Multi-View Unsupervised Column Subset Selection via Combinatorial Search (Student Abstract)

Guihong Wan[1,2,*], Ninghui Hao[2,3,*], Crystal Maung[4],
Haim Schweitzer[4], Chen Zhao[5], Kun-Hsing Yu[3], Yevgeniy R. Semenov[2]
[1]Departments of Biostatistics and Epidemiology, Harvard T. H. Chan School of Public Health, MA, USA
[2]Department of Dermatology, Massachusetts General Hospital, Harvard Medical School, MA, USA
[3]Department of Biomedical Informatics, Harvard Medical School, MA, USA
[4]Department of Computer Science, University of Texas at Dallas, Texas, USA
[5]Department of Computer Science, Baylor University, Texas, USA
guihong_wan@hsph.harvard.edu, ninghuihao@hms.harvard.edu

Abstract

Given a data matrix, unsupervised column subset selection refers to the problem of identifying a subset of columns that can be used to linearly approximate the original data matrix. This problem has many applications, such as feature selection and representative selection, but solving it optimally is known to be NP-hard. We consider multi-view unsupervised column subset selection, which extends the concept of (single-view) column subset selection to data represented in multiple views or modalities. We introduce a combinatorial search algorithm for this generalized problem. One variant of the algorithm is guaranteed to compute an optimal solution in a setting similar to the classical A* algorithm. Other suboptimal variants, in a setting similar to the weighted A* algorithm, are much faster and provide a solution along with a bound on its quality.

Introduction

Column Subset Selection (CSS). Let $X \in \mathbb{R}^{m\times n}$ be the data matrix and $0<k<n$ be an integer. The objective of (single-view, unsupervised) CSS is to identify k columns from X such that the following reconstruction error is minimized: $\min_\theta \|X - S_\theta S_\theta^+ X\|_F^2$, where θ is the index set of k selected columns from X, S_θ is the corresponding selection submatrix, S_θ^+ is the pseudoinverse of S_θ, and $\|\cdot\|_F$ denotes the Frobenius norm. CSS has many applications, such as unsupervised feature selection and representative selection. However, solving it optimally is known to be NP-hard, even if each column has two nonzero elements (Wan et al. 2024).

Multi-View Column Subset Selection (MVCSS). Let $X = \{X^1, X^2, \ldots, X^V\}$ be a set of data matrices from V different views, where $X^v \in \mathbb{R}^{m^v \times n}$ is the data matrix corresponding to the v-th view for $v=1,\ldots,V$. Let $\alpha = \{\alpha_1,\ldots,\alpha_V\}$ satisfying $\sum_v \alpha_v = 1$ be a set of non-negative weights that control the contribution of each view. The MVCSS problem is to select a common set of k columns across all views such that the weighted reconstruction error is minimized: $\min_\theta \sum_{v=1}^V \alpha_v \|X^v - S_\theta^v (S_\theta^v)^+ X^v\|_F^2$, where $S_\theta^v \in \mathbb{R}^{m^v\times k}$ is the selection matrix for the v-th view, corresponding to the k selected column indexes in θ.

[*]These authors contributed equally.

Copyright © 2025, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

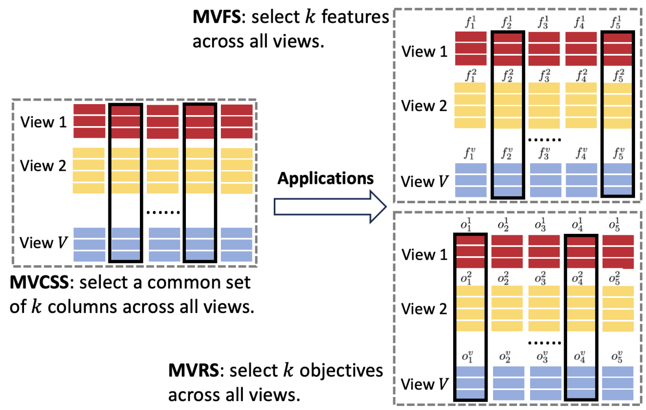

Figure 1: Illustrations of MVCSS and its Applications.

The MVCSS problem generalizes the CSS problem for single-view data. This generalization introduces additional complexity due to the selection across different views while maintaining consistency. We consider a more generalized MVCSS problem as defined below.

Definition 1. (Problem Statement) Given a set of data matrices $X = \{X^1, X^2, \ldots, X^V\}$, a set of non-negative weights $\alpha = \{\alpha_1, \ldots, \alpha_V\}$, an integer $0<k<n$, and a matrix norm $\|\cdot\|_\Phi$, the MVCSS problem is to identify a subset θ of k common columns across all views such that the following reconstruction error is minimized over all possible $\binom{n}{k}$ choices:

$$e(\theta, \alpha, \Phi) = \sum_{v=1}^{V} \alpha_v \|X^v - S_\theta^v (S_\theta^v)^+ X^v\|_\Phi. \quad (1)$$

The proposed algorithm is designed to optimize under any unitarily invariant matrix norm, including Schatten p-norm: $(\sum_{\tau=1}^m \sigma_\tau^p)^{1/p}$ and Ky Fan q-norm $\sum_{\tau=1}^q \sigma_\tau$ for $0<p\leq\infty$ and $1\leq q\leq m$, where $\sigma_1 \geq \ldots \geq \sigma_m$ are the singular values. MVCSS can be applied in tasks like multi-view feature selection (MVFS), where each column represents a feature (f_i^v: i-th feature for the v-th view) and a subset of features is needed to capture the shared structure, and multi-view representative selection (MVRS), where each column is an objective (o_i^v: i-th objective for the v-th view) (Figure 1).

Methodology

Inspired by previous work on CSS, such as (He et al. 2019), we address the MVCSS problem (1) within the combinatorial search framework. To achieve this, we need to construct the search graph and define the heuristic function to guide the search (Algorithm 1).

Algorithm 1: Combinatorial Search for MVCSS

Input: X: the multi-view data matrices. k: the desired number of columns. $h(\theta_i, k, \alpha, \Phi, \epsilon)$: the heuristic function.
Output: a subset θ of k columns.
Data structure: the fringe list F and the closed list C.
Initiation: Put the empty subset into F.

```
while L is nonempty do
    Select θ_p with the smallest h value from F. Ties are
    resolved in favor of the larger subset (depth).
    if θ_p contains k columns then
        Stop and return θ_p as the solution subset.
    else
        for each child θ_c of θ_p do
            if θ_c is not in C then
                Compute h(θ_c, k, α, Φ, ε) for θ_c.
                Put θ_c with the h value in F and C.
            end if
        end for
    end if
end while
```

Search Graph. Similar to (He et al. 2019), the search graph is constructed using subsets of columns indexed across all views. The root node corresponds to the empty subset, and the child nodes of a parent node are created by adding a new column into the subset of the parent node. We do not distinguish a node and a subset to simplify notation.

Heuristic Function. Let θ_i be a subset (node) in the search graph with k_i columns, where additional $\overline{k}$=k−k_i columns to be selected: $d_i = \min_{\overline{\theta}} e(\theta_i \cup \overline{\theta}, \alpha, \Phi)$ with $|\overline{\theta}|$=$\overline{k}$, which is hard to compute. We define following functions:

$$l_i = e(\theta_i \cup \overline{w}, \alpha, \Phi),\ u_i = e(\theta_i, \alpha, \Phi), \tag{2}$$

where $\overline{w}$ is the index set of the $\overline{k}$ eigenvectors corresponding to the $\overline{k}$ largest singular values of the residual matrices: $X_r^v = X^v - S_\theta^v (S_\theta^v)^+ X^v$. For l_i, the first $\overline{k}$ eigenvectors $U_{\overline{w}}^v$ of X_r^v is used to replace $S_{\overline{\theta}}^v$ in d_i. For u_i, the potential error reduction by the additional $\overline{k}$ columns is ignored. Thus, $l_i \leq d_i \leq u_i$. We use the following heuristic function to guide the search: $h(\theta_i, k, \alpha, \Phi, \epsilon) = l_i + \epsilon u_i$, where $\epsilon \in [0, \infty]$.

Theorem 1. Let e be the error of the solution provided by Algorithm 1. Let e_{opt} be the typical unknown smallest error.
a. (optimal variant) When $\epsilon = 0$, Algorithm 1 is guaranteed to give an optimal solution with $e = e_{\text{opt}}$.
b. (greedy variant) When $\epsilon = \infty$, Algorithm 1 greedily selects nodes to be expanded.
c. (general variant) For $\epsilon \in [0, \infty]$, the solution found by Algorithm 1 is bounded: $e \leq e_{\text{opt}} + b$, where $b = e - l_{\min}$, which is the smallest l value among nodes in the fringe list after finding the solution.

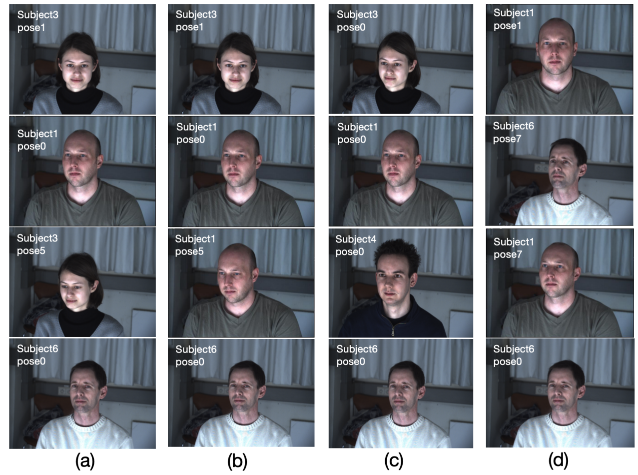

Figure 2: Visualization of the k=4 selected EPFL stereo faces with different approaches (column-wise). (**a**) The proposed algorithm with ϵ=0 captured 3 subjects and 3 poses (Average recall = 0.75). (**b**) The proposed algorithm with ϵ=0.1 and 0.5 provided same results, capturing 3 subjects and 3 poses (Average recall = 0.75). (**c**) K-medoid detected all 4 subjects but only 1 pose (Average recall = 0.675). (**d**) SP captured 2 subjects and 3 poses (Average recall = 0.675).

k		$\epsilon = 0$	$\epsilon = 0.1$	$\epsilon = 0.5$	Random	K-Medoids	SP
4	e	**13614**	15065	15065	41955	23421	68928
	b	**0**	6709	6709	–	–	–
8	e	**8736**	10717	12013	26250	15792	15341
	b	**0**	5941	7237	–	–	–

Table 1: Error (e) and bound (b) for the proposed algorithm compared to K-Medoids (Rdusseeun and Kaufman 1987) and SP (Joneidi et al. 2020). The "–" indicates that the algorithm does not compute a bound.

Experiments

We evaluated the proposed algorithm by comparing to baseline methods using the EPFL stereo face dataset (Fransens, Strecha, and Van Gool 2005), which consists of 100 participants whose faces were captured in 8 poses with 2 stereo cameras (2 views). We randomly selected k=$\{4, 8\}$ subjects and 4 poses as input data. Each facial appearance was represented by a 136-dimensional feature vector of facial landmarks extracted using the dlib library. The proposed algorithm was performed with equal weights for all views (α_1=...=α_V=$1/n$) under the Frobenius norm.

Figure 2 and Table 1 present the comparison results, demonstrating the effectiveness of the proposed algorithm.

Conclusions

This study proposed a combinatorial search algorithm for multi-view column subset selection, outperforming baseline methods in reconstruction accuracy for multi-view data analysis. Future work includes benchmarking against additional competitors, exploring dynamic weighting schemes, and applying the algorithm to real-world multi-view data.

References

Fransens, R.; Strecha, C.; and Van Gool, L. 2005. Parametric stereo for multi-pose face recognition and 3D-face modeling. In *International Workshop on Analysis and Modeling of Faces and Gestures*, 109–124. Springer.

He, B.; Shah, S.; Maung, C.; Arnold, G.; Wan, G.; and Schweitzer, H. 2019. Heuristic search algorithm for dimensionality reduction optimally combining feature selection and feature extraction. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, 2280–2287.

Joneidi, M.; Vahidian, S.; Esmaeili, A.; Wang, W.; Rahnavard, N.; Lin, B.; and Shah, M. 2020. Select to better learn: Fast and accurate deep learning using data selection from nonlinear manifolds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 7819–7829.

Rdusseeun, L.; and Kaufman, P. 1987. Clustering by means of medoids. In *Proceedings of the statistical data analysis based on the L1 norm conference*.

Wan, G.; Mao, W.; Semenov, Y. R.; and Schweitzer, H. 2024. Equivalence between Graph Spectral Clustering and Column Subset Selection (Student Abstract). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 23673–23675.