Rubin McLuen

Dr. Rahul Singh

CS:4980:0003

6 May 2024

Identifying Coregulated Yeast Genes Using Graph-theoretic Methods

*Final Project Report*

Abstract

This study assesses a range of similarity measures and graph algorithms for their effectiveness in identifying coregulated genes within a yeast gene expression matrix, utilizing a spectral community detection algorithm. The coherence of the resulting clusters was evaluated using figure of merit, which quantitatively measures the predictive power of the clusters. The methodology used in this study resulted in a quantitative comparison of the efficacy of various combinations of similarity and graph algorithms in discerning groups of coregulated genes within the dataset. A discussion of the meaning behind the relative performance of the methods tested is included.

Work Done After Final Presentation

Following the final presentation, an error was identified in the implementation of the Gabriel Graph algorithm. Initially, the algorithm incorrectly utilized Euclidean Distance to determine the circles for node connections, rather than using the appropriate distance measure specific to each distance matrix. This discrepancy resulted in inaccurate figures of merit for all Gabriel graphs, with the exception of those based on Euclidean Distance. This issue has been fixed, ensuring that the figure of merit for clusterings derived from Gabriel Graphs across all similarity measures are accurate. Unfortunately though, there was not enough time to run the Gabriel Graph algorithm again for Mahalanobis Distance and Squared Correlation, as that code takes an extremely long time.

Additionally, new figures have been created for inclusion in this report. The most significant of these is the table displaying the figure of merit of all methods, which represents the core results of the study.

Introduction

Identifying coregulated genes within an organism provides useful insights into the cellular and genetic processes of that organism. Coregulated genes are those that have similar expression levels under various conditions, suggesting a related functional role. Clustering a gene expression matrix is a method used to identify such relationships by assessing gene similarities and grouping genes with higher mutual similarity apart from less similar groups. However, the effectiveness of this method varies based on the approach and the data used. This study aims to quantitatively assess different similarity measures and graph algorithms for the accuracy of their role in clustering coregulated genes. The dataset that this study is based upon is a yeast gene expression matrix provided by S. Lonardi and Y. Cheng from UC Riverside.

To begin, there is an abundance of different methods that can be used to calculate similarity between genes, which may be distance-based or correlation based. The choice of the appropriate measure is highly dependent on the dataset and the relationships being examined. This study compares a wide range of these different measures.

Following the calculation of gene similarity, numerous clustering methods can be applied. Popular techniques include K-means Clustering, Self-Organizing Maps (SOMs), and Density-Based Spatial Clustering of Applications with Noise (DBSCAN), among countless others. The selection and effectiveness of a clustering algorithm are also contingent on the dataset and objectives of the study. This study specifically explores community detection: a form of clustering that involves constructing a graph based on the calculated gene similarities according to specific rules, and then applying a community detection algorithm to this graph to identify clusters. This study evaluates three different graph types using a singular community detection algorithm to compare the efficacy of each graph type.

After forming clusters of genes, it is essential to validate these clusters to confirm whether the genes are truly coregulated. Validation of this form can be external or internal. External validation involves comparing clusters against known biological data, using metrics such as the Rand Index to quantify the similarity between the resulting clusters and the established gene groups. However, such data is often not available in research settings. As a result, researchers typically are forced to rely on internal validation. Internal validation is the process of evaluating the coherence of the clusters based on the data itself. This study concentrates on internal validation, employing the framework described in “Validating Clustering for Gene Expression Data” by K. Y. Yeung, D. R. Haynor, and W.L. Ruzzo. This approach is used to assess the accuracy of the clusters derived from the various combinations of methods used in this experiment on the yeast gene expression dataset.

Experiment

*Overview*

This section describes the entire methodology used to generate the results of this study. Initially, the dataset was normalized to ensure uniformity in gene expression levels. Subsequent steps involved calculating similarity between genes using various methods, constructing different types of neighborhood graphs based on these similarity measures, and applying a spectral community detection algorithm to identify clusters of similar genes. Internal validation was performed by calculating the figure of merit for the clusters, with results generated for a range of cluster numbers from one to twenty.

*Data*

The dataset used for this research is a yeast gene expression matrix containing data for 2,882 yeast genes across 17 different conditions, which correspond to various stages of the yeast cell cycle. Each entry in the matrix denotes the expression level of a gene under a specific condition. The primary objective of this study is to evaluate the effectiveness of different methodologies in identifying coregulated genes from this dataset. This dataset is particularly suited for such an analysis because it includes groups of genes that exhibit highly similar expression levels across all conditions, indicating coregulation.

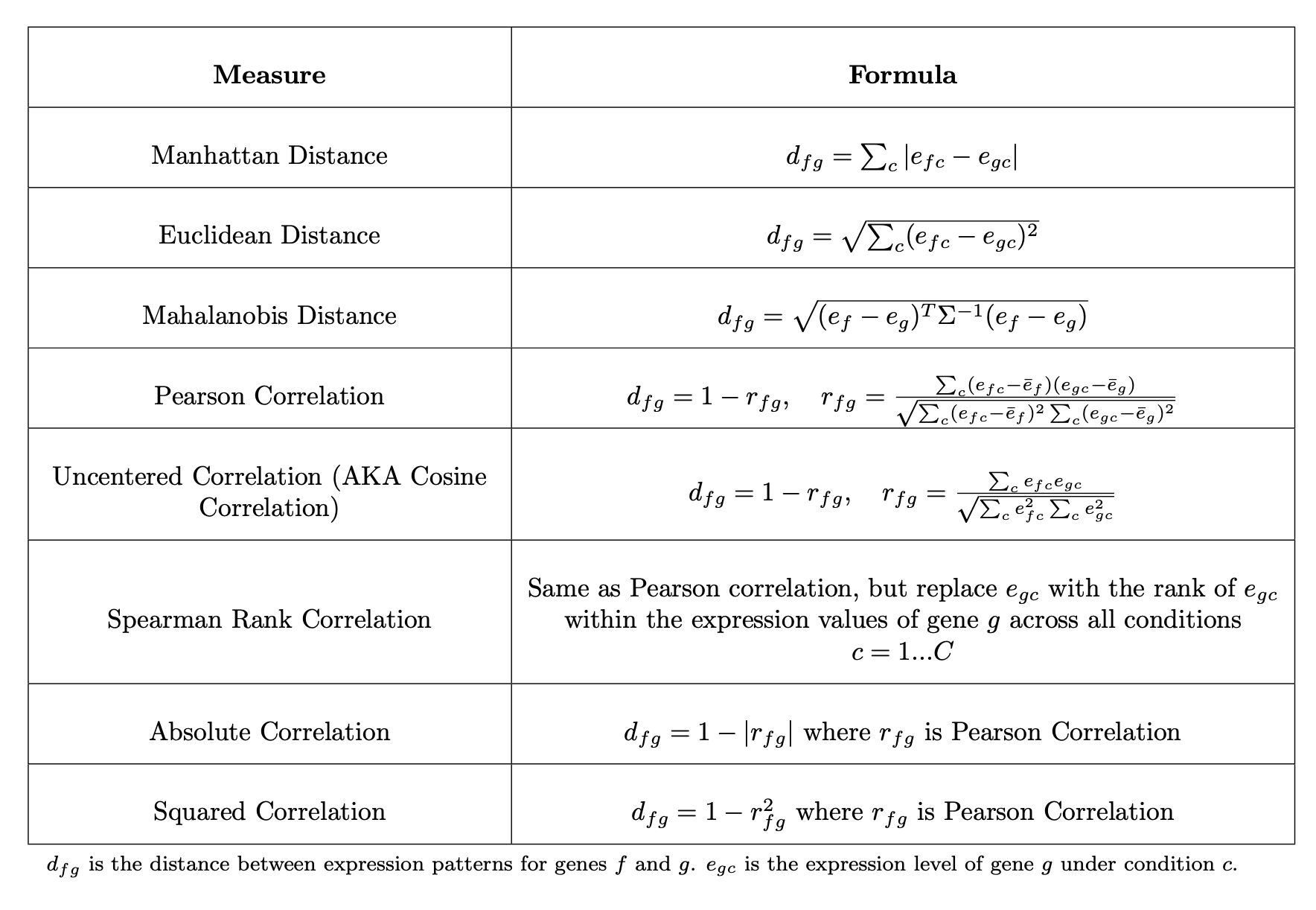
*Normalization*

The yeast gene expression matrix was standardized using z-score normalization, setting the mean to zero and variance to one. This step is done to ensure that gene expression levels could be compared accurately across different conditions and genes, negating potential inconsistency from data collection.

*Similarity Measures*

This study implemented eight different similarity measures, each quantifying the similarity between a pair of genes based on their expression levels across all conditions in the dataset. Each measure synthesizes this information into a single value representing the similarity between two genes. Both distance-based and correlation-based measures were implemented. Notably, correlation-based measures were adjusted by subtracting from the maximum value to effectively treat all measures as linear distance between two genes for clustering purposes. In this context, a score of 0 indicates perfect similarity according to the measure, with increasing scores representing greater dissimilarity. The specific measures selected and their corresponding formulas were derived from those listed in Patrik D’haeseleer’s paper, “How Does Gene Expression Clustering Work?”.

The eight similarity measures used in this study are given in the following table:



*Table 1: Measures of Similarity (Note: This table is a direct recreation of the one given in D’Haeseleer’s paper)*

In Table 1, the similarity measures are categorized as either distance or correlation. Three measures are distance based (Manhattan, Euclidean, and Mahalanobis), while five are correlation based (Pearson, Uncentered, Spearman, Absolute, and Squared). All of the correlation based measures are subtracted from one to be treated effectively as a distance metric.

Euclidean and Pearson Correlation are typically the most popular similarity measures used in clustering applications. However, each measure on this list highlights a unique relationship which may be more or less effective depending on the dataset and the objectives of the clustering.

Using these measures, a distance matrix was computed for each one, representing the similarity of every gene to every other gene in the dataset. Conceptually, this matrix can be envisioned as a complete graph where each gene is connected to every other gene via a weighted edge, with the weight signifying the distance between the two genes. The distance matrix serves as the foundation for constructing neighborhood graphs in the next step of the methodology.

*Neighborhood Graphs*

For this study, algorithms for three types of neighborhood graphs were implemented: Relative Neighbor Graph (RNG), Gabriel Graph (GG), and Connected-k-Nearest Neighbors Graph (c-kNNG). Each of these algorithms utilizes the distance matrix created from a similarity measure to establish unweighted, undirected edges between nodes, which represent genes within the dataset. Each neighborhood graph is created by following a specific rule that is detailed in the following subsections. The implementation of Relative Neighbor Graph and Gabriel Graph were inferred from the paper, “Relative neighborhood graphs and their relatives” (Jaromczyk, J.W. and G.T. Toussaint).

*Relative Neighbor Graph*

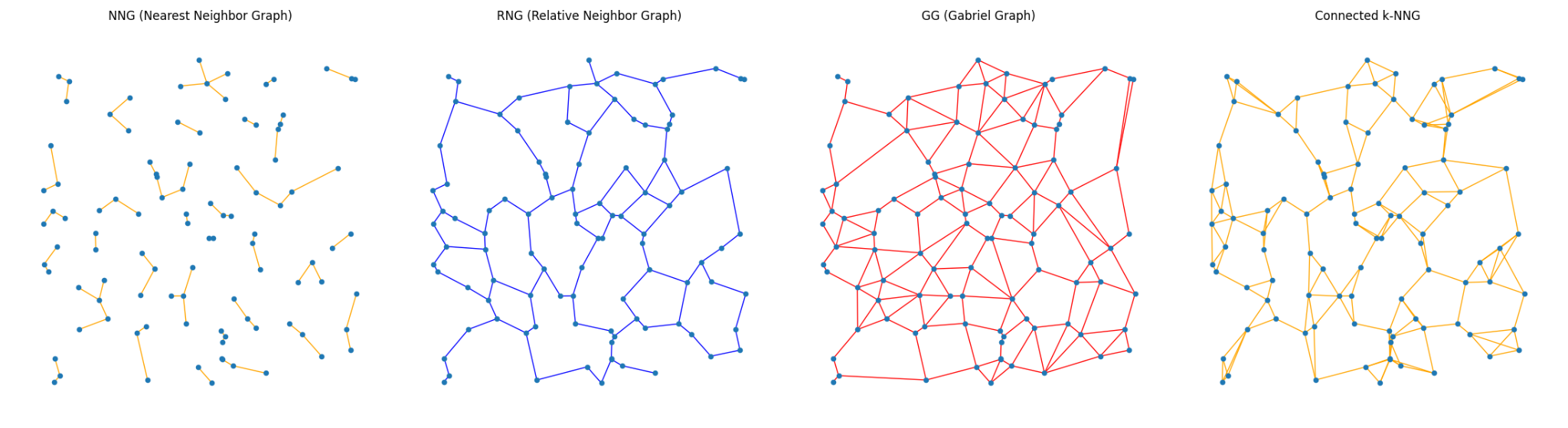
A Relative Neighbor Graph, or RNG, connects each pair of genes if there is no third gene that is closer to either of the genes than they are to each other. This criterion ensures that the edges in an RNG exclusively connect pairs that are relatively closer to each other compared to any other genes. Relative Neighbor Graphs are characterized by lower edge density, but they are effective in preserving the most important topological properties of the dataset. Notably, an RNG is a subset of a Gabriel Graph.

*Gabriel Graph*

A Gabriel Graph, or GG, establishes an edge between each pair of genes in the dataset if the circle whose diameter is the line segment between those two genes contains no other interfering genes. The space within this circle is defined using the same similarity measure used in constructing the graph. This criterion ensures that each connection in the GG represents a direct proximity between gene pairs without any other genes interfering within the specified circle. Gabriel Graphs typically have a relatively high edge density. A GG is a superset of an RNG, so it contains all of the edges and potentially more, as its criteria for forming edges is less strict.

*Connected k-Nearest Neighbor Graph*

A Connected k-Nearest Neighbor Graph, or c-kNNG, is constructed by sequentially connecting each node to its k-th nearest neighbor. The process begins with k=1 and k is increased progressively until the graph becomes connected. The construction halts at this point, resulting in a graph that has high connectivity among nodes that are close together, with sparse connections between more distant nodes. Unlike RNG and GG, a c-kNNG does not have a direct relation to these graphs. However, if the process were stopped at k=1, it would result in a Nearest Neighbor Graph (NNG), which is a subgraph of RNG and GG. However, an NNG usually yields a disconnected graph, making c-kNNG a more appropriate choice for this study, as it ensures that all components are connected for the community detection algorithm.



*Figure 1: Demonstration of Neighborhood Graphs on Random Points*

Figure 1 displays the structure of all of the graphs previously described for demonstrational purposes. The graphs in the image are based off of the Euclidean Distance between 100 random points. Note that NNG is not included in this study.

Utilizing the distance matrices derived from each of the eight different similarity measures of the yeast expression matrix, the three different types of neighborhood graphs were constructed for each measure. This resulted in a total of twenty-four different graphs. These combinations form the core subject of the comparative analysis in this report. Each of these twenty-four graphs is used as the basis for the next phase of the methodology: community detection.

*Community Detection*

In order to compare the various similarity measures and graph algorithms for this study, several strict requirements had to be met. This necessitated the selection of a very particular type of algorithm capable of identifying clusters. Only a subset of clustering algorithms known as community detection algorithms fit this requirement, as they form clusters based on the structure of a graph. The chosen algorithm was required to generate accurate clusters to allow for consistent comparison across different methods. Additionally, it needed to be very fast, given that the input graphs could contain upwards of millions of edges. Finally, the algorithm had to allow for the direct specification of the number of resultant clusters, a feature necessary for the type of internal validation performed later in this report.

With these criteria in mind, a spectral community detection algorithm was selected, based on the approach described in “A Tutorial on Spectral Clustering” by Ulrike von Luxburg. This algorithm was chosen for its ability to quickly form clusters in a lower-dimensional space, its capability to produce relatively accurate clusters, and its flexibility to output a predefined number of clusters.

*Community Detection Algorithm Detailed Explanation*

1. The algorithm takes as input a graph and a number *k*, where *k* represents the number of desired clusters. The graph in this case is one of the three neighborhood graphs previously described; a set of unweighted, undirected edges between genes.
2. The first step of the algorithm is calculating the Laplacian matrix of the graph, denoted as *L*. This matrix is obtained from the degree matrix *D* and the adjacency matrix *A*, where *L = D - A*. The degree matrix *D* is diagonal, with each diagonal element *Dii* equaling the sum of the weights of edges connected to node *i*. The adjacency matrix *A* captures the weights of edges between nodes. However, the graphs constructed in this study are unweighted and undirected, so the weights are set equal to one.
3. Next, the algorithm performs an eigenvalue decomposition on the Laplacian matrix. This step calculates the eigenvalues and their corresponding eigenvectors.
4. Following this, the first *k* eigenvectors associated with the smallest *k* non-zero eigenvalues are selected from the decomposition. These eigenvectors are chosen because they highlight the graph’s significant structural features in terms of the minimum separation between clusters.
5. The selected eigenvectors create a new feature matrix, where each row represents a gene and each column corresponds to one of the k selected eigenvectors. This matrix transforms the genes from the original graph representation into a *k*-dimensional Euclidean space, which makes it very fast and easy for traditional clustering algorithms to identify clusters.
6. Finally, *k*-means clustering is applied to the rows of the feature matrix. Each row, representing a gene in the transformed space, is grouped into *k* clusters based on their Euclidean distances within this new space. The resulting *k* clusters ideally represent meaningful subdivisions of the original graph, reflecting its inherent connectivity structure.

*Validation*

*Overview*

After forming clusters using various combinations of similarity measures and graph algorithms, internal validation was conducted using the approach detailed in "Validating clustering for gene expression data" by K. Y. Yeung, D. R. Haynor, and W. L. Ruzzo. This method assesses the predictive power of clusters by removing a condition from the dataset, clustering the remaining data, and then predicting the values of the omitted condition. The effectiveness of this prediction is quantified as the figure of merit, which has been shown to correlate with a higher Rand index when compared to established biological groups through external validation.

*Removing a Condition*

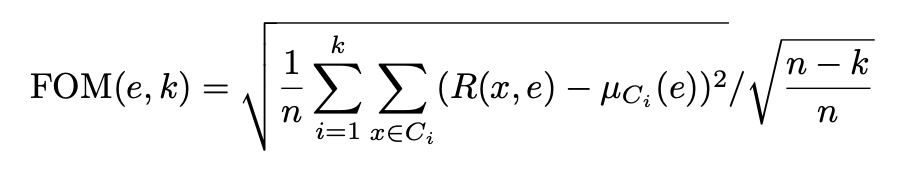
The actual procedure described in the original paper involves calculating the aggregate figure of merit. This is the process of sequentially removing each condition from the dataset, calculating the figure of merit for each removal and then summing these values. Due to computation restraints, as the Gabriel Graph algorithm is extremely slow, this study modified the approach by removing only one condition for all tests. Condition 5 was arbitrarily selected for removal. This adjustment is considered approximately accurate, as the authors of the referenced paper asserted that clusters formed by omitting any condition yield clusters similar to those obtained by removing any other condition, or the full dataset.

*Forming Clusters*

Upon removing condition five from the dataset, the process of forming clusters was executed on the remaining data. This involved calculating the distance matrix for all genes using the different similarity measures. Subsequently, graphs were constructed based on these distance matrices. Finally, the spectral community detection algorithm was applied to these graphs to form clusters.

*Figure of Merit*

After forming the clusters, the figure of merit for those clusters was calculated using the following equation:



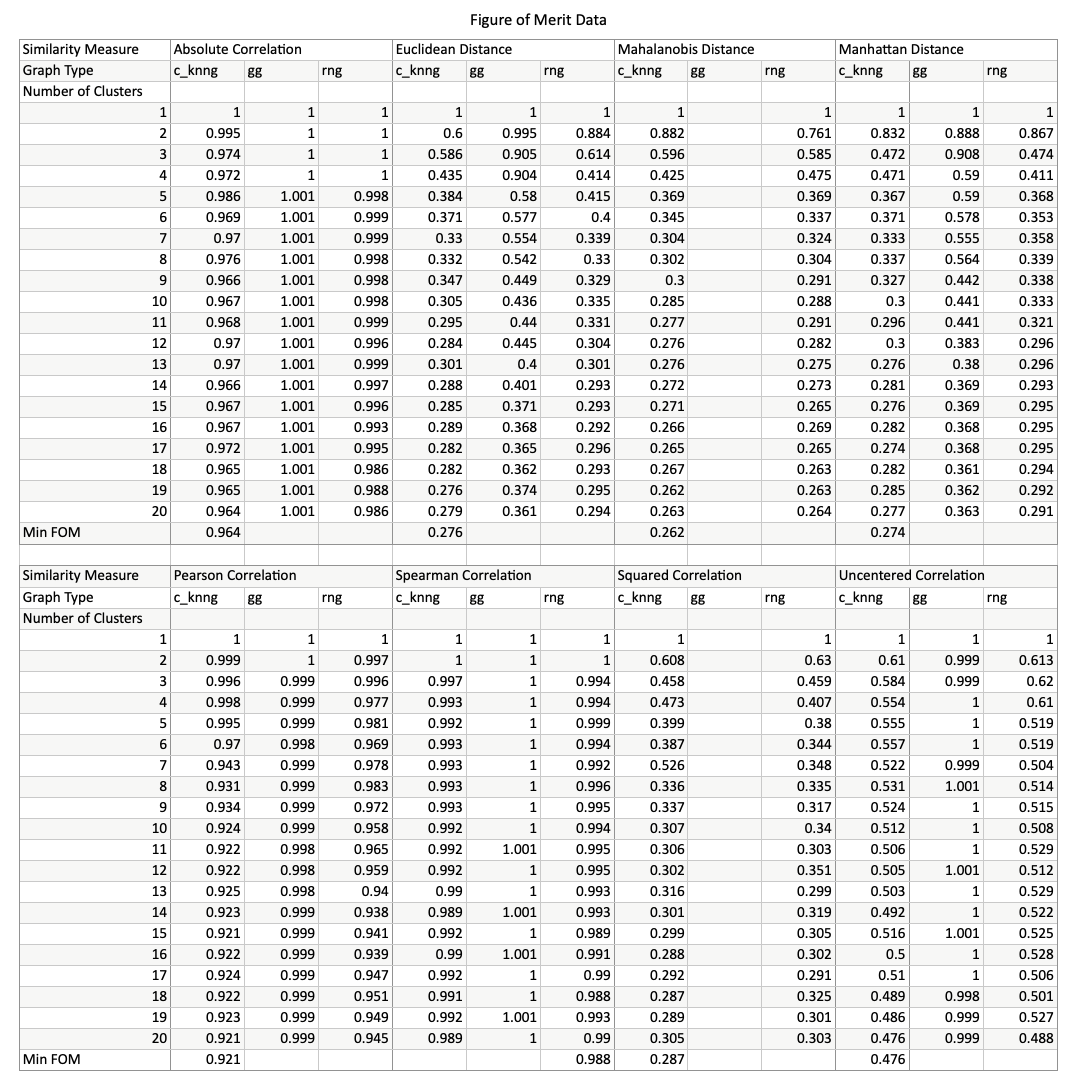
In this equation, *e* represents the condition used for validation, which in this study is condition five, the condition that was removed. *k* stands for the number of clusters, and *n* is the total number of genes. denotes cluster *i*, *R(x,e)* is the expression level of gene *x* under the validation condition *e* in the original expression matrix, and is the mean expression level of cluster *i* under validation condition *e*. The formula includes a division by the square root of , an adjustment to account for the influence of increasing the number of clusters on the Figure of Merit, which would otherwise always reduce it.

Figure of merit provides a measurement of the predictive power of clusters generated from the experiment. Ideally, a figure of merit of 1 indicates clusters that are completely random, and a score of 0 would suggest that the clusters are perfectly predictive of the omitted condition, *e*. The authors of the referenced paper for this procedure demonstrated that there is a strong negative correlation between the figure of merit and the Rand Index of external validation datasets, making it a valuable measure for validating whether the resulting clusters contain coregulated genes.

A limitation of the figure of merit is that it is only capable of providing accurate comparisons when the number of clusters produced by the algorithm is the same for both methods. This is why the use of an algorithm capable of explicitly setting the number of resulting clusters was necessary.

For this experiment, the clustering algorithm was configured to produce a range of one to twenty clusters, and the figure of merit was calculated for each combination of similarity measure and graph type to gauge their effectiveness. The results of performing this are given in the following section.

Results

​​

*Table 2: Figure of merit for all combinations of similarity measure and graph type, ranging from 1 to 20 clusters.*

*Overview*

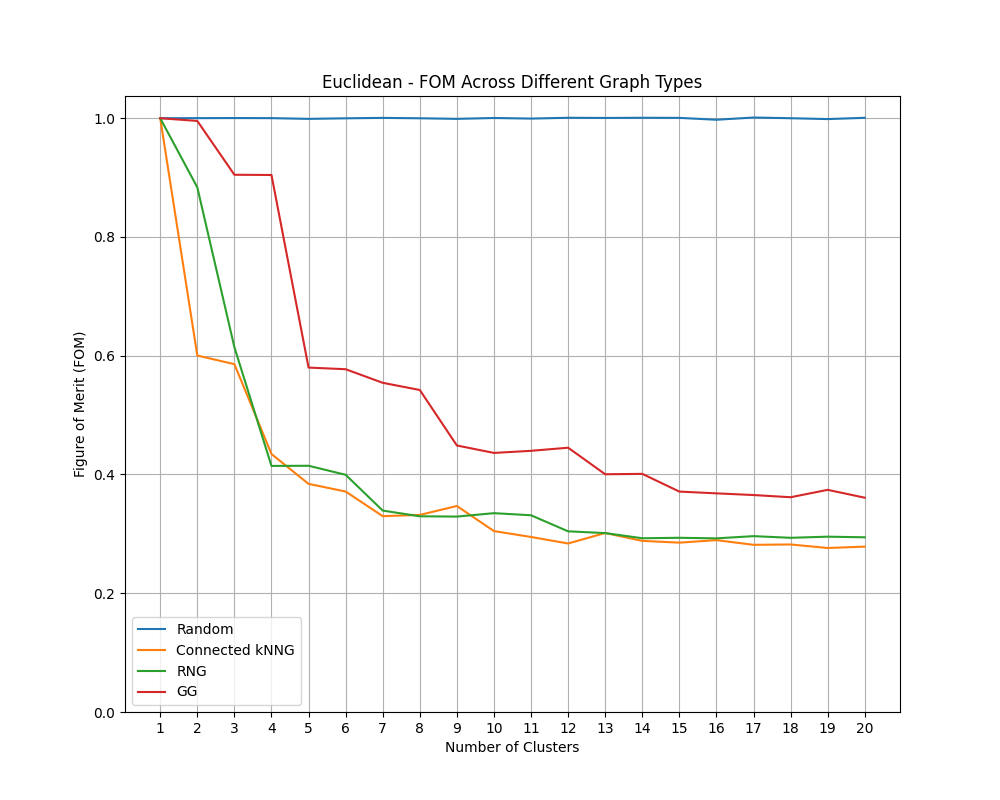
Table 2 details the figure of merit for each graph type across every similarity measure, evaluated over one to twenty clusters. Additionally, the minimum figure of merit for each similarity measure is highlighted at the bottom of the respective graph type column, displaying a comparison of the relative performance across both graph types and similarity measures.

*Graph Performance*

The analysis of graph performance reveals a clear ranking based on the lowest Figure of Merit: Connected-k-Nearest Neighbors Graph (c-kNNG), Relative Neighbor Graph (RNG), and Gabriel Graph (GG) in descending order of accuracy. This suggests that c-kNNG generally produces the most precise clusters that correlate most closely with real groups of co-regulated genes. While the rankings are mostly consistent, there are exceptions where the performance across graph types is highly similar, particularly noted in cases where the graphs yield nearly identical results. A notable deviation occurs with the Gabriel Graph using uncentered correlation, where the clusters' quality is comparable to random clustering. c-kNNG achieves the lowest figure of merit in seven out of the eight similarity measures, with Spearman being the outlier, but Spearman yielded clusters of markedly lower quality. Additionally, in terms of computational efficiency and complexity, c-kNNG is the fastest and least complex, followed by RNG, while GG lags significantly behind due to its high edge density and slower processing.

*Similarity Measure Performance*

In terms of similarity measures, the three distance-based measures (Euclidean, Manhattan, and Mahalanobis) exhibited close coherence metrics, each showing very low figures of merit across all graph types. They are closely followed by squared correlation. The uncentered correlation demonstrated significantly poorer performance relative to these leading measures. Then, the other three correlation-based measures (Pearson, Spearman, and Absolute) recorded the highest figures of merit, approaching 1, indicative of performance nearing that of random clustering.



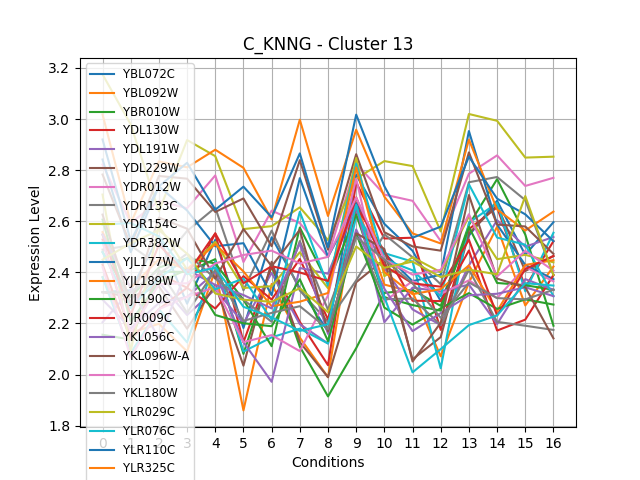
*Figure 2: Figure of Merit of all graph types for Euclidean Distance.*

*True Cluster Number*

Additional insights from Yeung et al.'s paper suggest that the sharp decrease in Figure of Merit generally stabilizes around the number of actual clusters present, which was observed through external validation. As illustrated in Figure 2, the decline for Euclidean distance plateaus at about 8 to 10 clusters, suggesting that the true number of groups of coregulated genes in this dataset likely falls around this range. This pattern of sharp decline is generally consistent across all the tested similarity measures in this study.

*Expression Pattern*

Lastly, a visualization of the expression pattern of all genes within a cluster—specifically, the cluster associated with the lowest figure of merit—is provided in Figure 3. This plot serves to validate the efficacy of the clustering methodology used in this study. Figure 3 effectively demonstrates that the most accurately derived clusters correspond to actual groups of co-regulated genes. This is evidenced by the strong similarity in expression levels across all genes within this cluster, reinforcing the accuracy of the clustering approach used.



*Figure 3 displays the expression pattern of cluster 13 for the 19-cluster clustering of the connected k-Nearest Neighbor Graph for Mahalanobis Distance.*

Discussion

*Graph Algorithms Analysis*

The performance discrepancies observed among the graph algorithms can be attributed largely to their structural characteristics. The Connected-k-Nearest Neighbors Graph (c-kNNG) emerged as the top performer, aligning with expectations given its design. c-kNNG establishes high connectivity between highly similar genes while maintaining sparser connections among dissimilar ones. This structural design is particularly conducive to identifying clusters of co-regulated genes. In contrast, the Relative Neighbor Graph (RNG), despite its less defined structure compared to c-kNNG, surprisingly preserves crucial topological features that are essential for distinguishing groups of highly similar genes. This describes why these two types of neighborhood graphs had roughly similar figures of merit.

The Gabriel Graph (GG), despite including all edges present in an RNG plus additional ones, performed less effectively. This outcome challenges the assumption that more information (i.e., additional edges) necessarily improves clustering results. Instead, the results suggest that the extra edges in GG might introduce noise and obscure the essential structure needed for effective clustering, thereby degrading performance rather than enhancing it.

*Similarity Measures Analysis*

Turning to the similarity measures, the nature of the problem—identifying genes that share similar expression levels across various conditions—suggests a preference for measures that evaluate direct differences in expression. The superior performance of distance-based measures (Euclidean, Manhattan, and Mahalanobis) confirms this, as these metrics straightforwardly quantify the differences in expression levels. These measures directly align with the goal of clustering co-regulated genes, which exhibit lesser variability in their expression profiles across the dataset.

Conversely, correlation-based measures like Pearson Correlation assess similarity based on the trend of gene expression rather than absolute values. While two genes might follow a similar trend (e.g., increasing or decreasing expression across conditions), their actual expression levels can be vastly different, leading to misleading conclusions about their co-regulation. This characteristic makes such measures less effective for predicting the values of omitted conditions when calculating the Figure of Merit, as they do not directly address the primary clustering objective of identifying co-regulated genes based on consistent expression levels.

In conclusion, the specific structural and operational characteristics of both graph algorithms and similarity measures critically influence their performance in clustering tasks. Understanding these nuances can provide deeper insights into their applicability and effectiveness in various genomic data analysis contexts.

Limitations

This study presented interesting insights into the relative performance of different methods for identifying coregulated genes. However, several limitations must be acknowledged:

1. The computational demands of graph algorithms, especially Gabriel Graphs, limited the extent to which the validation process could be conducted. Only one condition (Condition 5) was removed for testing due to these constraints, rather than performing the more robust aggregate figure of merit calculation across all conditions. This could impact the generalizability of the validation results.
2. The efficacy of the similarity measures relies on the assumption that the chosen metrics accurately reflect biological significance, which may not always hold true across different datasets or conditions. Particularly, the performance of correlation-based measures highlighted potential discrepancies where trends in gene expression did not necessarily correlate with functional co-regulation.
3. Only three types of graph algorithms were evaluated. While these were selected based on their theoretical applicability to the problem at hand, other graph types or advanced modifications of the evaluated graphs might provide different or improved insights.
4. The community detection algorithm was fixed to a spectral clustering method which, while effective, might not capture all potential cluster structures as dynamically as other clustering methods like hierarchical clustering or dynamic community detection. In addition this study was originally intended to include comparison of NNGs but this clustering algorithm doesn’t have the capability to work with disconnected graphs.
5. The validation of this study relied heavily on the figure of merit as a sole measure of cluster quality. While this provided a quantitative measure of predictive power, it might not fully capture other aspects of cluster validity such as biological relevance. These relationships could be found with more extensive validation, including external validation.
6. Results derived from the yeast gene expression matrix may not be universally applicable to other organisms or conditions. Gene expression can vary significantly across different organisms, stages of development, or in response to environmental factors, potentially requiring adjustments to the methodology.

By addressing these limitations in future studies, the robustness and applicability of the findings could be enhanced, leading to more useful tools and methods for genomic data analysis.

Conclusions

This study successfully demonstrated the utility of various graph-theoretic methods and similarity measures in identifying clusters of co-regulated genes within a yeast gene expression matrix. The investigation yielded several key findings:

1. The Connected-k-Nearest Neighbors Graph (c-kNNG) was found to be the most effective graph type for clustering gene expression data, due to its ability to emphasize strong connections between highly similar genes while minimizing less relevant relationships. This structure proved to be more conducive to identifying meaningful clusters compared to the Relative Neighbor Graph (RNG) and Gabriel Graph (GG), with the latter showing decreased performance due to the potential over-cluttering from excessive edges. This goes to show that what’s most important about the graph structure is if the edges are meaningful, as opposed to there simply being more edges.
2. Among the similarity measures tested, distance-based measures (Euclidean, Manhattan, and Mahalanobis) consistently outperformed correlation-based measures in terms of their ability to identify meaningful clusters. This suggests that direct measurement of differences in gene expression levels is a more reliable method for detecting co-regulation compared to measures that assess correlation, which may interpret aligned trends as similarity even when actual expression levels differ significantly.
3. The findings of this study show the importance of choosing appropriate methods for the analysis of gene expression data, as the selection of graph algorithms and similarity measures can significantly influence the results. This study provides a comparative framework that can assist researchers in selecting the most suitable methods for their specific needs.

Further Research

This study opens several avenues for further research to expand upon the findings and address the limitations encountered. Future studies could consider the following areas:

1. Investigating other types of graph algorithms could provide insights into alternative clustering strategies that may be more effective or suitable for different types of datasets.
2. Extending the analysis to include gene expression data from different organisms or from various conditions within the same organism could help validate the generalizability of the methods used. This would also allow researchers to understand how gene regulation and expression patterns differ across biological contexts.
3. Coupling computational methods with experimental validation, such as gene knock-out studies or functional assays, could enhance the biological relevance of the identified clusters. This would help confirm that the clusters predicted computationally are indeed functionally coregulated in vivo.
4. Implementing more comprehensive validation metrics beyond the figure of merit, such as stability analysis or consistency checks across multiple datasets, could provide a deeper understanding of the robustness and reliability of the clustering results.

By addressing these questions, future research can build on the foundational work presented in this study, enhancing the analytical tools available for genomic research and expanding their applicability to a broader range of biological and medical challenges.

References

1. D’haeseleer, Patrik. “How does gene expression clustering work?” Nature Biotechnology, vol. 23, no. 12, Dec. 2005, pp. 1499–1501, https://doi.org/10.1038/nbt1205-1499.
2. Eisen, Michael B., et al. “Cluster analysis and display of genome-wide expression patterns.” Proceedings of the National Academy of Sciences, vol. 95, no. 25, 8 Dec. 1998, pp. 14863–14868, https://doi.org/10.1073/pnas.95.25.14863.
3. Jaromczyk, J.W., and G.T. Toussaint. “Relative neighborhood graphs and their relatives.” Proceedings of the IEEE, vol. 80, no. 9, 1992, pp. 1502–1517, https://doi.org/10.1109/5.163414.
4. Von Luxburg, Ulrike. “A tutorial on spectral clustering.” Statistics and Computing, vol. 17, no. 4, 22 Aug. 2007, pp. 395–416, https://doi.org/10.1007/s11222-007-9033-z.
5. Yeung, K. Y., et al. “Validating clustering for gene expression data.” Bioinformatics, vol. 17, no. 4, 1 Apr. 2001, pp. 309–318, https://doi.org/10.1093/bioinformatics/17.4.309.