Self-Expressive Clustering of Binary Data via Group Sparsity

Andrew S. Lan
Rice University
e-mail: sl29@rice.edu

Christoph Studer
Cornell University
e-mail: studer@cornell.edu

Richard G. Baraniuk
Rice University
e-mail: richb@rice.edu

***Abstract*—We present a novel, computationally efficient approach to cluster binary-valued data using self-expressive representations. Given a binary-valued data matrix, we use sparse, self-expressive representations to cluster similar rows and similar columns. We formulate our method as a convex logistic matrix-factorization problem that relies on group sparsity to identify the key rows and columns that explain the observed data. We demonstrate the effectiveness of our approach on two educational datasets where we cluster similar learners and test questions.**

I. Introduction

Self-expressive data representations seek to explain observed data points as a linear combination of other observed data points [1]–[4]. This approach resembles dictionary learning (DL) [5], [6], where the dictionary simply corresponds to the observed data points. The advantages of such an approach are that (i) the resulting learning algorithms are convex and (ii) the learned representations enables interpretability by identifying so-called "eigen-data points" that are sufficient for explaining the remaining observed data.

Existing results on self-expressive representations focus exclusively on real-valued data [1], [2]. In many practical applications, however, the observed data is binary-valued or quantized [7]. To fill this gap, we focus on self-expressive data representations from binary-valued observations to identify the key eigen-data points, which enables us to cluster similar columns and/or rows of the data matrix. We formulate our method, referred to as Self-Expressive Clustering (SEC), as a convex logistic optimization problem and use sparsity-inducing penalties to select a small number of rows and columns to build concise self-representations. Our model and algorithm can be applied to a wide range of applications, including the clustering of similar questions (e.g., from a multiple-choice test) and learners in an educational setting, clustering of movies or viewers based on similar ratings in collaborative filtering applications, and clustering of similar users from dating datasets.

II. SEC: Self-Expressive Clustering

We consider the case of a (partially) observed $P \times N$ binary-valued data matrix $\mathbf{Y} \in \{\pm 1\}^{P \times N}$ consisting of response data of N users to P items. We assume that only the entries $Y_{i,j}$ in the index set Ω are observed. In contrary to the conventional, real-valued and one-sided self-representation models in [1], [2], we propose the following two-sided binary-valued observation model:

$$\mathbf{Y} = \operatorname{sign}(\mathbf{A}\overline{\mathbf{Y}} + \overline{\mathbf{Y}}\mathbf{B}^T + \mathbf{N}). \tag{1}$$

Here, the entries in $\mathbf{N}$ are i.i.d. standard logistic distributed. We construct the matrix $\overline{\mathbf{Y}}$ from the observed entries of $\mathbf{Y}$ as follows: $\overline{Y}_{i,j} = Y_{i,j}$ if $(i,j) \in \Omega$ and $\overline{Y}_{i,j} = 0$ otherwise. The $P \times P$ left matrix $\mathbf{A}$ and $N \times N$ right matrix $\mathbf{B}$ characterize how the data in $\mathbf{Y}$ is self-represented. Specifically, the entries $A_{i,j}$ with $i,j \in \{1,2,\dots,P\}$ indicate how row i is self represented by row j in $\overline{\mathbf{Y}}$, and the entries $B_{m,n}$ with $m,n \in \{1,2,\dots,N\}$ indicate how column m is self represented by column n in $\overline{\mathbf{Y}}$.

III. Optimization Problem and Algorithm

To identify a *small* number of rows and columns that explain the entire binary-valued data matrix sufficiently well, we minimize the negative log-likelihood $-\log p(\mathbf{Y}|\mathbf{A},\mathbf{B})$ of the data model (1) together with (group) sparsity-inducing penalties on $\mathbf{A}$ and $\mathbf{B}$ [8], [9] In particular, we solve the following optimization problem:

$$\text{(P)} \quad \begin{cases} \underset{\mathbf{A}\in\mathbb{R}^{P\times P},\mathbf{B}\in\mathbb{R}^{N\times N}}{\text{minimize}} & -\log p(\mathbf{Y}|\mathbf{A},\mathbf{B}) \\ & +\lambda(\sum_{i=1}^{P}\|\mathbf{a}_i\|_2 + \sum_{j=1}^{N}\|\mathbf{b}_j\|_2) \\ & +\gamma(\|\mathbf{A}\|_1 + \|\mathbf{B}\|_1), \\ \text{subject to} & \mathbf{A}\geq 0, \mathbf{B}\geq 0 \\ & \operatorname{diag}(\mathbf{A}) = \mathbf{0}, \operatorname{diag}(\mathbf{B}) = \mathbf{0}. \end{cases}$$

Here, $\mathbf{a}_i$ and $\mathbf{b}_j$ denote the i^{th} column of $\mathbf{A}$ and j^{th} column of $\mathbf{B}$, respectively. The group-sparsity penalties $\sum_{i=1}^{P}\|\mathbf{a}_i\|_2$ and $\sum_{j=1}^{N}\|\mathbf{b}_j\|_2$ enforce that only a small number of "eigen-columns" of $\mathbf{A}$ and $\mathbf{B}$ are selected. The ℓ_1-norm penalties on $\mathbf{A}$ and $\mathbf{B}$ are the sums of the absolute values of their entries. The constraints $\mathbf{A} \geq 0$ and $\mathbf{B} \geq 0$ induce entry-wise non-negativity and encourage representations of similar columns, and $\operatorname{diag}(\mathbf{A}) = \mathbf{0}$ and $\operatorname{diag}(\mathbf{B}) = \mathbf{0}$ inhibit pure self-expressive representations. The parameters λ and γ determine the amount of group sparsity and entry-wise sparsity, respectively, and are selected via cross-validation [10].

The problem (P) is convex, and we use a fast iterative shrinkage-thresholding algorithm (FISTA)-based algorithm [11], [12] to estimate $\mathbf{A}$ and $\mathbf{B}$. In each algorithm iteration, we perform a gradient step with respect to the smooth negative log-likelihood term and a projection step with respect to the non-smooth, sparsity-inducing penalties. The projection requires us to solve problems of the form

$$\underset{\operatorname{diag}(\mathbf{X})=\mathbf{0},\mathbf{X}\geq 0}{\text{minimize}} \quad \tfrac{1}{2}\|\mathbf{X}-\mathbf{Z}\|_F^2 + \lambda\textstyle\sum_i \|\mathbf{x}_i\|_2 + \gamma\|\mathbf{X}\|_1,$$

for which we use the alternating direction method of multipliers [13].

IV. Numerical Experiments

In order to demonstrate the efficacy of SEC, we evaluate its predictive performance on unobserved entries in two educational datasets. In each experiment, we learn $\mathbf{A}$ and $\mathbf{B}$ from 75% randomly selected entries in $\mathbf{Y}$, test on the other 25%, and report the prediction accuracy. Each experiment is averaged over 10 Monte Carlo trials. Table I shows the predictive performance (in percent) on both datasets. The results show that using both rows and columns to self-represent the data results in superior performance as opposed to using only $\mathbf{A}$ or $\mathbf{B}$. The proposed approach performs almost as well as the state-of-the-art quantized matrix completion (Q-MC) method in [7], which does not enable interpretability.

Figures 1 and 2 visualize the explanatory power of SEC using Dataset 1 consisting of $N = 43$ learners answering $P = 143$ questions in an undergraduate signal processing course. For simplicity, we visualize the results of our approach with only $\mathbf{A}$ or only $\mathbf{B}$. Circles represent the sparse set of selected "eigen-questions/learners," while boxes represent the other questions/learners. Thicker edges represent stronger similarity. We observe that a small number of eigen-learners or eigen-questions are capable of explaining the entire dataset. In Figure 1, for example, we see that most questions cluster around question 17 and 119; a careful inspection of these questions shows that they cover a large proportion of all concepts in the course.

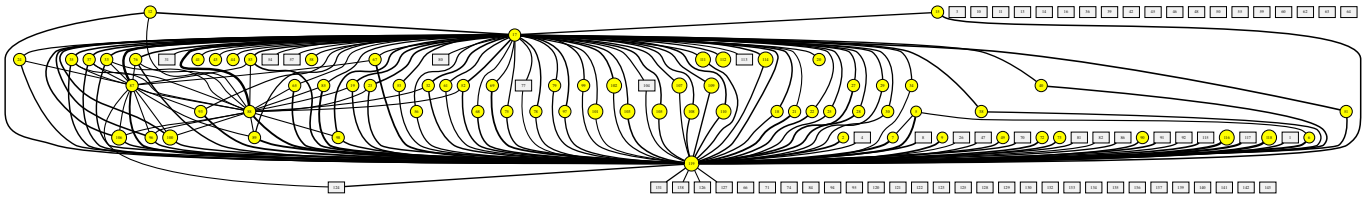

Fig. 1. Visualization of self-expressive clustering among the questions. Circles represent "eigen-questions" and boxes represent other questions.

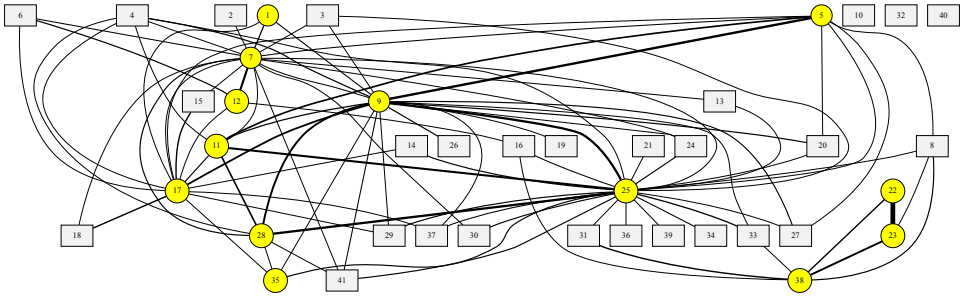

Fig. 2. Visualization of self-expressive clustering among the learners. Circles represent "eigen-learners" and boxes represent other learners.

TABLE I
PREDICTIVE PERFORMANCE OF SELF-EXPRESSIVE CLUSTERING

Dataset	$\mathbf{A}$ only	$\mathbf{B}$ only	$\mathbf{A}$ and $\mathbf{B}$	Q-MC [7]
1	$82.4 \pm 0.4\%$	$81.8 \pm 0.2\%$	$86.3 \pm 0.5\%$	$87.0 \pm 0.2\%$
2	$78.8 \pm 0.3\%$	$78.4 \pm 0.3\%$	$80.1 \pm 0.4\%$	$80.2 \pm 0.3\%$

REFERENCES

[1] E. Elhamifar, G. Sapiro, and R. Vidal, "See all by looking at a few: Sparse modeling for finding representative objects," in *IEEE Conf. on Computer Vision and Pattern Recognition*, June 2012, pp. 1600–1607.

[2] X. Fu, W. Ma, T. Chan, and J. Bioucas-Dias, "Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related," *arXiv Preprint arXiv:1409.4320*, Sep. 2014.

[3] E. L. Dyer, A. C. Sankaranarayanan, and R. G. Baraniuk, "Greedy feature selection for subspace clustering," *J. Mach. Learn. Res.*, vol. 14, no. 1, pp. 2487–2517, Jan. 2013.

[4] E. L. Dyer, "New theory and methods for signals in unions of subspaces," Ph.D. dissertation, Rice University, May 2014.

[5] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proc. 26th Annual Intl. Conf. on Machine Learning*, June 2009, pp. 689–696.

[6] M. Aharon, M. Elad, and A. M. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Sig. Proc.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[7] A. S. Lan, C. Studer, and R. G. Baraniuk, "Matrix recovery from quantized and corrupted measurements," in *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, May. 2014, pp. 4973–4977.

[8] F. Bach, R. Jenatton, J. Mairal, and G. Obozinski, "Optimization with sparsity-inducing penalties," *Foundations and Trends in Machine Learning*, vol. 4, no. 1, pp. 1–106, Jan. 2012.

[9] Y. C. Eldar, P. Kuppinger, and H. Bölcskei, "Block-sparse signals: Uncertainty relations and efficient recovery," *IEEE Trans. Sig. Proc.*, vol. 58, no. 6, pp. 3042–3054, Jun. 2010.

[10] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer, 2010.

[11] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Science*, vol. 2, no. 1, pp. 183–202, Mar. 2009.

[12] T. Goldstein, C. Studer, and R. G. Baraniuk, "A field guide to forward-backward splitting with a FASTA implementation," *arXiv Preprint arXiv:1411.3406*, Nov. 2014.

[13] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Foundations and Trends in Machine Learning*, vol. 3, no. 1, pp. 1–122, Jan. 2011.