Check for updates

SOFTWARE TOOL ARTICLE

A new method for detecting abrupt shifts in time series [version 1; peer review: 2 approved with reservations]

Chris A. Boulton 🔬, Timothy M. Lenton

Global Systems Institute, University of Exeter, Exeter, Devon, EX4 4QE, UK

Abstract

Abrupt shifts in time series are a topic of growing interest in a number of research areas. They can be caused by a range of different underlying dynamics, for example, via a mathematical bifurcation, or potentially as the result of an auto-correlated stochastic process (i.e. 'red' noise). Here we present a method that detects abrupt shifts by searching for gradient changes that occur over a short space of time. It can be automated, allowing many time series to be analysed by the user at once, such as from high spatial resolution data. Our method detects abrupt shifts regardless of their origin (which it cannot deduce). We present a comparison with the method of abrupt shift detection from the changepoint R package, which is based on changes in mean over the time series. Our method performs better on data with an underlying trend where comparisons of means may fail.

Keywords

Abrupt shifts, Time series analysis, Shift detection, R package, Methods

Open Peer Review

Reviewer Status ❓ ❓

	Invited Reviewers	
	1	**2**
version 1 published 28 May 2019	❓ report	❓ report

1   **Valerie N. Livina** 🔬 , National Physical Laboratory, Teddington, UK

2   **Sonia Kéfi** 🔬 , University of Montpellier, Montpellier, France

Any reports and responses or comments on the article can be found at the end of the article.

This article is included in the RPackage gateway.
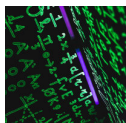
This article is included in the Mathematical, Physical, and Computational Sciences collection.

Corresponding author: Chris A. Boulton (c.a.boulton@exeter.ac.uk)

Author roles: Boulton CA: Conceptualization, Data Curation, Formal Analysis, Investigation, Methodology, Software, Visualization, Writing – Original Draft Preparation; **Lenton TM**: Conceptualization, Funding Acquisition, Supervision, Writing – Review & Editing

Competing interests: No competing interests were disclosed.

How to cite this article: Boulton CA and Lenton TM. **A new method for detecting abrupt shifts in time series [version 1; peer review: 2 approved with reservations]** F1000Research 2019, **8**:746 (https://doi.org/10.12688/f1000research.19310.1)

First published: 28 May 2019, **8**:746 (https://doi.org/10.12688/f1000research.19310.1)

Introduction

Abrupt shifts are found in time series in many fields, including but not limited to ecology[1,2], climate[3–6], medicine[7] and engineering[8]. Abrupt shifts in time series can be caused by a range of different underlying dynamics. They are sometimes described as 'tipping points'[3,6] and are caused by an underlying mathematical bifurcation[9], implying some change in control parameters governing the system in question. Alternatively, noise-induced transitions can occur between different attractors, triggered purely by stochastic variability. Sufficiently rapid forcing could also trigger an abrupt response from a system without passing a bifurcation[10]. In other situations, a strongly auto-correlated stochastic process ('red' noise) can produce what appear as abrupt shifts in time series, although it is debated whether they should be described as such[11].

Our method is based on the presumption that when people search for abrupt shifts in time series by eye, they are looking for significant changes in the gradient of the time series over a short space of time. Hence we have designed a method based on looking for anomalous rates of change in a time series and released it as an R package called 'asdetect'[12] There exist several other methods of detecting anomalous changes in time series (change-point analysis[13,14], STARS[15], etc) which are based on looking for significant shifts in the distribution of the data. These attempt to fit generalised models to the data, or use t-tests (or other variants) to detect a significant shift in the distribution of segments of the time series. In contrast, our algorithm focuses more on the transition itself rather than the resulting behaviour of the time series (i.e. differences between the situation before and after). Thus, the approaches can be viewed as complementary. Our method should be intuitive to users, particularly those with little statistics experience. It is also useful for analysing a number of time series at once when it would be otherwise impossible to view them all individually.

Figure 1 provides a graphical representation of how our method differs from the widely-used 'changepoint' analysis package from R[16]. This method is used in the function as_detect() in our R package. For a time series that exhibits a typical abrupt shift that users might be searching for (Figure 1a), both methods detect a shift. For a drifting time series, however, a change in mean is detected from the change-point algorithm but an abrupt shift is not detected by our method (Figure 1b). These example time series, particularly the time series that exhibits the abrupt shift are used throughout the article.

Our algorithm searches for anomalous changes in the gradient of a time series. We begin with a time series of length n (an example is shown in Figure 2), which we separate into l-sized segments. Across each segment, we fit a linear regression and record the gradient/coefficient of each of the n/l segments. For ls that are not factors of n and thus would give a remainder, we spread these remaining points equally at the start and end, such that our segments would focus on the centre of the time series rather than at the beginning or end.
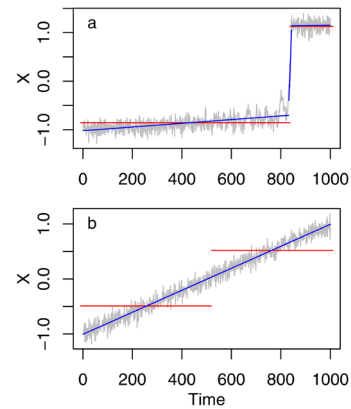

Figure 1. Cartoon example of how the method works. The tool looks for changes in the gradient of the time series (blue) compared to the changepoint analysis R package that looks for changes in mean value over time (red). (**a**) An example time series that undergoes a bifurcation and hence there is an abrupt shift. (**b**) A drifting time series where a change in the mean is detected by the changepoint algorithm but this does not relate to an abrupt shift. These example time series are discussed in more detail later.

We look for which of these gradients are significantly different by seeing which values lie outside ± 3 median absolute deviations (MADs) of the distribution of gradients. For those points in time that contribute to a significant gradient, we add or (subtract for significantly lower/negative gradients) 1 to the relevant l-long segment(s) of a 'detection time series' of length n which contains only 0s otherwise.

This process is repeated over a range of ls. The minimum and maximum l is chosen by the user (although both have a default; see below), and the process is repeated for all integers between these values. The resulting detection time series is divided by the number of ls tested, which gives a time series showing the (sign-included) proportion of times that a point contributed to a significant gradient. From this, a user-defined threshold (magnitude) can be used to determine if an abrupt shift is detected in the time series.

Our second function, where_as(), takes the detection time series as well as this threshold magnitude (between 0 and 1) to locate where the abrupt shifts are detected. The function records the indices where the detection time series is beyond the threshold magnitude, using the sign of the detection times series to distinguish increasing or decreasing abrupt shifts. The R function rle(), which computes the lengths of runs of equal values in a vector, is used on the differences of the list of indices that are above the threshold. This allows us to determine the sections of the time series that contain the detected abrupt shift(s). Within these subsets, we then find the maximum or minimum (depending on whether the detection is positive or negative) value to report as the position of an abrupt shift.
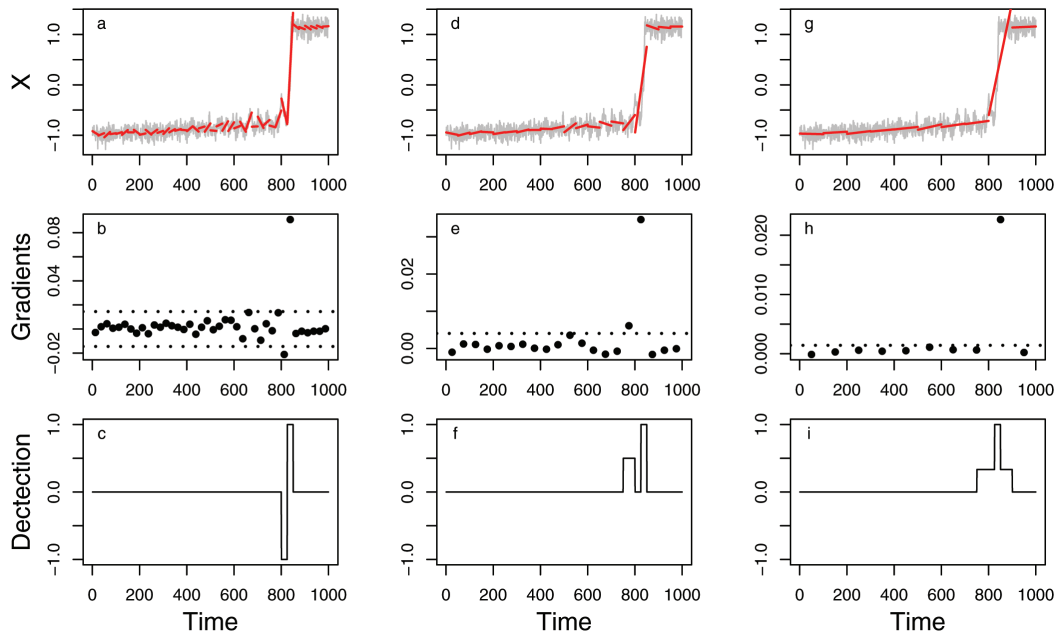
Figure 2. An example of our detection algorithm being used on an example time series. Three choices of *l*, 25 (**a–c**), 50 (**d–f**), and 100 (**g–i**) are shown. Full explanation of the plots is given in the main text. In (**a**), (**d**) and (**g**), the example time series is shown in grey with red lines showing the linear regressions fitted through each section. In (**b**), (**e**) and (**h**), the gradients of each linear regression are shown, along with dotted lines representing the 3 median absolute deviations (MADs). Panels (**c**), (**f**) and (**i**) show the evolving detection time series, which is described in detail in the main text.

While it is easy to find a single maximum/minimum in the detection time series, our second function allows any number of abrupt shifts to be found, depending on how strict the user wants to be. If no values in the detection time series are found to be above the threshold level, our function will issue a warning and instead return the maximum detected value and position.

Methods
Implementation
Our R package asdetect[12] contains a number of functions which allows the user to detect abrupt shifts in either their own data, or in example time series created by functions within the package.

Our algorithm in as_detect() begins by separating a time series of length *n* into *l*-sized segments. Within each segment, we fit a linear regression of the form $Y = a + bX$ where Y is the values in the time series, X is the vector, $\begin{bmatrix} 1 \\ : \\ l \end{bmatrix}$, *b* is the gradient of the regression model (which is recorded) and *a* is the intercept of the model (estimate of *Y* when X=0). For the set of gradients, we calculate how many are more than 3 median absolute deviations (MADs) away from the median of the set of gradients, where 1 MAD = median($|x$-median$(x)|$). MADs are a substitute for standard deviations, which could be heavily influenced by large outliers. Points in the time series that contribute to significant gradients have a value of 1 added or subtracted to a detection time series, which is then divided by the number of *l* used.

The model for bifurcating example time series (shown in Figure 1–Figure 4) is:

$$\dot{x} = -x^3 + x + \mu + \sigma\eta \qquad (1)$$

Where *x* is the time series, μ is a parameter that causes the bifurcation. It runs linearly from 0 to $\frac{2\sqrt{5}}{9}$ at *t*=900, up to *t*=1000. Stochastic white noise is applied to the system ($\sigma\eta$), sampled from a normal distribution with mean 0 and standard deviation σ which we set to 0.1. We start the time series at *x*=-1, the stable state when μ=0. This time series can be recreated with the function tip_eg() in our R package.

Our example time series of white noise and drifting white noise has the same level of stochastic forcing as the bifurcation time series above, centred on -1:

$$\dot{x} = -1 + D + \sigma\eta \qquad (2)$$

With the same parameter choices as in Equation (1). The value of D is set to 0 for our white noise time series, and ranges from 0 to 2 across the 1000 points of the drifting time series seen later. A similar time series, centred on 0 rather than -1 can be created from our function white_noise_eg() in our R package.

The flat start and end time series are created from combinations of the white noise and drifting white noise time series, using the D parameter. For example, for the flat start time series, D is

set to 0 for the first 200 points and then increases at the same rate as previously, such that it runs linearly from 0 to 1.6 over the over 800 time points. The time series for the flat end, D runs from 0 to 1.6 over the first 800 time points, and then remains at 1.6 for the final 200 time points.

Our Pacific Decadal Oscillation (PDO) Index example is taken from the Joint Institute for the Study of the Atmosphere and Ocean, and is the first principal component of sea surface temperatures in the North Pacific once an annual cycle and general warming trend have been removed. This time series is not included within our package.

We use the cpt.mean() function of the R package 'change-point', with its default settings as described in the help file of the function.

Operation

Our package is designed to be used in standard R and has no dependencies on other packages apart from installation. As asdetect[12] is hosted on GitHub, users will first need to download the 'devtools' package, using the command install. packages('devtools') in the R console. Then our package can be installed using the command devtools::install_github ('caboulton/asdetect') and then loaded with the command library('asdetect').

Results
Example case: underlying tipping point

Figure 2 shows an example of how the detection time series is built up, over three different l choices, for an example time series of data. A similar time series can be created with our tip_eg(), as described in the Methods. We note, as mentioned above, the full algorithm uses all integer values between the range prescribed, but only show three distinct values as a demonstration of how our method works. The model underlying the example time series exhibits a bifurcation at t=900; however, the system is forced with stochastic noise which causes the system to tip somewhat earlier. We stress that our algorithm is ignorant of how the shift in the time series occurs, but such a canonical example makes it clearer to illustrate its use.

We begin by using a value of l of 25 (Figure 2a) and fit the linear regressions through each 25 length segments (red lines; Figure 2a). There are two significant gradients found with l=25 (Figure 2b), a significantly negative gradient for the points 801–825 and a significantly positive gradient for points 826–850. This results in -1 and 1 values stored in the detection time series in Figure 2c. We then imagine we are next running the analysis using a value of 50 for l (Figure 2d), although in reality the value of l increases by 1 each time. This time there are significant positive gradients found for points 751–850 (Figure 2e), which counteracts the negatives found with l=25. This causes the points 801–825 to revert back to 0. We also divide the detection time series through by 2 as we have tried for both l=25 and 50 (Figure 2f). This results in a peak of 1 (gradients significantly positive for both values of l) for points 826–850, 0.5 for points 751–800 (those were only significantly

positive for l=50) and 0 for points 801–825 (those which were significantly negative for l=25, and positive for l=50), as well as the 0s elsewhere. Finally, we again run the analysis using l=100 (Figure 2g). We find a significantly positive gradient for 801–900 (Figure 2h), leading to a detection algorithm that contains 1s at points 826–850, with 0.33 assigned to the other points between 751–900 (Figure 2i).

Running the full algorithm, for all values of l between 5 and 300 (rather than just 3 specific choices of l as shown in Figure 2), for the example time series (Figure 3a), we use the resulting detection time series in our second function (Figure 3b). We use a threshold of 0.7 and find that points 778–872 are above this threshold. Within this, we find the highest detection (0.993) at point 838 (Figure 3b). Point 838 in the example time series is the first point the time series is positive and as such is a good indicator that for this particular underlying system the time series has switched from one (disappearing) basin of attraction to the other.

Alternative examples

Figure 4 shows that in contrast to the time series with an underlying bifurcation (Figure 4a, b), a time series of constant white noise (Figure 4c), has a maximum absolute detection value of (-0.115) and point 794 (Figure 4d). We also examine a time series of drifting white noise (Figure 4e), which shows a maximum detection value of 0.12 at point 518 (Figure 4f). These detection values from the other time series are much lower than those found in the time series with underlying bifurcation and would not be detected by our second algorithm (with only the
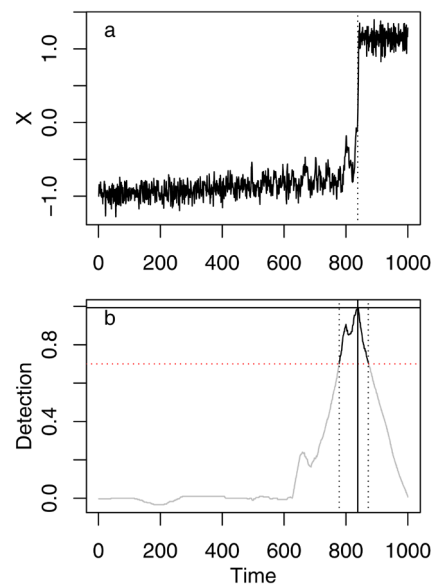
Figure 3. Determining where the abrupt shift is detected in our example time series. (**a**) The example time series. (**b**) Detection time series is shown in black where it is above the threshold value (0.7, vertical dotted line; as described in the main text) and grey otherwise. Vertical dotted lines denote the range where the detection time series is above the threshold (778-872) whereas the solid lines denote the maximum detection value (0.99) and position (838).
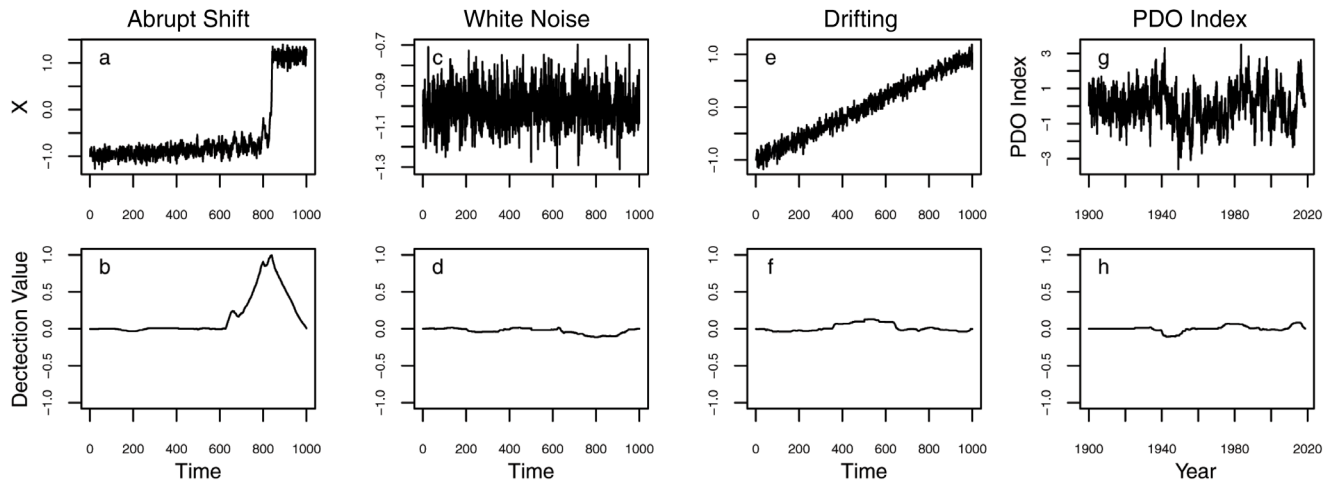
Figure 4. Examples of the detection algorithm run on different time series. (**a**, **b**) Abrupt shift (as in Figure 1 and Figure 2), (**c**,**d**) white noise, (**e**,**f**) a drifting time series, and (**g**,**h**) the PDO Index. Details of the detections are described in the main text.

maximum and a warning returned). We use the PDO index as a 4th example (Figure 4g). This is the major mode of climate variability in the North Pacific once the annual cycle and overall warning trend have been removed[17]. There is debate in the literature about whether or not there are abrupt shifts in the index, linked to changes in climate regime[18,19], or if it is actually a red noise process that drifts above and below a single mean state[11]. This real-world time series acts as an interesting case study for our algorithm. Our method detects no significant abrupt shift (Figure 4h) with a maximum absolute detection value of -0.108 in May 1942. This suggests that, at least according to our method, that the PDO is a red noise process, as no shifts are detected.

Anomalous 'flat' sections

A caveat of our algorithm is that it detects all anomalous gradients some of which would not be described as abrupt changes. For example, where a time series is relatively constant for a short period at the start before a longer period of increasing, then the short period at the start will be detected as anomalous (an unusually low gradient relative to the rest of the time series). This is also true if a short flat section is detected at the end of an increasing or decreasing time series. Examples of these cases are shown in Figure 5. Figure 5 also shows that these types of behaviour are not necessarily picked up strongly, with minimum detection values of -0.69 and -0.39 for increasing and decreasing time series, respectively. Similar behaviour is also detected for flat sections in otherwise decreasing time series. These are detected as increasing abrupt shifts (not shown).

We have created a third function, shift_type(), that checks if the abrupt shifts that are detected within time series are 'flat' or not. True abrupt shifts that most users will be searching for will have stronger gradients closer to the detected position than the gradient of the time series overall. Our third function compares the absolute value of the gradient of a linear regression model fitted locally around the detected position compared
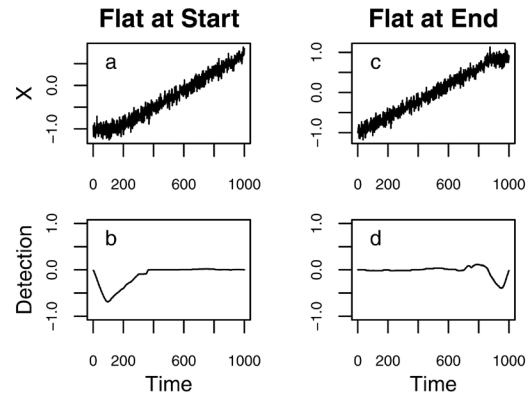
Figure 5. Different shift behaviour detected by our algorithm, namely flat sections. Flat sections occur at the (**a**, **b**) start or (**c**, **d**) end of increasing time series are detected as decreasing abrupt shifts as their gradients are significantly lower than the rest of the time series.

to the absolute gradient of a linear regression fitted across the full time series. We treat those shifts with smaller gradients compared to the overall gradient as 'flat'. The subset of the time series the local linear regression model is fitted to is a user defined choice, but we recommend (as is the default) it to be 10% of the total length of the time series, 5% either side. If a shift is detected within the first or last 5% (or within the length determined by the user), then the function will use data up to the start or end of the time series, even if this results in a smaller window that the model is fitted to.

Comparison to changepoint

To determine how our algorithm compares to the changepoint package, we compare the detections in six example time series, the tipping point time series, white noise, drifting white noise, and we also include two time series with flat sections at the beginning and end of otherwise increasing time series, which provides an example of what is described above. Analyses of the PDO index are also compared.

Table 1 shows the positions of abrupt shifts detected by both methods, including the detection value from our algorithm and the confidence value from changepoint. We also include the actual position that the abrupt shift is in the time series. In changepoint we use the cpt.mean() function. cpt.mean() searches for changes in the mean across the time series, and we use the 'AMOC' method such that a maximum of one will be detected. The position returned is the beginning of the second section that has a significantly different mean to the first.

We find that both methods agree exactly with where the abrupt shift occurs in the tipping point time series. There are no abrupt shifts detected in the white noise series, with a detection value of -0.115 only being picked up if the user used such a low threshold for detection. The same is true of our algorithm for the drifting time series. However, the changepoint algorithm detects a change at point 520 (with confidence 0.917). For the flat time series, the detections from our algorithm are much closer to the actual change in the time series, although the detection values are relatively low. This is useful as it is not the type of abrupt shift that would usually want to be detected but is still highlighted. These types of anomalies are also picked up by our third function. Changepoint detects changes about half way through the time series, with much higher confidence values. This is a simple result of the changepoint method looking for changes in the mean.

While changepoint manages to detect the tipping point time series as well as our own method does, it struggles with time series where there is no abrupt shift but an overall trend. It is also noticeable that the confidence values are a lot higher than our detection values in the 'flat' time series, despite being further away from the actual transition. However, we iterate that the changepoint function is looking for changes in the mean over the time series and so although is used for detecting abrupt shifts, there is no differentiating between fast and slow shifts. It detects a shift in the drifting time series as that is what it is designed to do. Our method, however, which focuses on gradients, finds no abrupt shift.

Discussion
Practicality and choice of variables
The main inputs of our algorithm are the choice of minimum and maximum l. This depends on the length of the time series

such that significant gradients can still be found. A significant gradient can only be found if there are three or more segments and thus the maximum value of l should be less than 1/3 of n, the length of the time series. Any choice of l above this will cause the maximum possible detection value to decrease as well as all the detection values to be lower than they otherwise would be. Having a choice of maximum l too low, however, may bias detection towards short term fluctuations in the time series rather than the type of abrupt shift-type behaviour that was intended. As a default of our function, a maximum value of l equal to 1/3 of n is used.

A choice for the minimum value is more difficult to determine. It is required to be big enough such that if a time series is 'noisy', then it will not detect significant gradient changes due to deviations caused by low-level noise. Less noisy (smoother) systems can have a smaller minimum value, and as such will pick up smaller abrupt shifts which would be difficult in a time series with more noise. We set a default of 5 points as a 'rule of thumb' for being able to find a significant coefficient in a linear regression model.

Another choice of variable is the threshold in our second function to determine where abrupt shifts are detected. Within a consecutive set of points, the function will search for the maximum value within this range. However, it does not take into consideration local maxima (or minima) that are still above the threshold. This is a design choice of our algorithm, as we assume that users will want to only find the largest abrupt shift locally, and that the smaller nearby shifts (local maxima/minima) have had their detection values inflated due to being near the large abrupt shift.

An example is found in our original case (Figure 2). We showed using a threshold of 0.7 that an abrupt shift was detected (Figure 3). However, using a threshold value of 0.88, we find another abrupt shift at point 800 (detection is 0.91), as well as the original at point 838 (detection: 0.99). This can be seen as a large excursion that (just) failed to leave the disappearing basin of attraction, in this case which is approaching an underlying bifurcation. Figure 6 shows that the local minimum in the detection time series is below the threshold and thus a new segment is detected. A threshold value of 0.95, again only detects the original shift.

Table 1. The six example time series used to compare our algorithm against the changepoint analysis package. Results in brackets are not detected as shifts by our algorithm, or deemed significant by the changepoint package.

Time series	Actual position of abrupt shift	Our algorithm detection position	Our algorithm detection value	Changepoint position	Changepoint confidence value
Tipping point	900*	838	0.993	838	0.998
White Noise	N/A	(794)	(-0.115)	N/A	N/A
Drifting	N/A	(518)	(0.128)	520	0.917
Flat start	150	(96)	(-0.689)	537	0.881
Flat end	850	(948)	(-0.392)	520	0.917
PDO Index	NA	(May 1942)	(-0.108)	(Feb 2014)	(3.6*10^-4)

*Although the bifurcation in the tipping point time series happens at 900, the noise pushes it to an abrupt shifts before this (as described in the main text).
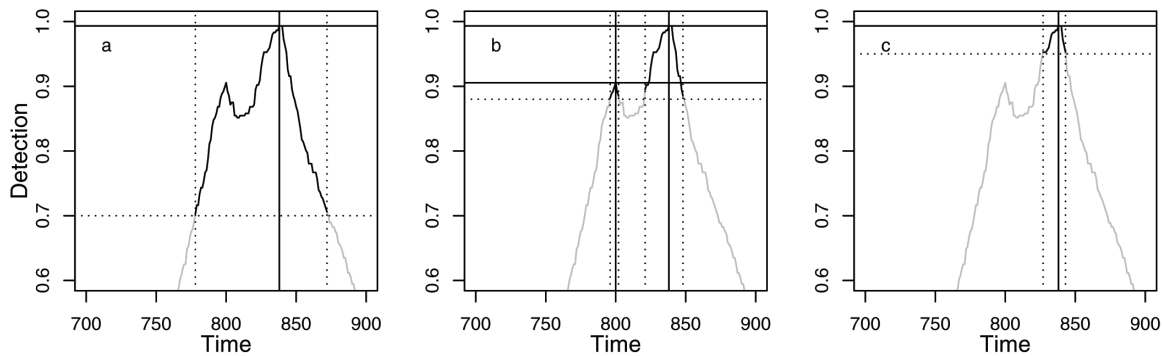
Figure 6. Comparison of detected abrupt shifts in example time series when using different thresholds. The example time series (Figure 1 and Figure 2) is tested when using a threshold of (**a**) 0.7 (as in Figure 2), (**b**) 0.88, and (**c**) 0.95. A second smaller shift is detected with the middle threshold (refer to Figure 1 and Figure 2 to link this back to the original time series).

We have outlined our method for detecting abrupt shifts in time series, which is based on looking for significant changes in gradient of the series. Our algorithms differ from those such as the changepoint R package, which looks for significant changes in mean values. We have shown that this method detects shifts in time series that drift when they would not ordinary be deemed abrupt shifts by users. Our alternative method does not suffer from this set-back. Furthermore, it is intuitive in its design, whilst allowing greater control for users to discover abrupt shifts of different magnitudes.

Data availability
Underlying data
The PDO Index time series can be found at the Joint Institute for the Study of the Atmosphere and Ocean website.

Other, stochastic, time series were created using our package. To create a time series that contains an abrupt shift, the function tip_eg() is used. An example of white noise with or without drift can be created with the white_noise_eg(). Time series that

began or ended with flat sections used a combination of two white_noise_eg() outputs that were stitched together to give the desired outcome.

Software availability
Source code available from: https://github.com/caboulton/asdetect/.

Archived source code at time of publication: https://doi.org/10.5281/zenodo.2634044[12].

License: MIT License.

References

1. Scheffer M, Bascompte J, Brock WA, *et al.*: **Early-warning signals for critical transitions.** *Nature.* 2009; **461**(7260): 53–9.
   PubMed Abstract | **Publisher Full Text**

2. Litzow MA, Urban JD, Laurel BJ: **Increased spatial variance accompanies reorganization of two continental shelf ecosystems.** *Ecol Appl.* 2008; **18**(6): 1331–1337.
   PubMed Abstract | **Publisher Full Text**

3. Lenton TM, Held H, Kriegler E, *et al.*: **Tipping elements in the Earth's climate system.** *Proc Natl Acad Sci U S A.* 2008; **105**(6): 1786–1793.
   PubMed Abstract | **Publisher Full Text** | **Free Full Text**

4. Held H, Kleinen T: **Detection of climate system bifurcations by degenerate fingerprinting.** *Geophys Res Lett.* 2004; **31**(23).
   Publisher Full Text

5. Dakos V, Scheffer M, van Nes EH, *et al.*: **Slowing down as an early warning signal for abrupt climate change.** *Proc Natl Acad Sci U S A.* 2008; **105**(38): 14308–14312.
   PubMed Abstract | **Publisher Full Text** | **Free Full Text**

6. Lenton TM: **Early warning of climate tipping points.** *Nat Clim Chang.* 2011; **1**: 201–209.
   Publisher Full Text

7. Scheffer M, Bolhuis JE, Borsboom D, *et al.*: **Quantifying resilience of humans and other animals.** *Proc Natl Acad Sci U S A.* 2018; **115**(47): 11883–11890.
   PubMed Abstract | **Publisher Full Text** | **Free Full Text**

8. Livina V, Barton E, Forbes A: **Tipping point analysis of the NPL footbridge.** *Journal of Civil Structural Health Monitoring.* 2014; **4**(2): 91–98.
   Publisher Full Text

9. Thompson JMT, Sieber J: **Predicting climate tipping as a noisy bifurcation: a review.** *Int J Bifurcat Chaos.* 2011; **21**(02): 399–423.
   Publisher Full Text

10. Ashwin P, Wieczorek S, Vitolo, R, *et al.*: **Tipping points in open systems: bifurcation, noise-induced and rate-dependent examples in the climate system.** *Philos Trans A Math Phys Eng Sci.* 2012; **370**(1962): 1166–1184.
    PubMed Abstract | **Publisher Full Text**

11. Rudnick DL, Davis RE: **Red noise and regime shifts.** *Deep Sea Research Part I:*

Oceanographic Research Papers. 2003; **50**(6): 691–699.
Publisher Full Text

12. Boulton CA: **asdetect: An R package for detecting abrupt shifts in time series**. 2019.
http://www.doi.org/10.5281/zenodo.2634045

13. Beaulieu C, Sarmiento JL, Mikaloff Fletcher SE, *et al.*: **Identification and characterization of abrupt changes in the land uptake of carbon.** *Global Biogeochem Cycles.* 2012; **26**.
Publisher Full Text

14. Beaulieu C, Chen J, Sarmiento JL: **Change-point analysis as a tool to detect abrupt climate variations.** *Philos Trans A Math Phys Eng Sci.* 2012; **370**(1962): 1228–1249.
PubMed Abstract | **Publisher Full Text**

15. Rodionov SN: **A sequential algorithm for testing climate regime shifts.** *Geophys Res Lett.* 2004; **31**.
Publisher Full Text

16. Killick R, Eckley IA: **changepoint: An R Package for Changepoint Analysis.** *J Stat Softw.* 2014; **58**(3): 19.
Publisher Full Text

17. Mantua NJ, Hare SR, Zhang Y, *et al.*: **A Pacific Interdecadal Climate Oscillation with Impacts on Salmon Production.** *B Am Meteorol Soc.* 1997; **78**: 1069–1080.
Publisher Full Text

18. Hare SR, Mantua NJ: **Empirical evidence for North Pacific regime shifts in 1977 and 1989.** *Prog Oceanogr.* 2000; **47**(2–4): 103–145.
Publisher Full Text

19. Scheffer M, Carpenter S, Foley JA, *et al.*: **Catastrophic shifts in ecosystems.** *Nature.* 2001; **413**(6856): 591–596.
PubMed Abstract | **Publisher Full Text**

Open Peer Review

Current Peer Review Status: ❓ ❓

Version 1

Reviewer Report 03 July 2019

❓ **Sonia Kéfi** iD

Institute of Evolution Sciences, BioDICée team, CNRS, IRD, EPHE, University of Montpellier, Montpellier, France

In this paper, the authors present a method and the R-package 'asdetect' to detect abrupt shifts in time series data. The approach relies on identifying anomalous changes in the gradient of a time series.

The tools provided by the R-package 'asdetect' are important and interesting. They will be a nice complement to the set of tools already available in the literature. For example, previous approaches such as one of the functions implemented in the R package 'changepoint' searches for changes in the mean, rather than the gradient.

The paper is well-written. The methods presentation could nonetheless be improved. I hereafter suggest some changes that may help improve the clarity and make the paper more accessible.

1. I would slightly reorganize the structure of the paper to gather all the methods in a single place in the paper. As it is now, part of the methods is in the introduction and the rest in the methods. Part of the information needed to understand the methods is also online, in the help of the R-package and in the Readme file of the github repository. It would greatly improve the clarity of the methods presentation if all the information needed was gathered in a single location.
Therefore, I would move the last 5 paragraphs of the introduction to the methods part (all the text after 'We begin with a time series of length n', at the bottom of the left column of p. 2). In the third paragraph of the left column of p. 2, I would also move the sentence mentioning the function 'as_detect()' to the methods. This will make the introduction less technical and more focused on giving the intuition of the methods. This will also decrease the amount of repetitions between the introduction and the methods.

2. In the introduction, after the third paragraph of the left column of p. 2, I would spend a few more words explaining the intuition of the approach and the difference with other methods, such as the ones implemented in the 'changepoint' package. Figure 1b is a convincing illustration of the differences in outputs given by the two approaches for drifting time series. However, there is also not quite enough information about 'changepoint' for the reader to truly understands what it does (and why, in the example of Figure 1b, it detects a shift in the middle of the time series). I am not

very familiar with 'changepoint', but it seems that the package provides several methods to detect changes in time series and that cpt.mean(), which is the function used by the authors for the comparison here, is only one of them. As mentioned by the authors, there are also other methods and packages available and it's unclear why so much emphasis is put on 'changepoint' in this paper. I would clarify all of this in the introduction, and in particular state that Figure 1b is one illustrative example of a striking difference between this approach and previous ones, obtained when using one of the possible functions of 'changepoint'.

3. The presentation of the methods could be improved by more explicitly mentioning the different steps of the analysis.
- Merging the end of the introduction with the start of the 'Implementation' part of the methods will help.
- Also, a graphical presentation of the different steps performed with the names of the corresponding functions of the package as well as what they take as input and what they give as output would be very useful.
- When the function as_detect() is mentioned in the text, it would help to say that is takes a time series as input and gives back a 'detection time series' as output.
- The 'detection time series' should be clearly defined the first time it is mentioned (i.e. the fact that it is a vector of the same length of the time series which will contains information about the proportion of times a significant gradient was detected at that time step). Right now, there is a vague explanation in the first paragraph of the right column of p. 2 ('For those points in time...'), but the actual explanation is in the results part of the paper. It could stay that way, but then the first time it is mentioned, instead of a general reference to Figure 2, a more specific reference to the left column of Figure 2 would be less confusing, because the reader only gets the information needed to understand the two other columns of Figure 2 (and especially how panels f and i are obtained) after reading the elements currently given in the methods. An alternative may be to move the explanation of Figure 2 to the methods.
- Also, it might be helpful to mention in the legend of Figure 2 that panel f contains the information from b and e, and that panel i is derived from information from b, e and h.
- In the second paragraph of the right column of p. 2, "although both have a default; see below". I would specify that this is in the discussion of the paper (e.g. replace 'see below' by 'see discussion').
- At the beginning of the last paragraph of p. 2, when the function where_as() is introduced, it would help to mention the two outputs of that function.
- It was not really clear to me what the function rle() gives as an output (last paragraph of p. 2). It would also help if such information was added to the readme file of the Github repository (which is really useful to get practically started with the package).
- In the methods, I would divide the 'implementation' part in two: one part explaining the approach (and including the last paragraphs of the current introduction), and another part for the presentation of the example time series used (i.e. in the right column of p. 3 create a new subparts called 'Data'?).

4. In the results, in the second paragraph of p. 6, I did not find the explanations of what happens for the flat time series very clear. Why is this 'useful as it is not the type of abrupt shift that would usually want to be detected'? I am not sure what the authors mean. How is this also 'picked up by the third function'? (Here, I would also mention the name of the function explicitly.)
- Caveats of the approach are presented in the results of the flat sections of time series. Is that the

only situation where this approach is expected to give a false positive? Are there other situations where this method would fail to detect a shift or wrongly detect one?

5. The authors briefly mention in the introduction of the paper that this approach can be viewed as complementary to others. I would mention that again in the discussion by saying something about the difficulty in detecting shifts in noisy data and the importance of using a combination of tools to improve detectability.

Minor points:
- In the abstract, it is mentioned that 'we present a comparison with the method...': this comparison is however not really systematic and complete, in particular since (as far as I understood) it uses only one of the functions of the package. I would reformulate this sentence e.g. as 'we illustrate differences with one of the commonly used function of the changepoint R package'.

- In the readme file of the Github repository, it would help to add that after installation, the package has to be loaded with library('asdetect'), as it is done in the text in the 'Operation' part.

- In the help of the as_detect() function, the default value for lowwl is not mentioned(?).

- In the results, first paragraph, the sentence 'a similar time series...' can be deleted. There is no need to repeat this.

- In the results, first paragraph, '..., but we only show three distinct values' ('we' lacking?).

- In the results, second paragraph, 'This results in -1 and 1': I would remind at the end of that sentence that the detection time series has the same length as the original time series.

- In the results, second paragraph of p. 4: 'in our second function': I would say the name of the function for clarity.

- First paragraph of p. 5: 'according to our method, the PDO is a red noise process' (remove 'that'?).

- Second paragraph of the left column of p. 5: I was a bit confused by the sentence: '...these types of behaviour are not necessarily picked up strongly...' but the value of -0.69 is pretty high, isn't?

- Second paragraph of the left column of p. 5, last sentence: maybe show this?

- Last paragraph of the left column of p. 5: 'We have created a third function...', but other functions have been mentioned in between. To avoid confusion, I would say 'we have created another function...'. Also, it would be helpful to add this function (with an example of use) in the readme file of the Github repository.

- Last paragraph of p. 5: maybe specify again that you use only one of the functions of change point (maybe if this is more clearly stated at the beginning of the paper, this won't be needed here anymore).

- Last paragraph of the results: '...and so although it is used for detecting...' ('it' missing?)

- First paragraph of the right column of p. 6: the maximum value of l is mentioned two times in that paragraph. Delete one of them.

- Third paragraph of the right column of p. 6: 'our second function': for clarity, mention the name of the function.

Is the rationale for developing the new software tool clearly explained?
Yes

Is the description of the software tool technically sound?
Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?
Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?
Partly

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Yes

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Ecology, theoretical ecology, alternative stable states, tipping points, facilitation, ecological networks.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

Reviewer Report 20 June 2019

https://doi.org/10.5256/f1000research.21169.r49169

? **Valerie N. Livina** (iD)
Data Science Group, National Physical Laboratory, Teddington, UK

The paper by Boulton and Lenton "A new method for detecting abrupt shifts in time series" proposes an alternative to the previously published change-point techniques (methodology and R package software).

I think the paper is interesting and can be indexed after revision. In particular, the authors may be interested to look at the recent paper: Beaulieu C. and R. Killick (2018)[1] and the corresponding software package 'EnvCpt' that performs automatic selection between a variety of trends, changepoints and autocorrelation models, found here.

On page 2, the authors mention that the red noise "can produce what appears as abrupt shift in time series, although it is debated whether they should be described as such". A time series with a dynamically changing scaling exponent of red noise actually provides the underlying mechanism of a gradual tipping, as confirmed by several indicators for early warning signals (see Prettyman *et al.*, EPL 2018[2]). This should be distinguished from the red noise with a single fixed scaling exponent that does not change with time (and that may be the case of absence of a tipping point). This needs to be discussed more clearly, otherwise an unprepared reader could be misled regarding the role of memory in tipping dynamics. In particular, this should be better described in the Abstract (or removed from there for simplicity).

Since the object of analysis is a single-variable time series, the authors actually calculate not gradient (which is by definition multi-variate) but a discrete derivative in each subset of data.

It would be good to see a systematic ensemble experiment estimating the success rate of the technique performance for a range of: jumps heights at change points (with respect to the series noise level), lengths of time series, and window sizes (similar to Fig.2 but in a broader range of parameters and with summary over multiple samples rather than a few examples).

It is not clear to me why the detection series is always within the interval [-1,1] – do the authors normalise this series? On page 2, the authors say that they "add or subtract 1 to the relevant l-long segments of a detection time series which contains only 0s otherwise"- but in (Fig.2f,i) there are also non-zero and non-one values, why?

The discussion of the threshold choice could be improved for better clarity, in my opinion. Maybe some diagram could help visually here.

References

1. Beaulieu C, Killick R: Distinguishing Trends and Shifts from Memory in Climate Data. *Journal of Climate* . 2018; **31** (23): 9519-9543 Publisher Full Text
2. Prettyman J, Kuna T, Livina V: A novel scaling indicator of early warning signals helps anticipate tropical cyclones. *EPL (Europhysics Letters)*. 2018; **121** (1). Publisher Full Text

Is the rationale for developing the new software tool clearly explained?

Yes

Is the description of the software tool technically sound?

Yes

Are sufficient details of the code, methods and analysis (if applicable) provided to allow replication of the software development and its use by others?

Yes

Is sufficient information provided to allow interpretation of the expected output datasets and any results generated using the tool?

Yes

Are the conclusions about the tool and its performance adequately supported by the findings presented in the article?
Partly

Competing Interests: No competing interests were disclosed.

Reviewer Expertise: Time series analysis.

I confirm that I have read this submission and believe that I have an appropriate level of expertise to confirm that it is of an acceptable scientific standard, however I have significant reservations, as outlined above.

The benefits of publishing with F1000Research:

- Your article is published within days, with no editorial bias

- You can publish traditional articles, null/negative results, case reports, data notes and more

- The peer review process is transparent and collaborative

- Your article is indexed in PubMed after passing peer review

- Dedicated customer support at every stage

For pre-submission enquiries, contact research@f1000.com

F1000**Research**