NSF-FODAVA: Efficient Data Reduction and Summarization

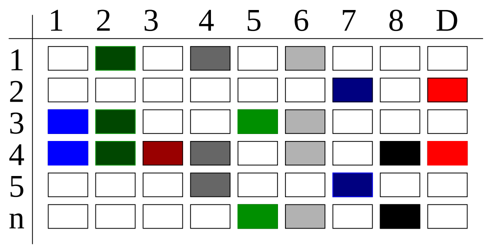
PI: Ping Li, Cornell University, 09/01/2008 - 08/31/2011

Deliverables: The following papers have acknowledged this support (the PI's only NSF grant).

1. P. Li, C. König, W. Gui, b-Bit Minwise Hashing for Estimating Three-Way Similarities, **NIPS 2010**
2. P. Li, Robust LogitBoost and Adaptive Base Class (ABC) LogitBoost, **UAI 2010**
3. P. Li, M. Mahoney, Y. She, Approximating Higher-Order Distances Using Random Projections, **UAI 2010**
4. P. Li, C. König, b-Bit Minwise Hashing, **WWW 2010**
5. F. Wang, P. Li, Efficient Nonnegative Matrix Factorization with Random Projections, **SDM 2010**
6. F. Wang, P. Li, Compressed Non-negative Sparse Coding, **ICDM 2010**
7. F. Wang, P. Li, C. König, Learning a Bi-Stochastic Data Similarity Matrix, **ICDM 2010**
8. P. Li, ABC-Boost: Adaptive Base Class Boost for Multi-Class Classification, **ICML, 2009**
9. P. Li, Compressed Counting, **SODA 2009**
10. P. Li, Improving Compressed Counting, **UAI 2009**
11. P. Li, Computationally Efficient Estimators for Dimension Reductions Using Stable Random Projections, **ICDM 2008**
12. P. Li, K Church, T. Hastie, One Sketch for All: Theory and Application of Conditional Random Sampling, **NIPS 2008**

Objective: “Shrinking” Massive Data

Data Matrix $\mathbf{A} \in \mathbb{R}^{n\times D}$: n rows and D columns, e.g., term-doc, image-pixel.



Characteristics of Modern Massive Data Sets (MMDS)

- **Massive**, e.g., n, $D \approx 10^{10}$, or even 2^{64}.
- Often **Dynamic**, e.g., data streams, $\mathbf{A}_t[i_t] = \mathbf{A}_{t-1}[i_t] + fun(i_t, I_t)$
- Often **Sparse**, e.g., text data, or some representations of image data
- Many applications only need **summary statistics**. For example, clustering uses distances, linear regression $\left(\mathbf{A}^{\mathsf{T}}\mathbf{A}\right)^{-\mathbf{1}}\mathbf{A}^{\mathsf{T}}\mathbf{Y}$ uses inner products.
- **Challenges**: store and transmit data; compute & maintain summary statistics

Computing Summary Statistics in Massive Data

Take first two rows of $\mathbf{A}$: $u_1, u_2 \in \mathbb{R}^D$. Many applications, e.g., machine learning and visualization, requires computing various summary statistics:

- **Distances**: Euclidean $d_2 = \sum_{i=1}^D |u_{1,i} - u_{2,i}|^2$;
  Manhattan $d_1 = \sum_{i=1}^D |u_{1,i} - u_{2,i}|$. L_p distance $d_p = \sum_{i=1}^D |u_{1,i} - u_{2,i}|^p$;
- **Inner product**: $a = \sum_{i=1}^D u_{1,i}u_{2,i}$; **Correlation**: $\rho = \frac{a}{\sqrt{\sum_{i=1}^D u_{1,i}^2 \sum_{i=1}^D u_{2,i}^2}}$.
- **Chi-Square**: $d_{\chi^2} = \sum_{i=1}^D \frac{|u_{1,i}-u_{2,i}|^2}{u_{1,i}+u_{2,i}}$.; **General** $d_g = \sum_{i=1}^D g(u_{1,i}, u_{2,i})$.
- **Multi-way association**: $\sum_{i=1}^D u_{1,i}u_{2,i}u_{3,i}$.

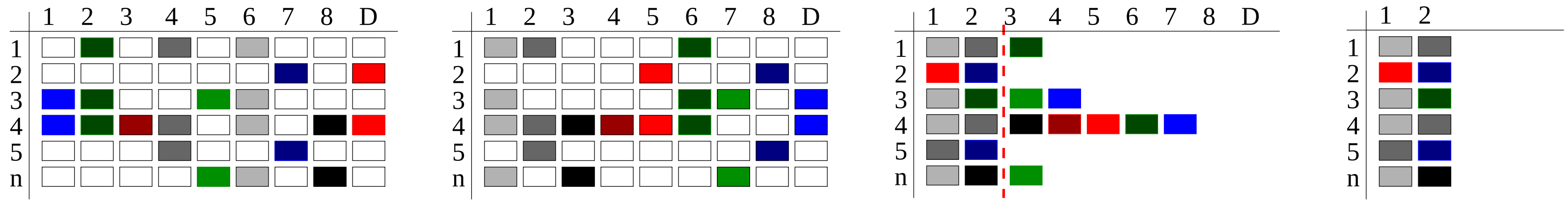
Challenges: Computationally expensive; massive storage; dynamic data.

Data Reduction Methods (PI has worked on)

- **Normal random projection** for efficiently computing the l_2 distances and inner products, applicable to dynamic data. Recently, we extend it to computing the l_p distances, for $p = 4, 6, 8...$
- **Cachy random projection** for computing the l_1 distances.
- **Stable random projection** for computing the l_p distances, $0 < p \leq 2$.
- **Compressed Counting**, a breakthrough in data stream computations, for computing the p-th frequency moments and Shannon entropy.
- **b-Bit Minwise Hashing**, for improving the conventional minwise hashing often by > 20-fold. Since minwise hashing is the standard tool in the context of search industry, this work has attracted good attention.
- **Conditional Random Sampling (CRS)**, a new technique for general sampling. Not in the poster presentation.

Conditional Random Sampling (CRS): One Sketch for All

Sparse Matrix — Random Permutation on Columns — Inverted Index (Nonzeros) — Sketches



Estimating procedure: Basically a trick (although finding it was a long process)

Random sample of size 10

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
u_1	0	3	0	2	0	1	0	0	1	2	1	0	1	0	2	0
u_2	1	0	0	0	1	2	0	1	0	0	3	0	0	2	1	1

Sketches of size 5

P_1:	2 (3)	4 (2)	6 (1)	9 (1)	10 (2)	11 (1)	13 (1)	15 (2)
P_2:	1 (1)	5 (1)	6 (2)	8 (1)	11 (3)	14 (2)	15 (1)	16(1)

Excluding 11(3) from sketches, two schemes are equivalent (for u_1 and u_2) conditioning on $D_s = \min(10, 11) = 10$. (Rigorous theory says $D_s = 10 - 1$)

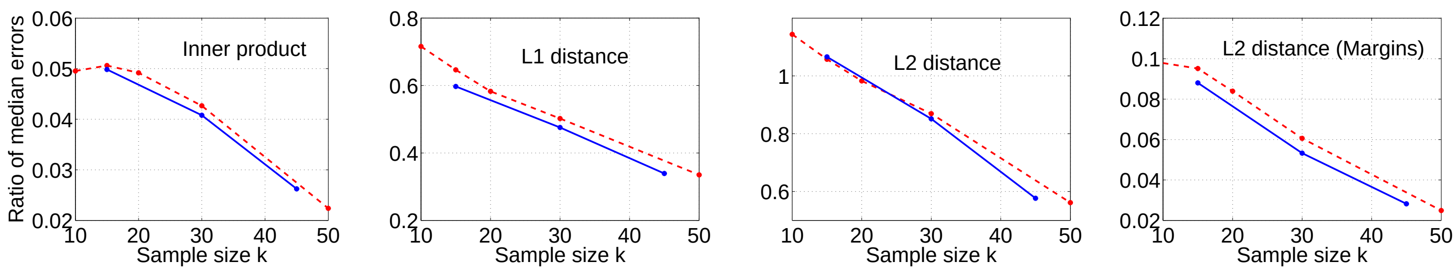
For another pair, e.g., u_1 and u_3, the (retrospective) sample D_s may be different. Also, this scheme works for more than two rows, and for dynamic streaming data.

Once there is a random sample, estimating any summary statistics is trivial, based on the same sketches. Thus, CRS is one-sketch-for-all.

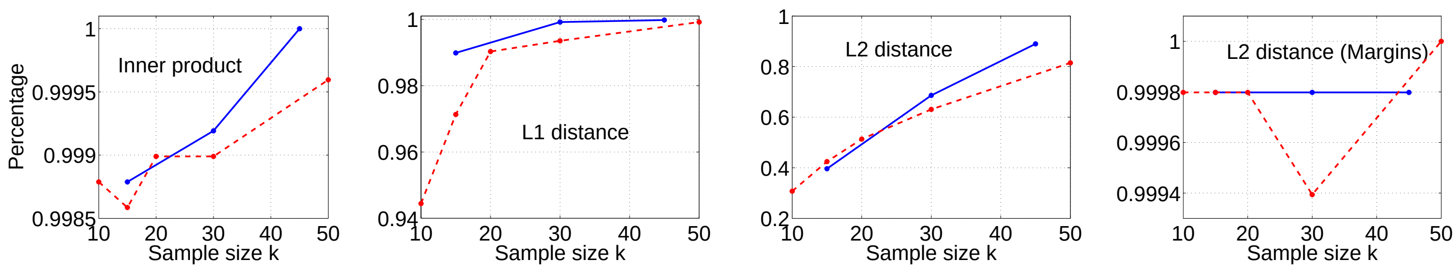
Comparisons with Random Projections

- **CRS is much more versatile**. Random projection is not one-sketch-for-all and only applicable to limited summary statistics.
- **CRS is more efficient**, since only one permutation is needed.
- **CRS can be less accurate** when the data are dense and/or heavy-tailed.
- **CRS is more accurate** if the data are sparse, binary, or nearly independent.

Values < 1 indicate CRS is more accurate



Percentage of data pairs for which CRS is more accurate



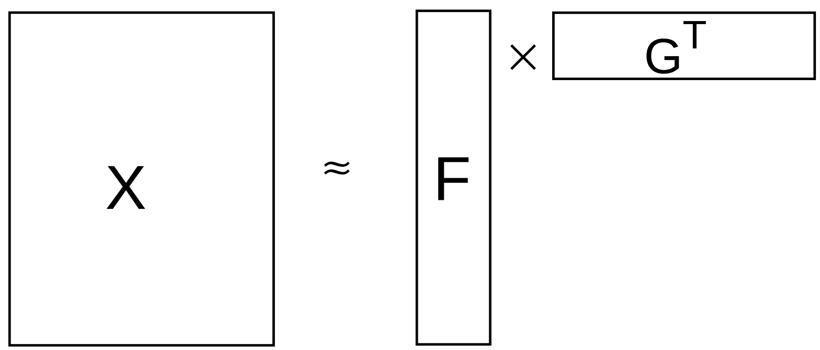
References for CRS

1. Ping Li, Kenneth Church, and Trevor Hastie, *One Sketch for All: Theory and Application of Conditional Random Sampling*, NIPS 2008
2. Ping Li, Kenneth Church, and Trevor Hastie, *Conditional Random Sampling: A Sketch-Based Sampling Technique for Spare Data*, NIPS 2006
3. Ping Li and Kenneth Church, *A Sketch Algorithm for Estimating Two-Way and Multi-Way Associations*, Computational Linguistics 2007
4. Ping Li and Kenneth Church, *Using Sketches to Estimate Associations*, EMNLP/HLT 2005

Efficient Matrix Factorization and Sparse Coding Using Random Projections

Fei Wang, Ping Li, Cornell University

Non-Negative Matrix Factorization (NMF) has many applications in machine learning and data mining including Vision, information retrieval and bioinformatics.



Approximate a non-negative data matrix $\mathbf{X} \in \mathbb{R}^{d\times n}$ by $\mathbf{X} \approx \mathbf{F}\mathbf{G}^{\mathsf{T}}$, $\mathbf{F} \in \mathbb{R}^{d\times r}$, $\mathbf{G} \in \mathbb{R}^{n\times r}$, by minimizing the loss in the matrix Frobenius norm:

$$J(\mathbf{F}, \mathbf{G}) = \left\|\mathbf{X} - \mathbf{F}\mathbf{G}^T\right\|_F^2,$$

subject to the non-negativity constraint: $F_{ij} \geq 0$, $G_{ij} \geq 0$.

Traditional Solutions to NMF and the Challenges

Lee and Seung's multiplicative updating rule: Starting with some (random) initialization of $\mathbf{F}$ and $\mathbf{G}$, repeat the following steps:

$$\mathbf{G}_{ij} \longleftarrow \mathbf{G}_{ij}\frac{\left(\mathbf{X}^T\mathbf{F}\right)_{ij}}{\left(\mathbf{G}\mathbf{F}^T\mathbf{F}\right)_{ij}}, \qquad \mathbf{F}_{ij} \longleftarrow \mathbf{F}_{ij}\frac{\left(\mathbf{X}\mathbf{G}\right)_{ij}}{\left(\mathbf{F}\mathbf{G}^T\mathbf{G}\right)_{ij}}.$$

Since then, many algorithms have been developed (e.g., in H. Park's group).

Fundamental challenges: Computationally intensive when $\mathbf{X}$ is too large. Infeasible to store the data matrix $\mathbf{X}$ in the memory in large applications.

Will random projections (RP) work?: Replacing $\mathbf{X}$ by $\mathbf{R}\mathbf{X}$, where entries of $\mathbf{R}$ are sampled from $N(0,1)$, violates the non-negativity of $\mathbf{X}$. What can we do?

Dual RP via semi-NMF: Alternatingly solve two semi-NMF problems on $\widetilde{\mathbf{X}}_d = \widetilde{\mathbf{R}}_d\mathbf{X}$ and $\widetilde{\mathbf{X}}_n = \mathbf{X}\widetilde{\mathbf{R}}_n^T$. Semi-NMF only imposes non-negativity on one of $\mathbf{F}$ and $\mathbf{G}$.

Dual Random Projections Via Semi-NMF

Semi-NMF multiplicative update rule: Generate two random matrices, $\widetilde{\mathbf{R}}_d \in \mathbb{R}^{k_1 \times d}$ and $\widetilde{\mathbf{R}}_n \in \mathbb{R}^{k_2 \times d}$, whose entries are i.i.d. $N(0,1)$. Repeat:

$$\mathbf{G}_{ij} \longleftarrow \mathbf{G}_{ij}\sqrt{\frac{(\widetilde{\mathbf{X}}_d^T\widetilde{\mathbf{F}})_{ij}^+ + [\mathbf{G}(\widetilde{\mathbf{F}}^T\widetilde{\mathbf{F}})^-]_{ij}}{(\widetilde{\mathbf{X}}_d^T\widetilde{\mathbf{F}})_{ij}^- + [\mathbf{G}(\widetilde{\mathbf{F}}^T\widetilde{\mathbf{F}})^+]_{ij}}}, \qquad \mathbf{F}_{ij} \longleftarrow \mathbf{F}_{ij}\sqrt{\frac{(\widetilde{\mathbf{X}}_n\widetilde{\mathbf{G}})_{ij}^+ + [\mathbf{F}(\widetilde{\mathbf{G}}^T\widetilde{\mathbf{G}})^-]_{ij}}{(\widetilde{\mathbf{X}}_n\widetilde{\mathbf{G}})_{ij}^- + [\mathbf{F}(\widetilde{\mathbf{G}}^T\widetilde{\mathbf{G}})^+]_{ij}}}$$

where $\widetilde{\mathbf{X}}_d = \widetilde{\mathbf{R}}_d\mathbf{X}$, $\widetilde{\mathbf{X}}_n = \mathbf{X}\widetilde{\mathbf{R}}_n^T$, $\widetilde{\mathbf{F}} = \widetilde{\mathbf{R}}_d\mathbf{F}$, $\widetilde{\mathbf{G}} = \widetilde{\mathbf{R}}_n\mathbf{G}$.

(Note that when the data are non-negative, using the square-root update slows down convergence.)

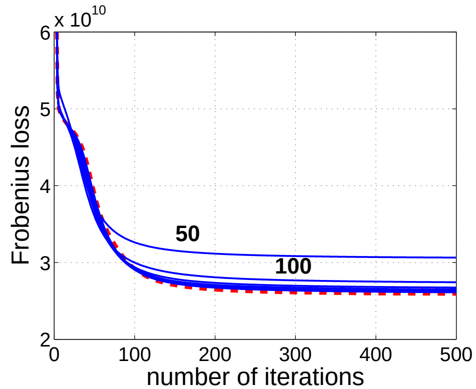
We have also implemented dual RP semi-NMF using other methods such as *active set* and *projected gradient*.

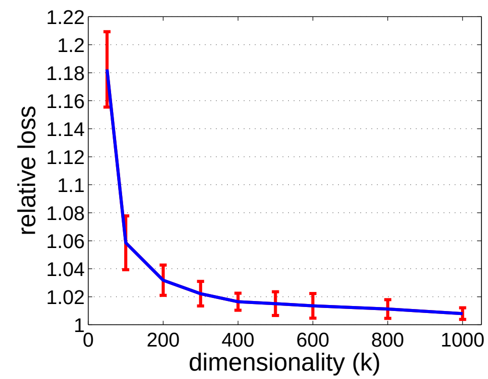
Table 1: Data set information for NMF experiments

Name	Dimension (d)	Size (n)	# Class
Microarray	12600	203	5
Gisette	5000	6000	2
COIL	16384	7200	100

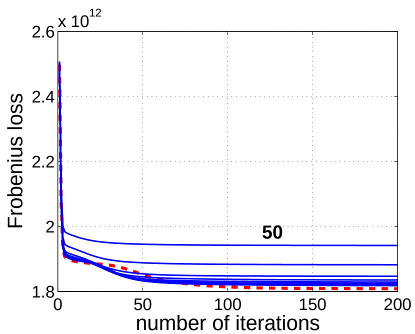
NMF with Random Projections Experiments

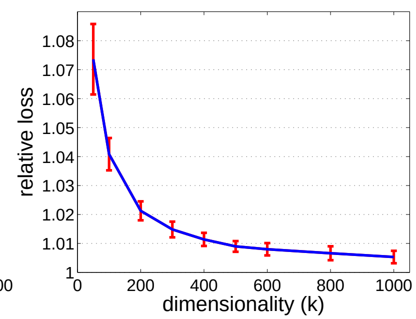
Microarray: Loss for projection size $k = 50$ to $k = 1000$.



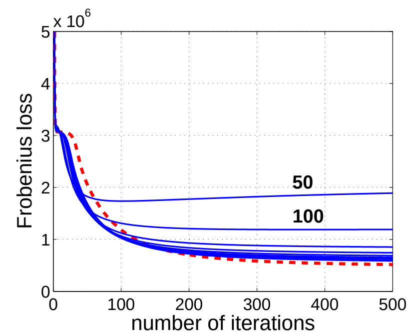


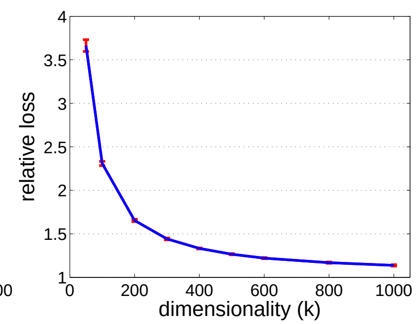
Gisette





Coil





Observations: with projection dimension $k \geq 500$, the accuracy is satisfactory (often within 1% errors), essentially independent of the original data matrix size.

Non-Negative Sparse Coding (NSC)

$\mathbf{X} = [\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n] \in \mathbb{R}^{d\times n}$, Basis matrix: $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_r] \in \mathbb{R}^{d\times r}$

Combination coefficient matrix: $\mathbf{G} = [\mathbf{g}_1, \mathbf{g}_2, \cdots, \mathbf{g}_n] \in \mathbb{R}^{r\times n}$

Approximate $\mathbf{X} \approx \mathbf{FG}$ by solving an optimization problem:

$$\min_{\mathbf{F},\mathbf{G}} \quad \sum_i^n \left\|\mathbf{x}_i - \mathbf{F}\mathbf{g}_i\right\|^2 + \lambda \left|\mathbf{g}_i\right|_1, \qquad s.t.\ \mathbf{F} \geqslant 0,\ \mathbf{G} \geqslant 0$$

Alternating optimization

1. Fix $\mathbf{F}$. Solve n independent ℓ_1 constrained (Lasso) optimization problems:

$$\min_{\mathbf{g}_i} \quad \left\|\mathbf{x}_i - \mathbf{F}\mathbf{g}_i\right\|^2 + \lambda \left|\mathbf{g}_i\right|_1, \qquad s.t. \mathbf{g}_i \geqslant 0, \qquad i = 1, 2, \cdots, n$$

2. Fix $\mathbf{G}$. Solve the following problem

$$\min_{\mathbf{F}} \quad \sum_i^n \left\|\mathbf{x}_i - \mathbf{F}\mathbf{g}_i\right\|^2 = \left\|\mathbf{X} - \mathbf{FG}\right\|_F^2, \qquad s.t.\ \mathbf{F} \geqslant 0$$

Solve NSC via Random Projections (Compressed NSC)

Solving $\mathbf{G}$ with $\mathbf{F}$ Fixed

$$\min_{\mathbf{g}_i} \quad \left\|\mathbf{R}_d\mathbf{x}_i - \mathbf{R}_d\mathbf{F}\mathbf{g}_i\right\|^2 + \lambda\left|\mathbf{g}_i\right|_1, \qquad s.t.\ \mathbf{g}_i \geqslant 0$$

where $\mathbf{R}_d \in \mathbb{R}^{k_d\times d}$ is a random matrix whose entries are sampled from i.i.d. $N(0.1)$. This is still a standard (non-negative) Lasso problem.

Solving $\mathbf{F}$ with $\mathbf{G}$ Fixed

$$\min_{\mathbf{F}} \quad \left\|\mathbf{X}\mathbf{R}_n - \mathbf{F}\mathbf{G}\mathbf{R}_n\right\|_F^2, \qquad s.t.\ \mathbf{F} \geqslant 0$$

which is solved by a semi-NMF-like updating rule:

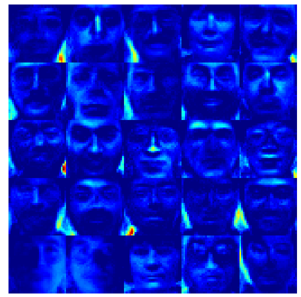
$$\mathbf{F} \longleftarrow \mathbf{F} \odot \sqrt{\frac{\mathbf{\Gamma}_+ + \mathbf{F}\mathbf{\Theta}_- + \mathbf{F}\text{diag}\left[\mathbf{1}^\mathsf{T}((\mathbf{\Gamma}_- + \mathbf{F}\mathbf{\Theta}_+)\odot\mathbf{F})\right]}{\mathbf{\Gamma}_- + \mathbf{F}\mathbf{\Theta}_+ + \mathbf{F}\text{diag}\left[\mathbf{1}^\mathsf{T}((\mathbf{\Gamma}_+ + \mathbf{F}\mathbf{\Theta}_-)\odot\mathbf{F})\right]}}$$

where

$$\mathbf{\Gamma} = \quad \mathbf{X}\mathbf{R}_n\mathbf{R}_n^\mathsf{T}\mathbf{G}^\mathsf{T}, \qquad \mathbf{\Theta} = \quad \mathbf{G}\mathbf{R}_n\mathbf{R}_n^\mathsf{T}\mathbf{G}^\mathsf{T}$$

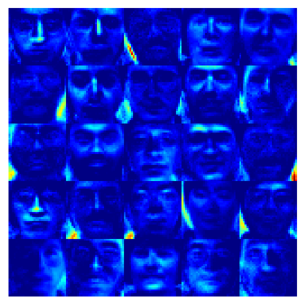
Experiments of Compressed NSC (CNSC)

The learned dictionary (base matrix) on Yale face data.

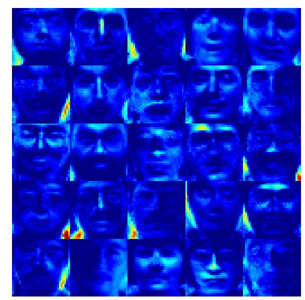


Original

k_d=50



k_d=500



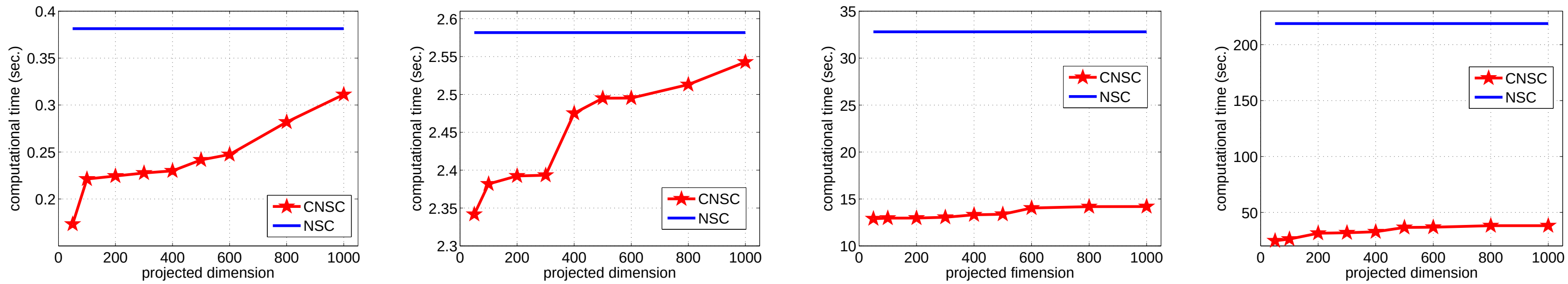
k_d=1000

Data sets	Dimensionality (d)	Size (n)
Yale	1024	165
YaleB	1024	2,124
COIL	16384	7,200
PIE	1024	11,554
SecStr	315	1,273,151

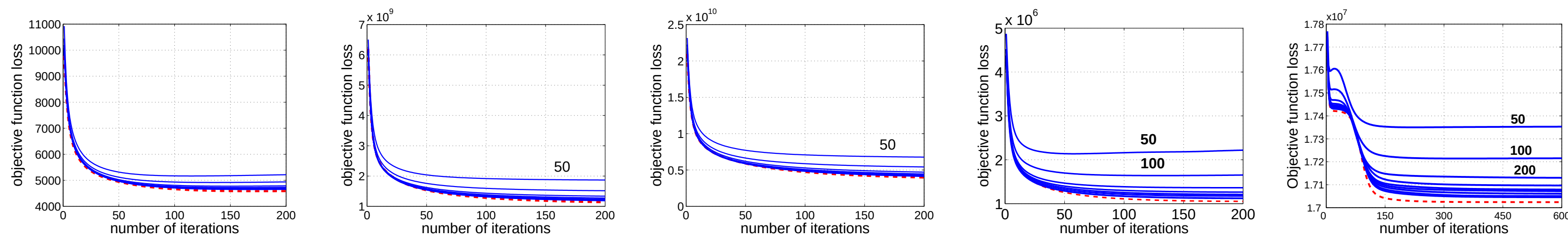
Experiments of Compressed NSC (CNSC)

Computational time comparisons: The larger the data set, the more saving.

Yale **YaleB** **PIE** **COIL**



Accuracy comparisons: Normally $k \geq 500$ can provide accurate solutions.



Yale **YaleB** **PIE** **COIL** **SecStr**

References for NMF and Sparse Coding

1. Fei Wang and Ping Li, *Efficient Non-Negative Matrix Factorization with Random Projections*, SDM 2010

2. Fei Wang and Ping Li, *Compressed Non-Negative Sparse Coding*, ICDM 2010