Detection of replication origins in individual cells using nanopore sequencing

Ela Fallik

October 28, 2019

Abstract

The problem of accurate and fast identification of replication origins and replication fork movements is a complex one. The use of nanopore sequencing can help significantly in finding a fast solution. The D-NAscent software measure replication fork movement on single molecules by detecting nucleotide analog (BrdU) signal currents on extremely long nanopore traces[1]. On yeast (sacCer3) DNA, this software does not meet with some basic prediction. We use part the D-NAscent software in combination of Hidden Markov Model to detect segments of BrdU - incorporated 6-mers. Then, we could detect replication origins and fork directions by adding BrdU to cells in their S phase.

Introduction

5-bromo-2'-deoxyuridine (BrdU) is an analog of thymidine, commonly used in the detection of proliferating cells in living tissues[2,3]. It can be incorporated into DNA during replication, in the S phase of the cell cycle. If it's present during DNA replication, it will substitute some of the thymidine in the strand.

Oxford Nanopore Technologies' (ONT's) MinION is a real-time device for ultra-long DNA sequencing. The device passes an ionic current through a protein pore. It unwound the double-stranded DNA substrate and inserts a single strand through the pore, causing disruption of the current. The current is recorded and then used as data for the Albacore base-calling software provided be ONT, which returns the appropriate DNA sequence. According to data released by ONT, the DNA moves through the pore in 6-mers, and the current signal for each 6-mer fits a Gaussian distribution. Additionally, there is a clear difference in the event distributions between 6-mers containing thymidine and BrdU, which gives us the possibility to detect the BrdU - incorporated regions in the DNA using this technology.

DNA replication is the action of synthesizing two DNA replicas from one original DNA strand. This is done by unwinding the double-stranded DNA, forming a replication fork containing two single-stranded templates, and complementing the templates to create two double-stranded DNA replicas. The replication origin is a sequence in the genome where replication is initiated. One of the complementary strands is called the leading strand, and it is synthesized continuously in the same direction as the growing replication fork, away from the replication origin. The second strand is the lagging strand, and it is synthesized in the opposite direction in short, separated segments.

Our objectives is to detect active replication origins and replication fork movements in the individual cell level fast and accurately. There are techniques that allows us to do so, but with high throughput or accuracy[4,5]. Recent work presented us with the D-NAscent software, which allows us to get the log likelihood of BrdU in certain 6-mers from individual nanopore sequencing reads, and to detect BrdU - incorporated regions in reads. This software threshold for classifying a position in the read as BrdU was 2.5, which by their report gave a true positive rate of $\sim$60% and false positive rate of $\sim$3%[1].

The problem we ran into is that some basic predictions were not met in the results of the D-NAscent software. For example, we would expect that the percentage of read with positive BrdU regions will be $< 5\%$. In addition, the BrdU segments should be long, about 5-20Kb, and longer BrdU segments are expected with longer pulse of BrdU. We haven't seen this happened in the output of D-NAscent:

```
Z0.R10000.replic.region: 17%
Z1.R2000.replic.region: 42%
Z-1.R2500.region: 66%
Z1.R10000.region: 12%
Z0.R2500.replic.region: 18%
Z-1.R2000.region: 78%
Z0.R5000.replic.region: 16%
Z-1.R10000.region: 34%
Z1.R10000.replic.region: 24%
Z-1.R5000.replic.region: 33%
Z0.R2000.region: 46%
Z0.R2000.replic.region: 21%
Z-1.R2000.replic.region: 49%
Z-1.R10000.replic.region: 35%
Z1.R2500.region: 22%
Z0.R5000.region: 30%
Z1.R5000.replic.region: 24%
Z1.R2500.replic.region: 37%
Z-1.R5000.region: 45%
Z0.R2500.region: 41%
Z1.R2000.region: 25%
Z0.R10000.region: 22%
Z-1.R2500.replic.region: 43%
Z1.R5000.region: 16%
```

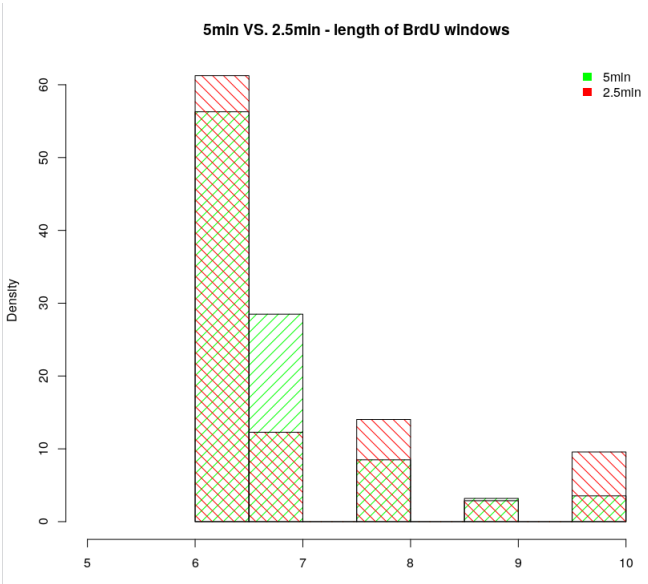


Figure 1: Analysis of the output of D-NAscent. Left: percentage of reads containing BrdU. Right: length of BrdU windows (segments) in 5 min vs 2.5 min exposure.

We would like to train an Hidden Markov Model to detect segments incorporating BrdU, using the log likelihood of certain 6-mers in the read, received from D-NAscent, as our data. Then, we could detect replication origins and fork directions by adding BrdU to cells in their S phase, sequencing DNA strands from those cells using nanopore sequencing, getting the log likelihood of BrdU in certain 6-mers in this sequencing using the D-NAscent software and identify the BrdU - incorporated segments using our algorithm.

Methods

Data

We used data obtained from DNA strands from cells of yeast (sacCer3) that were exposed to BrdU for 2.5 minutes in it's S phase. The strands passed nanopore sequencing and the D-NAscent software, which returned the file 2-5min_BrdU.detect (received from Itamar Simon lab). Those files contains the log likelihood of BrdU in certain 6-mers in this sequencing.

Additionally, we had lists of confirmed, likely and dubious segments of replication origin of sacCer3, that can be used to rate our results.

The Model

When plotted with the numerical estimation of the normal density function for all the read's scores, the numerical estimation of the normal density function for the 98.5 percentile and the mixture of the two, it can look somewhat close to a Gaussian Mixture Model. We decided to try an Hidden Markov Model with two states. The first state will have a Gaussian emission, and will represent segments without BrdU. The second will have a Gaussian Mixture Model emission, where one component is the Gaussian of the state without BrdU, and the second is a Gaussian representing the BrdU - incorporated 6-mers, such that over all this state will represent segments where some of the 6-mers incorporated with BrdU.

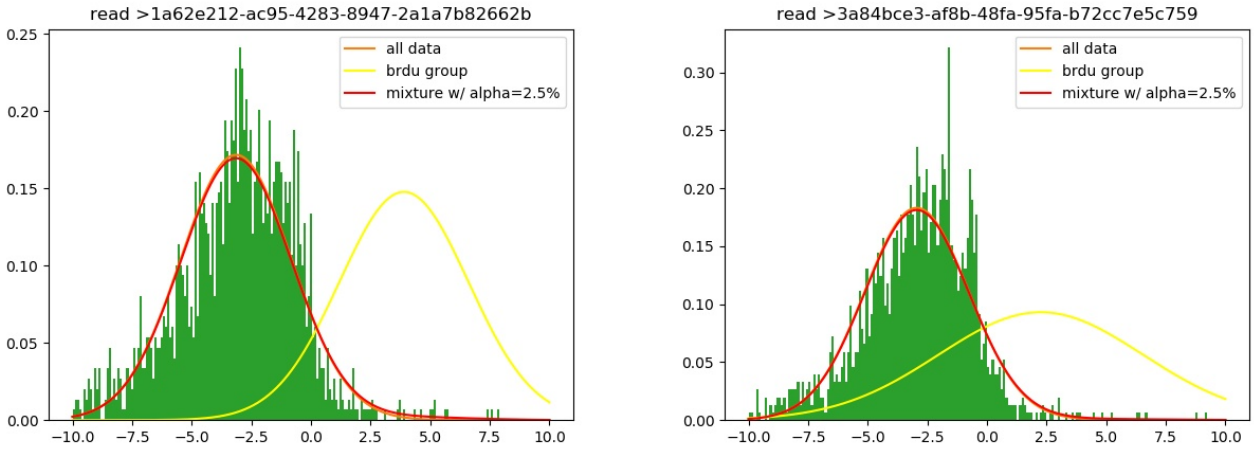


Figure 2: Histograms of scores for two reads, with a plot of GMM with two components: p_1 = the numerical estimation of the normal density function for all the read's scores, and p_2 = the numerical estimation of the normal density function for the 98.5 percentile (the 2.5% of the data with the biggest scores).

We trained our model on 1000 out of 155945 reads in the 2-5min_BrdU.detect file. The models were trained with a stream of scores from the reads (and not with separate reads).

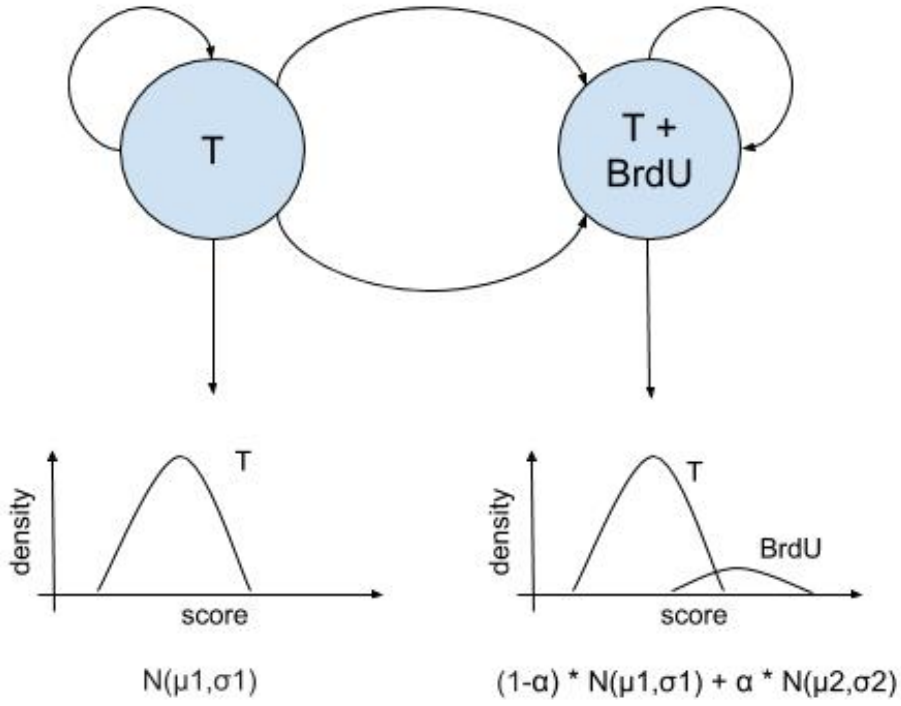


Figure 3: Illustration of the model: an HMM with two states, one represents no BrdU with Gaussian emission and the other represents segments with some BrdU with GMM emission.

Gaussian Mixture Model

Mixture model is a probabilistic model representing the present of number of sub-populations within an overall population. It's corresponds to the **mixture distribution** that represents the distribution of observation in the overall population: given a finite set of probability density functions (distributions of the sub-populations) $p_1(x), \ldots, p_n(x)$ and weights $w_1, \ldots, w_n$ such that $w_i \geq 0$ and $\sum_{i=1}^{n} w_i = 1$, the mixture distribution (meaning the distribution of the overall population) will be represented by the density function $f(x) = \sum_{i=1}^{n} w_i p_i(x)$. **Gaussian mixture model (GMM)** is a mixture model where the distribution of each sub-population is a Gaussian.

We used a GMM to represent the distribution of scores in a segment incorporated with BrdU. This GMM was composed of two components:

1. $p_1 = N(\mu_1, \sigma_1)$: Represents the majority of the 6-mers (out of those that was given a score by the D-NAscent), which was not affected by the BrdU.
2. $p_2 = N(\mu_2, \sigma_2)$: Represents the small group of the 6-mers which was affected by the BrdU.

In our data, the BrdU was inserted to the cells for 2.5 minutes, so we chose to take the weights to be $w_1 = 1 - \alpha$, $w_2 = \alpha$, where $\alpha = 0.025$, which gave us the GMM:

$$f(x) = (1 - \alpha) \cdot N(\mu_1, \sigma_1) + \alpha \cdot N(\mu_2, \sigma_2)$$

We fitted this model to our training data using the Expectation - Maximization algorithm (EM), that implement the maximum likelihood estimation for mixture models, with initial distributions: $p_1 =$ the numerical estimation of the normal density function for all the training data, and $p_2 =$ the numerical estimation of the normal density function for the 98.5 percentile (the 2.5% of the data with the biggest scores).

Hidden Markov Model

Hidden Markov Model (HMM) is a probabilistic model representing a Markov process in which the states are unobservable, meaning the state sequence that the model passes is unknown, and instead the output of each state in the sequence is visible. Formally, an HMM is composed by:

1. States: $\{S_1, ..., S_n\}$.
2. Transition matrix A with $a_{i,j} = \mathbb{P}(x_{t+1} = S_j | x_t = S_i)$, while x_t = the state corresponding to the observation in time t.
3. Initial state probabilities: $\pi = (\pi_1, ..., \pi_n)$.
4. Emissions distributions: $p_1, ..., p_n$, which can be density functions or discrete probabilities.

We built an HMM composed of two states:

1. T: represent segments without BrdU, with emission distribution $p_1 = N(\mu_1, \sigma_1)$ from the GMM we trained.
2. T + BrdU: represent segments where some of the 6-mers incorporated with BrdU, with the fitted GMM $f(x) = (1 - \alpha) \cdot N(\mu_1, \sigma_1) + \alpha \cdot N(\mu_2, \sigma_2)$ as emission function.

We fitted this model to our training data using the Baum - Welch algorithm, which is a version of the EM algorithm for HMM, that returns the transition matrix A and initial state probabilities that maximize the log likelihood of the training data. We took the initial parameters to be:

$$\pi = \begin{pmatrix} 0.8 \\ 0.2 \end{pmatrix}, A = \begin{pmatrix} 0.8 & 0.2 \\ 0.2 & 0.8 \end{pmatrix}$$

Once we have the model parameters, we have a prediction algorithm called Viterbi, that for a sequence of observation returns the states sequence that maximize the log likelihood of the given observation.

Results

Model Parameters

The training for our model (with 20 iterations for the GMM and for the HMM) returned the following values:

$$\pi = \begin{pmatrix} 0.806 \\ 0.193 \end{pmatrix}, A = \begin{pmatrix} 0.796620 & 0.203 \\ 0.196370 & 0.803 \end{pmatrix}, p_1 = N(-2.178, 2.173), p_2 = N(4.375, 3.364)$$

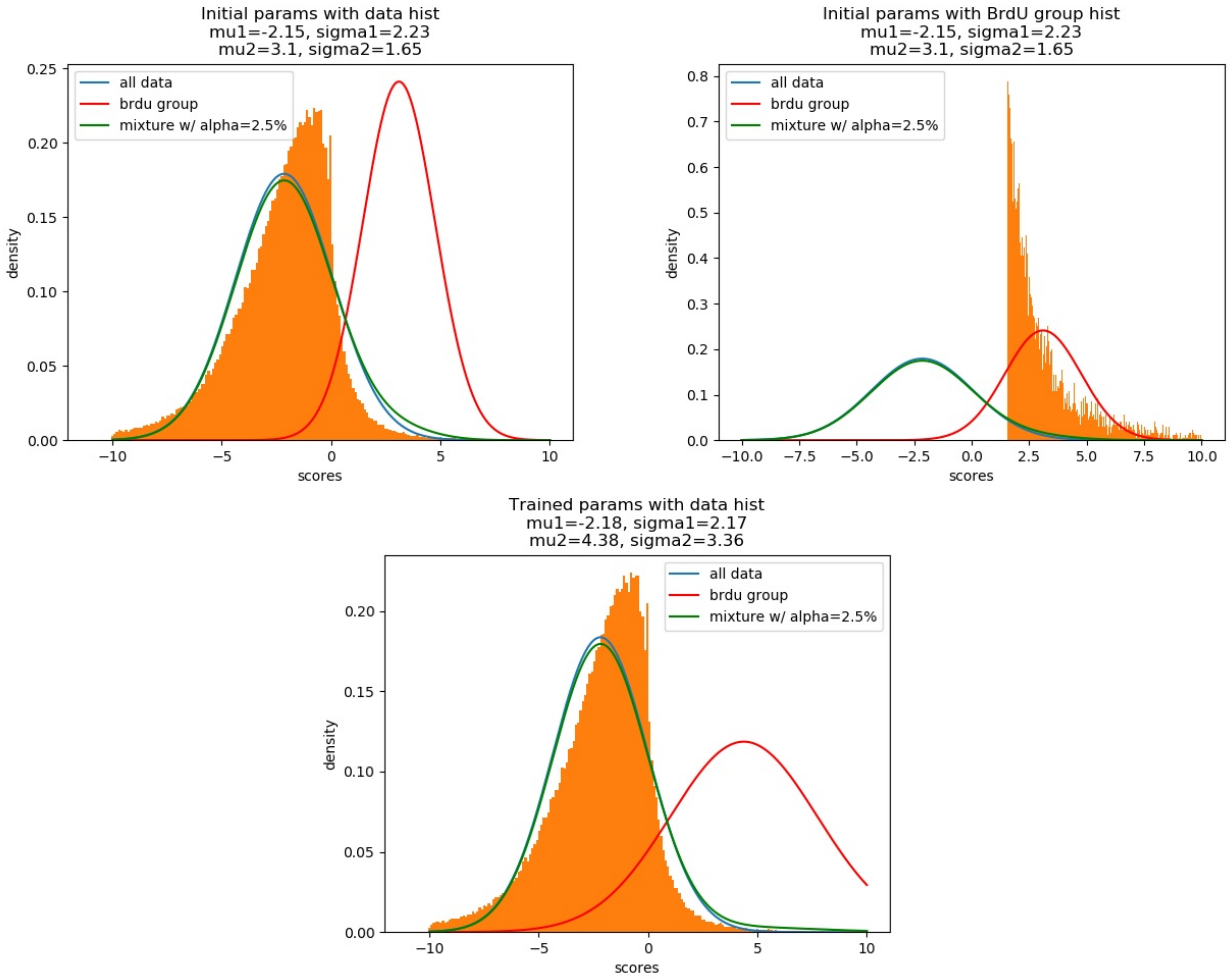
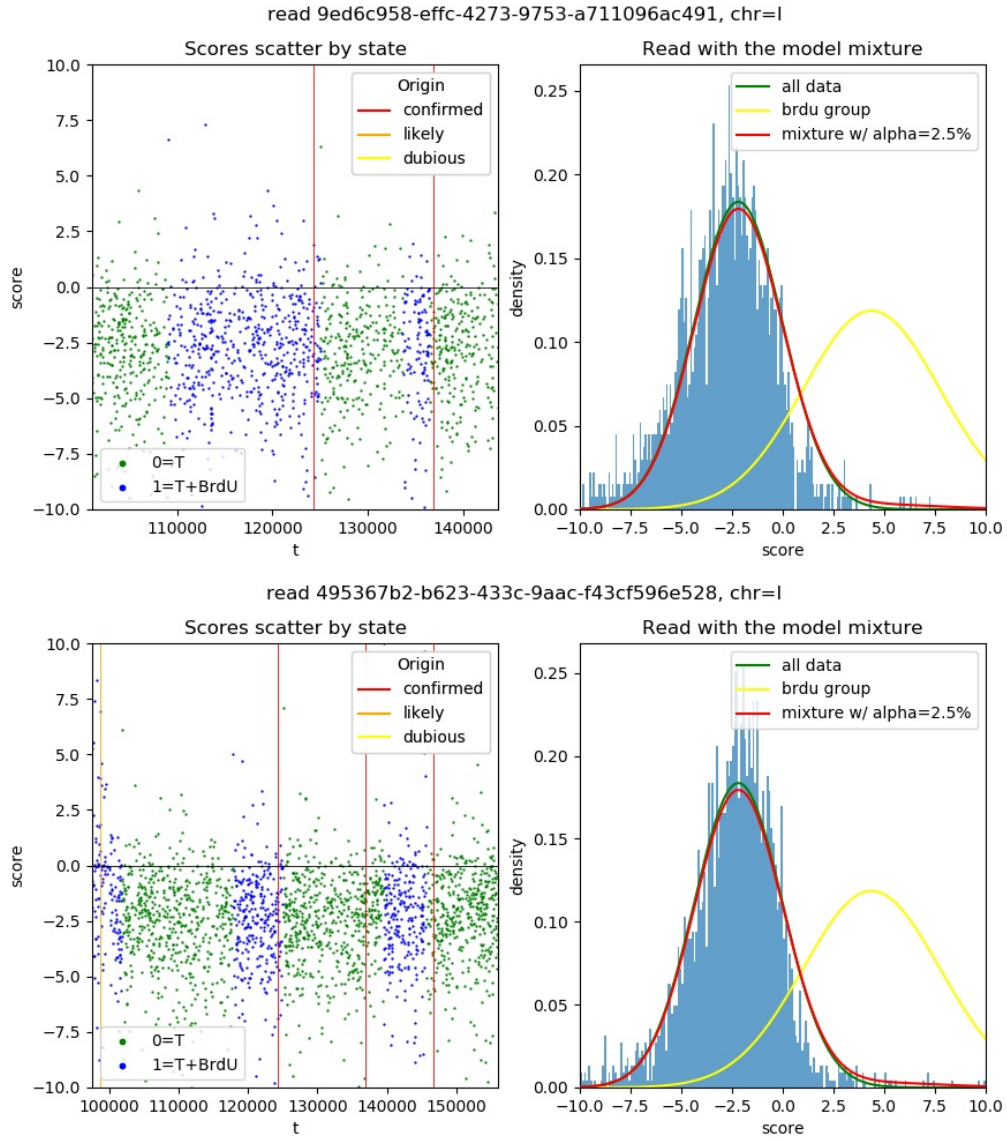


Figure 4: GMM initial parameters (top) and results parameters (bottom), with histograms of the data (stream of scores from 1000 first reads).

When taking stream of scores from several reads, the histogram does not look like a GMM. That is a problem for us, as we train our algorithm with a stream. In the future, we would like to find a way to fit the GMM with data from separate reads.

BrdU segmentation

We ran this model to predict the segmentation of some test reads. We plotted the 6-mer's scores for the read with their index in the read. Then we added the beginning of an replication origin as a vertical line (we separated between confirmed, likely or dubious origins). The desirable results will be that those line, in particular the confirmed ones, will intersect with a begging or an end of a BrdU segment. We doesn't yet have an effective way to rate the results, but from the plots it is clear that it's accuracy changes significantly from read to read. There are reads where the model recognizes the origins, but in many of the reads it missed several confirmed origins. In most of the reads where the algorithm got good results, the GMM is seem to be fit to the data of the read. In many of the reads where the algorithm missed the origins the GMM isn't at all close to the data histogram.



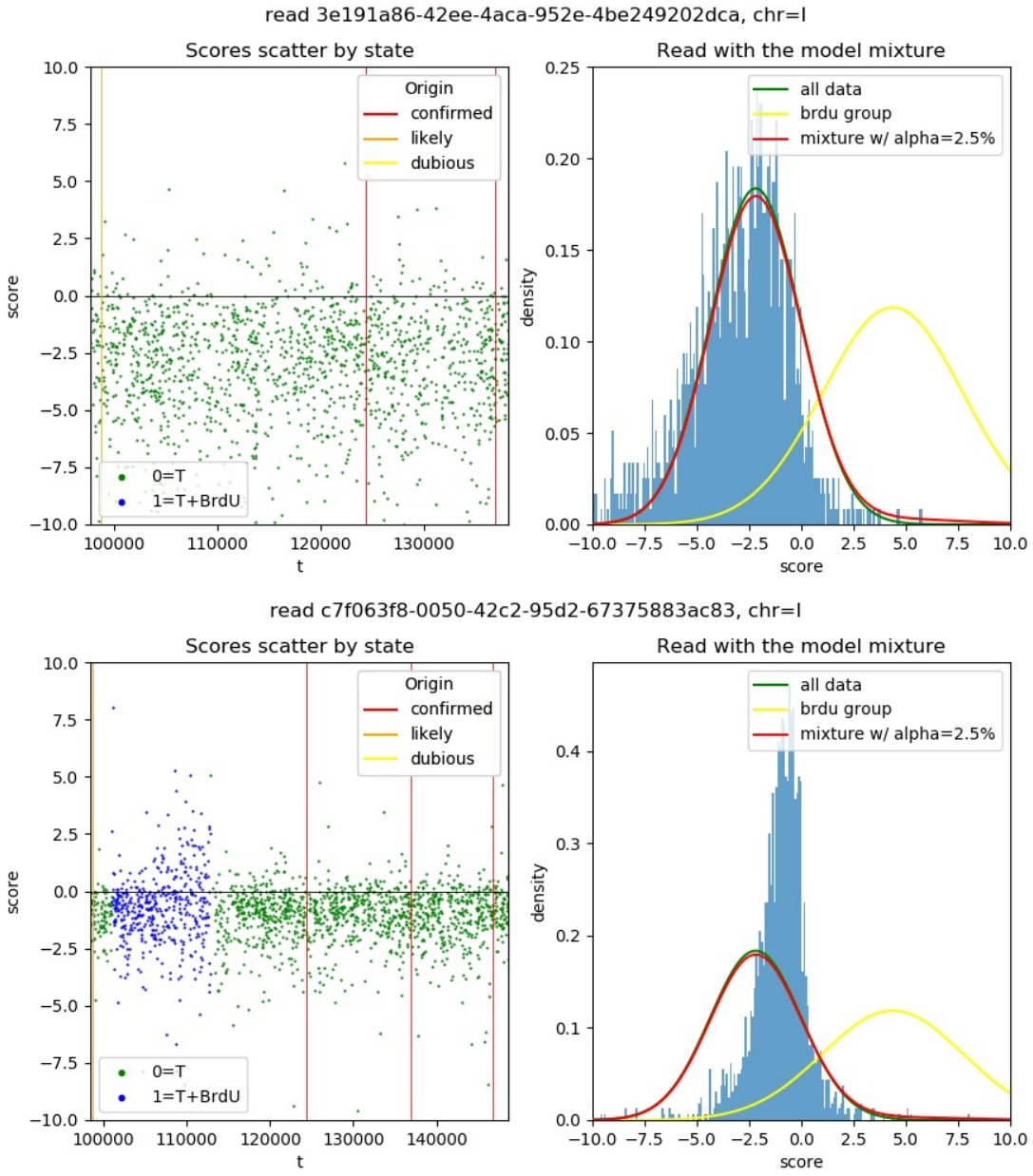


Figure 5: Plots of the HMM's results for 4 reads. Left: scatters of the 6-mer's scores for the read with their index in the read, with vertical lines representing the beginning of an replication origin (separated between confirmed, likely or dubious origins). Right: histograms of the read's scores with the GMM learned (from the 1000 first reads) and it's two components.

Discussion

The problem of accurate and fast identification of replication origins and replication fork movements is a complex one. With the development of the nanopore sequencing by ONT, we have now the possibility to sequence ultra-long DNA strands in short time, and the accuracy of this sequencing is improving. We are left with the question of identifying the BrdU - incorporated segments, by comparing the BrdU - incorporated DNA with regular sequencing. The method presented by the D-NAscent software reports good results, but as we saw it does not meet with some basic prediction. We used part of this software as we built an HMM model that rely on the software for it's input, and then run the Viterbi algorithm to return a segmentation of the read to two states, T and T+BrdU, that maximize the log likelihood of the observations (scores) of the read.

Our algorithm didn't achieved great results. We don't yet have a good way of rate it's performance, but it is clear that it's accuracy changes significantly from read to read.

There is a lot to do next: finding a good way to rate this kind of algorithm's performances, comparing its output with 2.5 minutes or 5 minutes exposure, finding the best initial parameters for the training process, and more.

Moreover, we can ask if using the scores received by the D-NAscent software is a good idea, and maybe find another way to present the data from the nanopore sequencing. Our model were trained with a stream of scores from the reads (and not with separate reads). When plotting a stream of data from the reads, we can see that it isn't similar to a GMM, even though each read can be seen as one. Next, we can adapt our model to fit better the stream of scores, or find a way to capture better the differences in each read's scores distributions.

References

[1] C.A. Mueller, et al."Capturing the dynamics of genome replication on individual ultra-long nanopore sequence reads, bioRxiv (2018), Article 44281

[2] Hua, H. & Kearsey, S. E. Monitoring DNA replication in fission yeast by incorporation of 5-ethynyl-2´-deoxyuridine. Nucl. Acids Res. 39, e60 (2011)

[3] Diermeier-Daucher, S. et al. Cell type specific applicability of 5-ethynyl-2´- deoxyuridine (EdU) for dynamic proliferation assessment in flow cytometry. Cytometry. A. 75, 535–546 (2009).

[4] Lacroix, J. et al. Analysis of DNA replication by optical mapping in nanochannels. Small. 12, 5963–5970 (2016).

[5] De Carli, F. et al. High-throughput optical mapping of replicating DNA. Small Methods 2, 1800146 (2018).