Bandit Change Point Detection

A PROJECT REPORT
SUBMITTED IN PARTIAL FULFILMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
Master of Technology
IN
Faculty of Engineering

BY

Braghadeesh Lakshminarayanan

Electrical Communication Engineering
Indian Institute of Science
Bangalore – 560 012 (INDIA)

July, 2020

Declaration of Originality

I, **Braghadeesh Lakshminarayanan**, with SR No. **04-02-04-10-51-18-1-15704** hereby declare that the material presented in the thesis titled

Bandit Change Point Detection

represents original work carried out by me in the **Department of Electrical Communication Engineering** at **Indian Institute of Science** during the years **2018-2020**.
With my signature, I certify that:

- I have not manipulated any of the data or results.
- I have not committed any plagiarism of intellectual property. I have clearly indicated and referenced the contributions of others.
- I have explicitly acknowledged all collaborative research and discussions.
- I have understood that any false claim will result in severe disciplinary action.
- I have understood that the work may be screened for any form of academic misconduct.

Date: Student Signature

In my capacity as supervisor of the above-mentioned work, I certify that the above statements are true to the best of my knowledge, and I have carried out due diligence to ensure the originality of the report.

Advisor Name: Advisor Signature

DEDICATED TO

Amma, Appa & Anna
for their unconditional love, care & support

Acknowledgements

First of all, I would like to thank my advisor Dr. Aditya Gopalan for his support and encouragement for the past one year. His teaching, the way he formulates a problem, his clear cut explanations to the doubts and his excellent board work have always been the source of inspiration for me. I consider myself to be fortunate to have attended his course on Online learning and Prediction, which is the best course I have taken at IISc. I thank him for being tolerant to all my silly mistakes that I made during the discussion and also thank him for consistently following up my progress, even in these tough times due to COVID-19.

Next, I would like to thank my brother, Dr. Chandrashekar Lakshminarayanan who has always been my academic parent. He always played a big role in helping me achieve successes in academics so far. Next, I would like to thank my parents for their consistent faith on me, for offering unconditional support, love and care. I also thank my maternal cousins Ajay Krishnan and Sweatha, who also inspired me to take up an interest in studies. Special thanks to Dr. Prabu Chandran for patiently clearing my doubts though I pester him so many times. I also thank Dr. Arun Rajkumar for his mock interviews.

Next, I thank my lab mates Mohammadi Zaki, Sayak Ray Chowdhury and Debangshu Bannerjee for their useful discussions related to research. I also thank my senior Varun Krishna for consistently encouraging and motivating me. I also thank Chandrashekar Sriram, who is the topper of our batch, for clearing my doubts related to the courses that I took during my first semester at IISc. I also thank Arvind Rameshwar, who is my partner in crime at IISc and all those discussions we had from A-mess to ECE department cannot be forgotten and always be remembered in my memory. I also thank Datta Prasad, Venky, Rajkumar Santhanam, Vinayak Killedar and Sethupathy for all the funny conversation that we had at IISc. Lastly, I thank Prakruthi Restaturant (which is closed now) for their best filter coffees and all A-mess workers for providing homely food.

Abstract

We consider the problem of bandit change point detection. In the conventional change point detection problem, the goal is to detect the change in the distribution of the samples/data as quickly as possible. The bandit change point detection problem is modelled using the framework of stochastic linear bandits, where the aim is to play arms or actions adaptively to detect the change in the bandit parameter. The problem combines both the aspects of adaptive sensing (from bandits) and the goal of quickest change detection from classical sequential testing.

We consider bandit change point detection under the setting where the observations from each actions are linear (in expectation) in the action's features. For this setting, we propose two novel approaches, namely (i) Confidence set based approach, (ii) GLRT based approach to define the stopping rules and propose algorithms to stop and detect the change based on the stopping rules, as quickly as possible. While there are well known existing solutions to the classical(non-adaptive) change detection problem, our approach enables identification of a change in distribution of samples by adaptively sensing the samples in stochastic linear bandit framework.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we will look at a shorter introduction to the change point detection problem and its potential areas of application. We then look at the need to consider the controlled change point detection Problem and see how bandits play a crucial role.

1.1 Change Detection Problem

We consider the change point detection problem in which the data are assumed to be drawn (i.i.d) from a distribution parameterized by a parameter $\theta^{(0)} \in \mathbb{R}^d$, called pre-change parameter and the underlying distribution of the data changes to a distribution parameterized by $\theta^{(1)} \in \mathbb{R}^d \subseteq \Theta^{(1)} \subseteq \mathbb{R}^d$ at an unknown time $\tau \in \mathbb{N}$.

$$\begin{aligned} X_1, X_2, \ldots, X_{\tau-1} &\overset{i.i.d}{\sim} \mathbb{P}_{\theta^{(0)}} \\ X_\tau, X_{\tau+1}, \ldots &\overset{i.i.d}{\sim} \mathbb{P}_{\theta^{(1)}} \end{aligned} \tag{1.1}$$

The goal is to detect the change in the parameter of the distribution of the data as quickly as possible.

Change point detection problem arises in many real world scenarios. For instance, in monitoring of temperature of reactors in any power plant, the observations received from the temperature sensors are noisy and it is crucial to detect the change in the distribution of these observations as quickly as possible. The following are the important notions in any detection problem, which arise as the consequence of noisy observations.

- **False Alarm:** Detecting that the change in the distribution has occurred when there is no change
- **Missed Detection:** Detecting that no change in the distribution when there is actually a change in the distribution

As one can see, any model/algorithm that makes missed detection in the above scenario cause potential damage to the plant while false alarm makes the model more unreliable.

In this thesis, the Missed Detection is interpreted as the detection with some finite delay (In conventional setting, the Missed detection is interpreted as no detection when there is actually a change in the distribution).

There are many existing solutions to the change detection problem. However, none of the existing solutions take controlled sensing of samples into the consideration. In the next section, we will look at the notion of controlled sensing and in the later chapters, we will see that the controlled sensing of samples gives us the better detection delay.

1.2 Controlled Sensing

In many practical applications, the number of dimensions of the parameter characterizing the distribution will be greater than one. For instance, in the case of monitoring the temperature of the reactors in the power plant, there are different sensors at different places of the reactors to monitor the temperature. If the number of temperature sensors is d, then this leads to consider the temperature measurement as d dimensional vector, $x \in \mathbb{R}^d$. There may be a change in only few of the coordinates of x, in which case the correct sensing of the temperature of these coordinates will give us the better detection.

The notion of having control over the samples will become clear in the subsection Problem Setup and one would appreciate the reason to view the change point detection problem through the framework of stochastic linear bandits.

1.2.1 Problem Setup

Consider the problem of detecting the change in the parameter that governs the observations from the sensors, sequentially over time $t \in \mathbb{N}$. The goal of the problem is to detect the change as quickly as possible. In this discussion, let us define the stopping rule and propose an algorithm to detect the change as quickly as possible.

Parameter: Let $\theta^{(0)} \in \mathbb{R}^d$ be the parameter under which the observations are drawn before the change point. Let $\theta^{(1)} \in \Theta^{(1)} \subseteq \mathbb{R}^d$ be the parameter under which the observations are

drawn after the change point. Let τ be the time at which $\theta^{(0)}$ changes to $\theta^{(1)}$, which we call it as the change point. Let θ_t be the parameter that governs the observation at time t which is defined as

$$\theta_t = \begin{cases} \theta^{(0)}, & \text{if } t < \tau \\ \theta^{(1)}, & \text{otherwise} \end{cases} \tag{1.2}$$

Actions: Let $\mathcal{A} \subseteq \mathbb{R}^d$ be the given set of actions in $\mathbb{R}^d$. Let $A_t \in \mathcal{A}$ be the action taken at time t. The Actions provide us the control over samples in the following way.

Samples: Let X_t be the samples observed at time t from the distribution, $\mathbb{P}(X_t|A_t, \theta_t)$, given by $\mathbb{P}(X_t|A_t, \theta_t) = \mathcal{N}\Big(\Big\langle A_t, \theta_t \Big\rangle, \sigma^2\Big)$ [1], where σ^2 is the known variance. That is,

$$X_t \sim \mathcal{N}\Big(\Big\langle A_t, \theta_t \Big\rangle, \sigma^2\Big) \tag{1.3}$$

From (1.3), it is clear that though the actual data is of $d(> 1)$ dimension, which is given by θ_t, the samples that we observe is of dimension 1. Specifically, in *uncontrolled* change point detection setting, i.e, with no control over the samples, the observation that is drawn at time t is given by $X_t \sim \mathcal{N}\Big(\Big\langle e_t, \theta_t \Big\rangle, \sigma^2\Big)$, where e_t is a standard basis vector in $\mathbb{R}^d$. In other words, we observe only one of the sensors observation (noisy) at any given time. In general, in *uncontrolled* change point detection setting, the observation X_t drawn at time t is given by $X_t \sim \mathcal{N}(f(\theta_t), \sigma^2)$, where f is some unknown real-valued function of θ_t. In the stochastic linear bandit change point detection setup, i.e, in the *controlled* change point detection setting, the control over the sample X_t at time t comes through the action A_t at t and this linear structure in observations help us to devise algorithms/sensing rules to adaptively choose actions A_t to draw sample X_t at time t, $\forall t \in \mathbb{N}$.

1.3 Prior Work

Change point detection is a well studied problem since mid half of 20^{th} century. The first of early change point detection works appeared in the year 1954 by E.S. Page [15]. In this work [15] the data is observed as a batch or in other words data is observed in off-line setting and CUSUM procedure is used to detect the change in the distribution of the data. Also, in this setting, both pre-change and post-change parameters are completely known and the source of randomness is only in the point in discrete time scale where the change occurs. Lorden [11]

[1] $\mathcal{N}(\mu, \sigma^2)$ denotes Gaussian distribution with μ as the mean and σ^2 as the variance. $\langle ., . \rangle$ denotes the inner product between two vectors.

extended the problem of change point detection to an on-line setting, where the data is observed sequentially and the goal is to detect whether the change in distribution has occurred or not based on data observed until recently in time scale. That is, the data are observed upto time $n \in \mathbb{N}$, considered in batches of sliding length and Likelihood Ratio Test (LRT) is performed on the batches to decide whether the change in the distribution has occurred. The stopping time $\tau^{'}$ is given by:

$$\tau' = \min \left\{ t \in [1, n] : \max_{s \in [1,t)} L_{s:t} \geq c \right\} \tag{1.4}$$

where $L_{s:t} := \sum_{t'=s+1}^{t} \log \frac{\mathbb{P}_{\theta^{(1)}}(X_{t'})}{\mathbb{P}_{\theta^{(0)}}(X_{t'})}$ is the Likelihood Ratio and c is the threshold, given as input to LRT. τ is the change point, $\tau' - \tau$ is the detection delay

However it is important that the detection procedure need to be adaptive and should not depend on total number of observations . Lorden's paper [11] also assumes that pre-change and post-change parameters are perfectly known. Also, classical works assume that the changes are abrupt and there is no notion of *detectable changes*[1]. [18] considers decentralized quickest change detection where simultaneous change in the distribution of the data of multiple sensors are considered. Each sensor send a quantized data (i.e 1 bit) to a fusion center where quantized data from the sensors are collected. CUSUM is applied with quantized data to decide whether change in the distribution has occurred or not. However, there is notion of controlled sensing in the sense that there is no selection of reliable sensor to observe data at each time.

The problem of not knowing the pre-change and post-change parameters is handled by [16], by introducing a prior distribution to the unknown parameters. However, the resulted detection statistic is hard to compute recursively since the prior is not a conjugate. Recent work of Maillard[12] considers the sequential setup under the assumption that pre-change and post-change parameters are unknown (but under the assumption that the parameters belong to the exponential family) and Generalized Likelihood Ratio Test (GLRT) is used to detect the change in the distribution. The setup is sequential in the sense that at any given time t, GLRT is applied to the data observed until time t and if the GLR exceeds the threshold, alarm is raised and change in the distribution is declared.

Formally, at time t, having observed $X_1, X_2, \ldots, X_t$, detection time/stopping time τ' is given

[1]If the ℓ_2 norm of difference between the pre-change and post change parameter is greater than Δ, we call such differences as detectable changes, where $\Delta > 0$.

by the following boolean test:

$$\tau' = \mathbb{I}\left\{ \max_{s\in[0,t)} {G^{\epsilon}}_{1:s:t} \geq c \right\} \tag{1.5}$$

where GLR is defined as:

$$\begin{aligned} {G^{\epsilon}}_{1:s:t} :=& \max_{\theta_3,\theta_4} \left(\sum_{t'=1}^{s} \log \mathbb{P}_{\theta_3}(X_{t'}) + \sum_{t'=s+1}^{t} \log \mathbb{P}_{\theta_4}(X_{t'}) \right) \\ & - \max_{\theta} \left(\sum_{t'=1}^{t} \log \mathbb{P}_{\theta}(X_{t'}) \right) \end{aligned}$$

The threshold c is tuned by the high probability control over false alarm. Also, the notion of detectable changes is established and the paper gives non-asymptotic lower bound on detection delay under the assumption that change point is fixed. Contrary to the classical work, where the detection delay is calculated subject to an upper bound on expected false alarm, [12] calculates the detection delay subject to high probability control over false alarm and results show that detection delay achieves the asymptotic lower bound given by [9] and also extend the analysis to non asymptotic regime. However, there is no notion of controlled sensing of the samples.

1.4 Organization of the Thesis

We start with the preliminaries in chapter 2, that are relevant to this thesis. Followed by preliminaries, we will look at the novel idea of bandit change point detection in chapter 3 where we discusses two key approaches, namely Confidence Set based approach and GLRT based approach, to detect the change point through the sequence of actions that define the notion of *controlled* sensing. We will also look at the lower bound for the detection delay in 3 and some calculations on time varying threshold in chapter 4. We will look at some of the numerical results/simulations for the proposed approaches in chapter 5. Finally we conclude with quick remarks on challenges and future work in chapter 6.

Chapter 2

Preliminaries

In this chapter, we will look at the tools/preliminaries needed for the thesis. This chapter covers the following topics: Kullback-Leibler Divergence (KL Divergence), Optimization Background, Concentration Bounds, Sub-Gaussian Properties, Hypothesis Testing and Stochastic Linear Bandits.

Note: We skip the proofs that are advanced and provide references for them wherever appropriate.

2.1 Kullback-Leibler Divergence (KL Divergence)

The Kullback-Leibler divergence, between two probability distributions on a random variable is a measure of the distance between them. Formally, given two probability distributions $p(x)$ and $q(x)$ over a discrete random variable X, the relative entropy given by $D(p||q)$ is defined as follows:

$$D(p||q) = \sum_{x \in \mathcal{X}} p(x) \log \frac{p(x)}{q(x)}$$

In the definition above $0 \log \frac{0}{0} = 0 \log \frac{0}{q} = 0$ and $p \log \frac{0}{1} = \infty$.

2.1.1 Properties of KL Divergence

1. Divergence is not symmetric. That is, $D(p||q) = D(q||p)$ is not necessarily true.

2. Divergence is always non-negative. This is because of the following:

$$\begin{aligned} D(p||q) &= \sum_{x\in\mathcal{X}} p(x)\log\frac{p(x)}{q(x)} \\ &= -\sum_{x\in\mathcal{X}} p(x)\log\frac{q(x)}{p(x)} \\ &= -\mathbb{E}_p\left[\log\frac{q}{p}\right] \\ &\overset{a}{\geq} -\mathbb{E}_p\left[\frac{q}{p}\right] \\ &= -\log\left(\sum_{x\in\mathcal{X}} p(x)\log\frac{q(x)}{p(x)}\right) \\ &= 0 \end{aligned}$$

   (a) follows from the Jensen's inequality and the concavity of log. Therefore $\boxed{D(P||q) \geq 0}$.

3. Divergence is a convex function on the domain of probability distributions. Formally,

   Lemma 2.1 *(Convexity of divergence). Let p_1 , q_1 and p_2 , q_2 be probability distributions over a random variable X and $\forall\lambda \in (0,1)$ define*

$$\begin{aligned} p &= \lambda p_1 + (1-\lambda)p_2 \\ q &= \lambda q_1 + (1-\lambda)q_2 \end{aligned}$$

   Then, $D(p||q) \leq \lambda D(p_1||q_1) + (1-\lambda)D(p_2||q_2)$

   To prove the lemma, we shall use the log-sum inequality [5], which can be proved by reducing to Jensen's inequality:

   Proposition 2.1 *(Log-sum Inequality). If $a_1, \ldots, a_n, b_1, \ldots, b_n$ are non-negative numbers, then*

$$\sum_{i=1}^{n} a_i \log\frac{1}{b_i} \leq \left(\sum_{i=1}^{n}\right)\log\left(\frac{\sum_{i=1}^{n} a_i}{\sum_{i=1}^{n} b_i}\right)$$

   Proof: [of Lemma 2.1]

Let $a_1(x) = \lambda p_1(x), a_2(x) = (1-\lambda)p_2(x)$ and $b_1(x) = \lambda q_1(x), b_2(x) = (1-\lambda)q_2(x)$. Then,

$$\begin{aligned}
D(p||q) &= \sum_x (\lambda p_1(x) + (1-\lambda)p_2(x)) \log \frac{\lambda p_1(x) + (1-\lambda)p_2(x)}{\lambda q_1(x) + (1-\lambda)q_2(x)} \\
&= \sum_x (a_1(x) + a_2(x)) \log \frac{a_1(x) + a_2(x)}{b_1(x) + b_2(x)} \\
&\overset{a}{\leq} \sum_x \left(a_1(x) \log \frac{a_1(x)}{b_1(x)} + a_2(x) \log \frac{a_2(x)}{b_2(x)} \right) \\
&= \sum_x \left(\lambda p_1(x) \log \frac{\lambda p_1(x)}{\lambda q_1(x)} + (1-\lambda) p_2(x) \log \frac{(1-\lambda)p_2(x)}{(1-\lambda)q_2(x)} \right) \\
&= \lambda D(p_1||q_1) + (1-\lambda) D(p_2||q_2)
\end{aligned}$$

Therefore, $\boxed{D(p||q) \leq \lambda D(p_1||q_1) + (1-\lambda) D(p_2||q_2)}$ □

2.2 Optimization Background

The optimization is a vast topic and hence we shall look at the preliminaries of optimization that are used in this thesis. We restrict ourselves to unconstrained optimization of convex functions, Gradient Descent algorithm followed by constrained optimization of convex functions subject to convex constraints.

We also omit the properties of Convex functions, Convex sets and the references for the same can be found in [13, 3, 4, 2]

2.2.1 Unconstrained Convex Optimization

Let us consider the following minimization problem

$$\min_{x \in \mathbb{R}^d} \quad f(x) \tag{2.1}$$

where $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ is a twice differentiable convex and smooth[1] real valued function over $\mathbb{R}^d$. The critical point i.e, the solution to the above problem is obtained by solving the following equation

$$\nabla f(x)|_{x=x^*} = 0 \tag{2.2}$$

[1] A twice continuously differentiable function $f : \mathbb{R}^d \longrightarrow \mathbb{R}$ is said to be smooth if $||\nabla f(y) - \nabla f(x)|| \leq L\,||x-y||$, $\forall x, y \in \mathbb{R}^d$ and for some $L \in \mathbb{R}$

which is the necessary and sufficient condition for $x = x^* \in \mathbb{R}^d$ to be the minimum for the above minimization problem (2.1). The solution to (2.2) is called local minimum

Fact: Every local minimum is a global minimum for a convex function and it is unique for smooth convex functions[3, 2].

But solving (2.2) to obtain minima for convex function f is computationally expensive. To see this, let us consider a following example problem:

$$\min_{x \in \mathbb{R}^d} \quad \frac{1}{2} x^T A x - b^T x \tag{2.3}$$

where $A \in \mathbb{R}^{d \times d}$, $b \in \mathbb{R}^d$ and A is positive definite[1].

Let $f(x) = \frac{1}{2} x^T A x - b^T x$

$$\begin{aligned} \nabla f(x)|_{x=x^*} &= 0 \\ \implies Ax^* &= 0 \\ \implies x^* &= A^{-1} b \end{aligned}$$

Caveat: The computation of A^{-1}[2]is expensive, especially when d is very large. Complexity is $O(d^3)$

So the machinery to find x^* should be carried out in a "Principled" manner and one such algorithm that saves the computation to find minimum of f i.e, x^*, is **Gradient Descent**

2.2.2 Gradient Descent Algorithm

In Gradient Descent Algorithm, we start at an arbitrary point, which we call it as initial point and move along the gradient towards the next point, and repeat until converging to a critical point[3]. We follow an iterative scheme to achieve the same, which is described as follows:

$$x^{(k+1)} = x^{(k)} - \alpha_k \nabla f(x^{(k)}) \tag{2.4}$$

where α_k is the step-size, $\alpha_k \in (0, 1)$, $x^{(k)}$ is the current point and $x^{(k+1)}$ is the next point and $\nabla f(x^{(k)})$ is the gradient at point $x^{(k)}$, $k = 1, \ldots, T$.

(2.4) implies that we move along the direction of negative gradient (Descent direction of the function). In a smooth convex function f, the minima occurs at $\nabla f(x) = 0$ and hence we

[1] A matrix $A \in \mathbb{R}^{d \times d}$is positive definite iff $x^T A x > 0, \forall x \in \mathbb{R}^d$

[2] The inverse for A exists since A is assumed to be positive definite in this example

[3] Critical point is a point at which gradient of the function f is zero i.e, $\nabla f(x) = 0$

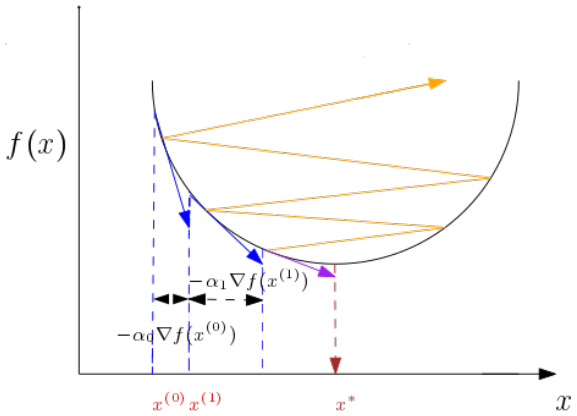


Figure 2.1: Illustration of Gradient Descent. Orange line indicates larger step sizes, which diverge from minima x^*. Purple line indicates smaller step size, which moves towards minima x^*.

move towards the minima in the negative direction of the gradient. The step size dictates the amount of descent that we make. Smaller the step size, slower will be the rate at which we reach towards minima. Larger the step size, larger will be the instability i.e, we keep oscillating around minima and would never reach minima. The figure 2.1 illustrates the Gradient Descent.

The convergence rate of Gradient Descent for a smooth convex function is $O\left(\frac{1}{\sqrt{T}}\right)$, where T is total number of steps in the iteration. Proof for the same can be found at [4, 13].

2.2.3 Constrained Optimization

Let us consider the following minimization problem

$$\begin{aligned} \min_{x \in S} \quad & f(x) \\ s.t \quad & h_j(x) \leq 0, j = 1, \ldots, l \\ & e_i(x) = 0, i = 1, \ldots, m \end{aligned} \tag{2.5}$$

where $S \subseteq \mathbb{R}^d$, f, h_j and e_i [1] are smooth and convex over $\mathbb{R}^d$. h_j are called inequality constraints and e_i are called equality constraints. Denote [2]

$$X = \left\{ x \in S : h_j(x) \leq 0, e_i(x) = 0, \forall i \in [m], \forall j \in [l] \right\}$$

[1] $f : \mathbb{R}^d \to \mathbb{R}$,$h_j : \mathbb{R}^d \to \mathbb{R}$,$e_i : \mathbb{R}^d \to \mathbb{R}$

[2] $[n] = \{1, \ldots, n\}$

where X is called feasible set. Hence the final goal of (2.5) can be posed as follows:
Goal: Minimize $f(x)$ subject to $x \in X$

Assume that $X \subseteq \mathbb{R}^d$ is non-empty. Now, let us define the **Lagrange function** as follows:

$$L(\lambda, \mu, x) = f(x) + \lambda^T \mathbf{h}(x) + \mu^T \mathbf{e}(x) \tag{2.6}$$

where $\mathbf{h}(x) = \begin{pmatrix} h_1(x) \\ \vdots \\ h_l(x) \end{pmatrix}$, $\lambda = \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_l \end{pmatrix}$, $\mu = \begin{pmatrix} \mu_1 \\ \vdots \\ \mu_m \end{pmatrix}$ & $\mathbf{e}(x) = \begin{pmatrix} e_1(x) \\ \vdots \\ e_m(x) \end{pmatrix}$. $\lambda_1, \ldots, \lambda_l$ and $\mu_1, \ldots, \mu_m$ are called Lagrange multipliers corresponding to inequality constraints $h_1, \ldots, h_l$ and equality constraints $e_1, \ldots, e_m$ respectively.

Notice that $L(.,.,x)$ is a convex function $\forall x \in X$ if $\lambda_j \geq 0, \forall j \in [l]$. Under certain conditions, called as **KKT** conditions, solving unconstrained minimization of L, i.e, solving

$$\min_{x \in \mathbb{R}^d} \quad L(\lambda^*, \mu^*, x) \tag{2.7}$$

is same as solving (2.5). λ^*, μ^* satisfy the **KKT** conditions, which are necessary and sufficient conditions for $x^* \in X$ to be the minimum for the problem (2.5). We now state the **KKT** conditions without proof, the proof of which can be found in [3].

KKT Conditions:

i $\nabla_x L(\lambda^*, \mu^*, x) = 0$

ii ${\lambda_j}^* h_j(x^*) = 0, \forall j \in [l]$ (Complementary Slackness)

iii ${\lambda_j}^* \geq 0, \forall j \in [l]$

2.3 Concentration Bounds

The central goal of any problem related to statistical learning is to know the actual mean of the data. Since the true mean of the data is unknown, it has to be learnt from the data in hand i.e, we need to estimate the mean of underlying data. The natural question that one would like to ask is how "good" the estimate of true mean. At high level, we will try to answer this question.

Note: Most of the materials covered in this section is heavily inspired from Chapter 5 of [10].

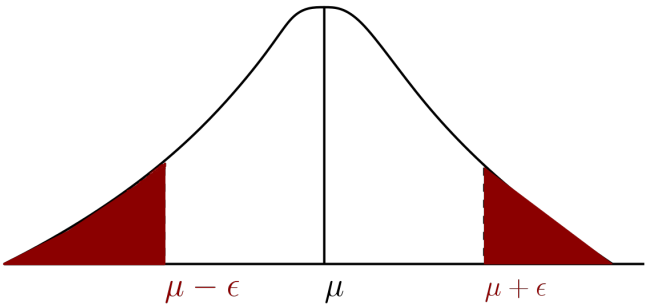


Figure 2.2: Illustration of Tail Probabilities. Shaded red part represent Tail Probabilities

Tail Probabilities Suppose that $X, X_1, \ldots, X_n$ is a sequence of independent and identically distributed random variables and assume that the true mean of this data is $\mu = \mathbb{E}[X]$ and variance $\sigma^2 = \mathbb{V}[X]$. The natural estimate of of true mean is:

$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i \tag{2.8}$$

which is called **sample mean** or **empirical mean**. Notice that $\mathbb{E}[\hat{\mu}] = \mu$, which means that $m\hat{u}$ is an unbiased estimator of μ. The variance calculation of $\hat{\mu}$ gives us the spread of distribution of $\hat{\mu}$ and it is equal to $\frac{\sigma^2}{n}$. This means that we expect the squared distance between μ and $\hat{\mu}$ to shrink as n grows larger at a rate of $\frac{1}{n}$ and scale linearly with variance of X, σ^2. But this does not tell us much about the distribution of the error $\mu - \hat{\mu}$. To do this, we analyse the probability that $\hat{\mu}$ overestimates or underestimates μ by more than some value $\epsilon > 0$. Precisely, we would like to ask how do the following quantities depend on ϵ

$$\mathbb{P}(\hat{\mu} \geq \mu + \epsilon) \quad \text{and} \quad \mathbb{P}(\hat{\mu} \leq \mu - \epsilon) \tag{2.9}$$

The expressions above (as a function of ϵ) are called **Tail Probabilities**. The first expression is called **Upper Tail Probability** and second one is called **Lower Tail Probability**. The term $\mathbb{P}(|\hat{\mu} - \mu| \geq \epsilon)$ is called **Two sided Tail Probability**. See figure 2.2 for the illustration of Tail Probabilities. In this section, we will restrict only to three important bounds on Tail Probability, which we list as follows.

2.3.1 Markov Inequality

Lemma 2.2 *For any non-negative random variable X and $\epsilon > 0$, the Markov inequality states that*

$$\mathbb{P}(X \geq \epsilon) \leq \frac{\mathbb{E}[X]}{\epsilon}$$

Proof: To justify the Markov inequality, let us fix a positive number a and consider the random variable Y_a defined by

$$Y_a = \begin{cases} 0, \text{ if } X < a, \\ a, \text{ if } X \geq a \end{cases}$$

It is clear that the relation $Y_a \leq X$ always holds and therefore,

$$\mathbb{E}[Y_a] \leq \mathbb{E}[X] \tag{2.10}$$

On the other hand,

$$\mathbb{E}[Y_a] = a\mathbb{P}(Y_a = a) = a\mathbb{P}(X \geq a) \tag{2.11}$$

Combining (2.10) and (2.9), we get

$$\boxed{\mathbb{P}(X \geq \epsilon) \leq \frac{\mathbb{E}[X]}{\epsilon}}$$

□

2.3.2 Chebyshev's Inequality

Lemma 2.3 *If X is a random variable with mean μ and variance σ^2, then*

$$\mathbb{P}(|X - \mu| \geq \epsilon) \leq \frac{\sigma^2}{\epsilon^2}$$

Proof: To justify the Chebyshev's inequality, we consider non-negative random variable $(X - \mu)^2$ and apply the Markov inequality with ϵ^2, i.e,

$$\mathbb{P}((X - \mu)^2 \geq \epsilon^2) \leq \frac{\mathbb{E}[(X - \mu)^2]}{\epsilon^2} = \frac{\sigma^2}{\epsilon^2}$$

The proof is completed by observing that the event $(X-\mu)^2 \geq \epsilon^2$ is identical to the event $|X-\mu| \geq \epsilon$. Therefore

$$\mathbb{P}((X-\mu)^2 \geq \epsilon^2) = \mathbb{P}(|X-\mu| \geq \epsilon)$$

$$\boxed{\mathbb{P}(|X-\mu| \geq \epsilon) \leq \frac{\sigma^2}{\epsilon^2}}$$

□

Using Chebyshev's inequality 2.3 in (2.10) gives us the the following bound:

$$\mathbb{P}(|\hat{\mu}-\mu| \geq \epsilon) \leq \frac{\sigma^2}{n\epsilon^2} \tag{2.12}$$

This result is good since it was so easily bought and relied on no assumptions other than the existence of the mean and the variance. However, the downside is this is asymptotic in nature and the bound is too loose. We shall now derive finite-time analogs, which are only possible by making Sub-Gaussian assumption.

Before proceeding to see what is a Sub-Gaussian random variable, let us see one more fundamental inequality on tail probability, which is called **Cramér-Chernoff** bound.

2.3.3 The Cramér-Chernoff Method

Suppose that X is a random variable and we want a bound for tail probability $\mathbb{P}(X \geq t)$. Let $\lambda > 0$. Then, we can bound the tail probability $\mathbb{P}(X \geq t)$ as follows. Sine $\lambda > 0$ and the exponential function is monotonically increasing, application of Markov Inequality yields

$$\begin{aligned}\mathbb{P}(X \geq t) \overset{a}{=} \mathbb{P}(\lambda X \geq \lambda t) &\overset{b}{=} \mathbb{P}(\exp \lambda X \geq \exp \lambda t) \\ &\overset{c}{\leq} \mathbb{E}[\exp \lambda X] \exp(-\lambda t)\end{aligned}$$

a and b follows from the fact that we are taking the probability of identical events. c follows from Markov inequality. Therefore,

$$\boxed{\mathbb{P}(X \geq t) \leq \min_{\lambda > 0} \mathbb{E}[\exp \lambda X] \exp(-\lambda t)} \tag{2.13}$$

Let us define $M_X(\lambda) = \mathbb{E}[\exp \lambda X]$. $M_X(\lambda)$ is called **Moment Generating Function (M.G.F)** of random variable, X. The name comes from the fact that moments of random variable X ($\mathbb{E}(X), \mathbb{E}(X^2) \ldots, \mathbb{E}(X^n)$ are called first order, second order, ..., n^{th} order moments respectively) can be generated by taking higher order derivatives of $M_x(\lambda)$. More details on

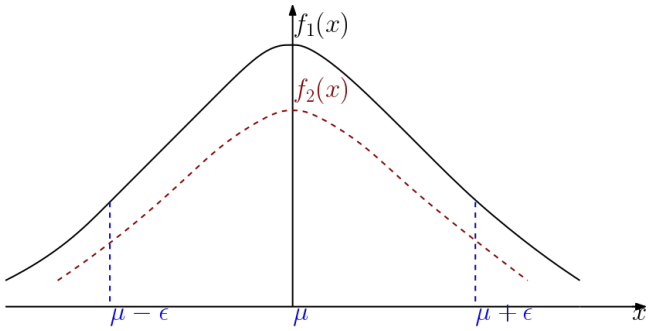


Figure 2.3: Illustration of Sub-Gaussian random variable. The red dotted graph shows Sub-Gaussian distribution and the black one represents Gaussian distribution. Notice that the tail of Sub-Gaussian is lighter as compared to Gaussian.

M.G.F can be found in standard probability texts such as [7, 8]. Inequality (2.13) is called Cramér-Chernoff bound.

Now, it is interesting to note that the tightness of the bound (2.13) depends on $M_X(\lambda)$. In particular, if, for any random variable X, it's **M.G.F** is bounded, then we can optimize (2.13) over λ to get tighter bound.

Let us now look at sub-Gaussian Random Variable, which has bounded **M.G.F**.

2.4 Sub-Gaussian Random Variable

Definition 2.1 *(Sub-Gaussianity). A random variable X is σ-sub-Gaussian if for $\forall \lambda \in \mathbb{R}$, it holds that $M_X(\lambda) = \mathbb{E}[\exp(\lambda X)] \leq \exp\left(\frac{\lambda^2\sigma^2}{2}\right)$.*

Gaussian random variable has a special property that it has bounded **M.G.F** $M_X(\lambda) = \exp\left(\frac{\lambda^2\sigma^2}{2}\right)$ and hence we can optimize over λ to get tighter bounds in (2.13). More generally, we can study the random variables whose **M.G.F**s are bounded above by **M.G.F** of Gaussian random variable. An illustration of Sub-Gaussian random variable is shown in figure 2.3

Remarks 2.1 *If for any random variable X, its **M.G.F** , $M_X(\lambda) = \infty$, then the tail probability is trivially bounded above by 1. We call such random variables **heavy tailed**. We call random variables with finite and bounded **M.G.F**, **light tailed**.*

Theorem 2.1 *If X is σ-Sub-Gaussian, then for any $\epsilon \geq 0$,*

$$\mathbb{P}(X \geq \epsilon) \leq \exp\left(-\frac{\epsilon^2}{2\sigma^2}\right) \tag{2.14}$$

Proof: We use Cramér-Chernoff method (2.13) to prove the theorem. Let $\lambda > 0$ be some constant to be tuned later. Then

$$\begin{aligned}\mathbb{P}(X \geq \epsilon) &= \mathbb{P}(\exp(\lambda X) \geq \exp(\lambda\epsilon) \\ &\overset{a}{\leq} \mathbb{E}[\exp(\lambda X)]\exp(-\lambda\epsilon) \\ &\overset{b}{\leq} \exp\left(\frac{\lambda^2\sigma^2}{2} - \lambda\epsilon\right)\end{aligned}$$

(a) is obtained by applying Markov inequality and (b) is obtained by using the definition 2.1 of Sub-Gaussianity. Now optimizing over λ by substituting $\lambda = \frac{\epsilon}{\sigma^2}$ we get

$$\boxed{\mathbb{P}(X \geq \epsilon) \leq \exp\left(-\frac{\epsilon^2}{2\sigma^2}\right)}$$

□

It is easy to show the same inequality for the lower tail. By using the union bound $\mathbb{P}(A \cup B) \leq \mathbb{P}(A) + \mathbb{P}(B)$, we see that $\mathbb{P}(|X| \geq \epsilon) \leq 2\exp(-\frac{\epsilon^2}{2\sigma^2}$. Setting the l.h.s of the latter inequality to $\delta \in (0, 1)$ gives more convenient and equivalent form

$$\mathbb{P}\left(|X| \geq \sqrt{2\sigma^2 \log\left(\frac{1}{\delta}\right)}\right) \leq \delta$$

It is easy to see that, for small δ, with high probability X takes the value in the interval

$$\left(-\sqrt{2\sigma^2 \log\left(\frac{1}{\delta}\right)}, \sqrt{2\sigma^2 \log\left(\frac{1}{\delta}\right)}\right)$$

In order to study the tail behaviour of $\hat{\mu} - \mu$ in (2.9), we need one more lemma, which we state without proof

Lemma 2.4 *Suppose that X is σ-Sub-Gaussian and X_1 and X_2 are independent and σ_1 and σ_2 Sub-Gaussian respectively, then:*

(a) $\mathbb{E}[X] = 0$ *and* $\mathbb{V}[X] \leq \sigma^2$.

(b) cX *is* $|c|\sigma$*-Sub-Gaussian* $\forall c \in \mathbb{R}$.

(c) $X_1 + X_2$ *is* $\sqrt{\sigma_1^2 + \sigma_2^2}$*-Sub-Gaussian*

Corollary 2.1 *Assume that $X_i - \mu$ are independent, σ-Sub-Gaussian random variables $\forall i \in [n]$. Then, for any $\epsilon \geq 0$,*

$$\mathbb{P}(\hat{\mu} \geq \mu + \epsilon) \leq \exp\left(-\frac{n\epsilon^2}{2\sigma^2}\right) \quad \textit{and} \quad \mathbb{P}(\hat{\mu} \leq \mu - \epsilon) \leq \exp\left(-\frac{n\epsilon^2}{2\sigma^2}\right)$$

where $\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$

Proof: By Lemma 2.4, it holds that $\hat{\mu} - \mu = \frac{1}{n}\sum_{i=1}^{n}(X_i - \mu)$ is $\frac{\sigma}{\sqrt{n}}$-Sub-Gaussian. Then by invoking Theorem 2.1 we get,

$$\boxed{\mathbb{P}(\hat{\mu} \geq \mu + \epsilon) \leq \exp\left(-\frac{n\epsilon^2}{2\sigma^2}\right) \quad \text{and} \quad \mathbb{P}(\hat{\mu} \leq \mu - \epsilon) \leq \exp\left(-\frac{n\epsilon^2}{2\sigma^2}\right)}$$

□

Setting the l.h.s of the inequality in 2.1 to $\delta \in (0, 1)$, we can get equivalent deviation form of the inequality in 2.1 as follows:

With probability at least $1 - \delta$

$$\mu \leq \hat{\mu} + \sqrt{\frac{2\sigma^2 \log\left(\frac{1}{\delta}\right)}{n}}$$

Symmetrically, with probability at least $1 - \delta$

$$\mu \geq \hat{\mu} - \sqrt{\frac{2\sigma^2 \log\left(\frac{1}{\delta}\right)}{n}}$$

We can invoke union bound to get two sided inequality $\mathbb{P}\left(|\hat{\mu} - \mu| \leq \sqrt{\frac{2\sigma^2 \log\left(\frac{1}{\delta}\right)}{n}}\right) \geq 1 - \delta$

Remarks 2.2 *The bound obtained in 2.1 is tighter than Chebyshev's (2.13) in the sense that* $\exp(-n\epsilon^2) \leq \frac{1}{n\epsilon^2}$. *Hence Sub-Gaussian assumption gives us tighter bounds on tail probabilities.*

2.5 Hypothesis Testing

Hypothesis Testing is a method of statistical inference that answers a particular question, based on the observed data. Let us try to understand the above statement via motivating examples:[1]

a) **Communication Systems**: Is there a message sent by the transmitter or not? (Based on the data at the receiver)

b) **Defense:** Is there an enemy plane or not?(Based on radar signals)

c) **Medical Science:** Is the patient suffering from a particular disease or not? (Based on medical observations)

d) **Economics:** Is the Indian economy in a recession? (Based on financial data)

The above examples give binary answers (Yes or No) to the questions that are asked on the underlying observed data. This is, in general, called Binary Hypothesis Testing and we, in this thesis, restrict ourselves to Binary Hypothesis Testing. Hypothesis refers to the belief about the particular question on the data (Yes or No in the above examples).

Types of Binary Hypothesis: Binary Hypothesis is usually classified into two types, namely,(i) **Null Hypothesis**, denoted by $\mathcal{H}_0$ and (ii) **Alternate/True Hypothesis**, denoted by $\mathcal{H}_1$. The former typically refers to the statistical insignificance in the data whereas latter refers to the statistical significance in the data. For instance, in chapter 1 the null hypothesis refers to no change in the distribution of the data whereas alternate/true hypothesis refers to the change in the distribution of the data. So Binary Hypothesis Testing is a statistical test procedure that declares either $\mathcal{H}_0$ or $\mathcal{H}_1$ based on the observed data.

Formally, let $X_1, X_2, \ldots, X_n$ be the set of independent and identically distributed random variables. Let $\mathbb{P}_0$ and $\mathbb{P}_1$ be two probability distributions on $X_1, \ldots, X_n$. Null hypothesis corresponds to the belief that $X_1, \ldots, X_n$ are observed according to the distribution $\mathbb{P}_0$ whereas True hypothesis corresponds to the belief that they are observed according to the distribution $\mathbb{P}_1$. i.e,

$$\mathcal{H}_0 : X_1, \ldots, X_n \overset{i.i.d}{\sim} \mathbb{P}_0$$

$$v.s$$

$$\mathcal{H}_1 : X_1, \ldots, X_n \overset{i.i.d}{\sim} \mathbb{P}_1$$

[1]Thanks to Dr. Aditya Gopalan, for the motivating examples, that appeared in the introductory lecture on Detection & Estimation Theory course at IISc, Bangalore

Goal: To declare either $\mathcal{H}_0$ or $\mathcal{H}_1$ after having observed $X_1, \ldots, X_n$, keeping the error as low as possible. The definition of error is given below.

There are well known testing procedures to achieve the above goal. However, we restrict ourselves to **Generalized Likelihood Ratio Test (G.L.R.T)**, which is a test procedure to achieve the above goal.

Error: Irrespective of the test procedures that we carry out to achieve the above goal, there are significant error scenarios that we run into. These errors are typically called **False Alarm or Type-I error** and **Missed Detection or Type-II error**. The former occurs when $\mathcal{H}_0$ is the actual hypothesis of the observed data but we declare $\mathcal{H}_1$ whereas the latter occurs when $\mathcal{H}_1$ is the actual hypothesis of the observed data but we declare $\mathcal{H}_0$. Formally, at an abstract level, we define False Alarm and Missed Detection as follows:

$$\begin{aligned} &\text{False Alarm (F.A)} : \mathbb{P}_0[\text{Decide}\,\mathcal{H}_1] \\ &\text{Missed Detection (M.D)} : \mathbb{P}_1[\text{Decide}\,\mathcal{H}_0] \end{aligned} \tag{2.15}$$

The events $\text{Decide}\,\mathcal{H}_1$ and $\text{Decide}\,\mathcal{H}_0$ correspond to the test procedure that we carry out. Another important thing to note is: we assumed that the distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ corresponding to the hypotheses $\mathcal{H}_0$ and $\mathcal{H}_1$ respectively. However this can be generalized to the case where both the distributions $\mathbb{P}_0$ and $\mathbb{P}_1$ are unknown. In such case, we assume that the distributions corresponding to $\mathcal{H}_0$ and $\mathcal{H}_1$ are characterized by the parameters $\theta^{(0)} \in \Theta^{(0)} \subseteq \mathbb{R}^d$ and $\theta^{(1)} \in \Theta^{(1)} \subseteq \mathbb{R}^d$ respectively, where $d \geq 1$. That is,

$$\begin{aligned} \mathcal{H}_0 : X_1, \ldots, X_n &\overset{i.i.d}{\sim} \mathbb{P}_{\theta^{(0)}} \\ v.s& \\ \mathcal{H}_1 : X_1, \ldots, X_n &\overset{i.i.d}{\sim} \mathbb{P}_{\theta^{(1)}} \end{aligned} \tag{2.16}$$

2.5.1 Generalized Likelihood Ratio Test (G.L.R.T)

Recall the setting (2.16). The Generalized Log Likelihood Ratio Test is defined as follows:

$$\text{Declare}\,\mathcal{H}_1 \text{ if } \log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}} \prod_{i=1}^n \mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}} \prod_{i=1}^n \mathbb{P}_{\theta^{(0)}}(X_i)}\right) \geq h, \text{ else Declare}\,\mathcal{H}_0 \tag{2.17}$$

where $h \geq 0$ is called threshold and the term $\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right)$ is called Generalized Log Likelihood Ratio (G.L.R). Note that

$$\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right) \geq h \overset{a}{\equiv} \max_{\theta^{(1)}\in\Theta^{(1)}}\sum_{i=1}^{n}\log\mathbb{P}_{\theta^{(1)}}(X_i) - \max_{\theta^{(0)}\in\Theta^{(0)}}\sum_{i=1}^{n}\log\mathbb{P}_{\theta^{(0)}}(X_i) \geq h \tag{2.18}$$

. The equivalent relation (a) follows from the product rule of log, followed by the fact that log is monotonically increasing function. In general, the distributions corresponding to $\mathcal{H}_1$ and $\mathcal{H}_0$ may not be known and hence we take Maximum Likelihood estimate (M.L estimate) of these distributions as given by $\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)$ and $\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)$. The product comes as a result of $i.i.d$ assumption and in more general setting the product is replaced by $\mathbb{P}_{\theta^{(0)}}(X_1,\ldots,X_n)$ and $\mathbb{P}_{\theta^{(1)}}(X_1,\ldots,X_n)$ corresponding to $\mathcal{H}_0$ and $\mathcal{H}_1$ respectively. Note that the events Decide $\mathcal{H}_1$ and Decide $\mathcal{H}_0$ in (2.15) corresponds to the events $\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right) \geq h$ and $\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right) < h$ respectively.

Choosing Threshold, h: The key step in G.L.R.T procedure is to choose appropriate threshold in (2.17). We invoke Neyman-Pearson lemma [14] to choose the threshold. That is, subject to allowable False Alarm (F.A) i.e, $\mathbb{P}_{\theta^{(0)}}\left(\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right) \geq h\right) \leq \delta$, where $\delta \in (0,1)$ defines the allowable F.A, we need to minimize Missed Detection (M.D) $\mathbb{P}_{\theta^{(1)}}\left(\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right) < h\right)$. From [14, 17] we can show that $\exists h > 0$ such that for $\mathbb{P}_{\theta^{(0)}}\left(\log\left(\frac{\max_{\theta^{(1)}\in\Theta^{(1)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(1)}}(X_i)}{\max_{\theta^{(0)}\in\Theta^{(0)}}\prod_{i=1}^{n}\mathbb{P}_{\theta^{(0)}}(X_i)}\right) \geq h\right) = \delta$, the decision rule (2.17) minimizes M.D and it is unique and optimal.

2.6 Stochastic Linear Bandits

In Multi-Armed Bandit setting [10], agent pulls an arm at each time $t \in [T]$ and receives a corresponding random reward X_t. The goal of the agent is to maximize the expected total

reward, i.e, maximize $\mathbb{E}[Reward(T)]$, where

$$Reward(T) = \sum_{t=1}^{T} X_t \tag{2.19}$$

X_t is the random reward earned at time t. In this setting, playing an arm allows the agent to learn only about that arm. The well known UCB (Upper Confidence Bound) algorithm's upper bound on the regret $\mathbb{O}\left(\sqrt{cKT\log T}\right)$, where $c > 0$ is a constant and $K \in \mathbb{N}$ is the number of arms. Note that the regret upper bound depends on the number of arms. Can we do better?

But in many practical applications, playing an arm allows agent to learn about the other arm which is not played. Many arms share the same reward structure. In this report, we consider one such special reward structure, which is linear. Bandits which have linear reward structure are called linear bandits and if the environment is stochastic (reward observed), we call this particular setting as Stochastic Linear Bandits. Each arm is represented by a unique feature vector in $\mathbb{R}^d$, where $d > 1 \in \mathbb{N}$ is the number of dimensions. We shall formally define the setting of stochastic linear bandits, as laid out by Yadkori et.al [1]. The detailed proof for the regret analysis of Stochastic Linear Bandits can be found in [1, 10] and the upper bound on the regret grows as $\mathbb{O}\left(\sqrt{cdT\log T}\right)$. Note that regret grows in the order of d as opposed to K in Stochastic Multi Armed Bandits, where d is number of dimensions.

2.6.1 Setting

At each round $t = 1, 2, 3, \ldots, T$:

- Agent plays $A_t \in \mathcal{A} \subseteq \mathbb{R}^d, \quad d = \#$ features.
- Agent observes reward:
  $X_t = \langle A_t, \theta_* \rangle + \eta_t$, where $\eta_t \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$ is the noise at time t.

θ_* is the unknown parameter in $\mathbb{R}^d$ that governs the reward of any chosen action $A_t \in \mathcal{A}$ at time t. The pseudo regret of the stochastic linear bandit is given by

$$Regret(T) = \sum_{t=1}^{T} \left\{ \max_{a \in \mathcal{A}} \langle a, \theta_* \rangle - \langle A_t, \theta_* \rangle \right\} \tag{2.20}$$

The goal is to play successive actions $A_t \in \mathcal{A}, \forall t \in [T]$ such that pseudo regret is minimum. Since we don't know the unknown parameter θ_*, we need to find an estimate for the same.

2.6.2 Estimating the Unknown Parameter

The regularized least square estimator is used to estimate the unknown parameter. Let $\lambda > 0$ be fixed. Then the estimate is given by:

$$\widehat{\theta}_t = \underset{\theta \in \mathbb{R}^d}{\operatorname{argmin}} \left\{ \sum_{s=1}^{t-1} (X_s - \langle A_s, \theta \rangle)^2 + \lambda ||\theta||_2^2 \right\}, \lambda > 0. \tag{2.21}$$

The solution to the above regularized least square estimate is given by:

$$\widehat{\theta}_t = V_t^{-1} \sum_{s=1}^{t} A_s X_s \tag{2.22}$$

where

$$V_t = V_0 + \sum_{s=1}^{t} A_s A_s^T, V_0 = \lambda I_{d \times d} \tag{2.23}$$

Now, we need to build set $C_t, \forall t \in [T]$ around the estimate $\widehat{\theta}_{t-1}$ such that with high probability, $\theta_* \in C_t$. we call this set (C_t) as confidence set. The confidence set C_t is given by:

$$C_t = \left\{ \theta \in \mathbb{R}^d : \left|\left| \theta - \hat{\theta}_{t-1} \right|\right|_{V_{t-1}} \leq \beta_t \right\} \tag{2.24}$$

where $\beta_t = O\left(\sqrt{d \log (t/\delta)} \right), \delta \in (0,1)$ [1].

The structure of C_t is an ellipse with center as $\hat{\theta}_{t-1}$. Illustration of the same is shown in figure 2.4

[1] $||x||_B := \sqrt{x^T B x}$, where B is a positive semi-definite matrix in $\mathbb{R}^d$ and $x \in \mathbb{R}^d$. $||x||_B$ is called Matrix-Norm of x with respect to matrix B

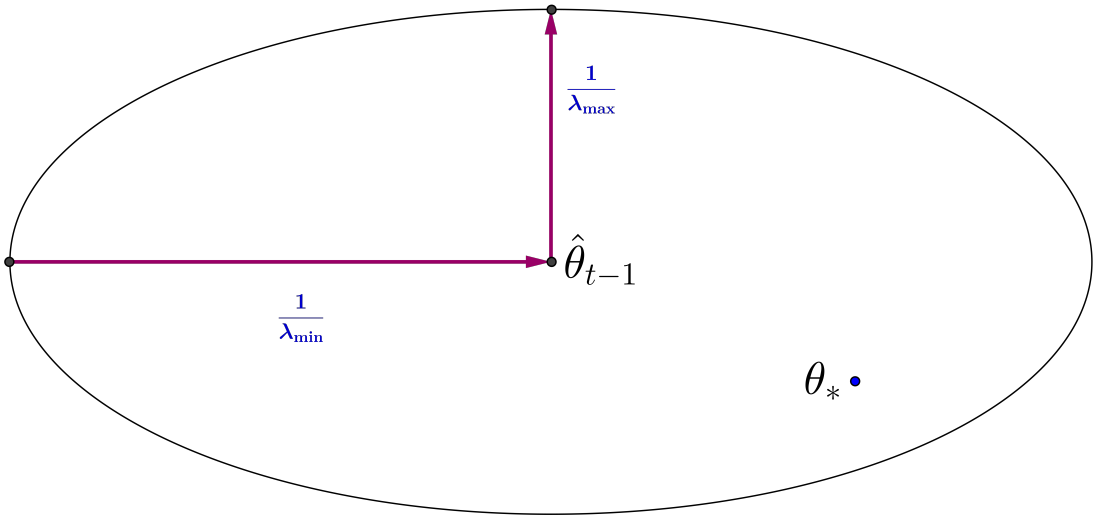


Figure 2.4: Confidence Set C_t. $\lambda_{\min}$ denotes smallest eigenvalue of V_{t-1} and $\lambda_{\max}$ denotes largest eigenvalue of V_{t-1} which are lengths of major axis and minor axis of ellipse C_t respectively.

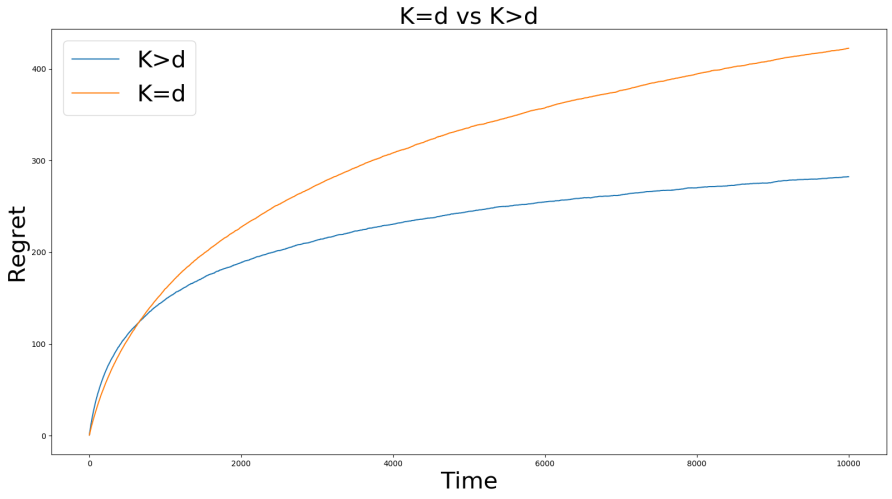


Figure 2.5: Regret Plot. The plot shows simple regret comparison between $K = d$ (Standard Stochastic Bandits case) and $K > d$ (Stochastic Linear Bandits case with number of arms greater than the dimension).

Chapter 3

Bandit Change Point detection

In this chapter, we will look at the novel bandit change point detection approaches. We consider two important approaches towards detecting the change point in the distribution, through Bandit framework: (i) Confidence Set based approach where we use Confidence Set around our estimate of the parameter of the distribution and propose stopping rule based on the location of pre-change parameter with respect to Confidence Set. (ii) We use GLRT (Generalized Likelihood Ratio Test) on the observed samples to detect the change i.e GLRT as stopping rule. We then build algorithms based on these two approaches.

3.1 Stochastic Linear Bandit Framework

In this report, we assume that pre-change parameter is known and post-change parameter is unknown and consider the high dimensional case ($d > 1$). A set of measurement devices/sensors is used to observe the data. We use the idea of high probability confidence set, which is the building block of Stochastic Linear Bandits [1] to define a valid stopping rule and propose algorithms (action rules) to detect the change. The confidence set also captures the notion of high probability control over false alarm and subject to this, the goal is to minimize the detection delay by playing successive actions from the given action set $\mathcal{A} \subset \mathbb{R}^d, d > 1$ sequentially over time. We also assume that the cardinality of the action set $|\mathcal{A}| > d$ and the action set is fixed.

We exploit the fact that the change point detection is a potential application of Stochastic Linear Bandits in the following way:

- Each measurement device/sensor can be viewed as an arm/action characterized by a low dimensional feature vector, $x \in \mathbb{R}^d$.

- The distributions before and after the change point under which the data are observed, can be characterized by the parameter vectors, $\theta^{(0)} \in \mathbb{R}^d$ (known) and $\theta^{(1)} \in \mathbb{R}^d$ (belongs to family $\Theta^{(1)} \subseteq \mathbb{R}^d$) respectively.
- Rewards are the linear combinations of the observations.
- The idea of confidence set, which is the building block of stochastic linear bandits [1], can be used to detect the change in the parameter.

The key difference between this work and other change point detection problems is that we have control over the samples, that are being observed, by choosing the suitable actions from the action set. The notion of reward in the case of Stochastic Linear Bandits is replaced by observations in this framework.

Control over observations/samples: As one can see, sample X_t at time t is distributed as :

$$X_t \sim \mathcal{N}\left(\left\langle A_t, \theta_t \right\rangle, \sigma^2\right) \tag{3.1}$$

By choosing different actions A_t at time t from the action set $\mathcal{A}$, we observe samples from the same distribution, but with different means, at time t. This gives us the control over the samples that are being observed in a time horizon.

3.2 Algorithms

We now propose two novel approaches to detect the change point. The first approach is based on the idea of confidence set [1] and the second approach is based on GLRT. These two approaches define the stopping rule, which is a prescription provided to any algorithm to detect the change. The ability to quickly detect the change depends on the algorithm, designed based on the stopping rule.

3.2.1 Confidence Set based approach

In this section, we propose the stopping rule based on the Confidence set, which is used in conventional Stochastic Linear Bandits [1], followed by algorithms to detect the change point, based on the stopping rule

3.2.1.1 Stopping Rule

Let us now define a stopping rule, S, which is as follows:

1. Build confidence set C_t with $\widehat{\theta}_t$ as the center, where $\widehat{\theta}_t$ is the point estimate of $\theta^{(0)}$, such that with high probability $\theta^{(0)} \in C_t$.

2. If $\theta^{(0)} \notin C_t$, the change point has occurred before time t.

Lemma 3.1 *S is a valid stopping rule*

Proof: Let $\theta^{(0)} \notin C_t$. Suppose the change has not occurred before time t. This implies that, with high probability $\theta^{(0)} \in C_t$, but this contradicts the fact that $\theta^{(0)} \notin C_t$. □

3.2.1.2 Sensing Rules

Since the actual change point denoted by τ is unknown, we may stop at time τ' greater than τ and hence there is a detection delay. Our goal is to keep this detection delay as small as possible. We now propose different algorithms to play successive actions $A_t \in \mathcal{A}$ for t $= \tau, \tau + 1 ... \tau'$ such that $\theta^{(0)} \notin C_{\tau'}$ and $\tau' - \tau$ is minimum.

These algorithms are then compared with random sampling which picks action one at a time uniformly at random. The proposed algorithms are specified below.

Sensing Rule 1: One of the possible sensing rules is to look at the position of $\theta^{(0)}$ from $\hat{\theta}_t$ at time t. For $t < \tau$, we know, from stopping rule S 3.1 and construction of confidence set in [1], that $\theta^{(0)}$ lies inside confidence set with high probability.

For $t \geq \tau$, the samples are drawn according to the distribution, parameterized by $\theta^{(1)}$, and hence $\hat{\theta}_t$ is no longer estimated based on $\theta^{(0)}$.

So, we need to choose actions successively over $t = 1, \ldots$ such that the actions make maximum magnitude of projection with $\theta^{(0)} - \hat{\theta}_t$. From lemma (3.1) and stopping rule S, $\theta^{(0)}$ will lie inside the ellipses for $t < \tau$, with high probability, irrespective of whatever actions that we choose until $t < \tau$. We now try to give pictorial and heuristic arguments to validate this sensing rule. Recall that C_t (2.24) is an ellipse with center as $\hat{\theta}_{t-1}$. The structure of the ellipse depends on V_t (2.22), which in turn depends on $\{A_1, \ldots, A_{t-1}\}$ (2.23). So, action chosen at time t, either shrinks or expands the ellipse C_t. In order to exclude $\theta^{(0)}$ as quickly as possible, we need to choose actions successively in the direction of $\hat{\theta}_{t-1} - \theta^{(0)}$ i.e, in the direction where the magnitude of the projection of action A_t at time t on $\hat{\theta}_{t-1}$ is maximum among all available actions at time t. The direction $\hat{\theta}_{t-1} - \theta^{(0)}$ defines the position of $\theta^{(0)}$ with respect to the ellipse C_t. Pictorial illustration of this sensing rule is given in the figure 3.1 and the pseudo-code is given below 1

Algorithm 1: Minimum angle between $\widehat{\theta}_t - \theta^{(0)}$ and action A_t at time t (Algorithm CS1)

Initialization: t=1,$\theta^{(0)}$, $V_1 = \mathbf{I}, S = \mathbf{0}$;
while $\theta^{(0)} \in C_t$ **do**
  for $a \in \mathcal{A}$ **do**
    $distance_{t+1}(a) = \ | \langle a, \widehat{\theta}_t - \theta^{(0)} \rangle |$
  end
  $A_t = \underset{a \in \mathcal{A}}{argmax} \quad distance_{t+1}(a)$
  $V_t = V_t + A_t A_t^T$
  if $t < \tau$ **then**
    $X_t \sim \mathcal{N}(\langle A_t, \theta^{(0)} \rangle, \sigma^2)$
  else
    $X_t \sim \mathcal{N}(\langle A_t, \theta^{(1)} \rangle, \sigma^2)$
  end
  $S = S + A_t X_t$
  $\widehat{\theta}_t = V_t^{-1} S$
  $t = t + 1$
end

Sensing Rule 2: Another possible sensing rule is to look at the position of $\theta^{(0)}$ from the closest boundary point of the ellipse C_t at every time $t = 1, \ldots$. Let $\theta_{t\min}$ be the closest boundary point to $\theta^{(0)}$. This is time varying and changes its position as the structure of the ellipse changes at each time t which can be seen from (2.24),(2.22),(2.23). Irrespective of the actions that we choose sequentially over time, for $t < \tau$, $\theta^{(0)}$ will lie in C_t with high probability, as it is evident from the construction of confidence set C_t and stopping rule S.

Following similar arguments from stopping rule 1, related to the structure of ellipse C_t, in order to exclude $\theta^{(0)}$ as quickly as possible once the change point occurs, we need to choose action in the direction of $\theta_{t\min} - \theta^{(0)}$. That is, by choosing the action that has maximum magnitude of projection or inner product with $\theta_{t\min} - \theta^{(0)}$. $\theta_{t\min} - \theta^{(0)}$ defines the position of $\theta^{(0)}$ with respect to ellipse C_t. Pictorial illustration of sensing rule 2 is given below 3.2

Finding the closest boundary point on the ellipse is the real challenge since this is a Quadratically Constrained Quadratic Programming (QCQP). However, we came up with some approximation followed by an iterative procedure (Gradient Descent) to find the closest boundary point.

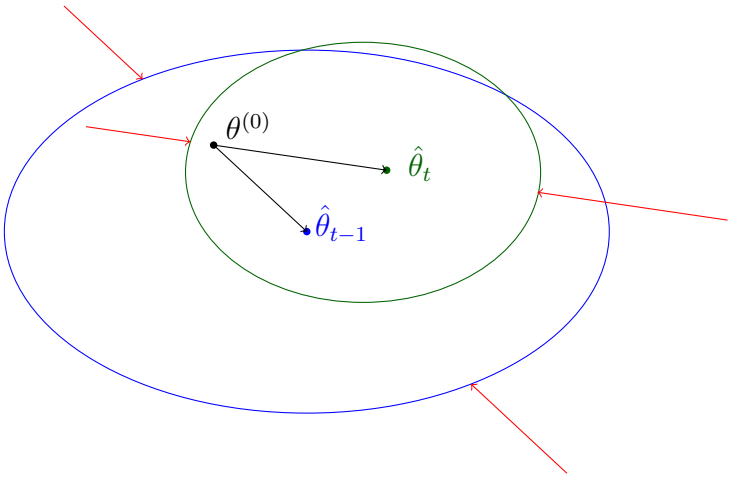


Figure 3.1: Pictorial representation of **Sensing Rule 1**. The blue coloured ellipse represents the ellipse C_t at time t. The red coloured arrow touching blue coloured ellipse is the action taken at time t. Note that this is the action that makes maximum magnitude of projection or inner product with $\theta^{(0)} - \hat{\theta}_{t-1}$. The green coloured ellipse the shrunken ellipse due to action A_t, that is picked according to Sensing Rule 1. Also, note that the center is moved due to change in V_{t-1} to $V_t = V_{t-1} + A_t A_t^T$.

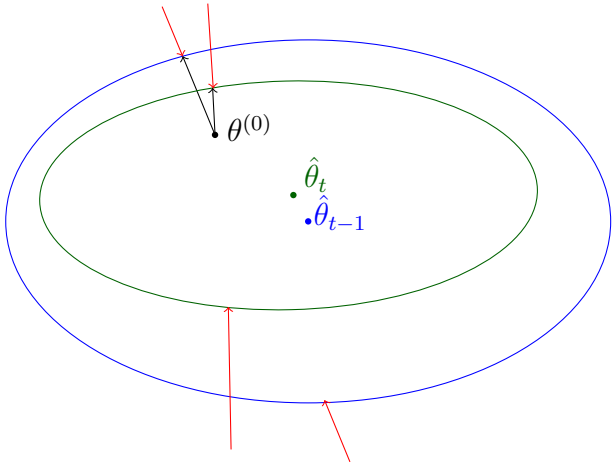


Figure 3.2: Pictorial illustration of **Sensing Rule 2**. The blue coloured ellipse represents the ellipse C_t at time t. The black arrow represents the projection of $\theta^{(0)}$ onto the surface of ellipse C_t, which is the closest boundary point $\theta_{t\min}$. The red coloured arrow denotes the action A_t taken at time t according to sensing rule 2, which makes maximum magnitude of projection or inner product with $\theta_{t\min} - \theta^{(0)}$. The green coloured ellipse represents the shrunken ellipse after taking action A_t at time t.

Let us formulate the closest boundary point problem as constrained optimization problem as follows:

$$\underset{\theta \in \mathbb{R}^d}{argmin} \quad \frac{1}{2}\left|\left|\theta - \theta^{(0)}\right|\right|_2^2 \tag{3.2}$$
$$\text{subject to } f(\theta) = 0$$

where $f(\theta') = (\theta' - \hat{\theta}_t)^T V_t (\theta' - \hat{\theta}_t) - {\beta_t}^2, \theta' \in \mathbb{R}^d$

Let the solution to the above optimization problem be θ_{tmin}
Define Lagrange function as:

$$L(\lambda, \theta) = \frac{1}{2}\left|\left|\theta - \theta^{(0)}\right|\right|_2^2 + \lambda\{(\theta - \hat{\theta}_t)^T V_t (\theta - \hat{\theta}_t) - \beta_t^2\} \tag{3.3}$$
$$\frac{\partial L(\lambda, \theta)}{\partial \theta} = 0 \tag{3.4}$$
$$\implies (\theta - \theta^{(0)}) = -\lambda \nabla f(\theta) \tag{3.5}$$

Expressing $f(\theta)$ using second order Taylor series expansion around $\theta^{(0)}$ gives:

$$\begin{aligned} f(\theta) =& f(\theta^{(0)}) + (\theta - \theta^{(0)})^T \nabla f(\theta^{(0)}) \\ &+ \frac{1}{2}(\theta - \theta^{(0)})^T V_t (\theta - \theta^{(0)}) = 0 \end{aligned} \tag{3.6}$$

Approximating $\nabla f(\theta^{(0)}) \approx \nabla f(\theta)$ and substituting in (9)

$$(\theta - \theta^{(0)}) = -\lambda \nabla f(\theta^{(0)}) \tag{3.7}$$

Substituting (3.7) in (3.6) gives

$$\lambda = \frac{f(\theta^{(0)}) + \frac{\beta_t^2}{2}}{||\nabla f(\theta^{(0)})||_2^2} \tag{3.8}$$

$$\theta_{tmin} = \theta^{(0)} - \frac{f(\theta^{(0)}) + \frac{\beta_t^2}{2}}{||\nabla f(\theta^{(0)})||_2^2} \nabla f(\theta^{(0)}) \tag{3.9}$$

Since the above solution is obtained through approximation, it may or may not be present on the ellipse. So we carry out an iterative procedure by taking equations (3.8) and (3.9) as initial values and do gradient descent on Lagrange function L. The obtained λ may be negative, but we take absolute value of λ to ensure that the Lagrange function convex quadratic. Now L is a function of θ and convex quadratic. We call this function as $L^{'}(\theta)$

We can also verify that $L^{'}(\theta)$ is ℓ-smooth where $\ell = ||I + 2\lambda V_t||_2$.

Lemma 3.2 *$L^{'}(\theta)$ is ℓ-smooth where $\ell = ||I + 2\lambda V_t||_2$.*

Proof: To verify that L' is ℓ-smooth, we need to show that $||\nabla L'(y) - \nabla L'(x)||_2 \leq \ell \, ||y - x||_2$.

$$\begin{aligned}
\nabla L^{'}(y) - \nabla L^{'}(x) &= (y - x) + 2\lambda\big[V_t(y - \hat{\theta}_t) - V_t(y - \hat{\theta}_t)\big] \\
&= (I + 2\lambda V_t)(y - x) \\
||\nabla L'(y) - \nabla L'(x)||_2 &\leq ||(I + 2\lambda V_t)||_2 \, ||(y - x)||_2
\end{aligned}$$

□

Now, we run Gradient Descent on $L^{'}$ with step size parameter [13] $\eta = \frac{1}{||(I+2\lambda V_t)||_2}$ i.e., $x_{t+1} = x_t - \eta \nabla L^{'}(x_t)$, with initial value $x_0 = \theta_{tmin}$. The pseudo-code of the sensing rule 2 is given below.

Algorithm 2: Minimum angle between $\theta_{t_{min}} - \theta^{(0)}$ and action A_t at time t (Algorithm CS2)

Initialization: t=1,$\theta^{(0)}$, $V_1 = \mathbf{I}, S = \mathbf{0}$;
while $\theta^{(0)} \in C_t$ **do**
  for $a \in \mathcal{A}$ **do**
    $distance_{t+1}(a) = \; | \langle a, \theta_{t_{min}} - \theta^{(0)} \rangle |$
  end
  $A_t = \underset{a \in \mathcal{A}}{argmax} \quad distance_{t+1}(a)$
  $V_t = V_t + A_t A_t^T$
  if $t < \tau$ **then**
    $X_t \sim \mathcal{N}(\langle A_t, \theta^{(0)} \rangle, \sigma^2)$
  else
    $X_t \sim \mathcal{N}(\langle A_t, \theta^{(1)} \rangle, \sigma^2)$
  end
  $S = S + A_t X_t$
  $\widehat{\theta}_t = V_t^{-1} S$
  $t = t + 1$
end

In stochastic linear bandits setting, by the construction of high probability confidence set around the pre change parameter, the probability of false alarm will be small and the goal boils down to keeping the probability of missed detection as low as possible. That is, in other words, keeping the expected detection delay as low as possible.

3.2.2 GLRT based approach

In this section, we discuss change point detection based on Generalized Likelihood Ratio Test (GLRT). We define stopping rule based on GLRT. GLR, specific to the problem, is given by:

$$\max_{\tau' \leq t} \max_{\theta^{(1)} \in \Theta^{(1)}} \sum_{s=\tau'}^{t-1} \log \frac{\mathbb{P}_{\theta^{(1)}}(X_s)}{\mathbb{P}_{\theta^{(0)}}(X_s)} \tag{3.10}$$

We first solve the inner optimization problem in (3.10). We make an assumption that $\Theta^{(1)}$ is a known convex set, specifically a half space, defined as follows:

$$\Theta^{(1)} = \{\theta \in \mathbb{R}^d \mid c^T\theta \leq d_1\} \tag{3.11}$$

where $c \in \mathbb{R}^d$ and $d_1 \in \mathbb{R}$ are known. The convex set dictates the notion of "detectable" changes i.e, what should be the minimum distance between pre-change and post-change parameter (in l_2 norm sense) so that the detection delay is finite ($< \infty$).

To begin with, let us define the objective function of the optimization problem to be $g_{t,\tau'}(\theta^{(1)})$, where

$$g_{t,\tau'}(\theta^{(1)}) = \sum_{s=\tau'}^{t-1} \log \frac{\mathbb{P}_{\theta^{(1)}}(X_s)}{\mathbb{P}_{\theta^{(0)}}(X_s)} \tag{3.12}$$

$$= (\theta^{(1)} - \theta^{(0)})^T \left(\sum_{s=\tau'}^{t-1} 2X_s A_s \right) - (\theta^{(1)} - \theta^{(0)})^T \left(\sum_{s=\tau'}^{t-1} A_s {A_s}^T \right) (\theta^{(1)} + \theta^{(0)}) \tag{3.13}$$

(3.13) is obtained by using (3.1)

Defining $b_{t,\tau'}$ and $V_{t,\tau'}$ as

$$b_{t,\tau'} = \sum_{s=\tau'}^{t-1} X_s A_s$$

$$V_{t,\tau'} = \sum_{s=\tau'}^{t-1} A_s A_s^T \tag{3.14}$$

.

After substituting b and V in (3.13), we get the final expression of $g_{t,\tau'}(\theta^{(1)})$ as

$$g_{t,\tau'}(\theta^{(1)}) = 2{\theta^{(1)}}^T b_{t,\tau'} - {\theta^{(1)}}^T V_{t,\tau'} \theta^{(1)} - 2{\theta^{(0)}}^T b_{t,\tau'} + {\theta^{(0)}}^T V_{t,\tau'} \theta^{(0)} \tag{3.15}$$

So, the inner optimization problem in (3.10) is posed as:

$$\hat{\theta}^{(1)} = \underset{\theta^{(1)} \in \Theta^{(1)}}{\operatorname{argmax}}\ g_{t,\tau'}(\theta^{(1)}) \tag{3.16}$$

Solving (3.16) is equivalent to solving the following optimization problem:

$$\hat{\theta}^{(1)} = \underset{\theta^{(1)} \in \Theta^{(1)}}{\operatorname{argmin}}\ {\theta^{(1)}}^T V_{t,\tau'} \theta^{(1)} - 2{\theta^{(1)}}^T b_{t,\tau'} \tag{3.17}$$

where $\Theta^{(1)}$ is defined as in (3.11). The above is a constrained convex optimization problem and we solve it using the well known Lagrange multiplier.

For (3.17), Lagrange function is given by:

$$L(\lambda, \theta^{(1)}) = {\theta^{(1)}}^T V_{t,\tau'} \theta^{(1)} - 2{\theta^{(1)}}^T b_{t,\tau'} + \lambda(c^T \theta^{(1)} - d_1) \tag{3.18}$$

KKT conditions for the above problem are:

1. $\nabla L(\lambda, \theta^{(1)})|_{\hat{\theta}^{(1)}, \lambda^*} = 0$
2. $\lambda^* \left(c^T \hat{\theta}^{(1)} - d_1 \right) = 0$ (Complementary slackness)
3. $\lambda^* \geq 0$

$$(1) \implies 2(V_{t,\tau'} \hat{\theta}^{(1)} - b_{t,\tau'}) + \lambda^* c = 0 \tag{3.19}$$

$$(2) \implies \lambda^* c^T \hat{\theta}^{(1)} = \lambda^* d_1 \tag{3.20}$$

using (3.19) in (3.20) we get

$$\lambda^* = \frac{2(\hat{\theta}^{(1)})^T(b_{t,\tau'} - V_{t,\tau'}\hat{\theta}^{(1)})}{d_1} \tag{3.21}$$

Substituting (3.21) in (3.19), we get the following final expression:

$$(V_{t,\tau'}\hat{\theta}^{(1)} - b_{t,\tau'}) + \underbrace{\frac{(\hat{\theta}^{(1)})^T(b_{t,\tau'} - V_{t,\tau'}\hat{\theta}^{(1)})}{d_1}}_{\text{scalar}} c = 0 \tag{3.22}$$

(3.22) implies that $(b - V\hat{\theta}^{(1)})$ is parallel to c.

$$\therefore (b_{t,\tau'} - V_{t,\tau'}\hat{\theta}^{(1)}) = \mu c \tag{3.23}$$

where $\mu \in \mathbb{R}$ is some scalar. Now,

$$\hat{\theta}^{(1)} = V_{t,\tau'}^{-1}(b_{t,\tau'} - \mu c) \tag{3.24}$$

If $V_{t,\tau'}^{-1}b_{t,\tau'}$ is not the minimum to the optimization problem (3.17), then by complementary slackness condition,

$$(\hat{\theta}^{(1)})^T c = d_1 \tag{3.25}$$

Substituting (3.24) in (3.25), we get

$$\mu = \frac{b_{t,\tau'}^T V_{t,\tau'}^{-1} c - d_1}{c^T V_{t,\tau'}^{-1} c} \tag{3.26}$$

Hence,

$$\hat{\theta}^{(1)} = V_{t,\tau'}^{-1}\left(b_{t,\tau'} - \left(\frac{b_{t,\tau'}^T V_{t,\tau'}^{-1} c - d_1}{c^T V_{t,\tau'}^{-1} c}\right) c\right) \tag{3.27}$$

Note that The inverse of $V_{t,\tau'}$ may not exist $\forall t, \tau^{'}$ since $V_{t,\tau'}$ is the summation of rank-1

matrices (3.14). So we solve (3.16) with regularization. That is, we solve

$$\underset{\theta^{(1)}\in\Theta^{(1)}}{\operatorname{argmax}}\; g_{t,\tau'}(\theta^{(1)}) + \beta\left|\left|\theta^{(1)}\right|\right|_2^2 \tag{3.28}$$

where $\beta > 0$. Following similar steps as (3.18),...,(3.31) we get solution to (3.28) as

$$\hat{\theta}^{(1)}_{PE} = V^{-1}_{t,\tau'}(\beta)\left(b_{t,\tau'} - \left(\frac{b^T_{t,\tau'}V^{-1}_{t,\tau'}(\beta)c - d}{c^T V^{-1}_{t,\tau'}(\beta)c}\right)c\right) \tag{3.29}$$

where,

$$V_{t,\tau'}(\beta) = V_{t,\tau'} + \beta I_{d\times d} \tag{3.30}$$

$\hat{\theta}^{(1)}_{PE}$ is the solution to optimization problem (3.28). We call this solution as Regularized Plug-in Estimate (PE) of unknown post-change parameter $\theta^{(1)}$ We now use this Regularized Plug-in Estimate in the GLR defined in (3.10) and use GLRT as the stopping rule to detect the change.

3.2.2.1 Stopping Rule

Stopping rule is given by standard GLRT which compares (3.10) with a positive threshold. If GLR exceeds this threshold, then we infer that the change has occurred. We use Plug-in Estimate of $\theta^{(1)}$ in (3.10). Formally,

$$\max_{\tau'\le t} g_{t,\tau'}(\hat{\theta}^{(1)}_{PE}) \ge h \tag{3.31}$$

where h is the threshold. The threshold is calculated subject to low probability false alarm. That is, under no change scenario ($\theta^{(0)} = \theta^{(1)}$),

$$\mathbb{P}\left\{g_{t,\tau'}(\hat{\theta}^{(1)}_{PE}) \ge h\right\} \le \delta \tag{3.32}$$

where $\delta \in (0,1)$. δ defines the desired level of false alarm probability. We need to calculate the threshold that satisfies (3.32). The exact computation of the threshold needs careful analysis of concentration bound and hence in this report we skip the analysis and calculate the threshold through Monte-Carlo simulation. However in Chapter 4, we will look at how we can possibly proceed to calculate the threshold.

Having defined the stopping rule, let us now look at the sensing rule that plays successive actions to detect the change based on the stopping rule as quickly as possible.

3.2.2.2 Sensing Rule

In the classical change point detection problem, the pre-change and post-change parameters are known and the asymptotic lower bound [11, 9] on the expected detection delay turns out to be

$$\mathbb{E}_{H_1}[\tau^{'} - \tau] \geq \frac{c_1 \log \frac{1}{\delta}}{D(\mathbb{P}_{\theta^{(1)}}, \mathbb{P}_{\theta^{(0)}})} \tag{3.33}$$

where $\tau^{'}$ is the time at which the change is detected, τ is the actual change point and $c_1 > 0$ is a constant.

The sample obtained at time t depends on the action A_t that is picked at time t. $D(.,.)$ denotes the KL divergence. Let π be the probability distribution over the arms $\mathcal{A} = \{A_1, \ldots, A_K\}$ and let π_i denote the prior on arm $A_i \in \mathcal{A}$. The sample X_t observed at time t, depends on the action taken at t and the action is drawn according to probability distribution π. That is

$$X_t \sim \begin{cases} \mathcal{N}(A_1^T\theta, \sigma^2)\, w.p & \pi_1 \\ \vdots & \\ \mathcal{N}(A_K^T\theta, \sigma^2)\, w.p & \pi_K \end{cases} \tag{3.34}$$

where $\pi_1, \ldots, \pi_K$ denote the prior on the actions $A_1, \ldots, A_K$ respectively and

$$\theta = \begin{cases} \theta^{(0)},\ t < \tau \\ \theta^{(1)},\ t \geq \tau \end{cases}$$

Note: We explicitly use $X_s \sim \mathcal{N}(A_s^T\theta, \sigma^2)$ in (3.10) for $s \leq t-1$ since we have history of actions and observations with us, observed until time $t-1$. This should not be confused with (3.34).

Now, $D(\mathbb{P}_{\theta^{(1)}}, \mathbb{P}_{\theta^{(0)}})$ can be explicitly written using (3.34) as follows:

$$D(\mathbb{P}_{\theta^{(1)}}, \mathbb{P}_{\theta^{(0)}}) = D(\sum_{i=1}^{K} \pi_i \mathbb{P}_{\theta^{(1)}}(i), \sum_{i=1}^{K} \pi_i \mathbb{P}_{\theta^{(0)}}(i)) \tag{3.35}$$

where $\mathbb{P}_{\theta^{(1)}}(i) = \mathcal{N}(A_i^T\theta^{(1)}, 1)$ and $\mathbb{P}_{\theta^{(0)}}(i) = \mathcal{N}(A_i^T\theta^{(0)}, 1)$, $\forall i \in [K]$

Using the convexity of $D(.,.)$ in (3.35), we get,

$$D(\sum_{i=1}^{K} \pi_i \mathbb{P}_{\theta^{(1)}}(i), \sum_{i=1}^{K} \pi_i \mathbb{P}_{\theta^{(0)}}(i)) \leq \sum_{i=1}^{K} \pi_i D(\mathbb{P}_{\theta^{(1)}}(i), \mathbb{P}_{\theta^{(0)}}(i)) \tag{3.36}$$

Using (3.36) in (3.33),

$$\mathbb{E}_{H_1}[\tau^{'} - \tau] \geq \frac{c_1 \log \frac{1}{\delta}}{\sum_{i=1}^{K} \pi_i D(\mathbb{P}_{\theta^{(1)}}(i), \mathbb{P}_{\theta^{(0)}}(i))} \tag{3.37}$$

In order to minimize the R.H.S of (3.37), the optimal action is to pick action A_i that has maximum $D(\mathbb{P}_{\theta^{(1)}}(i), \mathbb{P}_{\theta^{(0)}}(i))$. In (3.37), we use Regularized Plug-in Estimate of $\theta^{(1)}$, i.e, $\hat{\theta}^{(1)}_{PE}$ obtained in (3.29).

The above result follows from a hand wavy argument. One can show that (3.37) can be obtained through change of measure inequality [6]. The pseudo-code is given below

Algorithm 3: Plug-in Estimate (Algorithm PE)

Initialization: $t = 1, \theta^{(0)}, \eta, T, c, d$;

for $t = 1, \ldots T$ **do**

  for $\tau^{'} = 1, \ldots t-1$ **do**

    b=0,V=0 **for** $s = \tau^{'}, \ldots t-1$ **do**

      $b = b + X_s A_s$

      $V = V + A_s A_s^T$

    end

    $\hat{\theta}^{(1)} = \underset{\theta^{(1)} \in \Theta^{(1)}}{\text{argmax}} g_{t,\tau'}(\theta^{(1)})$ **if** $(g_{t,\tau'}(\hat{\theta}^{(1)}) > \eta)$ **then**

      change $= t$ **break**

    end

  end

  $A_t = \underset{A_i \in \mathcal{A}}{\text{argmax}} D(\mathbb{P}_{\theta^{(1)}}(i), \mathbb{P}_{\theta^{(0)}}(i))$

  if $t < \tau$ **then**

    $X_t \sim \mathcal{N}(\langle A_t, \theta^{(0)} \rangle, \sigma^2)$

  else

    $X_t \sim \mathcal{N}(\langle A_t, \theta^{(1)} \rangle, \sigma^2)$

  end

end

Output: change

Plots for 1, 2 and 3 are given in the chapter 5. They are also compared with naive Random Sampling algorithm where actions are taken uniformly at random. In chapter 5 we show this comparison.

Chapter 4

Calculation of Threshold in Bandit Change Point Detection

In this chapter, we will look at directions to compute threshold in Bandit Change Point Detection problem. The calculation is incomplete and require further attention. However let us see how can we possibly proceed.

4.1 Linear Bandit Setup

Parameter: Let $\theta^{(0)} \in \mathbb{R}^d$ be the parameter under which the observations are drawn before the change point. Let $\theta^{(1)} \in \Theta^{(1)} \subseteq \mathbb{R}^d$ be the parameter under which the observations are drawn after the change point. Let τ be the time at which $\theta^{(0)}$ changes to $\theta^{(1)}$, which we call it as the change point. Let θ_t be the parameter that governs the observation at time t which is defined as

$$\theta_t = \begin{cases} \theta^{(0)}, & \text{if } t < \tau \\ \theta^{(1)}, & \text{otherwise} \end{cases} \tag{4.1}$$

Actions: Let $\mathcal{A} \subseteq \mathbb{R}^d$ be the given set of actions in $\mathbb{R}^d$. Let $A_t \in \mathcal{A}$ be the action taken at time t.

Observations: Let X_t be the observations drawn at time t from the distribution $\mathbb{P}(X_t|A_t, \theta_t)$ given by $\mathbb{P}(X_t|A_t, \theta_t) = \mathcal{N}\left(\left\langle A_t, \theta_t \right\rangle, \sigma^2\right)$, where σ^2 is the known variance. i.e,

$$X_t = \left\langle A_t, \theta_t \right\rangle + \eta_t \tag{4.2}$$

where $\eta_t \overset{iid}{\sim} \mathcal{N}(0, \sigma^2)$

4.2 GLRT for Linear Bandit Setup

4.2.1 GLR

The Generalized Likelihood Ratio (GLR) for the Bandit is given by,

$$\max_{\tau' \leq t} \max_{\theta^{(1)} \in \Theta^{(1)}} \sum_{s=\tau'}^{t-1} \log \frac{\mathbb{P}_{\theta^{(1)}}(X_s)}{\mathbb{P}_{\theta^{(0)}}(X_s)} \tag{4.3}$$

We first solve the inner optimization problem in (4.3). We make an assumption that $\Theta^{(1)}$ is a known convex set, specifically a half space, defined as follows:

$$\Theta^{(1)} = \{\theta \in \mathbb{R}^d \mid c^T\theta \leq d\} \tag{4.4}$$

where $c \in \mathbb{R}^d$ and $d \in \mathbb{R}$ are known.

To begin with, let us define the objective function of the optimization problem to be $g_{t,\tau'}(\theta^{(1)})$, where

$$g_{t,\tau'}(\theta^{(1)}) = \sum_{s=\tau'}^{t-1} \log \frac{\mathbb{P}_{\theta^{(1)}}(X_s)}{\mathbb{P}_{\theta^{(0)}}(X_s)} \tag{4.5}$$

$$= (\theta^{(1)} - \theta^{(0)})^T \left(\sum_{s=\tau'}^{t-1} 2X_s A_s \right) - (\theta^{(1)} - \theta^{(0)})^T \left(\sum_{s=\tau'}^{t-1} A_s A_s^T \right) (\theta^{(1)} + \theta^{(0)}) \tag{4.6}$$

4.6 is obtained by using (4.2).

Defining $b_{t,\tau'}$ and $V_{t,\tau'}$ as

$$b_{t,\tau'} = \sum_{s=\tau'}^{t-1} X_s A_s$$

$$V_{t,\tau'} = \sum_{s=\tau'}^{t-1} A_s A_s^T \tag{4.7}$$

.

After substituting b and V in (20), we get the final expression of $g_{t,\tau'}(\theta^{(1)})$ as

$$\boxed{g_{t,\tau'}(\theta^{(1)}) = 2{\theta^{(1)}}^T b_{t,\tau'} - {\theta^{(1)}}^T V_{t,\tau'}\theta^{(1)} - 2{\theta^{(0)}}^T b_{t,\tau'} + {\theta^{(0)}}^T V_{t,\tau'}\theta^{(0)}} \tag{4.8}$$

So, the inner optimization problem in (4.3) is posed as:

$$\hat{\theta}^{(1)}_{PE} = \underset{\theta^{(1)} \in \Theta^{(1)}}{\operatorname{argmax}} \quad g_{t,\tau'}(\theta^{(1)}) \tag{4.9}$$

Recall from (3.16) of chapter 3 that the solution to the above problem (4.9) yields (3.27) that has $V^{-1}_{t,\tau'}$, where the inverse may not exist. So we solve (4.9) with regularization and recall that (3.28) solves the regularized version of (4.9) and we call the solution to this problem (4.9) as Regularized Plug-in Estimate $\hat{\theta}^{(1)}_{PE}$ (3.29) of $\theta^{(1)}$ Recall (4.2). $b_{t,\tau'}$ in (4.7) can be written, using (4.2) as:

$$\begin{aligned} b_{t,\tau'} &= \sum_{s=\tau'}^{t-1} (A_s^T \theta_t + \eta_s) A_s \\ &= \left(\sum_{s=\tau'}^{t-1} A_s A_s^T \right) \theta_t + \sum_{s=\tau'}^{t-1} A_s \eta_s \\ &= V_{t,\tau'}\theta_t + \sum_{s=\tau'}^{t-1} A_s \eta_s \end{aligned}$$

Therefore,

$$\boxed{b_{t,\tau'} = V_{t,\tau'}\theta_t + \sum_{s=\tau'}^{t-1} A_s \eta_s} \tag{4.10}$$

where θ_t comes from definition (4.1)

By plugging in the Regularized Plug-in Estimate $\hat{\theta}^{(1)}_{PE}$ in (4.8), we get,

$$\boxed{g_{t,\tau'}\left(\hat{\theta}^{(1)}_{PE}\right) = \left(\hat{\theta}^{(1)}_{PE} - \theta^{(0)}\right)^T \left\{ V_{t,\tau'}\left(\theta - \theta^{(0)}\right) + V_{t,\tau'}\left(\theta - \hat{\theta}^{(1)}_{PE}\right) + 2\sum_{s=\tau'}^{t-1} A_s \eta_s \right\}} \tag{4.11}$$

4.2.2 Stopping Rule

Stopping rule is given by standard GLRT which compares (4.3) with a positive threshold. If GLR exceeds this threshold, then we infer that the change has occurred. We use Plug-in Estimate of $\theta^{(1)}$ in (4.3). Formally,

$$\max_{\tau' \leq t} g_{t,\tau'}\left(\hat{\theta}^{(1)}_{PE}\right) \geq h \tag{4.12}$$

where h is the threshold.

4.3 Calculation of Threshold

The threshold is calculated subject to low probability false alarm. That is, under no change scenario ($\theta^{(0)} = \theta^{(1)}$),

$$\mathbb{P}\left\{ \max_{\tau' \leq t} g_{t,\tau'}\left(\hat{\theta}^{(1)}_{PE}\right) \geq h \right\} \leq \delta, \quad \text{for} \quad \tau' = 1, \ldots, t \tag{4.13}$$

where $\delta \in (0,1)$. δ defines the desired level of false alarm probability. We need to calculate the threshold that satisfies (4.13). Thus we need to 'control' false alarm with in the desired level δ by careful calculation of threshold.

Note: Threshold h at time t depends on the history at t given by $\mathcal{H}_t = \sigma(A_0, X_0, \ldots, A_{t-1}, X_{t-1})$.

4.3.1 Towards First Step

Let us first try to see simpler version of (4.13), i.e, for fixed τ', we will try to look at

$$\mathbb{P}\left\{ g_{t,\tau'}\left(\hat{\theta}^{(1)}_{PE}\right) \geq h \right\} \leq \delta \tag{4.14}$$

Suppose there is no change, i.e, $\theta_t = \theta^{(0)}$, $\forall t$. Then the $g_{t,\tau'}\left(\hat{\theta}^{(1)}_{PE}\right)$ becomes:

$$\begin{aligned} g_{t,\tau'}\left(\hat{\theta}^{(1)}_{PE}\right) &= \left(\hat{\theta}^{(1)}_{PE} - \theta^{(0)}\right)^T \left\{ V_{t,\tau'}\left(\theta^{(0)} - \hat{\theta}^{(1)}_{PE}\right) + 2\sum_{s=\tau'}^{t-1} A_s \eta_s \right\} \\ &= 2\sum_{s=\tau'}^{t-1} \alpha_s \eta_s - \left|\left|\hat{\theta}^{(1)}_{PE} - \theta^{(0)}\right|\right|^2_{V_{t,\tau'}} \end{aligned}$$

where $\alpha_s = \left\langle A_s, \hat{\theta}_{PE}^{(1)} - \theta^{(0)} \right\rangle$ Therefore,

$$\boxed{g_{t,\tau'}\left(\hat{\theta}_{PE}^{(1)}\right) = 2\sum_{s=\tau'}^{t-1}\alpha_s\eta_s - \left|\left|\hat{\theta}_{PE}^{(1)} - \theta^{(0)}\right|\right|_{V_{t,\tau'}}^2} \tag{4.15}$$

Let $X = \sum_{s=\tau'}^{t-1}\alpha_s\eta_s$. Let us also assume that the regularization parameter $\beta > 0$ in (3.30) is very small ($\beta << 1$) and approximate $\hat{\theta}_{PE}^{(1)} \approx V_{t,\tau'}^{-1}b_{t,\tau'}$, which is the solution to unconstrained version of (4.9). This is a crude assumption since the solution $V_{t,\tau'}^{-1}b$ may be close to $\theta^{(0)}$ in terms of Euclidean distance, in which case detection of the change point is difficult.

Calculation of α_s:

$$\begin{aligned}
\alpha_s &= \left\langle A_s, \hat{\theta}_{PE}^{(1)} - \theta^{(0)} \right\rangle \\
&\overset{a}{=} \left\langle A_s, V_{t,\tau'}^{-1}\left(V_{t,\tau'}\theta^{(0)} + \sum_{s=\tau'}^{t-1}A_s\eta_s\right) - \theta^{(0)} \right\rangle \\
&= \left\langle A_s, V_{t,\tau'}^{-1}\sum_{s=\tau'}^{t-1}A_s\eta_s \right\rangle
\end{aligned}$$

(a) is obtained by substituting the value of b from equation (4.10).

Therefore,

$$\boxed{\alpha_s = \left\langle A_s, V_{t,\tau'}^{-1}\sum_{s=\tau'}^{t-1}A_s\eta_s \right\rangle} \tag{4.16}$$

Calculation of $\left|\left|\hat{\theta}_{PE}^{(1)} - \theta^{(0)}\right|\right|^2_{V_{t,\tau'}}$:

$$\begin{aligned}
\left|\left|\hat{\theta}_{PE}^{(1)} - \theta^{(0)}\right|\right|^2_{V_{t,\tau'}} &= \left\langle \hat{\theta}_{PE}^{(1)} - \theta^{(0)}, V_{t,\tau'}\left(\hat{\theta}_{PE}^{(1)} - \theta^{(0)}\right)\right\rangle \\
&\overset{(a)}{=} \left\langle V_{t,\tau'}^{-1} b_{t,\tau'} - \theta^{(0)}, V_{t,\tau'}\left(V_{t,\tau'}^{-1} b_{t,\tau'} - \theta^{(0)}\right)\right\rangle \\
&= \left\langle V^{-1} b_{t,\tau'} - \theta^{(0)}, b_{t,\tau'} - V\theta^{(0)}\right)\right\rangle \\
&\overset{(c)}{=} \left\langle V_{t,\tau'}^{-1} \sum_{s=\tau'}^{t-1} A_s \eta_s, \sum_{s=\tau'}^{t-1} A_s \eta_s \right\rangle \\
&= \left|\left|\sum_{s=\tau'}^{t-1} A_s \eta_s\right|\right|^2_{V_{t,\tau'}^{-1}}
\end{aligned}$$

(a) is obtained by substituting $\hat{\theta}_{PE}^{(1)} \approx V_{t,\tau'}^{-1} b_{t,\tau'}$ and (c) is obtained by substituting the value of $b_{t,\tau'}$ using (4.10).

Therefore,

$$\boxed{\left|\left|\hat{\theta}_{PE}^{(1)} - \theta^{(0)}\right|\right|^2_{V_{t,\tau'}} = \left|\left|\sum_{s=\tau'}^{t-1} A_s \eta_s\right|\right|^2_{V_{t,\tau'}^{-1}}} \tag{4.17}$$

The term $\sum_{s=\tau'}^{t-1} A_s \eta_s$ in (4.16) and (4.17) is the modified S_t given in

$$S_t = \sum_{l=1}^{t} A_l \eta_l$$

But what we have in (4.16) and (4.17) is

$$S_{t-\tau'} = \sum_{l=\tau'}^{t-1} A_l \eta_l \tag{4.18}$$

Using (4.18), (4.16) and (4.17) in (4.14) we get the following:

$$\mathbb{P}\left\{ \left|\left|S_{t-\tau'}\right|\right|^2_{V^{-1}_{t,\tau'}} \geq h \right\} \leq \delta \tag{4.19}$$

Remarks 4.1 *$V_{t,\tau'}$ in [10, 1] is $V_{t,1}(\beta) = V_{t,1} + \beta I_{d\times d}$, where $\beta > 0$ is regularizer and $V_{t,1}(\beta)$ is invertible and $V_{t,\tau'}$ defined in (4.19) need not be invertible always (can be made invertible by introducing regularization, with regularizer to be extremely small).*

Remarks 4.2 *What we have in (4.19) is a deviation bound for a fixed $\tau^{'}$. We need to get final bound on (4.13), for random $\tau^{'}$. From [1, 10], the threshold h that is used to satisfy (4.19) ($\tau^{'} = 1$) is $h = 2\log\left(\frac{1}{\delta}\right) + \log\left(\frac{|V_{1,t-1}(\beta)|}{\beta^d}\right)$[1]. Hence, intuitively, the approximate threshold h to satisfy (4.19) is $h = 2\log\left(\frac{1}{\delta}\right) + \log\left(\frac{\underset{\tau'\leq t}{max}|V_{t,\tau'}(\beta)|}{\beta^d}\right)$, where $\beta << 1$*

[1] $|A|$ denotes the determinant of matrix $A \in \mathbb{R}^{d\times d}$

Chapter 5

Experiments and Observations

In this chapter, we will look at the experiments that are carried out to compare sensing rules, Sensing Rule 1 (1) and Sensing Rule 2 (2) whose stopping rule is based on Confidence Set, with naive Random Sampling, a sensing rule where actions are taken uniformly at random. We also compare the performance of stopping rules i.e, confidence set based approach and GLRT based approach under uniform sampling rule. We finally compare Plug-in Estimate sensing rule (3) with Random Sampling and we compare 1, 2 and 3.

5.1 Experiment 1 Setup

In this section, the histogram plots and scatter for Random Sampling and GLRT based sensing rule 3, Confidence Set based sensing rules 1 and 2 are attached, for change point $\tau = 100, 500, 1000$. We also use fewer actions to see The following are the list of parameters used:

- Number of dimension of the pre-change and post-change parameters $d = 2$ and number of arms/actions $K = 3 * d = 6$
- Pre-change parameter $\theta^{(0)} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$
- Post-change parameter $\theta^{(1)} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \end{bmatrix}$
- Hyper-plane parameters described by its normal c and intercept d_1 are taken to be

$c = \begin{bmatrix} -\frac{1}{\sqrt{2}} \\ 1 \\ -\frac{1}{\sqrt{2}} \end{bmatrix}$ and $d_1 = -1$. Hyperplane: $c^T\theta^{(1)} \leq d_1$.

- Action set $\mathcal{A} = \begin{bmatrix} 0 & 1 & \frac{1}{\sqrt{2}} & -1 & 1 & \frac{-1}{\sqrt{2}} \\ 1 & 0 & \frac{1}{\sqrt{2}} & 1 & -1 & \frac{-1}{\sqrt{2}} \end{bmatrix}$
- Number of iterations = 100.

Note: The threshold used is $h = 2\log\left(\frac{1}{\delta}\right) + \log\left(\frac{max_{\tau' \leq t}|V_{t,\tau'}(\beta)|}{\beta^d}\right)$, $\delta = 0.01$[1]. The experiment initially set to run for fixed time instants ($t \leq K$) to play all arms once and get initial estimate of $\theta^{(1)}$.

In the action set, ideally action 3 which is $\begin{bmatrix} \frac{1}{\sqrt{2}} \\ 1 \\ \frac{1}{\sqrt{2}} \end{bmatrix}$ is best action to play according to sensing rule 3 to achieve the detection delay as low as possible. We experimentally verify that sensing rule 3 indeed learns to pick action 3 and achieves better detection delay as compared to Random Sampling.

Note: The challenging part of GLRT based detection is to find appropriate threshold subject fixed low probability false alarm and we heuristically fix the threshold as given above so that false alarm in both sensing rule 3 and Random Sampling is same and low. Also this threshold is not time varying and it is fixed. Calculating exact threshold remains undone and the focus will be on the same in future. The pictorial description of arms and hyper-plane is given in figure 5.1

5.1.1 Plots corresponding to Confidence Set (CS) based approach

In this section, we will look at the histogram and scatter plots, comparing Random Sampling rule and 1 and 2. We also compare 1 and 2.

5.1.2 Plots corresponding to GLRT based approach

In this section, we will look at the histogram and scatter plots, comparing Random Sampling rule and 3. We also look at histogram plot of arms that are pulled during the series of episodes(runs).

[1] $|A|$ denotes the determinant of matrix $A \in \mathbb{R}^{d \times d}$

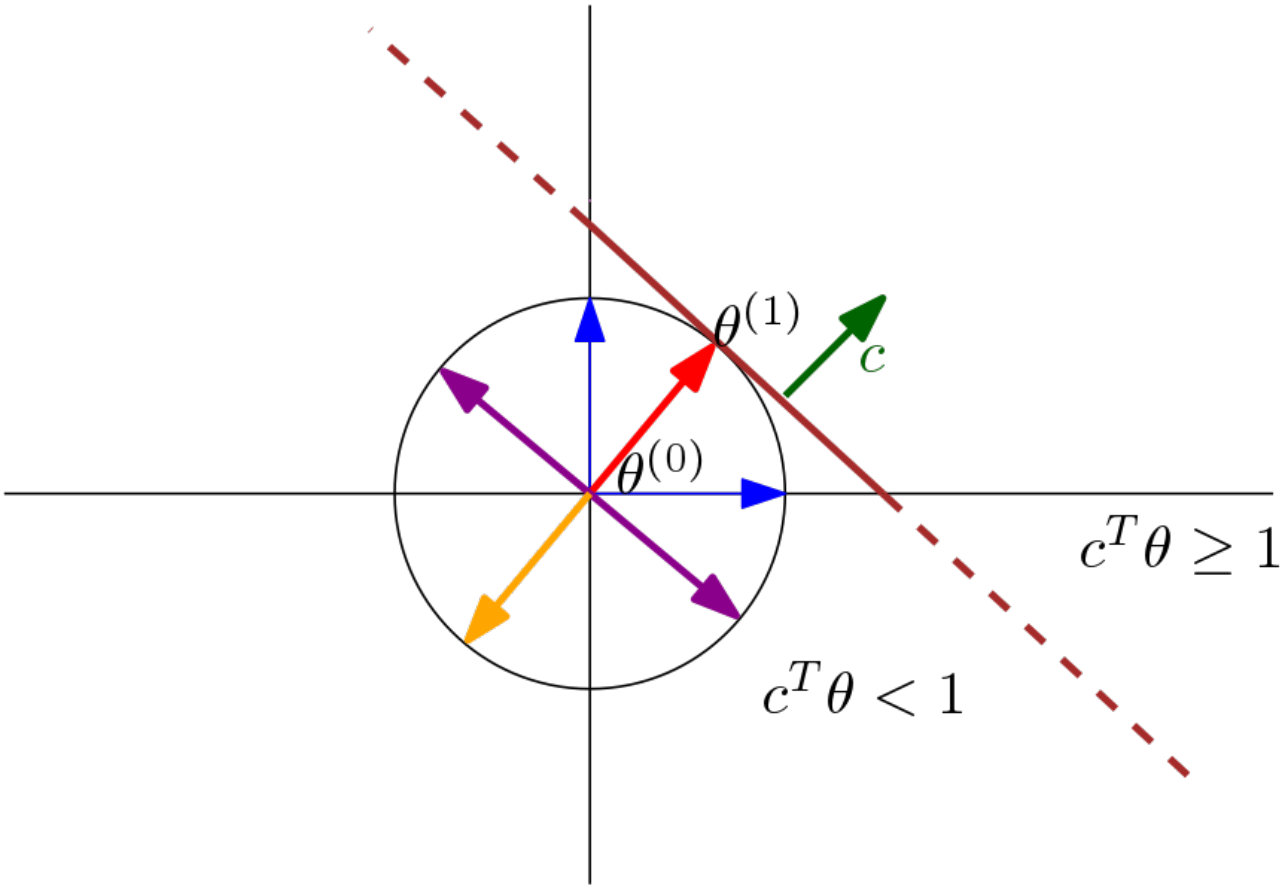
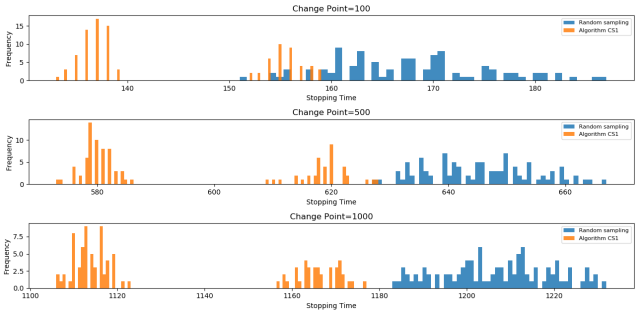
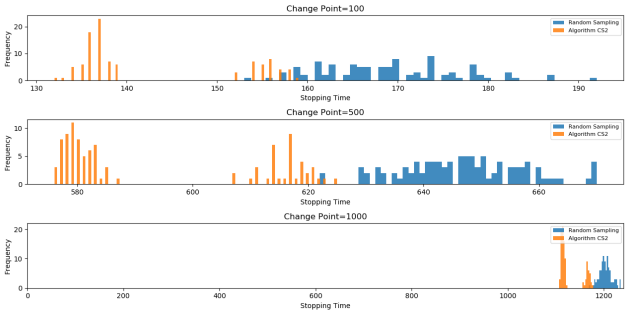


Figure 5.1: Description of action set and hyper-plane in the experiment setup 1. Red vector illustrate action 3 in the action set $\mathcal{A}$. c is the normal to the hyper-plane. Purple coloured vectors represent actions which are perpendicular to the direction $\theta^{(1)} - \theta^{(0)}$. Blue vectors represent standard basis vectors in $\mathbb{R}^2$. The brown coloured line represents the hyper-plane



(a) Comparison of Algorithm CS1 1 and Random Sampling with respect to stopping time. Orange bar represents Algorithm CS1 1 and blue bar represents Random Sampling



(b) Comparison of Algorithm CS1 2 and Random Sampling with respect to stopping time. Orange bar represents Algorithm CS2 2 and blue bar represents Random Sampling.

Figure 5.2: Histogram plots of sensing rules 1, 2 and Random Sampling corresponding to Confidence Set based stopping rule . CS denotes Confidence Set. Horizontal axis represent stopping time of sensing rules and vertical axis represents frequency of stopping at specific stopping time

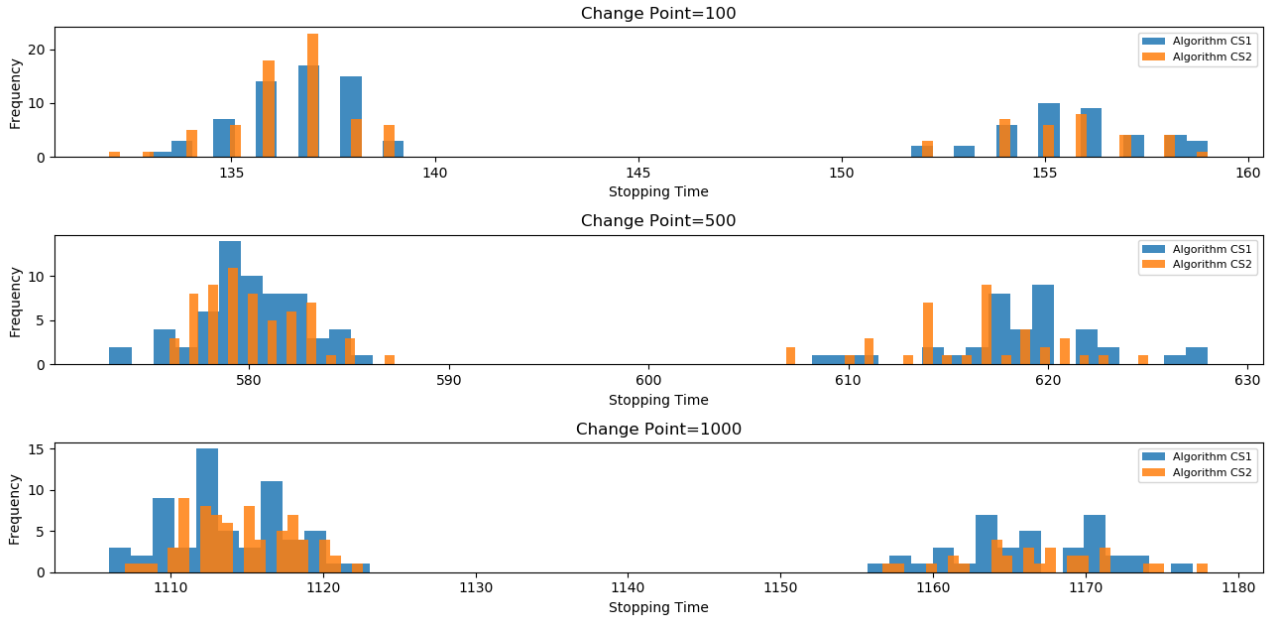
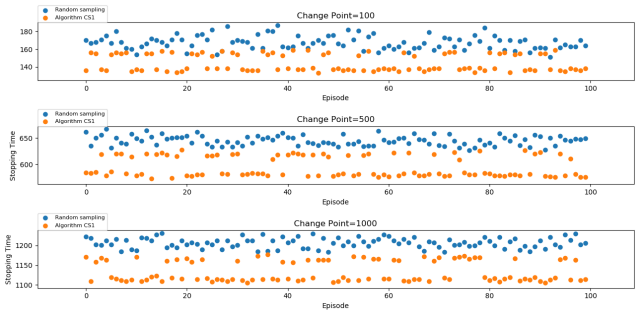
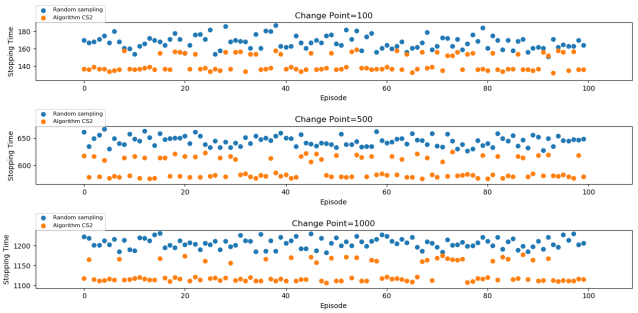


Figure 5.3: Histogram plot comparison of Algorithm CS1 1 and Algorithm CS2 2 with respect to stopping time. Orange bar represents Algorithm CS2 2 while blue bar represents Algorithm CS1 1



(a) Comparison of Algorithm CS1 1 and Random Sampling with respect to stopping time. Orange dots represents Algorithm CS1 1 and blue dots represents Random Sampling



(b) Comparison of Algorithm CS1 2 and Random Sampling with respect to stopping time. Orange dots represents Algorithm CS2 2 and blue dots represents Random Sampling.

Figure 5.4: Scatter plots of sensing rules 1, 2 and Random Sampling corresponding to Confidence Set based stopping rule . CS denotes Confidence Set. Horizontal axis represents episode and verical axis represents the stopping time. An episode corresponds to one simulation run of Experiment 1

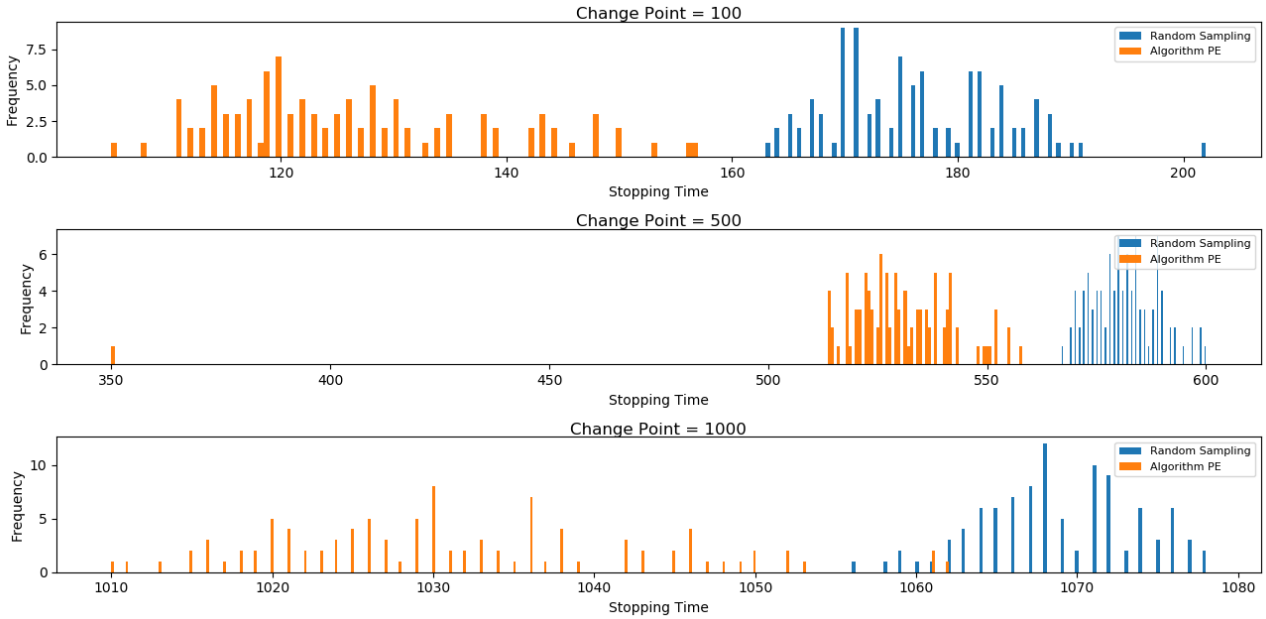


Figure 5.5: Histogram plot comparison of Algorithm PE 3 and Random Sampling with respect to stopping time. Orange bar represents Algorithm PE 3 and blue bar represents Random Sampling

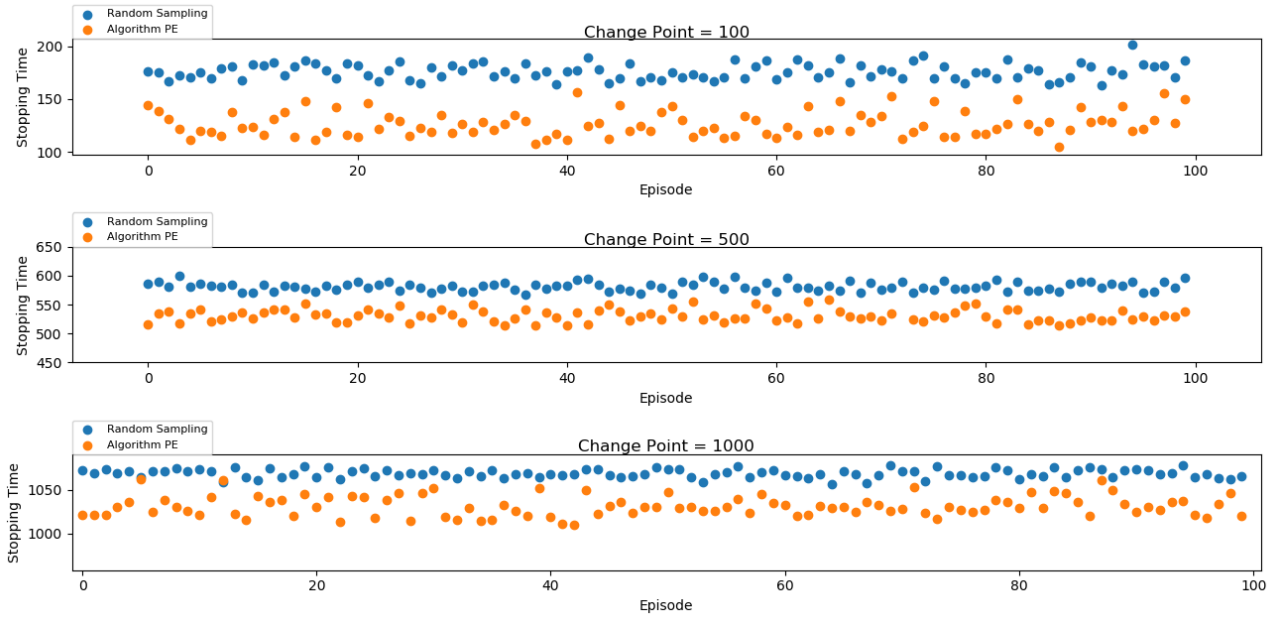


Figure 5.6: Scatter plot comparison of Algorithm PE 3 and Random Sampling with respect to stopping time. Orange dots represents Algorithm PE 3 and blue dots represents Random Sampling

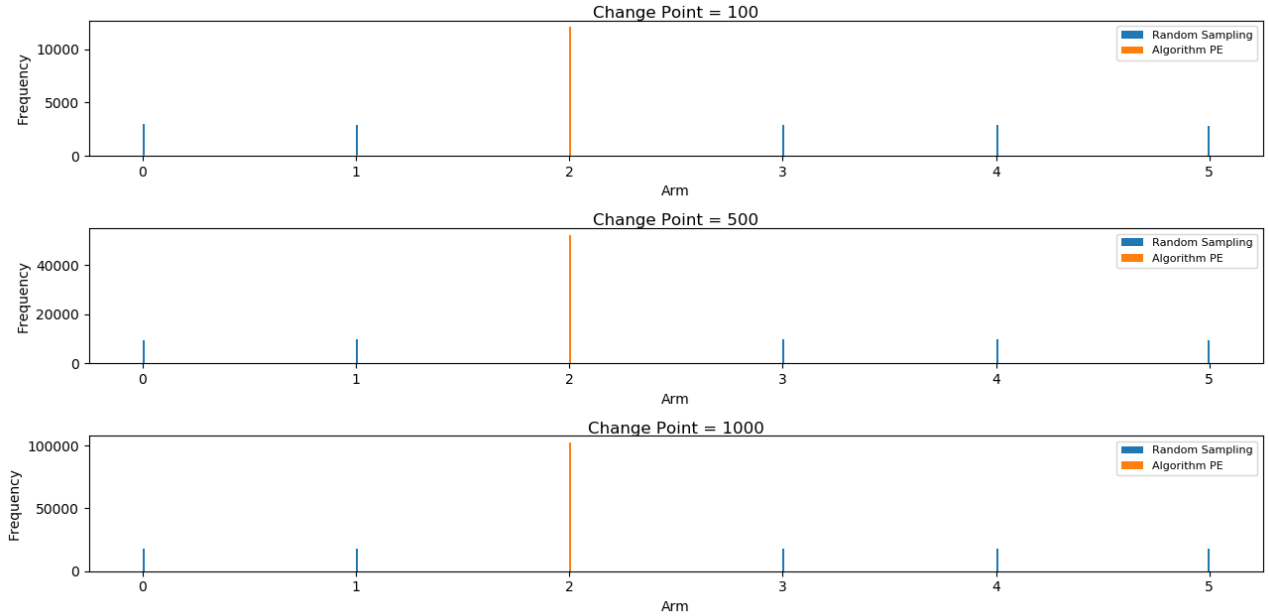


Figure 5.7: Histogram plot comparison of arms/actions pulled/taken over series of episodes. It is interesting to see action 3 (denoted by 2 in the horizontal axis) of action set $\mathcal{A}$ is sampled the most, which aligns itself along the direction of $\theta^{(1)} - \theta^{(0)}$. Horizontal axis represents Arm/Action index corresponding to action set $\mathcal{A}$ and vertical axis represents frequency of pull.

5.2 Comparison of Confidence Set (CS) and GLRT based approaches

In this section, we will look at comparison of CS and GLRT based approaches. Towards this, we first compare (both histogram and scatter plots are attached for better visualization) Random Sampling rule, whose stopping time is based on CS and GLRT. We experimentally verify that GLRT stopping rule is quickest (optimal in this sense) in terms of detection delay. We then compare Algorithm PE 3 against Algorithm CS1 1 and Algorithm CS2 2. We see that, via experiment, Algorithm PE is quickest in terms of detection delay as compared to Random Sampling rule, Algorithm CS1 1 and Algorithm CS2 2.

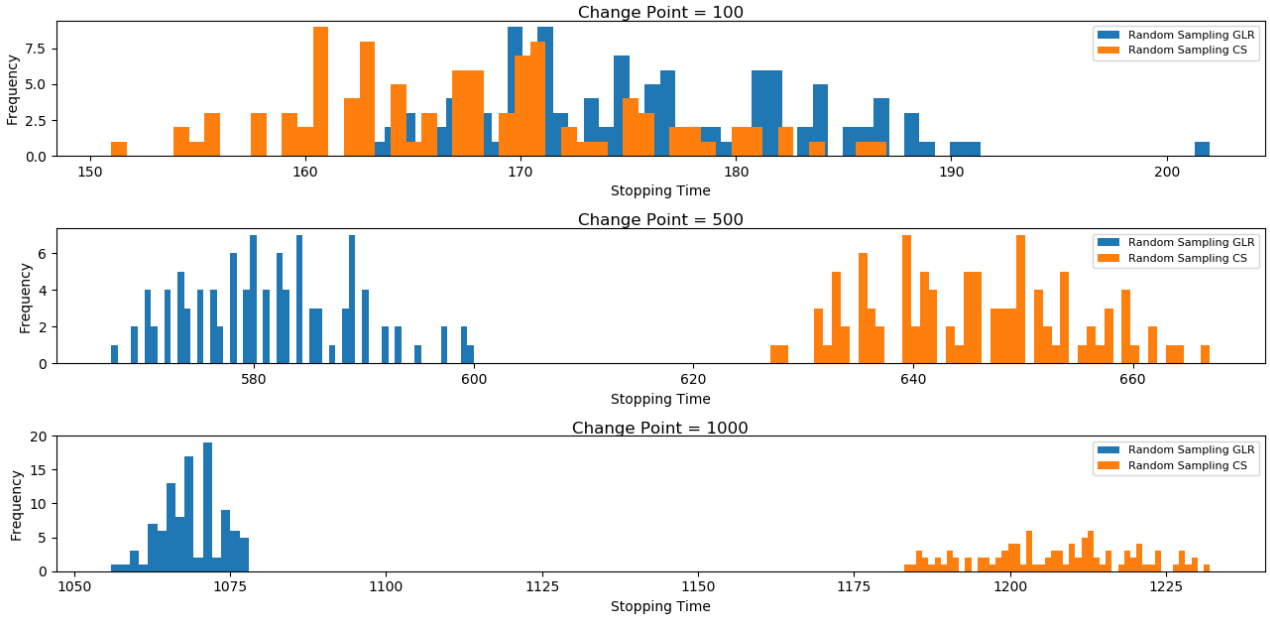


Figure 5.8: Histogram plot comparison of Random Sampling rule with CS and GLRT as stopping rules. Blue bar represents Random Sampling rule with GLRT as stopping rule while orange bar represents Random Sampling rule with CS as stopping rule. Horizontal axis represents stopping time and vertical axis represents frequency of stopping. CS denotes Confidence Set.

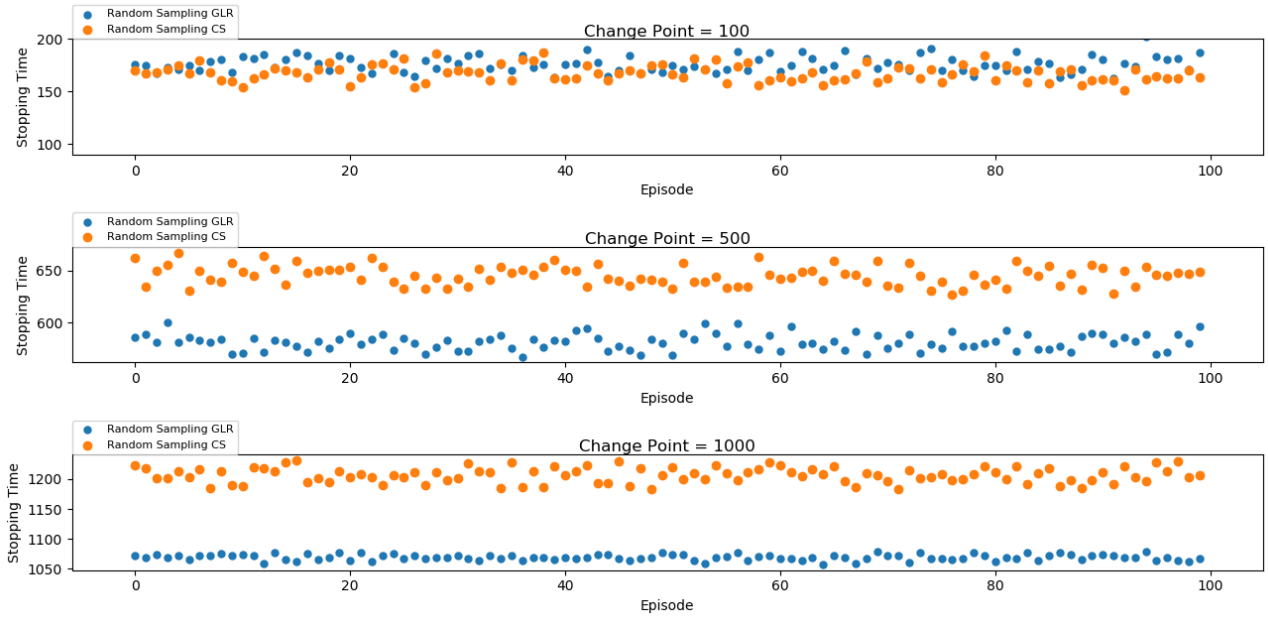


Figure 5.9: Scatter plot comparison of Random Sampling rule with CS and GLRT as stopping rules. Blue dots represent Random Sampling rule with GLRT as stopping rule while orange dots represent Random Sampling rule with CS as stopping rule. Horizontal axis represents episodes and vertical axis represents stopping time. CS denotes Confidence Set.

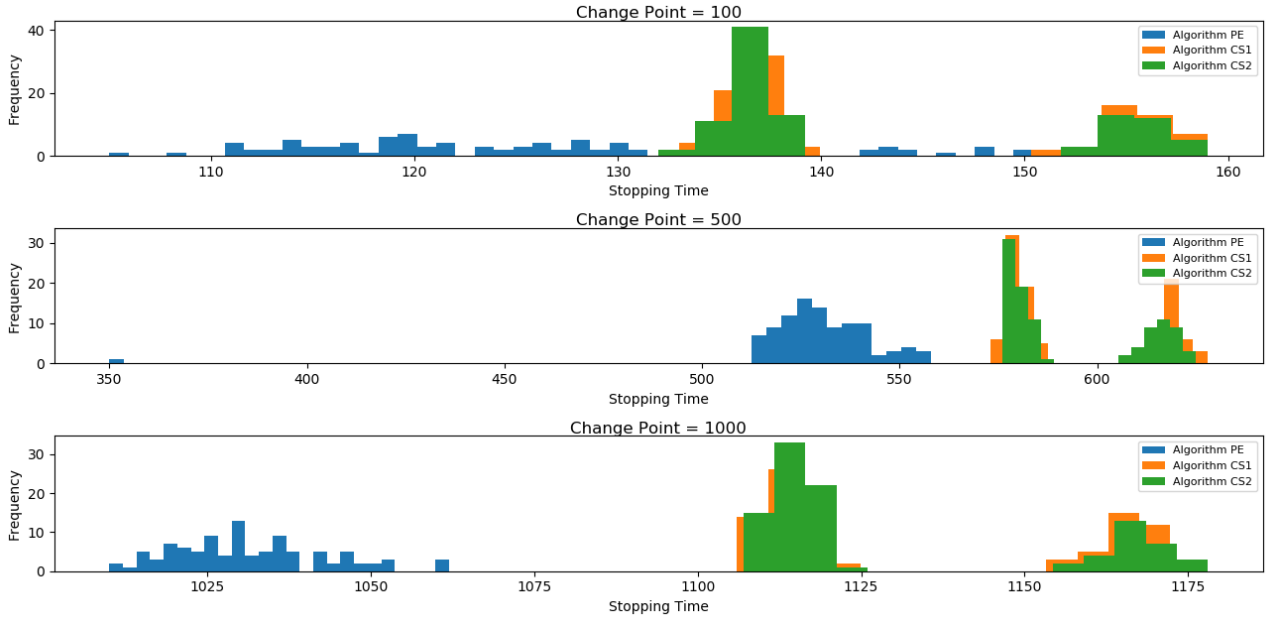


Figure 5.10: Histogram plot comparison of Algorithm PE 3, Algorithm CS1 1 and Algorithm CS2 2. Horizontal axis represents stopping time and vertical axis represents frequency of stopping. CS denotes Confidence Set.

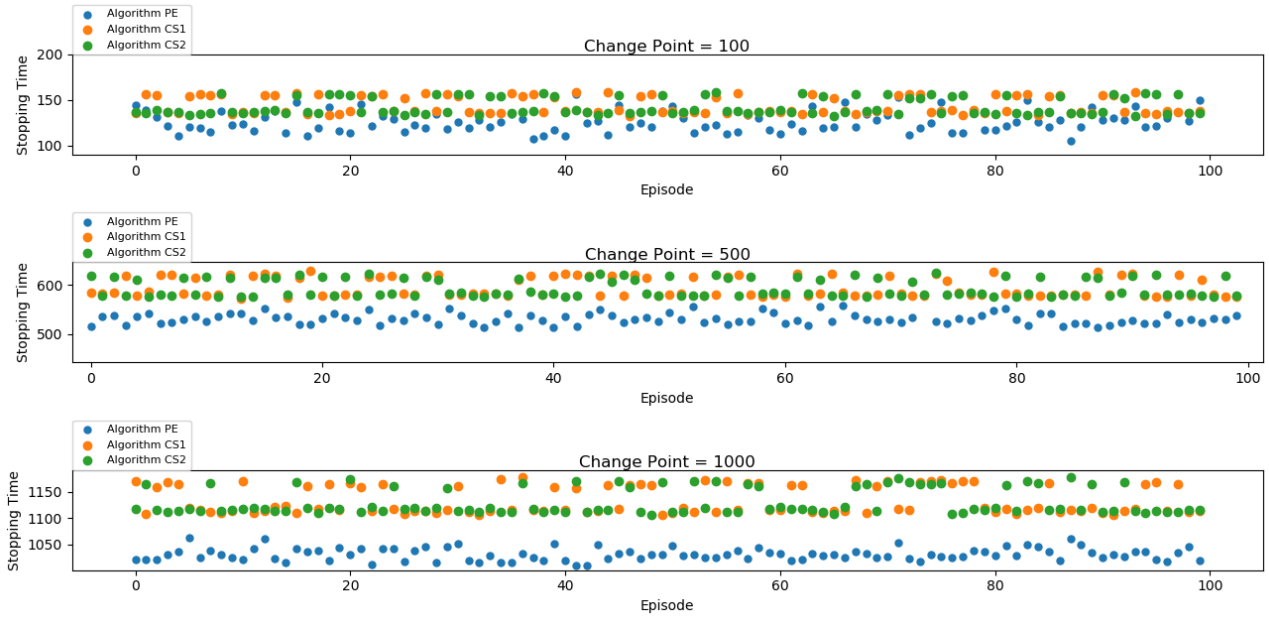


Figure 5.11: Scatter plot comparison of Algorithm PE 3, Algorithm CS1 1 and Algorithm CS2 2. Horizontal axis represents episodes and vertical axis represents stopping time. CS denotes Confidence Set.

Chapter 6

Summary

To conclude, we have proposed novel change point detection procedures by introducing controlled sensing of samples. We call this controlled sensing of samples followed by change detection procedure as bandit change point detection. We introduced two stopping rules to detect the change in the distribution, namely Confidence Set based stopping rule and GLRT based stopping rule. Based on these stopping rules, we proposed algorithms/sensing rules to detect the change based on the stopping rule. We also looked at the calculation of threshold, though incomplete but provided some insights as to how one can proceed to calculate threshold in GLRT based approach in bandit change point detection problem. We have also seen a experimental demonstration of proposed stopping rules and sensing rules and observed the following inferences from the plots.

6.1 Inference from the plots

In this section we shall list out some of the inferences from the plots

- Figures 5.8 and 5.9 show that GLRT based stopping rule is quickest in terms of detection delay (the difference between stopping time and actual change point) as compared to Confidence Set based stopping rule, for increasing change point, τ.
- Given GLRT based stopping rule, sensing rule 3 is quickest as compared to random sampling.
- Also, sensing rule (Algorithm PE) 3 is quickest as compared to sensing (Algorithm CS1 and Algorithm CS2) 1 and 2.
- Figure 5.7 shows that the sensing rule (Algorithm PE) 3 plays the "best" action , action 3 in the action set $\mathcal{A}$ and detects the change point quickly as compared to Random

Sampling. The theoretical lower bound (3.37) infers that the detection delay is minimum if we play action that has least KL divergence between post-change parameter $\theta^{(1)}$ and $\theta^{(0)}$. We insert Plug-in Estimate of $\theta^{(1)}$ each time and the estimate improves if we take more samples.

In future work, a more concrete threshold calculation needs to be done and we need to look at the upper bound on the detection delay with GLRT as the stopping rule.

Bibliography

[1] Yasin Abbasi-yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems 24*, pages 2312–2320, 2011. 21, 24, 25, 26, 44

[2] D.P. Bertsekas. *Convex Optimization Theory*. Athena Scientific optimization and computation series. Athena Scientific, 2009. ISBN 9781886529311. 8, 9

[3] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, USA, 2004. ISBN 0521833787. 8, 9, 11

[4] Sébastien Bubeck. Convex optimization: Algorithms and complexity, 2014. 8, 10

[5] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, New York, NY, USA, 1991. 7

[6] Aurelien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Conference On Learning Theory*, pages 998–1027, 2016. 36

[7] G.R. Grimmett and D.R. Stirzaker. *Probability and random processes*, volume 80. Oxford university press, 2001. 15

[8] P.G. Hoel, S.C. Port, and C.J. Stone. *Introduction to Probability Theory*. Houghton Mifflin series in statistics. Houghton Mifflin, 1971. 15

[9] T.L. Lai and Xing H. Sequential change-point detection when the pre- and post-change parameters are unknown. *Sequential Analysis*, 29, 2010. 5, 35

[10] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2018. 11, 20, 21, 44

[11] G. Lorden. Procedures for reacting to a change in distribution. *The Annals of Mathematical Statistics*, 42, 1971. 3, 4, 35

[12] Odalric-Ambrym Maillard. Sequential change-point detection: Laplace concentration of scan statistics and non-asymptotic delay bounds. *Proceedings of the 30th International Conference on Algorithmic Learning Theory*, 98, 2019. 4, 5

[13] Yurii Nesterov. *Introductory Lectures on Convex Optimization: A Basic Course*. Springer Publishing Company, Incorporated, 1 edition, 2014. 8, 10, 30

[14] Pearson Egon Sharpe Neyman Jerzy and Pearson Karl. On the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231:289–337, 1933. 20

[15] E.S. Page. A test for a change in a parameter occurring at an unknown point. *Biometrika*, 42:523–527, 1955. 3

[16] M. Pollak. Average run lengths of an optimal method of detecting a change in distribution. *The Annals of Statistics*, pages 749–779, 1987. 4

[17] H. Vincent Poor. *An Introduction to Signal Detection and Estimation (2nd Ed.)*. Springer-Verlag, Berlin, Heidelberg, 1994. 20

[18] Venugopal V Veeravalli. Decentralized quickest change detection. *IEEE Transactions on Information theory*, 47:1657–1665, 2001. 4