UDMCA – A Shiny App for Single Species Univariate Changepoint Analysis

Udani A. Wijewardhana[1], Madawa Jayawardana[1, 2, 3], Denny Meyer[1]

[1] Department of Statistics, Data Science and Epidemiology, Swinburne University of Technology, Hawthorn, Victoria, Australia
[2] Peter MacCallum Cancer Centre, Melbourne 3000 Victoria, Australia
[3] Sir Peter MacCallum Department of Oncology, The University of Melbourne, Parkville 3010 Victoria, Australia

* uwijewardhana@swin.edu.au

Changepoint Analysis - Statistical packages used

User can use four Bayesian changepoint methods to identify the significant predictors and the significant changes of the abundance level using raw data (abundance/occurrence). Still these techniques allow to detect changepoints for a single location (univariate scenario). Therefore, the data entered by users are considered as a single location data when analyse changepoints.

changepoint package:

The changepoint package implements various mainstream and specialised changepoint methods for finding single and multiple changepoints within data which includes many popular non-parametric and frequentist methods (Killick, Haynes and Eckley, 2016).

breakpoint package:

The breakpoint package implements variants of the Cross-Entropy (CE) method to estimate both the number and the corresponding locations of break-points in biological sequences of continuous and discrete measurements. The proposed method primarily built to detect multiple break-points in genomic sequences. However, it can be easily extended and applied to other problems (Priyadarshana and Sofronov, 2016).

cumSeg package:

The cumSeg package (Muggeo, 2010) estimates the of number and location of change points in mean-shift (piecewise constant) models which is useful to model genomic sequences of continuous measurements. The algorithm first estimates the highest number of change points using the efficient 'segmented' algorithm of Muggeo (2003) and then select some of them using a generalized BIC criterion by applying the lar's algorithm of Efron et al. (2004) (Muggeo, 2010).

bcp package:

The bcp package provides an implementation of the Barry and Hartigan (1993) product partition model for the normal errors change point problem using Markov Chain Monte Carlo. It also extends the methodology to regression models on a connected graph (Wang and Emerson, 2015) and allows estimation of change point models with multivariate responses (Erdman and Emerson, 2007).

Shiny App

UDMCA is a Shiny web application that allows to visualize changepoints by using Bayesian changepoint techniques implemented in ‘changepoint’, ‘breakpoint’, ‘cumSeg’ and ‘bcp’. To carry out these analyses users simply need to click the buttons that create the input files required, execute the software and process the output to generate tables of values and plots with the results.

The application consists of 2 pages with main window:

1. Main window allows the user to upload the input files (data file) and also gives the option to normalize or standardize the counts or predictors data.
2. Changepoint analysis page that user can choose a Bayesian changepoint method to carry out the changepoint analysis.

This page gives the option to fit Bayesian changepoints for annual raw data. Since still developed univariate changepoint methods, app will consider the annual data CSV as a single located dataset. Four Bayesian changepoint packages (changepoint, breakpoint, cumSeg and bcp) could be used to find the significant changes. This page also visualizes the relevant changepoint profile plots. For bcp package user has the option to use predictor variables to analyse. As the algorithms stated in the page, user can change the relevant parameters for each method and find the best scenario.

Shiny

Shiny is a web application framework for R that enables to build interactive web applications which includes a combination of R scripts: a user-interface script called `ui.R` and a server script called `server.R`. The user-interface script controls the layout and appearance of the application and the server script contains the R objects and the instructions about how they are displayed.

The functionality called reactivity in shiny applications have three kinds of objects: reactive sources, reactive conductors, and reactive endpoints. The source typically is user input through a browser interface. A reactive endpoint is usually something that appears in the user’s browser window, such as a plot or a table of values. Reactive sources are accessible through the input object, and reactive endpoints are accessible through the output object. It’s also possible to put reactive components in between the sources and endpoints. These components are called reactive conductors. A conductor can both be a dependent and have dependents. In other words, it can be both a parent and child in a graph of the reactive structure. Reactive conductors can be useful for encapsulating slow or computationally expensive operations. For example, imagine that you have this application that takes a value `input$n` and prints the nth value in the Fibonacci sequence, as well as the inverse of n^{th} value in the sequence plus one (Shiny - Reactivity - An overview, 2020).

To create Shiny applications there is no web development experience required, although it is possible to use HTML, CSS, or JavaScript to achieve greater flexibility and customization. Shiny applications can be run locally by users that have the application files and R installed in their computer. Applications can also be hosted as a web page at its own URL and can be navigated through the internet with a web browser. This facilitates its use to people without R knowledge (Moraga, 2015).

This app includes several control widgets, where users can interact with to send messages to the application. In the main window, we can upload the data files by using a file upload control which is created by the `fileInput` function. We also can specify the variables normalization options by selecting the appropriate names from boxes containing the possible choices. These boxes are created with the `selectInput` function. This function also available in species distribution model page and changepoint analysis page as well. Changepoint analysis page includes `numericInput` function which allows to enter a single number or a range to fit the model. DataTable is used to display interactive tables containing the information of the variables of interest (Xie, 2016). HTML widgets are created with JavaScript libraries and embedded in Shiny by using the HTML widgets package (Vaidyanathan et al., 2016). These two HTML widgets are included into the application by calling an output for the widget in the user- interface and assigning a render call to the output on the server side in the same way as `renderPlot` functions work. DT to create data tables. Information about all the dependencies is shown in Table 1.

Table 1: Softwares and R packages used for developing UDMCA

	Name	**Description**
Software		
R	R Core Team, 2017	Language and environment for statistical computing and graphics.
R Packages		
dplyr	Wickham and Francois, 2016	A fast, consistent tool for working with data frame like objects, both in memory and out of memory.
ggplot2	Wickham, 2009	Creates elegant data visualisations using the grammar of graphics.
htmlwidgets	Vaidyanathan *et al.,* 2016	Provides a framework for easily creating R bindings to JavaScript libraries.
		Web Application Framework for R.
shiny	Chang *et al.,* 2016	
changepoint **breakpoint** **bcp** **cumSeg**	Killick, Haynes and Eckley, 2016 Priyadarshana and Sofronov, 2016 Erdman and Emerson, 2007 Muggeo, 2010	Bayesian changepoint analysis techniques.
DT	Xie, 2016	Create data tables.
Matrix	Bates et al., 2019	A rich hierarchy of matrix classes, including triangular, symmetric, and diagonal matrices, both dense and sparse and with pattern, logical and numeric entries.
lyr	Wickham, 2020	A set of tools that solves a common set of problems: you need to break a big problem down into manageable pieces, operate on each piece and then put all the pieces back together. For example, you might want to fit a model to each spatial location or time point in your study, summarise data by panels or collapse high-dimensional arrays to simpler summary statistics.

Installation:

Users can launch the application by https://udani-wijewardhana.shinyapps.io/UDMCA/.
Users can access the GitHub repository by https://github.com/uwijewardhana/UDMCA.

References

Aminikhanghahi S, Cook DJ. (2017) A Survey of Methods for Time Series Change Point Detection. Knowl Inf Syst. 2017;51(2):339–367. doi:10.1007/s10115-016-0987-z

Shiny.rstudio.com. 2020. Shiny - Reactivity - An Overview. [online] Available at: https://shiny.rstudio.com/articles/reactivity-overview.html [Accessed 29 April 2020].

Killick R, Haynes K and Eckley IA (2016). changepoint: An R package for changepoint analysis. R package version 2.2.2, https://CRAN.R-project.org/package=changepoint.

Priyadarshana, W. J. R. M., and G. Sofronov. (2016). breakpoint: An R Package for Multiple Break-Point Detection via the Cross-Entropy Method. CRAN Repository (2016): 1-25.

Muggeo, V. M. R., and G. Adelfio. (2011). Efficient change pointdetection for genomic sequences of continuous measurements.Bioinformatics 27:161-166. http://dx.doi.org/10.1093/bioinformatics/btq647

Muggeo, V.M.R. (2003), Estimating regression models with unknown break-points. Statist. Med., 22: 3055-3071. doi:10.1002/sim.1545

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Leastangle regression.Annals of Statistics,32, 407– 489.

Barry,D. and Hartigan,J.A. (1993) A Bayesian analysis for change point problems.J. Am.Stat. Assoc.,88, 309– 319.

Erdman C., Emerson J.W. (2007), bcp: An R Package for Performing a Bayesian Analysis of Change Point Problems, Journal of Statistical Software, 23(3), 1-13, URL http://www.jstatsoft.org/v23/i03/.

Wang X. and Emerson J.W. (2015), Bayesian Change Point Analysis of Linear Models on General Graphs, Working Paper.

Xie Y (2016). DT: A Wrapper of the JavaScript Library 'DataTables'. R package version 0.2, URL https://CRAN.R-project.org/package=DT.

Vaidyanathan R, Xie Y, Allaire J, Cheng J, Russell K (2016). htmlwidgets: HTML Widgets for R. R package version 0.7, URL https://CRAN.R-project.org/package=htmlwidgets.

R Core Team (2017). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org.

Wickham H, Francois R (2016). dplyr: A Grammar of Data Manipulation. R package version 0.5.0, URL https://CRAN.R-project.org/package=dplyr.

ChangW, Cheng J, Allaire J, Xie Y,McPherson J (2016). shiny: Web Application Framework for R. R package version 0.14.1, URL https://CRAN.R-project.org/package=shiny.

Zeileis A, Grothendieck G (2005). "zoo: S3 Infrastructure for Regular and Irregular Time Series." Journal of Statistical Software, **14**(6), 1–27. doi: 10.18637/jss.v014.i06.