Searching for Events in Time Series Using Scan Statistics

D. Preston, P. Protopapas
1Initiative in Innovative Computing, Harvard University, MA, USA, 2Harvard Smithsonian CfA, MA, USA
E-mail: dan_preston@harvard.edu

Introduction

The discovery of events in time series can have important implications, such as identifying stellar flares in astronomy or abnormal spikes in CPU usage. Research in scan statistics has been used in finding significant clusters of points. One such method uses several “sliding window” sizes to perform analysis, by which processes are performed on the data inside the window. Improving upon this, we developed a method to identify the duration of these events at any arbitrary window size. It begins by ranking individual points in a time series, performing a statistical analysis for each pair of starting points and window sizes, and using optimization methods to quickly identify low probability (and thus significant) events.

2. Algorithm

2.1. Data Mining Two-Dimensional Spatial Clusters

The problem of finding events in time series can be considered the same as detecting overdensities in a two-dimensional set of data. Our goal is to determine which subset of this data has the largest deviation from its baseline. Several methods can be used to determine this subset, such as average-multiresolution partitioning. [1] A more common approach in time series data is to use a sliding window (a subset of consecutive points) and analyze the values inside that window.

In addition to finding this subset sequence of points, one must determine whether or not the event is statistically significant. In order to do this, we must model the noise (i.e. determine the baseline). If the event deviates significantly from this, or within a given p-value, it can be considered significant. One way to model this is to recompute the events distances at the points. Unfortunately, this must be done for each time series. We provide a general statistical model for solving this problem.

2.2. Modeling Noise and Determining Statistical Significance

It would be useful for an analyst to consider a time series with a uniform distribution of 2.2. Modeling Noise and Determining Statistical Significance

It would be useful for an analyst to consider a time series with a uniform distribution of 2.2. Modeling Noise and Determining Statistical Significance

It would be useful for an analyst to consider a time series with a uniform distribution of 2.2. Modeling Noise and Determining Statistical Significance

It would be useful for an analyst to consider a time series with a uniform distribution of values (\(N\), dependent on the number of values being added together (\(w\)), and the bounds on the values (\(A\)).

By ranking the values in a time series and obtaining this uniform distribution of points, we can ultimately consider different probabilistic models with regards to sums inside of sliding windows. Consider a starting point \(x\) and a window size \(w\), and let \(Q(w)\) be the sum of the ranked values inside that window. We claim that those sums in the outer tails of the distribution (see Figure 2) of all possible sums are highly unlikely to have occurred by chance, as they deviate drastically from the null hypothesis.

2.3. Finding the Lowest Probability Window Sum

In order to find the sub sequence of \(T\) most likely to be an event, we must consider all possible pairs of \((x,w)\). A naive approach to solve this problem would be to perform a brute force search on all possible pairs, and identify the lowest p-value. This search would require \(O(n^2)\) to compute. For very large datasets, this is unacceptable, and an accurate approximation would be preferred.

A two-dimensional, \((x,w)\), a surface plot of the probabilities is very revealing. Figure 3 depicts an example of a significant event in a time series. After examination, one notices an obvious cluster of low probability points. By locating the minimum of this cluster of low-probability points, one can pinpoint the exact time and duration of the event in the time series, by finding the starting point \(x\) and window size \(w\). In order to find these clusters programatically and efficiently, the problem can be reduced to the well-explored problem of minimization (a special case of the class of optimization algorithms). One example of such an algorithm is Powell’s method [2].

In order to avoid local minima, several random restarts of the minimization algorithm are suggested. A clustering algorithm can be performed on the results to provide a concise list of possible events. (Note: this is also useful in identifying multiple events in a single time series)

3. Application to the MACHO data set

Over 85,000 light curves (time series measuring the magnitude of a star for each observation) were analyzed from the MACHO data set. Thirteen known Microlensing events [3], which occur when an object passes in front of a star, were included in this data set. All 13 events were identified by the algorithm and were ranked in the top 150 results considered most likely to be significant events. Figure 4 shows an example of one of the identified light curves.

Figure 1. Comparison of distribution models. The ranking of points is preferable to a curved probability distribution model, as it is uniform.

Figure 2. Probability distribution of all possible sums for \(w=2, f_w=50\). The simulated version is red, and the analytic version is green.

Figure 3. On the left is a time series with two likely events on the right is the surface plot (\(s\) by \(w\)) of the probability of the sum. Two events were identified by the algorithm at (58, 26) and (145, 12).

Figure 4. Light curve from MACHO data set with identified Microlensing event.

5. Summary

We developed a method for event discovery in time series data. By converting each time series to rank space, we removed the need to model the noise for each individual time series. Furthermore, by using a probabilistic model that considers all possible sums of these ranked points, we are able to use a single general model for all time series. With this information, we are able to identify the most likely candidate for an event (that with the lowest p-value), and compare against a threshold p-value to determine statistical significance. Lastly, we introduce the use of an optimization algorithm for quick computation.

6. Reference

