changepoint.online: An R Package for Online Changepoint Analysis

Andrew Connell, Rebecca Killick, David Matteson

2018-08-09

Abstract

One of new the key challenges in changepoint analysis is the need for an online approach. The **changepoint.online** package has been developed to provide users with a choice of changepoint search methods to use in conjunction with a given changepoint method and in particular provides an implementation of the recently proposed PELT algorithm much like the **changepoint** package does. The **changepoint.online** in fact has the same functionality and can produce the same results for offline data as the **changepoint** package however, it can also analysise online data. This article describes the methodology of the new online approachs and the key differences between the two methods with simulated and practical examples. Another key change is the *ECP* test statistic is now available, which stems from the **ecp** package.

Keywords: online, changepoint, energy time series, cluster analysis, R

1. Introduction

In recent years, there has been an increase in applications in which data is collected in an online manner, also known as streaming data. In many cases, it is not feasible or advisable to store this data prior to analysis. Thus, there is a growing need to develop analysis methods specifically for the online setting, see for example Gama and Rodrigues (2007). The ability to detect changes in the distributional properties of a time series accurately and efficiently in an online fashion has important implications in many industrial applications, for example in forecasting Koop and Potter (2007), consumer profiling (Whittaker et al., 2007) and fault detection in industrial process control (Lai, 1995); it is also a critical consideration in (online or offline) processing of time series segments and data storage applications Golab and Özsu (2003).

The problem of online changepoint analysis of time series has been considered extensively in the literature due to its importance in many scientific fields (see Page, 1954; Hawkins et. al, 2003). The changepoint problem for online data streams is typically formulated in a sequential hypothesis test framework in which, as new data arrives, the hypothesis of a change is tested, conditional on the data observed in the past. If a changepoint is found to be statistically significant, the changepoint detection method is "reset", i.e. the changepoint analysis restarts from the next observed data point. As far as we are aware, this methodology is only available in the **cpm** R package which provides a selection of distributional and nonparametric data assumptions.

A criticism of the sequential resetting or "conditional" formulation of the multiple changepoint problem above is that there is no accumulation of useful information as the data stream progresses: as soon as a change is found, the analysis is restarted as if the arriving data is an independent dataset, and thus previous information is not retained for assessing future changepoint locations. In particular, any errors made by the changepoint detection method early in the data stream propagate such that they affect any inference made about subsequent changes in the stream, adding uncertainty to those potential change locations. In addition, since any new data after a change are considered independent, detected changes in the past cannot be re-evaluated to improve accuracy in location estimation. This is not conducive to a streaming data environment where both speed and accuracy are valued.

Recently, Killick et al. (2012) proposed a changepoint search algorithm, named the Pruned Exact Linear Time (PELT) algorithm, which is shown to be exact and whose complexity is linear in the number of data points. Since the PELT method achieves the benefits of both exactness and computational efficiency, it has thus been shown to perform particularly well for applications in which the analysis of large datasets is required. A further development for nonparametric cost functions which provides an approximate pruning step, e-cp3o, is given in James and Matteson (2015). These algorithms are available in the popular changepoint and ecp packages on CRAN.

Our motivation for considering these algorithms is that, since they reassess the locations of potential changepoints each time new data arrives, the propagation of errors in detected changes is circumvented when performing inference about changes later in the stream. Furthermore, the exact (for PELT) and minimal computational complexity of the algorithms ensures these techniques find changes in data streams in an efficient and accurate fashion. Due to the dynamic programming in these algorithms they are inherently online in nature but were only available in R packages **changepoint** and **ecp** which require the full data for analysis. This package mirrors the functionality of the changepoint and ecp packages in terms of functionality but for provided initialisation functions and update functions required in an online setting. This paper describes the **changepoint.online** package, available for R from the Comprehensive R Archive Network (CRAN) at http://CRAN.R-project.org/package=changepoint.online.

2. Changepoint Detection

3. Introduction to the package and the 'ocpt' class

The **changepoint** package introduced a new object class called 'cpt' to store changepoint analysis objects. This section provides an introduction to the new 'ocpt' structure in **changepoint.online** which follows on from the 'cpt' class previously introduced. The new 'ocpt' class contains the functions within the 'cpt' class however, it also combines them with new functions that are approrpiate to the online method.

Each of the core functions outputs an object of the 'ocpt' S4 class. The class has been constructed such that the 'ocpt' object contains the main features required for the online changepoint analysis and future summaries. Several objects, although masked from the user and the summary, are filled within slots as they are required for the update functions. Each of the objects are stored within a slot entry in the 'ocpt' class. The slots within the class are:

- *data.set* – a time series ('ts') object containing the numeric values of the data;
- *cpttype* – characters describing the type of changepoint sought e.g., mean, variance;
- *method* – characters denoting the single or multiple changepoint search method applied.
- *test.stat* – characters denoting the test statistic, i.e., assumed distribution/distribution - free method;
- *pen.type* – characters denoting the penalty type;
- *pen.value* – the numeric value of the penalty used in the analysis;
- *cpts* – vector giving the estimated changepoint locations ending in the length of the time series in the data.set slot;
- *ncpts.max* – the numeric maximum number of changepoints searched for, e.g., 1, 5, Inf;
- *param.est* – a list of parameters where each element in the list is a vector of the estimated numeric parameter values for each segment;
- *date* – the system time / date when the analysis was performed.
- *version* - Version number of the package used when the analysis was run.
- *lastchangelike* - vector containing the likelihood of the optimal segmentation up to each timepoint.
- *lastchangecpts* - vector containing the last changepoint prior to each timepoint.

- *numchangecpts* - stores the current number of changepoints detected.
- *checklist* - vector of locations of the potential last changepoint for next iteration. Used only for update.
- *ndone* - length of the time series when analysis begins.
- *nupdate* - length of the time series to be analysed in this update.
- *cost_func* - the cost function used to decide possible changepoints.
- *shape* - only used when cost_func is the gamma likelihood. Otherwise 1.

Currently the method slot is masked from the user as PELT is the only method implemented. However, more methods are being added and at that point this option will be unmasked. Slots like version, although have nothing to do with the algorithm, are useful for continuity for the user who is running updates over a significant period of time.

The **changepoint.online** package is designed so that users who are already comfortable with the **changepoint** package will easily be able to use the online version. Therefore, much like with the **changepoint** package, we have created accessor and replacement functions to control the access and replacement of slots rather than using the @ accessor. Although the @ symbol is still available to use, from the feedback from the **changepoint** package users, it was shown that the accessor and replacement functions aided ease-of-use for those unfamiliar with S4 classes. The accessor functions are simply the slot names. For example data.set(x) displays the vector of data contained within the 'cpt' object x. The class slots are automatically populated with the correct information obtained from the completed analysis. Further demonstration of how the accessor and replacement functions work in practice are given in the examples within each section and can be found with the *man* section of the package itself.

All the functions described above are related to the 'ocpt' class within the **changepoint.online** package. However a further class 'ecp.ocpt' exists which contains all of the above as well as some additional information which is required in the case of the '*ECP*' test statistic. This information includes:

- *GofM* - goodness of fit model.
- *delta* - the window size used to calculate the calculate the complete portion of our approximate test statistic.
- *alpha* - the moment index used for determining the distance between and within segments.
- *verbose* - a flag indicating if status updates should be printed.

This class is created seperately rather than masking these options in cases when they are not needed to ensure that each process is at its optimal efficiency.

4. Usage and Examples

5. Conclusion

References

Gama, João, and Pedro Pereira Rodrigues. 2007. "Stream-Based Electricity Load Forecast." Edited by Joost N. Kok, Jacek Koronacki, Ramon Lopez de Mantaras, Stan Matwin, Dunja Mladenič, and Andrzej Skowron. Berlin, Heidelberg: Springer Berlin Heidelberg, 446–53.

Golab, Lukasz, and M. Tamer Özsu. 2003. "Issues in Data Stream Management." *SIGMOD Rec.* 32 (2). New York, NY, USA: ACM: 5–14. https://doi.org/10.1145/776985.776986.

Koop, Gary, and Simon M. Potter. 2007. “Estimation and Forecasting in Models with Multiple Breaks.” *The Review of Economic Studies* 74 (3). [Oxford University Press, Review of Economic Studies, Ltd.]: 763–89. http://www.jstor.org/stable/4626160.