Scientific Journal of Informatics
Vol. 12, No. 2, May 2025
p-ISSN 2407-7658 https://journal.unnes.ac.id/journals/sji/index e-ISSN 2460-0040

Performance Evaluation of Cheng & Church (CC) and Spectral Biclustering Algorithms under Collinearity and Overlap Conditions

Siti Hafsah[1], Indahwati[2*], Hari Wijayanto[3]
[1, 2, 3]School of Statistics and Data Science, Mathematics, and Informatics, IPB University, Indonesia

Abstract.
Purpose: This study aims to address methodological challenges in evaluating biclustering algorithms under simultaneous collinearity and overlap, which often co-occur in real world multivariate data but are rarely analyzed simultaneously. This research highlights the importance of understanding how these structural challenges affect local pattern detection in data mining applications.
Methods: A simulation study was conducted using synthetic matrices embedded with two constant biclusters under 15 combinations of collinearity levels (ρ = 0.3,0.6,0.9) and overlap degrees (none, small, large). Each scenario was replicated 100 times. Performance was assessed using the Liu and Wang Index (ILW), while a three-way ANOVA tested the effects of algorithm type, collinearity, and overlap.
Result: Spectral Biclustering maintained stable ILW scores despite increasing collinearity, while CC performed better in low-overlap scenarios but was more sensitive to collinearity. Under high collinearity and large overlap, both algorithms experienced notable degradation. The ANOVA confirmed all main effects and interactions were significant ($p < 0.001$).
Novelty: This study contributes empirical evidence regarding the influence of interacting structural characteristics on biclustering performance. The results deliver practical insights for selecting suitable algorithms and emphasize the potential advantages of hybrid approaches that integrate the stability of spectral methods with the adaptability of residual-based techniques.

Keywords: Cheng and church biclustering, Spectral biclustering, Collinearity, Overlap, Liu and wang index
Received June 2025 / **Revised** July 2025 / **Accepted** July 2025

INTRODUCTION

The era of big data presents complex challenges in multivariate data analysis, mainly due to collinearity between variables and overlap between entities. These two conditions often co-occur and can make it difficult to identify latent patterns, which is important in fields such as bioinformatics, social sciences and economics. Exploratory data analysis (EDA) commonly uses clustering techniques to group entities based on similarities, but traditional methods such as K-Means only process rows or columns separately, and thus are unable to capture local patterns in subsets of rows and columns [1],[2].

Biclustering analysis is an extension of clustering methods that allows simultaneous clustering on both row and column dimensions [3]. This technique recognizes that only a subset of the data matrix may show similar patterns in a given subset of objects and subset of variables, forming a submatrix [4]. Taking into account the local characteristics of the data, biclustering aims to identify a submatrix $S \subseteq R \times C$ that exhibits pattern coherence among the set of rows R and columns C [5]. This coherence can be defined based on the uniformity of values, correlations, or certain trend patterns in the submatrices formed. One well-known approach is the Cheng & Church (CC) algorithm, which detects constant bicluster by minimizing the Mean Squared Residue (MSR), a metric that measures the local deviation to the mean within a bicluster [6],[7]. In contrast, Spectral Biclustering uses singular value decomposition (SVD). This technique projects the data into a lower-dimensional eigenspace to reveal a checkerboard-shaped structure. This method is known for its robustness against noise and computational efficiency, especially when applied to high-dimensional data [8]-[9].

*Corresponding author.
Email addresses: sitihafsah12_siti@apps.ipb.ac.id (Hafsah), indahwati@apps.ipb.ac.id (Indahwati)*, hari@apps.ipb.ac.id (Wijayanto)
DOI: 10.15294/sji.v12i2.26413

Initially, biclustering was developed and widely used in the analysis of gene expression profiles obtained from microarray experiments[10]-[11]. However, as it has evolved, the application of biclustering has expanded to various fields beyond bioinformatics, such as water consumption modeling [12], macroeconomic forecasting [13], social vulnerability mapping [14], and consumer behavior analysis [15], Research on biclustering in text mining, market data analysis [15], prediction and identification of abnormal energy consumption [16]. This diversity of applications demonstrates the need for algorithmic flexibility, given that data structures can vary significantly between fields. Therefore, various biclustering algorithms have been proposed to detect different types of coincidence patterns-from constant, additive, to multiplicative and order-preserving structures [17].

To date, there is no standardized guideline in the selection of biclustering algorithms for certain types of data. Algorithm selection is based on the merits of previous research, the characteristics of the resulting bicluster, as well as the ability to handle overlap between biclusters [18].This lack of standards is an important issue considering that biclustering is an NP-Hard problem that relies on heuristic approaches and is prone to the trap of local optimal solutions [19],[20]. The effectiveness of biclustering algorithms is not only affected by the underlying pattern type, but also by structural challenges such as overlap between biclusters and collinearity between variables. Overlap can blur the boundaries between clusters by allowing row and column members to overlap, while collinearity between variables can distort similarity measures and hinder local pattern detection [15], [21].

One of the main challenges in biclustering is the presence of overlapping structures, where multiple rows or columns can be members of more than one bicluster simultaneously. This phenomenon is common in biological and social data, where entities are often involved in several underlying processes [22]. Another important challenge is the collinearity between variables, which can disguise latent patterns and decrease the discriminative ability of clustering algorithms [23]. Collinearity is often encountered in multivariate data, and in model-based approaches, this condition can cause the covariance matrix to become singular and result in unstable parameter estimates [24]. Various studies have assessed the robustness of biclustering algorithms to various forms of data irregularities, such as Gaussian noise, missing values, and heterogeneous pattern structures [17],[25],[26]. Several studies have highlighted the structural challenges in biclustering, particularly overlap between biclusters and collinearity between variables. [20] developed G-Bic, a synthetic data generator that models overlapping biclusters to test the robustness of the algorithm in complex scenarios. [27] proposed a graph-based vertex splitting approach to handle overlap, and stated that this problem belongs to the NP-Complete class. Meanwhile, [24] showed that collinearity significantly degrades the performance of the algorithm, although Cheng & Church is more robust in simulated biological data. While overlap and collinearity have been studied separately, few studies have examined their combined impact. In fact, they often co-occur in real data and can obscure the bicluster structure and reduce algorithm accuracy. This methodological gap is worth exploring further.

This study aims to evaluate the performance of two representative algorithms Cheng & Church and Spectral Biclustering in simulated conditions with varying degrees of collinearity and overlap. Through a factorial design, this study assesses the robustness and sensitivity of the algorithms, and provides empirical recommendations for the selection of biclustering methods on data with complex structures.

METHODS

This study adopts a simulation-based experimental design to systematically evaluate the performance of two biclustering algorithms CC and Spectral Biclustering under controlled structural conditions involving collinearity and overlap. The simulation framework enables the generation of synthetic data matrices with precisely embedded bicluster patterns, allowing for comprehensive performance assessment. The overall methodology includes four main stages: (1) data simulation and bicluster insertion, (2) structural manipulation through collinearity and overlap configurations, (3) algorithm implementation, and (4) evaluation using quantitative performance metrics. Each stage is elaborated in the following subsections.

Data simulation and bicluster insertion

The data in this study was generated synthetically in the form of a two dimensional 50×50 matrix, which represents a small scale of data under controlled conditions. Each element in the initial matrix is generated from a standard independent normal distribution, $N(0,1)$, so it contains no initial regularity.

Two biclusters of constant type are inserted into the matrix. These biclusters are submatrices with homogeneous values generated from a normal distribution $N(\mu, 0.1)$, with $\mu > 0$. The pattern is designed to have significant contrast against the random background. The row and column positions for each bicluster were chosen randomly to avoid positional bias. The data structure was then modified based on two main factors, namely:

1) Collinearity among variables: Formed using a multivariate normal distribution with correlated columns. The correlation parameter was varied at levels 0.3 (low), 0.6 (moderate), and 0.9 (high).
2) Degree of overlap, defined by the proportion of shared rows and columns between the two biclusters—categorized into no overlap, small overlap (5–15 cells), and large overlap (>15 cells).

After bicluster insertion and structure modification, the entire matrix is reshuffled in the row and column dimensions. This process aims to disguise the bicluster positions and simulate irregularities as in real data. This simulation procedure is applied to every combination of conditions to ensure replicable and tractable results.

Research stages

This research is carried out through a series of systematic stages that aim to generate bicluster structures in a controlled manner, apply variations in data complexity, and evaluate the performance of two biclustering algorithms under controlled simulation conditions. The following series of steps illustrates the overall research workflow.

Background data generation

The background data is formed in the form of a two dimensional matrix of size 50×50 with each element being a realization of an independent normal distribution $N(0,1)$. This matrix acts as a random background without the presence of structured patterns, thus providing a neutral basis for controlled insertion of bicluster patterns.

Constant type bicluster insertion

Two constant-type bicyclers are inserted into each matrix. The rows and columns forming the bicluster are chosen randomly, while the values of the elements inside the bicluster are generated from a normal distribution $N(\mu, 0.1)$, with $\mu > 0$. This insertion aims to form homogeneous blocks of data that have a contrast to the random background, thus representing the presence of local structure in the data. The size, position of the bicluster, as well as the overlap configuration were determined based on the experimental design outlined in Table 1.

Table 1. Criteria for bicluster size and position

Biclustering	Ukuran	μk	level of overlap					
			no overlap		small overlap		large overlap	
			row	column	row	column	row	column
1	25 × 25	$N(15, 0.1)$	6:30	1:25	3:27	1:25	3:27	1:25
2	20 × 25	$N(10, 0.1)$	31:50	26:50	22:41	21:45	13:32	11:35

The varying degree of overlap between bicyclers is designed to represent the complexity of data structures commonly encountered in empirical practice. Low overlap indicates a condition where about 5 to 15 cells in both row and column dimensions have dual membership in both bicluster, while high overlap occurs when more than 15 cells share membership. This design aims to simulate overlapping phenomena commonly encountered in real data, such as observations associated with more than one condition. A 50×50 matrix was chosen to maintain a balance between computational efficiency and ease of interpretation, while allowing enough space for two interacting biclusters. The degree of collinearity ($\rho = 0.3, 0.6, 0.9$) reflects the variation in correlation prevalent in multivariate data [24]. Meanwhile, the degree of overlap was explicitly determined to fulfill the research objective, which was to evaluate the sensitivity of the algorithm to different levels of data structure complexity.

Matrix randomization

After the bicluster insertion process, the matrix is reshuffled independently in rows and columns. This step is done to obscure the position of the bicluster so that it is not directly identified by the algorithm, while simulating the characteristics of empirical data which is generally not spatially organized.

Addition of collinearity structure

Collinearity structure between variables is added by generating data from a multivariate normal distribution. The correlation parameter between columns is set in three levels of correlation: ρ = 0.3 (low collinearity), 0.6 (medium), and 0.9 (high). This variation aims to simulate the interdependency between features as commonly found in social and biological data.

Experiment scenario design

To evaluate the performance of the biclustering algorithm under various data structure conditions, a total of 15 experimental scenarios were designed. The scenarios are categorized as follows:

1) Collinearity-only scenarios
   Three matrices with increasing correlation levels ($\rho = 0.3, 0.6, 0.9$) and no overlap were used to isolate the effect of inter-variable collinearity.
2) Overlap only scenarios
   Datasets without imposed collinearity (i.e., columns generated independently) were used to isolate the effect of overlapping biclusters. Three overlap conditions were tested: no overlap, small overlap, and large overlap.
3) Overlap and collinearity combination scenarios
   Consists of nine combinations of the three levels of collinearity and overlap, to observe the interactive effect of the two factors together. Details of the scenario combinations are presented in Table 2.

Table 2. Overlap and collinearity combination scenarios

Scenario Category	Collinearity (ρ)	Overlap
Collinearity effect only	0.3	-
	0.6	-
	0.9	-
Overlap effect only	-	No overlap
	-	Small overlap
	-	Large overlap
Combination of collinearity × overlap	0.3	No overlap
	0.3	Small overlap
	0.3	Large overlap
	0.6	Small overlap
	0.6	Large overlap
	0.6	No overlap
	0.9	No overlap
	0.9	Small overlap
	0.9	Large overlap

Implementation of biclustering algorithms

This study uses a biclustering approach to identify hidden local patterns in a two-dimensional data matrix. Biclustering is a method of simultaneous grouping of rows and columns of a matrix that aims to find submatrices with coherent characteristics, thus allowing the spread of local patterns that are not detected through conventional clustering methods [28]. In general, biclustering divides the matrix $A_{I\times J}$ into smaller submatrices $B_{n\times m}$ based on a certain subset of rows and columns, with the quality of the bicluster measured by the residuals, which are the differences between the actual and predicted values calculated from the averages of the rows, columns, and the bicluster as a whole [29].

This research evaluates two biclustering algorithms that represent the two main approaches in local pattern detection, namely the CC algorithm and Spectral Biclustering. Each algorithm is implemented using R software.

1) Cheng & Church (CC)
   The Cheng & Church (CC) algorithm was introduced by [30] as one of the pioneering approaches in biclustering, especially for gene expression data. This method aims to identify two-dimensional submatrices (biclusters) that exhibit stable local patterns in both row and column dimensions simultaneously. The main criterion used to assess the coherence of a bicluster is the Mean Squared Residue (MSR), which is a measure of the average squared deviation of the elements in a submatrix from its local structure [31]. Mathematically, the MSR for a bicluster with the set of rows I and columns J is defined as the average of the squared differences between the element values and their constituent components, consisting of the row average, column average, and overall average in the submatrix. The formal equation 1 is [32].

$$MSR_{(I,J)} = \frac{1}{|I||J|} \sum_{i=1}^{I} \sum_{j=1}^{J} \left(y_{ij} - \alpha_i - \beta_j + \mu\right)^2 \quad (1)$$

The equation uses the notation y_{ij} to express the value of the data element in the i-th row and j-th column. The average value of all elements in the i-th row is denoted by α_i, while β_j represents the average of all elements in the j-th column. The μ represents the global average of all elements in the bicluster submatrix. Thus, the MSR value represents the squared deviation of each element from the row and column averages, corrected for the global bicluster value.

A smaller MSR value indicates a higher level of coherence. Therefore, the CC algorithm aims to find a bicluster that has an MSR value smaller than a certain threshold, called the residual threshold (δ) [33]. δ can be determined by the greedy iterative search method as recommended by [34].

The bicluster search process is conducted in three main stages. The first stage is Multiple Node Deletion, which deletes a number of rows and/or columns that have the largest contribution to the increase in MSR until all remaining elements are below the threshold. The individual contribution of a row and column to the MSR is calculated by equations 2 and 3:

$$d(i) = \frac{1}{|J|} \sum_{j=1}^{J} \left(y_{ij} - \alpha_i - \beta_j + \mu\right)^2 \quad (2)$$

$$d(j) = \frac{1}{|I|} \sum_{i=1}^{I} \left(y_{ij} - \alpha_i - \beta_j + \mu\right)^2 \quad (3)$$

The values $d(i)$ and d(j) represent the residual contributions of the i-th row and j-th column, respectively. If these contribution values exceed the threshold δ, then the row or column will be removed from the submatrix.

The second stage is Single Node Deletion, which is the process of refining the results by deleting one row or one column gradually to reduce the MSR value more selectively. The third stage is Node Addition, which is the addition of rows or columns that have previously been deleted, if their contribution to the MSR value is below the δ threshold, and the addition does not cause a significant increase in the residual value.

Once the bicluster is found, all elements included in it are replaced with dummy values or noise. This is done to prevent the same elements from being detected again in the next iteration of the bicluster search [35]. The process continues iteratively until the maximum number of bicyclers is reached or until there are no other structures that satisfy the residual constraints.

2) Spectral Biclustering

Spectral Biclustering is a biclustering algorithm that uses a spectral transformation approach through singular value decomposition (SVD) to detect coherent patterns between subsets of rows and subsets of columns in a data matrix [36]. In contrast to residual-based algorithms such as CC, this approach utilizes the linear algebraic structure of the matrix and enables the detection of symmetric structures or checkerboard patterns, i.e. bidirectional coherent patterns reflecting simultaneous linkages between rows and columns.

The algorithm begins with a normalization process of the A data matrix to remove the influence of scale between rows and columns. Three commonly used normalization approaches are:

a) Independent Rescaling of Rows and Columns (IRRC): rows and columns are normalized independently using diagonal matrices R and C, where each diagonal element represents the average of the row and column,
b) Bi-stochastization: iterates IRRC until all rows and columns have uniform total values,
c) Log-interaction transformation: transforming multiplicative scaled data into additive with a logarithmic function [21].

In this study, we applied the log-interaction transformation as introduced by Kluger et al. [32], in order to linearize multiplicative effects and enhance bicluster detectability in high-dimensional settings. After the normalization process, the adjusted matrix A' is decomposed using SVD as follows by equations 4:

$$A = U\Sigma V^T \tag{4}$$

In the equation, A' is the normalized matrix of the initial data matrix A, which has been adjusted to remove the effect of scale or variability between rows and columns. The matrices U and V are the eigenvector matrices (feature vectors) that represent the main orientations of the row and column dimensions after projection to the spectral space, respectively. Meanwhile, Σ is a diagonal matrix containing singular values that reflect the relative contribution of each pair of singular vectors to the overall structure of the data.

To assess the effectiveness of the algorithm in detecting the bicluster corresponding to the inserted structure, three types of evaluation metrics are used: internal coherence, structure matching, and result stability. The metrics used refer to evaluation practices in the biclustering literature [37].

Algorithm evaluation

To measure the extent to which the biclustering algorithm is able to detect the bicluster structure embedded in the simulated data, the Liu and Wang Index (ILW) is used as the main evaluation metric measuring the quality of bicluster groups by assessing the accuracy of an algorithm to obtain the actual bicluster in the data matrix [38]. Liu and Wang index as as of conformity between detected bicluster and ground truth bicluster based on the similarity of row and column members [39]. Mathematically, ILW for a bicluster pair is defined in equation 5:

$$I_{Liu\&Wang}(M_{opt,}M) = \frac{1}{n}\sum_{i=1}^{n} max\left(\frac{[I_k \cap \hat{I}_k] + [J_k \cap \hat{J}_k\ ]}{[I_k \cup \hat{I}_k\ ] + [J_k \cup \hat{J}_k\ ]}\right) \tag{5}$$

In the equation, n denotes the number of bicluster evaluated; I_k and J_k are the set of rows and columns of the k-th bicluster embedded (ground truth); while $\hat{I}_k$ and $\hat{J}_k$ are the set of rows and columns of the bicluster detected by the algorithm. ILW calculates the average of the product of the Jaccard Index in the row and column dimensions. The larger the ILW value (close to 1), the higher the degree of agreement between the actual bicluster structure and the algorithm's detection results [19].

This method was chosen because it is able to quantitatively evaluate the performance of biclustering algorithms and take into account the two-dimensional correspondence simultaneously. In addition, ILW is flexible and can be applied to simulated data with explicitly defined ground truth.

Replication

To ensure statistical reliability, each of the 15 experimental scenarios was replicated 100 times. At each replication, a new matrix was generated with predefined collinearity and overlap structures, and both algorithms were applied independently. The obtained bicluster was compared with the ground truth using the Liu and Wang Index (ILW). Descriptive statistics, including the mean and standard deviation of ILW scores, were calculated for each condition. Comparative evaluation was performed across scenarios to identify the influence of collinearity, overlap, as well as algorithm type. Visualization and inferential analysis, including three-way ANOVA, were used to test the significance of the main effects and their interactions.

The overall simulation and analysis workflow is illustrated in Figure 1.

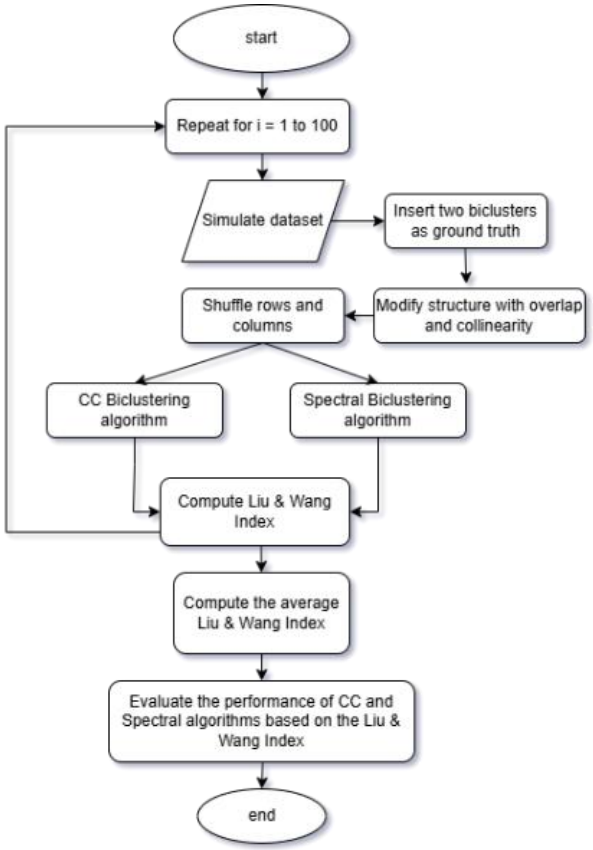


Figure 1. Flowchart of simulation data analysis stages

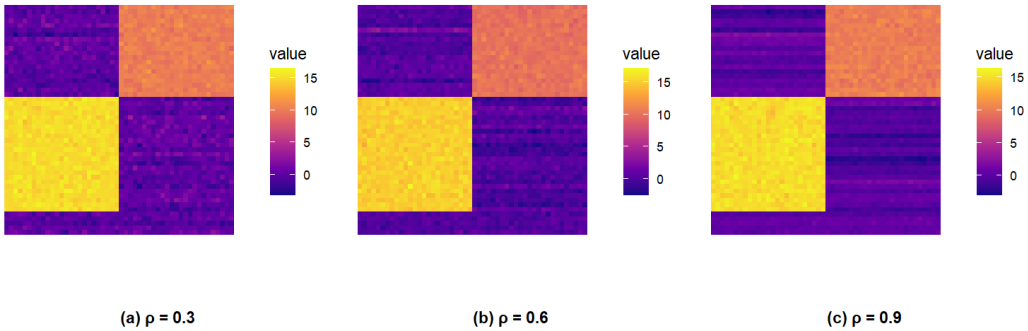
RESULTS AND DISCUSSIONS

Simulation Data Exploration

Exploration of the simulated data was conducted to ensure that the bicluster structure as well as background data characteristics, such as the collinearity between variables and the degree of bicluster overlap, were represented realistically and consistently. This process includes three main settings, namely:

Inter-variable collinearity scenario

In this scenario, two biclusters are inserted into the background data with three levels of correlation between variables, namely $\rho = 0.3, 0.6,$ and 0.9. To form the correlation structure, a covariance matrix Σ is used, which is formed based on the value of ρ and used in the generation of multivariate normal data. Each simulation matrix is 50×50 in size.



(a) ρ = 0.3 (b) ρ = 0.6 (c) ρ = 0.9

Figure 2. Bicluster visualization on background data with varied collinearity

The visualization in Figure 2 shows that the higher the correlation value between variables, the more difficult it is to distinguish the bicluster blocks from the background. This is due to the homogenization of values between columns which makes the visualization less contrasting. At ρ = 0.3, the two biclusters appear clear and contrasting. But as ρ increases to 0.9, the blocks appear to blend in more with the background.

Overlap scenario between biclusters

Overlap refers to the condition when two or more biclusters have partially overlapping row or column elements. In this simulation, overlap is controlled in three levels, namely no overlap, small overlap, and large overlap. The overlap level is not determined by the absolute number of cells, but by the proportion of overlapping rows and/or columns between two biclusters. The size and number of biclusters were kept constant, with two bicluster blocks inserted in each scenario.

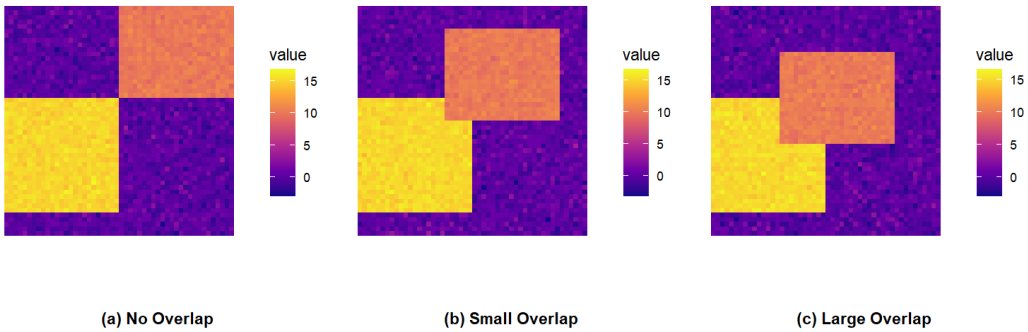


Figure 3. Visualization of bicluster structures with varying degrees of overlap

Figure 3 illustrates the effect of overlap level on the visual clarity of the bicluster structure. In the no overlap condition (panel a), the boundaries between biclusters are contrasted and clearly separated from the background. When a small overlap is introduced (panel b), some areas start to overlap with each other resulting in a smoother value transition in the overlap region. At a large overlap (panel c), the two biclusters tend to merge, resulting in a color gradation pattern that no longer shows distinct block boundaries, making the bicluster structure more difficult to visually identify.

Combination of collinearity and overlap scenario

To illustrate more complex conditions, data was simulated with a combination of collinearity levels ($\rho = 0.3, 0.6, \text{and } 0.9$) and overlap levels between biclusters. This scenario represents real conditions where variables are correlated and the data structure is not fully segregated.

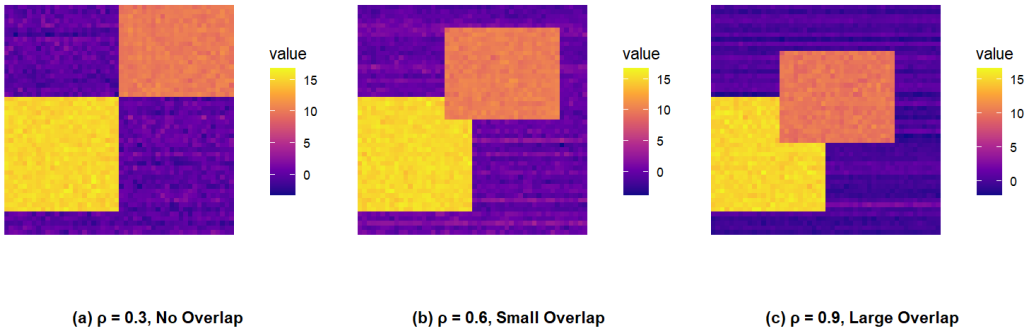


Figure 4. Visualization of bicluster structure with combination of collinearity and overlap

Figure 4 displays three representative combinations of the entire scenario for the two-bicluster configuration. In the condition of $\rho = 0.3$ without overlap (panel a), the bicluster structure is clearly visible and separated from the background. When $\rho = 0.6$ and overlap is introduced (panel b), the patterns between the biclusters start to merge. In the condition of $\rho = 0.9$ with high overlap (panel c), the bicluster structure becomes visually unrecognizable due to the combination of strong correlation and extensive

overlap. For conciseness, only these three representative scenarios are displayed in Figure 4, while the complete set of nine visualizations is available upon request.

Randomization of matrix data structure

To avoid positional bias in bicluster detection, row and column randomization is performed after insertion of the bicluster structure. This process aims to disguise the original location of the bicluster to resemble real conditions where patterns are not always in fixed positions.

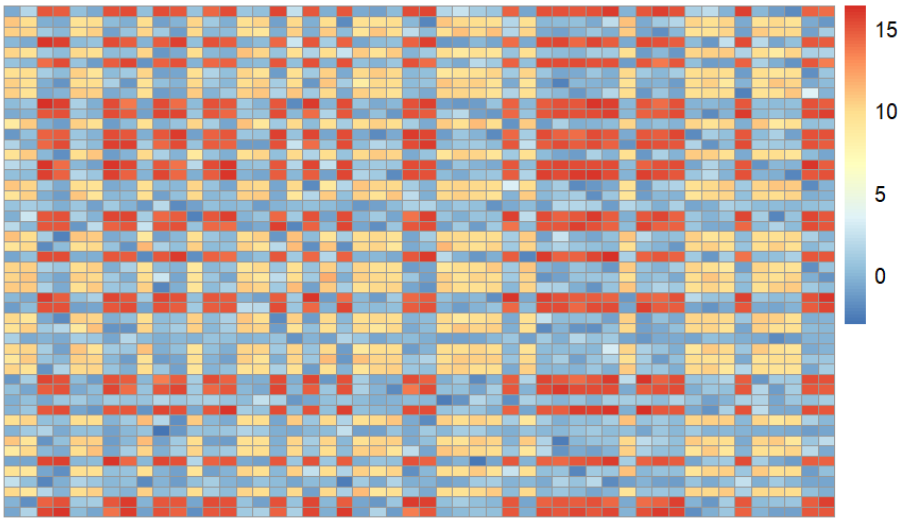


Figure 5. Visualization of bicluster structure randomization results

Figure 5 shows the results of randomization on data with two biclusters initially arranged diagonally. After permuting the rows and columns, the structure remains but is spread across the matrix area. This confirms that the biclustering algorithm detects patterns based on the structure of the values, not the geometric position. Randomization is performed using the sample() function for rows and columns independently. This process is part of the simulation standard for validating the biclustering algorithm against spatial variations in data structure.

Simulation data analysis results

Effect of collinearity on biclustering performance

Based on the simulation results at three levels of collinearity ($\rho = 0.3, 0.6, \text{and } 0.9$). It was found that the performance of both biclustering algorithms decreased as the correlation between variables increased. The average Liu & Wang Index (ILW) value shows a consistent decrease at high collinearity, especially for the CC algorithm. This reflects the sensitivity of these algorithms to information redundancy between variables that obscures coherent local patterns.

Technically, CC algorithm uses a deviation-to-mean-based Mean Squared Residue (MSR) value reduction approach, which is less effective when variables are collinear. Under such conditions, the bicluster structure formed is blurred due to fluctuations in values between variables that are not independent. In contrast, the Spectral algorithm shows better robustness as the singular decomposition (SVD) based approach is able to reduce the dimensionality and extract the principal components of complex data structures that contain correlations between variables.

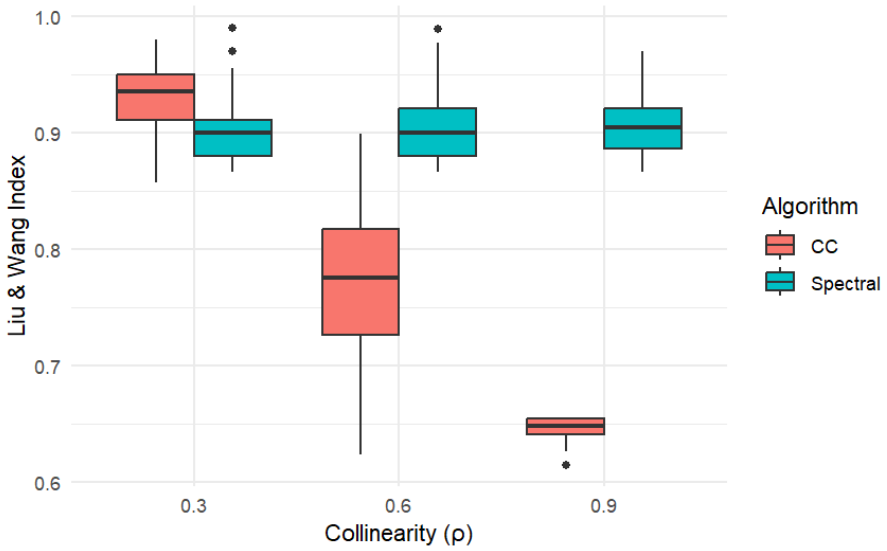


Figure 6. Distribution of ILW values against collinearity between variables

The difference in performance between the two algorithms in dealing with collinearity is shown in Figure 6, which shows the distribution of ILW values based on each level of collinearity. It can be seen that the ILW value of the CC algorithm drops sharply from 0.9306 ($\rho = 0.3$) to 0.6468 ($\rho = 0.9$). In contrast, the Spectral algorithm maintained a relatively stable value, from 0.9023 to 0.9048, even though the collinearity increased significantly.

Table 3. Mean and standard deviation of ILW values per combination of ρ and algorithm

No	Collinearity (ρ)	Algorithm	Mean ILW	SD ILW
1	0.3	CC	0.9306	0.0278
2	0.3	Spectral	0.9023	0.026
3	0.6	CC	0.7678	0.0659
4	0.6	Spectral	0.9044	0.029
5	0.9	CC	0.6468	0.0088
6	0.9	Spectral	0.9048	0.0253

Table 3 shows the mean value and standard deviation of ILW for each combination of ρ value and algorithm. The standard deviation of ILW for the Spectral algorithm ranges from 0.0253 to 0.029, indicating a stable performance in detecting biclusters at various levels of collinearity. In contrast, CC algorithm showed increased fluctuations in ILW values, characterized by the highest standard deviation of 0.0659 at $\rho = 0.6$. This finding indicates that Spectral not only provides higher average results but is also more consistent with changes in the correlation structure between variables.

Effect of overlap on algorithm performance

Simulations were conducted to evaluate the effect of bicluster overlap on the performance of the biclustering algorithm by isolating the collinearity factor. Three conditions were tested: no overlap, small overlap, and large overlap. The simulation results show that the higher the degree of overlap, the Liu & Wang Index (ILW) value tends to decrease in both algorithms, reflecting the decreased accuracy in detecting the true bicluster structure.

Table 4. Mean and standard deviation of ILW values per overlap level and algorithm

No	Overlap	Algorithm	Mean ILW	SD ILW
1	No Overlap	CC	0.9475	0.018
2	No Overlap	Spectral	0.9047	0.0304
3	Small Overlap	CC	0.8083	0.0307
4	Small Overlap	Spectral	0.775	0.0195
5	Large Overlap	CC	0.7749	0.0401
6	Large Overlap	Spectral	0.6832	8.00E-04

Table 4 shows that in the no overlap condition, the CC algorithm produces the highest ILW value of 0.9475, followed by Spectral at 0.9047. However, as the overlap increases, their performance decreases. The decline

in Spectral is sharper, from 0.9047 to 0.6832 at large overlaps. In contrast, the ILW value of CC decreased more moderately, from 0.9475 to 0.7749. This indicates that although CC is unstable, it is still able to maintain a relatively higher level of accuracy under large overlap conditions. In terms of consistency, the Spectral algorithm shows a very low standard deviation, especially at large overlaps (0.0008), while CC has a higher prediction variability (standard deviation of 0.0401). This shows that Spectral provides more distributionally stable results, despite its lower accuracy compared to CC.

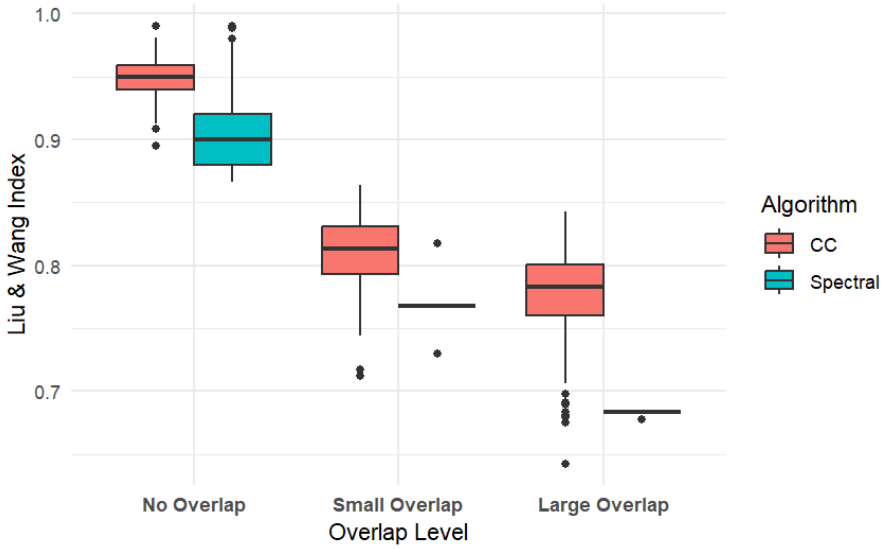


Figure 7. Distribution of ILW values against overlap level.

Figure 7 supports this finding, showing a more even distribution of ILW values in Spectral, compared to a wider spread in CC. However, the checkerboard approach used by Spectral tends to be exclusive and less adaptive to the intersection area between rows and columns. In contrast, CC algorithm utilizes an iterative mechanism of row/column deletion and addition, and is thus more flexible in handling overlaps, albeit at the cost of increased fluctuation in results. Thus, for data containing overlaps between subsets of rows and columns, CC algorithm may be a more suitable choice than Spectral. However, at large levels of overlap, the performance of both still degrades significantly.

Interaction of collinearity and overlap on algorithm performance

Table 5. Mean and standard deviation of ILW values per overlap level and algorithm

No.	Collinearity (ρ)	Overlap Type	Algorithm	Mean ILW	SD ILW
1	0.3	No overlap	CC	0.9437	0.0244
2	0.3	No overlap	Spectral	0.9031	0.0291
3	0.3	Small overlap	CC	0.7792	0.0372
4	0.3	Small overlap	Spectral	0.7739	0.0175
5	0.3	Large overlap	CC	0.6906	0.0737
6	0.3	Large overlap	Spectral	0.6797	0.0276
7	0.6	No overlap	CC	0.7716	0.0644
8	0.6	No overlap	Spectral	0.9115	0.0362
9	0.6	Small overlap	CC	0.6933	0.057
10	0.6	Small overlap	Spectral	0.7671	0.0337
11	0.6	Large overlap	CC	0.5659	0.0545
12	0.6	Large overlap	Spectral	0.6696	0.0484
13	0.9	No overlap	CC	0.6452	0.0112
14	0.9	No overlap	Spectral	0.9054	0.0287
15	0.9	Small overlap	CC	0.5685	0.0056
16	0.9	Small overlap	Spectral	0.7529	0.0657
17	0.9	Large overlap	CC	0.4985	0.0288
18	0.9	Large overlap	Spectral	0.6696	0.0301

Table 5 presents the mean and standard deviation of ILW for each combination of conditions. At low collinearity ($\rho = 0.3$), both algorithms show high performance, with CC recording an ILW of 0.9437 at no-

slice overlap. However, as the overlap increases, the performance of both degrades. The most significant decrease occurs for CC at high collinearity and large overlap, with ILW dropping to 0.4985.

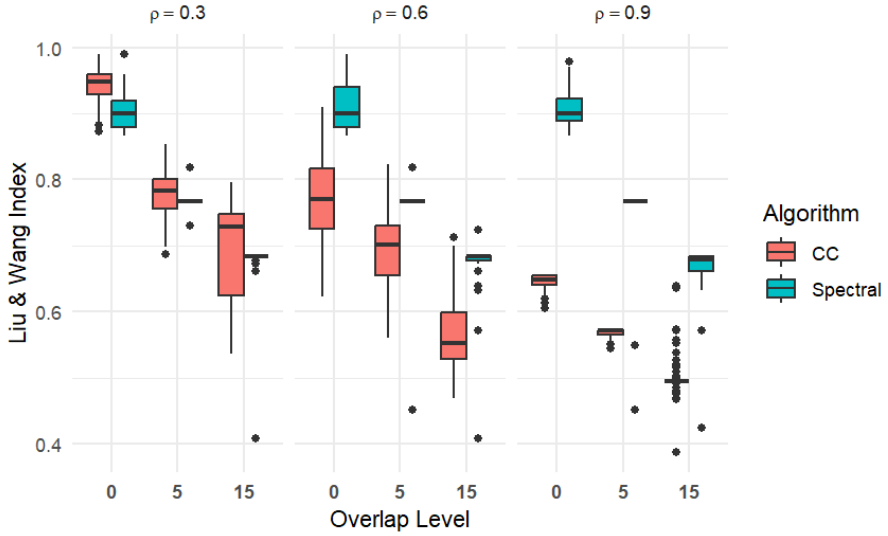


Figure 8. ILW distribution based on a combination of collinearity and overlap

Figure 8 shows the distribution of ILW across all combinations. In general, increasing collinearity and overlap reduces the algorithm's ability to accurately detect bicluster. Spectral shows relatively better robustness to high collinearity, as evidenced by the ILW values that remain high at ρ = 0.9 with low overlap. In contrast, CC excels at small overlaps, but loses accuracy when the data structure becomes complex. In Figure 7, 8 and 9 overlap levels are encoded numerically: 0 represents no overlap, 5 indicates small overlap, and 15 denotes large overlap.

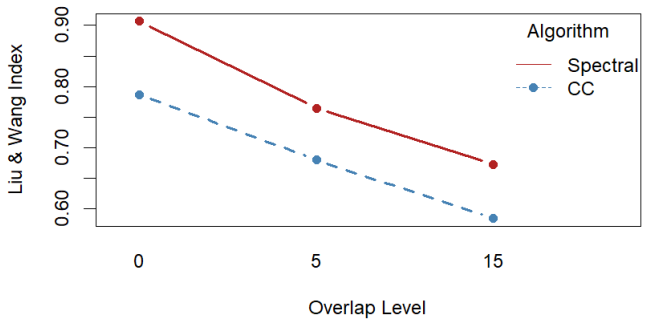


Figure 9. Algorithm interaction and overlap in collinearity combinations

The bidirectional interaction between algorithm and overlap is visualized in Figure 9, which shows the decreasing trend of ILW in both algorithms. The decrease of ILW in Spectral tends to be gentle, while CC experiences a sharper decrease. This shows that Spectral is more stable under complex combination conditions, while CC is more sensitive to high overlap, especially when combined with medium to high collinearity.

The three-way interaction pattern is visualized in Figure 10 as a mean ILW heatmap. The darkest color in the combination of high collinearity and large overlap for the CC algorithm indicates the lowest performance. In contrast, Spectral exhibits a more gradual color pattern, showing consistency, albeit with a gradual decrease.

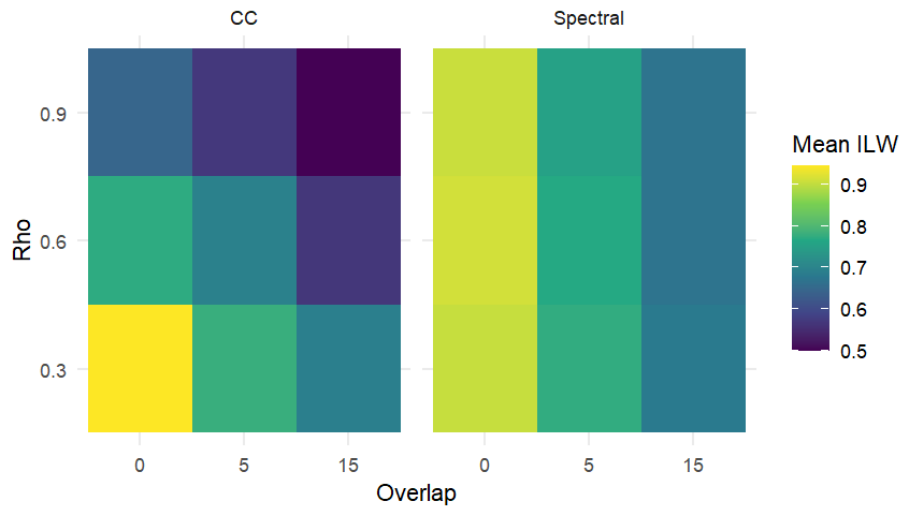


Figure 10. Heatmap of mean ILW per combination

Statistically, the results of the three-way analysis of variance (ANOVA) in Table 6 show that all the main factors (algorithm, collinearity and overlap), as well as all the interactions between these factors, have a significant effect on ILW values ($p < 0.001$). This finding indicates that the accuracy of biclustering algorithms is affected by a complex combination of the correlation between variables and the degree of overlap between biclusters.

These results have practical implications for various domains that require local pattern detection, such as biomedical signal analysis, market segmentation, and climate data. In such contexts, understanding the sensitivity of algorithms to overlapping and highly correlated data structures can be an important reference in determining preprocessing strategies and selecting appropriate algorithms. However, the limitation of this study lies in the use of synthetic data. Therefore, further validation using real-world data is necessary to ensure the generalizability of the results.

Table 6. Results of three-way ANOVA on ILW values

Source of Variation	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Algorithm	1	4.267	4.267	2437.74	< 2e-16
Collinearity (ρ)	2	4.448	2.224	1270.61	< 2e-16
Overlap	2	14.319	7.16	4090.13	< 2e-16
Algorithm × Collinearity	2	3.786	1.893	1081.51	< 2e-16
Algorithm × Overlap	2	0.115	0.057	32.74	1.09E-14
Collinearity × Overlap	4	0.172	0.043	24.54	< 2e-16
Algorithm × Collinearity × Overlap	4	0.263	0.066	37.51	< 2e-16
Residuals	1782	3.119	0.002		

Based on the overall findings, Table 7 is organized as a practical guide for algorithm selection. Spectral is recommended for data with high collinearity and low overlap, due to its distributional stability. Meanwhile, CC is more suitable for data with small overlap and moderate collinearity. Under extreme conditions (high collinearity and large overlap), the performance of both algorithms degrades significantly, so they are not recommended without additional preprocessing.

Table 7. Algorithm recommendations based on collinearity and overlap combination conditions

Collinearity (ρ)	Overlap Type	Recommended Algorithm
0.3	No overlap	CC or Spectral
0.6	Small overlap	CC
0.9	No overlap	Spectral
0.9	Large overlap	Not recommended
0.3	Large overlap	CC

The Results and Discussion sections should be integrated concisely and clearly to avoid redundancy and extensive repetition of findings. The discussion must emphasize the implications and significance of the analytical outcomes rather than restating the numerical results explicitly. Tables and graphs presented must

each highlight unique aspects of the analysis. Furthermore, statistical analyses should robustly address the research questions posed. In discussing findings, references should not duplicate citations previously introduced in the background section. Additionally, comparisons with relevant prior research findings should be explicitly incorporated to contextualize the study within existing literature.

CONCLUSION

This study evaluates the performance of two biclustering algorithms, namely Cheng & Church (CC) and Spectral, in detecting bicluster structures in data with varying degrees of collinearity and overlap. Simulations were conducted by inserting two constant biclusters into the background data matrix, and then tested with various combinations of correlations between variables ($\rho = 0.3$; 0.6; and 0.9) and the degree of overlap between rows or columns (none, small, and large). Performance evaluation uses the Liu & Wang Index (ILW) as a measure of accuracy in reconstructing the true bicluster structure. The results show that collinearity and overlap, both individually and interactively, have a significant influence on the accuracy of the algorithm. Spectral shows better robustness to high collinearity, especially when the overlap is low, as the Singular Value Decomposition (SVD) based approach is able to extract principal components from high-dimensional data. However, its performance drops drastically when the overlap is high, as the checkerboard pattern used is less flexible in recognizing overlapping regions. In contrast, the CC algorithm excels on data with small overlap, although it is more sensitive to high collinearity. The Mean Squared Residue (MSR) based iterative mechanism makes it adaptive to noise and local variations, but also triggers performance fluctuations when the data structure becomes complex.

This finding was reinforced by the results of the three-way ANOVA, which showed that all the main factors and their interactions significantly affected the ILW values ($p < 0.001$). This confirms that the effectiveness of biclustering algorithms not only depends on the method used, but is also strongly influenced by the overall structural characteristics of the data. In addition to comparing the performance of the algorithms, this finding also provides greater insight into how the structural complexity of the data affects the success of local pattern detection. These implications are important in fields such as bioinformatics, environmental modeling and consumer behavior analysis, where data structures are often overlapping and correlated.

Practically speaking, the Spectral algorithm is recommended for data with high collinearity and low overlap as it provides stable and reliable results. Meanwhile, the CC algorithm is more suitable for data with moderate collinearity and small overlap which requires more structural flexibility. Under extreme conditions (high collinearity and large overlap), both algorithms show significant performance degradation, requiring additional preprocessing steps such as dimensionality reduction or redundant feature removal.
For future research, it is necessary to validate these findings on real-world data to ensure the generalizability of the results to more complex and uncontrolled conditions. In addition, the development of hybrid methods that combine Spectral robustness with the flexibility of iterative algorithms such as CC is a potential step to improve the effectiveness of biclustering in dealing with overlapping and collinear data structures simultaneously.

REFERENCES

[1] M. M. A. Waqar, *A personalized reinforcement learning recommendation algorithm using bi-clustering techniques*. 2025. doi: 10.1371/journal.pone.0315533.

[2] M. Sundari, P. R. Sihombing, and K. F. Hakim, "Perbandingan Metode Analisis Gerombol K-Rataan Dan Bicluster," *Lomb. J. Sci.*, vol. 3, no. 1, pp. 1–11, 2021.

[3] P. Patowary, R. Sarmah, and D. K. Bhattacharyya, "Developing an effective biclustering technique using an enhanced proximity measure," *Netw. Model. Anal. Heal. Informatics Bioinforma.*, vol. 9, no. 1, 2020, doi: 10.1007/s13721-019-0211-7.

[4] J. Kléma, F. Malinka, and F. železný, "Semantic biclustering for finding local, interpretable and predictive expression patterns," *BMC Genomics*, vol. 18, no. Suppl 7, 2017, doi: 10.1186/s12864-017-4132-5.

[5] W. Zhang *et al.*, "Robust Integrative Biclustering for Multi-view Data".

[6] V. A. Padilha and R. J. G. B. Campello, "A systematic comparative evaluation of biclustering techniques," *BMC Bioinformatics*, vol. 18, no. 1, pp. 1–25, 2017, doi: 10.1186/s12859-017-1487-1.

[7] I. M. Afnan, H. Wijayanto, and A. H. Wigena, "Identifying Poverty Vulnerability Patterns in Indonesia using Cheng and Chruch ' s Algorithm," vol. 8, no. 4, pp. 1262–1277, 2024.

[8] H. A. Majd *et al.*, "Evaluation of Plaid Models in Biclustering of Gene Expression Data," vol. 2016,

2016, doi: 10.1155/2016/3059767.
[9] M. L. P and I. M. Sumertajaya, "Evaluasi Kinerja Spectral Biclustering dalam Identifikasi Potensi Produksi Komoditas Hortikultura di Indonesia," vol. 21, no. 3, pp. 365–382, 2024.
[10] S. Cao *et al.*, "Pipeline for characterizing alternative mechanisms (PCAM) based on bi-clustering to study colorectal cancer heterogeneity," *Comput. Struct. Biotechnol. J.*, vol. 21, pp. 2160–2171, 2023, doi: 10.1016/j.csbj.2023.03.028.
[11] O. Maâtouk, W. Ayadi, H. Bouziri, and B. Duval, "Evolutionary Local Search Algorithm for the biclustering of gene expression data based on biological knowledge," *Appl. Soft Comput.*, vol. 104, p. 107177, 2021, doi: 10.1016/j.asoc.2021.107177.
[12] M. G. Silva, S. C. Madeira, and R. Henriques, "Water Consumption Pattern Analysis Using Biclustering: When, Why and How," *Water (Switzerland)*, vol. 14, no. 12, Jun. 2022, doi: 10.3390/w14121954.
[13] C. Pang, "Construction and Analysis of Macroeconomic Forecasting Model Based on Biclustering Algorithm," *J. Math.*, vol. 2022, 2022, doi: 10.1155/2022/7768949.
[14] P. A. Kaban, R. Kurniawan, R. E. Caraka, B. Pardamean, B. Yuniarto, and Sukim, "Biclustering Method to Capture the Spatial Pattern and to Identify the Causes of Social Vulnerability in Indonesia: A New Recommendation for Disaster Mitigation Policy," *Procedia Comput. Sci.*, vol. 157, pp. 31–37, 2019, doi: 10.1016/j.procs.2019.08.138.
[15] B. Wang, Y. Miao, H. Zhao, J. Jin, and Y. Chen, "A biclustering-based method for market segmentation using customer pain points," *Eng. Appl. Artif. Intell.*, vol. 47, pp. 101–109, 2016, doi: 10.1016/j.engappai.2015.06.005.
[16] F. Divina, F. A. G. Vela, and M. G. Torres, "Biclustering of smart building electric energy consumption data," *Appl. Sci.*, vol. 9, no. 2, 2019, doi: 10.3390/app9020222.
[17] S. Babichev, V. Osypenko, V. Lytvynenko, M. Voronenko, and M. Korobchynskyi, "Comparison Analysis of Biclustering Algorithms with the use of Artificial Data and Gene Expression Profiles," *2018 IEEE 38th Int. Conf. Electron. Nanotechnology, ELNANO 2018 - Proc.*, no. November, pp. 298–304, 2018, doi: 10.1109/ELNANO.2018.8477439.
[18] E. N. Castanho, H. Aidos, and S. C. Madeira, "Biclustering fMRI time series: a comparative study," *BMC Bioinformatics*, vol. 23, no. 1, Dec. 2022, doi: 10.1186/s12859-022-04733-8.
[19] N. Kavitha Sri and R. Porkodi, "An extensive survey on biclustering approaches and algorithms for gene expression data," *Int. J. Sci. Technol. Res.*, vol. 8, no. 9, pp. 2228–2236, 2019.
[20] E. N. Castanho, J. P. Lobo, R. Henriques, and S. C. Madeira, "Correction: G-bic: generating synthetic benchmarks for biclustering (BMC Bioinformatics, (2023), 24, 1, (457), 10.1186/s12859-023-05587-4)," *BMC Bioinformatics*, vol. 25, no. 1, p. 12859, 2024, doi: 10.1186/s12859-023-05628-y.
[21] R. Henriques and S. C. Madeira, "Biclustering with Flexible Plaid Models to Unravel Interactions between Biological Processes," *IEEE/ACM Trans. Comput. Biol. Bioinforma.*, vol. 12, no. 4, pp. 738–752, Jul. 2015, doi: 10.1109/TCBB.2014.2388206.
[22] X. Xu, S. Zhang, J. Guo, and T. Xin, "Biclustering of Log Data: Insights from a Computer-Based Complex Problem Solving Assessment," *J. Intell.*, vol. 12, no. 1, 2024, doi: 10.3390/jintelligence12010010.
[23] G. Khalaf and M. Iguernane, "Multicollinearity and a ridge parameter estimation approach," *J. Mod. Appl. Stat. Methods*, vol. 15, no. 2, pp. 400–410, 2016, doi: 10.22237/jmasm/1478002980.
[24] C. Nnamani and N. Ahmad, "Biclustering Models Under Collinearity in Simulated Biological Experiments," *Matematika*, vol. 39, no. 3, pp. 227–238, 2023, doi: 10.11113/matematika.v39.n3.1461.
[25] J. Li, J. Reisner, H. Pham, S. Olafsson, and S. Vardeman, "Biclustering with missing data," *Inf. Sci. (Ny).*, vol. 510, pp. 304–316, 2020, doi: 10.1016/j.ins.2019.09.047.
[26] M. G. Silva, S. C. Madeira, and R. Henriques, "A Comprehensive Survey on Biclustering-based Collaborative Filtering," *ACM Comput. Surv.*, vol. 56, no. 12, 2024, doi: 10.1145/3674723.
[27] M. Bentert, P. G. Drange, and E. Haugen, "Overlapping Biclustering," pp. 1–16, 2025.
[28] M. Alzahrani, H. Kuwahara, W. Wang, and X. Gao, "Gracob: A novel graph-based constant-column biclustering method for mining growth phenotype data," *Bioinformatics*, vol. 33, no. 16, pp. 2523–2531, 2017, doi: 10.1093/bioinformatics/btx199.
[29] M. B. Ferraro, P. Giordani, and M. Vichi, "A class of two-mode clustering algorithms in a fuzzy setting," *Econom. Stat.*, vol. 18, pp. 63–78, 2021, doi: 10.1016/j.ecosta.2020.03.006.
[30] Y. Cheng and G. M. Church, "Biclustering of Expression Data," *Proc. 8th Int. Conf. Intell. Syst. Mol. Biol. ISMB 2000*, no. February 2000, pp. 93–103, 2000.

[31] E. N. Castanho, H. Aidos, and S. C. Madeira, "Biclustering data analysis : a comprehensive survey," vol. 25, no. 4, 2024.
[32] A. López-Fernández, D. S. Rodríguez-Baena, and F. Gómez-Vela, "gMSR: A multi-GPU algorithm to accelerate a massive validation of biclusters," *Electron.*, vol. 9, no. 11, pp. 1–15, 2020, doi: 10.3390/electronics9111782.
[33] B. Pontes, R. Giráldez, and J. S. Aguilar-Ruiz, "Biclustering on expression data: A review," *Journal of Biomedical Informatics*, vol. 57. Academic Press Inc., pp. 163–180, Oct. 2015. doi: 10.1016/j.jbi.2015.06.028.
[34] J. Di Iorio, F. Chiaromonte, M. A. Cremona, and M. A. Cremona, "On the bias of H-scores for comparing biclusters, and how to correct it," *Bioinformatics*, vol. 36, no. 9, pp. 2955–2957, 2020, doi: 10.1093/bioinformatics/btaa060.
[35] İ. ÇİL, S. G. ÇAKAR, N. SARI, and O. EYDEMİR, "İkili Kümeleme Yaklaşımıyla Suç Bölgelerinin Tespiti ve İkili Kümeleme Yöntemlerinin Karşılaştırılması," *Sak. Univ. J. Comput. Inf. Sci.*, vol. 2, no. 3, pp. 145–157, 2019, doi: 10.35377/saucis.02.03.648342.
[36] Y. Kluger, R. Basri, J. T. Chang, and M. Gerstein, "Biclustering Spektral dari Data Microarray : Gen dan Kondisi Coclustering," 2024.
[37] S. Sun and K. C. Das, "On the second largest normalized Laplacian eigenvalue of graphs," *Appl. Math. Comput.*, vol. 348, pp. 531–541, 2019, doi: 10.1016/j.amc.2018.12.023.
[38] H. Ben Saber and M. Elloumi, "A Comparative Study of Clustering and Biclustering of Microarray Data," *J. Eng. Technol.*, vol. 6, no. 2, pp. 239–257, Dec. 2016, doi: 10.21859/jet-060223.
[39] H. Ben Saber and M. Elloumi, "A New Survey on Biclustering of MicroArray Data," pp. 165–183, 2014, doi: 10.5121/csit.2014.41314.