# Mini-Project (ML for Time Series) - MVA 2025/2026

Adonis Jamal adonis.jamal@student-cs.fr
Fotios Kapotos fotiskapotos@gmail.com

December 12, 2025

**What is expected for these mini-projects?** The goal of the exercise is to read (and understand) a research article, implement it (or find an implementation), test it on real data and comment on the results obtained. Depending on the articles, the task will not always be the same: some articles are more theoretical or complex, others are in the direct line of the course, etc... It is therefore important to balance the exercise according to the article. For example, if you have reused an existing implementation, it is obvious that you will have to develop in a more detailed way the analysis of the results, the influence of the parameters etc... Do not hesitate to contact us by email if you wish to be guided.

**The report** The report must be at most FIVE pages and use this template (excluding references). If needed, additional images and tables can be put in Appendix, but must be discussed in the main document. The report must contain a precise description of the work done, a description of the method, and the results of your tests. Please do not include source code! The report must clearly show the elements that you have done yourself and those that you have reused only, as well as the distribution of tasks within the team (see detailed plan below.)

**The source code** In addition to this report, you will have to send us a Python notebook allowing to launch the code and to test it on data. For the data, you can find it on standard sites like Kaggle, or the site https://timeseriesclassification.com/ which contains a lot of signals!

**The oral presentations** They will last 10 minutes followed by 5 minutes of questions. The plan of the defense is the same as the one of the report: presentation of the work done, description of the method and analysis of the results.

**Deadlines** Two sessions will be available :

- **Session 1**
  - Deadline for report: December 14th (23:59)
  - Oral presentations: December 15th and 17th (precise times TBA)
- **Session 2**
  - Deadline for report: January 4th (23:59)
  - Oral presentations: January, 5th and 7th (precise times TBA)

# 1 Introduction and contributions

In the rapidly evolving field of sports analytics, understanding team tactics is crucial for performance analysis. In fluid team sports like soccer, analyzing team formation is one of the most intuitive methods to interpret tactics from the perspective of domain participants. However, extracting these insights from spatiotemporal tracking data presents significant challenges. Players frequently switch positions temporarily or engage in abnormal situations like set-pieces, making it difficult to distinguish between a genuine tactical change instructed by a coach and a transient movement noise. Existing approaches often fail to address this dynamic, either assuming formations remain constant throughout a match or reacting too sensitively to frame-by-frame changes [1, **?**, **?**].

To address these limitations, this project studies the framework proposed by Kim et al. in [2]. The article introduces a novel, unsupervised change-point detection (CPD) framework designed to distinguish tactically intended formation changes from temporary role swaps. The methodology consists of two distinct phases: Formation Change-Point Detection (FormCPD), which identifies shifts in the global spatial configuration of the team, and Role Change-Point Detection (RoleCPD), which detects long-term tactical changes in individual player roles within a specific formation period. By utilizing graph-based statistics and Delaunay triangulation, the method aims to provide a robust timeline of tactical instructions.

**Contributions and Work Repartition** In alignment with the course requirements, this report details our reproduction and extension of the SoccerCPD framework. The organization of the project is as follows:

- **Work Repartition:** The workload was divided based on the two-step nature of the pipeline. Fotios Kapotos focused on the first stage, FormCPD, analyzing the generation of role-adjacency matrices and the segmentation of formation periods. Adonis Jamal concentrated on the second stage, RoleCPD, investigating the sequence of role permutations and the detection of intra-formation tactical shifts.
- **Source Code:** We utilized the open-source Python implementation provided by the authors to run the core algorithms. However, we performed a deep-dive analysis into the code to understand the underlying mechanics of the discrete g-segmentation and re-implemented specific analytical components to verify limitations.
- **Experiments:** Beyond verifying the results on sample data, we conducted a novel experiment titled the "Messi Effect". This analysis aims to compare formation structural differences and player stationarity in matches involving Lionel Messi versus those without him, testing the model's ability to capture player-centric tactical gravity. Moreover, we explored the stationarity of players across different formations and roles, providing insights into individual player behavior within team tactics. Finally, we address a limitation of the original method by providing offensive and defensive formation change-point detection, enhancing the interpretability of tactical shifts.

# 2 Methodology

## 2.1 Formation Change-Point Detection (FormCPD)

The FormCPD pipeline transforms raw player trajectories $X(t) \in \mathbb{R}^{N\times 2}$ into a temporal sequence of formation graphs $\{A(t)\}_{t\in T}$. Discrete g-segmentation is then applied to the sequence $\{A(t)\}$ to detect change-points $T_1 < \cdots < T_m$ such that $A(t) \sim \mathcal{F}_i$ for all $t \in T_i$. Finally, the resulting segments are clustered through role alignment and hierarchical aggregation to extract canonical formation types.

### 2.1.1 Role Assignment

The first step of FormCPD assigns each player a latent spatial role following [1], aiming to model stable positional zones rather than player identities. Each role $k \in \{1, \ldots, N\}$ is represented by a Gaussian component $(\mu_k, \Sigma_k)$. At time $t$, player positions $x_{i,t} \in \mathbb{R}^2$ are modeled as

$$x_{i,t} \sim \mathcal{N}(\mu_{z_{i,t}}, \Sigma_{z_{i,t}}),$$

under the hard constraint that no two players share the same role at a given frame.

Parameters are estimated via a constrained EM algorithm. In the E-step, a cost matrix built from negative log-likelihoods is solved using Hungarian matching to assign players to roles one-to-one. In the M-step, $(\mu_k, \Sigma_k)$ are updated from all positions assigned to each role across time. This produces hard, frame-wise coupled assignments while enforcing temporal consistency through shared role parameters.

Figure 2 illustrates how role-based coloring reveals coherent spatial clusters even when player trajectories cross, highlighting the ability of the model to capture stable formation structure.

### 2.1.2 Encoding Spatial Structure via Delaunay Triangulation

Once role assignments are available, each frame is encoded as a graph describing the local spatial relationships between roles. This is obtained by computing a Delaunay triangulation of the role locations

$$V(t) = \{v_1(t), \ldots, v_N(t)\} \subset \mathbb{R}^2.$$

The Delaunay triangulation connects two roles when they share an edge in the triangulated mesh, defining adjacency based solely on relative geometry without introducing distance thresholds or fixed neighborhood sizes. From this, a binary role-adjacency matrix $A(t) \in \{0,1\}^{N\times N}$ is constructed.

Each frame is therefore represented as an undirected graph whose vertices correspond to roles and whose edges encode local spatial proximity. This representation emphasizes the *topological* structure of the formation rather than precise metric distances. Combined with role indexing, it remains invariant to player permutations: transient exchanges of positions do not alter the role-adjacency graph as long as players remain within the same spatial zones.

Figure 3 shows Delaunay triangulations computed from simulated role positions before and after a formation change from 4–4–2 to 3–4–3. The change induces a clear reconfiguration of spatial adjacency relationships.

Figure 4 presents the corresponding role-adjacency matrices. Instantaneous matrices (top row) exhibit small fluctuations due to geometric noise, whereas averaging over each formation segment (bottom row) yields stable, formation-specific patterns used as inputs for clustering.

#### 2.1.3 Change-Point Detection via Discrete g-Segmentation

Formation change-points are detected on the temporal sequence of role-adjacency matrices $\{A(t)\}$ using discrete g-segmentation [3], a graph-based CPD method suitable for high-dimensional and discrete observations. Pairwise dissimilarities are measured with the Manhattan distance

$$d_M\big(A(t), A(t')\big) = \|A(t) - A(t')\|_{1,1}.$$

Frames with excessive role switching (rate $> 0.7$) are removed to discard abnormal game situations. The scan statistic identifies candidate change-points $\tau$, which are retained as significant if they satisfy constraints on $p$-value ($< 0.01$), segment duration (at least 5 minutes on both sides), and inter-segment dissimilarity. A recursive procedure is applied to recover multiple change-points, yielding a partition of the match into formation intervals $T_1 < \cdots < T_m$.

#### 2.1.4 Formation Clustering

Each formation segment $T_i$ is represented by a formation graph $F(T_i) = (V(T_i), A(T_i))$, obtained by averaging the role locations and role-adjacency matrices over the stable frames $t \in T_i^*$.

When comparing formations across segments or matches, role indices are not directly comparable since labels are arbitrary. We therefore align roles between two formation graphs by solving an optimal assignment problem using the Hungarian algorithm on the Euclidean distances between vertex sets $V(T_i)$ and $V(T_j)$, yielding a permutation matrix $Q$. This alignment ensures that homologous spatial roles are matched prior to comparison.

Formation similarity is then measured using the Manhattan distance between aligned adjacency matrices,

$$d(F_i, F_j) = \|QA(T_i)Q^\top - A(T_j)\|_{1,1},$$

which captures differences in *topological structure* rather than absolute positioning. Based on these pairwise distances, agglomerative hierarchical clustering is applied to group formation graphs into a small number of canonical formation types.

### 2.2 Role CPD

The goal of RoleCPD [2] is to detect long-term tactical changes in player roles (e.g., a winger swapping sides with another winger permanently) while ignoring temporary switches (e.g., overlapping runs or covering defensive duties). This process operates within a single Formation Period $(T_i)$ identified in the previous step.

#### 2.2.1 Formal Representation

We define the inputs and mathematiccal framework based on the SoccerCPD protocol:

- **Input:** A sequence of "Temporary Role Permutations" $\{\pi_t\}_{t=1}^{|T_i|}$.

- **Role Permutation ($\pi_t$):** At every frame $t$, the Role Representation step assigns a role $X_p$ to every player $p$. Since roles are distinct, this assignement is a permutation of the initial canonical role:

$$\beta_t(p) = \pi_t(X_p)$$

Where $\beta_t$ is the player-to-temporary-role mapping at time $t$.

#### 2.2.2 The Distance Metric

To detect a change, we must quantify the difference between the team's configuration at time $t$ and time $t'$. Since the data consists of permutations (non-Euclidean), we cannot use standard Euclidean distance. We use the Hamming Distance normalized by the number of roles ($N = 10$ outfield players):

$$d(\pi_t, \pi_{t'}) = \frac{1}{N} \sum_{p \in P} \mathbb{1}_{\pi_t(X_p) \neq \pi_{t'}(X_p)}$$

This metric represents the "Switch Rate" or the proportion of players whose roles differ between two frames.

#### 2.2.3 Change-Point Detection Algorithm

The paper utilizes Discrete g-segmentation, a graph-based change-point detection method effective for repeated observations in non-Euclidean space. The procedure is as follows:

1. **Preprocessing:** Calculate the Switch Rate relative to the dominant permutation. Exclude frames with a switch rate $> 0.7$ (likely temporary switches during set-pieces or abnormal situations).
2. **Segmentation:** Apply the dectection algorithm recursively on the sequence of permutations using the Hamming distance.
3. **Significance Test:** A change point $\tau$ is significant if:
    - The p-value of the scan statistic is $< 0.01$.
    - The segmentation duration is sufficient (robustness against noise).
    - The most frequent permutations (Instructed Roles) in the segments before and after $\tau$ must be distinct.

## 3 Data

Our study utilizes two distinct types of spatiotemporal datasets to evaluate the SoccerCPD framework: high-frequency GPS tracking data used in the original SoccerCPD study, and event-stream data for our novel "Messi Effect" experiment. This section analyzes the properties, quality, and preprocessing requirements of these signals.

### 3.1 Fitogether Tracking Data

For the reproduction of the core algorithm and methods, we utilized the sample dataset provided by the authors.

- **Source & Format:** The data originates from the K-League (South Korea) and consists of player trajectories recorded at 10 Hz. The raw input is a sequence of 2D coordinates $x_{i,t} \in \mathbb{R}^2$ representing the position of player $i$ at frame $t$.
- **Data Diagnosis & Quality:** We performed a preliminary diagnosis of the signal stationarity and completeness. The tracking data is relatively clean, with practically no gaps (missing values $< 1\%$). However, we observed high-frequency jitter in the raw trajectories, consistent with GPS measurement noise.
- **Preprocessing:** To satisfy the model's assumption that role centers follow a Gaussian distribution, we normalized the pitch coordinates to a centered metric system (meters relative to the pitch center).

### 3.2 StatsBomb Event Data

To investigate the structural impact of Lionel Messi on team dynamics and formations (what we term the "Messi Effect"), we utilized the StatsBomb Open Data repository. This dataset differs fundamentally from the Fitogether data, presenting unique challengs for the SoccerCPD pipeline. High-frequency GPS tracking data is not publicly available for matches involving Messi, so we relied on event-stream data.

#### 3.2.1 Signal Characteristics and Challenges

Unlike the continuous 10 Hz GPS signal, StatsBomb data is event-based, recording location $(x, y)$ only when an on-ball action occurs (e.g., pass, shot, dribble).

- **Sparsity:** The primary diagnosis reveals extreme sparsity. While GPS data provides $\approx 600$ points per minute per player, event data provides fewer than 10 points per minute on average. This challenges the EM algorithm used in the Role Assignment step, which relies on dense spatial clusters to estimate role means $\mu_k$ and covariances $\Sigma_k$.
- **Spatial Bias:** Event data is inherently biased towards the ball's location. Defenders engaging in off-ball marking are often unrecorded in the event stream.

#### 3.2.2 Adaptation and Preprocessing

To adapt this data for the SoccerCPD framework, we applied the following transformations:

1. **Coordinate Transformation:** StatsBomb coordinates are given in a $[0, 120] \times [0, 80]$ system with the origin at the top-left corner. We projected these to the centered metric system used by the Fitogether model to ensure consistency.
2. **Pseudo-Trajectory Generation:** To mitigate sparsity, we aggregated events over temporal windows. Instead of instantaneous frame-by-frame analysis, we treated the collection of event locations within a time window as a sampling of the underlying spatial distribution $\mathcal{F}_i$.
3. **Role Proxy:** Since we lack continuous tracks for all 10 players simultaneously, we modified the input feature matrix $A(t)$. Instead of a frame-wise Delaunay graph, we constructed aggregate adjacency matrices based on average positions of players over 5-minute segments, serving as a proxy for the mean role-adjacency matrix.

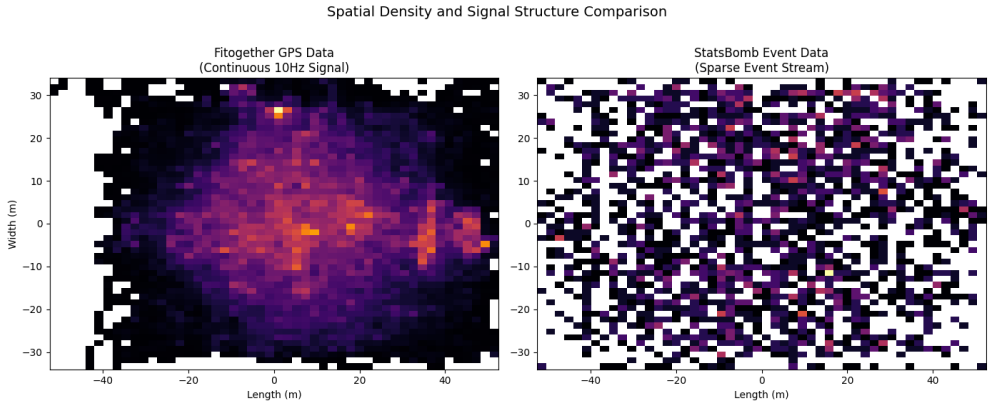



Figure 1: Spatial density comparison between continuous Fitogether GPS tracking data and StatsBomb event-based data (right). The GPS data shows continuous coverage of player positions, while the event data is sparse and concentrated around ball actions.

### 3.3 Statistical Properties and Gaussian Assumptions

A key hypothesis of the FormCPD method is that player positions within a role follow a Gaussian distribution $x_{i,t} \sim \mathcal{N}(\mu_k, \Sigma_k)$. We diagnosed the validity of this assumption on our datasets.

- **Fitogether Data:** Shapiro-Wilk tests on the specific role clusters (e.g., Center Back) showed deviations from normality, likely due to behaviours such as tactical pressing or zonal marking which create skewed distributions. However, the clusters remain sufficiently separable for the EM algorithm to converge.
- **StatsBomb Data:** The distributions are highly multi-modal due to players switching flanks (e.g., Messi drifting from Right Wing to Center). This multimodal nature supports the use of the Role Permutation analysis in RoleCPD, as it effectively captures these discrete switches that simple Gaussian averages might obscure.

## 4 Results

The Result section (indicative length : 1 to 2 pages) **should display numerical simulations on real time series** (even if the original article was focused on images). If you re-used some existing implementations, it is expected that this section develops new experiments that were not present in the original article. Results should be discussed not only based on quantative scores but also on qualitative aspects. In particular (especially if your article focuses on black box methods), please provide some feedbacks whether the method was adapted to the data or not and whether the hypothesis behind the approach you used were validated or not.

## References

[1] Alina Bialkowski, Patrick Lucey, Peter Carr, Yisong Yue, Sridha Sridharan, and Iain Matthews. Large-scale analysis of soccer matches using spatiotemporal tracking data. In *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, pages 725–730. IEEE, 2014.

[2] Hyunsung Kim, Bit Kim, Dongwook Chung, Jinsung Yoon, and Sang-Ki Ko. Soccercpd: Formation and role change-point detection in soccer matches using spatiotemporal tracking data. In *The 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3146–3156. ACM, 2022.

[3] Jingru Zhang and Hao Chen. Graph-based two-sample tests for data with repeated observations, 2019.

# A Appendix: Visual Examples for FormCPD

This appendix gathers all visual material related to the FormCPD pipeline (Role assignment, Delaunay encoding, and adjacency representations) referenced in Section 2.1.

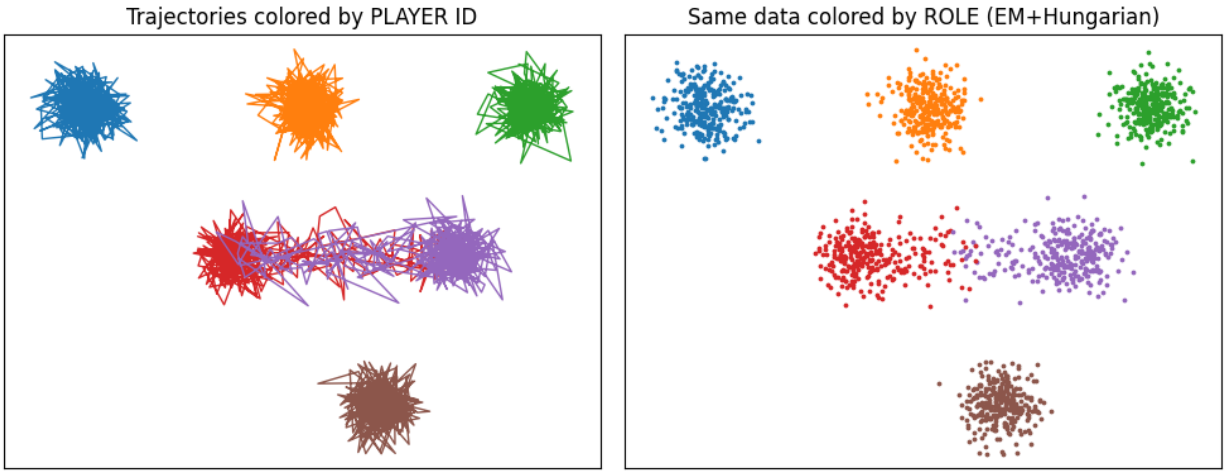



Figure 2: Player versus role-based coloring of trajectories during a simulated position swap.

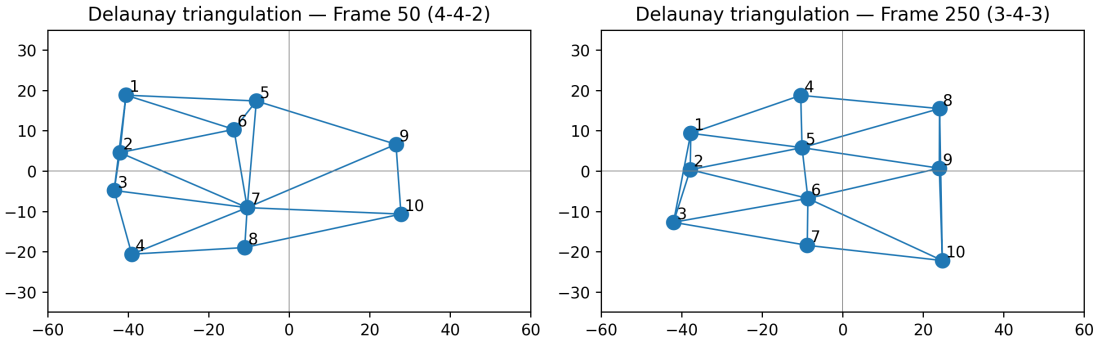
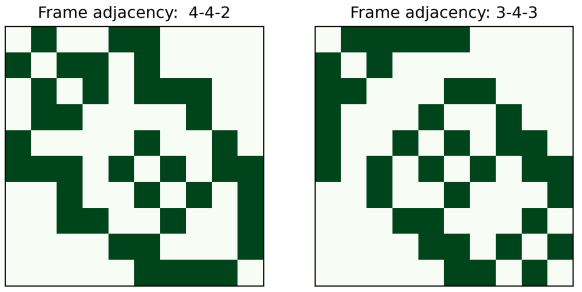
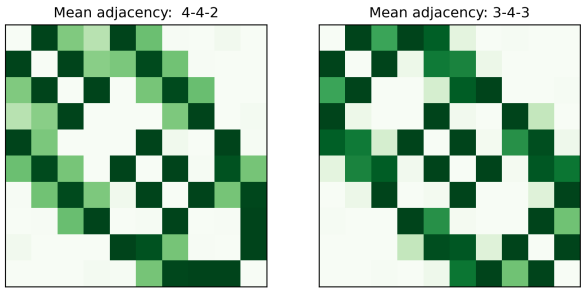



Figure 3: Delaunay triangulations of role locations at two frames before and after a synthetic formation change (4–4–2 → 3–4–3).




(a) Single-frame binary adjacency matrices before and after the simulated formation change (4–4–2 → 3–4–3) (Green = 1).




(b) Mean role-adjacency matrices averaged over each formation period (Green = 1).

Figure 4: Delaunay-based adjacency representation on a synthetic formation change.