UNIVERSITÀ DI PISA

A Fuzzy Density-based Clustering Algorithm for Streaming Data

Andrea Aliperti[1], Alessio Bechini[1], Francesco Marcelloni[1], Alessandro Renda[1,2]

[1]University of Pisa, Dept. of Information Engineering

[2]University of Florence, Dept. of Information Engineering

Outline

- Motivation and goals
- Related works and baselines
- Proposed algorithm: **SF-DBSCAN**
- Experimental Results
- Conclusion

Importance of Mining Data Stream

Every minute [1]

- approximately 500.000 tweets are sent
- more than 4.000.000 query searches on Google are performed

Huge amount of **data streams** are generated at *very high speed* by several applications:

- Social Networks
- Sensor Networks
- Stock Market
- … and many others

1. https://www.internetlivestats.com/one-second/,

Main challenges in clustering data streams

A *stream* P is an *ordered sequence of data objects*

$$\mathrm{P} = \{\boldsymbol{p}_1,\ \boldsymbol{p}_2, \dots ,\ \boldsymbol{p}_N\}$$

where each object $\boldsymbol{p}_i$ is described as an n-dimensional feature vector

- **Potentially unbounded** sequence of objects
- Characteristics may evolve over time due to **concept drift**
- **Number of clusters may change** over time

Motivation and Goal

Desirable properties of streaming clustering algorithms

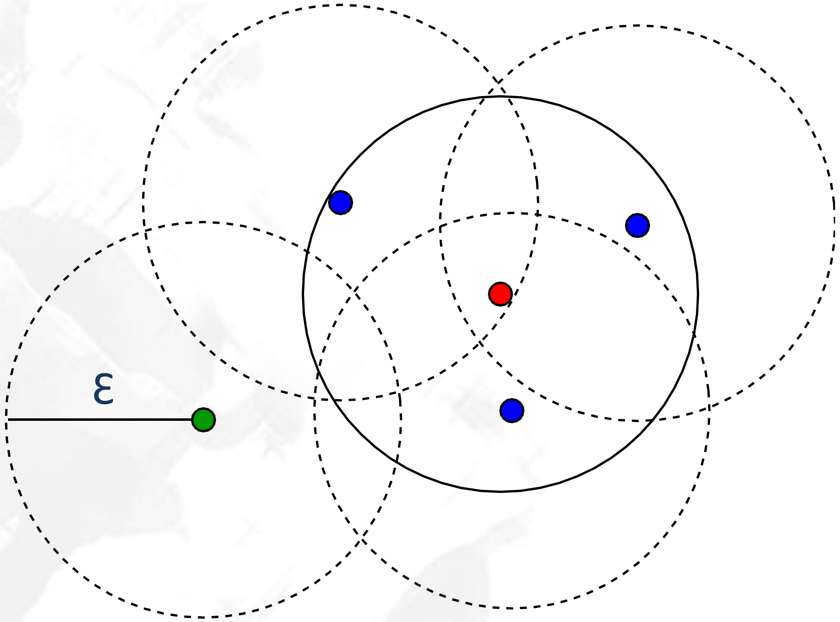
- Effectiveness in dealing with **concept drift**
- Dealing with a **number of clusters** which may **change over time**
- Handling potentially **unbounded sequence** of objects
- Detection of **arbitrary shaped** clusters
- Partitioning data without prior knowledge of **number of clusters**
- Ability to handle **noise**
- Reduced **sensitivity to input parameters**

SF-DBSCAN: A fuzzy extension of DBScan Clustering Algorithm for Streaming Data

DBSCAN:

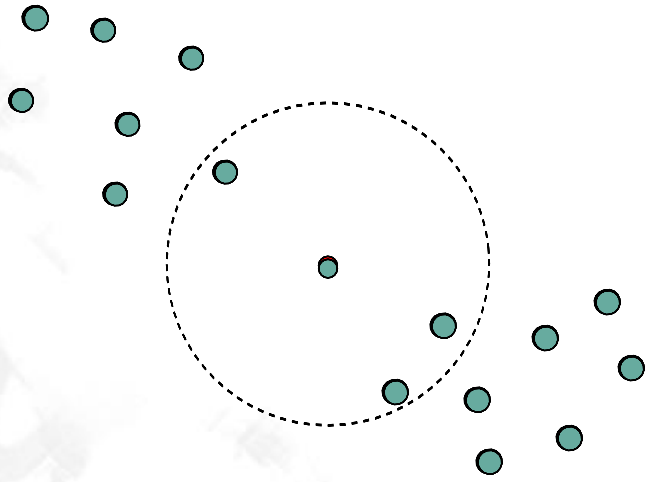
- Requires the definition of two parameters:
  - ***Ɛ***: defines the *neighborhood* size
  - ***MinPts****:* number of points required for a core
- Partitions data into **connected *dense* regions** separated by ***sparse* regions**
  - Distinction between Core, Border, Noise objects
- Drawbacks:
  - **High sensitivity to input parameters**
  - Developed for **static dataset**

Streaming Proposal: S-DBSCAN



S-DBSCAN: Streaming DBScan

- **Update** the partition at each new object
  - Key idea: check the status of the new object and all the objects in its neighborhood

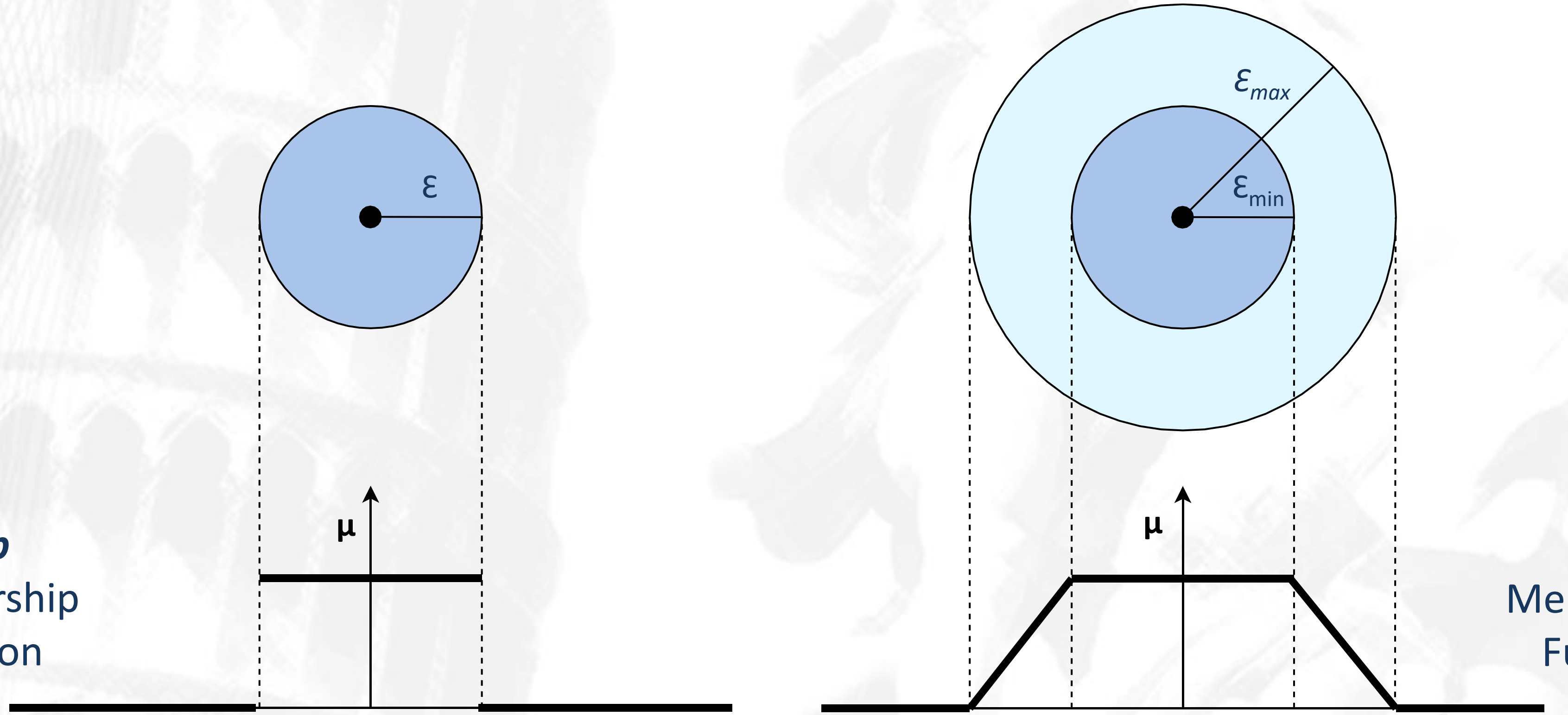


- **Assumptions:**
  - Deal with a **bounded** sequence of objects
    - No memory constraints, no need for fading mechanism

- **Goal:** define a crisp baseline for the streaming setting

SF-DBSCAN: Streaming Fuzzy DBSCAN

- Basic idea: fuzzy membership function
  - Previously adopted only in *static* implementations

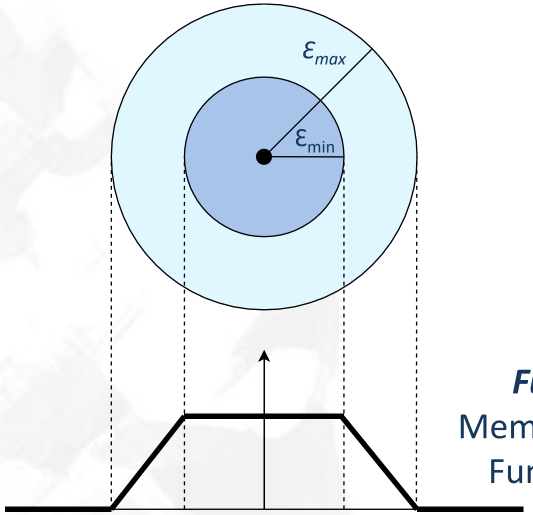


Crisp
Membership
Function

Fuzzy
Membership
Function

SF-DBSCAN: Streaming Fuzzy DBSCAN

- *As in the original DBSCAN implementation:*
  - Only objects within ε_{min} are considered for the election of ***core objects***
- *Differently from original DBSCAN implementation:*
  - a ***border object*** belongs to a cluster with a membership degree that can be lower than 1, depending on its distance from the **closest core object**
  - One object can belong to the border of multiple clusters

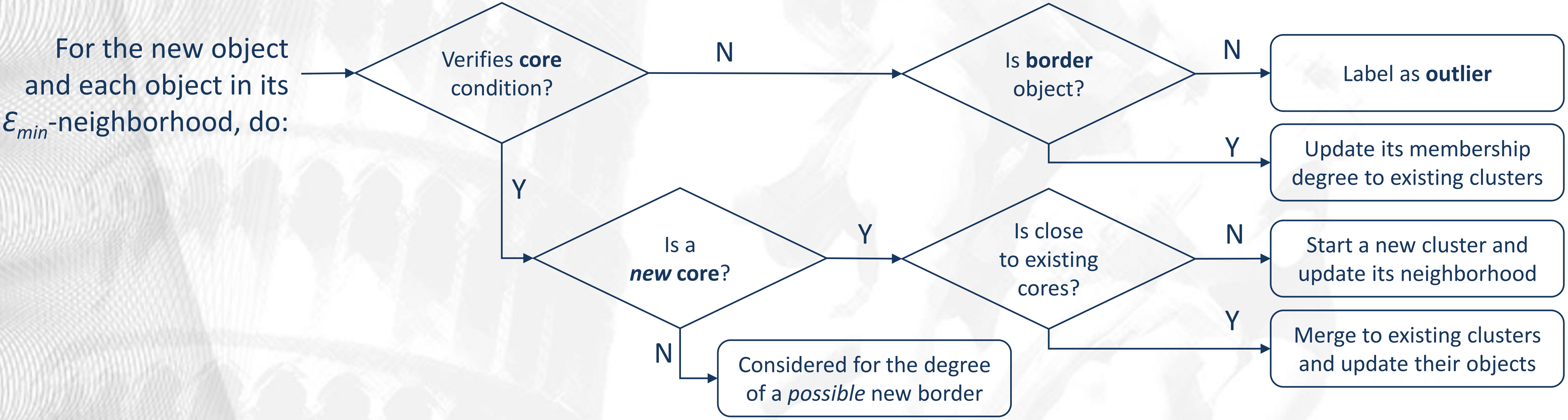


Fuzzy Membership Function

- SF-DBSCAN can discover **clusters with *fuzzy overlapping borders***

SF-DBSCAN: Streaming Fuzzy DBSCAN

- Parameters initialization: $\mathcal{E}_{min}$, $\mathcal{E}_{max}$, *MinPts*
- Definition of Data Structures:
  - List of already consumed objects
  - Membership degree of border objects to each cluster



Experimental Evaluation

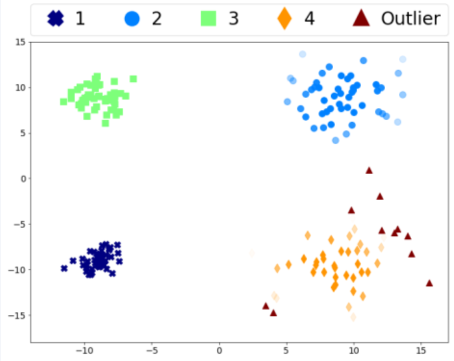
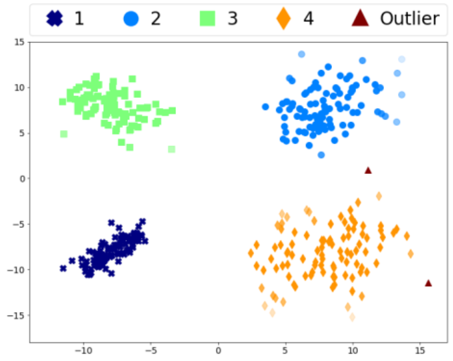
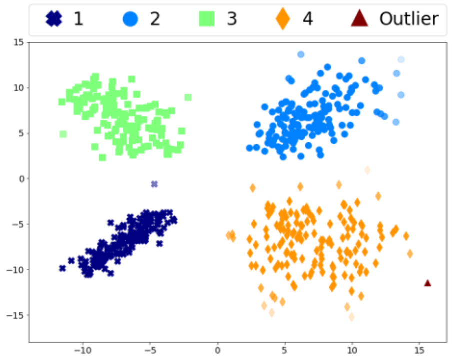
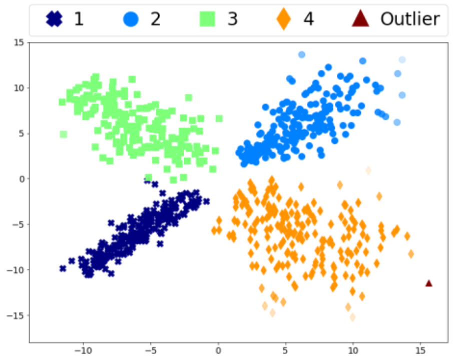
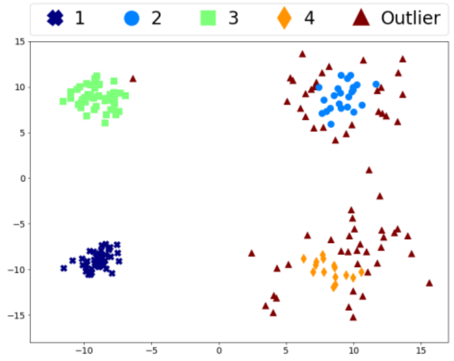
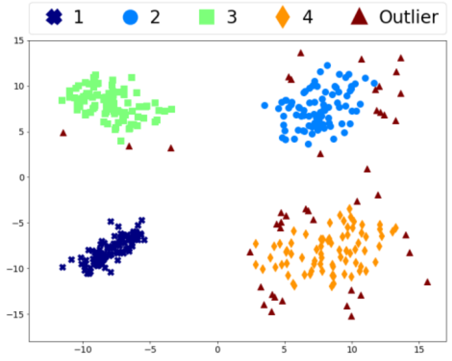
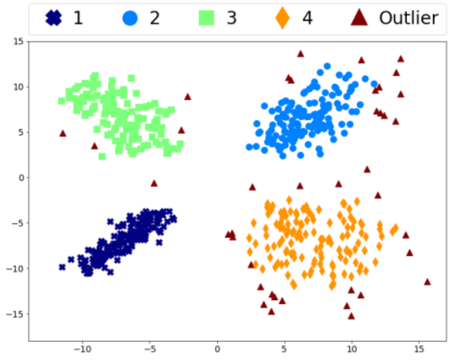
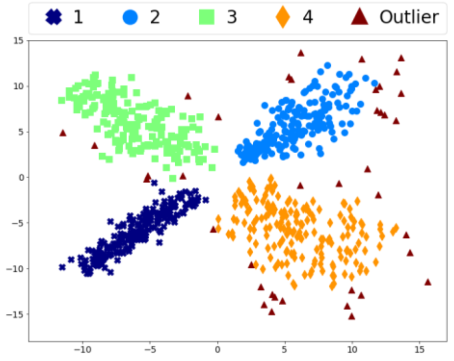
- Datasets:
  - Synthetic datasets from GaussianMotionData[1] collection
    - Gaussian distributions with concept drift
    - Selection of **2D datasets** with a **limited number of objects**

- Accuracy Evaluation of **SF-DBSCAN compared to S-DBSCAN** in terms of:
  - Visual Analysis
  - Adjusted Rand Index

1. Márquez, David G., et al. "**A novel and simple strategy for evolving prototype based clustering.**" Pattern Recognition 82 (2018): 16-30.

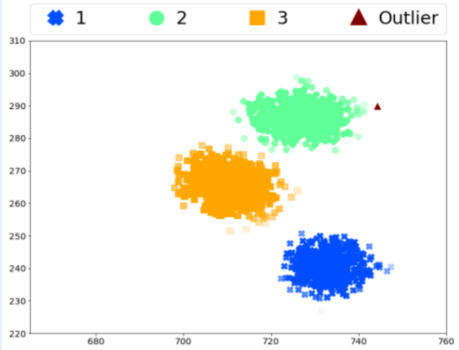
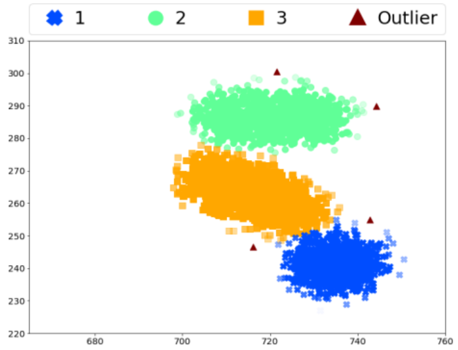
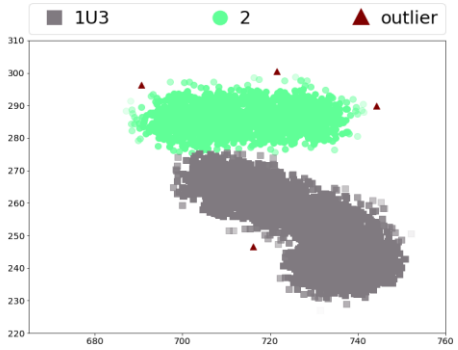
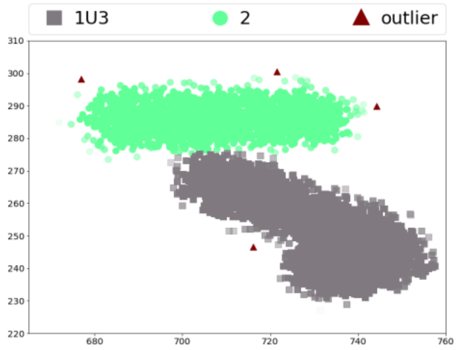
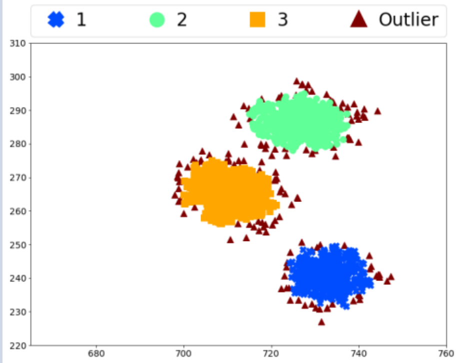
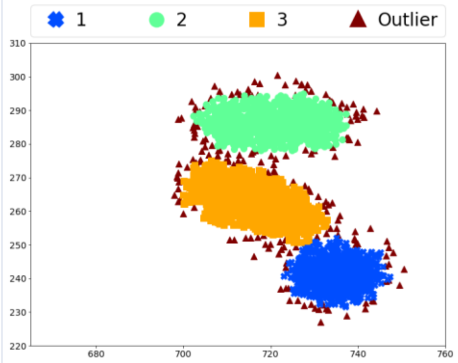
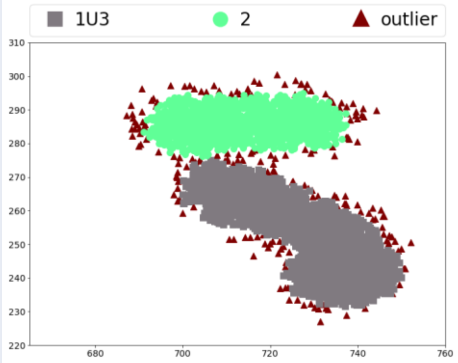
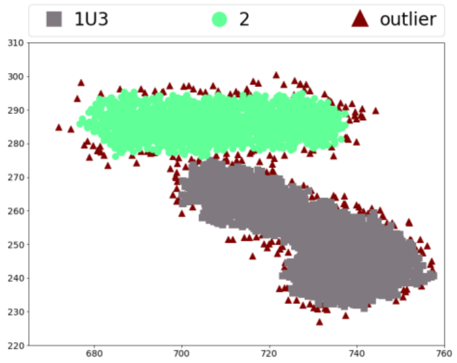
Experimental Evaluation: SF-DBSCAN vs S-DBSCAN

Dataset **GMD-4C2D800Linear** : 4 Clusters - 2D – 800 objects

	Percentage of analyzed objects			
	25%	**50%**	**75%**	**100%**
SF-DBSCAN *MinPts*, ε_{min}, ε_{max}				
S-DBSCAN *MinPts*, ε_{min}				

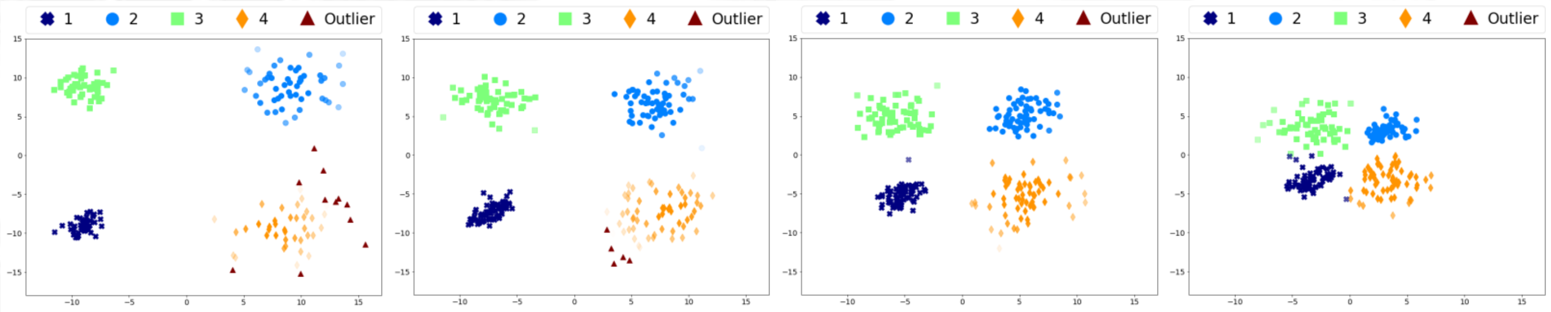
Experimental Evaluation: SF-DBSCAN vs S-DBSCAN

Dataset **3C2D7500Merge**: 3Clusters - 2D - 7500 objects

	Percentage of analyzed objects			
	25%	**50%**	**75%**	**100%**
SF-DBSCAN *MinPts*, ε_{min}, ε_{max}				
S-DBSCAN *MinPts*, ε_{min}				

Conclusions

- Proposal of **SF-DBSCAN:** a new **Streaming Fuzzy** extension of **DBSCAN**
  - Captures fuzzy clusters with overlapping borders
  - Effective in dealing with concept drift
  - Outperforms crisp baseline on benchmark datasets
- Future developments
  - Further extension to deal with **unbounded sequences**
    - Adoption of a *forgetting* mechanism: damped window model



Thank you for your attention

Alessandro Renda

alessandro.renda@unifi.it

Ph.D. Student – Smart Computing

University of Pisa, Dept. Information Engineering

Computational Intelligence Group