MOA: Massive Online Analysis, a Real-time Analytics Open Source Framework

Albert Bifet[1], Geoff Holmes[1], Bernhard Pfahringer[1], Jesse Read[1]
Philipp Kranen[2], Hardy Kremer[2], Timm Jansen[2] and Thomas Seidl[2]

[1]University of Waikato, New Zealand
[2]RWTH Aachen University, Germany

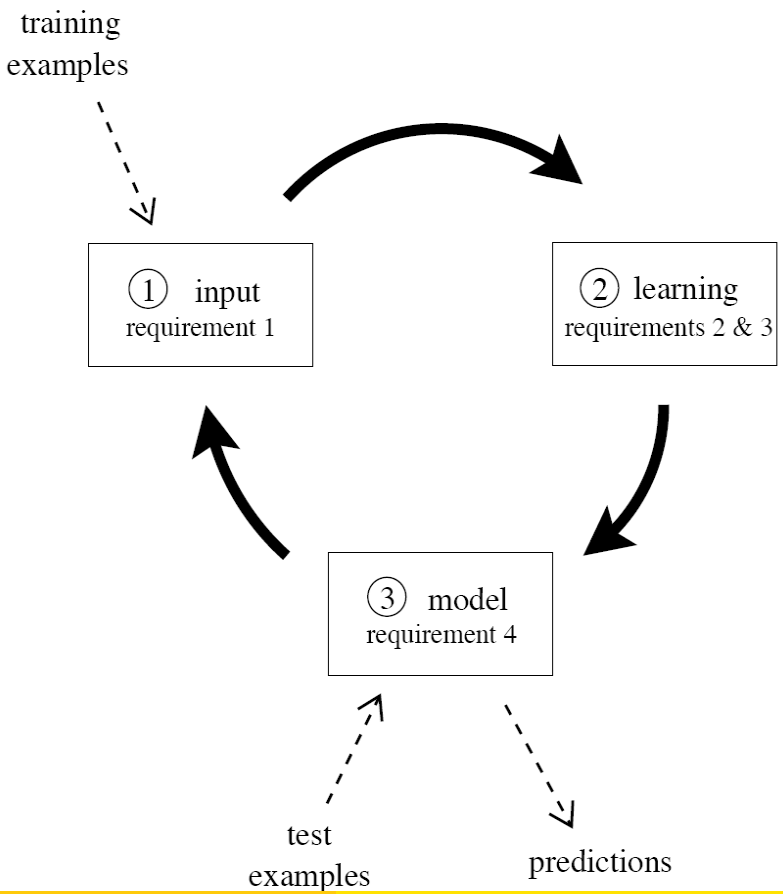
September 8, 2011

Why MOA?

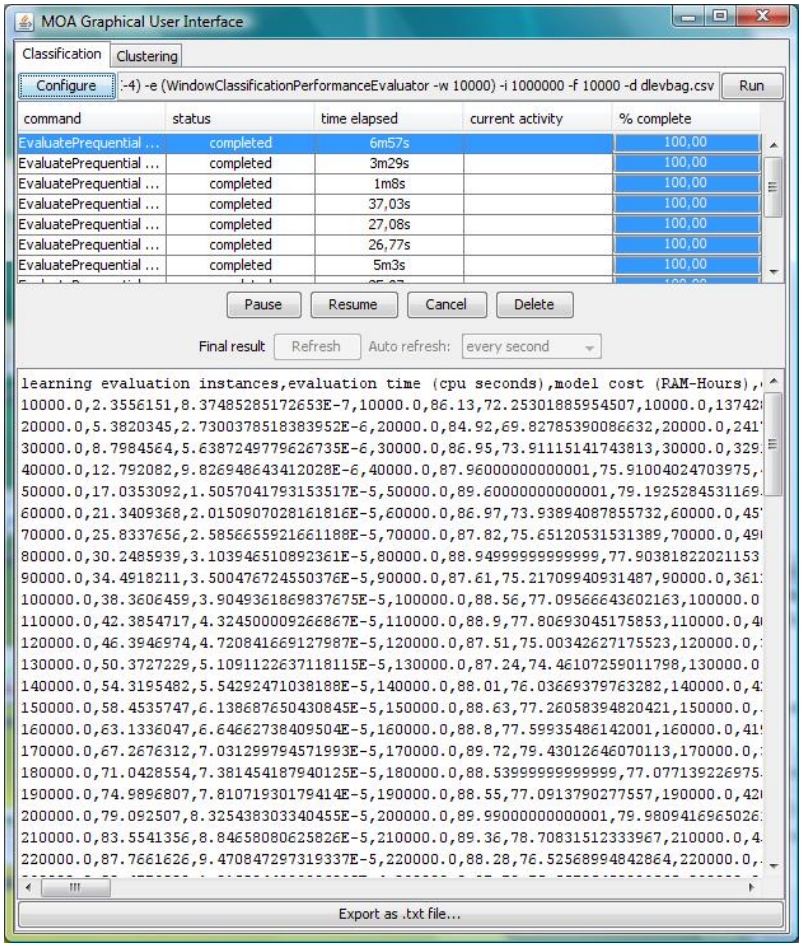
- {M}assive {O}nline {A}nalysis is a framework for online learning from data streams.
- Data Streams
  - Sequence is potentially infinite
  - High amount of data: (almost) constant space
  - High speed of arrival: (almost) constant time per example
  - Once an element from a data stream has been processed, it is discarded or archived

Data stream learning cycle

1. Process an example at a time, and inspect it only once (at most)
2. Use a limited amount of memory
3. Work in a limited amount of time
4. Be ready to predict at any point



Classification



Classifiers: Hoeffding Decision Trees, Hoeffding Option Trees, Bagging, Boosting, Naive Bayes, Perceptrons.

Multi-Label Classification

Examples can be associated with *multiple* labels

- multi-label stream generators
- several state of the art methods
  - ECC Ensembles of classifier-chains
  - EPS Ensembles of Pruning Sets
  - Multi-label Hoeffding Trees
  - Multi-label adaptive bagging methods.

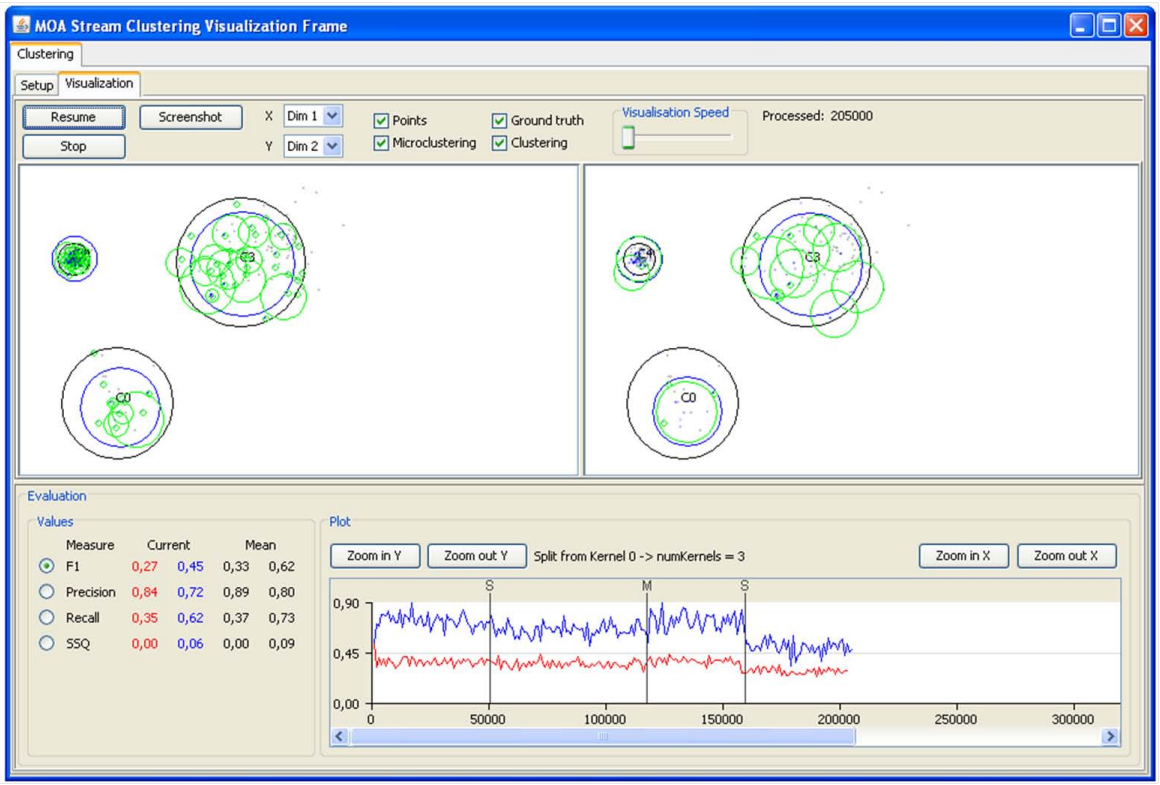
New Evolving Data Stream Generators

- Random RBF with Drift
- LED with Drift
- Waveform with Drift
- Hyperplane
- SEA Generator
- STAGGER Generator

Plus

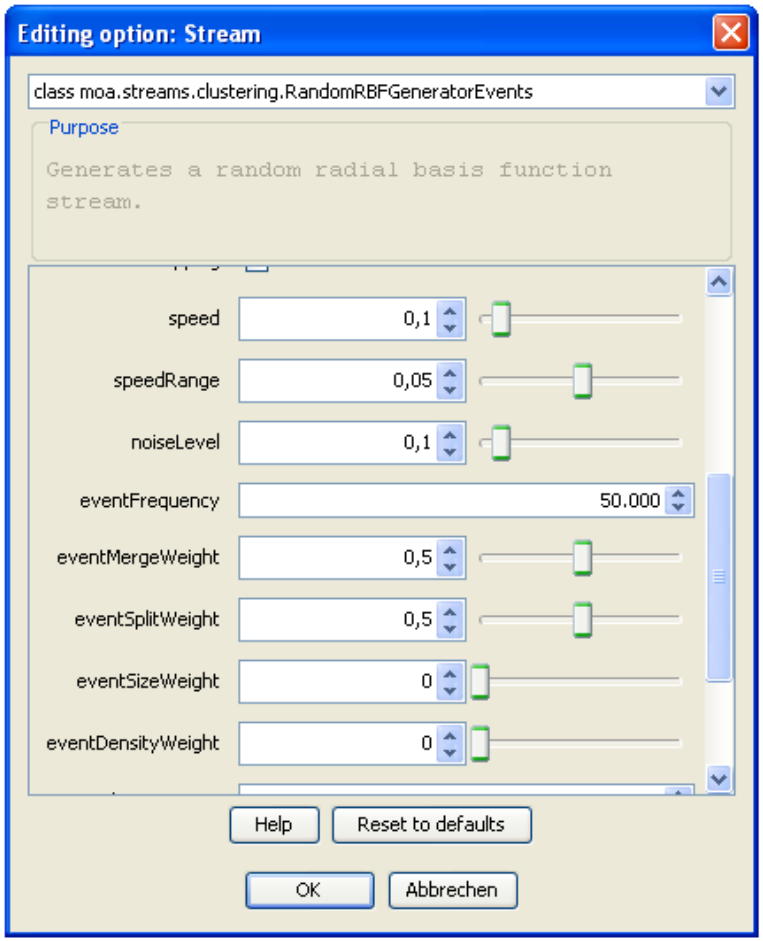
- Sigmoidal shifts from one generator to the next
- Twitter access

Clustering



Evaluation Measures: Rand statistic, Precision, Recall, F1, van Dongen criterion , Redundancy, Compactness, Overlap, MSE, Silhouette Coefficient, Variation of Information, V-Measure, Completeness, Homogeneity, GT cross entropy, FC cross entropy, CMM.

More on Clustering



Clusterers:

- StreamKM++
- CluStream
- ClusTree
- Den-Stream
- D-Stream
- CobWeb

Frequent Closed Subgraph Mining

- Methods for mining frequent closed subgraphs
  - Incremental: IncGraphMiner
  - Sliding Window: WinGraphMiner
  - Adaptive: AdaGraphMiner using `ADWIN` to monitor change
- Approach based on coresets and relative support

MOA Implementation

http://moa.cs.waikato.ac.nz

- MOA is implemented in pure JAVA, inter-operates with Weka
- MOA is easy to use and extend
- MOA is under active development: regression, kernels, ...
- MOA : give it a go :-)