Mini-Project (ML for Time Series) - MVA 2022/2023

Paule Grangette paule.grangette@telecom-paris.fr
Mona Mokart mona.mokart@telecom-paris.fr

September 5, 2024

1 Introduction and contributions

1.1 SoccerCPD

SoccerCPD is a method of temporal analysis on soccer matches that consists in detecting formation changes within the match as well as player main role changes. The authors of the article work in two steps determining first the formation changes and then within each of these formation periods, the role periods.

To avoid working directly on the coordinates of the players, the authors worked on adjacency matrices and role sequences of the players.

For the formation change point detection (CPD) they use a g-seg method implemented in R based on graphs with recursion. Then they apply criteria such as a p-value threshold, a minimum duration or a minimum distance between before and after each selected change point to validate if it is significant.

For main role CPD, the authors work on the role sequence of the players. they use the same g-seg method on the sequence of role assignments using another type of distance for the algorithm. They still use recursion and a criterion on the "before" and "after" sequences. The overall context of the method is explained in the Figure 5.

1.2 Our work

Since the authors use a method simply imported into R to do their segmentation, it is not possible to implement the method ourselves. The code provided with the paper is more about computing the adjacency matrices and doing clustering on the determined formations and roles. Moreover we had access to the complete data of one single match because the data of the other matches were aggregated so it was very limiting.

First Mona, used and modified the authors code to obtain the results of the given single match to reproduce the comparison of the g-seg method with other classic CPD methods for formation and role periods. After that, she first worked on external data and then synthesized matches data to do better analyses. From this she was able to better compare the method with other new methods not proposed by the authors and to analyze the influence of the parameters used by the authors. The analysis part is fully coded by us. Secondly, Paule questioned the relevance of using graphs to detect formation and roles changes. Indeed, the authors start by making a role assignment to detect formation changes and then role changes. Paule developed a method that uses multivariate time series to compute placement changes to first detect role permutations and then detect formation changes. The performance of evaluation of the method is quite difficult since we only had one match to test our code. All has been coded by us (*players_MCPD.py*, *form_vs_role.py*, *MCPD.ipynb*) except SSA class.

2 Method

2.1 CPD with adjacency matrix and role sequence

The authors chose not to use directly the GPS raw data but instead on adjacency matrices. To do that they first use an Expectation-Maximization (EM) method with the Hungarian algorithm in order to associate each player $p \in P$ with a role $X_p(t) \in [\![1, 10]\!]$ at each instant t of the match (note that $X_p(t) \neq X_q(t)$ if $p \neq q$). The objective of this is that since the players move a lot and change roles often it is difficult to have an overall view of the formation. Roles are more stable than players. Then from the position of the roles, they perform Delaunay triangulation to get the adjacency matrix $A(t)$.

$$A_{pq}(t) = \begin{cases} 1, & \text{if } X_p(t) \text{ and } X_q(t) \text{ are adjacent} \\ 0, & \text{else} \end{cases}$$

Formation CPD : To detect formation changes the g-segmentation method is applied on the sequence of $\{A(t)\}_{t=1,\ldots T}$ and computes a statistic R(t). Then, at each recursion a potential change point $\tau = \arg\max R(t)$ is validated if it verifies the three criteria:

- p-value(τ) < 0.01
- minimal duration of a formation period : 5min
- $\|\text{avg}(A[t_{start} : \tau]) - \text{avg}(A[\tau : t_{end}]\|_1 > 7$

Here, the cost function for the g-seg method is the l1 distance.

Main Role CPD : It is important to distinguish the role of a player from his main role. The role of the player $X_p(t)$ is already determined by the EM algorithm. The role periods denote a change in the **main** role. To detect the role periods, g-segmentation is applied to the sequence of roles. There is same the criterion on the p-value, in addition to the condition that the main role (most frequent role) attribution must be different for the period before and after the change point τ. Here, the cost function for the g-seg method is the Hamming distance.

2.2 Alternative approach with multivariate time series for CPD with GPS data

To elaborate the method that test the relevance of graphs' structure, we used the only match that was given by the authors (using the second half-time for the 'training' and the first for validation). We also used other data (SKAB data) to see performance of both methods in change points detection.

The variables in the multivariate time series are the euclidean positions of each player in the match. As we wanted to find patterns in the multivariate time series, we first proceeded a principal component analysis (PCA) (cf. Figure 8).

Then, as the time series was very noisy, we performed pre-processings before applying a CPD method. We first applied a singular spectral analysis (SSA) to get the first component of the decomposition which corresponds to the time series denoised. To remove the remaining outliers, we applied also a filter to the values repartition in the histogram of the signal. Finally, to detect change points, we applied the Pruned Exact Linear Time (PELT) method from the *ruptures* library. Finally, we elaborated a method to check whether the detected change points were significant or not, inspired by the one of the authors. To find significance of change points, we use two conditions : having subpart-time of at least 5min and having a sufficient relative error between the mean position of players before and after change points. Then we look for permutations which implies or not role permutations. If no permutation there is a formation change point, else, there is a role permutation but it can be also a formation change point. We detect that last part by aligning roles and checking again the same relative error.

3 Data

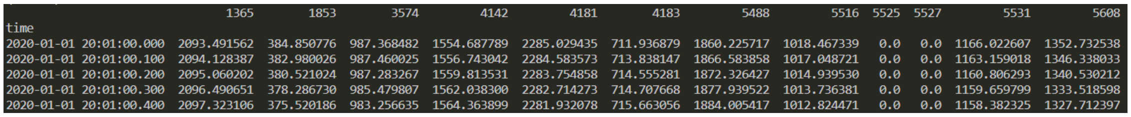
3.1 Raw data

In our article, we are studying GPS data of players during football matches. The authors had collaborated with the Korean First League of Football and had experts annotate formations and role permutations within 750 teams during matches (the time of changements, types of formation, aligned roles by team). Because of the confidentiality of these data, they only provided data for one match in their github and we could not run parts of the code concerning the clustering parts of the methods. This was a bit limiting to compare the results of authors on other data but we managed to compare the CPD part.

The data provided contained information about players in the match (such as id, uniform name), metadata such as start and end of each half of the game, position in width and depth of each player on the field and his speed each tenth of second.

We needed to take care of several things with these GPS data. First, after the first half time, there is a symmetry in the positional data due to change of field. Secondly, the GPS data recording starts before the game kicks off and continues through halftime but meta-data were provided for that.

When we tried to implement an alternative method to graphs, we needed to apply methods for



```
                                1365        1853        3574         4142         4181        4183         5488         5516  5525  5527         5531         5608
time
2020-01-01 20:01:00.000  2093.491562  384.850776  987.368482  1554.687789  2285.029435  711.936879  1860.225717  1018.467339   0.0   0.0  1166.022607  1352.732538
2020-01-01 20:01:00.100  2094.128387  382.980026  987.460025  1556.743042  2284.583573  713.838147  1866.583858  1017.048721   0.0   0.0  1163.159018  1346.338033
2020-01-01 20:01:00.200  2095.060202  380.521024  987.283267  1559.813531  2283.754858  714.555281  1872.326427  1014.939530   0.0   0.0  1160.806293  1340.530212
2020-01-01 20:01:00.300  2096.490651  378.286730  985.479807  1562.038300  2282.714273  714.707668  1877.939522  1013.736381   0.0   0.0  1159.659799  1333.518598
2020-01-01 20:01:00.400  2097.323106  375.520186  983.256635  1564.363899  2281.932078  715.663056  1884.005417  1012.824471   0.0   0.0  1158.382325  1327.712397
```

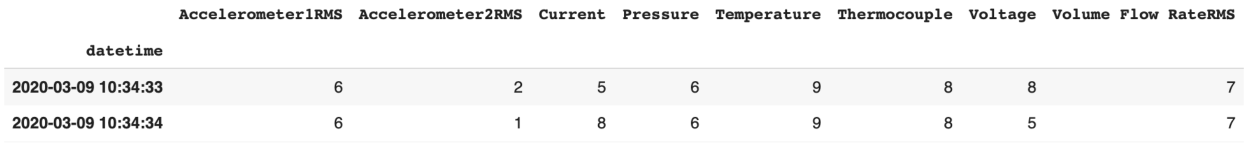
Figure 1: First rows of multivariate time series : columns'name are players' id.

denoising and outliers removal, as seen during lessons. Normally, when we apply these methods, we make an assumption of stationarity which is no longer validated in our case. This is why it is sometimes difficult, especially with the histogram method as we can see in figure 3, to remove outliers around change points or even within the stationarity phases.

3.2 Additional data

SKAB : In order to have a large enough sample to analyze the g-seg method, we worked on the SKAB dataset. SKAB gathers several features of the flow of a fluid in a circuit over time and has the advantage of having the groundtruth of change-points. Since the g-seg method needs discrete values as input, we discretized the continuous features by dividing values into bins (cf Figure below). The dataset used had 16 samples of sequences.

Synthesized Dataset : SKAB was not a very suitable dataset for a cpd method applied to soccer



datetime	Accelerometer1RMS	Accelerometer2RMS	Current	Pressure	Temperature	Thermocouple	Voltage	Volume Flow RateRMS
2020-03-09 10:34:33	6	2	5	6	9	8	8	7
2020-03-09 10:34:34	6	1	8	6	9	8	5	7

matches. This is why we tried to recreate the adjacency matrix dataset from the groudtruths and aggregated data at our disposal. We had access to the formation periods as well as the average value of the adjacency matrix corresponding to the period. Thus by interpreting the average as a distribution it was possible to create graphs from the distributions. You can see in Figure 6 an example of synthesized graphs. We reproduced in total 20 sequences (20 soccer matches) of adjacency matrices, with time intervals of 5 seconds.

Due to lack of information about the players, we could not recreate the role sequences, so we could not analyze the method on the main role changes.

4 Results

4.1 Results on the authors method

We analyzed the results with two metrics, the Hausdorff distance and the F1 score for time series explained in this article. First we reproduced the comparison with the Binary Segmentation method with different kernel and with the rank method. The results for the match 17985 are in Table 3 & 4, however they are not relevant since it is computed on only on match. Here are the results for the SKAB and synthesized datasets, for the synthesized dataset we added a new method "window" which computes the discrepancy curve and then applies a peak detection.

Method	gseg-avg	gseg-union	linear	rbf	cosine	rank
F1 score	0.47	0.47	0.13	0.40	0.32	0.40
Hausdorff (in sec)	221	221	361	217	238	230

Table 1: Results for SKAB dataset

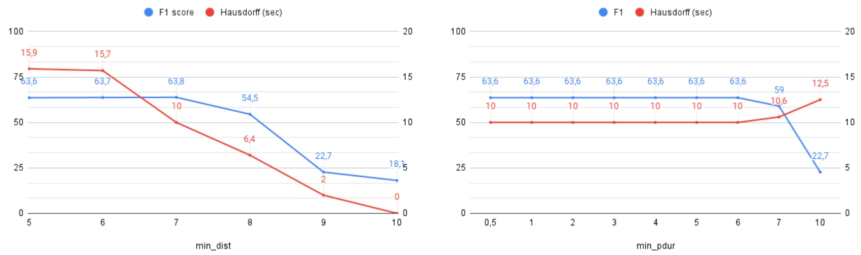
Method	gseg-avg	gseg-union	linear	rbf	cosine	rank	window
F1 score	0.64	0.64	0.0	0.77	0.82	0.86	0.73
Hausdorff (in sec)	10	10	nan	8.25	7	5.75	7.22

Table 2: Results for the reproduced dataset

In Table 1 & 2 all the methods seem bad for SKAB even if gseg seems better, but in the synthesized dataset it is not the case and the rank method is by far the best one. We stopped working on SKAB after that since it was not adapted data.

Then we focused on the gseg-avg method, and played with different parameters used for the criteria in formation periods, such as the max p-value, the min of duration period and the min of distance between the "before" and "after" sequences.

As you can see on the following figure, for the max p-value we tried values in $[0.5, 0.1, 0.05, 0.01, 0.005, 0.001]$ and we also tried to change the l1 distance to the l2 distance but for both experiences it did not change anything.



Performance wrt min_dist

Performance wrt min_pdur

Figure 2

4.2 Approach multivariate time series

When we performed PCA to get a denser representation of the data, we used the second component for the rest of the analysis as the data were not centered. Then, when we applied SSA, we needed to segment the signal in frames of 5000 values at most, as the time series was very large (around 22000 time values) : a trade-off was required as well as for SSA window size between the execution time and frame size large enough to not consider stationary change as outlier. (see Figure 9 for results). To remove remaining outliers, we use quantile thresholds in the histogram method, which we set to 0.05 and 0.95 after several trials : we prefered to cut too many values rather than to let too many outliers through. Finally, the PELT method has some drawbacks : its execution requires several minutes (around 3 to 5min) by time series, certainly because we use a huge value of β, otherwise we would have had too many outputs.

The results were satisfying in terms of change point detection since we detect three change points (see Figure 3). The checking part of the approach, gives also good results since our results are consistent with ground truths (see Figure 4), even though it is difficult to evaluate it with so few data. We can see that the method works less well with the SKAB dataset (see Fig. 14).
Nevertheless, the objective of this approach was not so much to compete in terms of performance with the authors' method, but rather to try to prove that one could also free oneself from the graph method, which we achieved to some extent.

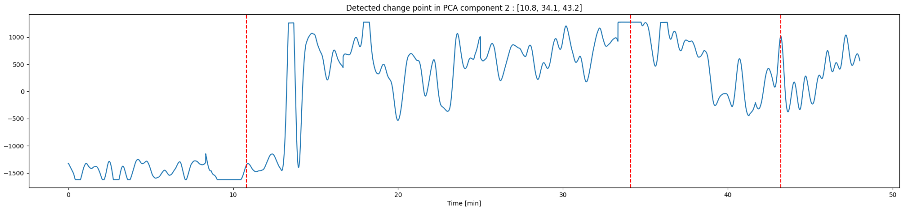
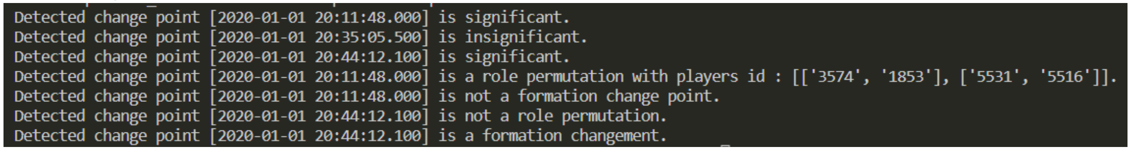


Figure 3: Cleaned signal with PELT parameter $\beta = 500000000$, ground truth : $[11, 41]$



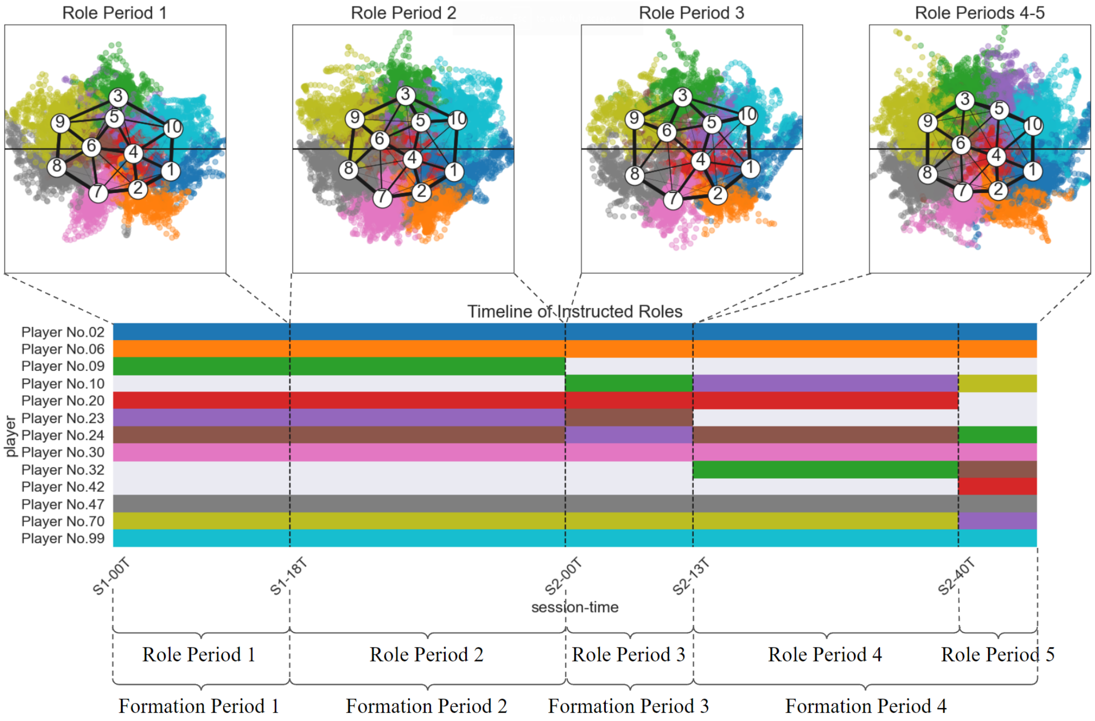
```
Detected change point [2020-01-01 20:11:48.000] is significant.
Detected change point [2020-01-01 20:35:05.500] is insignificant.
Detected change point [2020-01-01 20:44:12.100] is significant.
Detected change point [2020-01-01 20:11:48.000] is a role permutation with players id : [['3574', '1853'], ['5531', '5516']].
Detected change point [2020-01-01 20:11:48.000] is not a formation change point.
Detected change point [2020-01-01 20:44:12.100] is not a role permutation.
Detected change point [2020-01-01 20:44:12.100] is a formation changement.
```

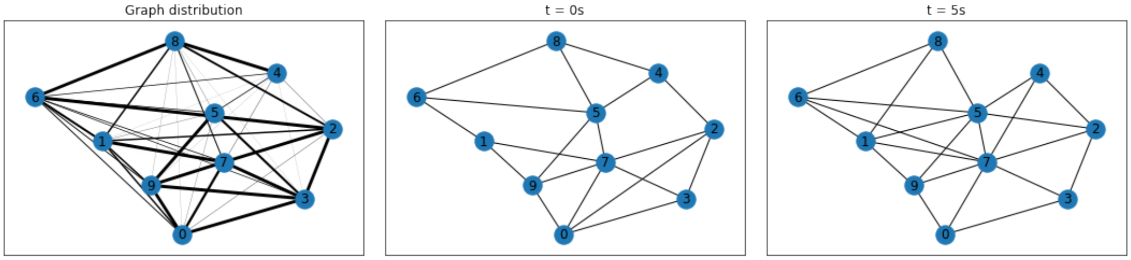
Figure 4: Detection significant change point and formation and role changes, ground truth : $[11 : \text{role permutation}, 41 : \text{formation change}]$

5 Appendix

5.1 SoccerCPD method

Figure 5: SoccerCPD method





Average adjacency matrix | Reproduced graph at t=0s | Reproduced graph at t=5s

Figure 6: Example of synthesized $A(t)$

Method	gseg-avg	gseg-union	linear	cosine	rbf	rank
Hausdorff (in sec)	0	1	32	2	0	0

Table 3: Results for match 17985, formation period

5.2 Alternative approach on multivariate time series

Method	gseg-avg	gseg-union
Hausdorff (in sec)	0	0

Table 4: Results for match 17985, role period

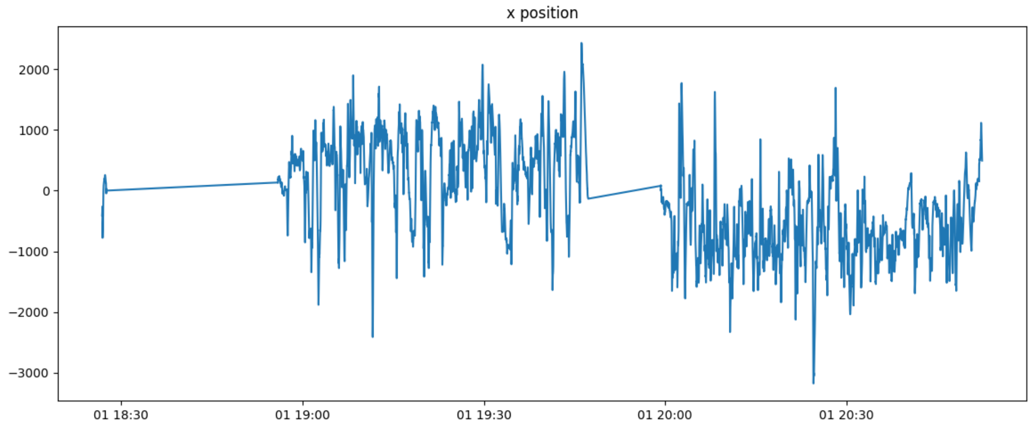


Figure 7: Raw data of the centered position in length of a player with respect to the position of the team

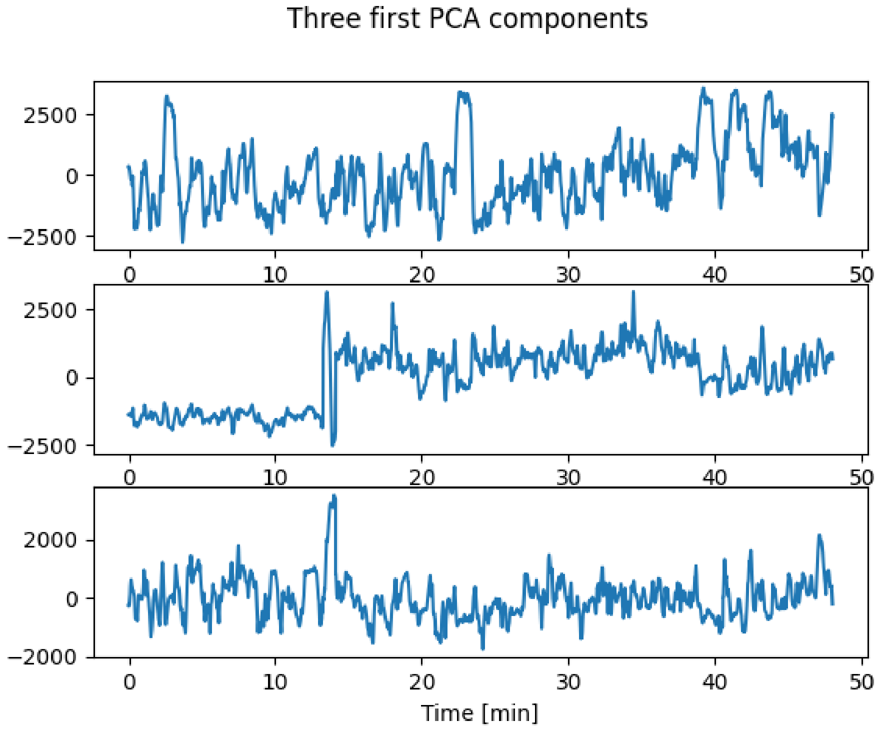


Figure 8: Three first components of PCA over the multivariate time series (GPS data)

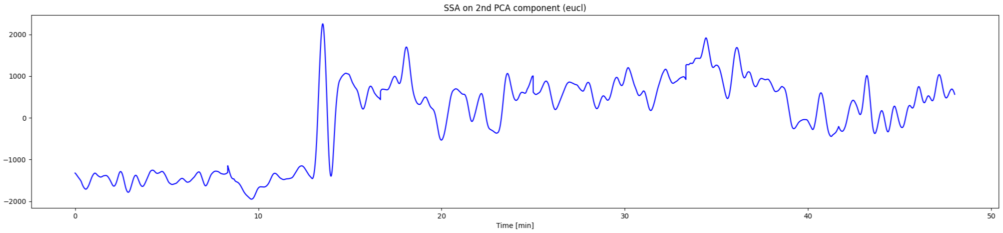


Figure 9: Application of SSA over the second component of PCA (GPS data)

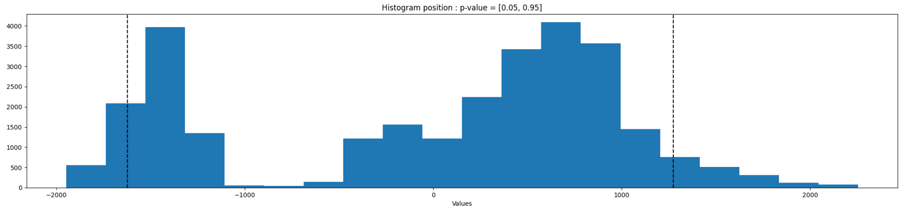


Figure 10: Histogram of the first component of SSA (GPS data)

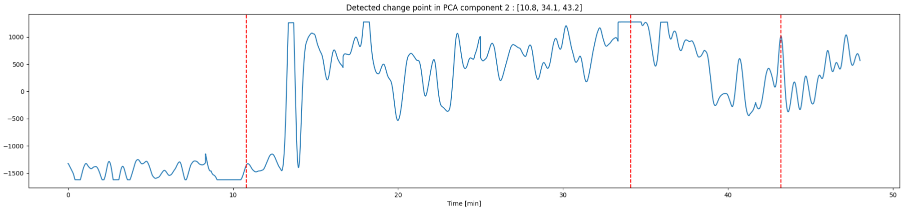


Figure 11: Cleaned signal (GPS data) with detected change points for $\beta = 500000000$, ground truth : $[11, 41]$

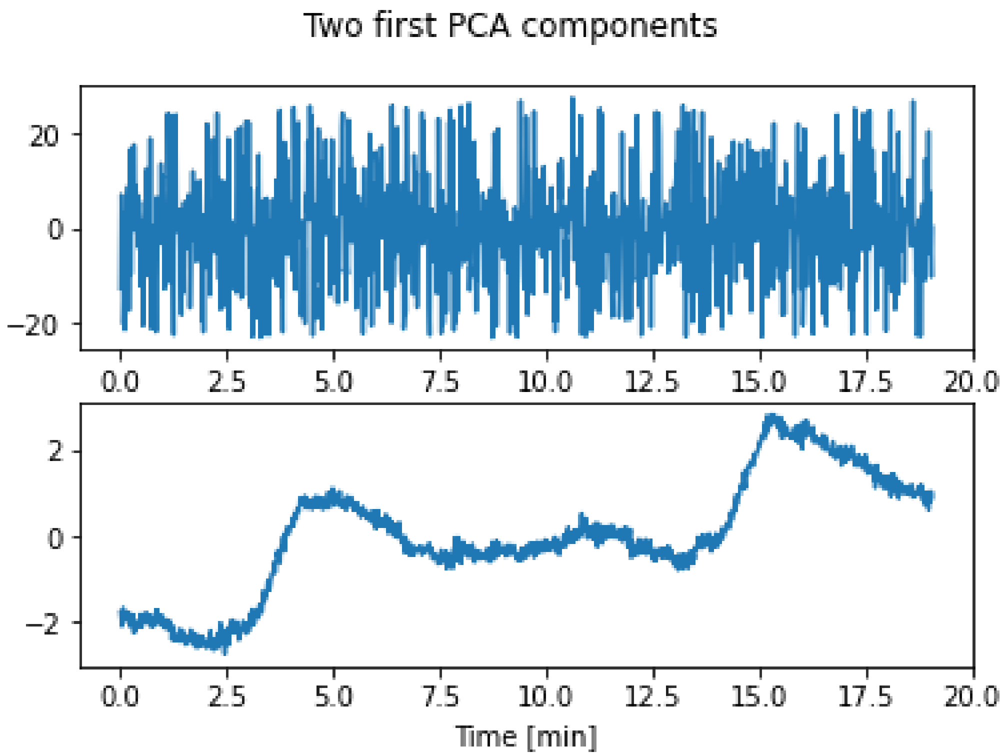


Figure 12: Two first components of PCA over SKAB data

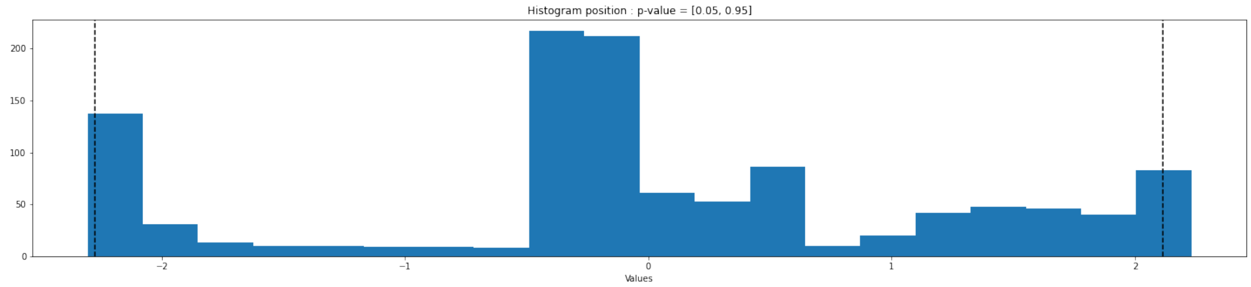


Figure 13: Histogram of the first component of SSA on SKAB dataset

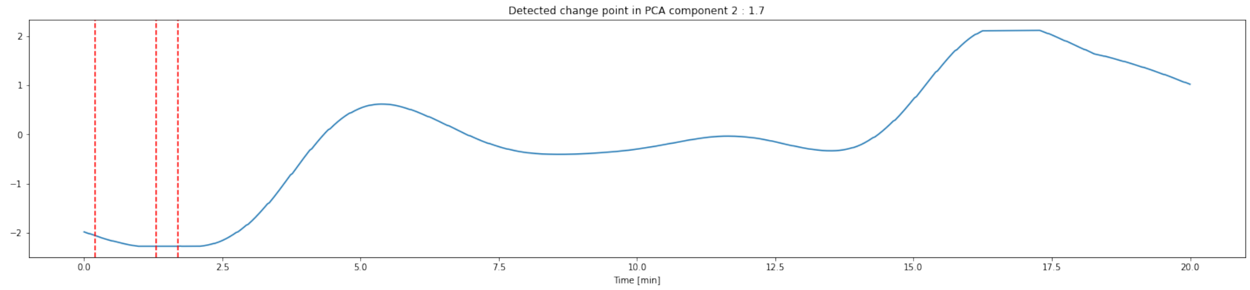


Figure 14: Cleaned signal (SKAB dataset) with PELT parameter $\beta = 50$, with ground truth : $[10, 11, 16, 17]$