

# Summary of Sparse Biclustering of Transposable Data

## 1 Biclustering

Biclustering, block clustering, co-clustering, or two-mode clustering is a data mining technique which allows simultaneous clustering of the rows and columns of matrix.

## 2 Assumptions

- each matrix element is normally distributed with a bicluster-specific mean;
- the biclusters partition the rows and columns of the matrix.

## 3 Sparse biclustering

### 3.1 Model assumptions

- The biclusters all are constant biclusters, in which all elements take on approximately a constant value.
- $X_{ij} \sim N(\mu_{kr}, \sigma^2)$  for  $i \in C_k, j \in D_r, k = 1, \dots, K$  and  $r = 1, \dots, R$ , and they are independent with each other.

### 3.2 Model

The model is:

$$X_{ij} = \mu_{kr} + \varepsilon_{ij} \text{ where } \varepsilon_{ij} \sim N(0, \sigma^2)$$

**Given parameters:**  $K, R, \lambda$ ;

**Unknown parameters:**  $\mu_{kr}, \{C_k\}, \{D_r\}$ .

Maximizing the log likelihood of the data under the model with inducing sparsity by using a LASSO penalty, we arrived at:

$$\underset{C_1, \dots, C_K, D_1, \dots, D_R, \mu \in R^{K \times R}}{\text{minimize}} \left\{ \frac{1}{2} \sum_{k=1}^K \sum_{r=1}^R \sum_{i \in C_k} \sum_{j \in D_r} (X_{ij} - \mu_{kr})^2 + \lambda \sum_{k=1}^K \sum_{r=1}^R |\mu_{kr}| \right\} \quad (1)$$

where  $\lambda$  is a non-negative tuning parameter.

### 3.3 An extension to tensor

#### 3.3.1 Model assumptions

- The clusters all are constant clusters, in which all elements take on approximately a constant value.
- $X_{ijm} \sim N(\mu_{krm}, \sigma^2)$  for  $i \in C_k, j \in D_r, m \in E_l, k = 1, \dots, K, r = 1, \dots, R, l = 1, \dots, L$ , and they are independent with each other.

#### 3.3.2 Model

The model is:

$$X_{ijm} = \mu_{krm} + \varepsilon_{ijm} \text{ where } \varepsilon_{ijm} \sim N(0, \sigma^2)$$

**Given parameters:**  $K, R, L, \lambda$ ;

**Unknown parameters:**  $\mu_{krm}, \{C_k\}, \{D_r\}, \{E_l\}$ .

Maximizing the log likelihood of the data under the model with inducing sparsity by using a LASSO penalty, we arrived at:

$$\begin{aligned} & \underset{C_1, \dots, C_K, D_1, \dots, D_R, E_1, \dots, E_L, \mu \in R^{K \times R \times L}}{\text{minimize}} \left\{ \frac{1}{2} \sum_{k=1}^K \sum_{r=1}^R \sum_{l=1}^L \sum_{i \in C_k} \sum_{j \in D_r} \sum_{m \in E_l} (X_{ijm} - \mu_{krm})^2 + \lambda \sum_{k=1}^K \sum_{r=1}^R \sum_{l=1}^L |\mu_{krm}| \right\} \\ & \hspace{15em} (2) \end{aligned}$$

where  $\lambda$  is a non-negative tuning parameter.

#### 3.3.3 Algorithm

---

##### Algorithm 1 A

---

Initialize  $C_1, \dots, C_K, D_1, \dots, D_R$  and  $E_1, \dots, E_L$  by performing one-way k-means clustering on the columns and on the rows of the data matrix  $X$ .

**repeat**

(a) Holding  $C_1, \dots, C_K, D_1, \dots, D_R$  and  $E_1, \dots, E_L$  fixed, solve (1) with respect to  $\mu$  using LASSO regression.

(b) Holding  $\mu, D_1, \dots, D_R$  and  $E_1, \dots, E_L$  fixed, solve (1) with respect to  $C_1, \dots, C_K$ , by assigning the  $i$ th observation to the row cluster for which  $\sum_{r=1}^R \sum_{l=1}^L \sum_{j \in D_r} \sum_{m \in E_l} (X_{ijm} - \mu_{krm})^2$  is smallest.

(c) repeat (a).

(d) Holding  $\mu, C_1, \dots, C_K$  and  $E_1, \dots, E_L$  fixed, solve (1) with respect to  $D_1, \dots, D_R$ , by assigning the  $i$ th observation to the column cluster for which  $\sum_{k=1}^K \sum_{l=1}^L \sum_{i \in C_k} \sum_{m \in E_l} (X_{ijm} - \mu_{krm})^2$  is smallest.

(e) repeat (a).

(f) Holding  $\mu, C_1, \dots, C_K$  and  $D_1, \dots, D_R$  fixed, solve (1) with respect to  $E_1, \dots, E_L$ , by assigning the  $i$ th observation to the cluster of the third dimension for which  $\sum_{k=1}^K \sum_{r=1}^R \sum_{i \in C_k} \sum_{j \in D_r} (X_{ijm} - \mu_{krm})^2$  is smallest.

**until** Convergence

---

## 4 A spectral interpretation for biclustering

The optimization problem

$$\underset{A^T A=I_K, B^T B=I_K}{\text{maximize}} \quad \|A^T X B\|_F^2 \quad (3)$$

under two additional constraints:

- The elements of the  $k$ th column of A are 0 or  $\frac{1}{\sqrt{n_k}}$  with  $n_k \in Z^+$ ,  $\sum_{k=1}^K n_k = n$ .
- The elements of the  $k$ th column of B are 0 or  $\frac{1}{\sqrt{p_r}}$  with  $p_r \in Z^+$ ,  $\sum_{r=1}^K p_r = p$ .

makes (2) equivalent to the biclustering optimization problem (1) when  $\lambda = 0, K = R$ . So, with  $K = R$ , the biclustering problem (1) when  $\lambda = 0$  can be relaxed in order to yield the SVD.

## 5 Tuning parameter selection

$$BIC = np \times \log(RSS) + (q + 1)\log(np)$$

## 6 Simulation study

**Standard:** clustering error rate(CER), sparsity rate, sparsity error rate, proportion of correctly identified zeros(C.Zeros) and non-zeros(C.Non-zeros).

### 6.1 Definitions

**Clustering error rate (CER):**

Using adjusted rand index to measure the agreement between any two partitions for the data tensor. In this case, we have three kinds of CER in total: rowCER, columnCER and the CER of the third dimension. To be more specific, consider the rowCER. Denote  $S$  as the set of rows.  $T$  is the true partition of  $S$  and  $J$  is the clustering result with respect to rows. Here,

- a, the number of pairs of elements/labels in  $S$  that in the same subset in  $T$  and in the same subset in  $J$ .
- b, the number of pairs of elements/labels in  $S$  that in the different subsets in  $T$  and in the different subset in  $J$ .

$$rowCER = \frac{a + b}{C_n^2}$$

Intuitively,  $a + b$  can be considered as the number of agreements between  $T$  and  $J$  and  $c + d$  as the number of disagreements between  $T$  and  $J$ .

### 6.2 No bicluster means exactly equal to zero

**Conclusions:** The biclustering with  $\lambda = 0, 200$  leads to consistently better results than independent clustering of the rows and columns.

### 6.3 Some bicluster means exactly equal to zero

**Conclusions:** IP fails to identify any biclusters in this simulation set-up. SSVD and LAS perform comparably in this setting. But by far the best overall performance is achieved by sparse biclustering proposal with a large value of  $\lambda$ .

### 6.4 Multiplicative biclusters

**Conclusions:** SSVD has the best results in this simulation set-up, as in this set-up there are multiplicative biclusters.

### 6.5 Overlapping multiplicative biclusters

**Conclusions:** Both SSVD and sparse biclustering performs pretty good though the set-up violates the assumptions of sparse biclustering.

## A Additional biclustering results of Table 2

True value of (K,R)	n	p	Overall Accuracy	Selected K	Selected R
K=2, R=4	250	100	74%	2(0.0000)	3.7(0.0769)
K=2, R=4	20	50	16%	2.02(0.0318)	2.74(0.1090)

## B Additional simulation biclustering results of Table 3

Method	n	p	Row CER	Column CER	Sparsity Rate
k-means	20	50	0.3621(0.0223)	0.3407(0.0046)	0
Bicluster $\lambda = 0$	20	50	0.3509(0.0220)	0.3217(0.0058)	0
Bicluster $\lambda = 200$	20	50	0.3654(0.0206)	0.4136(0.0155)	0.4455(0.0260)
Bicluster $\lambda = 400$	20	50	0.4841(0.0099)	0.6751(0.0217)	0.8074(0.0553)
Bicluster $\lambda = 800$	20	50	0.4909(0.0061)	0.7478(0.0017)	1(0)
k-means	250	100	0.1202(0.0188)	0.1649(0.0089)	0
Bicluster $\lambda = 0$	250	100	0.1077(0.0177)	0.0958(0.0103)	0
Bicluster $\lambda = 200$	250	100	0.1104(0.0178)	0.0982(0.0105)	0.0610(0.0123)
Bicluster $\lambda = 400$	250	100	0.1119(0.0181)	0.1074(0.0097)	0.1192(0.0161)
Bicluster $\lambda = 800$	250	100	0.1171(0.0185)	0.1358(0.0098)	0.1889(0.0212)

## C Several results in tensor clustering

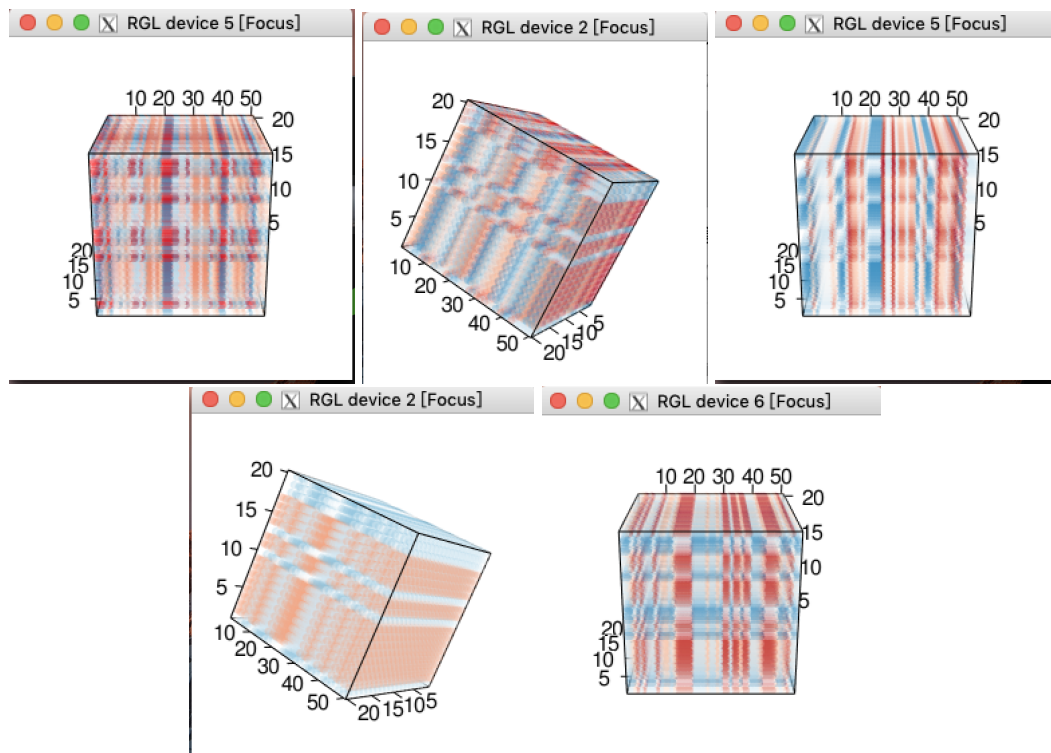


Figure 1: Several clustering results(first two: true data matrix, others: clustering results)