text.x=element_text(angle=45,hjust=1), panel.background = element_re

library(EnvStats)

crits <- qemp(p = c(0.990, 0.95, 0.90), obs = max_T$max_T)
```

7.4. **Codes for applications on CUSUM of log return.**

Below is R codes for applications on CUSUM of log return

```
#plotting time series
library(quantmod)
library(rugarch)
library(rmgarch)
library(tseries)
library(TSA)

#dat <- read.csv("gold in sterling.csv", header = TRUE)

dat <- read.csv("GME.csv", header = TRUE)

prices <- dat[,2]

n <- length(prices);
#log_return <- log(prices[-1]/prices[-n])
#lrest <- log(prices[-1])-log(prices[-n])
log_return <- diff(log(prices), lag=1)

#plot with zero line
plot(log_return, ylab="", xlab="", type ="l")
abline(h=0,col ="darkgrey", lty = 3,lwd = 3)

#CUSUM of log return
library(dplyr)
library(ggplot2)
library(rugarch)
library(tseries)
library(fBasics)
library(zoo)
library(lmtest)
library(forecast)

seq_change <- function(data, kappa, sd) {

  df2 <- data %>% mutate(cumsum = cumsum(data$x),
                         ID = row_number(), #ID is reset every sequence for calculat
                         total =sum(data$x)) %>%
    mutate(length_df = length(ID)) %>%
    mutate(frac = ID/length(ID)) %>%
    mutate(temp1 = frac * total) %>%
    mutate(absolute = abs(cumsum - temp1)) %>%
    mutate(T1 = absolute / (sqrt(length(ID)))) %>%
    mutate(T2 = T1/((frac * (1-frac))^kappa)) %>%
    mutate (T = T2/sd)
```

```
  df2$T[is.na(df2$T)] <- 0
  #print(df2)

  #Find the max value of T and return k where the max value is, this is the cuto
  #if it is tie return the smaller k
  maxk <- df2$k[max.col(t(df2$T), ties.method = "first")]
  maxT <- df2$T[max.col(t(df2$T), ties.method = "first")]

  maxk_plot <- paste("Max k = ", maxk,",")
  maxT_plot <- paste("Max T = ", round(maxT,2))
  kp <- paste("Kappa = ", kappa)

  maxplot <- ggplot(df2, aes(x = k, y = T)) +
    #geom_col(width = 0.5, position = position_dodge(0.1),fill="black")+
    geom_col()+
    theme_light()

  print(maxplot)

  return (list(maxT=maxT,maxk=maxk))
}

recurs_seq <- function(data, sd){
  change_point <- seq_change(data,kappa,sd)
  if (change_point$maxT >= 2.5) { # Kolmogorov critical values
    #record the change point location
    locations <<- c(locations,change_point$maxk)
    left_cut <- subset(data, k<=change_point$maxk) %>% select(k, x)
    #print(left_cut)
    recurs_seq(left_cut,sd)
    right_cut <- subset(data, k>change_point$maxk) %>% select(k, x)
    #print(right_cut)
    recurs_seq(right_cut,sd)
  }
}

#####---------- Running the expirement m times ----------
# array that stores the location of change
locations <- c()

kappa <- 0.45
#gold <- read.table("gold in sterling 1516.csv", header = TRUE, sep = ',')
gold <- read.table("GME.csv", header = TRUE, sep = ',')
goldts <- zoo(gold$price, as.Date(as.character(gold$Date), format = c("%d-%b-%y"
# gold_num <- coredata(goldts)
# x<- gold_num
```

```
#log_return
gold_rets <- log(goldts/lag(goldts, -1))
gold_ret_num <- coredata(gold_rets)
x<- gold_ret_num

#create a dataframe
df <- data.frame(x) %>% mutate(k = row_number())
#sample mean
y_bar <- mean(gold_ret_num)
#sample variance (population variance)
s <- mean((gold_ret_num - y_bar)^2)
#sample sd
sd <- sqrt(s)

recurs_seq(df, sd)

locations.freq = table(locations)
loc_df <- as.data.frame(locations)

ggplot(loc_df,aes(x = locations)) + geom_histogram(aes(x = locations,y = ..count..),
  xlab("Location") + ylab("Frequency") + theme_light()
```

References

[1] Aue, A. and Horváth, L.: Structural breaks in time series. *Journal of Time Series Analysis* **23**(2013), 1–16.
[2] Csörgő, M. and Horváth, L.: *Limit Theorems in Change–Point Analysis.* Wiley, New York, 1997.
[3] Horváth, L. and Rice, G.: Extensions of some classical methods in change point analysis (with discussions) *TEST* **23**(2014), 219–290.
[4] Horváth, L., Rice, G. and Zhao, Y.: Detecting multiple changes in linear models. Preprint.
[5] DasGupta, A.: *Asymptotic Theory of Statistics and Probability*, 2008.