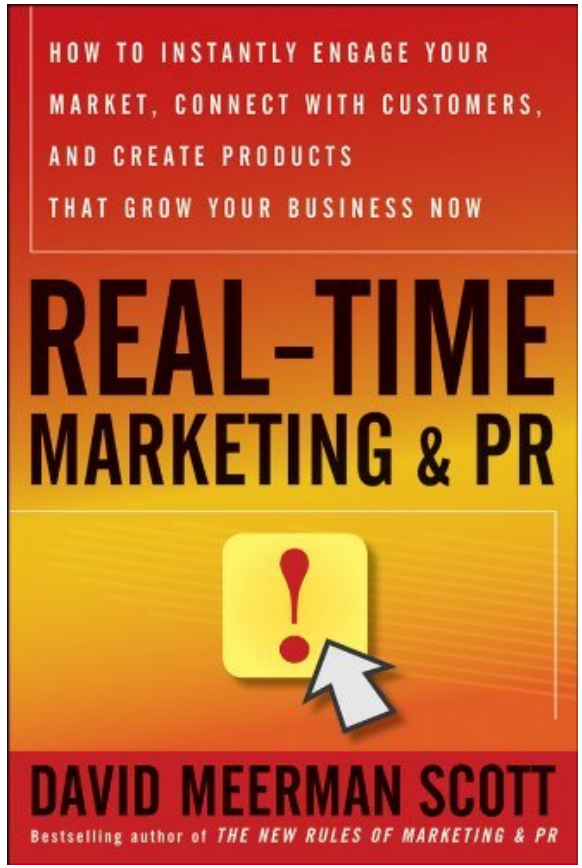
Real-Time Big Data Stream Analytics

Albert Bifet @abifet

Business Applications of Social Network Analysis (BASNA)
Shenzhen, 14 December 2014

Real time analytics

Real time analytics



Companies need to know what is happening right **now** to **react** and **anticipate** to detect new business opportunities through **Social Networks** and **Real-time Analytics**.

Outline

Big Data Stream Analytics

MOA: Massive Online Analysis

SAMOA: Scalable Advanced Massive Online Analysis

Outline

Big Data Stream Analytics

MOA: Massive Online Analysis

SAMOA: Scalable Advanced Massive Online Analysis

Data Streams

Data Streams

- Sequence is potentially infinite
- High amount of data: sublinear space
- High speed of arrival: sublinear time per example
- Once an element from a data stream has been processed it is discarded or archived

Big Data & Real Time

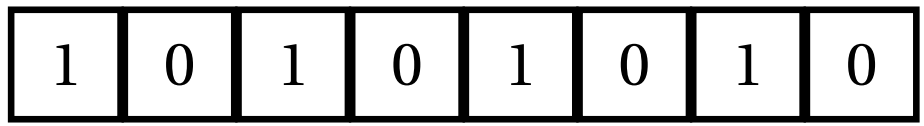
Data Streams

Approximation algorithms

- Small error rate with high probability
- An algorithm $(\varepsilon, \delta)-$approximates F if it outputs $\tilde{F}$ for which $\Pr[|\tilde{F} - F| > \varepsilon F] < \delta$.

Big Data & Real Time

8 Bits Counter



What is the largest number we can store in 8 bits?

Programming Techniques
S.L. Graham, R.L. Rivest Editors

Counting Large Numbers of Events in Small Registers

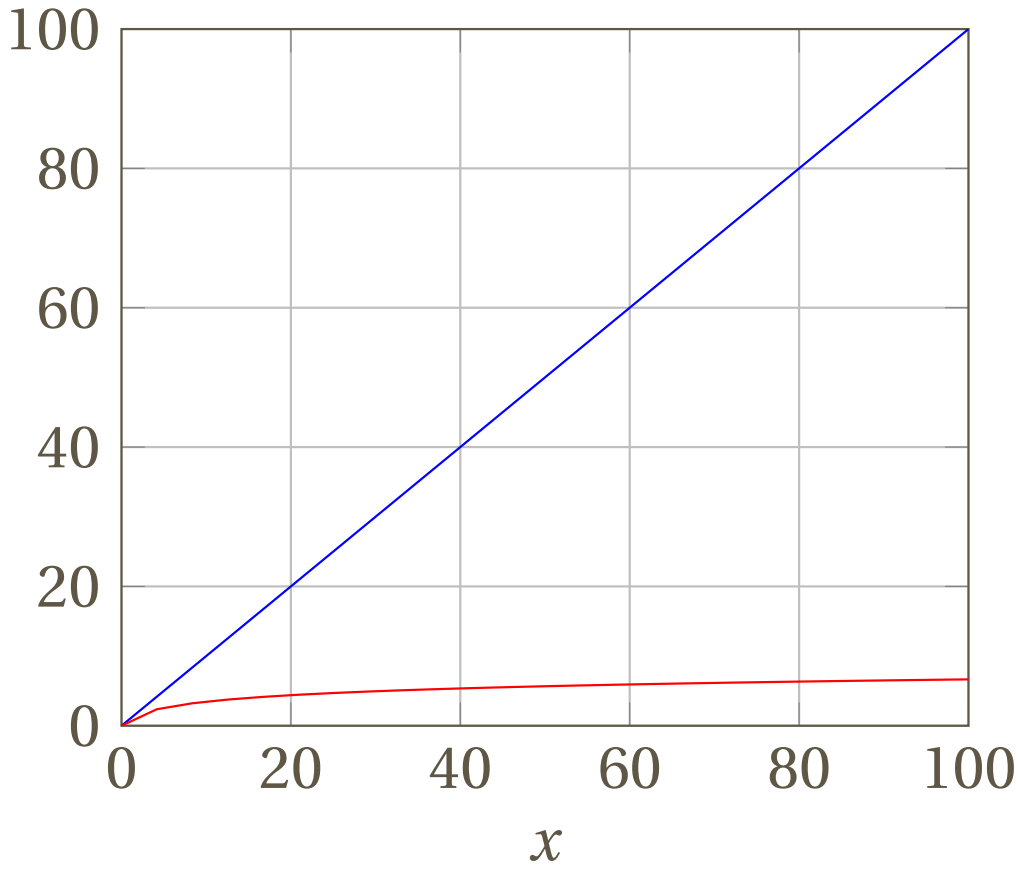
Robert Morris
Bell Laboratories, Murray Hill, N.J.

It is possible to use a small counter to keep approximate counts of large numbers. The resulting expected error can be rather precisely controlled. An example is given in which 8-bit counters (bytes) are used to keep track of as many as 130,000 events with a relative error which is substantially independent of the number n of events. This relative error can be expected to be 24 percent or less 95 percent of the time (i.e. $\sigma = n/8$). The techniques could be used to advantage in multichannel counting hardware or software used for the monitoring of experiments or processes.

What is the largest number we can store in 8 bits?

8 Bits Counter

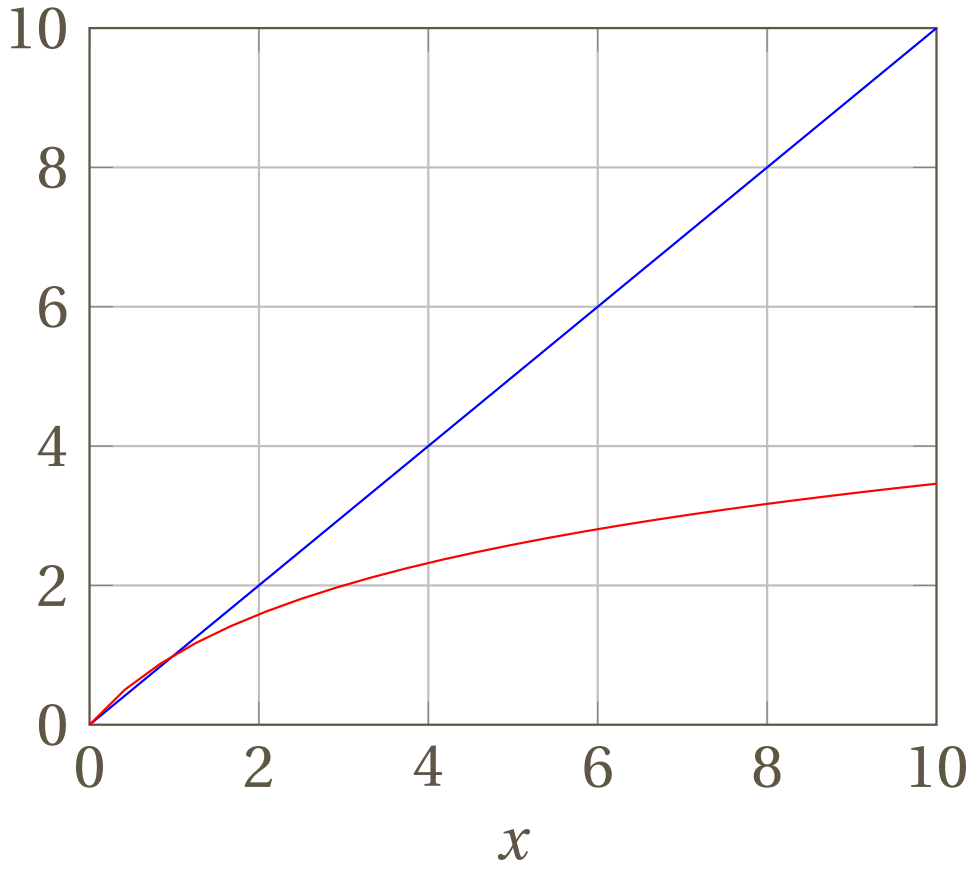
$$f(x) = \log(1+x)/\log(2)$$



$f(0) = 0, f(1) = 1$

8 Bits Counter

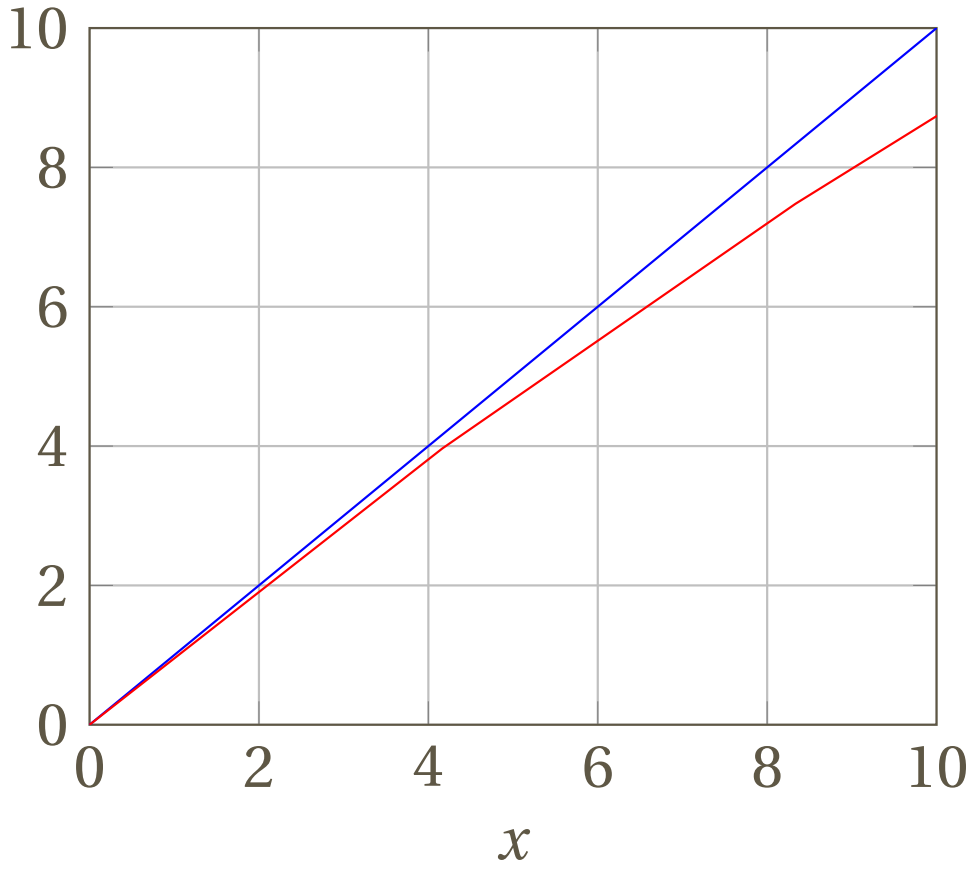
$$f(x) = \log(1+x)/\log(2)$$



$f(0) = 0, f(1) = 1$

8 Bits Counter

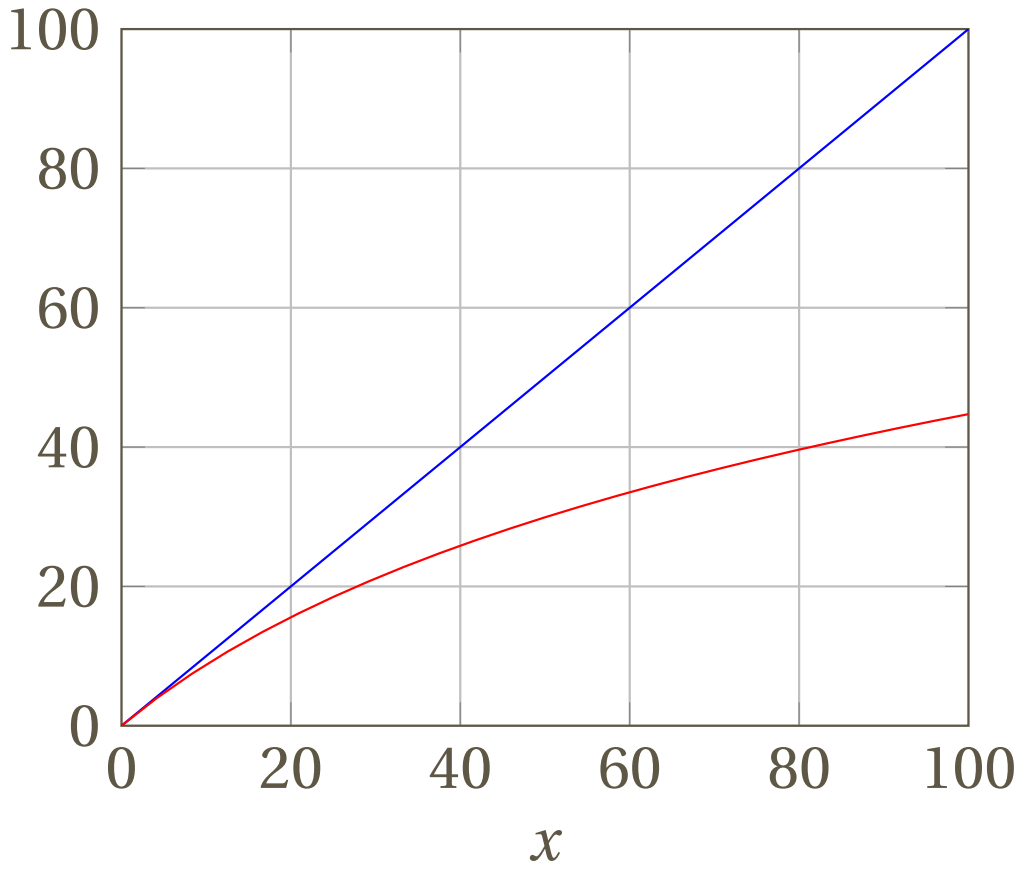
$$f(x) = \log(1 + x/30)/\log(1 + 1/30)$$



$f(0) = 0, f(1) = 1$

8 Bits Counter

$$f(x) = \log(1 + x/30)/\log(1 + 1/30)$$



$f(0) = 0, f(1) = 1$

8 Bits Counter

MORRIS APPROXIMATE COUNTING ALGORITHM

```
1  Init counter c ← 0
2  for every event in the stream
3       do rand = random number between 0 and 1
4          if rand < p
5             then c ← c + 1
```

What is the largest number we can store in 8 bits?

8 Bits Counter

MORRIS APPROXIMATE COUNTING ALGORITHM

```
1  Init counter c ← 0
2  for every event in the stream
3       do rand = random number between 0 and 1
4          if rand < p
5             then c ← c + 1
```

With $p = 1/2$ we can store 2×256
with standard deviation $\sigma = \sqrt{n/2}$

8 Bits Counter

MORRIS APPROXIMATE COUNTING ALGORITHM

```
1  Init counter c ← 0
2  for every event in the stream
3       do rand = random number between 0 and 1
4          if rand < p
5             then c ← c + 1
```

With $p = 2^{-c}$ then $E[2^c] = n+2$ with variance $\sigma^2 = n(n+1)/2$

8 Bits Counter

MORRIS APPROXIMATE COUNTING ALGORITHM

```
1  Init counter c ← 0
2  for every event in the stream
3       do rand = random number between 0 and 1
4          if rand < p
5             then c ← c + 1
```

$$\text{If } p = b^{-c} \text{ then } E[b^c] = n(b-1)+b,$$
$$\sigma^2 = (b-1)n(n+1)/2$$

Data Streams

Example

Puzzle: Finding Missing Numbers

- Let π be a permutation of $\{1, \ldots, n\}$.
- Let π_{-1} be π with one element missing.
- $\pi_{-1}[i]$ arrives in increasing order

Task: Determine the missing number

Data Streams

Example

Puzzle: Finding Missing Numbers

- Let π be a permutation of $\{1, \ldots, n\}$.
- Let π_{-1} be π with one element missing.
- $\pi_{-1}[i]$ arrives in increasing order

Task: Determine the missing number

Use a n-bit vector to memorize all the numbers ($O(n)$ space)

Data Streams

Example

Puzzle: Finding Missing Numbers

- Let π be a permutation of $\{1, \ldots, n\}$.
- Let π_{-1} be π with one element missing.
- $\pi_{-1}[i]$ arrives in increasing order

Task: Determine the missing number

Data Streams:
$O(\log(n))$ space.

Data Streams

Example

Puzzle: Finding Missing Numbers

- Let π be a permutation of $\{1,\ldots,n\}$.
- Let π_{-1} be π with one element missing.
- $\pi_{-1}[i]$ arrives in increasing order

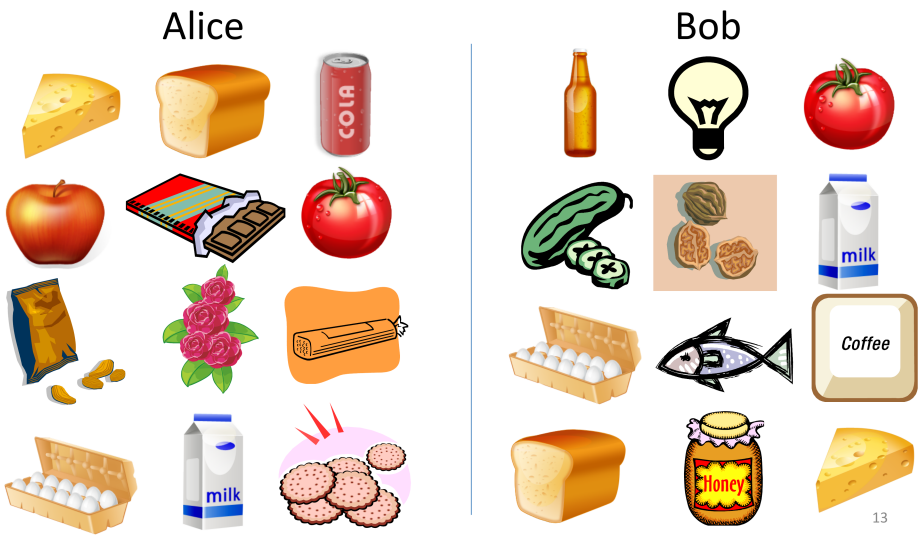
Task: Determine the missing number

Data Streams: $O(\log(n))$ space.

Store

$$\frac{n(n+1)}{2} - \sum_{j \le i} \pi_{-1}[j].$$

Min-wise independent hashing



Similarity: Jaccard Coefficient of A and B

$$= |A \cap B| / |A \cup B|$$

Min-wise independent hashing (Broder et al., 2000)

Let $h : \mathit{Items} \to [0, 1]$ be a random hash function.
Find the 4 items bought by either Alice or Bob with the smallest hash values:

1. Find the 4 smallest hash values for Alice, and the 4 for Bob.
2. Aggregate the results.
3. Estimate the similarity.

Example on estimating the Jaccard similarity in data streams.

Min-wise independent hashing

Alice

h(

) = 0.06

h(

) = 0.09

h(
) = 0.1

h(
) = 0.12

Bob

h(

) = 0.03

h(
) = 0.07

h(

) = 0.09

h(
) = 0.096

1. Find the 4 smallest hash values for Alice, and the 4 for Bob.

Min-wise independent hashing

2. Aggregate the results

h() = 0.03 h() = 0.07 h() = 0.06 h() = 0.09

3. Estimate the similarity.
   - Among the 4 items with the smallest hash values, only milk is bought by Alice and Bob, thus the estimated similarity is 1/4

Example on estimating the Jaccard similarity in data streams.

STRIP

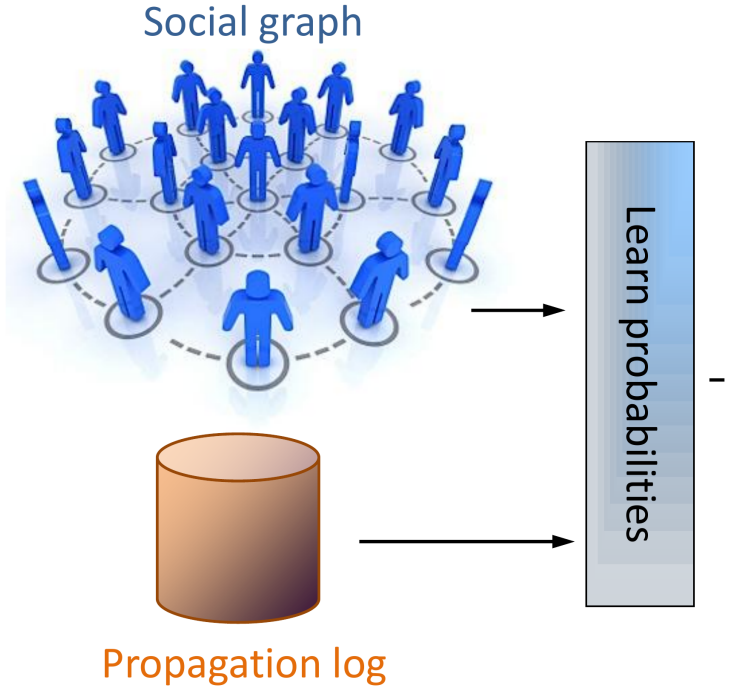
Stream Learning of Influence Probabilities

Konstantin Kutzkov – IT University of Copenhagen, Denmark
Albert Bifet, Francesco Bonchi – Yahoo! Labs, Barcelona
Aristides Gionis – Aalto University and HIIT, Finland

KDD 2013

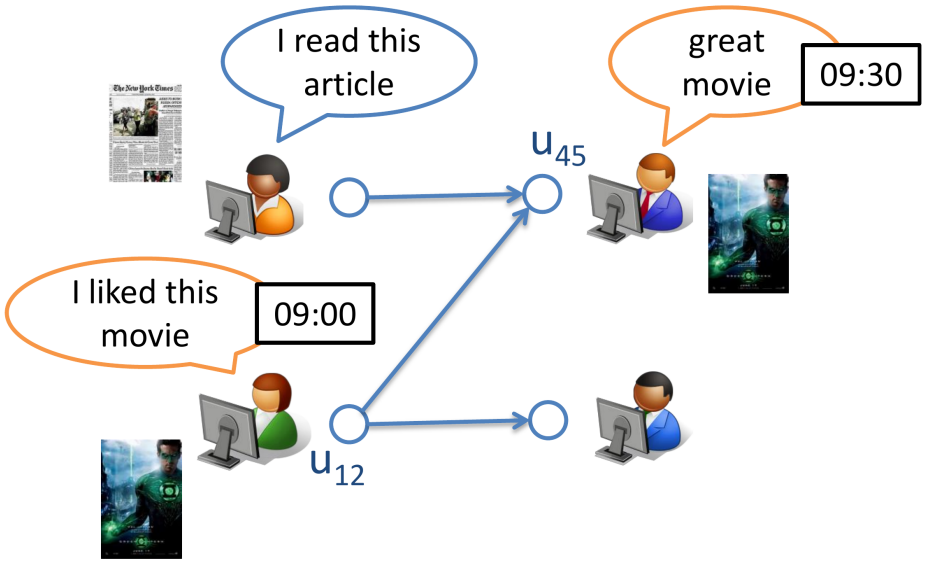
Stream Learning of Influence probabilities

Learning influence strength along links in social networks



- Real-time interactions
- Small amount of time and memory
- Only one pass

Learning influence strength along links in social networks

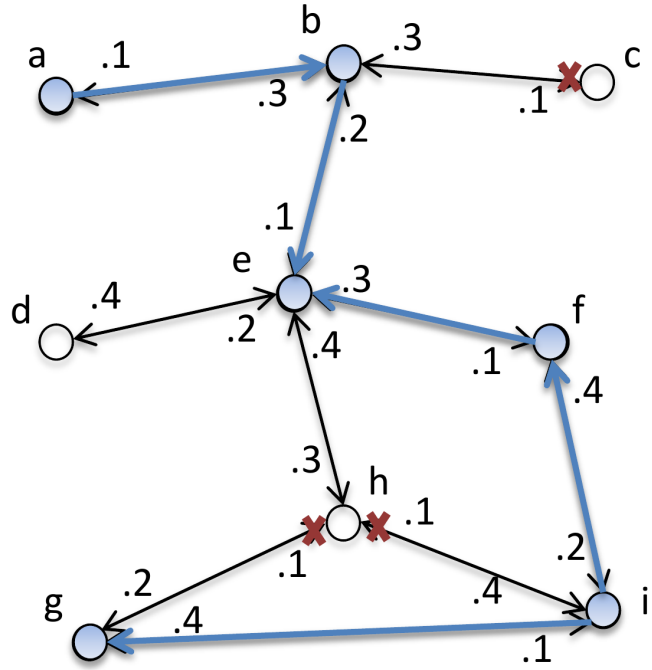


Action	Node	Time
a	u_{12}	1
a	u_{45}	2
a	u_{32}	3
a	u_{76}	8
b	u_{32}	1
b	u_{45}	3
b	u_{98}	7

Social Graph and Log of Propagation

Learning influence probabilities

- Approximate solutions
  - Superlinear space
  - Linear space
  - Sublinear space
- Landmark and sliding window



STRIP: Stream Learning of Influence probabilities

Learning influence probabilities

- The problem of learning the probabilities was first considered in Goyal et al. WSDM'10.
- In particular the following definition was shown to give a good prediction of propagation:
  Let $A_{u2v}(t)$ be the number of actions that propagated from user u to user v within time t and $A_{u|v}$ the total number of actions performed by either u or v. Then

$$p_{u,v} = A_{u2v}(t)/A_{u|v}$$

Similar to Jaccard coefficient for estimating the similarity between sets A and B = $|A \cap B|/|A \cup B|$.

Outline

What is MOA?

{M}assive {O}nline {A}nalysis is a framework for online learning from data streams.

- It is related to WEKA
- It includes a collection of offline and online as well as tools for evaluation: classification, regression, clustering, outlier detection, concept drift detection and recommender systems
- Easy to extend
- Easy to design and run experiments

History - timeline

WEKA

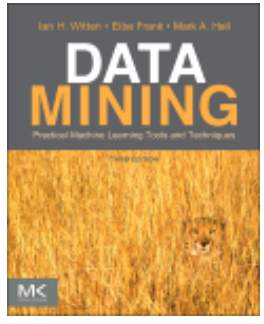
- 1993 - WEKA : project starts (Ian Witten)
- Mid 1999 - WEKA 3 (100% Java) released

MOA

- Nov 2007 - First public release of MOA: Richard Kirkby, Geoff Holmes and Bernhard Pfahringer
- 2009 - MOA Concept Drift Classification
- 2010 - MOA Clustering
- 2011 - MOA Graph Mining, Multi-label classification, Twitter Reader, Active Learning
- 2014 - MOA Outliers, and Recommender System
- 2014 - MOA Concept Drift

WEKA

- Waikato Environment for Knowledge Analysis
- Collection of state-of-the-art machine learning algorithms and data processing tools implemented in Java
  - Released under the GPL
- Support for the whole process of experimental data mining
  - Preparation of input data
  - Statistical evaluation of learning schemes
  - Visualization of input data and the result of learning

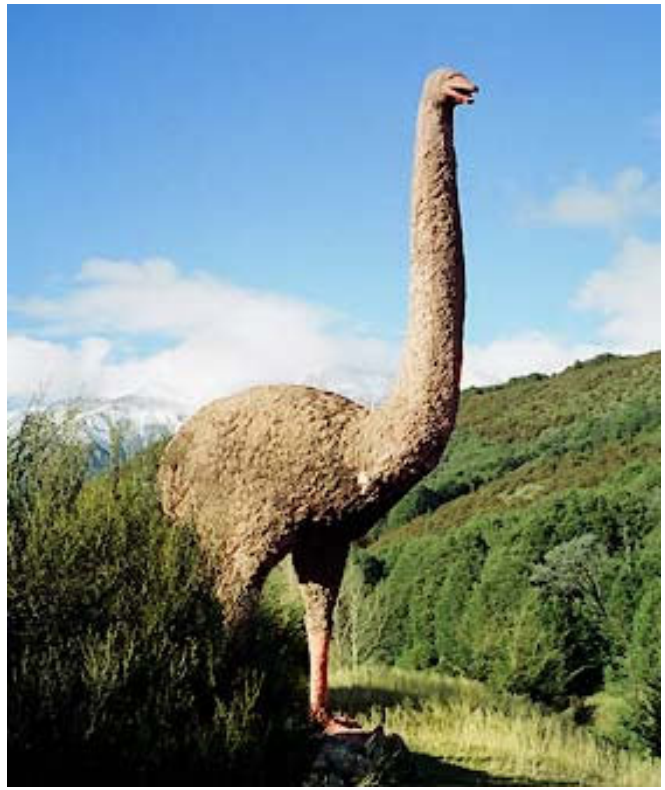


- Used for education, research and applications
- Complements “Data Mining” by Witten & Frank & Hall

WEKA: the bird

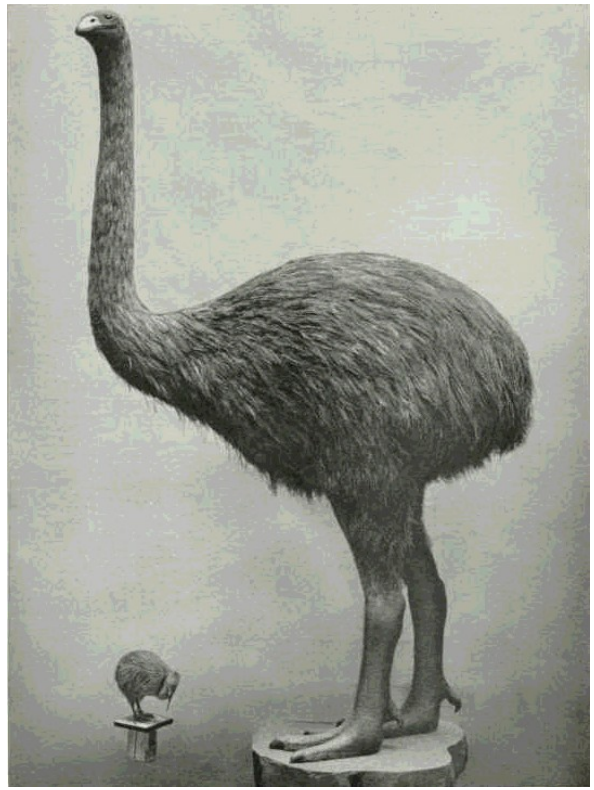
MOA: the bird

The Moa (another native NZ bird) is not only flightless, like the Weka, but also extinct.



MOA: the bird

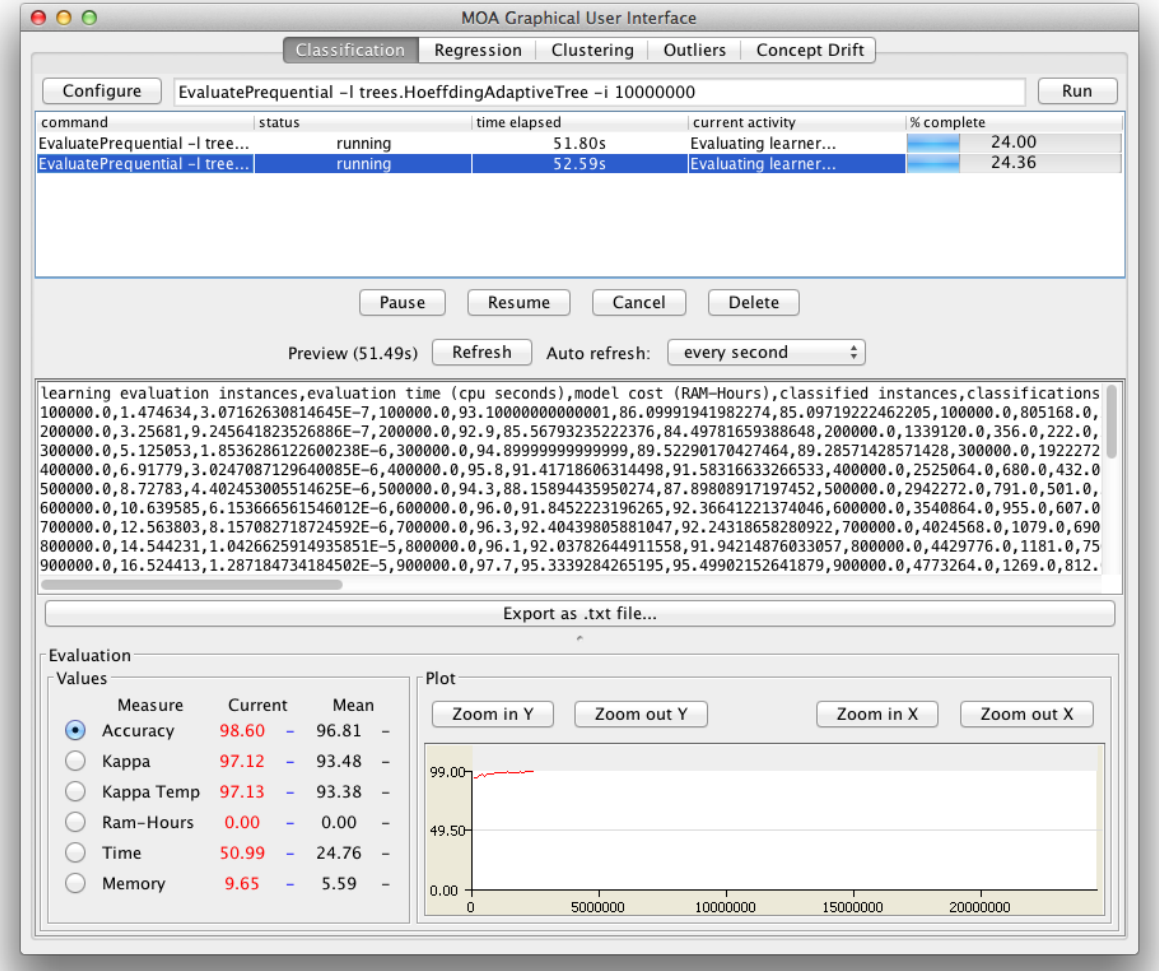
The Moa (another native NZ bird) is not only flightless, like the Weka, but also extinct.



MOA: the bird

The Moa (another native NZ bird) is not only flightless, like the Weka, but also extinct.

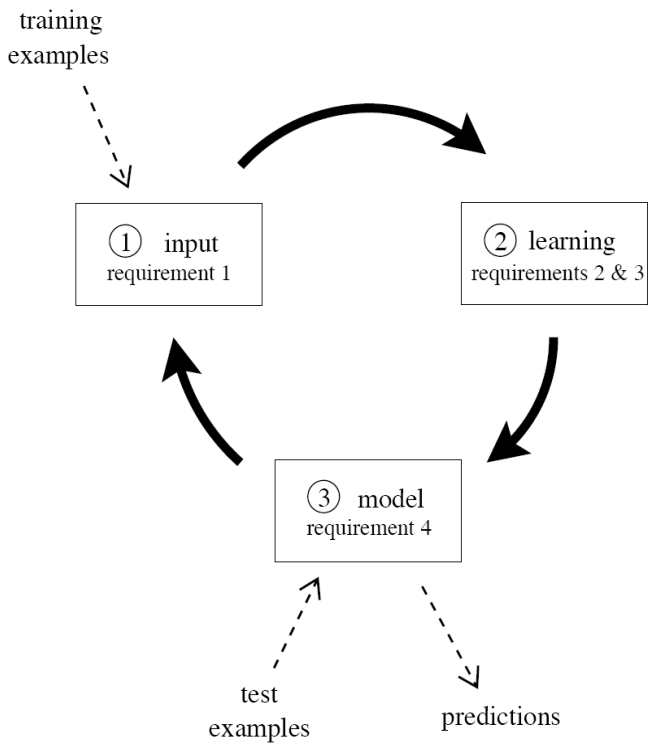
Classification Experimental Setting



Classification Experimental Setting

Evaluation procedures for Data Streams

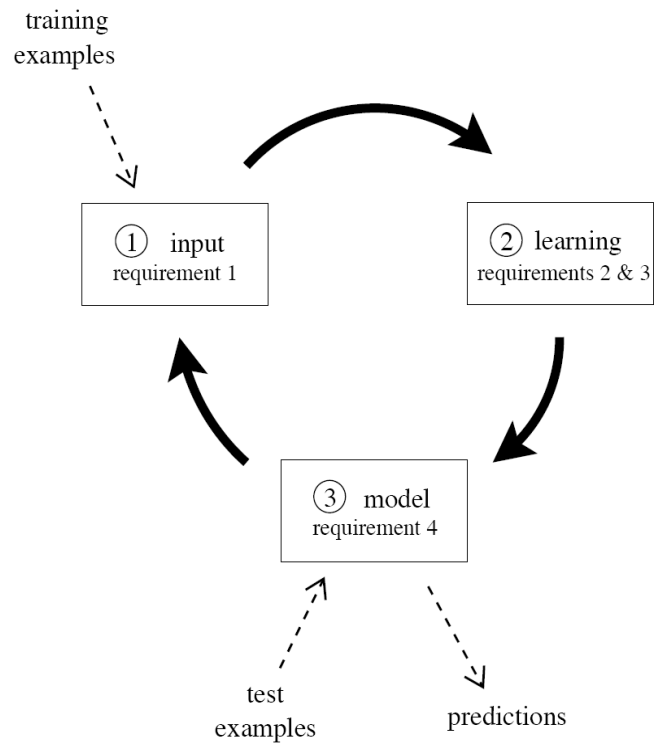
- Holdout
- Interleaved Test-Then-Train or Prequential



Classification Experimental Setting

Data Sources

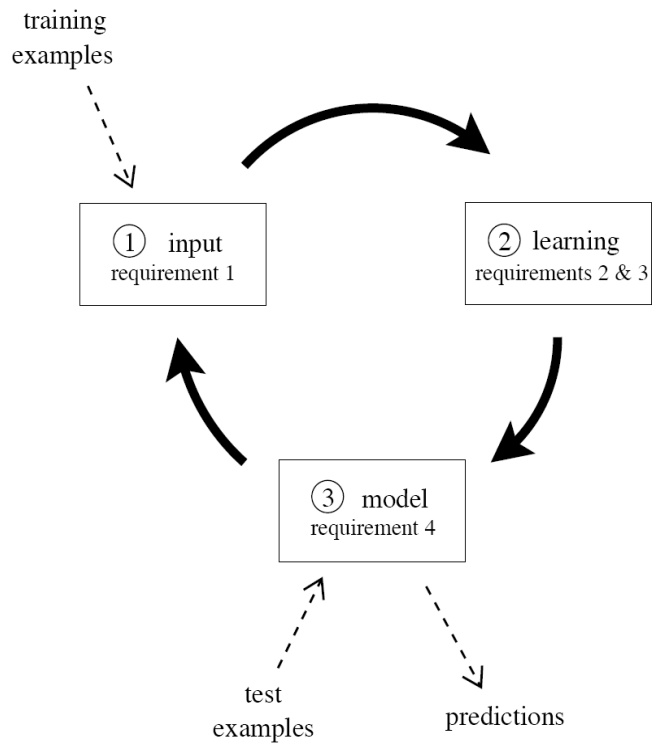
- Random Tree Generator
- Random RBF Generator
- LED Generator
- Waveform Generator
- Hyperplane
- SEA Generator
- STAGGER Generator



Classification Experimental Setting

Classifiers

- Naive Bayes
- Decision stumps
- Hoeffding Tree
- Hoeffding Option Tree
- Bagging and Boosting
- Random Forests
- ADWIN Bagging and Leveraging Bagging
- Adaptive Rules
- Logistic Regression, SGD

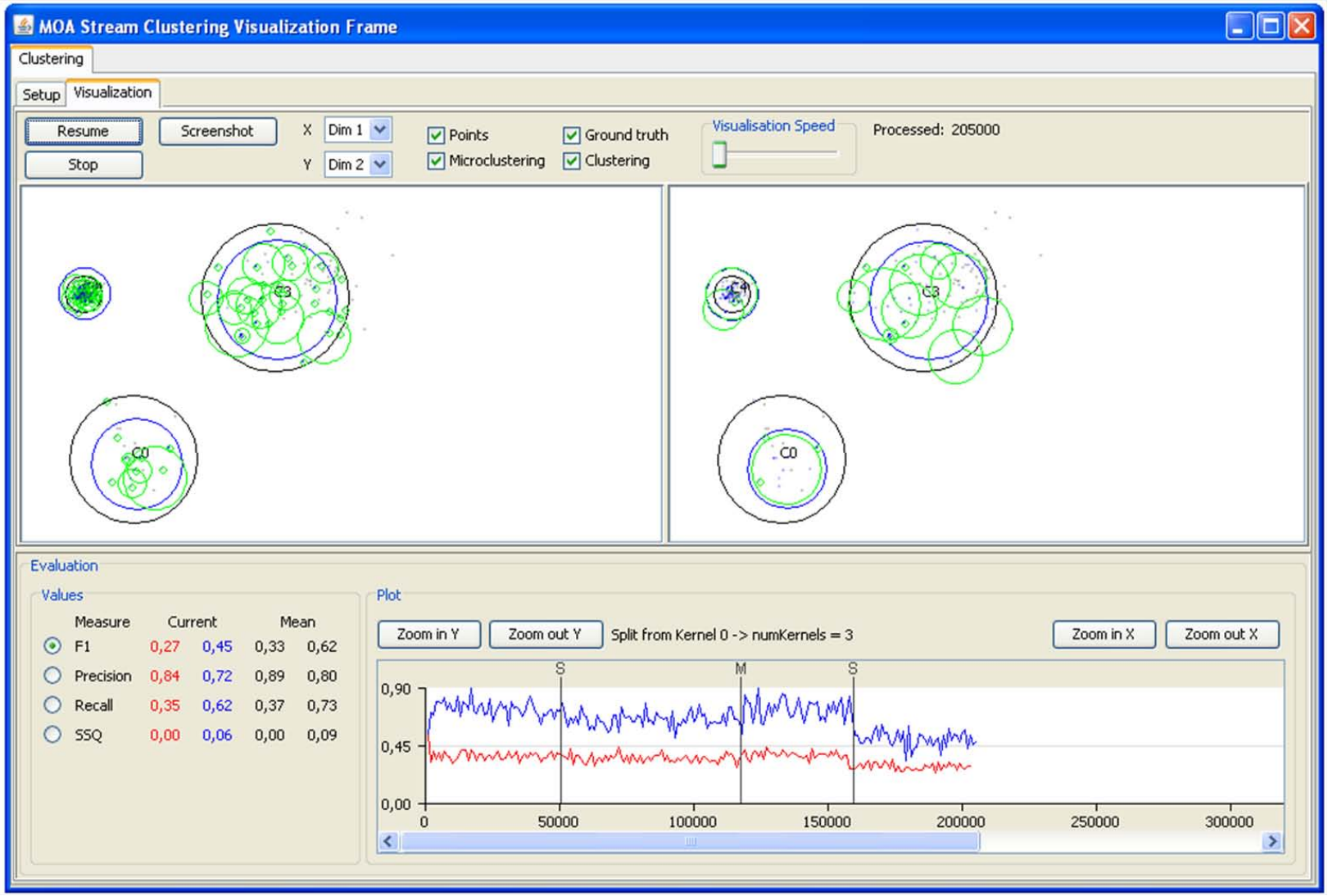


RAM-Hours

RAM-Hour
Every GB of RAM deployed for 1 hour

Cloud Computing Rental Cost Options

Clustering Experimental Setting



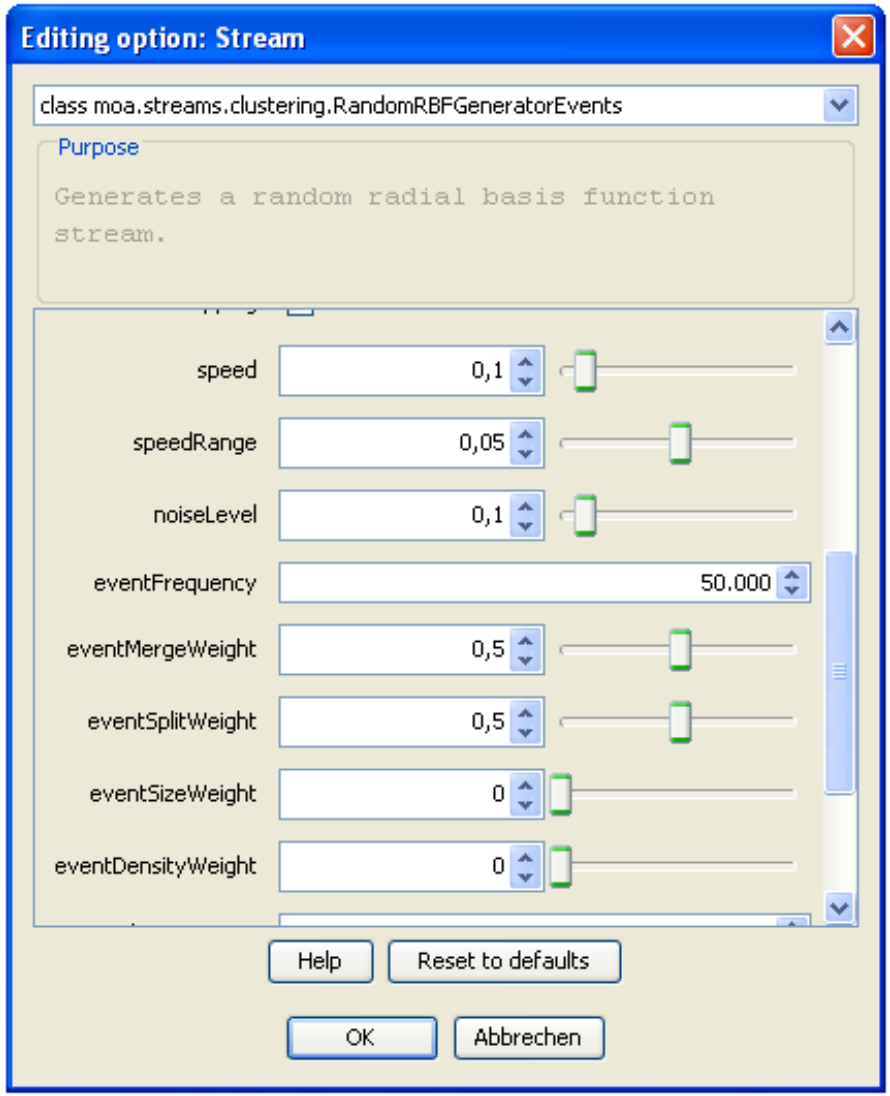
Clustering Experimental Setting

Internal measures	**External measures**
Gamma	Rand statistic
C Index	Jaccard coefficient
Point-Biserial	Folkes and Mallow Index
Log Likelihood	Hubert Γ statistics
Dunn's Index	Minkowski score
Tau	Purity
Tau A	van Dongen criterion
Tau C	V-measure
Somer's Gamma	Completeness
Ratio of Repetition	Homogeneity
Modified Ratio of Repetition	Variation of information
Adjusted Ratio of Clustering	Mutual information
Fagan's Index	Class-based entropy
Deviation Index	Cluster-based entropy
Z-Score Index	Precision
D Index	Recall
Silhouette coefficient	F-measure

Clustering Experimental Setting

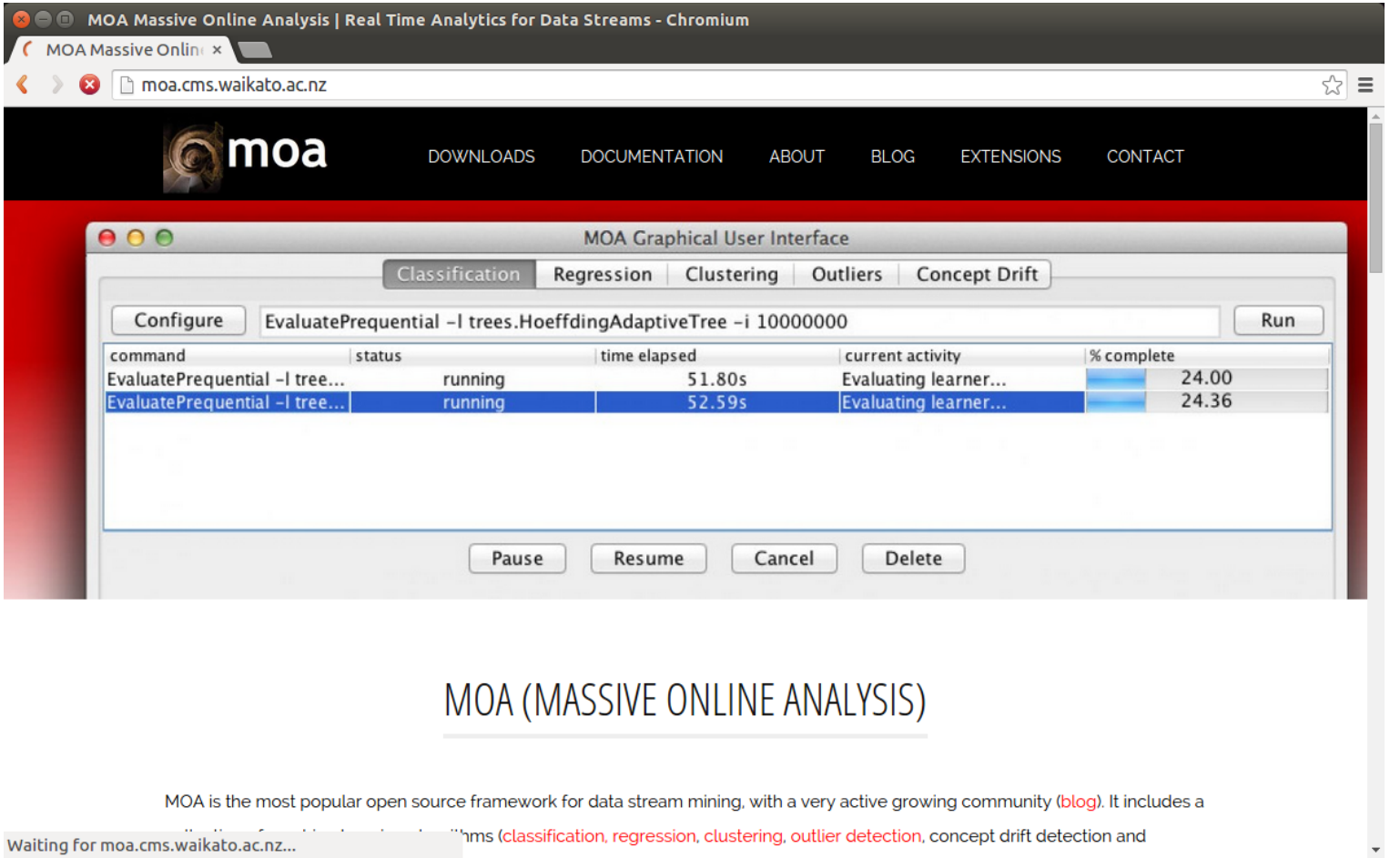
Clusterers

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



Web

http://www.moa.cms.waikato.ac.nz



Easy Design of a MOA classifier

- `void resetLearningImpl ()`
- `void trainOnInstanceImpl (Instance inst)`
- `double[] getVotesForInstance (Instance i)`

Easy Design of a MOA clusterer

- `void resetLearningImpl ()`
- `void trainOnInstanceImpl (Instance inst)`
- `Clustering getClusteringResult()`

Extensions of MOA

- Multi-label Classification
- Active Learning
- Regression
- Closed Frequent Graph Mining
- Twitter Sentiment Analysis

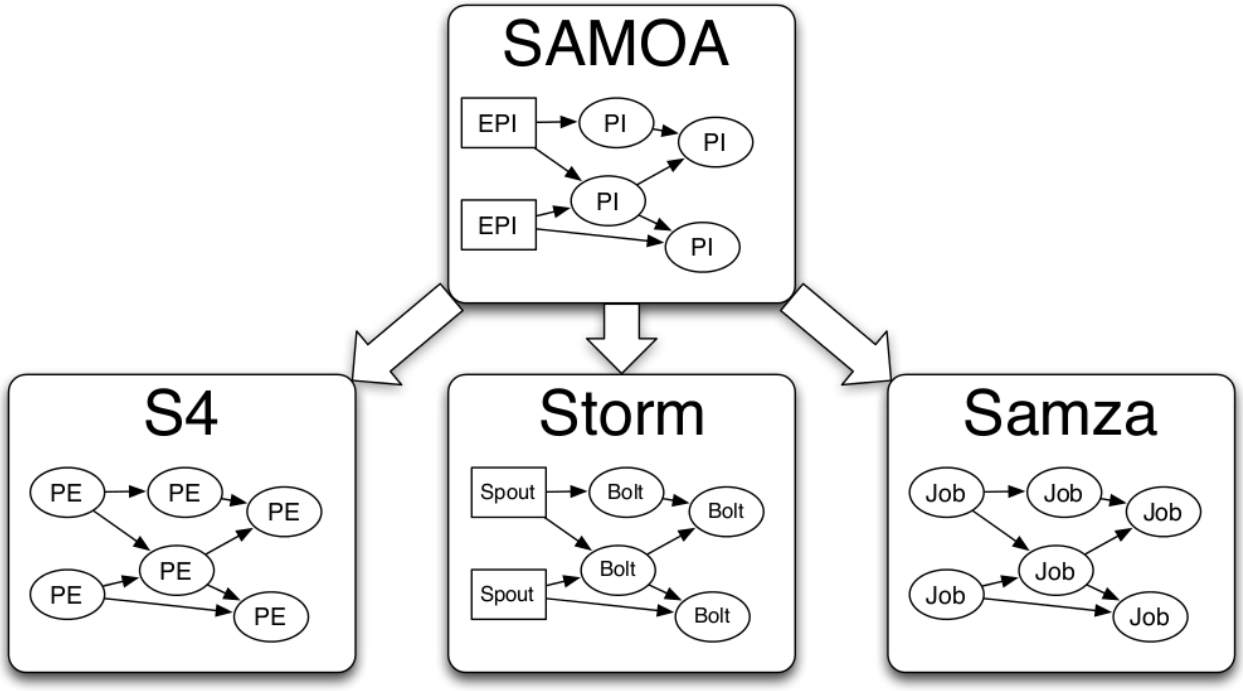
Outline

Big Data Stream Analytics

MOA: Massive Online Analysis

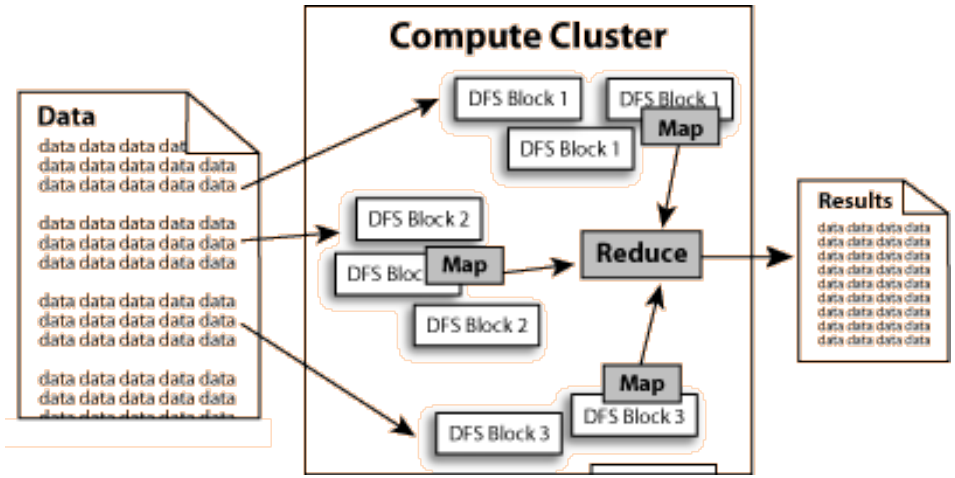
SAMOA: Scalable Advanced Massive Online Analysis

SAMOA



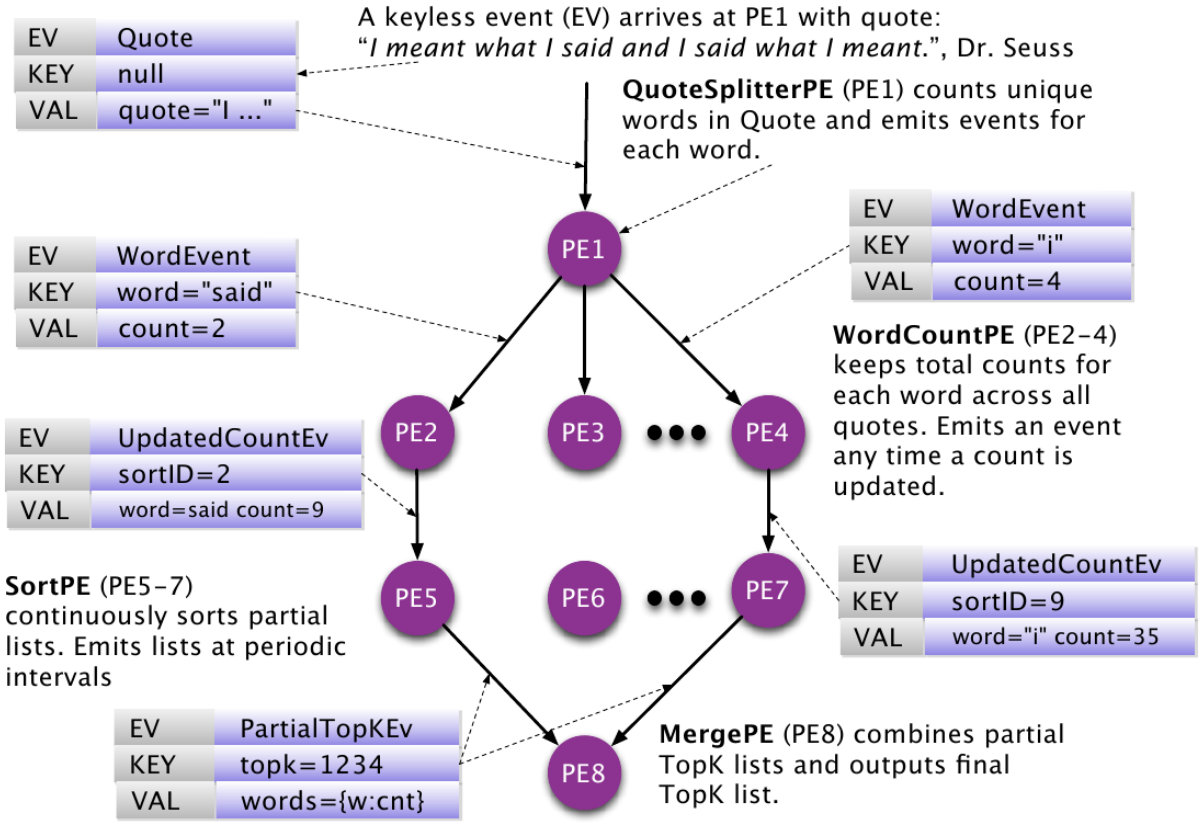
SAMOA is distributed streaming machine learning (ML) framework that contains a programing abstraction for distributed streaming ML algorithms.

Hadoop

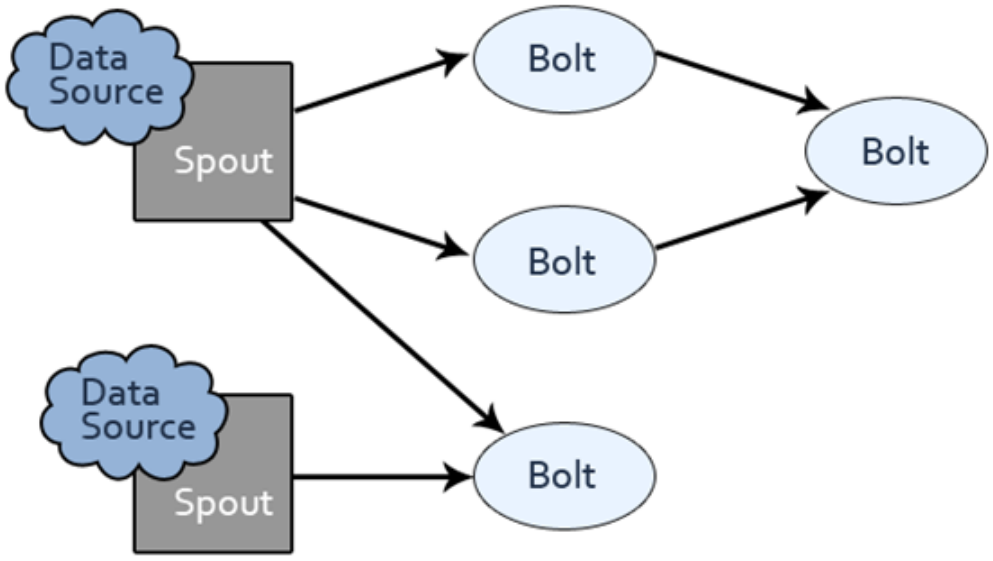


Hadoop architecture

Apache S4



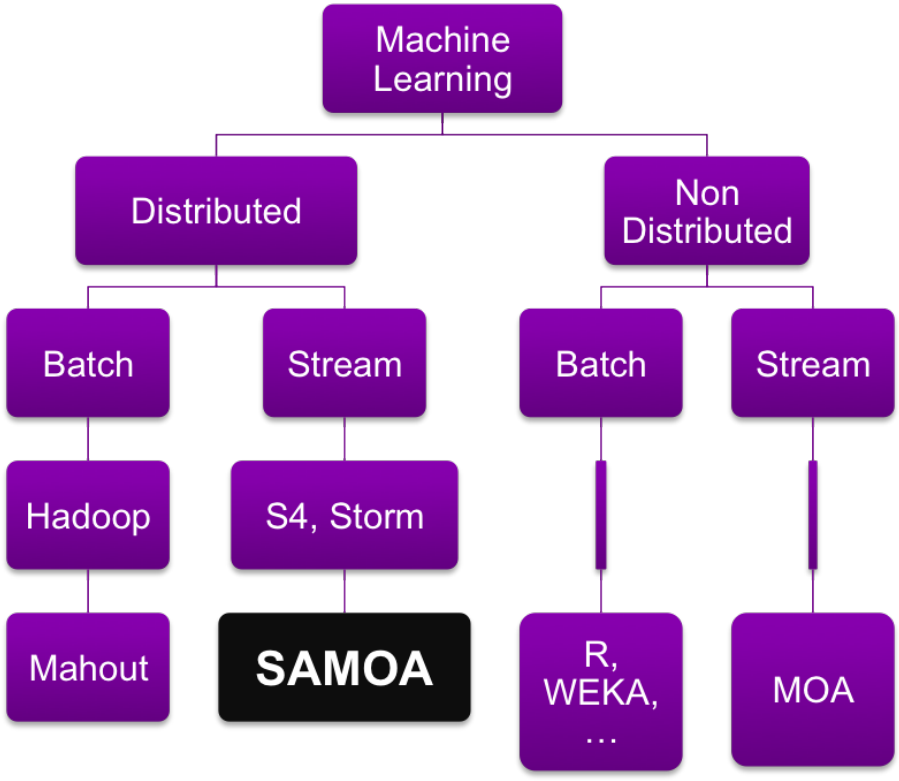
PE ID	PE Name	Key Tuple
PE1	QuoteSplitterPE	null
PE2	WordCountPE	word="said"
PE4	WordCountPE	word="i"
PE5	SortPE	sortID=2
PE7	SortPE	sortID=9
PE8	MergePE	topK=1234



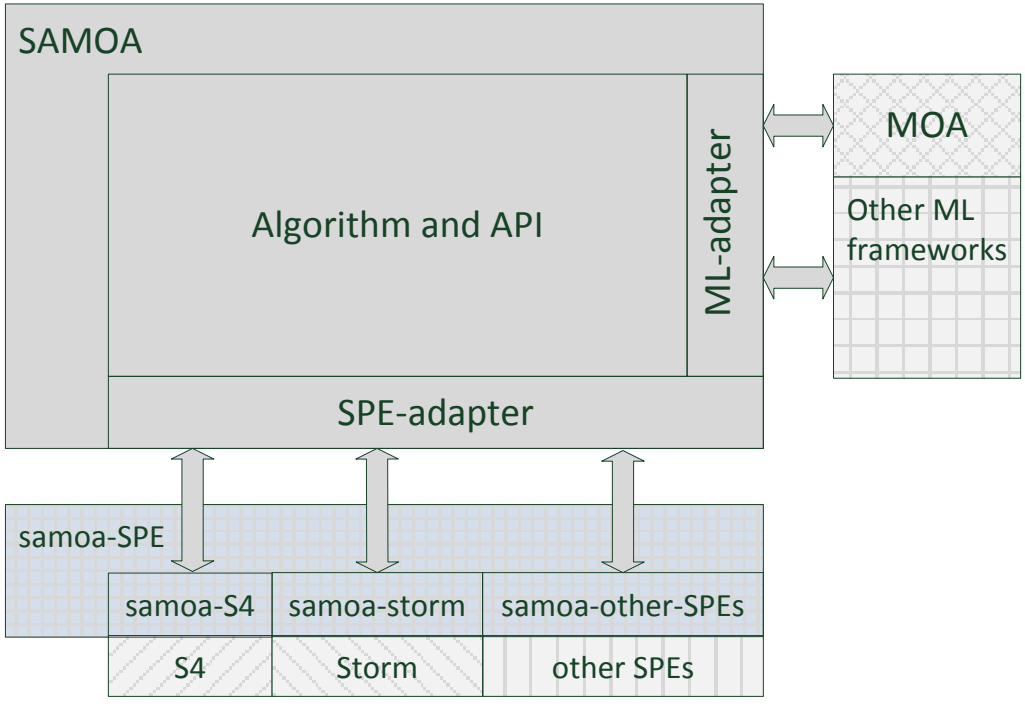
A Storm Topology

Stream, Spout, Bolt, Topology

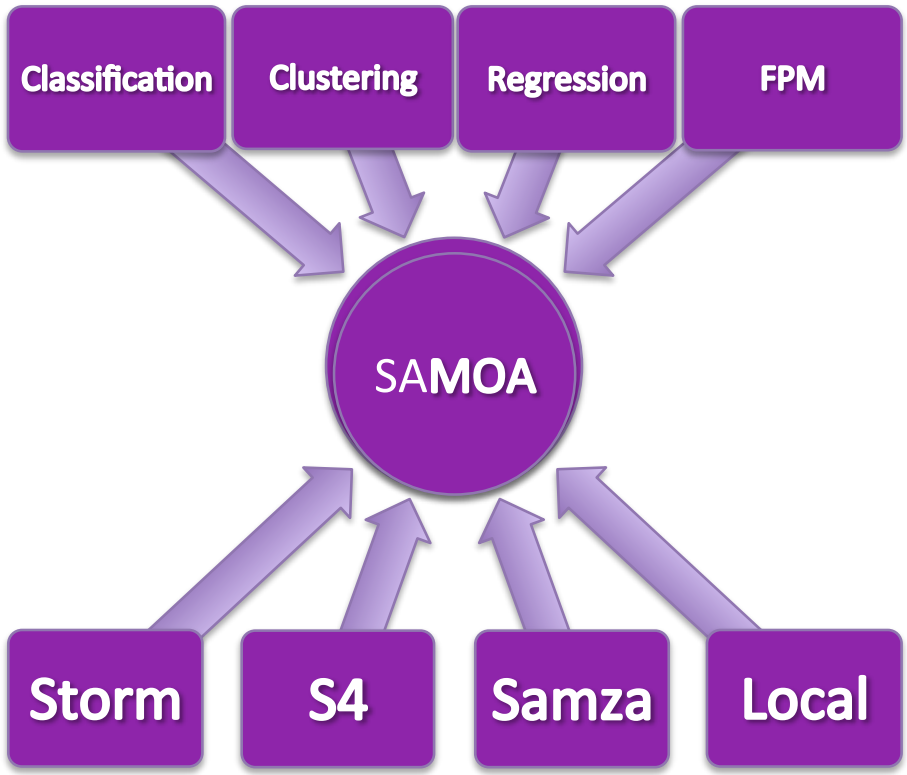
SAMOA


Machine Learning
Distributed
Non Distributed
Batch
Stream
Batch
Stream
Hadoop
S4, Storm
Mahout
SAMOA
R, WEKA, …
MOA

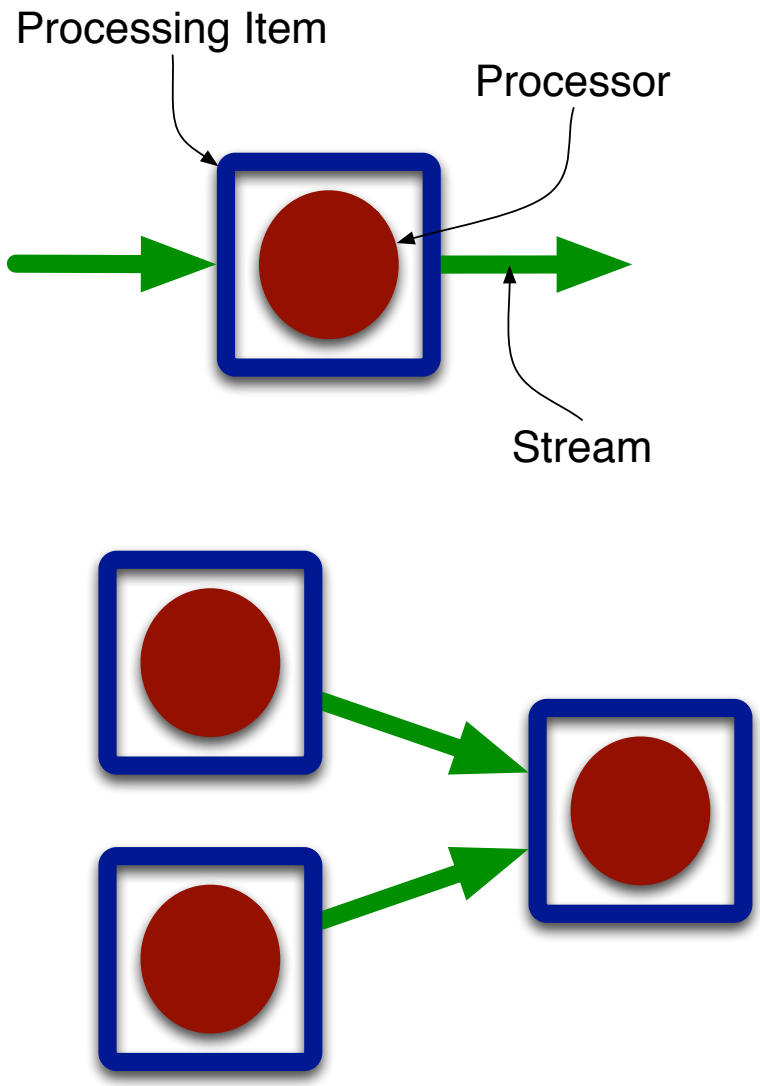
SAMOA


SAMOA
Algorithm and API
ML-adapter
SPE-adapter
MOA
Other ML frameworks
samoa-SPE
samoa-S4
samoa-storm
samoa-other-SPEs
S4
Storm
other SPEs

SAMOA


Classification
Clustering
Regression
FPM
SAMOA
Storm
S4
Samza
Local

SAMOA ML Developer API

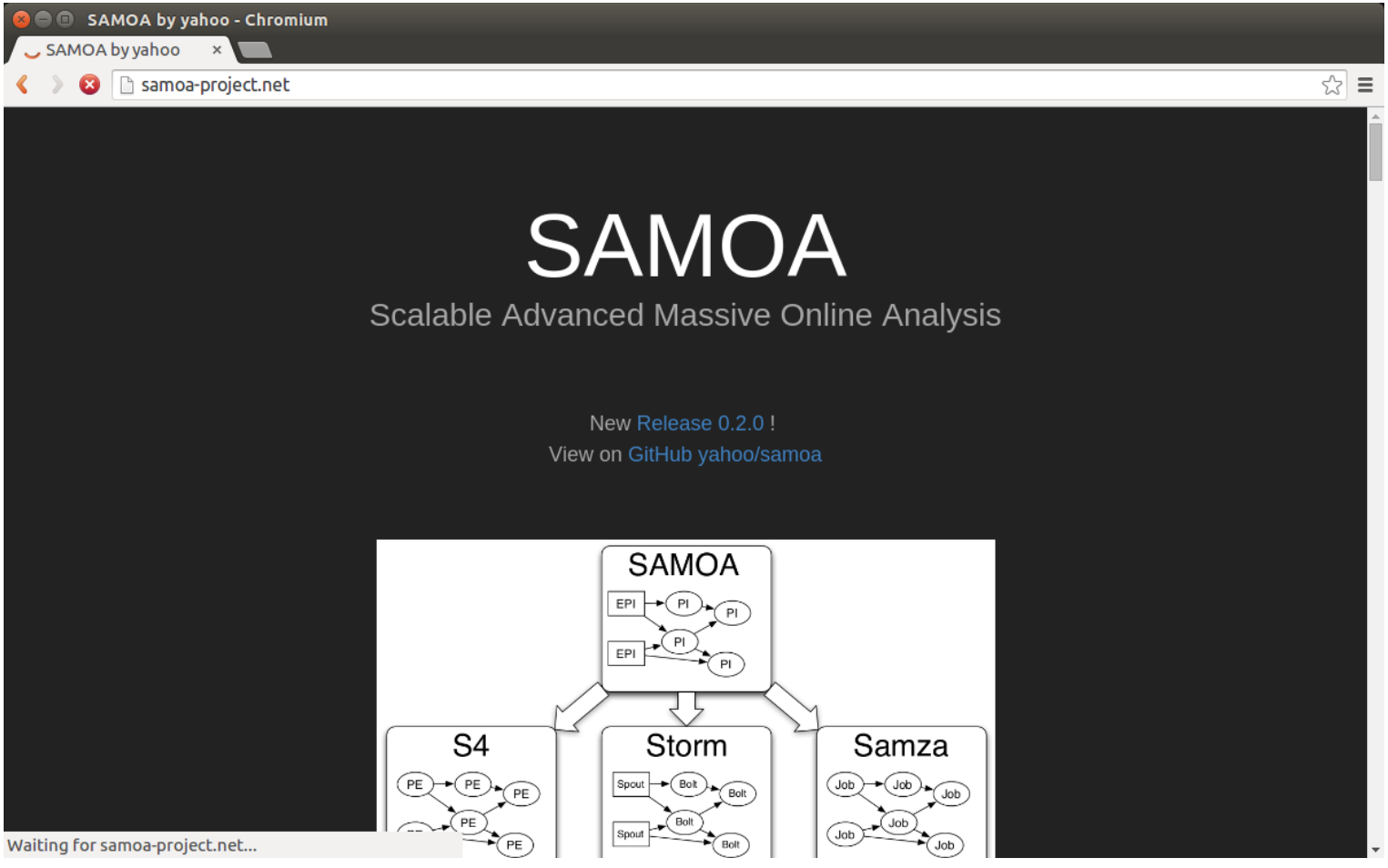


SAMOA ML Developer API

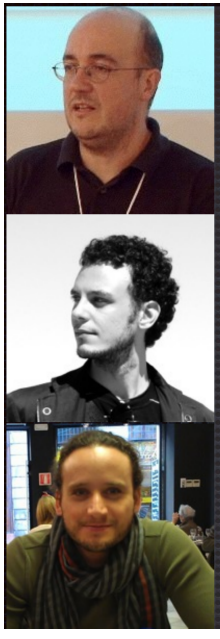
```
TopologyBuilder builder;
Processor sourceOne = new SourceProcessor();
builder.addProcessor(sourceOne);
Stream streamOne = builder.createStream(sourceOne);

Processor sourceTwo = new SourceProcessor();
builder.addProcessor(sourceTwo);
Stream streamTwo = builder.createStream(sourceTwo);

Processor join = new JoinProcessor());
builder.addProcessor(join)
  .connectInputShuffle(streamOne)
  .connectInputKey(streamTwo);
```

Web

http://samoa-project.net/



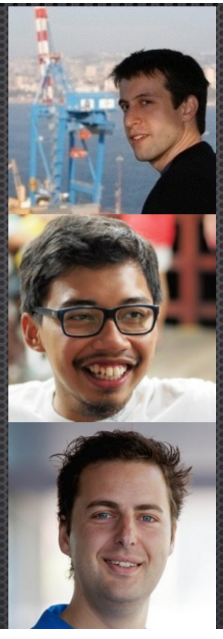
The Team



Albert
Bifet

Gianmarco
De Francisci Morales

Nicolas
Kourtellis



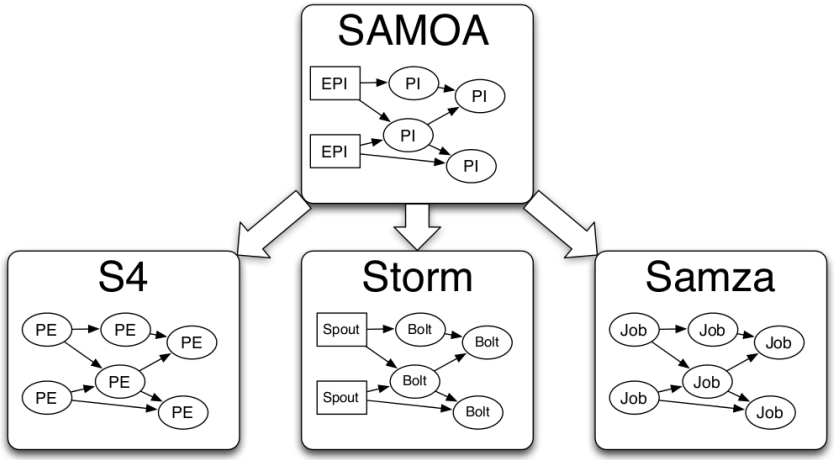
Matthieu
Morel

Arinto
Murdopo

Olivier
Van Laere

Summary

1. To improve benefits, companies need to satisfy costumers. To do this they need to know
   - what potential costumers needs
     - social networks
     - real-time analytics
2. MOA deals with evolving data streams
3. SAMOA is distributed streaming machine learning



Thanks!

@abifet

http://moa.cms.waikato.ac.nz/
@moadatamining

http://samoa-project.net/
@samoa_project