25th International Congress on Modelling and Simulation, Darwin, NT, Australia, 9 to 14 July 2023
mssanz.org.au/modsim2023

A Bayesian change-point approach to nanopore basecalling

S. Shen [a], **G. Sofronov** [a]

[a] *School of Mathematical and Physical Sciences, Macquarie University, Sydney, Australia*
Email: georgy.sofronov@mq.edu.au

Abstract: Understanding the genetic makeup of organisms is a very important goal in bioinformatics. DNA sequencing, the process of determining the order of the nucleotide bases in DNA, can now be performed quickly and cheaply with commercially available devices no bigger than a USB stick. The latest DNA sequencers use nanopore technologies to capture long, repetitive DNA structures with great success, however, the reported reading accuracy needs improving. One main source of error occurs during the basecalling process when raw nanopore signals outputted by the sequencers are being translated into genetic codes. The distinctive feature of basecalling lies in that not only do the nanopore signals need to be segmented, but they also need be grouped into four types, each representing a genetic code. In this paper, we propose a novel basecalling algorithm using change-point detection methods and Markov chain Monte Carlo (MCMC) sampling techniques. We use real and simulated data to demonstrate the effectiveness of the proposed algorithm.

Keywords: *Nanopore data, DNA sequencing, segmentation algorithm, Markov chain Monte Carlo, change-point detection*

1 INTRODUCTION

The basic idea of nanopore sequencing technologies involves a protein nanopore set in an electrically resistant polymer membrane (see Figure 1). By setting a voltage across this membrane, an ionic current is created and is flowing through the pore. When a DNA molecule passes through this hole, disruption to the current occurs. It is possible to identify which molecule passes through the pore by measuring changes in the electrical current. Further details about nanopore technologies can be found in Deamer et al. (2016). DNA sequencing is the process of determining the order of the nucleotide bases in DNA. Over the last few years, DNA sequencing using the enzyme-based nanopore technologies has become increasingly popular. One source of error occurs during which the raw nanopore signals are being translated into genetic alphabet (A, C, G and T). This process is called basecalling. One of the challenges of basecalling is that not only do the raw nanopore signals need to be segmented, but they also need be classified into four types, each representing a genetic code.

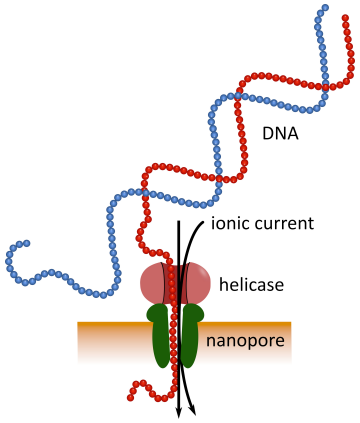

Figure 1. Nanopore sequencing diagram. Image from Andreas et al. (2019).

In this paper, we propose a novel basecalling algorithm using change-point detection methods and Markov chain Monte Carlo (MCMC) sampling techniques. Change-point detection methods have been widely used in bioinformatics applications (see, for example, Evans et al. (2011); Sofronov et al. (2009); Sofronov (2011); Polushina and Sofronov (2011, 2013, 2014, 2016); Priyadarshana and Sofronov (2012)). Most change-point methods can identify the number of change-points and their locations but there are very few discussions around a change-point algorithm that can also group the segments simultaneously.

2 STATISTICAL MODEL

We consider the nanopore data X as a time series with time points $t = 1, \ldots, T$, where T represents the total length of the data. Let x_t be a real data value at a particular time point t, $X = (x_1, \ldots, x_T)$. We assume X consists of K homogeneous segments. Let the start of the k-th segment be s_k, $k = 1, 2, \ldots, K$. For ease of notation, we assume the first change-point is always at the start of a segment where $t = 1$, so $s_1 = 1$. The change-points, therefore, are $(s_1, s_2, \ldots, s_K)$, where the last change-point s_K is at the start of the K-th segment. We assume the data within each segment is normally distributed with a unique mean and a common variance but the number of segments or change-points and their locations are unknown, $x_t = c_k + \epsilon_t$, where c_k is the mean level of the k-th segment and ϵ_t is the error term, $\epsilon_t \sim \mathsf{N}(0, \sigma^2)$, where σ^2 is the common variance.

Assume there are 4 distinct levels (or groups) that each c_k could belong to. If the levels represent a particular DNA alphabet, there are 4 groups (A, C, G, and T). Assume μ_g and τ_g^2 are the mean and the variance of the distribution for group g, $g = 1, 2, 3, 4$. We can thus write c_k as $c_k \sim \mathsf{N}(\mu_g, \tau_g^2)$. Let π be the probabilities that each segment can be assigned to one of the 4 groups, $\pi = (\pi_1, \pi_2, \pi_3, \pi_4)$, $\pi_1 + \pi_2 + \pi_3 + \pi_4 = 1$.

We follow a methodology similar to one used in Sadia et al. (2019) to build the posterior likelihood functions based on the statistical model introduced above. The posterior likelihood function is the probability of

generating the observed sequence of segments for any given parameter values.

Firstly, if ϕ is the probability that a data point is a change point, for our model with K segments and start positions $s = (s_1, s_2, \ldots, s_k)$ with $s_1 = 1, s_K \leq T$, the following probability statement is true:

$$p(K, s|\phi) = \phi^{K-1}(1-\phi)^{T-K}. \tag{1}$$

We assume each segment can be assigned to one of 4 groups with probabilities $\pi = (\pi_1, \ldots, \pi_4)$. Let g_k be the group of the k-th segment. Let b_g be the number of segments that belong to group g, i.e. b_1 is the number of segments belonging to the first group and b_2 is the number of segments belong to the second group, etc. The probability of a specific assignment of K segments into g groups is therefore:

$$p(g|K, \pi) = \prod_{g=1}^{4} \pi_g^{b_g}. \tag{2}$$

If c_k is the mean of the k-th segment and the value of c_k depends on the mean and the variance of the group c_k belongs to, then we can write the probability of the mean for all segments as:

$$p(c|g, \mu, \tau^2) = \prod_{k=1}^{K} \mathsf{N}(c_k|\mu_g, \tau_g^2). \tag{3}$$

The probability density function of the data is the product of the probability density functions over all the segments. We can express this as:

$$p(X|K, s, c, \sigma^2) = \prod_{t=1}^{T} \mathsf{N}(x_t|\sigma^2). \tag{4}$$

Using Bayes' theorem, the posterior distribution of the parameters is:

$$p(K, S, g, c, \phi, \pi, \sigma^2, \mu, \tau^2 \mid X) \propto p(X, K, S, g, c \mid \phi, \pi, \sigma^2, \mu, \tau^2)\, p(\phi)\, p(\pi)\, p(\sigma^2)\, p(\mu)\, p(\tau^2). \tag{5}$$

The first term on the right hand side of (5) is the joint distribution of X, K, s, g and c given the prior parameters, and it can be expressed as:

$$p(X, K, S, g, c \mid \phi, \pi, \sigma^2, \mu, \tau^2) = p(X \mid K, S, c, \sigma^2)\, p(K, S \mid \phi)\, p(c \mid g, \mu, \tau^2)\, p(g \mid K, \pi).$$

Using (1), (2), (3), and (4), we can see that the posterior distribution (5) is proportional to:

$$\begin{aligned} &p(X, K, S, g, c \mid \phi, \pi, \sigma^2, \mu, \tau^2)\; p(\phi)\, p(\pi)\, p(\sigma^2)\, p(\mu)\, p(\tau^2) \\ &\quad = \phi^{K-1}(1-\phi)^{T-K} \prod_{k=1}^{K} \mathsf{N}(c_k \mid \mu_{g_k}, \tau_{g_k}^2)\pi_{g_k} \prod_{t=1}^{T} \mathsf{N}(x_t \mid c_k, \sigma^2)\, p(\phi)\, p(\pi)\, p(\sigma^2)\, p(\mu)\, p(\tau^2). \end{aligned}$$

3 ALGORITHM

The proposed change-point basecalling algorithm was developed within the framework of the Generalized Gibbs Sampler; see Keith et al. (2004, 2008). The goal of the algorithm is to achieve the following two objectives simultaneously: 1) identification of change points, and 2) classification of each segment divided by the change points into four groups. There are three main parts to the algorithm: insertion, deletion, and update model parameters.

3.1 Insertion

A new change-point z is randomly proposed in segment k between s_k and s_{k+1} breaking the segment into two and new values of g_k and c_k are drawn for the proposed left and right segments. Let c'_k and g'_k be the new values for the proposed left segment and c''_{k+1} and g''_{k+1} be the new values for the proposed right segment. The g'_k and g''_k are generated using the group probability occurrence $\pi_1, \ldots, \pi_g$, and c'_k and c''_k are generated using a normal distribution with parameters $(\mu_{g'_k}, \tau^2_{g'_k})$ and $(\mu_{g''_{k+1}}, \tau^2_{g''_{k+1}})$. Following Sadia et al. (2019), it can be shown that the new change-point is rejected with a probability proportional to P_1 given by

$$P_1(\text{insert}) = (1-\phi)\prod_{t=s_k}^{s_{k+1}} p(\epsilon_t|0,\sigma^2)\frac{1}{(d_k - s_k)}\frac{1}{M(K)},$$
$$p(\epsilon_t|0,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\epsilon_t^2}{2\sigma^2}\right), \qquad \epsilon_t = x_t - c_k,$$

$M(K)$ is the total number of moves for a sequence with K segments. The new point is accepted with a probability proportional to P_0

$$P_0(\text{insert}) = \phi \prod_{t=s_k}^{z} p(\epsilon'_t|0,\sigma^2) \times \prod_{t=z+1}^{s_{k+1}-1} p(\epsilon''_t|0,\sigma^2)\frac{1}{M(K+1)},$$
$$p(\epsilon'_t|0,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\epsilon_t'^2}{2\sigma^2}\right), \qquad \epsilon'_t = x_t - c_k,$$
$$p(\epsilon''_t|0,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}}\exp\left(-\frac{\epsilon_t''^2}{2\sigma^2}\right), \qquad \epsilon''_t = x_t - c_k.$$

The probability of accepting the new change-point z is $P_0(\text{insert})/(P_0(\text{insert}) + P_1(\text{insert}))$.

3.2 Deletion

The deletion step merges the current segment k with the previous segment $k-1$. It is performed from segment $k = 2, \ldots, K$ (we cannot delete when we are at the first segment as there is no previous segment). New values of g_k and c_k are drawn for the new proposed merged segment. Similar to the Insertion step, g_k is generated using the group probability occurrence $\pi_1, \ldots, \pi_g$ and c_k is generated from a normal distribution with parameters $(\mu_{g_k}, \tau^2_{g_k})$. The probability of rejecting the deletion is proportional to P_0 given by

$$P_0(\text{delete}) = \phi \prod_{t=s_{k-1}}^{z} p(\epsilon'_t|0,\sigma^2) \times \prod_{t=z+1}^{s_{K+1}-1} p(\epsilon''_t|0,\sigma^2)\frac{1}{M(K)}.$$

The probability of accepting the deletion is proportional to P_1

$$P_1(\text{delete}) = (1-\phi)\prod_{t=s_{k-1}}^{s_{k+1}-1} p(\epsilon_t|0,\sigma^2)\frac{1}{(s_k - 1 - s_{k-1})}\frac{1}{M(K-1)},$$

The probability of accepting the deletion of the change-point is $P_1(\text{delete})/(P_0(\text{delete}) + P_1(\text{delete}))$.

3.3 Update model parameters

At the end of an iteration (after insertion and deletion are performed for each segment), we update model parameters, including group assignments g_k, mean levels c_k and parameters σ^2, ϕ, π, μ, τ^2. The updated parameter values are used for the next iteration. The algorithm uses conventional Gibbs updates. The Slice sampler is used to draw from non-standard distributions.

4 NUMERICAL RESULTS

In this section, we provide the results of a numerical study, in which we apply the proposed algorithm to both simulated and real data sets.

4.1 Artificial nanopore data

We generate normally-distributed data of length 1000 with 8 change-points representing the sequence AGTCTTGATA. The mean and the variance for each of the four type of nucleotide generated are chosen based on the information given in Derrington et al. (2010). We use different initial values of group mean/variance and vary other model parameters to observe their effects. In this example, we use the following prior distributions: $\phi \sim \mathsf{Beta}(1,1)$, $\pi \sim \mathsf{Dirichlet}(1,1,1,1)$, $\tau^2 \sim \mathsf{Inv\text{-}Gamma}(3,3)$, $\sigma^2 \sim \mathsf{Inv\text{-}Gamma}(3,3)$.

Figure 2 shows the generated data (black dots) with the corresponding nucleotide type. The average profile obtained by the proposed algorithm is plotted in red. The average profile were calculated using the last 100 iterations. We also looked at the percentage of correct group assignments versus the number of iteration. We have noticed that the number of correct group assignments stabilises after 50 or so iterations and it is around 95%. Various initial mean values for the algorithm were tried. Figure 3 illustrates that the group mean levels start oscillating near the true mean levels after around the first 50 iterations regardless of the initial values.

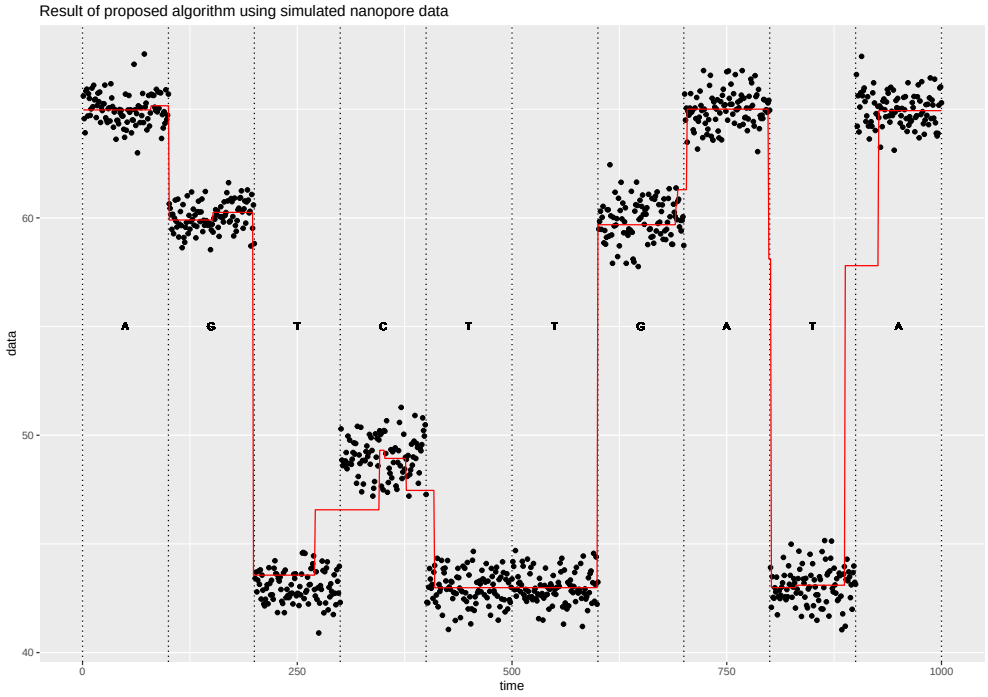

Figure 2. An average profile plot of the proposed algorithm for the artificial data.

4.2 Real data analysis

Here we present the analysis of a small segment of the real nanopore data outputted by the MinION sequencer (see Jain et al. (2016)). Raw nanopore data can also be referred to as squiggles because of the shape of the signals when you plot it. Figure 4 shows a small section of a read of the raw signal with the length of 400 (black dots) and the average profile (red line), which follows the data very well detecting most of the peaks and troughs. Since this is a real nanopore data set, we do not know the true profile and, therefore, we look for agreement between the different change-point methods. The issue is that existing change-point packages use the change-point methodology in a more general sense and do not take into account the nature of the DNA sequencing data nor do they consider any grouping of similar segments. We compare our algorithm with the `bcp` package, which is a Bayesian R package based on the algorithm created by Barry and Hartigan (1993). The algorithm was designed to detect changes in the mean of independent Gaussian observations. The package returns the posterior probability of a change-point occurring at each time index in the series. The resulting posterior means are shown in blue in Figure 4. Both algorithms show very good agreement in the identification of the change-point locations.

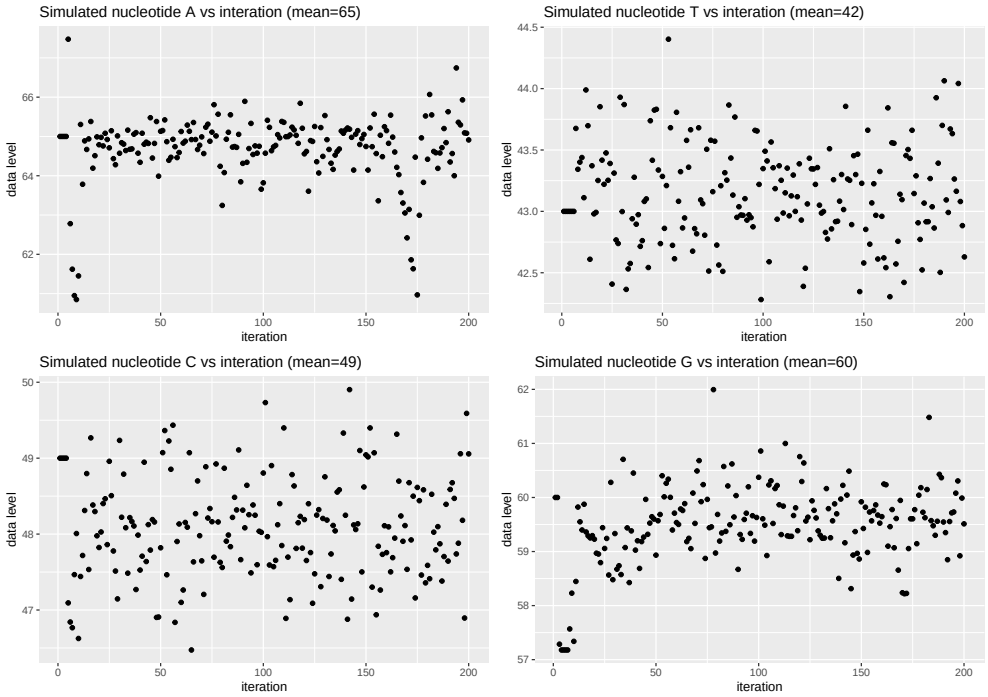

Figure 3. Group mean levels versus the iteration number.

5 CONCLUSIONS

Change-point detection methods have been widely used in bioinformatics, however, there have been no change-point methods developed for nanopore basecalling. The proposed algorithm focuses on identifying the locations of change-points and groups the segments into specific types in the context of basecalling. A thorough comparison with other basecalling algorithms and implementation of the method in R package `breakpoint` (see Priyadarshana and Sofronov (2015, 2016)) as well as the development of a model that takes into consideration dependency of data (for example, see Ma et al. (2020); Ma and Sofronov (2020)) is a matter for the future research. In addition, since basecalling can be performed in real time during sequencing, it is also possible to use sequential change-point methods (for example, see Sofronov et al. (2012)).

REFERENCES

Andreas, M., Kerstin, F., Roland, S., Thomas, H., 2019. Sequencing of mRNA from whole blood using nanopore sequencing. J. Vis. Exp. 148, e59377.

Barry, D., Hartigan, J.A., 1993. A Bayesian analysis for change point problems. Journal of the American Statistical Association 88, 309–319.

Deamer, D., Akeson, M., Branton, D., 2016. Three decades of nanopore sequencing. Nature Biotechnology 34, 518–524.

Derrington, I.M., Butler, T.Z., Collins, M.D., Manrao, E., Pavlenok, M., Niederweis, M., Gundlach, J.H., 2010. Nanopore DNA sequencing with MspA. Proceedings of the National Academy of Sciences 107, 16060–16065.

Evans, G., Sofronov, G., Keith, J., Kroese, D., 2011. Estimating change-points in biological sequences via the Cross-Entropy method. Annals of Operations Research 189, 155–165.

Jain, M., Olsen, H.E., Paten, B., Akeson, M., 2016. The Oxford Nanopore MinION: delivery of nanopore sequencing to the genomics community. Genome Biology 17, 1–11.

Keith, J., Sofronov, G., Kroese, D., 2008. The generalized Gibbs sampler and the neighborhood sampler, in: Monte Carlo and Quasi-Monte Carlo Methods 2006. Springer, pp. 537–547.

Keith, J.M., Kroese, D.P., Bryant, D., 2004. A generalized Markov sampler. Methodology and Computing in Applied Probability 6, 29–53.

Ma, L., Grant, A., Sofronov, G., 2020. Multiple change point detection and validation in autoregressive time

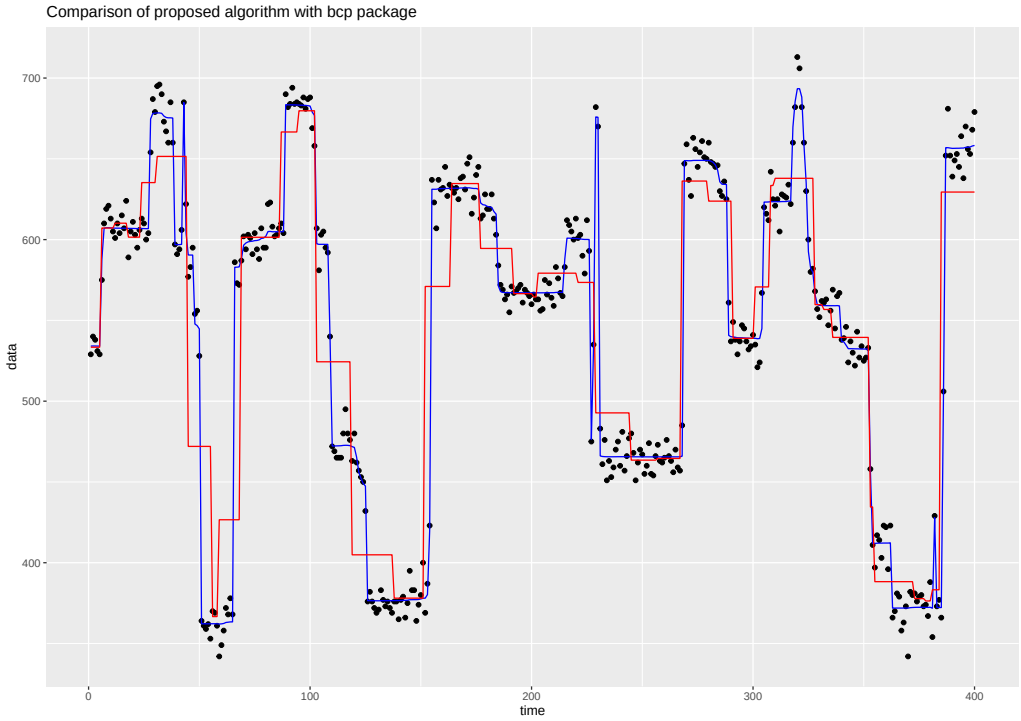

Figure 4. Real nanopore data (black dots) and estimated profiles obtained by the proposed algorithm (red line) and `bcp` package (blue line).

series data. Statistical Papers 61, 1507–1528.

Ma, L., Sofronov, G., 2020. Change-point detection in autoregressive processes via the Cross-Entropy method. Algorithms 13, 1–16.

Polushina, T., Sofronov, G., 2011. Change-point detection in biological sequences via genetic algorithm, in: 2011 IEEE Congress of Evolutionary Computation, CEC 2011, pp. 1966–1971.

Polushina, T., Sofronov, G., 2013. A hybrid genetic algorithm for change-point detection in binary biomolecular sequences, in: 12th IASTED International Conference on Artificial Intelligence and Applications (AIA 2013), pp. 1–8.

Polushina, T., Sofronov, G., 2014. Change-point detection in binary Markov DNA sequences by the Cross-Entropy method, in: 2014 Federated Conference on Computer Science and Information Systems, pp. 471–478.

Polushina, T., Sofronov, G., 2016. A Cross-Entropy method for change-point detection in four-letter DNA sequences, in: 2016 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology, pp. 1–6.

Priyadarshana, M., Sofronov, G., 2012. A modified Cross Entropy method for detecting multiple change points in DNA count data, in: 2012 IEEE Congress on Evolutionary Computation (CEC 2012), pp. 1–8.

Priyadarshana, M., Sofronov, G., 2015. Multiple break-points detection in array CGH data via the Cross-Entropy method. IEEE/ACM Transactions on Computational Biology and Bioinformatics 12, 487–498.

Priyadarshana, M., Sofronov, G., 2016. breakpoint: an R package for multiple break-point detection via the Cross-Entropy method.

Sadia, F., Boyd, S., Keith, J.M., 2019. Bayesian change-point modeling with segmented arma model. PLOS ONE 13, 1–23.

Sofronov, G., 2011. Change-point modelling in biological sequences via the Bayesian adaptive independent sampler, in: International Conference on Telecommunication Technology and Applications, pp. 122–126.

Sofronov, G., Evans, G., Keith, J., Kroese, D., 2009. Identifying change-points in biological sequences via sequential importance sampling. Environmental Modeling and Assessment 14, 577–584.

Sofronov, G., Polushina, T., Priyadarshana, M., 2012. Sequential change-point detection via the Cross-Entropy method, in: 11th Symposium on Neural Network Applications in Electrical Engineering, pp. 185–188.