Visualising the Cluster Structure of Data Streams

Dimitris K. Tasoulis[1] Gordon Ross[2] Niall M. Adams[2]

[1]Institute for Mathematical Sciences, [2]Department of Mathematics
Imperial College London, South Kensington Campus
London SW7 2PG, United Kingdom
{d.tasoulis,gordon.ross,n.adams}@imperial.ac.uk

Imperial College
London

Evolving Data Streams

Data Nature

The data of interest comprises multiple sequences that **evolve** over time.

- Algorithms must have the capacity to adapt rapidly to changing dynamics of the sequences.
- The results of analysis are useful if they are available immediately.
- Scalability in the number of sequences is becoming increasingly desirable.

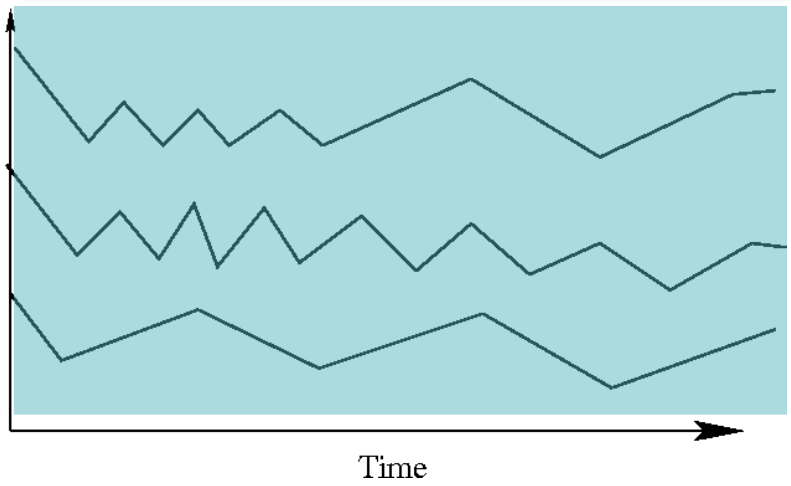
Evolving Data Streams

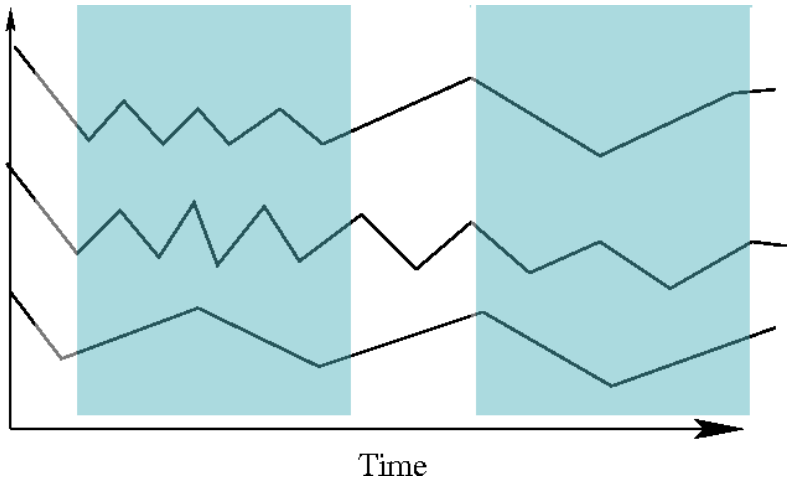
Time series Analysis

Inference using the complete information

Data Streams

Inference using only partial information





- It is not possible to store the complete information.
- We are only interested in information relevant to the current time.

Evolving Data Streams

In this work we assume that:

- Data are arriving sequentially in time from some mixture distribution.
- The mixture components gradually change over time.
- Components may vanish, and new ones can appear.

A forgetting procedure is usually employed that attaches decreasing weight to historical data, so as to gradually diminish their effect.

Clustering and Density Estimation

Definition

Clustering refers to the partitioning of a set of objects into groups (clusters) such that objects within the same group are more similar to each other than objects in different groups.

The data space can be regarded as the empirical probability density function (pdf) of the data. In this sense *local maxima of the pdf can be thought to correspond to centres of clusters.*

- References to clustering date back to the antiquity but one of the first comprehensive foundations of clustering methods was published by Tryon in 1939 (R.C. Tryon, Cluster Analysis, Ann Arbor, MI, Edward Brothers, 1939).

Density Estimation

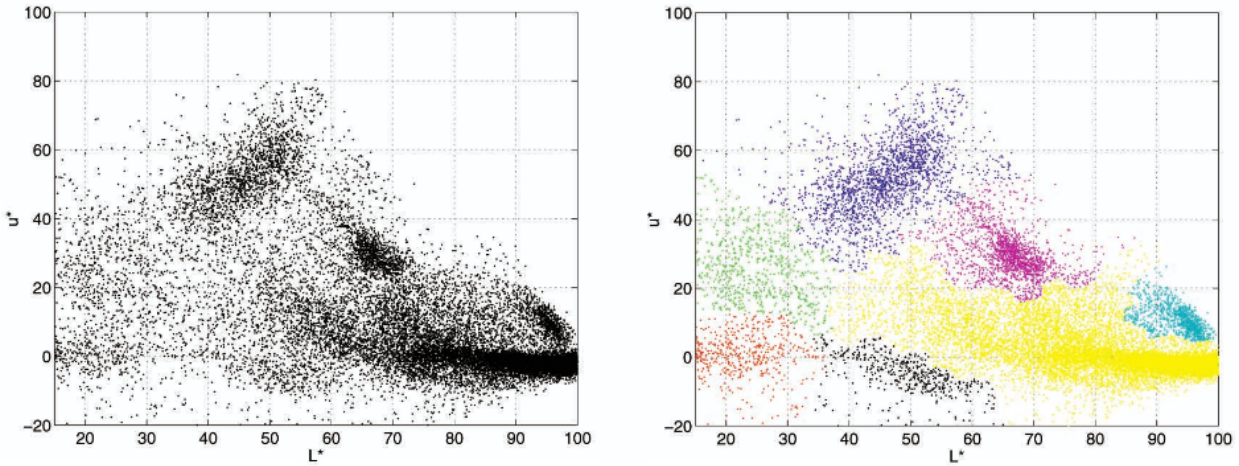
The rationale behind density estimation is that the data space can be regarded as the empirical probability density function (pdf) of the data.
Let a data set $X = \{x_1, \ldots, x_n\}$ The multivariate kernel density estimate $\hat{f}_H(x)$, is computed at point x as:

$$\hat{f}_H(x) = C_{f,H} \frac{1}{n} \sum_{i=1}^{n} K_H \left(H^{-1}(x - x_i) \right), \tag{1}$$

- H is the *bandwidth matrix*.
- $K_H() : \mathbb{R}^d \mapsto \mathbb{R}$
- $C_{f,H}$ is a normalization constant dependent on the kernel function, the dimension of the data.
- $\hat{f}_H(x) \propto f(x)$.

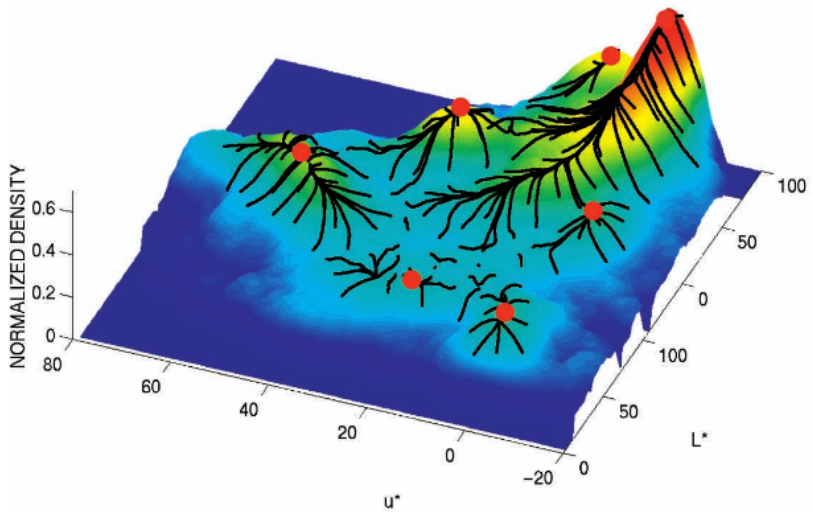
Clustering and Density Estimation

Mean Shift is one of the most successful density clustering methods: Each point is *mean shifted* towards the local gradient estimate of the density function:



(a)

(b)



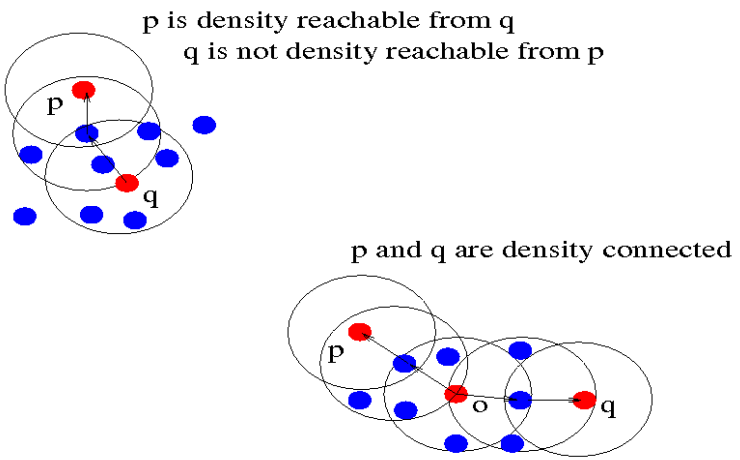
(c)

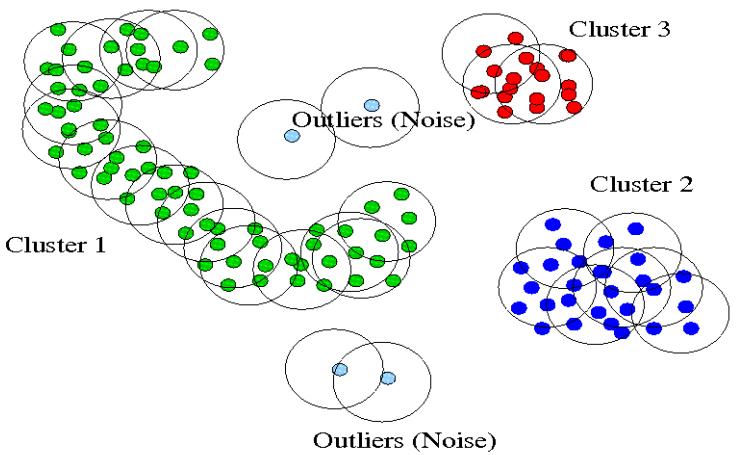
Density Based Clustering Examining Neighbours

core points

In a neighbourhood of a given radius (*Eps*) for each point in a cluster at least a minimum number of objects (*MinPts*) should be contained.

- Each point in their neighbourhood is considered as “Directly Density-Reachable”.
- Chain of“Directly Density-Reachable” points form clusters.





The Micro-Clustering Framework

A micro-cluster is defined by the quantities w, c, r, which try to summarize the information about the data density on a particular area.

Definition

(core-micro-cluster) A micro-cluster $MC_t(w, c, r)$ is defined as core-micro-cluster $CMC_t(w, c, r)$ at time t for a group of streaming points $\{x_i, t_i\}$ $i = 1, \ldots, n$, and parameters ϵ, μ it when $w \geqslant \mu$ and $r \geqslant \epsilon$. Where $w = \sum_{i=1}^{n} T_\lambda(t_i)$ is the micro-cluster's weight, $c = \frac{\sum_{i=1}^{n} x_i T_\lambda(t_i)}{w}$, is its center and r its radius $r = \sum_{i=1}^{n} \frac{T_\lambda(t_i)\|c - x_i\|}{w}$.

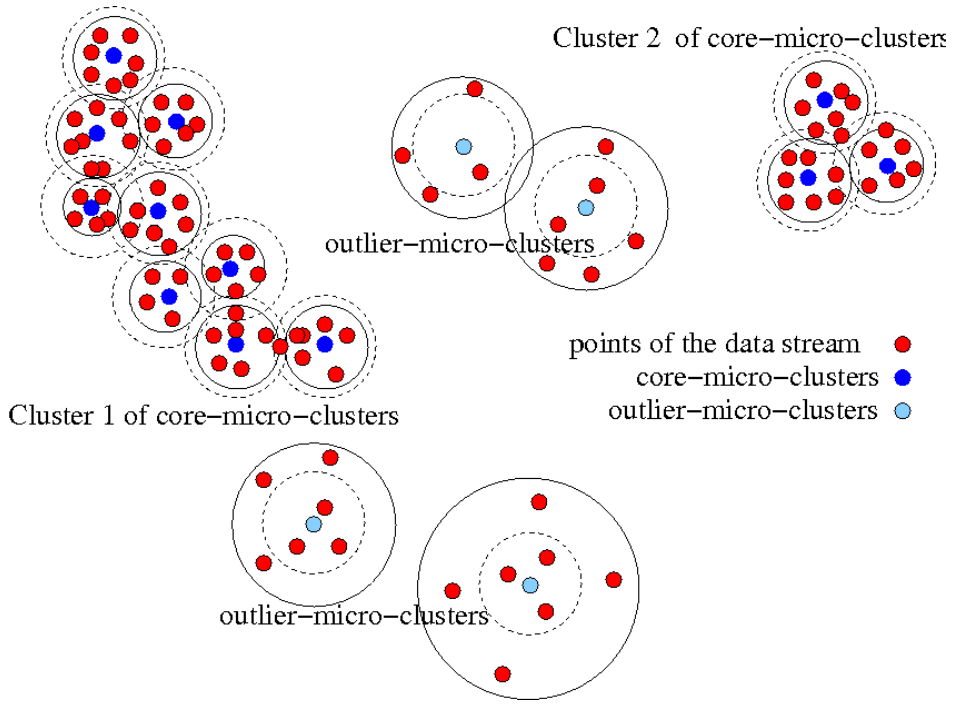
Two types of micro-clusters

- *Potential core-micro-clusters*, when $w_c \leqslant \beta\mu$ and $r \leqslant \epsilon$.
- *Outlier-micro-clusters*, when $w_c > \beta\mu$ and $r > \epsilon$.

The DenStream Algorithm

Procedure **ListMaintain**

1. Initialize two lists *PL*, *OL*; one for the core-micro-clusters, and the other for the outlier-micro-clusters.
2. Each time a new point $p = \{x, t\}$ arrives do one of the following:
   1. Attempt to merge p into its nearest core-micro-cluster c_p: If the resultant micro-cluster has a radius $r > \epsilon$, then the merge is omitted.
   2. Attempt is made to merge p into its nearest outlier-micro-cluster o_p: if $r > \epsilon$, the merge is omitted. Otherwise, if the subsequent weight w of o_p exceeds μ, then move o_p to *PL*.
   3. A new outlier-micro-cluster is created, centered at p.
3. Periodically prune from *PL* and *OL*, the micro-clusters for which $w_c \leqslant \beta\mu$, and $w_c \leqslant \xi$ respectively.

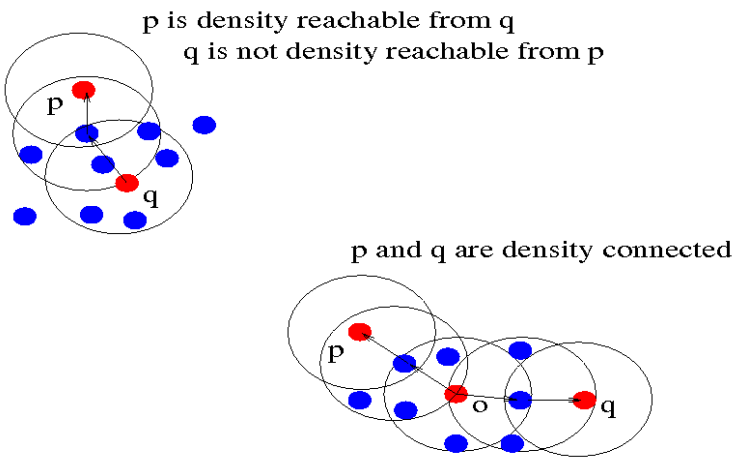


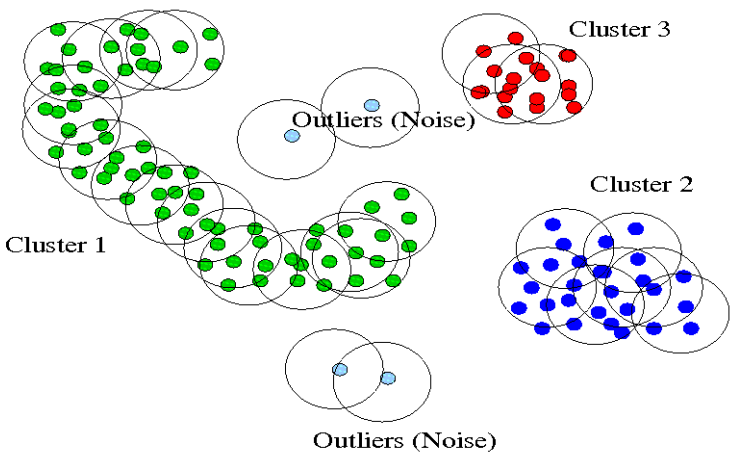
Density Based Clustering Examining Neighbours

core points

In a neighbourhood of a given radius (*Eps*) for each point in a cluster at least a minimum number of objects (*MinPts*) should be contained.

- Each point in their neighbourhood is considered as “Directly Density-Reachable”.
- Chain of“Directly Density-Reachable” points form clusters.





Visualizing Clusters in Static Datasets

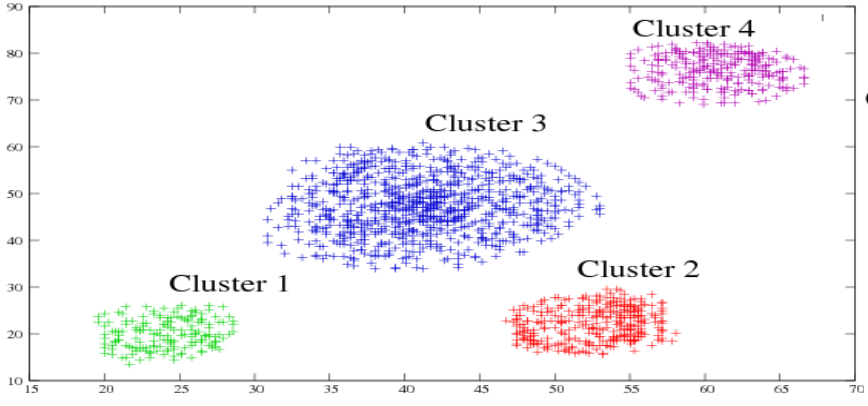
"core-level"

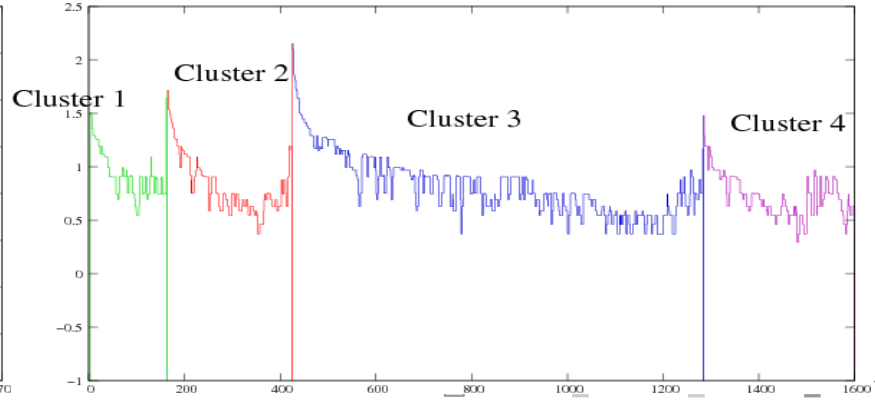
The distance from p to its *MinPts*-nearest neighbour.

"reachability-distance"

For two objects p, q in the database the reachability of p wrt. q is defined as $RDist(p, q) = \max\{$ core level of p, $dist(p, q)\}$.

- Each object is positioned so that all objects before it have the minimum reachability distance to it.
- The cluster-ordering of a data set can be represented and understood graphically.





Stream Cluster Visualization

A time changing mapping of the clustering structure to a user understandable format, operating in a real-time environment.

Micro-cluster neighbourhood

Let $\epsilon \in \mathbb{R}$, be a user defined parameter and *PL* a potential core-micro-cluster list. Then for a potential core-micro-cluster c_p, we define the micro-cluster neighbourhood of c_p, as

$$N(c_p) = \{c_q \in OL | dist(c_p, c_q) \leqslant 3.0\epsilon\}.$$

The function $dist(c_p, c_q)$, returns the euclidean distance between the centers of c_p and c_q.

Micro-cluster core-level

Let $\epsilon \in \mathbb{R}, \beta \in \mathbb{R}, \mu \in \mathbb{N}$. The core-level of c_p, $CLev(c_p)$ is defined as:

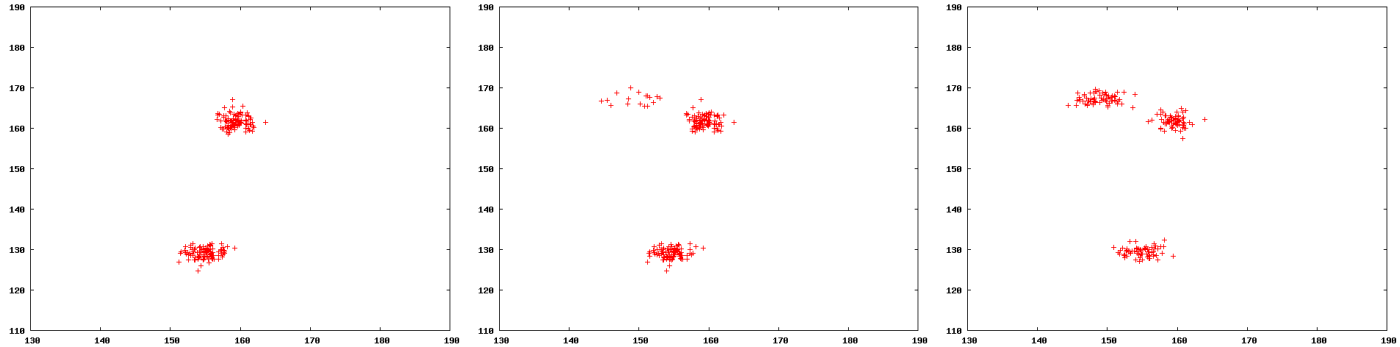
$$CLev(c_p) = \text{radius} \quad \text{of} \quad c_p$$

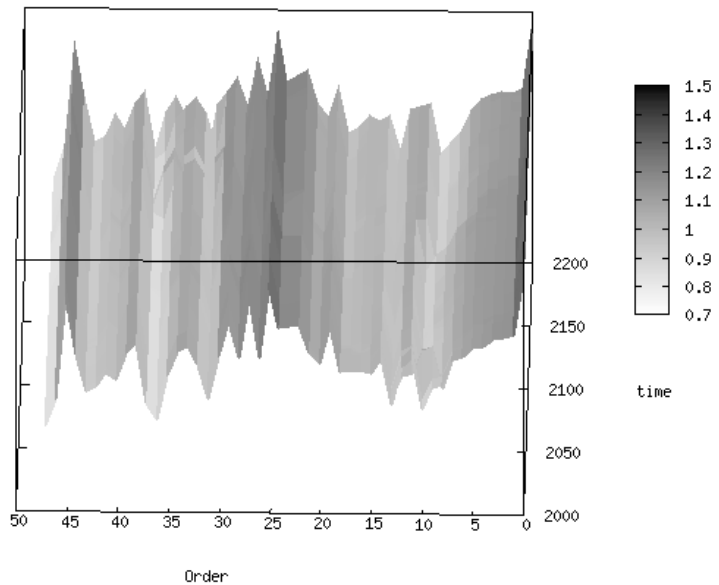
Stream Cluster Visualization: StreamOptics

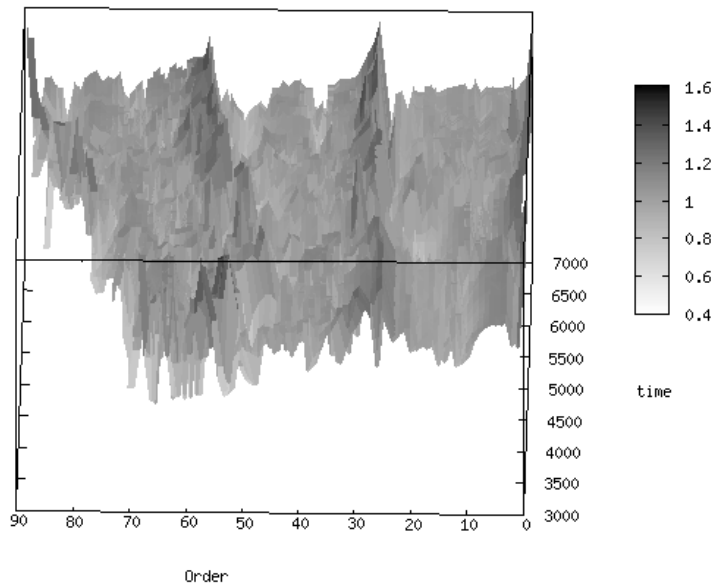
Procedure **StreamOptics**

1. While there is still a micro-cluster c_p in PL that has a neighbourhood size $|N(c_p)|$ larger than one initialize a list S of all the micro-clusters in $|N(c_p)|$.
2. Remove c_p from PL and added to OL.
3. Remove all micro-clusters in $|N(c_p)|$ from PL.
4. For each c_l in S, compute $RDist(c_l, c_p)$.
5. For each c_l in S, insert to S all the micro-clusters in $N(c_l)$.
6. Remove form PL all the micro-clusters in S.
7. Insert to OL the object with the smallest $RDist(c_l, c_p)$ from S, until S is empty.

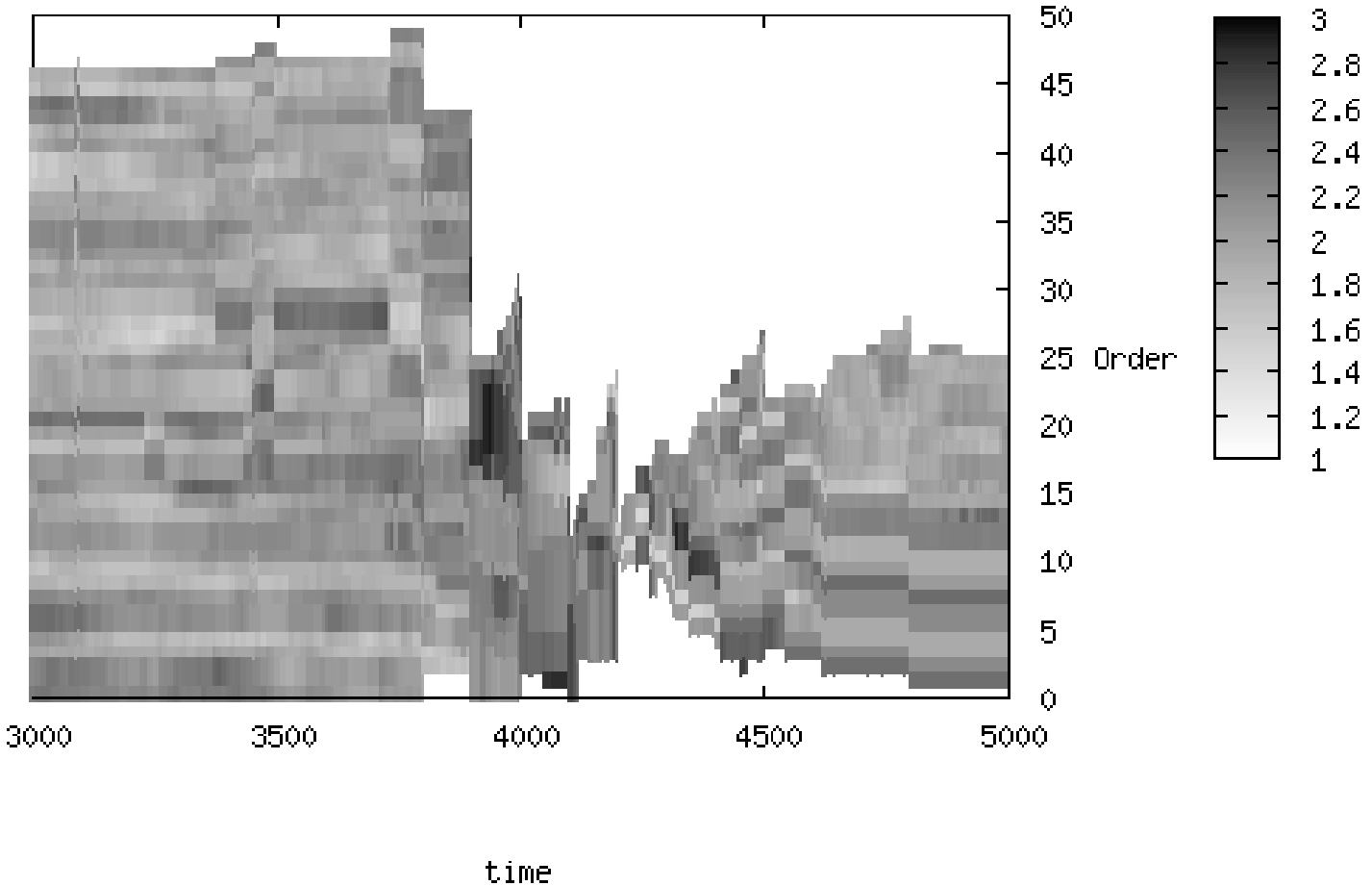
StreamOptics: Spawning Clusters



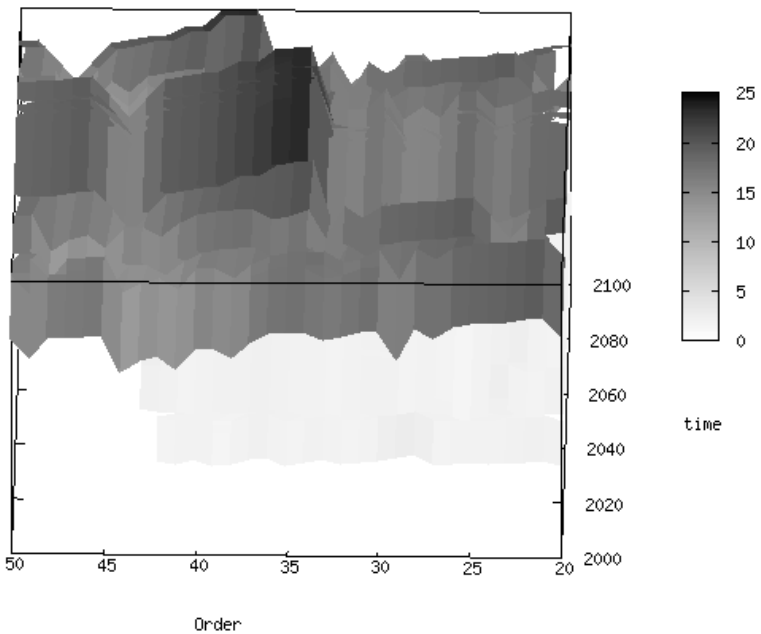


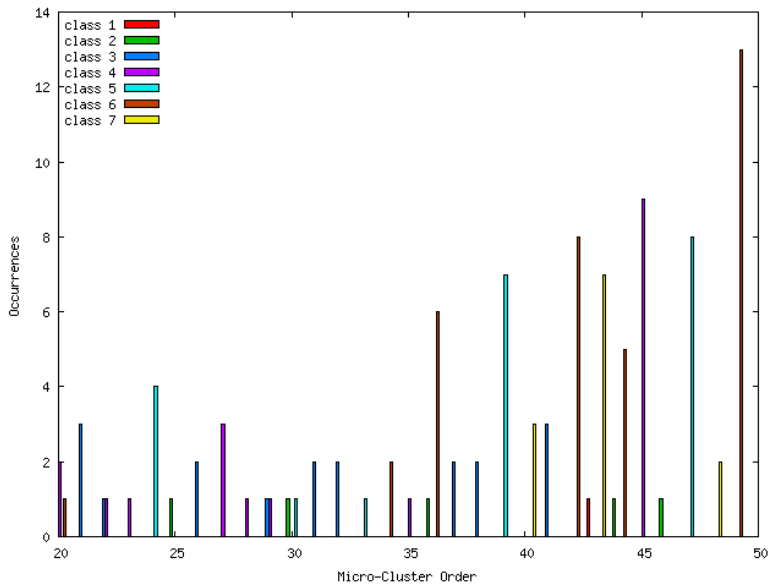


StreamOptics: Disappearing Clusters

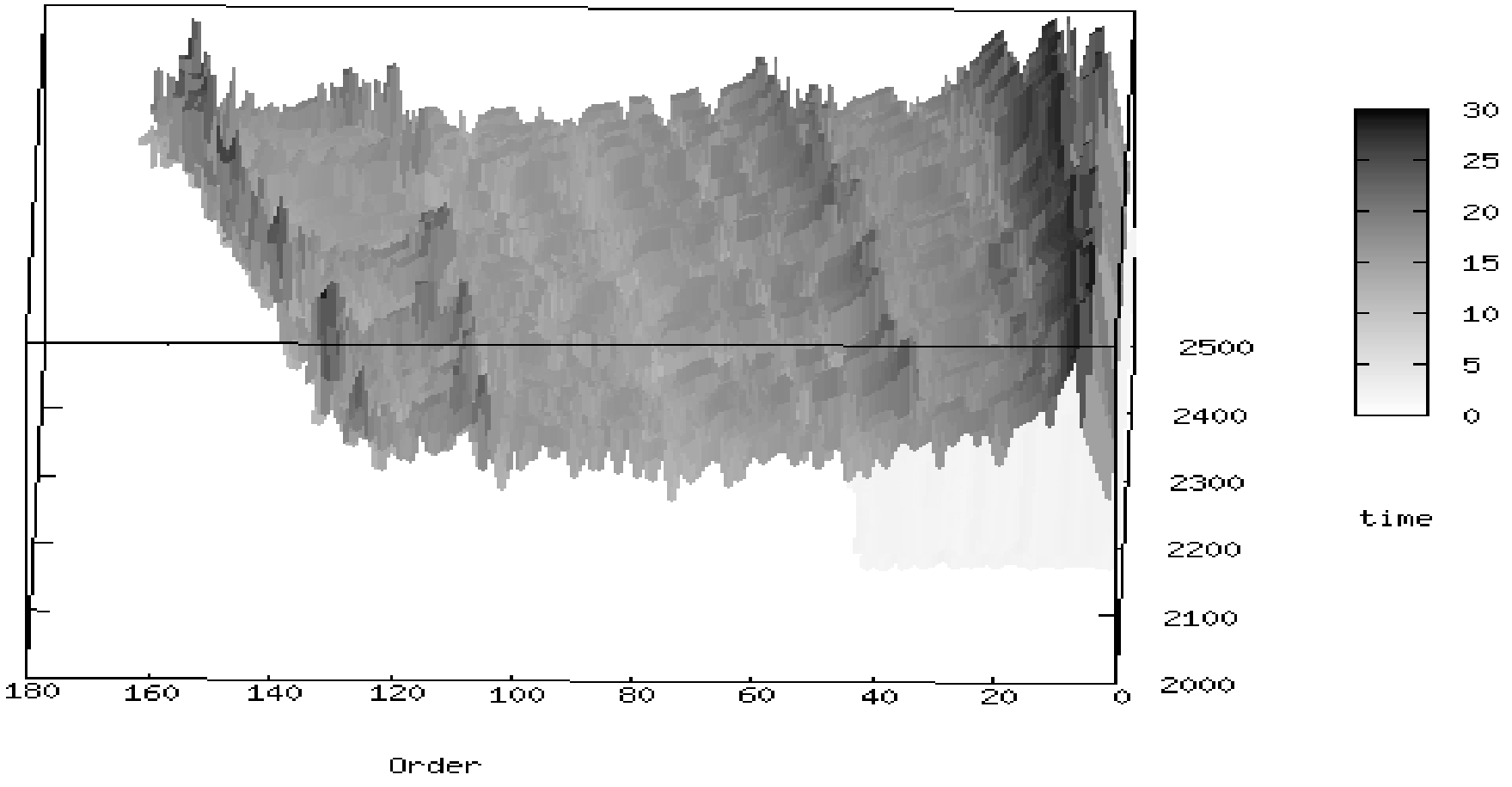


StreamOptics: The Forest CoverType data set





StreamOptics: The Forest CoverType data set



Concluding Remarks

- Methods that can visualise the change of the clustering structure through time have only been investigated in lower dimensional situations or via projection.
- We hybridise a stream clustering framework with an extension of OPTICS.

- The method aims to provide insight into both the clustering structure and its evolution in time.
- We can identify the change in cluster structure, (spawning clusters, fading clusters).
- The abilities of the method are demonstrated in a real world dataset.
- Incorporating other ideas, such as projected clustering, to deal with very high dimensional spaces.

.....