APPLYING KERNEL CHANGE POINT DETECTION TO FINANCIAL MARKETS

Tyler Manning-Dahan

A thesis
in
The Department
of
Engineering and Computer Science

Presented in Partial Fulfillment of the Requirements
For the Degree of Master of Applied Science
Concordia University
Montréal, Québec, Canada

March 2020

Concordia University
School of Graduate Studies

This is to certify that the thesis prepared

By: **Tyler Manning-Dahan**

Entitled: **Applying Kernel Change Point Detection to Financial Markets**

and submitted in partial fulfillment of the requirements for the degree of

Master of Applied Science

complies with the regulations of this University and meets the accepted standards with respect to originality and quality.

Signed by the final examining commitee:

_____ Chair

_____ Examiner

_____ Examiner

_____ Examiner

_____ Supervisor

Approved _____

Chair of Department or Graduate Program Director

_____ 20 _____ _____

Abstract

Applying Kernel Change Point Detection to Financial Markets

Tyler Manning-Dahan

Text of abstract.

Acknowledgments

I would like to thank my supervisor Dr. Jia Yuan Yu for accepting me into his lab and giving me the opportunity to further develop my academic career with this Master's thesis. I have become a much better researcher thanks to him and I am very grateful for his mentorship.

Several fellow researchers have been very helpful in writing this thesis. Thank you especially to Thomas Flynn and Damien Garreau who took a lot time of out their schedu le to answer my questions and inform particular directions of this thesis.

I would also like to thank all my colleagues at DRW who have helped me learn about the finance industry over the past three years. I am especially grateful to Yves, Neil and Laura, who have been very patient with me and have me taught me a lot about understanding financial markets, writing clean code and building robust statistical models.

Lastly, I would like to thank my fiancée, Tanya, who initially pushed me to go back to school while I was still young and had the opportunity.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In the first half of the twentieth century, Walter Shewart pioneered the use of statistical control charts for detecting real-time changes in variation. Shewart was interested in reducing the unexpected causes of variation in the manufacturing processes that produced faulty manufacturing equipment [48]. Shewart's method involved charting the process measurements over time and detecting when a statistical process was no longer exhibiting an expected level of variation. Once this detection occurred, the process was stopped and was not restarted until the cause of the variation was fixed. Shewart's control charts were one of the first formal methods to solve the problem of detecting changes in a distribution of a sequence of random variables. This problem is now known more generally as the *change point detection problem*. Many industries make use of change point techniques for real-time decision support systems. The following are a few motivating examples.

1.1.1 Health Care

Health care is an important area for quickly detecting signal changes. Some recent studies include applications to heart rate monitoring [56] [51], epilepsy signal segmentation [36], and multi-modal MRI lesion detection [6] to name a few. Quickly detecting changes to a patient's health is absolutely necessary for any system to be of practical use. However, this quick detection must be balanced with high accuracy as

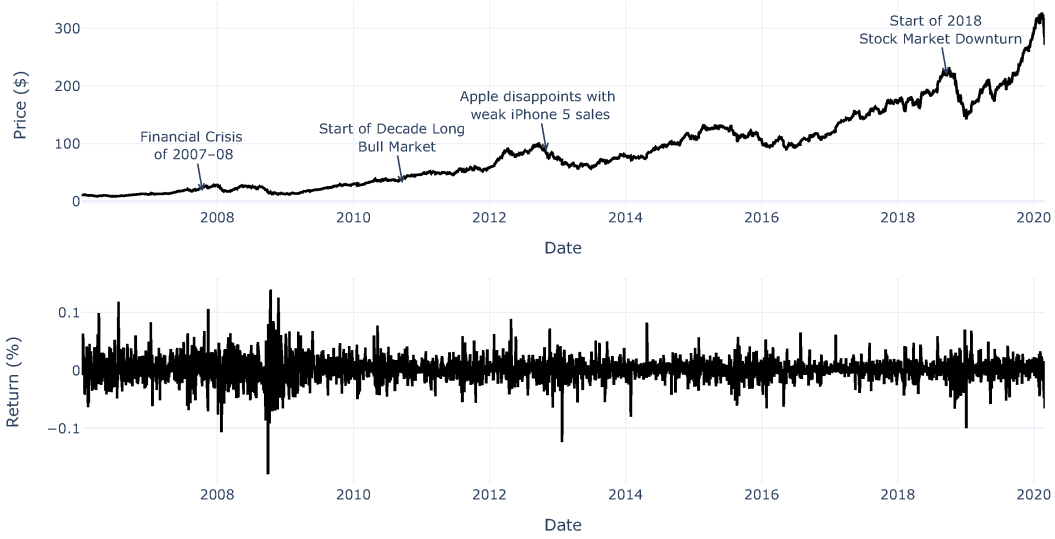
false positives or missed detections could have life-threatening consequences. Therefore, balancing missed change points with falsely identified change points is a central theme to online change point detection.

1.1.2 Financial Applications

The application of accurate and timely change point detection is popular in the finance sector where shifts in asset prices can suddenly happen. Change point detection is particularly hard in financial applications because of the non-stationary data typically observed in asset price time series. A common goal is detecting key, historical moments in the market such as stock market crashes [3] or the sub-prime mortgage crisis [59]. Note, in the financial literature, change points are also referred to as structural breaks, but for this thesis we will use the broader term change points.

An online, quick detection technique is proposed in [44], where a modified Shiryaev - Roberts procedure is used to detect a change point in a single stock's daily returns. See Figure 1 for the kind of of change points the authors try to detect. They compare their non-parametric method with other classic control chart methods using speed of detection and false alarm rate as measures of performance.

Figure 1: Daily stock prices (**top**) and daily returns (**bottom**) for Apple Inc. (NYSE: AAPL) for the period from January 1, 2006 through February 28, 2020.



Detecting changes in variance is explored in [28]. The authors propose an offline change point algorithm that minimizes a global cost function by using an adaptive regularization function. The algorithm is applied to the absolute returns of the FTSE 100 stock index and the US dollar-Japanese Yen foreign intra-day exchange rate to detect changes in asset price volatility. The change points identified in the FTSE 100 coincided with key market events such as the stock market crash that occurred on October 14^{th}, 1987 and breaking the 5000 price barrier in August 1997.

For more applications to options markets and arbitrage opportunities, see section 1.3.6 of [53].

1.2 Characteristics of the change point problem

A number of surveys of the literature already exist [1] [42], therefore we will not cover all existing methods but rather touch upon several, important factors to consider when approaching a change point detection problem. Across the literature, these factors determine what methods are available to practitioners.

The first factor is selecting between *parametric* and *non-parametric* techniques. Deciding between these two broad techniques is dependent on the prior knowledge one wants to encode into the problem. For example, if it is known that data is generated by a distribution from an exponential family of distributions, then the problem can be subsetted from the space of all possible distributions to a smaller space of distributions. For example, Shewart control charts and CUSUM change point techniques are both parametric techniques based on the Gaussian-family of distributions [43] [10]. In other settings, it is not possible to leverage information about the data and non-parametric techniques must be used instead [7].

The second factor is deciding whether change points should be detected *offline* or *online*. Some algorithms are offline—also referred to as batch algorithms or retrospective or *posteriori* change point detection—and they are applied in an ex-post fashion after the dataset has been completely acquired [54]. If change points must be detected as soon as possible, then waiting for the entire dataset to be acquired is not feasible and methods that operate on data streams must be used. Methods that fall into the category are referred to as online change point detection methods. The aforementioned Shewart control chart and CUSUM algorithm are both designed

for data that is streamed in real-time. In the statistical literature, online methods of change point detection are also referred to sequential change point detection [53]. For this thesis, the terms will be used interchangeably.

The third factor is determining if there are multiple change points or only one change point to detect. This is an important factor for offline change point detection where the decision to detect one or more change points is often chosen at the outset [25]. Detecting multiple change points could also be relevant for the online case if a situation arises where the window of time series under consideration may contain more than one change point. However, most online change point methods are designed to detect a single change point at a time.

Finally, the last factor to address is determining what statistical changes a change point detection algorithm should detect. Many methods focus solely on detecting changes in the mean of a distribution [30] or changes in variance [19] [24]. Methods like kernel change point detection do not focus on detecting a specific statistical parameter, but rather detecting that a change occurred in some moment of the distribution [2]. This is especially useful in situations where very little is known about the data.

This thesis will concern itself with online change point detection, where data is received in a streaming nature. We assume no prior distributional characteristics on the data and operate in a completely non-parametric setting.

1.3 Our Contributions

The contributions of this work are several fold. First, to our knowledge, no paper has explored the state of the book over time or applied change point detection techniques to order book liquidity levels. Second, given the strong emphasis of theoretical results in the change point detection community, we test several, recent online kernel change point algorithms on synthetic data and real-world data. Classification error as well as time to detection are compared across the various methods. The focus of recent algorithms is on kernel methods such as KCUSUM, NEWMA and Mstats-CPD that will be discussed in detail in Chapter 3.

1.4 Chapter Overview

Below is short description of each chapter and its contents. Ideally, each chapter should be read chronologically, but effort has been made so that each chapter is as self-contained as possible.

Chapter 2 provides a background on hypothesis testing and its relation to the change point detection problem. The online change point problem is formulated along with measures for evaluating performance. The chapter closes out by reviewing classic methods for detecting change points on streaming data.

Chapter 3 examines kernel change point detection as it is a focus of this thesis. A short background on the maximum mean discrepancy and its use in two-sample hypothesis tests is covered first. This is followed by a review of the most competitive online, kernel change point detection methods.

Chapter 4 applies our change point detection algorithm to liquidity in financial markets. A model of the limit order book is presented and the construction of the financial dataset is explained.

Chapter 5 concludes the thesis by summarizing all the results and discussing future avenues for research.

Chapter 2

Background

This chapter describes how the change point problem will be formulated in this thesis and, by extension, how all methods will be described using the change point detection problem notation. Because online change point detection is closely related to two-sample testing, a background on statistical hypothesis testing is presented first.

2.1 Hypothesis Testing

Let x and y be random variables defined on the topological space $\mathcal{X}$ with probability distributions P and Q respectively. Assume we draw n observations from P and m observations from Q resulting in two samples $X = \{x_1, x_2, ..., x_n\} \sim P$ and $X = \{y_1, y_2, ..., y_n\} \sim Q$, where each sample is independently and identically distributed with respect to P and Q. The main question posed in this thesis is can we determine if P and Q are statistically the same or different distributions.

To answer this question, we use the *statistical hypothesis testing* framework as it is described in [8]. Generally, a *hypothesis* is a statement about a population parameter, θ. Examples of a population parameter are the population mean, variance or other higher order moments. In hypothesis testing, we trying to determine whether one of two complementary hypotheses is true. The first, denoted by H_0, is called the *null hypothesis* and it states that $\theta \in \Theta_0$ where Θ_0 is some subset of the parameter space. The second hypothesis, denoted by H_1, is called the *alternative hypothesis* and it states that $\theta \in \Theta_0^c$. For instance, if it is believed P and Q are distinguishable by

their population means, μ_P and μ_Q, then the possible hypotheses are:

$$\begin{cases} H_0 : \mu_Q \in \{\mu_P\} & \text{(same mean, i.e. } P = Q\text{)} \\ H_1 : \mu_Q \notin \{\mu_P\} & \text{(different mean, i.e. } P \neq Q\text{)}. \end{cases} \tag{1}$$

How do we pick between H_0 and H_1 in equation 1? Every hypothesis test relies on a corresponding *test statistic* T that is a real-valued random variable. Because P and Q are unknown in our context, an estimate $\hat{T}$ can be calculated using X and Y such that $T : \mathcal{X}^n \times \mathcal{X}^m \to \mathbb{R}$, which yields:

$$\hat{T} = T(X, Y). \tag{2}$$

This test statistic is compared to a significance level, $\alpha \in [0, 1]$ that is chosen at the outset. Common choices for α are $0.1, 0.05$ and 0.01 [38]. Finally, a *p-value* is calculated using the test statistic $\hat{p} = P(T \geq \hat{T}|H_0)$. The p-value is the probability of observing $\hat{T}$ under the null hypothesis. A p-value $< \alpha$ would be improbable under the null hypothesis, therefore, in this situation it is rejected and the alternative hypothesis is accepted.

Given the binary outcome of a two-sample test, it is clear the hypothesis test can fail in the two following ways. The first is rejecting the null hypothesis when it is correct. This is known as a false positive or a type-I error and is upper-bounded by the chosen significance level, α. It is equivalent to following conditional probability: $\mathbb{P}(\text{reject } H_0|H_0 \text{ is true})$. The second possible source of error is a false negative or type-II error . The probability of committing a type-II error is denoted as $\beta = \mathbb{P}\left(\text{accept } H_0|H_0 \text{ is false}\right)$. The quantity $1 - \beta$ is referred to as the *power* of a test. Maximizing test power is an important part of designing new algorithms and is typically used to compare different methods.

Often there is a trade-off between type-I and type-II errors and the practitioner must decide how to balance the two given their domain-specific knowledge of the problem. In some cases, it may be desirable to sacrifice one for the other. For example, in the medical field [32], a false positive diagnosis (type-I error) may be more desirable than missing a diagnosis (type-II error) which would result in never giving treatment to a patient.

Recall P and Q are unknown, X and Y are the observed data and the other settings are chosen according to the specific hypothesis test. This means the main

decision left up to the practitioner is determining what statistical test to use. In other words, what T should be used for evaluating equation 2? The choice should depend on how P and Q may differ from each other. For example, the Student t-test is a two-sample test for determining if samples of univariate data come from a population with the same mean [52]. A generalization of the Student t-test for the multivariate case is the Hotelling T^2 test that compares whether the means of two multivariate samples are significantly different [22]. Both of these are parametric tests as they assume the samples are normally distributed.

Alternatively, non-parametric tests make no assumptions about the distributions P and Q. For example, the Kolmogorov-Smirnov test (KS test) [37] can determine whether or not two univariate samples come from the same distribution. This is done by computing the *supremum* of the difference of the empirical cumulative distribution functions from each sample. The KS test does not specify what distribution the samples come from, only if they differ according to the KS statistic. More recently, the kernel two-sample test introduced in [15] is another flexible, non-parametric test. It is not limited to one dimensional data, and can be applied to non-numeric data. It is based on the *maximum mean discrepancy* (MMD) statistic and is capable of detecting any kind of change in distribution. It is a focus in this thesis and is discussed in more detail in section 3.1.

2.2 Problem Formulation

Because there is no official standard formulation for the online change point detection problem in the literature, we use the description in [27] as the basis for the following problem description.

2.2.1 Change Point Detection Problem

Consider a data stream X to be a sequence of random variables, $X_1, X_2, X_3, ...$ where each X_t for $t = 0, 1, 2, ...$ is generated by some probability distribution P_t and each X_t is independent of the one that came before it. A change point occurs at time $t+1$ if $P_t \neq P_{t+1}$. In reality it is impossible to detect a real change point instantaneously.

The most recent w observations of X will be compared to some reference set of X where $w \in \mathbb{Z}^+$. Let this recent set be denoted as $X_t^w = \{X_{t-w}, ..., X_{t-1}, X_t\}$ whose

length is the window, w. Let the reference set be denoted by $X_t^{ref} = \{X_0, X_1, ..., X_{t-w}\}$ where the distribution of $P_0 = P_1 = ...P_{t-w}$ is sampled from a common distribution P. What is this reference set? It is data that is known to be distributed consistently prior to a change point. In the process control literature [5], it is known as the in-control distribution. We will use these terms interchangeably as they all capture the same idea. We do not explicitly state how many observations are stored for X_t^{ref} because this depends on the change point detection method used.

Let the probability distribution, P_{t+1}, denote be the distribution after the change point has occurred. At each time, t, the online change point detection problem is performing the following hypothesis test:

$$\begin{cases} H_0 : X_{ref} \text{ and } X_w \sim P & \text{(no change point occured)} \\ H_1 : X_{ref} \sim P \text{ and } X_w \sim Q & \text{(a change point occured)} \end{cases} \tag{3}$$

If the null hypothesis is true then the streaming data is distributed consistently along P and no change point exists. If the null hypothesis is rejected, the time series may be partitioned by a change point, t^*, that signifies all data from $t \geq t^*$ are distributed differently than data from $t < t^*$. Because we are operating in a non-parametric setting, the distributions P and Q are assumed to be completely unknown.

Many change point detection algorithms define a statistic that is computed using each set before and after the possible change point. If the statistic is above a threshold, $h \in \mathbb{R}$, then time t is classified as a change point. The estimated change point time is denoted as $\hat{t}$ and can be compared to underlying true change point, t^*, if it is provided.

The size of the window, w, is an important consideration that is typically chosen based on the problem being solved. A small window will detect change points more frequently resulting in fast change point detection but at the cost of more incorrect detections. A large window will have the opposite problem. In this case the change point detector will have less incorrect detections but will be slower to detect real change points.

2.2.2 Performance Measures

Two main issues arise when detecting change points in a stream of data. The first is detecting a change point when there is no actual statistical change in the observed

sequence. These are typically called *false alarms* or false detections in the change point detection literature. The false alarm rate is defined by a metric known as the *average time to false alarm* (TTFA). It is defined as:

$$TTFA = \mathbb{E}[\ \hat{t} \mid t^* \text{ does not exist}], \tag{4}$$

where it is the expected number of observations that are observed before a change point is incorrectly detected. In other words, it is the average amount of time until a change is detected given a sequence of observations with no change. Therefore, a larger value of TTFA is preferable. From a hypothesis testing perspective, this is equivalent to rejecting H_0 in when it should not be rejected, i.e. type I error. In the literature it is also referred to average run length (ARL) or average run length under the null (ARL_0).

The second issue is not detecting a change point when one occurs. This could be caused by detecting a change point much too late for it to be of any use or simply missing it altogether. It is measured by the missed number of detections and can be evaluated as a percentage by:

In cases where a change point is detected, the average detection delay (ADD) is used to estimate the average time it takes for a change point to be detected. It is computed by comparing the time difference between a predicted change point and an actual change point. This difference is then normalized the total by the total number of change points:

$$\text{ADD} = \frac{\sum_i^{\#CP} |\hat{t}_i - t^*|}{\#CP} \tag{5}$$

The closer the predicted change points are to the actual change points, the smaller the average detection delay. Because it relies on ground truth change points, it can only be used when experimenting on synthetic data where all the information is available.

If labelled change points are available for a real world dataset or a synthetic dataset, then the ground truth change point vector, t^*, is known. Evaluating performance in this case is the same as evaluating a binary classifier from a supervised learning problem. For example, the metrics from discussed in hypothesis testing, type-I error (false positives) and type-II error (false negatives), are commonly used for evaluating change point detection performance. The false positive rate (FPR) can be calculated by $FPR = FP/N$. Plotting the false positive rate and the true positive rate, gives the receiver operator characteristic curve. The area under this

curve (AUC), is calculated and compared to random baseline performance of 0.5 that would be equivalent to classification by tossing a coin.

Other standard classifier metrics can also used for comparing $\hat{t}$ and t^*. This includes the F1-Score that is based on a classifier's precision and recall:

$$F_1(\hat{t}, t^*) = 2 * \frac{\text{precision*recall}}{\text{precision} + \text{recall}} \tag{6}$$

F1-Score is defined as the harmonic mean of precision and recall. Precision is defined as the ratio of true positives (TP) to the number of true positives (TP) and false positives (FP) and recall is defined as the ratio the number of true positives to the number of true positives plus the number of false negatives. F1-Score is best when F1 = 1 (perfect precision and recall) and reaches its worst value at F1 = 0. Depending on the context, any other classifier evaluation tools such as the Receiver Operating Characteristics Curve and the Precision Recall Curve may be used as well.

2.3 Classic Algorithms

Presented below are the fundamental approaches to online change point detection that have been very influential on modern approaches. Many modern algorithms are variants of the classic algorithms discussed below. In the following sections let X_t be defined as it is in 2.2.1.

2.3.1 Shewart Control Chart

Shewart control charts were originally designed to detect changes in the mean of a process where the values being observed are assumed to be Gaussian distributed. As the data arrives, the data is batched into samples of size N. The sample mean, $\bar{X} = \frac{1}{N}\sum_{t=1}^{N} X_t$, is the then calculated and compared it to a known, true mean μ^*. Similarly, it is assumed the standard deviation, σ, is known in advance but it can also be estimated. If the absolute difference is greater than a threshold, then a change point is declared at the current batch. Therefore, the decision rule is defined as,

$$|\bar{X} - \mu^*| > \kappa \frac{\sigma}{\sqrt{N}}, \tag{7}$$

where $\kappa \in \mathbb{R}$ is a constant that controls how sensitive the algorithm is. Typically, it is set to $\kappa = 3$ as this coincides with the observations within 3 standard deviations of

the mean. Under the assumption that the data is distributed normally, approximately 99.7% of the observations are distributed in this region, therefore a change point is declared if it falls outside this region. The true mean is assumed to known and is defined as $\mu^* = \mathbb{E}[X_t]$. In applications, the true mean can also be replaced by some target specification that a process must adhere to.

Tuning the hyper-parameters can drastically change the performance of the algorithm. The parameter κ is used to control the trade-off between false alarms and missed detections. Choosing a lower κ makes the control chart detect change points more often and, consequently, increases the false alarms. Whereas a higher κ results in less false detections but also more missed detections. The chosen sample size, N, is also critical and its effect on the performance of Shewart control charts is studied in [18].

2.3.2 CUSUM

Similar to the Shewart control chart, the CUSUM algorithm tracks a statistic over time relative to a predetermined threshold. CUSUM is best applied to a process that is already under control. It can be thought of accumulating the information of current and past samples.

The algorithm is defined by a statistic, $S_t \in \mathbb{R}$, that is recursively updated after each sample, X_t, is observed, such that:

$$\begin{cases} S_0 = 0 & \text{(Initialization)} \\ S_t = \max(0, S_{t-1} + Z_t) & \text{for t=1,2,...,} \end{cases} \tag{8}$$

where $Z_t = ln(\frac{f_1(X_i)}{f_0(X_i)})$ and the statistic S_t is compared to a threshold $h \in \mathbb{R}$ that is predetermined by the user. Functions f_0 and f_1 are probability density functions for the pre-change distribution and post-change distribution. If $S_t \geq h$ then a change point is declared at time t and the algorithm is either stopped or restarted. Given that the statistic only flags change points when greater than a threshold, this algorithm only detects positive changes in the distribution. In [43], it is suggested to combine two CUSUM algorithms to detect positive and negative changes in a distributional parameter.

As a parametric algorithm, it is assumed f_0 and f_1 are known at the outset. In most applications, this is quite limiting and unrealistic. Therefore, in cases where

they are unknown, maximum likelihood estimates of each distribution's parameters are usually computed.

Several extensions to the CUSUM algorithm have been proposed such as the filtered-derivative extension introduced in [4], which uses the change of the discrete derivative of a signal over time to detect a change point. In [35], a fast initial response (FIR) CUSUM algorithm is proposed where the starting value of initial cumulative sums adapts over time. Instead of resetting S_0 to zero as shown above, it is reset to a non-zero value, typically based on the threshold chosen. This gives the algorithm a head-start in quickly detecting when a process is out of control and is especially useful for processes that don't start in control.

Finally, since CUSUM is typically better at detecting small shifts in signals and the Shewart control chart is faster at detecting larger changes, the two can be combined [34]. The combined Shewart-CUSUM algorithm leverages the strengths of both techniques for better overall performance. See [57] and [55] for more details.

2.3.3 EWMA

First described in [46] as a "geometric average", the exponentially weighted moving average (EWMA) is a type of moving average that applies exponential weighting to time series samples. Initially used as a forecasting technique in the econometrics field for smoothing noisy functions, the EWMA can also be used for determining out of control processes as shown in [23]. Rather than weighing all observations uniformly like the standard CUSUM algorithm or a simple moving average, a decay factor (also called a forgetting factor), $\lambda \in [0, 1]$, is used to control how much weight is distributed over the previous observations. As each new observation arrives, the EWMA statistic, $E_t \in \mathbb{R}$, is recursively updated and compared to a threshold. If the EWMA statistic exceeds the threshold then the process is deemed out of control or, in other words, a change point is detected.

The EWMA statistic is calculated as follows at each time step, t:

$$E_t = \lambda X_t + (1 - \lambda) E_{t-1}.$$

As $\lambda \to 1$, the EWMA gives more and more weight to the most recent observations similar to a Shewhart control chart, which gives weight to the last observation only. Conversely, as $\lambda \to 0$, the weights are distributed further into the past giving the

EWMA a longer memory similar to the CUSUM algorithm. Therefore, a EWMA control chart can be interpreted as a trade-off between a Shewhart control chart and a CUSUM control chart.

For detecting deviations away from a mean target value, control limits may be calculated in a similar manner to the Shewart control chart. In [23], control limits for the EWMA are chosen to be $\pm 3\sigma\sqrt{\frac{\lambda}{2-\lambda}}$. Like the Shewart control chart, it is assumed the standard deviation, σ, is known in advance but it can be estimated.

As with the other methods previously mentioned, the standard EWMA is a parametric method as it assumes the time series has some in-control average that is known prior to use. This makes is difficult to apply in situations where the data is coming from unknown distributions. It is however very fast due to its recursive structure and does not hold a lot of data in memory making it appealing for live data streams that need fast data processing.

Chapter 3

Kernel Change Point Detection

3.1 Maximum Mean Discrepancy

Suppose n samples from a set $X = \{x_1, x_2, ..., x_n\}$ and m samples from a different set $Y = \{y_1, y_2, ..., y_m\}$ are derived from a metric space $\mathcal{X}$. Assume both are distributed as $X \sim P$ and $Y \sim Q$ respectively. As mentioned in 2.1, there needs to be some notion of distance, $d(X, Y)$, that can be computed between the two samples in order to conduct a two-sample test. Intuitively, if P and Q are very similar then $d(X, Y)$ should be close to zero. On the other hand, if P and Q are very different then $d(X, Y)$ should be large.

One such family of statistical distances is the group of *integral probability metrics*, first described in [41]. The basic setup of these metrics is consider a space of functions, $\mathcal{F}$, where every function $f \in \mathcal{F}$ is a mapping such that $f : \mathcal{X} \to \mathbb{R}$. An IPM between P and Q can be defined as:

$$d(P, Q) = \sup_{f \in F} |\mathop{\mathbb{E}}_{X \sim P}[f(X)] - \mathop{\mathbb{E}}_{Y \sim Q}[f(Y)]|. \tag{9}$$

The choice of the space of functions, $\mathcal{F}$, determines the kind of distance measured between P and Q. [Discuss example using KS distance.]

In [15], the authors propose setting $\mathcal{F}$ to the unit ball in a Reproducing Kernel Hilbert space (RKHS), denoted by $\mathcal{H}$, i.e. $\mathcal{F} = \{f : ||f||_{\mathcal{H}} \leq 1\}$. The fact that a RKHS is chosen means there exists a *feature map*, $\phi : \mathcal{X} \to \mathcal{H}$. In [49], the formal definition of the maximum mean discrepancy (MMD) is defined as:

$$\text{MMD}(P, Q) = ||\mathop{\mathbb{E}}_{X \sim P}[\phi(X)] - \mathbb{E}_{Y \sim Q}[\phi(Y)]||_{\mathcal{H}}. \tag{10}$$

Where the MMD can be interpreted as the distance in $\mathcal{H}$ between the mean embeddings of the features. As long as a kernel mean embedding can be defined on the given data structure, the MMD can be used. This is why it can be applied to non-numeric data such as strings, graphs, and other structured domains [21].

Due to the reproducing property, the feature map, ϕ, has a corresponding kernel function, $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ where $k(x, y) = \langle \varphi(x), \varphi(y) \rangle_{\mathcal{H}}$. The proofs for this setting are rigorous and can be explored in [49] and XXXX and out of scope for discussion in this thesis. For our purposes it is important to understand that the kernel map, ϕ, does not need to be explicitly computed thanks to the *kernel trick*. Instead, we can only concern ourselves with the computation of the kernel function. Therefore, 10 can be rewritten as:

$$\mathrm{MMD}(P, Q) = \sup_{f \in F} | \mathop{\mathbb{E}}_{X \sim P}[f(X)] - \mathop{\mathbb{E}}_{Y \sim Q}[f(Y)]|. \tag{11}$$

If the kernel mapping into the Hilbert space is injective then k is called a *characteristic* kernel function. In [13], it is proven using a characteristic function to compute the MMD implies that $||\mu_P - \mu_Q|| = 0$ if and only if $P = Q$. As [40] points out, there is no loss if information in this case.

Typically, the actual distributions P and Q are unknown, making μ_P and μ_Q also unknown. Therefore, for applications, empirical estimates using the sampled data X and Y must be used instead. As shown in [40], the empirical estimate of the kernel mean embeddings, denoted by $\hat{\mu}_P$ and $\hat{\mu}_Q$ respectively, is

$$\hat{\mu}_P := \frac{1}{n} \sum_{i=1}^{n} k(x_i, \cdot) \tag{12}$$

$$\hat{\mu}_Q := \frac{1}{n} \sum_{i=1}^{n} k(x_i, \cdot). \tag{13}$$

As shown in [50], $\hat{\mu}_P$ is an unbiased estimate of μ_P and by law of large numbers, $\hat{\mu}_P$ converges to as $n \to \infty$.

In [15], the unbiased, estimate of the squared MMD is shown to be:

$$\widehat{\mathbf{MMD}}_u^2(X, Y) = \frac{1}{m(m-1)} \sum_{i=1}^{m} \sum_{j=1}^{m} k\left(x_i, x_j\right) - \frac{2}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} k\left(x_i, y_j\right) + \frac{1}{n(n-1)} \sum_{i=1}^{n} \sum_{j=1}^{n} k\left(y_i, y_j\right) \tag{14}$$

and the biased estimation of the squared MMD is

$$\widehat{\mathbf{MMD}}_b^2(X,Y) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} k\left(x_i, x_j\right) + \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m} k\left(y_i, y_j\right) - \frac{2}{nm}\sum_{i=1}^{n}\sum_{j=1}^{m} k\left(x_i, y_j\right). \tag{15}$$

Common characteristic kernel functions used in practice are the radial basis function kernel, $k(x,y) = e^{-\frac{1}{2\sigma^2}||x-y||^2}$ and the laplace kernel, $k(x,y) = e^{-\frac{1}{\sigma^2}||x-y||}$, where $\sigma > 0$. In [16], the authors recommend selecting the bandwidth kernel, σ, based on the *median heuristic*, which computes σ according to,

$$\sigma^2 = \text{median}\{||x_i - x_j|| : i, j = 1, ..., n\}. \tag{16}$$

This heuristic is a good starting point, however, it has been shown to be suboptimal in high-dimensional and small-sample cases as shown in [39] and [45]. Choosing the proper kernel is often done on a per case basis by maximizing test power. Overall kernel selection is a difficult problem that is still actively researched.

Originally in [15], it was thought the MMD does not suffer from the curse of dimensionality when used to compare distributions in higher dimensions. However, it was shown in [45] that indeed the MMD does struggle in higher dimensions.

Using the properties of the MMD and characteristic kernel functions, a non-parametric two-sample test can now be defined to solve the following non-parametric two-sample hypothesis test. Assuming P and Q are defined as above,

In [15], the authors defined a two-sample kernel test statistic using the MMD as a distance measure for comparing two probability distributions. 1. Choosing a rejection threshold, $c_\alpha > 0$, and kernel function, k, as defined above. 2. Estimate $\text{MMD}(P,Q)$ using equation 6 or 7. 3. If $M\hat{M}D > c_\alpha$, reject H_0, else accept H_0

Notice the choice of kernel function and threshold here is critical for assessing performance. Because noisy samples are used to estimate the distributions of P and Q, it will almost certainly never be exactly zero even if a characteristic kernel function is used and even if both samples come from the exact same distribution.

3.2 Related Work

One of the first papers to use the term *kernel change point detection* was in [11]. The authors present an online kernel change point detection model based on single

class support vector machines (ν-SVMs). They train a single class support vector on a past set, $\mathbf{x}_{t,1} = x_{t-m_1}, ..., x_{t-1}$ of size m_1 and train another single class support vector on a future set $\mathbf{x}_{t,2} = x_t, ..., x_{t+m_2-1}$ of size m_2. A ratio is then computed between the two sets that acts as the dissimilarity measure in Hilbert space. If the sets are sufficiently dissimilar over some predetermined threshold,then a change point is assigned to the time step that spits the two sets of data. The authors argue that a dissimilarity measure between kernel projection of points in a Hilbert space should estimate the *density supports* rather than estimate the probability distributions of each set of points. While this approach inspired a lot of interesting research that will be discussed below, it has not been studied since.

In [17], the authors approached the off-line change point problem with a fixed number of change points using kernel change point detection. This was further extended to an unknown number of change points in [2]. Garreau shows their KCP procedure outputs an off-line segmentation near optimal with high probability. Offline kernel change point detection differs in techniques than the online one, but they do have several common benefits. One is that it operates on any kind of data for which a kernel can properly reproduce a Hilbert space. For example, it can be applied to image data, histogram data, as well as $d-$dimensional vectors in $\mathbb{R}^d$. Lastly, the authors recommend choosing the kernel based on best possible signal to noise ratio. Therefore, some prior knowledge of a reference or training set is necessary for calibrating the kernel.

In [31], the authors make use of the B-test introduced in [58] and develop an offline and online change point detection algorithm called the MStats algorithm (the authors also refer to this algorithm as the Scan-B algorithm in a follow-up paper). At each time-step, the online model samples new data from a window of size B_0 and does a B-test with N past samples that are kept as reference samples.

$$Z_{B_0,t} := \frac{1}{N} \sum_{i=1}^{N} \mathrm{MMD}_u^2 \left(X_i^{(B_0,t)}, Y^{(B_0,t)} \right)$$

The resulting test statistic is then normalized by $Z_{B_0,t}/\sqrt{Var[Z_{B_0}]}$ where the authors provide a theoretical calculation of $\mathrm{Var}[Z_B]$. If the normalized test statistic exceeds some predefined threshold then a change point is declared. The B-test is memoryless in the sense that the statistic is calculated each time and only the value of the last calculation has any weight. This is similar to a control chart that calculates

a z-score at each iteration. Adjusting the size of the window, B_0, results in the usual trade-off of performance in online change point detection. A smaller block size will have a smaller computational cost and a smaller detection delay but will result in higher type II error and, as a result, worse test power.

From here, theoretical bounds are developed for the average run length and expected detection delay. Experiments are done on real-data sets including a speech dataset and the Human Activity Sensing Consortium (HASC) dataset where the performance was better than the relative density-ratio (RDR) algorithm described in [33].

More recently in [9], a kernel change point detection method is proposed that uses deep generative models to augment the test power of the kernel two sample test statistic. They point out MMD's lack of test power when using limited samples from the new distribution, Q, which may easily leading to over-fitting with kernels. Thus they use a generative adversarial neural network (GAN), trained on historical samples of $X \sim P$ with noise injected into X. This surrogate distribution is then used in conjunction with possible change points to improve the test power of a modified MMD measure that makes use of compositional kernels.

The method is compared to other prominent change point methods for off-line change detection such as the aforementioned MStats-KCPD, LSTNet, and Gaussian process change point models. All comparisons are done on synthetic data with piecewise i.i.d. data. All methods are benchmarked using the AUC metric for classification performance and it is shown the KL-CPD method is competitive or better than the state of the art methods. Furthermore, the AUC performance is maintained as the dimensionality of the data is increased, making their kernel learning framework very interesting for future off-line change point detection. It remains to be seen if this framework can be adopted in an online context where time to detection is a key constraint on practicality.

In the online setting, several methods use kernel embeddings with a two-sample hypothesis test. This is done in a similar vein to the classic CUSUM and Shewart control charts. They all make use of the maximum mean discrepancy (MMD) test statistic for a two-sample kernel hypothesis test.

A modified, "no-prior-knowledge" exponentially-weighted moving average (NEWMA) is introduced in [26]. Based on the standard exponentially weighted moving average,

NEWMA computes two EWMA statistics of different weights. If the difference between the two EWMA statistics exceeds a predefined threshold then a changepoint is declared at that time step. The point of using two EWMA statistics is for one to have a larger forgetting factor. Any recent changes in a distribution will weigh heavily on one statistic, resulting in a large difference between the two statistics. If this this statistical difference exceeds a predefined threshold, then a change point is declared at this point in time.

Since a standard EWMA is a parametric method, the authors apply a kernel mapping function, Ψ, to the data prior to applying the exponential weights. This provides a memoryless, non-parametric, online changepoint detection method that does not need to constantly store all previously streamed data. Once the statistics are updated at each iteration, the raw data may be discarded. This characteristic makes it especially useful in applications where security is a concern.

While kernel mean embeddings could be used for approximating, Ψ, as is the case for standard implementations of MMD, this would require the storage of past examples of data. Because the authors aim to reduce run-time cost and storage cost, they use a Random Fourier Features (RFF) approach for estimating Ψ (Note RFF is sometimes referred to as *random kitchen sinks*). There are several approaches available for optimizing the RFF approach that are well studied in the literature, namely the standard RFF implementation, the FastFood implementation introduced in [29], and Optical Processing Unit implementation from [47].

The three implementations of RFF and are compared to the MStats (Scan-B) algorithm by running empirical experiments on synthetic and real datasets. The synthetic datasets are run using streaming data that is generated from different Gaussian mixture models. They use an audio dataset for testing on real data. The variants of NEWMA are similar, if not better than MStats (Scan-B) in terms of missed detection percentage. In terms of average detection delay and false alarm trade-off, the NEWMA algorithm and its variants appear to be mildly better as well. The largest advantage of the NEWMA variants over the MStats (Scan-B) method is in the execution time. MStats (Scan-B)'s execution scales linearly with window size, while NEWMA's execution time does not depend on window size.

Finally, in a recent, preprint paper [12], a kernel CUSUM (KCUSUM) algorithm is proposed, where the classic CUSUM algorithm is adapted using the MMD statistic

for online detection. The authors use a modified, unbiased MMD statistic that can be computed in linear time. This formulation of the MMD statistic was originally defined in section 6 of [15] as:

$$\mathrm{MMD}_l^2[\mathcal{F}, X, Y] := \frac{1}{m_2} \sum_{i=1}^{m_2} h\left(\left(x_{2i-1}, y_{2i-1}\right), \left(x_{2i}, y_{2i}\right)\right)$$

Where,

$$h\left(\left(x_i, x_j\right), \left(y_i, y_j\right)\right) := k\left(x_i, x_j\right) + k\left(y_i, y_j\right) - k\left(x_i, y_j\right) - k\left(x_j, y_i\right)$$

The algorithm functions as follows, every two observations, the MMD_l is calculated using newly observed data points and data points sampled from some *reference* distribution that is known at the outset. This reference distribution can be thought of as the "in-control" distribution of the data-stream that new observations are compared to. The calculated MMD_l acts as the update term to the cumulative sum statistic, hence the name KCUSUM. If this kernel cumulative sum statistic exceeds some predefined threshold, then a change point is identified.

Besides its speed of computation, an additional benefit of MMD_l is it is normally distributed under the null distribution unlike the quadratically-calculated MMD_u. This facilitates analysis of bounds and provides statistical guarantees for worst-case detection delays and time to false alarm rate. While this non-parametric approach can detect any change in the distribution of a sequence, it does struggle with more complicated distributional changes such as variance changes of a single dimension and changes beyond first and second-order moments.

3.3 Our Approach

This section describes our novel method for change point detection.

3.4 Synthetic Datasets

A common difficulty in change point detection is evaluating the performance of an algorithm with datasets that aren't overly simplistic and difficult enough to ascertain some real world use.

Unlike fields like image recognition where datasets like MNIST provide a common benchmark, there are no standard datasets that are widely used across the change point detection literature for evaluating new methods. Most papers propose experiments that are relevant for the specific problem they are trying to solve but lack examples or explanations of when their method would not be applicable. Furthermore, because change point evolved out of the statistics literature, many papers focus on theoretical results and provide minor experimental results if any.

Given the empirical focus of this thesis, we attempt to put together the most comprehensive experiments using synthetic. To the best of our knowledge, no change point detection paper covers as many variations as presented in this thesis. While synthetic datasets are idealistic in their formulation, they provide a good starting point for comparing different methods because many variables can be controlled for. Often in the real world, the exact location of change points is not known. Therefore, it is important for the evaluation of a change point detection algorithm that it performs competitively on synthetic data.

Inspired by recent papers [9] and [12] that attempt to bridge the gap between the statistics and machine-learning literature, the following synthetic datasets are created: change in mean, scaling variance, alternating between two Gaussian mixtures, and alternating between random distributions. It is truly hard to properly generalize all the possible situations a non-parametric algorithm may be used in, but the synthetic cases presented in this thesis are common across domains and cover a range of applications.

For a change in mean, a change point is inserted in the time series at some random time where the mean is shifted either positively or negatively. There are two variants to this scenario. In the first, the mean change is in all dimensions simultaneously. In the second variation, the mean change is in only one dimension making it harder to detect.

For each experiment above, a Monte-Carlo approach is used to estimate time to false alarm, detection delay, and test power.

Table 1: Synthetic Datasets Summary

Type of Change	No. of Dimensions	Length	No. of Changepoints
Mean (all dimensions)	20	5000	50
Mean (single dimension)	20	5000	50
Variance	20	5000	50
Frequency	1	5000	50
Correlation	2	5000	50
Blobs	10	10 000	50
GMMs	50	10 000	50

Chapter 4

Application to Market Liquidity

In finance, *market liquidity* describes how quickly a market participant may buy or sell an asset without causing significant fluctuations in the price. In liquid markets, there is minimal impact to the price by quickly buying (selling) it. In illiquid markets, a purchase (sale) of the asset will cause the price to rise (fall) resulting in adverse price selection if the buyer (seller) is making a large transaction. In both cases, market liquidity is in constant fluctuation. It is this fluctuation that is the focus of this thesis. By detecting changes in market liquidity a market participant can identify periods of when trading may be more or less favourable. For example, an institutional investor wishing to unwind a large position over the course of several days may want to quickly detect rapid changes in a liquidity that could negatively impact their execution strategy. The following description of the limit order book is based on the notation used in [14].

4.1 Properties of the Limit Order Book

All modern financial markets are set-up as a double-sided auction between buyers (bid side) and the sellers (ask side) called limit order books (LOBs). A market participant may post orders at specific prices on either side of the book. A limit order, denoted by x, is defined as a tuple containing a price, p_x, and a size, w_x where $|w_x| > 0$:

$$x = \langle p_x, w_x \rangle. \tag{17}$$

A limit order's size indicates how many units of an asset a buyer (seller) is willing

to trade at p_x. The prices for which an order can be submitted at are discrete prices.

The LOB at time, t, is denoted as $\mathcal{L}(t)$, and it represents the set of all orders currently active at time t. All bid orders are at the best bid price or lower. The best bid price at time, t, is denoted as:

$$b(t) = \max_{\{x \in \mathcal{L}(t) | w_x < 0\}} p_x \tag{18}$$

All ask orders are placed at the best ask price or higher. The best ask price at time, t, is denoted as:

$$a(t) = \min_{\{x \in \mathcal{L}(t) | w_x > 0\}} p_x \tag{19}$$

The *bid-ask spread* or simply the spread at a time, t, is defined as:

$$s(t) = a(t) - b(t) \tag{20}$$

The normalized spread, $s_n(t) \geq 0$, can be defined by:

$$s_n(t) = \frac{s(t)}{\pi} - 1 \tag{21}$$

There are many types of orders that can be placed in the LOB. In this context, we assume all orders in the LOB at a given time are basic limit orders. That is, they are orders placed at a specific price with a specific size. Therefore, at a price, p, and a time, t, the total number of orders is:

$$n^b(p, t) = \sum_{\{x \in \mathcal{L}(t) | w_x < 0, p_x = p\}} |w_x| \tag{22}$$

On the ask side, it is similarly defined as:

$$n^a(p, t) = \sum_{\{x \in \mathcal{L}(t) | w_x > 0, p_x = p\}} w_x \tag{23}$$

Everytime a trade occurs or a resting limit order is cancelled at that price, size is removed from the LOB at that price. If the size at the $b(t)$ reaches zero then the spread increases the $b(t)$ decreases.

Every LOB has two configuration parameters: the tick size, $\pi > 0$, which is the smallest possible price increment between orders at different prices and lot size, which dictates the smallest amount that can be traded. As mentioned in [14], LOB

is essentially a one-dimensional lattice where the dimension is price and every point on the price axis is a multiple [1] of π.

There are two possible distributional changes that are of interest. The first is how the entire distribution of liquidity changes over time. These changes are often associated with events happening in the market such as the start of U.S. trading hours, the release of economic data or a planned speech by the head of the federal reserve. There may also be unplanned events that affect the overall liquidity in the book such as a the 9/11 terrorist attack or a virus outbreak like the Corona virus.

The second is how the distribution of the bid orders differs from the distribution of the ask orders. Typically, the distribution of liquidity on each side of the book is more or less symmetric. However, there have been studies that have shown times when there is an asymmetry between the two sides. This asymmetry has been shown to correlate to price changes.

4.2 Dataset Construction

While many types of markets operate using a basic order book, this thesis will be focused on futures markets found in the Chicago Mercantile Exchange (CME). The reasoning for this is two-fold. The first, unlike bond markets or currency pair markets, liquidity for many futures products is concentrated on a single exchange. This simplifies analysis of liquidity by not having to aggregate data across several exchanges into a synthetic ladder. The second advantage is, unlike equity markets where stocks and exchange traded funds (ETFs) are traded, there are no dark pools available for the futures markets under consideration in this thesis. Again this simplifies where the liquidity data for an instrument may appear.

The futures that will be used in our dataset are based on a combination of selecting the most traded contracts on the CME and selecting futures that represent different asset classes. They are summarized in the following table:

Table 2: Summary of Futures Studied

[1] In most cases these multiples of π are strictly positive but in particular markets, such as ED packs and bundles, there may be negative integers of π, i.e. a price axis with positive and negative prices.

Name	Asset Class	CME Symbol	Tick Size	Other
E-mini S&P 500	Equity	ES	0.25	1.402
10 Year T-Notes	Fixed Income	ZN	3.442	0.299
Crude Oil	Commodity	CL	3.442	0.299
Gold	Commodity	GC	3.442	0.299
Euro FX	Currency	6E	3.442	0.299

For each future contract in table 2, the state of the order book is sampled every 60 seconds from 7 AM to 4 PM EST for a period of 45 days. The characteristics of the order book that are sampled at every time step, t_I, are: 1) Quantity, q, at the top N levels of the book, $q_1 ... q_N$ 2) Spread, $s : [0, \mathbb{Z}^+]$, normalized by tick size, where spread of zero indicates there is no spread.

While spread is not indicative of liquidity but rather an absence of liquidity, we believe it is an important feature to include because all other things being equal, a change in spread is an important change in liquidity distribution for market participants.

Chapter 5

Conclusion

5.1 Summary of the thesis

Add description of thesis, summarizing the key points from each chapter and what this thesis attempted to add to the literature.

5.2 Discussion and Future Work

One focus of this thesis was using the MMD measure for two-sample testing and, by extension, for change point detection. It would interesting to explore other statistical distances that could compare two distributions non-parametrically. It would be interesting to re-use the same approaches explored in this thesis, but with the MMD swapped for a different distance measure.

Recently, interesting research has been made on re-purposing binary classifiers as two-sample tests. In [20], random forests are compared to different implementations of MMD. The authors run many tests comparing the test power across the different two-sample tests and their random forest. The results are interesting because while the random forest classifier is not better in every situation, it is better on hard two-sample tests such as the blobs dataset. Therefore, it remains to be seen whether these classifier approaches to two-sample testing can be adapted to online change point detection in an efficient manner.

- lopez paper

Bibliography

[1] Samaneh Aminikhanghahi and Diane J Cook. A survey of methods for time series change point detection. *Knowledge and information systems*, 51(2):339–367, 2017.

[2] Sylvain Arlot, Alain Celisse, and Zaıd Harchaoui. Kernel change-point detection. *arXiv preprint arXiv:1202.3878*, 6, 2012.

[3] Sayantan Banerjee and Kousik Guhathakurta. Change-point analysis in financial networks. *arXiv preprint arXiv:1911.05952*, 2019.

[4] MICHELE Basseville. Edge detection using sequential methods for change in level–part ii: Sequential detection of change in mean. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 29(1):32–50, 1981.

[5] Sotiris Bersimis, Stelios Psarakis, and John Panaretos. Multivariate statistical process control charts: an overview. *Quality and Reliability engineering international*, 23(5):517–543, 2007.

[6] Marcel Bosc, Fabrice Heitz, Jean-Paul Armspach, Izzie Namer, Daniel Gounot, and Lucien Rumbach. Automatic change detection in multimodal serial mri: application to multiple sclerosis lesion evolution. *NeuroImage*, 20(2):643–656, 2003.

[7] E Brodsky and Boris S Darkhovsky. *Nonparametric methods in change point problems*, volume 243. Springer Science & Business Media, 2013.

[8] George Casella and Roger L Berger. *Statistical inference*, volume 2. Duxbury Pacific Grove, CA, 2002.

[9] Wei-Cheng Chang, Chun-Liang Li, Yiming Yang, and Barnabás Póczos. Kernel change-point detection with auxiliary deep generative models. *arXiv preprint arXiv:1901.06077*, 2019.

[10] Jie Chen and Arjun K Gupta. *Parametric statistical change point analysis: with applications to genetics, medicine, and finance*. Springer Science & Business Media, 2011.

[11] Frédéric Desobry, Manuel Davy, and Christian Doncarli. An online kernel change detection algorithm. *IEEE Trans. Signal Processing*, 53(8-2):2961–2974, 2005.

[12] Thomas Flynn and Shinjae Yoo. Change detection with the kernel cumulative sum algorithm. *arXiv preprint arXiv:1903.01661*, 2019.

[13] Kenji Fukumizu, Arthur Gretton, Xiaohai Sun, and Bernhard Schölkopf. Kernel measures of conditional dependence. In *Advances in neural information processing systems*, pages 489–496, 2008.

[14] Martin D Gould and Julius Bonart. Queue imbalance as a one-tick-ahead price predictor in a limit order book. *Market Microstructure and Liquidity*, 2(02):1650006, 2016.

[15] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *Journal of Machine Learning Research*, 13(Mar):723–773, 2012.

[16] Arthur Gretton, Ralf Herbrich, Alexander Smola, Olivier Bousquet, and Bernhard Schölkopf. Kernel methods for measuring independence. *Journal of Machine Learning Research*, 6(Dec):2075–2129, 2005.

[17] Zaid Harchaoui and Olivier Cappé. Retrospective mutiple change-point estimation with kernels. In *2007 IEEE/SP 14th Workshop on Statistical Signal Processing*, pages 768–772. IEEE, 2007.

[18] Salah Haridy, Ahmed Maged, Saleh Kaytbay, and Sherif Araby. Effect of sample size on the performance of shewhart control charts. *The International Journal of Advanced Manufacturing Technology*, 90(1-4):1177–1185, 2017.

[19] Douglas M Hawkins and KD Zamba. A change-point model for a shift in variance. *Journal of Quality Technology*, 37(1):21–31, 2005.

[20] Simon Hediger, Loris Michel, and Jeffrey Näf. On the use of random forest for two-sample testing. *arXiv preprint arXiv:1903.06287*, 2019.

[21] Thomas Hofmann, Bernhard Schölkopf, and Alexander J Smola. Kernel methods in machine learning. *The annals of statistics*, pages 1171–1220, 2008.

[22] Harold Hotelling. The generalization of student's ratio. In *Breakthroughs in statistics*, pages 54–65. Springer, 1992.

[23] J Stuart Hunter. The exponentially weighted moving average. *Journal of quality technology*, 18(4):203–210, 1986.

[24] Carla Inclan and George C Tiao. Use of cumulative sums of squares for retrospective detection of changes of variance. *Journal of the American Statistical Association*, 89(427):913–923, 1994.

[25] Venkata Jandhyala, Stergios Fotopoulos, Ian MacNeill, and Pengyu Liu. Inference for single and multiple change-points in time series. *Journal of Time Series Analysis*, 34(4):423–446, 2013.

[26] Nicolas Keriven, Damien Garreau, and Iacopo Poli. Newma: a new method for scalable model-free online change-point detection. *arXiv preprint arXiv:1805.08061*, 2018.

[27] Daniel Kifer, Shai Ben-David, and Johannes Gehrke. Detecting change in data streams. In *VLDB*, volume 4, pages 180–191. Toronto, Canada, 2004.

[28] Marc Lavielle and Gilles Teyssiere. Adaptive detection of multiple change-points in asset price volatility. In *Long memory in economics*, pages 129–156. Springer, 2007.

[29] Quoc Viet Le, Tamás Sarlós, and Alexander Johannes Smola. Fastfood: Approximate kernel expansions in loglinear time. *arXiv preprint arXiv:1408.3060*, 2014.

[30] Tze-San Lee. Change-point problems: bibliography and review. *Journal of Statistical Theory and Practice*, 4(4):643–662, 2010.

[31] Shuang Li, Yao Xie, Hanjun Dai, and Le Song. M-statistic for kernel change-point detection. In *Advances in Neural Information Processing Systems*, pages 3366–3374, 2015.

[32] Matthew D Lieberman and William A Cunningham. Type i and type ii error concerns in fmri research: re-balancing the scale. *Social cognitive and affective neuroscience*, 4(4):423–428, 2009.

[33] Song Liu, Makoto Yamada, Nigel Collier, and Masashi Sugiyama. Change-point detection in time-series data by relative density-ratio estimation. *Neural Networks*, 43:72–83, 2013.

[34] James M Lucas. Combined shewhart-cusum quality control schemes. *Journal of Quality Technology*, 14(2):51–59, 1982.

[35] James M Lucas and Ronald B Crosier. Fast initial response for cusum quality-control schemes: give your cusum a head start. *Technometrics*, 24(3):199–205, 1982.

[36] Rakesh Malladi, Giridhar P Kalamangalam, and Behnaam Aazhang. Online bayesian change point detection algorithms for segmentation of epileptic activity. In *2013 Asilomar Conference on Signals, Systems and Computers*, pages 1833–1837. IEEE, 2013.

[37] Frank J Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.

[38] DS Moore and GP McCabe. Introduction to the practice of statistics, 2nd. *Edition. WH Freedom and Company, New York, New York*, 1993.

[39] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Arthur Gretton, and Bernhard Schölkopf. Kernel mean estimation and stein effect. In *International Conference on Machine Learning*, pages 10–18, 2014.

[40] Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, Bernhard Schölkopf, et al. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2):1–141, 2017.

[41] Alfred Müller. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29(2):429–443, 1997.

[42] Yue S Niu, Ning Hao, Heping Zhang, et al. Multiple change-point detection: A selective overview. *Statistical Science*, 31(4):611–623, 2016.

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

[44] Andrey Pepelyshev and Aleksey S Polunchenko. Real-time financial surveillance via quickest change-point detection methods. *arXiv preprint arXiv:1509.01570*, 2015.

[45] Aaditya Ramdas, Sashank Jakkam Reddi, Barnabás Póczos, Aarti Singh, and Larry Wasserman. On the decreasing power of kernel and distance based nonparametric hypothesis tests in high dimensions. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[46] SW Roberts. Control chart tests based on geometric moving averages. *Technometrics*, 1(3):239–250, 1959.

[47] Alaa Saade, Francesco Caltagirone, Igor Carron, Laurent Daudet, Angélique Drémeau, Sylvain Gigan, and Florent Krzakala. Random projections through multiple optical scattering: Approximating kernels at the speed of light. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6215–6219. IEEE, 2016.

[48] Walter Andrew Shewhart. *Economic control of quality of manufactured product*. ASQ Quality Press, 1931.

[49] Alex Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf. A hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory*, pages 13–31. Springer, 2007.

[50] Bharath K Sriperumbudur, Kenji Fukumizu, Arthur Gretton, Bernhard Schölkopf, Gert RG Lanckriet, et al. On the empirical estimation of integral probability metrics. *Electronic Journal of Statistics*, 6:1550–1599, 2012.

[51] M Staudacher, S Telser, A Amann, H Hinterhuber, and M Ritsch-Marte. A new method for change-point detection developed for on-line analysis of the heart beat variability during sleep. *Physica A: Statistical Mechanics and its Applications*, 349(3-4):582–596, 2005.

[52] Student. The probable error of a mean. *Biometrika*, pages 1–25, 1908.

[53] Alexander Tartakovsky, Igor Nikiforov, and Michele Basseville. *Sequential analysis: Hypothesis testing and changepoint detection*. Chapman and Hall/CRC, 2014.

[54] Charles Truong, Laurent Oudre, and Nicolas Vayatis. A review of change point detection methods. *arXiv preprint arXiv:1801.00718*, 2018.

[55] JO Westgard, T Groth, T Aronsson, and CH De Verdier. Combined shewhart-cusum control chart for improved quality control in clinical chemistry. *Clinical Chemistry*, 23(10):1881–1887, 1977.

[56] Ping Yang, Guy Dumont, and John Mark Ansermino. Adaptive change detection in heart rate trend monitoring in anesthetized children. *IEEE transactions on biomedical engineering*, 53(11):2211–2219, 2006.

[57] Emmanuel Yashchin. On the analysis and design of cusum-shewhart control schemes. *IBM Journal of Research and Development*, 29(4):377–391, 1985.

[58] Wojciech Zaremba, Arthur Gretton, and Matthew Blaschko. B-test: A non-parametric, low variance kernel two-sample test. In *Advances in neural information processing systems*, pages 755–763, 2013.

[59] Xiaoqian Zhu, Yongjia Xie, Jianping Li, and Dengsheng Wu. Change point detection for subprime crisis in american banking: From the perspective of risk dependence. *International Review of Economics & Finance*, 38:18–28, 2015.