Three-Dimensional Clustering Method on Gene Expression Dataset Using the Gene Cube Approach

A. N. Ayudhiya, S. M. Soemartojo[1, a)] and T. Siswantining[2, b)]

Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Indonesia, Depok, Indonesia

[a)] Corresponding author: almaira.nabila@sci.ui.ac.id; saskya@sci.ui.ac.id
[b)] titin@sci.ui.ac.id

Abstract. Nowadays, the interest in gene expression data analysis has grown rapidly. This is because the analysis of information generated from gene expression data analysis has enabled the founding of biological phenomenon. As how the data itself is reasoned, the clustering approach has become a technique that has been widely applied in understanding gene expression data. As technology advances, the triclustering technique has been used for many biological analysis, but none has yet taken the role of chromosomes into account. In this study, the chromosome identity of each gene was accounted before doing clustering, which was done by forming a three-dimensional structure, i.e. chromosome-gene-observation. The structure is known as a gene cube. This gene cube approach is implemented in the expression of bladder cancer gene data. To find out whether the cube structure of this gene can provide the desired information through finding the pattern of gene expression, K-means clustering which initial step is optimized using the K-means++ algorithm and δ-Trimax triclustering method are applied. The K-means clustering method produces optimal clusters with a small value of Davies Bouldin index on the gene, observation, and chromosome axis. Meanwhile, the δ-Trimax triclustering method produces optimal triclusters using the best threshold obtained based on the smallest value of tricluster diffusion. The gene cube structure has been shown to provide the desired information from a new perspective, namely the dimensions of chromosomes.

Keywords: Davies Bouldin, K-Means, δ-Trimax, Tricluster Diffusion

INTRODUCTION

The growth of very large data these days makes data analysis methods a very important needs to do. The use of data mining can help convert large data sizes into useful information. One of the methods used in data mining is clustering. Clustering is a process of partitioning a collection of data observations into several subsets known as clusters, so that observations in clusters have similar characteristics to each other but differ from the characteristics of observations in other clusters [1]. Clustering can only be applied to data with one dimension separately, namely to classify only observations or attributes. However, when working with data, it is possible to group observations on certain attributes only, not on all attributes. For this purpose, biclustering is performed. Biclustering can cluster two-dimensional data simultaneously.

Over time, the need for data analysis has grown from two-dimensional to three-dimensional data, where the third dimension is referred to as context. Therefore, an analysis that can cluster three-dimensional data simultaneously is needed. Triclustering can cluster the observation, attributes, and context dimensions simultaneously. The resulting tricluster is a subset of observations, attributes, and contexts from three-dimensional data [2].

Clustering has been applied in many fields of study, one of which is in the field of bioinformatics as a tool to understand the information contained in the microarray gene expression dataset. The use of microarray technology has made it possible to measure the level of expression of thousands of genes under several experimental conditions. The resulting data are arranged in a numerical matrix known as an expression matrix [3]. Each element of this data matrix represents the level of numerical expression of a gene under certain experimental conditions. With the development of microarray technology, the interest in extracting useful knowledge from gene expression data has increased rapidly, because analysis of this information can make it possible to find certain biological phenomenon [4].

To perform clustering of microarray data, usually a two-dimensional matrix with the form of Gene × Observation will first be created and then the appropriate clustering algorithm is applied. An example of an algorithm that is often used is the K-means algorithm. Biclustering has also been widely used in analyzing these two-dimensional matrices. In addition, there has been a lot of literature regarding microarray time-series data analysis. In microarray time-series data analysis, a series of microarray experiments are carried out at different time points and then triclustering is applied to a three-dimensional matrix in the form of Time × Gene × Observation [5].

Of many applications that have been made to biological analysis, none of these methods have yet taken into account the role of chromosomes. According to Adriana Miele (2008) [6], expression of gene data is controlled by regulatory elements which can be located alongside a chromosome, in some cases, even located on other chromosomes. Said regulatory elements are proteins produced by a gene regulator, namely genes whose expression products play a role in regulating the expression of other genes.

Chromosome mapping has also proven to be a promising method in finding expression patterns between genes [7]. The main idea of chromosome mapping is to map genes to their chromosome region and if correlation exists between genes then this correlation will be seen from the location of the gene in the chromosome region. Genes that have a similar expression pattern tend to be in close positions along the chromosome. In conducting chromosome mapping analysis, several approaches have been made, but only by paying attention to two-dimensional aspects, namely the correlation of genes and individual observations. Therefore, there is one more dimension to consider, namely the chromosomes themselves.

In this study, an approach is used in the form of a three-dimensional structure with the shape of Chromosome × Gene × Observation, where later this three-dimensional structure is referred to as a gene cube. Then, to see whether the structure of the gene cube can provide the desired information by finding patterns in gene expression, a K-means algorithm which initial steps are optimized using the K-means++ algorithm and the δ-Trimax triclustering algorithm are applied. The expected results from this study are to find genes, observations, and chromosomes that have similar expression patterns, where later this information can be used as a consideration for medical experts to determine the therapeutic target of a disease.

MATERIALS AND METHOD

Dataset

This research used gene expression dataset from bladder cancer with different stages of tumor that was obtained through the Gene Expression Omnibus (GEO) facility from the National Center for Biotechnology Information (NCBI) website. The dataset used is GSE7476 that consisted of 9 bladder cancer observations and 3 control observations, where each observation consisted of 45,746 genes.

K-Means Clustering Method

K-Means Clustering Algorithm

The K-means clustering algorithm is a partitioning clustering analysis method. The first step is to select k observations as the initial centroids. Then, calculating the distance between each observation to the centroid and grouping them into clusters with the closest centroid. Here is an equation for calculating the Euclidean distance:

$$d(x_i, c_j) = \sqrt{\sum_{j=1}^{k} (x_i - c_j)^2}, i = 1, \dots, N; j = 1, \dots, k \qquad (1)$$

where $d(x_i, c_j)$ is the distance between the ith observation and the jth centroid. After that, determine the new centroid by calculating the average of observations in each cluster using the following equation:

$$c_j = \frac{1}{N_i}\sum_{j=1}^{N_i} x_{ij}\, , i = 1, \dots, k; j = 1, \dots, N_i \tag{2}$$

where N_i is the number of observations in cluster i [8].

K-Means++ Clustering Algorithm

The accuracy of the K-means algorithm is very dependent on the value of the initial centroid selected. Therefore, to help improve the accuracy of the K-means algorithm, the K-means++ algorithm is used. Regardless of the selection of the initial centroid, the following algorithms are the same as the usual K-means algorithm. That is, the K-means++ algorithm is a K-means algorithm that is combined with a better selection of initial centroids [9]. The following are the steps for the K-means++ algorithm to select centroids, assuming the number of clusters are k:

1. Choose one observation from the data randomly. The selected observation is the initial centroid and is denoted as c_1.
2. Calculate the distance between each observation against c_1. The distance between the mth observation (x_m) and the jth centroid (c_j) is denoted as $d(x_m, c_j)$.
3. Choose the next centroid, c_2, randomly with probability

$$\frac{d^2(x_m, c_1)}{\sum_{j=1}^{n} d^2(x_j, c_1)}, j = 1, \dots, n \tag{3}$$

   where n is the number of observations on the data.
4. To select the jth centroid (c_j):
   a) Compute the distance between each observation and each centroid and assigns each observation to the nearest centroid.
   b) For $m = 1, \dots, n$ and $p = 1, \dots, j-1$, choose the jth centroid randomly with probability

$$\frac{d^2(x_m, c_p)}{\sum_{\{h; x_h \in C_p\}} d^2(x_h, c_p)} \tag{4}$$

   where C_p is the set of all observations closest to c_p.
5. Repeat step 4 until the centroid has been selected.

Davies Bouldin Index

Davies Bouldin index is used to measure the quality of clustering [10]. The value of the Davies Bouldin index is based on the value of cohesion and separation. In a clustering, cohesion is defined as the sum of the data proximity to the centroid of the cluster. Meanwhile, the separation is based on the distance between the centroids of the cluster. To find out the cohesion in the ith cluster, the sum of square within cluster (SSW) equation is used as follows:

$$SSW_i = \frac{1}{m_i}\sum_{j=1}^{m_i} d(x_j, c_i) \tag{5}$$

where m_i is the amount of data in the ith cluster, x_j is the jth observation, c_i is the centroid of the ith cluster, and d is the distance of each data to the centroid calculated using the Euclidean distance. Meanwhile, to determine the separation between clusters, the sum of square between clusters (SSB) equation is used as follows:

$$SSB_{i,j} = d(c_i, c_j) \tag{6}$$

After the cohesion and separation values are obtained, a ratio measurement $(R_{i,j})$ is carried out to determine the comparison value between the ith cluster and the jth cluster. A good cluster is a cluster that has the smallest possible cohesion value and the largest possible separation value. The ratio value is calculated using the following equation:

$$R_{i,j} = \frac{SSW_i + SSW_j}{SSB_{i,j}} \tag{7}$$

The ratio value obtained is used to find the Davies Bouldin index value using the following equation:

$$DBI = \frac{1}{k}\sum_{i=1}^{k} \max_{i \neq j}(R_{i,j}) \tag{8}$$

where k is the number of clusters used. The smaller the Davies Bouldin index value obtained (≥ 0), the better the quality of the clusters obtained through K-means clustering.

δ-Trimax Triclustering Algorithm

The δ-Trimax method is composed of multiple node deletion, single node deletion, and node addition algorithms [11]. These algorithms do the elimination and addition of nodes iteratively to produce a tricluster that has mean squared residual (R) score less than δ, where δ is the threshold determined by the researcher. If previously Bhar (2013) applied the δ-Trimax triclustering algorithm on the dimensions of time, genes, and observations, in this study the time dimension was replaced by the chromosome dimension, thus ending up with the following formula for the Mean Squared Residual (MSR):

$$\begin{aligned} R &= \frac{1}{|\boldsymbol{G}||\boldsymbol{O}||\boldsymbol{C}|}\sum_{g\in\boldsymbol{G},o\in\boldsymbol{O},c\in\boldsymbol{C}} \left(m_{goc} - m_{goC} - m_{GoC} - m_{GOc} + 2m_{GOC}\right)^2 \\ &= \frac{1}{|\boldsymbol{G}||\boldsymbol{O}||\boldsymbol{C}|}\sum_{g\in\boldsymbol{G},o\in\boldsymbol{O},c\in\boldsymbol{C}} r_{goc}^2 \end{aligned} \tag{9}$$

where $|\boldsymbol{G}|$, $|\boldsymbol{O}|$, and $|\boldsymbol{C}|$ is the sum of the dimensions of the genes, observations, and chromosomes of a tricluster, m_{goc} is the element of the tricluster with the gene g, in the chromosome c, for the observation o, m_{gOC} is the mean of the gth gene, m_{GoC} is the mean of the oth observation, m_{GOc} is the mean of the cth chromosome, and m_{GOC} is the mean of the tricluster. Lower residual scores indicate greater levels of coherence and better tricluster quality.

$$m_{gOC} = \frac{1}{|\boldsymbol{O}||\boldsymbol{C}|}\sum_{o\in\boldsymbol{O},c\in\boldsymbol{C}} m_{goc} \tag{10}$$

$$m_{GoC} = \frac{1}{|\boldsymbol{G}||\boldsymbol{C}|}\sum_{g\in\boldsymbol{G},c\in\boldsymbol{C}} m_{goc} \tag{11}$$

$$m_{GOc} = \frac{1}{|\boldsymbol{G}||\boldsymbol{O}|}\sum_{g\in\boldsymbol{G},o\in\boldsymbol{O}} m_{goc} \tag{12}$$

$$m_{GOC} = \frac{1}{|\boldsymbol{G}||\boldsymbol{O}||\boldsymbol{C}|}\sum_{g\in\boldsymbol{G},o\in\boldsymbol{O},c\in\boldsymbol{C}} m_{goc} \tag{13}$$

Tricluster Diffusion

Tricluster Diffusion (TD) is used to estimate the quality of a tricluster produced based on the mean squared residual and its volume. For a given set of n tricluster, Tricluster Diffusion is defined as follows:

$$\begin{aligned} TD_i &= \frac{MSR_i}{Volume_i} \\ &= \frac{MSR_i}{|\boldsymbol{G}_i||\boldsymbol{O}_i||\boldsymbol{C}_i|} \end{aligned} \tag{14}$$

where MSR_i is the mean squared residual of the ith tricluster, and $|\boldsymbol{G}_i|$, $|\boldsymbol{O}_i|$, and $|\boldsymbol{C}_i|$ each is the number of dimensions of genes, observations, and chromosomes in the ith tricluster. The smaller the value of the Tricluster Diffusion, the better the quality of a tricluster [12].

RESULTS AND DISCUSSION

Formation of the Gene Cube

To form the gene cube structure, the initial dataset is divided into two-dimensional matrices based on their chromosomes. More specifically, first take all the genes located on the first chromosome to be placed in the three-dimensional matrix formed. Next, take all the genes that are located on the second chromosome to be placed in the three-dimensional matrix, and so on. The sequential addition of the two-dimensional matrix eventually created a third dimension in the form of chromosomes. This structure is shown in Figure 1.

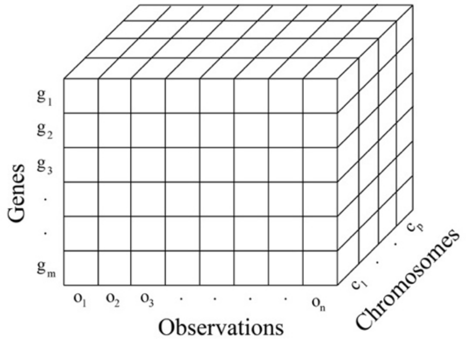

FIGURE 1. Gene cube.

This gene cube structure is a three-dimensional matrix $g_i \times o_j \times c_k$, where genes $(g_i, i = 1, \dots, m)$ are on the y-axis, observations $(o_j, j = 1, \dots, n)$ are on the x-axis, and the chromosomes $(c_k, k = 1, \dots, 24)$ are on the z-axis. Then, to sequence genes contained on a chromosome, each gene from the kth chromosome (c_k) is placed next to the gene that has a similar expression profile to the previous chromosome $c_1, \dots, c_{k-1}$. This similarity is measured using the Euclidean distance between the gene under study and the centroid of the gene on the previous chromosome, $c_1, \dots, c_{k-1}$, along the chromosome axis.

Each chromosome may not have the same number of genes [13], so that it will cause a missing slot in the three-dimensional matrix that is formed. To solve this problem, insert nan values (not a number) in all missing slots. The nan values are data types provided by the NumPy library in the Python program. This data type can later be processed effectively by the functions contained in the Python program, so that further analysis can be carried out properly. An illustration of the insertion of nan values into the missing slot of the gene cube structure is shown in Figure 2.

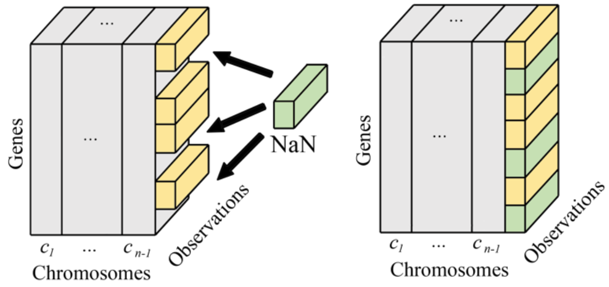

FIGURE 2. Illustration of the insertion of nan values in the missing slot of the gene cube.

The chromosome dimensions consist of autosomes, namely chromosome 1 to chromosome 22, and gonosomes, namely X and Y chromosomes. X and Y chromosomes are referred to as chromosome 23 and 24, respectively. List of the number of genes on each chromosome is presented in Table 1.

TABLE 1. The number of genes on each chromosome.

Chromosome	Number of Genes	Chromosome	Number of Genes
Chromosome 1	4510 genes	Chromosome 13	987 genes
Chromosome 2	3184 genes	Chromosome 14	1528 genes
Chromosome 3	2648 genes	Chromosome 15	1492 genes
Chromosome 4	1838 genes	Chromosome 16	1852 genes
Chromosome 5	2141 genes	Chromosome 17	2527 genes
Chromosome 6	2551 genes	Chromosome 18	806 genes
Chromosome 7	2272 genes	Chromosome 19	2668 genes
Chromosome 8	1661 genes	Chromosome 20	1281 genes
Chromosome 9	1812 genes	Chromosome 21	600 genes
Chromosome 10	1798 genes	Chromosome 22	1093 genes
Chromosome 11	2487 genes	Chromosome 23 (X)	1512 genes
Chromosome 12	2410 genes	Chromosome 24 (Y)	88 genes

It appears that not all chromosomes have the same number of genes. The chromosome that has the largest number of genes is chromosome 1 with 4510 genes. Because the largest number of genes is 4510, the dimension of the genes is the cube structure is obtained with dimensions of genes × observations × chromosomes, namely 4510 × 12 × 24.

Clustering Analysis Using the K-Means Algorithm on the Gene Cube

To perform K-means clustering analysis, the three-dimensional structure that has been formed will be transformed into a two-dimensional structure by taking the corresponding centroids along each axis. In general, there are three cross sections formed from the gene cube. The first is a cross section across the gene axis which concerns the dimension of one gene (g_i), all observations, and all chromosomes. Furthermore, the second is a cross section across the observation axis which concerns the dimension of one observation (o_j), all genes, and all chromosomes. Finally, the third one is a cross section across the chromosome axis which concerns the dimension of one chromosome (c_k), all genes, and all observations. The cross sections are illustrated in Figure 3.

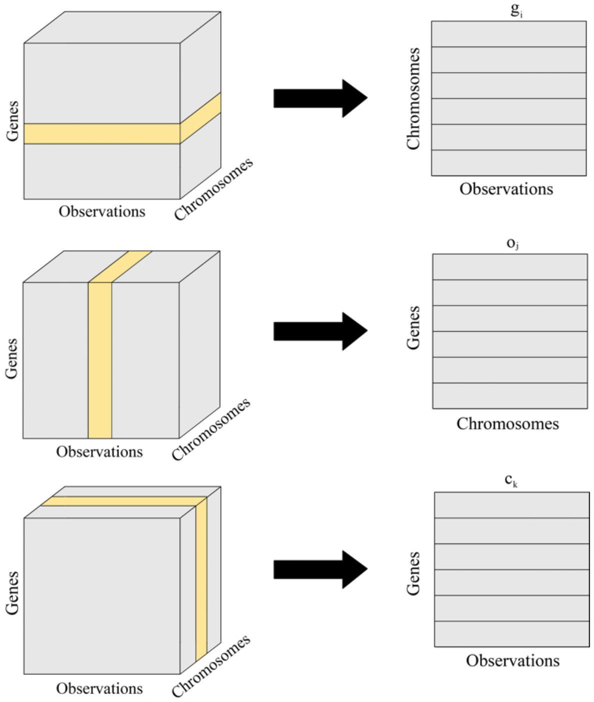

FIGURE 3. Illustration of cross sections of the gene cube.

K-Means Clustering Across the Gene Axis

The number of clusters (k) used to perform K-means clustering across the gene axis is 4. Figure 4 shows the gene division plots in cluster 1, 2, and 4.

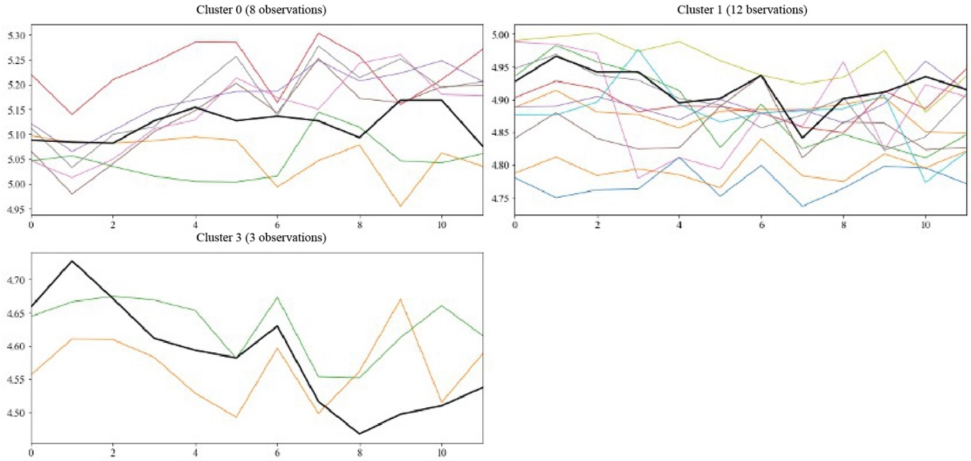

FIGURE 4. Gene division plots for cluster 1, 2, and 4.

The Davies Bouldin index of the resulting cluster is 0.33062. Therefore, it can be concluded that the gene cube can group observations that have similar expressions well.

K-Means Clustering Across the Observation Axis

The number of clusters (k) used to perform K-means clustering across the observation axis is 2500. Figure 5 shows the gene division plots in cluster 1228, 1231, 1234, and 1243.

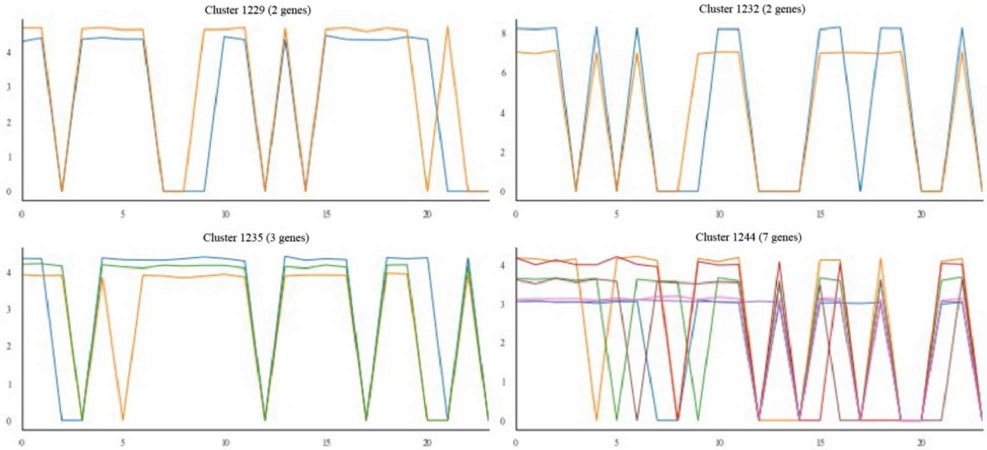

FIGURE 5. Gene division plots for cluster 1229, 1232, 1235, and 1244.

The Davies Bouldin index of the resulting cluster is 0.63860. Because the value of the index is still in the range of 0 and 1, the resulting cluster can be said to have good quality. Therefore, it can be concluded that the gene cube can group genes that have similar expressions well.

K-Means Clustering Across the Chromosome Axis

The number of clusters (k) used to perform K-means clustering across the chromosome axis is 1800. Figure 6 shows the gene division plots in cluster 1708, 1712, 1714, and 1715.

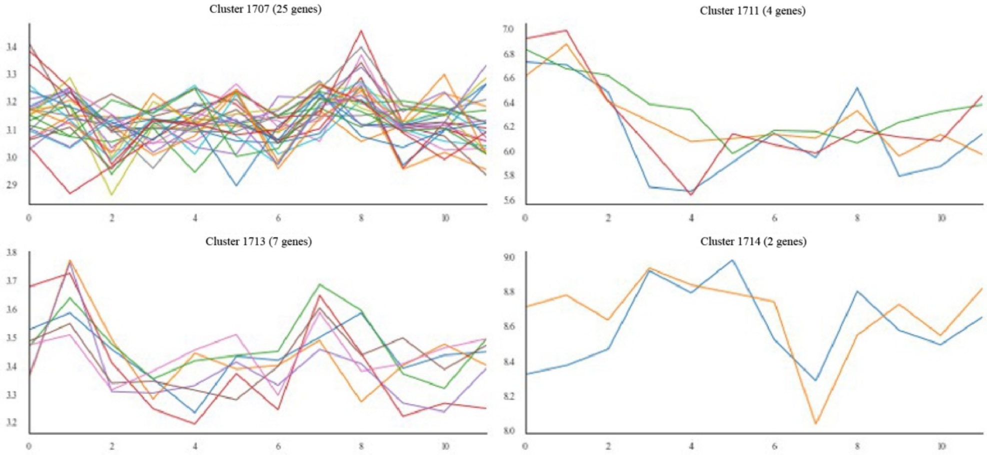

FIGURE 6. Gene division plots for cluster 1707, 1711, 1713, and 1714.

The Davies Bouldin index of the resulting cluster is 0.66134. Because the value of the index is still in the range of 0 and 1, the resulting cluster can be said to have good quality. Therefore, it can be concluded that the gene cube can group genes that have similar expressions well.

Triclustering Analysis Using the δ-Trimax Algorithm on the Gene Cube

In this study, five thresholds δ values, that is 0.0466; 0.0566; 0.0732; 0.0765; and 0.0817, and three thresholds λ values, that is 1.05; 1.15; and 1.25 were used so there were 15 simulations carried out using all five thresholds δ and all three λ thresholds. Based on the comparison of the simulation results, the best simulation is when the threshold $\delta = 0.0765$ and threshold $\lambda = 1.15$ where the average of its tricluster diffusion value is 1.52081×10^{-7}. In this simulation, 13 triclusters were produced which are presented in Table 2. The triclusters presented have been sorted starting from the best quality tricluster which has the smallest tricluster diffusion value.

TABLE 2. List of triclusters generated using $\delta = 0.0765$ and $\lambda = 1.15$.

No.	Tricluster Diffusion	Dimensions (Gene × Observation × Chromosome)
1	1.18767×10^{-7}	2106 × 12 × 24
2	1.20914×10^{-7}	1979 × 12 × 23
3	1.23117×10^{-7}	1889 × 12 × 19
4	1.26634×10^{-7}	1358 × 12 × 18
5	1.30937×10^{-7}	2076 × 12 × 20
6	1.34220×10^{-7}	1452 × 12 × 18
7	1.38313×10^{-7}	1910 × 12 × 16
8	1.38879×10^{-7}	1416 × 12 × 12
9	1.29655×10^{-7}	1338 × 12 × 12
10	1.79960×10^{-7}	1620 × 12 × 24
11	1.90850×10^{-7}	1442 × 12 × 10
12	$2{,}11273 \times 10^{-7}$	1699 × 12 × 14
13	$2{,}23538 \times 10^{-7}$	1577 × 12 × 21

By looking at the Table 2, it is known that each tricluster generated in this simulation is grouped across all observations. Based on the previous studies, the differentiation process of bladder cancer can be used to detect genes that have the potential to cause cancer. So, the genes that are grouped from each tricluster can be used to stimulate the development of therapy in that gene group.

Discussion

The implementation of the gene cube approach in this study is using the gene expression data of bladder cancer patients. From the results of the implementation, it is known that the K-means algorithm produces groups of gene expressions that have similar pattern on each axis of each dimension. K-means clustering was successful in finding optimal clusters with small Davies Bouldin index values on the gene, observation, and chromosome axis. Further on, based on 15 simulations conducted by δ-Trimax triclustering using different δ and λ, the best simulation is obtained, where the tricluster generated from this simulation has the smallest average tricluster diffusion value. In this simulation, optimal tricluster is produced, some of which are thought to be a group of gene expression that has the characteristics of bladder cancer. Therefore, the gene group in this tricluster can be used by medical experts as a target to stimulate the development of therapy on these genes. The gene cube structure manifests a greater opportunity for gene expression analysis as it facilitates the possibility of examining different combinations of experimental variables.

CONCLUSION

In this research, a data structure known as gene cube structure has been used. Gene cube is a three-dimensional matrix, where the dimensions consist of genes, observations, and chromosomes. The advantage of this gene cube approach is that the data structure is formed by considering the chromosomes of each gene. This approach can be useful in understanding the mechanisms of disease and tumors in general. By testing the gene cube using K-means algorithm which initial steps is optimized using K-means++ algorithm and δ-Trimax triclustering algorithm, it has been proven that gene cube structure could provide information about groups of gene expression that have a similar pattern.

REFERENCES

1. J. Han, M. Kamber, and J. Pei, *Data Mining: Concept and techniques* (Morgan Kaufmann Publishers Inc, California, 2011). (978-0-12-381479-1)
2. R. Henriques and S. C. Madeira, ACM Computing Surveys, **95**, 3941-3958 (2004). (10.1145/3195833)
3. P. Baldi and G. W. Hatfield, *DNA Microarrays and Gene Expression: From Experiments to Data Analysis and Modeling* (Cambridge University Press, Cambridge, 2011). (978-0521176354)
4. R. Harpaz and R. Haralick, International Conference on Pattern Recognition, **2**, 670-674 (2006).
5. P. Mahanta, H. Ahmed, D. K. Bhattacharyya, and J. Kalita, National Conference on Emerging Trends and Applications in Computer Science, 1-6 (2011).
6. A. Miele and J. Dekker, Molecular buiSystems, **4**, 1046-1057 (2008).
7. B. A. Cohen, R. D. Mitra, J. D. Hughes, and G. M. Church, Nature Genetics, **26**, 183-186 (2000).
8. J. Wang and X. Su, IEEE 3rd International Conference on Communication Software and Networks, 44-46 (2011).
9. D. Arthur and S. Vassilvitskii, Proceedings of the Annual ACM-SIAM Symposium on Discrete Algorithms, **8**, 1027-1035 (2007).
10. D. Davies and D. Bouldin, IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-1, **2**, 224-227 (1979).
11. A. Bhar, M. Haubrock, A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and E. Wingender, Algorithms for Molecular Biology, **8** (2013).
12. A. Bhar, M. Haubrock, A. Mukhopadhyay, and E. Wingender, BMC Bioinformatics, **16** (2015).
13. I. Pathak and B. Bordoni, *Genetics, Chromosomes* (StatPearls Publishing, 2020).