# Streaming and Parallelized Coresets construction and its applications

Wei Ma  Max Ma

CMU 10805, 2016 Fall

# Outline

- Motivation
- Coresets
- Conceptual tree based architecture
- Asynchronized architecture
- Experiments

# Motivation

- Huge "volume" and "velocity" of the data being produced
- Limited computation and storage resources
- How to get a SKETCH of the full dataset?
- A coreset yields $(1+\varepsilon)$ approximation to the original dataset.

# Coresets: Definition

**Definition**

A small number of data set $S$ can approximate the measures of whole point sets $P$. Note $S$ is not necessarily a subset of $P$, where we refer $S$ is a strong coreset of $P$. Mathematically,

$$(1-\varepsilon)\mu(S) \leq \mu(P) \leq (1+\varepsilon)\mu(S) \tag{1}$$

- Gaussian Mixture: Likelihood
- K-means: $L^2$ distance

# Coresets: Variants

- Singular Value Decomposition (SVD):
  - Strong coresets: may generate new data
  - Used for $(j, k)$-projective clustering: projecting $n$ rows data to any set of $k$ affine subspaces, each of dimension at most $j$
  - $|C| \sim \mathcal{O}(\log(n))$
- Adaptive Sampling:
  - Weak coresets: preserve original data
  - $|C| \sim \mathcal{O}(poly(d))$

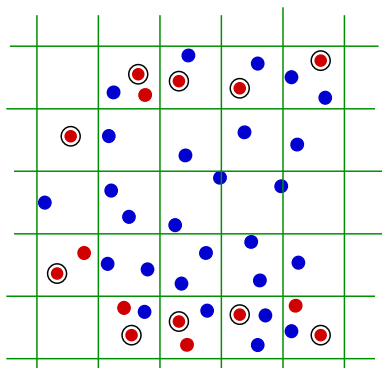


Figure: Illustration of adaptive sampling

# Coresets: Cool feature

**Takeaway Message**

Coresets are closed under UNION operation.

- Construct coresets in parallel
- Friendly to new data

However, no practical implementation of coresets construction available.

# Conceptual tree based architecture

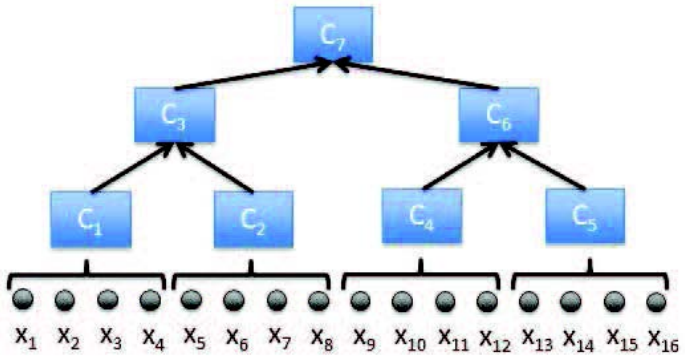



Figure: Tree based construction for coresets

- All-reduce framework
- Low I/O, high computational intensity, not good for Hadoop/Spark
- Single core reading; Multi-core processing; In memory
- Coreset construction is more related to high performance computing (HPC), good for MPI.

# Asynchronized architecture

- Data structure: $m$ data slots with level $l$
- K processors, each processor can:
  - Read data into a slot and mark as level 1
  - Merge slots at same level and increase the level by 1
  - If no data/same level slots can be read/merged, merge slots from different levels
- Only one slot will remain active, and it is the final coreset

# MPI implementation

A lots of advances techniques in MPI are adopted.

- One-sided communication: remote memory access
- MPI_FILE_IO: shared file handlers

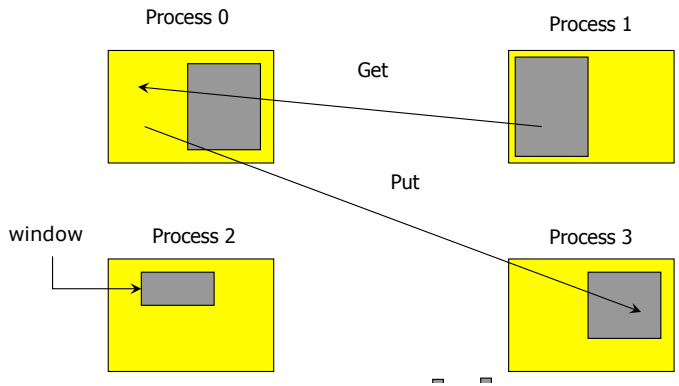



Figure: MPI One-sided communication

- Implemented by Open MPI C++
- https://github.com/Lemma1/Distributed-Coresets

# Experiments: fake data test

- Intel(R) Xeon(R) CPU L5420@2.50GHz, 8 cores, 64-bit, 16 GB memory
- $d = 100, |C| = 100, m = 20$

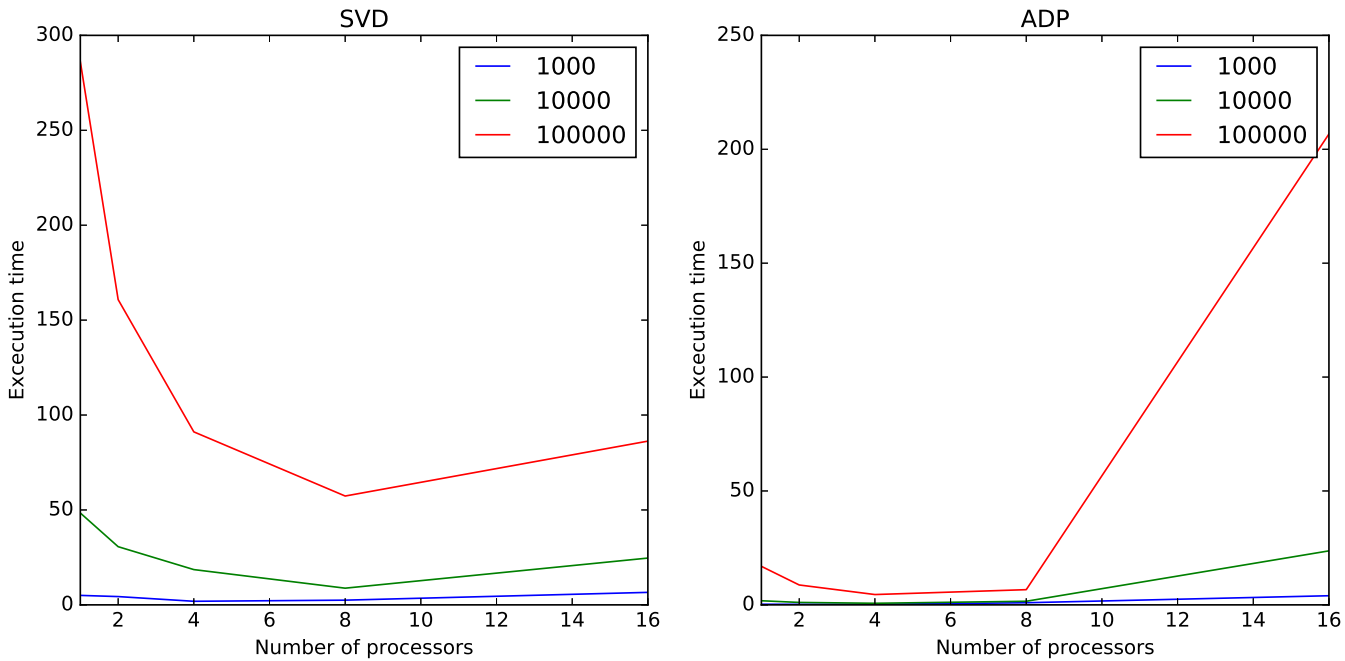



Figure: Runing time on different data set

# Experiments: MNIST

- The MNIST database of handwritten digits, available from this page, has a training set of 60,000 examples, and a test set of 10,000 examples
- The shape of each digit is $8 \times 8$

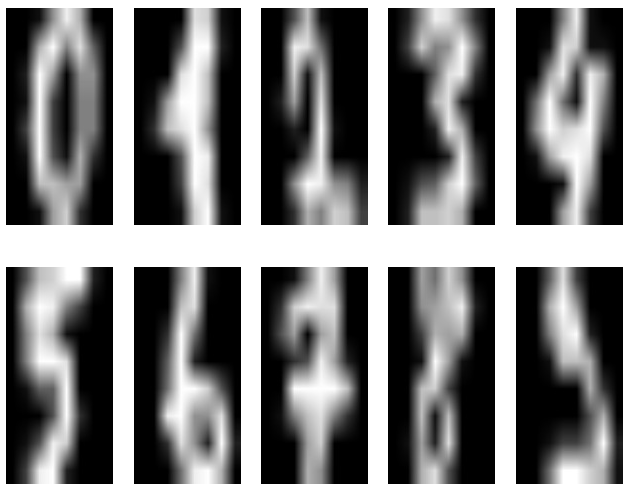
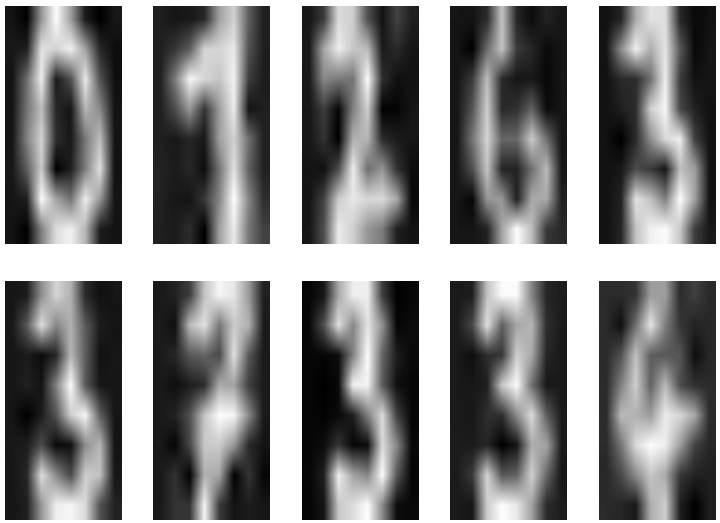


Figure: Example of MNIST data

# Experiments: MNIST - cont



(a) SVD wich coreset size 30

(b) ADS wich coreset size 30

# Experiments: MNIST - cont

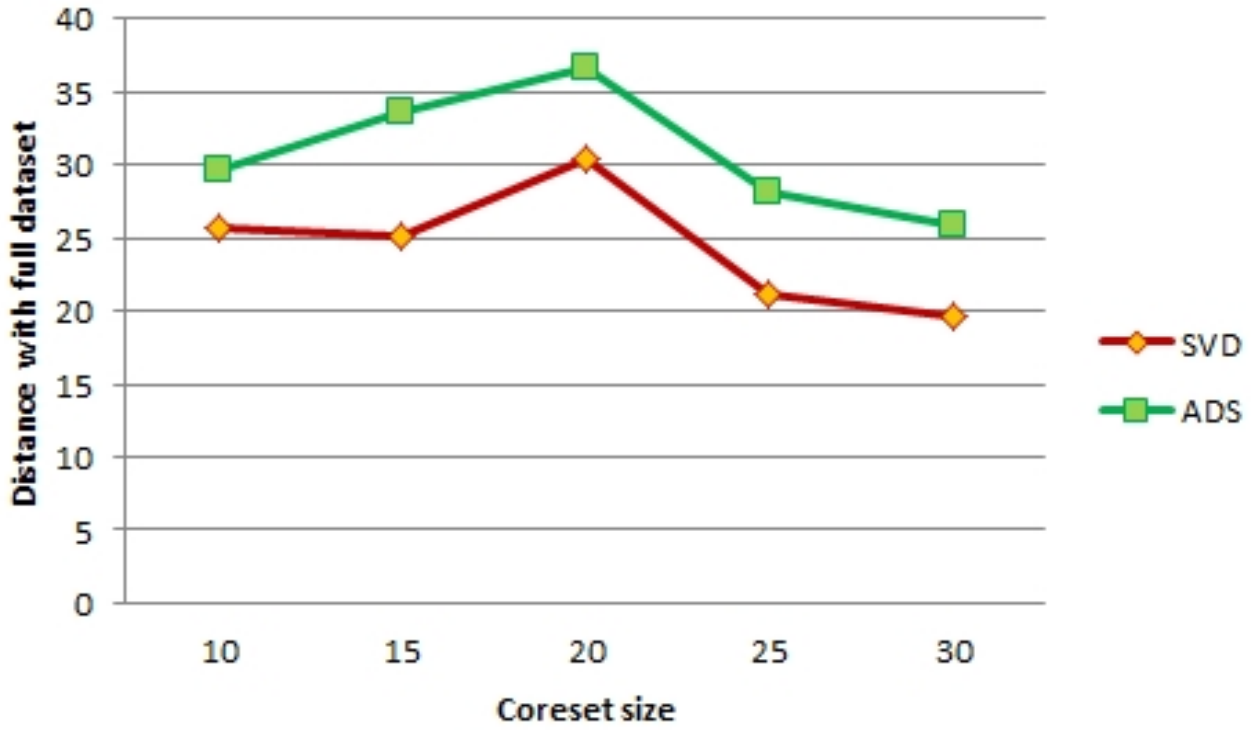
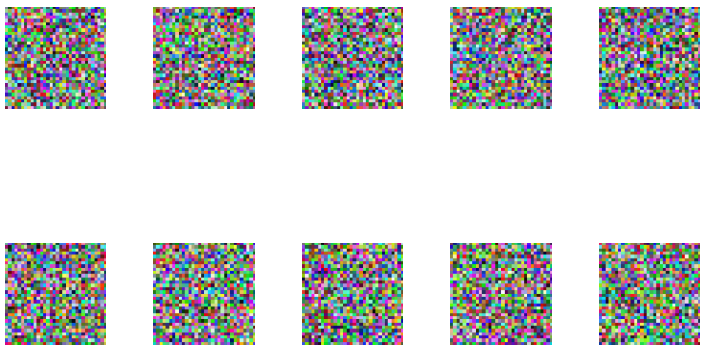



Figure: Accuracy on coreset size

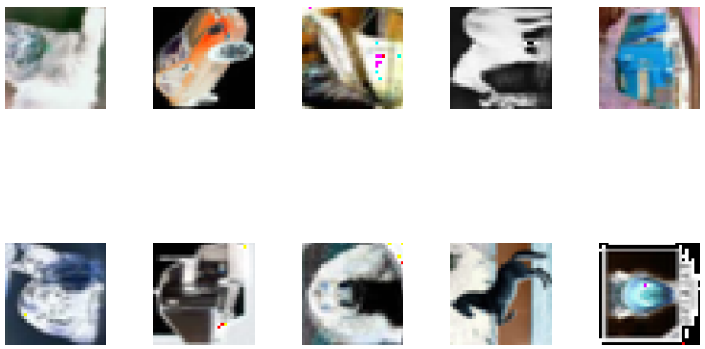
# Experiments: CIFAR

- The CIFAR-10 are labeled subsets of the 80 million tiny images dataset.
- The shape of each image is $32 \times 32 \times 3$

# Experiments: CIFAR - cont



(a) SVD wich coreset size 30



(b) ADS wich coreset size 30

*Thanks*