IIT PALAKKAD

Sketching Discrete Valued Sparse Matrices

IEEE GlobalSIP 2018

Lakshmi Narasimhan Theagarajan

29 November 2018

Indian Institute of Technology Palakkad

Outline

1. Matrix Sketching
2. Matching Pursuit based Matrix Sketching
3. Constrained Message Passing
4. Numerical Results
5. Conclusions

Matrix Sketching

Introduction

Matrix sketching

Recovery of a large dimensional sparse matrix $\mathbf{X}$ from its sketch $\mathbf{Y}$.

- The sketch $\mathbf{Y} \triangleq \mathbf{A}\mathbf{X}\mathbf{B}^T$ is a low dimensional linear observation.
- The sketching (or measurement) matrices $\mathbf{A}$ and $\mathbf{B}$ are known and satisfy restricted isometry property (RIP).
- The matrices $\mathbf{A}$ and $\mathbf{B}$ are of dimension $L \times M$ and $P \times N$, respectively.
- Typically, $\max(L, P) \ll \min(M, N)$.

Applications

- Image processing.
- MIMO communication and error control coding.
- Random graph identification.

Classical sparse recovery

Formally, the matrix sketching problem can be stated as

$$\min_{\widehat{\mathbf{X}} \in \mathbb{R}^{M \times N}} \|\widehat{\mathbf{X}}\|_1 \quad \text{s.t. } \mathbf{A}\widehat{\mathbf{X}}\mathbf{B}^T = \mathbf{Y}. \tag{1}$$

By vectorizing the matrices $\mathbf{X}$ and $\mathbf{Y}$, this problem can be reduced to the compressed sensing framework as

$$\min_{\widehat{\mathbf{x}} \in \mathbb{R}^{MN}} \|\widehat{\mathbf{x}}\|_1 \quad \text{s.t. } \mathbf{C}\widehat{\mathbf{x}} = \mathbf{y}. \tag{2}$$

$$\widehat{\mathbf{x}} = \mathrm{vec}(\widehat{\mathbf{X}}), \quad \mathbf{x} = \mathrm{vec}(\mathbf{X}), \quad \mathbf{y} = \mathrm{vec}(\mathbf{Y}), \quad \mathbf{C} = \mathbf{B} \otimes \mathbf{A}$$

Several algorithms are known in the literature to solve (2).

In the recent past, algorithms to directly solve (1) have also been proposed.

Discrete valued matrix sketching

Let $\mathcal{A}$ be a finite set of non-zero elements and $\bar{\mathcal{A}} \triangleq \mathcal{A} \cup 0$.

Now, the discrete valued matrix sketching problem can be stated as

$$\min_{\widehat{\mathbf{X}} \in \bar{\mathcal{A}}^{M \times N}} \|\widehat{\mathbf{X}}\|_1 \quad \text{s.t. } \mathbf{A}\widehat{\mathbf{X}}\mathbf{B}^T = \mathbf{Y}, \tag{3}$$

$$\min_{\widehat{\mathbf{x}} \in \bar{\mathcal{A}}^{MN}} \|\widehat{\mathbf{x}}\|_1 \quad \text{s.t. } \mathbf{C}\widehat{\mathbf{x}} = \mathbf{y}. \tag{4}$$

Our goal is to solve the general and noisy version of the above problem, which can be formulated as

$$\min_{\widehat{\mathbf{X}} \in \bar{\mathcal{A}}^{M \times N}} \|\widehat{\mathbf{X}}\|_1 + \lambda \|\mathbf{Y} - \mathbf{A}\widehat{\mathbf{X}}\mathbf{B}^T\|_2, \tag{5}$$

$$\min_{\widehat{\mathbf{x}} \in \bar{\mathcal{A}}^{MN}} \|\widehat{\mathbf{x}}\|_1 + \lambda \|\mathbf{y} - \mathbf{C}\widehat{\mathbf{x}}\|_2. \tag{6}$$

Matching Pursuit based Matrix Sketching

Matching pursuit

Iteratively recover the support by projecting the residue over the atoms until the residue is minimized.

$\mathbf{Y}$ can be written as sum of MN rank-one matrices

$$\mathbf{A}\mathbf{X}\mathbf{B}^T = \sum_{i=1}^{M}\sum_{j=1}^{N} X_{ij}\mathbf{a}_i\mathbf{b}_j^T = \sum_{k=1}^{K} X_{i_k j_k}\mathbf{a}_{i_k}\mathbf{b}_{j_k}^T. \quad (7)$$

Here, the atoms are $\mathbf{\Phi}_n \triangleq \mathbf{a}_{i_n}\mathbf{b}_{j_n}^T$.

Projection

The support can be recovered by the following projection operation

$$s_k = \underset{i,j}{\operatorname{argmax}} \left|\mathbf{a}_i^T\mathbf{Y}\mathbf{b}_j\right|. \quad (8)$$

[1] J. Tropp, “Greed is good: Algorithmic results for sparse approximation,” IEEE Transactions in Information Theory, vol. 50, no. 10, 2004.

Matching pursuit based direct matrix sketching

1 Initialize

Residue: $\mathbf{R}_0 = \mathbf{Y}$, support set $\mathcal{S}_0 = \{\emptyset\}$, and iteration index $k = 1$.

2 Projection

Find the best matching index $s_k \triangleq \{i_k, j_k\}$.

3 Update Support

$\mathcal{S}_k = \mathcal{S}_{k-1} \cup s_k$.

4 Update Residue

Compute $\mathbf{R}_k$.

5 Check and Terminate

Stop if $k \geq K$, else increment k and go to Step 2.

Computing the residue

The key component of the proposed matrix sketching algorithm is the computation of the residue for discrete valued sparse matrices.

$$\mathbf{R}_k = \min_{X_n \in \mathcal{A}} \left\| \mathbf{Y} - \sum_{n=1}^{k} X_n \mathbf{\Phi}_n \right\|_2 . \tag{9}$$

State-of-art: solve (9) using linear filtering or convex programming.

However, such methods can be very much sub-optimal for discrete valued matrices.

Optimal solution: Maximum a posteriori (MAP) solver. Has a computational complexity of $O(|\mathcal{A}|^k LP)$.

Proposed solution: Belief propagation (BeP) based solver. Has a low computational complexity of $O(LP|\mathcal{A}|k^2)$.

Residue computation using belief propagation

The observation can be written as

$$\mathbf{Y} = \sum_{n=i}^{k} X_n \mathbf{\Phi}_n + \underbrace{\sum_{m=k+1}^{K} X_m \mathbf{\Phi}_m}_{\triangleq \mathbf{\Gamma}_k, \text{ bias matrix}} + \underbrace{\mathbf{W}}_{\text{noise matrix}} \tag{10}$$

BeP algorithm iteratively computes the approximate MAP solution.

- Passes beliefs over a bipartite graph with k factor nodes and LP observation nodes.
- The observation nodes compute $\Pr(Y_{ij}|X_n = x \in \mathcal{A})$ and send these beliefs to the corresponding factor nodes.
- Using these beliefs, the factor nodes compute the posterior probability $\Pr(X_n = x \in \mathcal{A}|\mathbf{Y}, \mathbf{\Phi})$.
- Near optimal for large dimensions.

Constrained Message Passing

Greedy is *not always* good

Can we do better than matching pursuit?

- Sparse recovery algorithms known in the literature are optimal in the mean squared error (MSE) sense but not in the symbol error rate (SER) sense.
- For recovering discrete valued sparse quantities, greedy algorithms may not always be optimal.
  - The residues computed through linear techniques for discrete quantities do not always achieve optimality.
  - The residues computed through iterative techniques introduce the problem of error propagation.
  - Sub-optimal performance at low values of SNR.
- One solution: MAP based recovery algorithm.

Constrained message passing

The proposed solution is a message passing algorithm to compute the MAP solution in the vectorized model.

An approximately MAP solution is obtained by constructing a bipartite graph using the posterior probability

$$\Pr(\mathbf{x}|\mathbf{y}) \propto \Pr(\mathbf{y}|\mathbf{x}) \Pr(\mathbf{x}) = \Pr(\mathbf{y}|\mathbf{x}) \Pr(\mathbf{x}|\mathbf{s}) \Pr(\mathbf{s}). \tag{11}$$

Here, $\mathbf{s}$ is the support vector of $\mathbf{x}$.

Unlike traditional message passing algorithms, here, we compute the posterior probabilities $\Pr(\mathbf{x}|\mathbf{y})$ subject to the constraint given by the support.

This is referred to as constrained message passing (CoMP) algorithm.

Messages in CoMP

There are LP observation nodes and MN factor nodes.

Observation to factor node message

- The likelihoods $\Pr(y_i|x_j = x \in \bar{\mathcal{A}})$.
- Easy to compute with central limit theorem for high values of K.

Factor to observation node message

- Applying the support constraints, the posteriors are evaluated as $\Pr(x_j = x \in \bar{\mathcal{A}}|s_j = s \in \{0,1\}, y_i) \Pr(s_j = s \in \{0,1\}|y_i)$.
- The symbol-posteriors can be simplified as,

$$\begin{aligned} \Pr(x_j = 0|s_j = 0, y_i) &= 1, \\ \Pr(x_j = x \in \mathcal{A}|s_j = 0, y_i) &= \Pr(x_j = 0|s_j = 1, y_i) = 0. \end{aligned}$$

The constraints

In evaluation of the support-posteriors $\Pr(s_j|y_i)$, the sparsity structure in $\mathbf{x}$ is utilized.

- Let $\mathcal{G}_1, \cdots, \mathcal{G}_Q$ be sets of indices and $\bigcup_{q=1}^{Q} \mathcal{G}_q = \{1, \cdots, MN\}$.
- Let $K_q = \|\mathbf{s}_{\mathcal{G}_q}^T\|_0$, i.e., the number of non-zero elements in $\mathbf{x}_{\mathcal{G}_q}^T$.
- The sparsity constraint $\mathcal{C}_q$: $\sum_{j \in \mathcal{G}_q} s_j = K_q$.
- Hence, a valid $\mathbf{x}$ should satisfy Q constraints $\mathcal{C}_1, \cdots, \mathcal{C}_Q$.
- Let $\mathcal{C}'_j$ be the set of indices of the constraints to which s_j belongs, $\mathcal{C}'_j \subseteq \{1, \cdots, Q\}$.

Now, the support-posteriors can be evaluated as

$$\Pr(s_j|y_i) = \Pr(s_j|y_i, \mathcal{C}'_j) \propto \Pr(\mathcal{C}'_j|s_j, y_i) \Pr(s_j). \tag{12}$$

Example

Consider $\mathbf{X}$ to be a K-sparse matrix. Here,

$$Q = 1, \qquad \mathcal{G}_1 = \{1, \cdots, MN\}, \qquad K_1 = K, \qquad \mathcal{C}_1 : \sum_{j=1}^{MN} s_j = K.$$

The support-posteriors are

$$\Pr(s_j = 0|y_i) \propto \Pr\left(\sum_{l \neq j}^{MN} s_l = K \Big| y_i\right) \frac{K}{MN} \propto \Pr\left(\sum_{l \neq j}^{MN} s_l = K \Big| y_i\right),$$

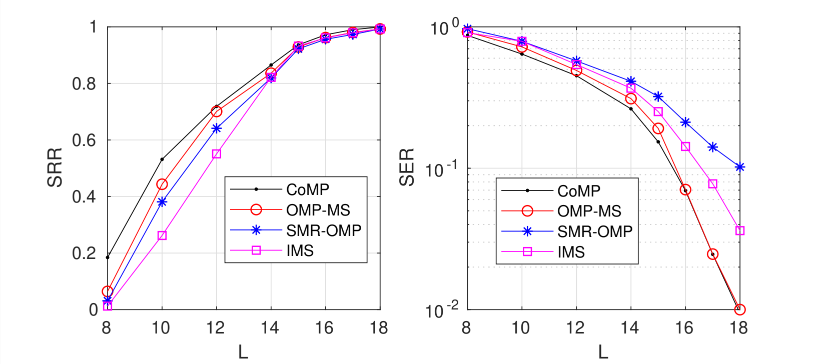
$$\Pr(s_j = 1|y_i) \propto \Pr\left(\sum_{l \neq j}^{MN} s_l = K - 1 \Big| y_i\right) \frac{K}{MN} \propto \Pr\left(\sum_{l \neq j}^{MN} s_l = K - 1 \Big| y_i\right).$$

The above probabilities can be obtained by convolution of $K - 1$ Bernoulli distributions obtained from the likelihoods.

This computation can be further simplified using central limit theorem for large values of K.

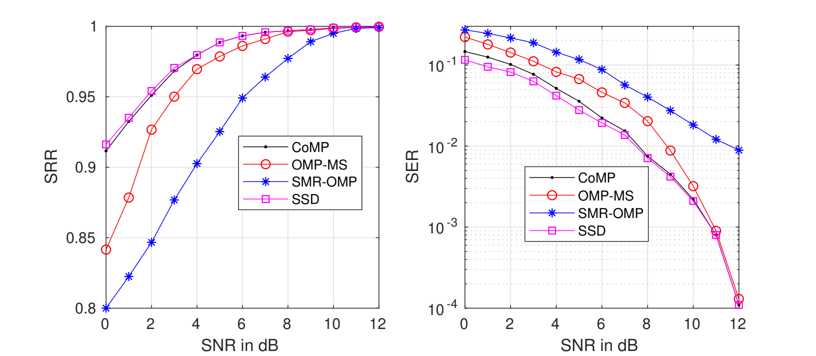
Numerical Results

Recovery performance



Performance of CoMP and OMP-MS for $M = N = 32$, $K = 20$, and different sketching matrices sizes.

Recovery performance



Performance of CoMP and OMP-MS for $M = N = 8$, $L = P = 6$, $K = 10$, and different values of SNR.

Conclusions

Summary

- We described the discrete valued sparse matrix sketching problem, which naturally arises in several practical scenarios.
- We presented a matching pursuit based matrix sketching algorithm with low computational complexity.
- We presented the constrained message passing algorithm which provides very good recovery performance even at low values of SNR and has the flexibility to exploit arbitrary sparsity structures.

Thank You!