CS 109a - Milestone #2

Alex Lin Melissa Yu

October 22, 2016

1 Summary of Papers

The general goal of these papers is to determine whether or not a time series contains anomalies. The default assumption is that the data is exchangeable (i.e. the data points are independent and identically distributed). The three papers wish to measure the degree to which this notion of exchangeability is falsified in an online environment in which the data points arrive one by one. They are linked in their approach towards this goal; each of them uses an algorithm to (1) construct a sequence of p-values over time and (2) utilize the p-values to create Martingales that evaluate the deviation from the assumption of exchangeability. After a certain threshold, their algorithms can decide whether to reject the exchangeability notion.

The first paper was written by Vovk et. al (2003). Here, they introduce the general technique for constructing p-values that the two later papers also adopt. For the nth example z_n, they define the associated p-value p_n as

$$p_n = \frac{|i : \alpha_i > \alpha_n| + \theta_n \, |i : \alpha_i = \alpha_n|}{n}$$

where α_i is a noncomformity score for example z_i and θ_n is a random number from $[0, 1]$. Essentially, small p-values will be given to unusual examples that don't quite fit the observed data up until that point. For the second step, their system for designing Martingales involves exponentiation by some $\epsilon \in [0, 1]$. Thus, they refer to their constructions as *power Martingales*. The nth Martingale of the family indexed by ϵ is

$$M_n^{(\epsilon)} = \Pi_{i=1}^{n}(\epsilon p_i^{\epsilon-1})$$

They give some analysis for determining a good ϵ. They also introduce two other ideas - simple mixture and sleepy jumper - that are a bit more complex but follow the same general paradigm. Upon testing their algorithm on USPS data, they discover that their best result rejects exchangeability at a significance level of 10^{-18}.

Ho (2005) proposes two tests for exchangeability using randomized power martingales $M_n^{(\epsilon)}$ in the second phase of the form:

$$M_n^{(\epsilon)} = \Pi_{i=1}^{n}(\epsilon p_i^{\epsilon-1})$$

The first test rejects series where the martingale value $M_n^{(\epsilon)}$ exceeds a threshold. However, his studies showed that very small p-values tend to inflate these martingale values. Ho suggests to use the martingale difference $M_n^{(\epsilon)} - M_{n-1}^{(\epsilon)}$ as another test to address this problem. Both tests perform well on simulated data streams.

Fedorova et. al utilizes "plug-in martingales" S_n in the second phase of the form

$$S_n = \Pi_{i=1}^n (f_i(p_i))$$

where $f_i(p_i)$ is a betting function. In this paper, the authors estimate a probability density function for p from the accumulated p-values at each point and take it as their betting function. This approach differs from the other approaches presented earlier in that it does not assume a fixed betting function, but rather tunes the function to the data stream. The paper tests the algorithm on the USPS and Statlog Satellite datasets and shows that for stable sequences of p-values, their plug-in martingale provides asymptotically the best result.

2 Project Proposal

For our project, we propose creating a command-line program that will read in a .csv file of data in an on-line format and subsequently evaluate the three different algorithms of these papers. Our program will implement the algorithms and empirically compare their performances. The dataset that we propose to use for testing can be found here:

`https://yahooresearch.tumblr.com/post/114590420346`
`/a-benchmark-dataset-for-time-series-anomaly`

It is an open-sourced dataset provided by Yahoo of 367 different time series, each with around 1000 observations recorded over time. The time series encode information about Yahoo's traffic and have points in which unusual traffic occurs. Each series is accompanied by an indicator series that denotes whether or not each entry in an anomaly. (These binary values were determined by humans).

Thus, we will evaluate the speed at which the three algorithms determine anomalies in the Yahoo data, as well as the accuracy. For the first phase, we will feed the time series data in the order that it is provided to our program and see how quickly the three algorithms can identify the anomalies. For the second phase, we will randomize the data in an attempt to forgo the anomalies and see if the algorithms can correctly identify whether or not a stream of data satisfies exchangeability. In the end, we plan to provide graphs to visualize the performance of the algorithms and make final conclusions. We are excited about the project and think that it will be very interesting to work on.