Efficient Data Reduction and Summarization

Research Progress Report

Ping Li

Department of Statistical Science

Faculty of Computing and Information Science

Cornell University

December 03, 2009

Publications

Papers that acknowledged the FODAVA grant:

1. Ping Li, *ABC-Boost for Multi-Class Classification*, ICML, 2009
2. Ping Li, *Improving Compressed Counting*, UAI, 2009
3. Ping Li, *Compressed Counting*, SODA, 2009
4. Ping Li, *Computationally Efficient Estimators for Dimension Reduction in* L_α *Using Stable Random Projections*, IEEE ICDM, 2008
5. Ping Li, Kenneth Church, and Trevor Hastie, *One Sketch for All: Theory and Applications of Conditional Random Sampling*, NIPS, 2008

Major Research Progress

1. Efficient dimension reduction algorithms with guaranteed performance.

   Stable random projections for estimating L_α distances, where $0 < \alpha \leq 2$.

2. Efficient dimension reduction algorithms invented for sparse data.

   Conditional Random Sampling (CRS), well suitable for text data.

3. Efficient data stream computation algorithms

   Both stable random projections and CRS are applicable to dynamic data.

4. Compressed Counting Efficient data stream algorithms invented by taking advantage of the fact that most data are non-negative. Especially suitable for computing entropy of data streams in network traffic monitoring and anomaly detection.

5 Boosting algorithms for classification

Adaptive Base Class (ABC) Boost,

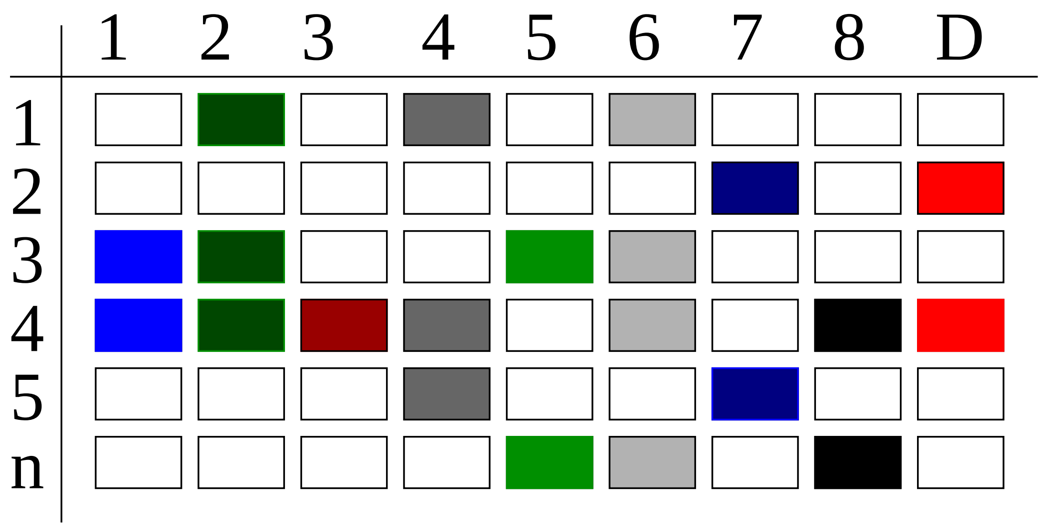
Robust logitboost, ABC-MART, ABC-LogitBoost.

Surprisingly significant improvements over Friedman's (and Friedman et. al.) classical algorithms in many datasets.

This project was not included in the original proposal; we are still actively seeking funding to continue this work.

Modern Data Matrix

Data matrix $\mathbf{A} \in \mathbb{R}^{n \times D}$: n rows and D columns.



- Massive eg, both n, $D \approx 10^{10}$
- Dynamic eg, high-speed data streams
- Often Sparse eg, text data

Massive Data Summarization and Some Challenges

Summarization is fundamental in learning, visualization, and linear algebra.

- Summary statistics of individual rows (or columns)
  eg, αth moment $\sum_{i=1}^{D} |u_i|^\alpha$, entropy, etc.

- Summary statistics between rows (or columns)
  eg, dot products, αth distance $\sum_{i=1}^{D} |u_i - v_i|^\alpha$, χ^2 distance, etc.

Some challenges

- Memory intensive Loading $\mathbf{A} \in \mathbb{R}^{n\times D}$ may be infeasible.
  Loading all pairwise (eg, n^2) distances of $\mathbf{A}$ can be easily infeasible.

- CPU intensive

- Dynamic updating

From Exact Answers to Approximations

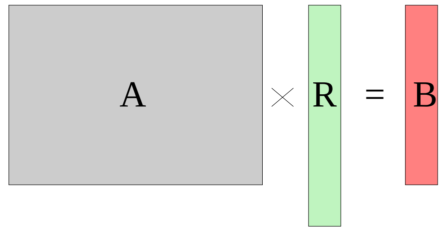
(Good) Approximate summary statistics (eg distances) often suffice

- Visualization systems only need a certain resolution.
- Good (robust) algorithms are stable even using approximate inputs.

Simple random sampling (eg using a few columns) is not enough

- Not accurate.
- Not suitable for sparse data.

(Symmetric) Stable Random Projections



- Original data matrix $\mathbf{A} \in \mathbb{R}^{n \times D}$: n rows and D columns,
  Massive, eg, both n, $D = O\left(10^{10}\right)$.
  Possibly dynamic, according to the Turnstile model.

- Projection matrix $\mathbf{R} \in \mathbb{R}^{D \times k}$: D rows and k columns, $k \ll n, D$
  Entries are samples of a symmetric α-stable distribution.
  $\alpha = 2$: Normal distribution. $\alpha = 1$: Cauchy distribution.

- Projected matrix $\mathbf{B} \in \mathbb{R}^{n \times k}$: n rows and k columns
  Viewed as a sketch of $\mathbf{A}$, which may be discarded.

Symmetric α-Stable Distributions

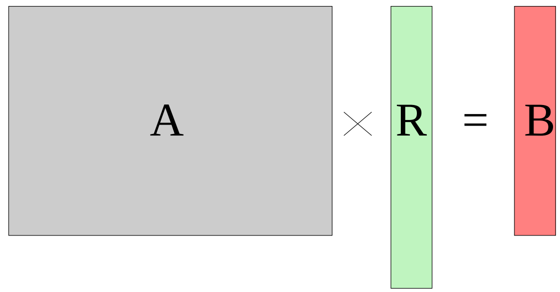
Denoted by $S(\alpha, d)$, where $0 < \alpha \leq 2$.

Two random variables $Z_1 \sim S(\alpha, 1)$ and $Z_2 \sim S(\alpha, 1)$.

For any constants C_1 and C_2

$$Z = C_1 \times Z_1 + C_2 \times Z_2 \sim S\left(\alpha, |C_1|^\alpha + |C_2|^\alpha\right)$$

For example, weighted sum of normals is also normal ($\alpha = 2$).



Therefore, the projected matrix $\mathbf{B}$ contains information about

1. αth moment, $\sum_{i=1}^{D} |u_i|^{\alpha}$, of each row of $\mathbf{A}$.
2. αth distance, $\sum_{i=1}^{D} |u_i - v_i|^{\alpha}$, between any two rows of $\mathbf{A}$.

Applications of Symmetric Stable Random Projections

- Data visualization algorithms

  Multi-dimensional scaling (MDS) requires a pairwise similarity matrix.

- Machine Learning algorithms

  SVM (support vector machine) requires a $O(n^2)$ pairwise distance matrix.

- Information retrieval

  Finding (filtering) nearly duplicate docs (often measured by distance)

- Databases

  Estimating join sizes (dot products) for optimizing query execution.

- Dynamic data stream computations

  Estimating summary statistics for visualizing/detecting anomaly real-time

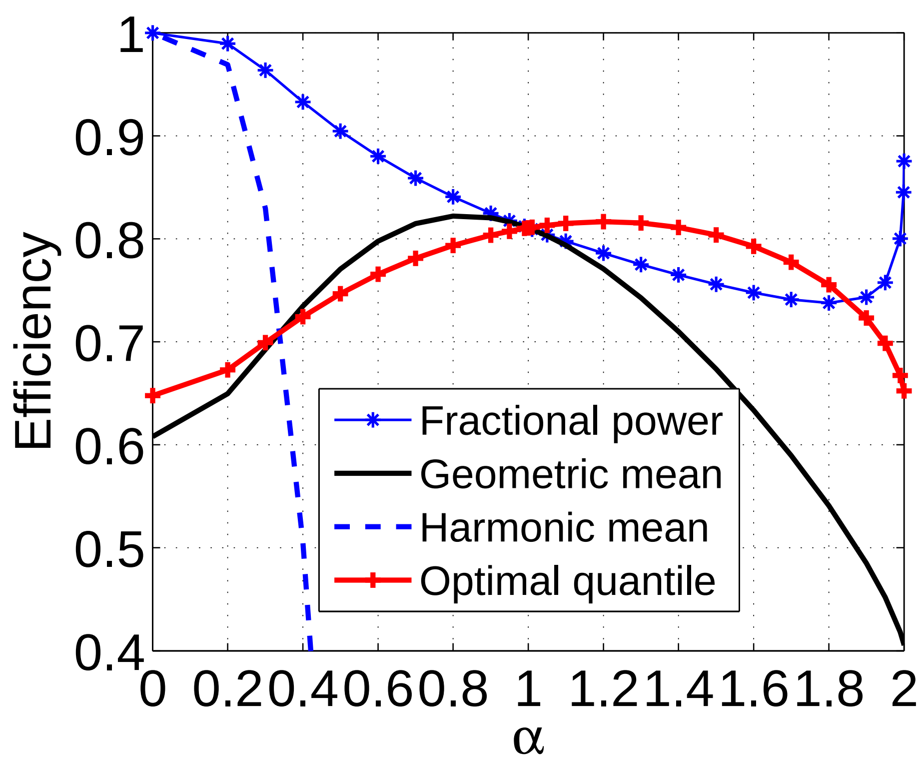
Recent Progress in Symmetric Stable Random Projections

- After random projections, the task boils down to estimating the scale parameter from stable samples: x_j, $j = 1$ to k.
- An ideal estimator: (1) accurate ($\Longrightarrow$ small k) (2) computationally efficient.
- Previous estimators were expensive: (1) geometric mean; (2) harmonic mean; (3) fractional power.
- The optimal quantile estimator is both accurate and computationally efficient.
  Ping Li, IEEE ICDM 2008

Cramér-Rao Efficiencies (Accuracies)

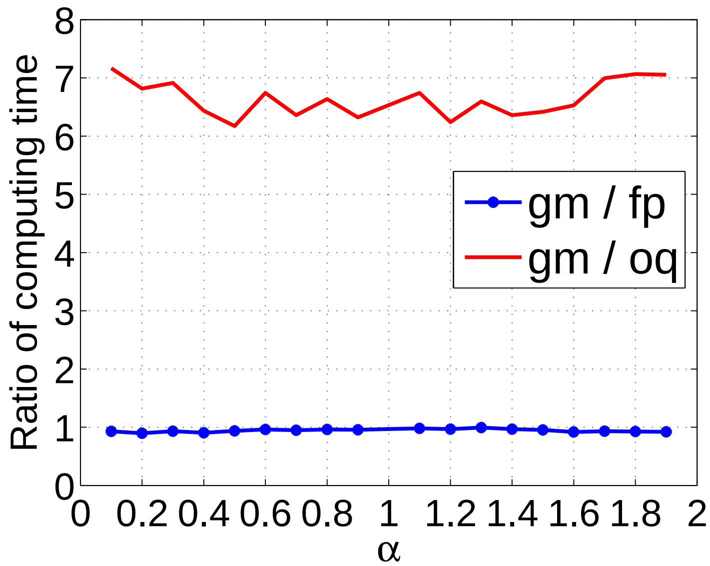
Cramér-Rao efficiencies (the higher the better, max = 1.00) of various estimators.

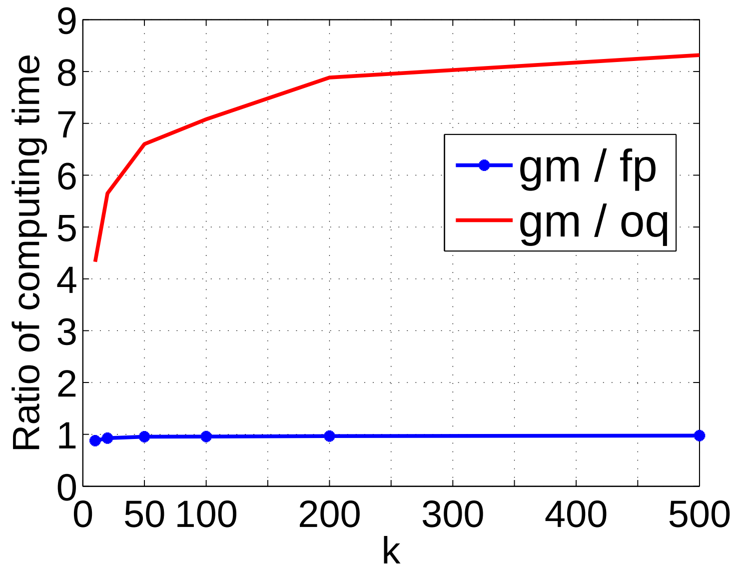
The optimal quantile estimator is competitive in terms of accuracy.



Computational Efficiencies (Speeds)

The optimal quantile (oq) estimator is a magnitude more computationally efficient.





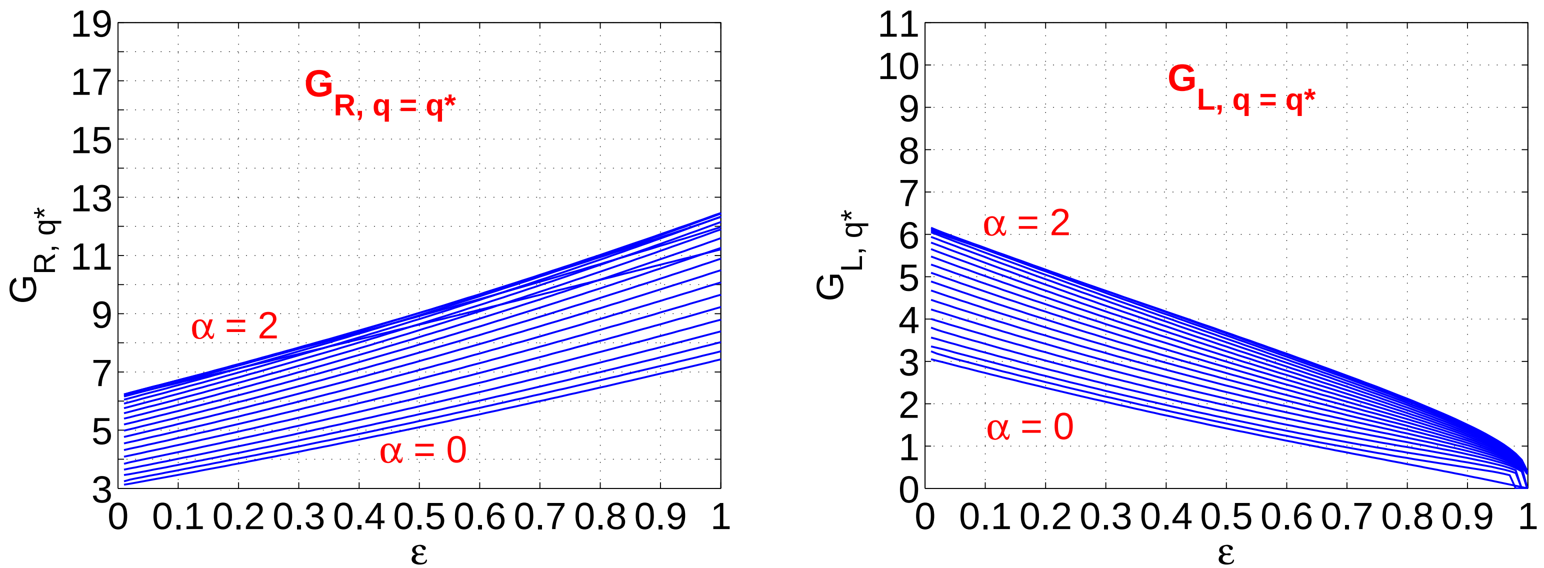
Tail Bounds (Performance Guarantee)

Denote the optimal quantile estimator by $\hat{d}_{(\alpha),oq}$ and the true value by $d_{(\alpha)}$.

$$\mathbf{Pr}\left(\hat{d}_{(\alpha),oq} \geq (1+\epsilon)d_{(\alpha)}\right) \leq \exp\left(-k\frac{\epsilon^2}{G_R}\right), \epsilon > 0,$$

$$\mathbf{Pr}\left(\hat{d}_{(\alpha),oq} \leq (1-\epsilon)d_{(\alpha)}\right) \leq \exp\left(-k\frac{\epsilon^2}{G_L}\right), 0 < \epsilon < 1,$$

The constants, G_R and G_L are complicated but they can be easily plotted.



The required sample size (number of projections):

Using $\hat{d}_{(\alpha),oq}$ with $\boxed{k \geq \frac{G}{\epsilon^2}\left(2\log n - \log\delta\right)}$, any pairwise l_α distance among n points can be approximated within a $1 \pm \epsilon$ factor with probability $\geq 1 - \delta$.

Conditional Random Sampling (CRS)

The method of random projections exhibits many weaknesses:

- Didn't consider data sparsity; but large-scale datasets are often highly sparse.
- Could only work for the l_α distance for a particular α.

Conditional Random Sampling (CRS) partially overcomes those weaknesses:

- Designed specifically for sparse data.
- One-sketch-for-all: the same sample is re-used, not just for l_α distances.
- Not necessarily less accurate than random projections.
- Already applied in industry.
- Recent progress: *Ping Li et. al., NIPS 2008*.

Compressed Counting (CC)

- Applicable to dynamic data streams following strict-Turnstile model.
- Achieving an "infinite" improvement over symmetric projections when $\alpha \approx 1$.
- Applications in estimating entropy real-time for network anomaly detections.
- Papers: *Li SODA 2009, Li UAI 2009* .

Turnstile Data Stream Model

At time t, an incoming element : $\boxed{a_t = (i_t, I_t)}$

$i_t \in [1, D]$ index, I_t: increment/decrement.

Updating rule : $\boxed{A_t[i_t] = A_{t-1}[i_t] + I_t}$

Goal : Count αth moment $F_{(\alpha)} = \sum_{i=1}^{D} A_t[i]^\alpha$

Strict-Turnstile model : $A_t[i] \geq 0$ always, suffices for almost all applications.

For example, the strict-Turnstile model for an online bookstore

t=0

0	0	0	0	0	0	....	0
IP 1	IP 2	IP 3	IP 4			....	IP D

t=1 arriving stream = (3, 10) user 3 ordered 10 books

0	0	10	0	0	0	....	0
IP 1	IP 2	IP 3	IP 4			....	IP D

t=2 arriving stream = (1, 5) user 1 ordered 5 books

5	0	10	0	0	0	....	0
IP 1	IP 2	IP 3	IP 4			....	IP D

t=3 arriving stream = (3, −8) user 3 cancelled 8 books

5	0	2	0	0	0	....	0
IP 1	IP 2	IP 3	IP 4			....	IP D

Counting: Trivial if $\alpha = 1$, but Non-trivial in General

Goal : Count $F_{(\alpha)} = \sum_{i=1}^{D} A_t[i]^\alpha$, where $\boxed{A_t[i_t] = A_{t-1}[i_t] + I_t}$.

When $\alpha \neq 1$, counting $F_{(\alpha)}$ exactly requires D counters. (but D can be 2^{64})

When $\alpha = 1$, however, counting the sum is trivial, using a simple counter.

$$F_{(1)} = \sum_{i=1}^{D} A_t[i] = \sum_{s=1}^{t} I_s,$$

Compressed Counting (CC) captures this intuition

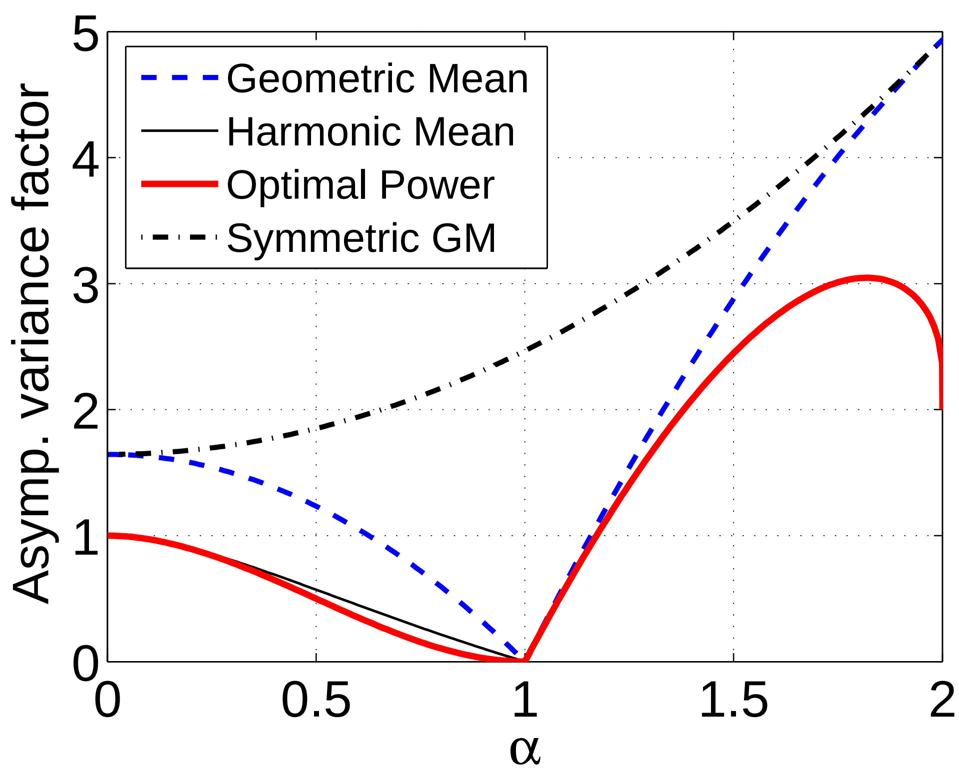
Symmetric stable random projections totally ignore this fact.

Dramatic Variance Reduction

Symmetric GM: the estimator for symmetric stable projections in *Li, SODA 2008*.

Harmonic and geometric means: estimators for CC introduced in *Li, SODA 2009*.

Optimal Power: the estimator for CC introduced in *Li, UAI 2009*.



Estimating Shannon Entropy Using Moments

Shannon entropy is widely used, eg., in Web and networks:

$$H = -\sum_{i=1}^{D} \frac{A_t[i]}{F_{(1)}} \log \frac{A_t[i]}{F_{(1)}}, \qquad F_{(1)} = \sum_{i=1}^{D} A_t[i]$$

Shannon entropy may be approximated by Rényi entropy:

$$H_\alpha = \frac{1}{1-\alpha} \log \frac{\sum_{i=1}^{D} A_t[i]^\alpha}{\left(\sum_{i=1}^{D} A_t[i]\right)^\alpha}$$

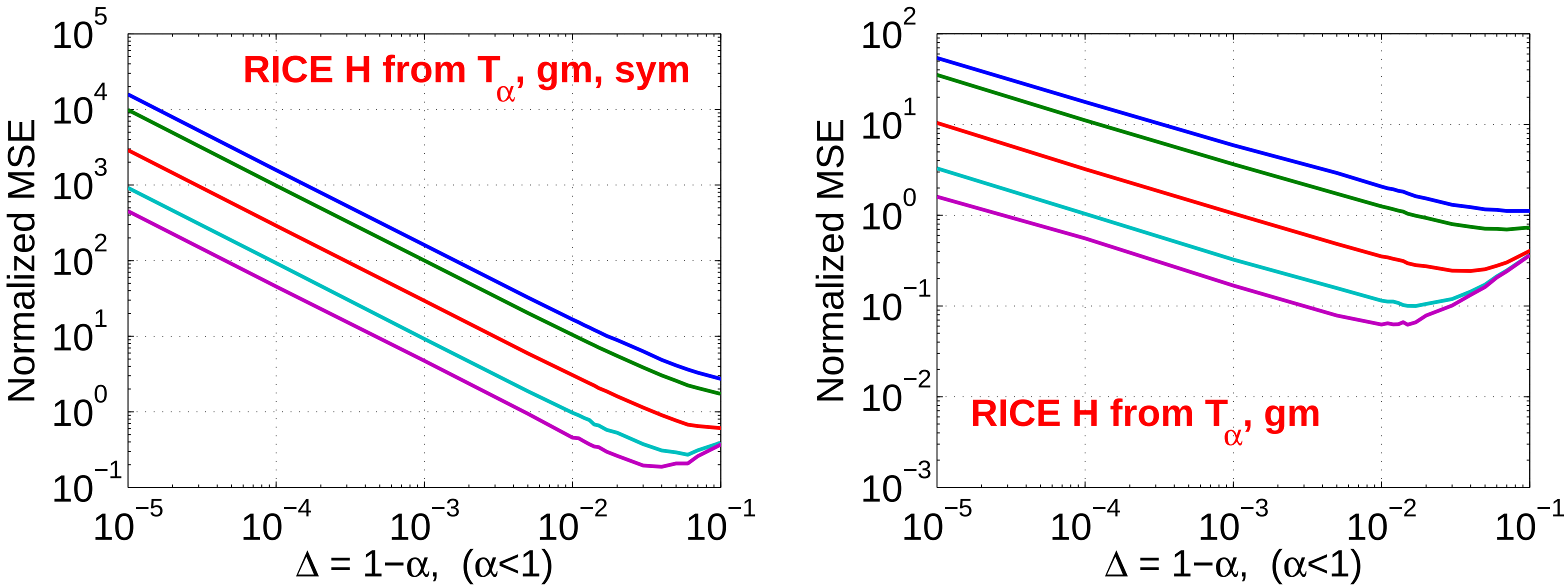
or Tsallis entropy:

$$T_\alpha = \frac{1}{\alpha - 1}\left(1 - \frac{F_{(\alpha)}}{F_{(1)}^\alpha}\right).$$

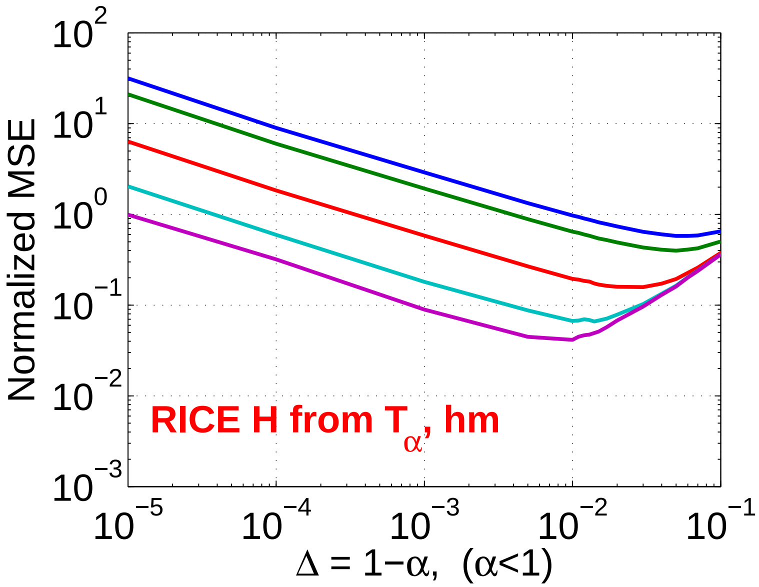
$$\lim_{\alpha\rightarrow 1} H_\alpha = \lim_{\alpha\rightarrow 1} T_\alpha = H, \quad \text{as } \alpha \rightarrow 1$$

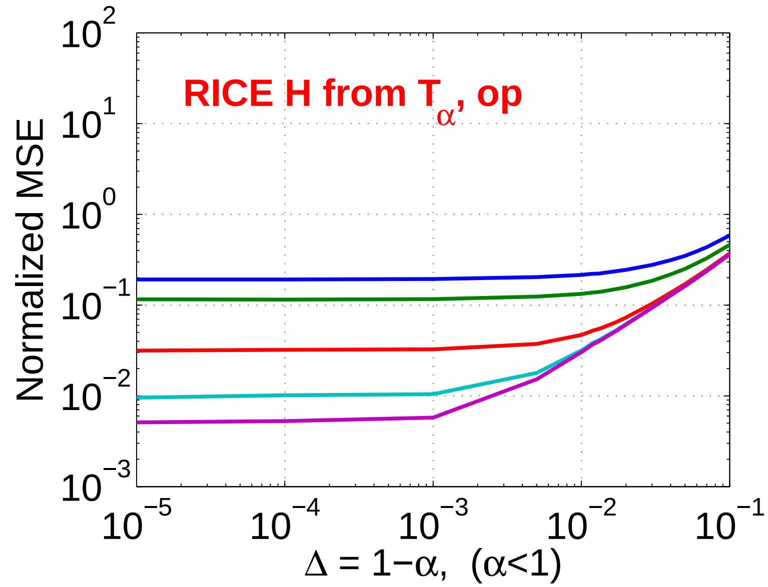
Normalized MSE for Estimating Entropy

(Symmetric) Stable Random Projections CC



- Sample size $k = 5$, 10, 100, 1000, 4000, from top to bottom.
- CC significantly improves symmetric stable projections.
- The geometric mean (gm) estimator is not good.





The *optimal power (op)* estimator is a truly practical algorithm for entropy estimation.

Boosting (Tree) Algorithms For Classification

- Classification is a one of the most basic tasks in machine learning.

- We developed Adaptive Base Class (ABC) Boost and implemented it using Friedman's classical MART algorithm: $\Longrightarrow$ ABC-MART. *Ping Li, ICML 2009*

- We are developing Robust Logitboost, which provides a stable implementation of logitboost (Friedman et. al. 2000)

- We are also developing ABC-LogitBoost.

- Many new directions have been identified and will be exploited.

- *This line of work was not included in the original proposal; we are actively seeking funding to continue this project*.

An Empirical Study for Classification

- MART, ABC-MART, Robust LogitBoost, ABC-LogitBoost, on large datasets.

- Comparisons with SVM are available. For example, SVM achieved $< 60\%$ classification accuracy on UCI-Poker datasets while we obtained $> 90\%$.

- Comparisons with Deep Learning are also available; ours are competitive.

Table 1: Datasets for multi-class Classification

dataset	# Classes	# training	# test	# features
Covertype	7	290506	290506	54
Poker525k	10	525010	500000	10
PokerT1	10	25010	500000	10
PokerT2	10	25010	500000	10
Mnist10k	10	10000	60000	784
M-Basic	10	12000	50000	784
M-Rotated	10	12000	50000	784
M-Image	10	12000	50000	784
M-Rand	10	12000	50000	784
M-RotImg	10	12000	50000	784
Letter4k	26	4000	16000	16
Letter2k	26	2000	18000	16

Table 2: Summary of test mis-classification errors (smaller is better).

Dataset	**mart**	**abc-mart**	**robust logitboost**	**abc-logitboost**
Covertype	11350	10454	10765	9727
Poker525k	7061	2424	2704	1736
PokerT1	43575	34879	46789	37345
PokerT2	42935	34326	46600	36731
Mnist10k	2815	2440	2381	2102
M-Basic	2058	1843	1723	1602
M-Rotated	7674	6634	6517	5959
M-Image	5821	4727	4703	4268
M-Rand	6577	5310	5020	4711
M-RotImg	24912	23072	22962	22343
Letter4k	1370	1149	1252	1055
Letter2k	2482	2220	2309	2034

Table 3: Error rates of various algorithms (including SVM and Deep Learning). www.iro.umontreal.ca/∼lisa/twiki/bin/view.cgi/Public/DeepVsShallowComparisonICML2007 (Note that, we simply fixed our base learner tree-size to be 20).

	M-Basic	M-Rotated	M-Image	M-Rand	M-RotImg
SVM-RBF	**3.05%**	11.11%	22.61%	14.58%	55.18%
SVM-POLY	3.69%	15.42%	24.01%	16.62%	56.41%
NNET	4.69%	18.11%	27.41%	20.04%	62.16%
DBN-3	3.11%	**10.30%**	16.31%	**6.73%**	47.39%
SAA-3	3.46%	**10.30%**	23.00%	11.28%	51.93%
DBN-1	3.94%	14.69%	16.15%	9.80%	52.21%
MART	4.12%	15.35%	11.64%	13.15%	49.82%
ABC-MART	3.69%	13.27%	9.45%	10.62%	46.14%
Robust LogitBoost	3.45%	13.03%	9.41%	10.04%	45.92%
ABC-LoigitBoost	3.20%	11.92%	**8.54%**	9.42%	**44.69%**

Practical Advantages

MART, ABC-MART, Robust LogitBoost, ABC-LogitBoost are well suited for industry applications:

- Few parameters. Performance is not sensitive to parameters; tuning is easy.
- No need to clean, normalize, kernelize the data.
- Easily scaling up to millions of samples.
- Not affected by irrelative features, automatically doing variable selections.
- Friedman's MART algorithm has been widely used in industry.