Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 8 No. 3 (2024) 361 - 367 e-ISSN: 2580-0760

Comparison of Matrix Decomposition in Null Space-Based LDA Method

Carissa Devina Usman[1], Farikhin[2], Titi Udjiani[3]
[1,2,3]Department of Mathematics, Faculty of Science and Mathematics, Diponegoro University, Semarang, Indonesia
[1]carissa.devina2@gmail.com, [2]farikhin.math.undip@gmail.com, [3]udjianititi@lecturer.undip.ac.id

Abstract

Problems with small sample sizes and high dimensionality are common in pattern recognition. Almost all machine learning algorithms degrade in high-dimensional data, so that singularities in the scatter matrices, the main problem of the Linear Discriminant Analysis (LDA) technique, might result. A null space-based LDA (NLDA) has been conceived to address the singularity issue. NLDA aims to maximize the distance between classes in the null space of the within-class scatter matrix. In the earliest research, the NLDA method was performed by computing eigenvalue decomposition and singular value decomposition (SVD). This research led to several new implementations of the NLDA method using other matrix decompositions. The new implementations include NLDA using Cholesky decomposition and NLDA using QR decomposition. This paper compares the performance of three NLDA methods using different matrix decompositions, namely SVD, Cholesky decomposition, and QR decomposition. Two sets of data were used in the experiments that used three different NLDA algorithms. To determine the performance of the NLDA methods, the classification accuracy of the three methods was measured using the Confusion Matrix. The results show that the NLDA method using SVD has the best performance when compared to the other two methods, achieving 77.8% accuracy for the Colon dataset and 98.8% accuracy for the TKI-resistance dataset.

Keywords: *linear discriminant analysis; small sample size; null space; singular value decomposition (SVD); Cholesky decomposition; QR decomposition*

How to Cite: C. D. Usman, Farikhin, and Titi Udjiani, "Comparison of Matrix Decomposition in Null Space-Based LDA Method", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 8, no. 3, pp. 361 - 367, Jun. 2024.
DOI: https://doi.org/10.29207/resti.v8i3.5637

1. Introduction

Research on image recognition has been going strong for decades. A few examples of related applications include smart cards, surveillance systems, biometrics, information security, and access control [1]. Real-world data sets often have too many dimensions. Almost all machine learning algorithms degrade in high-dimensional data because high-dimensional data is likely to contain noise, redundant (correlated between variables), and features with small variances that can cause a phenomenon called the curse of dimensionality [2]. Besides causing high computation time [3], high dimensionality with a much smaller number of samples than dimensionality can also cause overfitting [4], thus reducing the performance of machine learning algorithms. Not all features in high-dimensional data are relevant to the problem at hand, so it is necessary to reduce them. Dimensionality reduction is useful for improving the performance of machine learning algorithms, memory efficiency, reducing computational costs, and visualization [5]. The objective is to reduce the number of dimensions of a high-dimensional dataset without sacrificing any of the useful information contained therein [6].

Many dimension reduction techniques have been suggested in recent decades. Linear Discriminant Analysis (LDA) and Principal Component Analysis (PCA) are statistical-based methods now widely utilized for dimensionality reduction. LDA is a class-based (supervised) dimensionality reduction method, while PCA is a dimensionality reduction method but not class-based (unsupervised), so PCA cannot guarantee maximum separation between classes [6]. For classification, LDA is generally more suitable than PCA [7].

If the dimension of the data is much greater than the number of training vectors, then the within-class matrix and the total scatter matrix are singular. This weakness is considered a main problem of LDA, which is prevalently known as the singularity problem caused by Small Sample Size (SSS) [8]. Classical LDA cannot handle the SSS problem, so many variations of LDA are proposed to overcome the SSS problem. Among them are Direct LDA (DLDA) [9], Regularized LDA (RLDA) [10], Subspace LDA (SLDA) [11], Null space LDA (NLDA) [12], ILDA [13], GO-LDA [14], and many more.

Here, we zero in on NLDA as it pertains to the SSS issue. There have been many variations of NLDA with new and faster implementations. In [15] and [12], the NLDA method is performed by calculating eigenvalue decomposition and singular value decomposition (SVD) to obtain the optimal transformation matrix. However, it turns out that the NLDA method has a high computational cost, so in [16] and [17], a new method is proposed without calculating eigenvalue decomposition and SVD but using QR decomposition. For high dimensionality and small sample size problems, [18] proposed a subspace method and introduced the notion of a tenuous null subspace and its associated projection operator. Another development of NLDA is also discussed in [19], namely by applying Cholesky decomposition to the scatter matrix in a class. A reference collection of variations of NLDA methods is discussed in [8].

The three types of NLDA methods that will be discussed are methods that apply matrix decomposition, namely the SVD method, Cholesky decomposition, and QR decomposition on within-class scatter matrices, respectively. Previous research has not compared the performance of NLDA methods that use the three matrix decompositions. The research in [16] discusses two variations of the NLDA method using SVD, and one using QR decomposition. Then the research in [17] also discusses two variations of the NLDA method using SVD and two others using QR decomposition, but experiments are only carried out on the NLDA method using QR decomposition.

2. Research Methods

2.1 Classical Linear Discriminant Analysis (LDA)

Given a data matrix $A = \{a_1, a_2, \dots, a_n\} = [A_1, \dots, A_k] \in \mathbb{R}^{m\times n}$, where $a_i \in \mathbb{R}^m$ is the i-th training sample of m dimensional space for $i = 1,2, \dots, n$, $A_i \in \mathbb{R}^{m\times n_i}$ is the collection of training samples from the i-th class for $i = 1,2, \dots, k$, and $\sum_{i=1}^{k} n_i = n$. Let N_i be the collection of column indices for each i-th class, where a_j, for $j \in N_i$, is a member of that class. In the classic LDA, within-class scatter matrices, between-class scatter matrices, and total scatter matrices, are defined as Formula 1.

$$S_w = \sum_{i=1}^{k} \sum_{j\in N_i} (a_j - c_i)(a_i - c_i)^T = H_w H_w^T \quad (1)$$

$$S_b = \sum_{i=1}^{k} n_i (c_i - c)(c_i - c)^T = H_b H_b^T$$

$$S_t = \sum_{i=1}^{n} (a_i - c)(a_i - c)^T = H_t H_t^T = S_b + S_w$$

Where c_i is the local centroid of the i-th class and is defined as $c_i = \frac{1}{n_i} A_i e_i$, where $e_i = (1,1, \dots, 1)^T \in \mathbb{R}^{n_i}$, c is the global centroid and is defined as $c = \frac{1}{n} Ae$, where $e = (1,1, \dots, 1)^T \in \mathbb{R}^n$. H_b, H_w, and H_t matrices are defined as Formula 2.

$$H_w = [A_1 - c_1 e_1^T, \dots, A_k - c_k e_k^T] \in \mathbb{R}^{m\times n}$$

$$H_b = [\sqrt{n_1}(c_1 - c), \dots, \sqrt{n_k}(c_k - c)] \in \mathbb{R}^{m\times k} \quad (2)$$

$$H_t = A - ce^T \in \mathbb{R}^{m\times n}$$

Finding the best projection matrix G is the goal of the conventional LDA approach, which involves solving the following optimisation problem is shown in Formula 3.

$$G = arg\ max_G trace((G^T S_w G)^{-1}(G^T S_b G)) \quad (3)$$

trace(·) demonstrates the trace operator. By tackling the accompanying expanded eigenvalue issue, we can get the answer for Formula 4.

$$S_b g = \lambda S_w g, \lambda \neq 0 \quad (4)$$

whose columns in G are the eigenvectors that correspond to the $n - 1$ highest eigenvalues.

2.2 Null Space LDA (NLDA) using Singular Value Decomposition (SVD)

The null space of S_w must be calculated to find the optimum matrix G. Because data is highly dimensional, this null space may be large. [20] improved the efficiency of the algorithm in [21] via the initial step of removing the null space of S_t.

Let $H_t = U\Sigma V^T$ be the SVD [22] of H_t. Here H_t is defined by Formula 2, and U and V are orthogonal,

$$\Sigma = \begin{bmatrix} \Sigma_t & 0 \\ 0 & 0 \end{bmatrix},$$

$\Sigma_t = \mathbb{R}^{t\times t}$ is the diagonal entries sorted in the non-ascending order, and $t = rank(S_t)$. Then

$$S_t = H_t {H_t}^T = U\Sigma V^T V \Sigma^T U^T = U \begin{bmatrix} \Sigma_t^2 & 0 \\ 0 & 0 \end{bmatrix} U^T.$$

Let $U = (U_1, U_2)$ be U divided by $U_1 \in \mathbb{R}^{m\times t}$ and $U_2 \in \mathbb{R}^{m\times(m-t)}$. Data may be projected into the subspace spanned by the columns of U_1 to remove the null space from S_t. After removing the null space from S_t, the scatter matrices are $\tilde{S}_b$, $\tilde{S}_w$, and $\tilde{S}_t$. That is

$$\tilde{S}_b = U_1^T S_b U_1,\ \tilde{S}_w = U_1^T S_w U_1, \text{ and } \tilde{S}_t = U_1^T S_t U_1.$$

$G = U_1 N$ gives the optimal transformation of NLDA with the calculated U_1. Here N solves the following optimization problem [9] as shown in Formula 5.

$$N = arg\ max_{N^T \tilde{S}_w N = 0} trace(N^T \tilde{S}_b N) \quad (5)$$

that is, the columns of N are in the null space of $\tilde{S}_w$, while maximizing trace$(N^T \tilde{S}_b N)$.

Let W be a matrix whose columns of W span the null space of $\tilde{S}_w$. Next, $N = WM$ is applied to the next determined matrix M. for each M, the constraint in Formula 5 satisfies $N = WM$, so we can calculate the optimal M by maximizing

$$trace\left(M^T W^T \tilde{S}_b W M\right).$$

By requiring an orthogonality constraint on M [20], the optimal M is given by the eigenvectors of $W^T \tilde{S}_b W$ associated with the nonzero eigenvalues. The calculation of the eigen decomposition of $\tilde{S}_w$ yields the matrix W. The optimal transformation of NLDA is given by

$$G = U_1 W M.$$

The NLDA algorithm using SVD is shown in Table 1[12].

Table 1. NLDA Algorithm using SVD

NLDA algorithm using SVD
Input: matrix A Output: transformation matrix G
1. From the matrix H_t, calculate the reduced SVD of H_t i.e., $H_t = U_1 \Sigma_t V_1^T$;
2. Form the matrix $\tilde{S}_b = U_1^T S_b U_1$ and $\tilde{S}_w = U_1^T S_w U_1$;
3. Compute the matrix W which is the null space of $\tilde{S}_w$ by eigen decomposition;
4. Form the matrix M which is the upper eigenvector of $W^T \tilde{S}_b W$;
5. $G = U_1 W M$.

2.3. Null Space LDA (NLDA) using Cholesky Decomposition

In this section, $x_j^{(i)}$ is used to indicate that a sample x is the jth sample of the ith class. In this case, H_w can be expressed as

$$H_w = [x_1^{(1)} - c_1, \dots, x_{n_1-1}^{(1)} - c_1, x_{n_1}^{(1)} - c_1, \dots,$$
$$x_1^{(c)} - c_k, \dots, x_{n_k-1}^{(c)} - c_k, x_{n_k}^{(c)} - c_k] \in \mathbb{R}^{m \times n}$$

Assuming the last sample of each class is removed from H_w, we get [19]

$$H_w^{rem} = [x_1^{(1)} - c_1, \dots, x_{n_1-1}^{(1)} - c_1, \dots, x_1^{(c)} - c_k,$$
$$\dots, x_{n_k-1}^{(c)} - c_k] \in \mathbb{R}^{m \times (n-k)}$$

The range space of H_w and S_w are also proved to be equal in Formula 1. Then, we can see that the space spanned by H_w^{rem} is the range space of S_w.

Let $\bar{S}_W = (H_w^{rem})^T S_w H_w^{rem}$ and $\bar{S}_B = (H_w^{rem})^T S_b H_w^{rem}$. To calculate the eigenvectors of $\bar{S}_W{}^{-1} \bar{S}_B$, we use the following steps [19]:

Compute the Cholesky decomposition [23] of $\bar{S}_W$, i.e., $\bar{S}_W = (\bar{R}_W)^T \bar{R}_W$ where $\bar{R}_W \in \mathbb{R}^{(n-k)\times(n-k)}$ is an upper triangular matrix.

Let $\bar{H}_B = (H_w^{rem})^T H_b$, then $\bar{S}_B = \bar{H}_B \bar{H}_B{}^T$. Compute the SVD of $(R_W^{-1})^T \bar{H}_B$ to obtain $\bar{U}_B$.

Compute $R_W^{-1} \bar{U}_B$. The column of $R_W^{-1} \bar{U}_B$ are the eigen vectors $\bar{S}_W{}^{-1} \bar{S}_B$ corresponding to nonzero eigenvalues.

To compute $\bar{S}_W$ efficiently, let $\bar{H}_w = (H_w^{rem})^T H_w$, then $\bar{S}_W = \bar{H}_w \bar{H}_w{}^T$. First, compute $(H_w^{rem})^T H_w^{rem}$, then compute $(H_w^{rem})^T H_w$ from $(H_w^{rem})^T H_w^{rem}$ by using the relationship between H_w and H_w^{rem}.

The optimal projection matrix in S_w's null space may be quickly determined by considering the following technique. Let

$$H_{nul} = H_b - H_w^{rem}((H_w^{rem})^T H_w^{rem})^{-1}(H_w^{rem})^T H_b \in \mathbb{R}^{m \times (k-1)}$$

and the SVD of H_{nul} be

$$H_{nul} = U_{nul} \Sigma_{nul} V_{nul}^T.$$

Based on the proof of the theorem in [19], $G_{nul} = U_{nul}$. However, this method does not require the SVD of H_{nul} to obtain G_{nul}.

The NLDA algorithm using Cholesky decomposition is shown in Table 2 [19].

Table 2. NLDA Algorithm using Cholesky Decomposition

NLDA algorithm using cholesky decomposition
Input: matrix A Output: transformation matrix G
1. From the matrices H_w^{rem} and H_b, we can calculate $\bar{H}_B = (H_w^{rem})^T H_b$;
2. Calculate $(H_w^{rem})^T H_w^{rem}$ and its Cholesky decomposition, i.e. $(H_w^{rem})^T H_w^{rem} = (\bar{R}_w^{rem})^T \bar{R}_w^{rem}$;
3. Calculate the matrix H_{nul}, i.e. $H_{nul} = H_b - H_w^{rem}(((\bar{R}_w^{rem})^{-1}((\bar{R}_w^{rem})^{-T}\overline{H_B}))$;
4. Calculate $H_{nul}^T H_{nul}$ and its eigen decomposition, i.e. $H_{nul}^T H_{nul} = EDE^T$;
5. Calculates the optimal projection matrix on the nul space of S_w, i.e. $G_{nul} = H_{nul} E D^{-1/2}$;
6. Obtained the transformation matrix $G = G_{nul}$.

2.4 Null Space LDA (NLDA) using QR Decomposition

The NLDA method suggested in [16] makes assumptions about the linear independence of the training data vectors in order to simplify things. Within the NLDA method, there is only a one-step economical QR decomposition of an $m \times (n-1)$ matrix if each training data vectors are linearly independent [17].

Taking into consideration that the subtraction vector is constructed from the first samples of each class, then $b_j^{(i)} = a_{j+1}^{(i)} - a_1^{(i)}$, for $i = 1, \dots, k$ and $j = 1, \dots, n_i - 1$. H_w^{diff} and H_b^{diff} defined, as

$$H_w^{diff} = \left[b_1^1, \dots, b_{n_1-1}^1, b_1^2, \dots, b_{n_2-1}^2, \dots, b_1^k, \dots, b_{n_k-1}^k\right] \in \mathbb{R}^{m \times (n-k)}$$

Without a doubt, S_w and H_w share the same range space. In this case, H_b^{diff} is defined as

$$H_b^{diff} = [f_1, f_2, \dots, f_{k-1}] \in \mathbb{R}^{m\times(k-1)}$$

where $f_i = c^{(i+1)} - c^{(1)}$ for $i = 1, \dots, k$.

Assume that n data points in the data matrix $A \in \mathbb{R}^{m\times n}$ are linearly independent. It means $rank(S_b) + rank(S_w) = rank(S_t)$, which is true in many applications with high-dimensional data [17], then we obtain $\gamma = rank(S_t) = n - 1$ and $[H_w^{diff}, H_b^{diff}] \in \mathbb{R}^{m\times(n-1)}$ is a full column rank matrix.

Let $q = rank(H_w^{diff})$, $\gamma = rank([H_w^{diff}, H_b^{diff}])$, and the economic QR decomposition [24] of the matrix $[H_w^{diff}, H_b^{diff}] \in \mathbb{R}^{m\times(n-1)}$ be

$$[H_w^{diff}, H_b^{diff}] = [\tilde{Q}_1, \tilde{Q}_2]\begin{bmatrix}\tilde{R}_{11} & \tilde{R}_{12}\\ 0 & \tilde{R}_{22}\end{bmatrix}$$

where $\tilde{Q}_1 \in \mathbb{R}^{m\times q}$ and $\tilde{Q}_2 \in \mathbb{R}^{m\times(\gamma-q)}$ are column orthogonal matrices, $\tilde{R}_{11} \in \mathbb{R}^{q\times(n-k)}$ and $\tilde{R}_{22} \in \mathbb{R}^{(\gamma-q)\times(k-1)}$ are of full row rank. Next, $\tilde{Q}_2$ solves the optimization problem [17].

From the explanation above, it can be seen that if *n* data points in the data matrix $A \in \mathbb{R}^{m\times n}$ are linearly independent, then the optimization problem may be solved using one step QR decomposition.

The NLDA algorithm using QR decomposition is shown in Table 3 [17].

Table 3. NLDA Algorithm using QR Decomposition

NLDA algorithm using QR decomposition
Input: matrix A whose columns are linearly independent Output: transformation matrix G
1. From the matrices H_w^{diff} and H_b^{diff}, 2. Calculate the QR decomposition of $[H_w^{diff}, H_w^{diff}]$ i.e. $[H_w^{diff}, H_w^{diff}] = [\tilde{Q}_1, \tilde{Q}_2]\begin{bmatrix}\tilde{R}_{11} & \tilde{R}_{12}\\ 0 & \tilde{R}_{22}\end{bmatrix}$; 3. $G = \tilde{Q}_2$.

In general, machine learning is more effective on bigger datasets for pattern recognition, therefore applying it to datasets with a small sample size is certain to cause problems. Machine learning's accuracy and resilience degrade with decreasing dataset sizes. Sparsity is an inherent property of high-dimensional spaces [25]. Information extraction from limited datasets, deep learning techniques for data augmentation, and dimensionality reduction in complicated big data analyses are some of the approaches that have been investigated in an effort to address these issues [26].

2.5 Evaluation Method

The evaluation is carried out by computing the accuracy of each NLDA model. Find the accuracy value with the help of the Confusion Matrix. An actual value vs the model's predicted value comparison is shown via a confusion matrix [27]. Confusion Matrix is used as a metric to analyze how machine learning classifiers perform on datasets, making it possible to define a wide variety of performance metrics. Figure 1 is an overview of the Confusion Matrix.

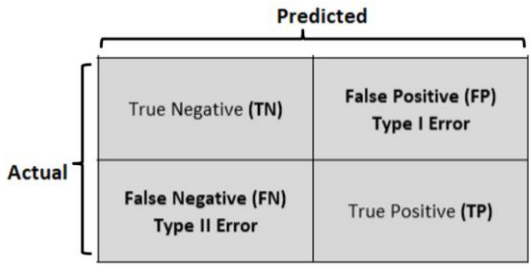

Figure 1. Confusion Matrix

TP value means the number of correct positive predictions. FP value means the example is actually negative, but the classifier marked it as positive. FN value means the example marked by the classifier is negative but actually positive. Finally, TN is where the examples are entirely wrong.

When the dataset contains more than two classes, the matrix grows with multiple classes. For example, if there are three classes, then the matrix is a 3 x 3 matrix. Whatever the size of the confusion matrix, the method for interpreting it is the same.

The accuracy of a prediction system is defined as the proportion of accurate predictions to total data [28]. Formula 6 is used to determine the accuracy value.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

3. Results and Discussions

3.1 Results

Experimental studies are conducted and discussed in this section to evaluate the three NLDA methods. We compared three NLDA methods by visualizing the results and showing their classification performance. Python programming was used in this experiment to model the NLDA method, calculate the linear discriminant for each model, and evaluate the experimental results. The evaluation method in section 2.5 is used for all NLDA methods.

Experiments were conducted on two datasets obtained from Openml web and Orange software (https://www.openml.org/search?type=data&status=active&id=45087). The 2000-dimensional Colon Cancer dataset, consisting of 62 data samples and two classes, and the 467-dimensional TKI-resistance dataset, consisting of 280 data samples and three classes, were divided into two, 70% for training and 30% for testing. The linear discriminant (LD) result can be calculated by multiplying the testing data matrix A with the transformation matrix G, which is then used to create data visualizations as in Figure 2 and Figure 3.

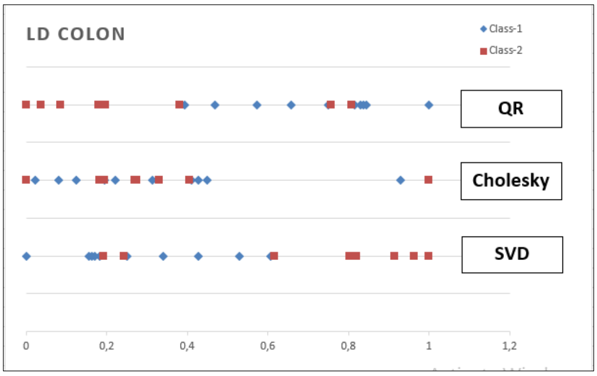

Figure 2. Visualization of Three NLDA Models from the Colon Dataset

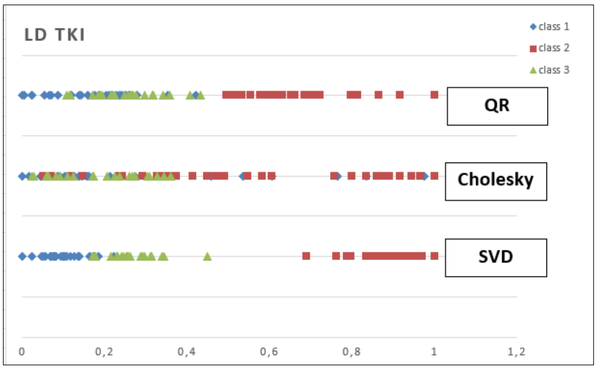

Figure 3. Visualization of Three NLDA Models from the TKI-resistance Dataset

Excel was used to create the visual representation seen in the graph above. Next, we use Formula 5 to create a Confusion Matrix which is presented in Figure 4 and Figure 5. Meanwhile, Table 4 shows the accuracy of all NLDA models.

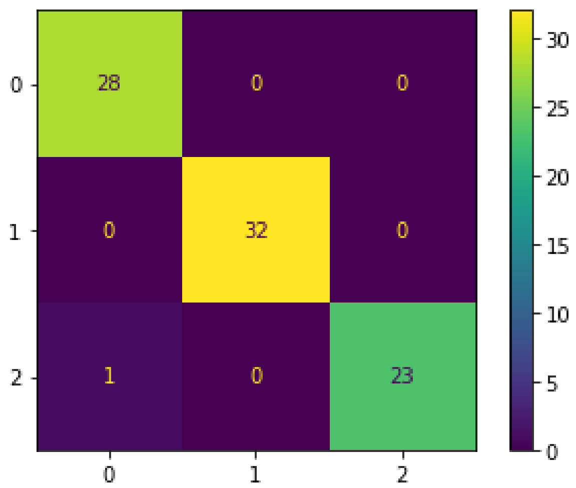

Figure 4. Confusion Matrix of TKI-resistance Dataset with NLDA using SVD

Table 4. The Accuracy of Three NLDA Methods

Dataset	Using SVD	Using Cholesky Decomposition	Using QR Decomposition
Colon	77.8%	66.6%	77.8%
TKI-resistance	98.8%	64.3%	80.9%

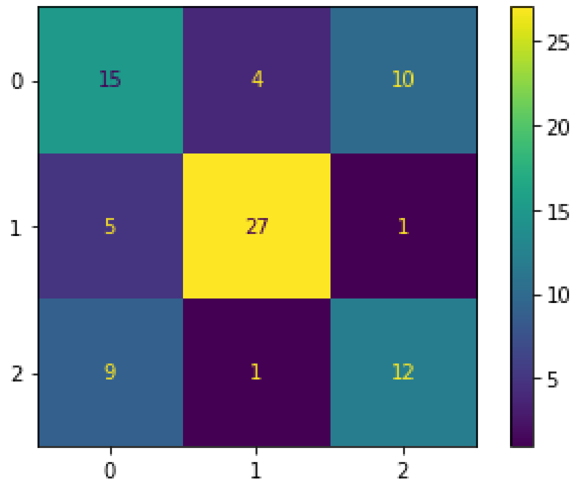

Figure 5. Confusion Matrix of TKI-resistance Dataset with NLDA using Cholesky Decomposition

3.2 Discussions

Figure 2 is the linear projection diagram of the Colon dataset with the NLDA method, using SVD, Cholesky decomposition, and QR decomposition, respectively. Similarly, Figure 3 is a linear projection diagram of the TKI-resistance dataset with the NLDA method, using SVD, Cholesky decomposition, and QR decomposition, respectively.

Figure 2 shows that QR and SVD have good class separation between class 1 and class 2, although overfitting still occurs in both. It means that the best NLDA method for the Colon dataset uses SVD and QR decomposition, as evidenced by the accuracy results in Table 4, which is 77.8%, both with SVD and QR decomposition. While in Figure 2 shows that SVD has the best class separation between class 1, class 2, and class 3. It means that the best NLDA method for the TKI-resistance dataset uses SVD, which can be proven from the accuracy results in Table 4, which is 98.8%.

Figure 4 and Figure 5 show the Confusion Matrix of the TKI-resistance dataset, using the NLDA method with SVD and Cholesky Decomposition, respectively. The numbers 0, 1, and 2 on the left side and bottom indicate the class name, while the colour indicates the amount of data. Figure 4 shows that almost all data is classified correctly, i.e. 28 data in class 0, 32 data in class 1, and 23 data in class 2. There is only 1 data that is classified in class 0, but in fact, it is in class 2. In contrast to Figure 5, 30 data have been misclassified. Only a few data are classified correctly, i.e. 15 data in class 0, 27 data in class 1, and 12 data in class 2.

The difference in results in the three methods can occur because of the difference in treatment after obtaining the scatter matrix. In the first method, the H_t matrix is used to calculate the SVD, thus obtaining the U matrix to calculate the transformation matrix G. The second method requires the H_w^{rem} matrix to calculate the Cholesky decomposition, thus obtaining the R matrix, the lower triangular matrix. However, the Cholesky decomposition can only be calculated if the matrix $(H_w^{rem})^T H_w^{rem}$ is non-singular and positive definite [23]. Since the matrix is singular, SVD is still used in this method. The H_{nul} matrix is used to replace the U matrix (SVD result matrix) to obtain the transformation matrix G. The combination of the H_w^{diff} and H_b^{diff} matrices in the third method is used to calculate the QR decomposition, where the Q matrix is needed to find the transformation matrix G.

In the second method, subtraction of the last sample of each class is applied to the H_w matrix. Similarly, in the third method, subtraction of the first sample of each class is applied to the H_w matrix. It was not applied in the first method. It turned out that these treatments did not give better results for the Colon and TKI-resistance datasets. It means that NLDA using SVD gives the best results for these datasets.

4. Conclusions

In this paper, we discuss three NLDA models using different matrix decompositions, i.e., SVD, Cholesky decomposition, and QR decomposition. In particular, we compare the steps taken after obtaining the scatter matrix. The scatter matrix in NLDA with Cholesky and QR decomposition is treated almost the same. Experiments on two datasets have shown the effectiveness of the three NLDA methods. The SVD approach outperforms the others on this dataset in terms of accuracy and class separation in the visualization output of 77.8% for the Colon dataset and 98.8% for the TKI-resistance dataset. Future research is expected to develop variations of the NLDA method to improve accuracy and reduce overfitting on the Colon and TKI-resistance datasets.

References

[1] A. Benouareth, "An efficient face recognition approach combining likelihood-based sufficient dimension reduction and LDA," *Multimed. Tools Appl.*, vol. 80, no. 1, pp. 1457–1486, 2021, doi: 10.1007/s11042-020-09527-9.

[2] B. Li, Z. T. Fan, X. L. Zhang, and D. S. Huang, "Robust dimensionality reduction via feature space to feature space distance metric learning," *Neural Networks*, vol. 112, pp. 1–14, 2019, doi: 10.1016/j.neunet.2019.01.001.

[3] K. S. Chandrasekar and P. Geetha, "A new formation of supervised dimensionality reduction method for moving vehicle classification," *Neural Comput. Appl.*, vol. 33, no. 13, pp. 7839–7850, 2021, doi: 10.1007/s00521-020-05524-z.

[4] J. P. Gygi, S. H. Kleinstein, and L. Guan, "Predictive overfitting in immunological applications: Pitfalls and solutions," *Hum. Vaccines Immunother.*, vol. 19, no. 2, 2023, doi: 10.1080/21645515.2023.2251830.

[5] Y. Li, B. Liu, Y. Yu, H. Li, J. Sun, and J. Cui, "3E-LDA: Three Enhancements to Linear Discriminant Analysis," *ACM Trans. Knowl. Discov. Data*, vol. 15, no. 4, 2021, doi: 10.1145/3442347.

[6] Z. He, M. Wu, X. Zhao, S. Zhang, and J. Tan, "Representative null space LDA for discriminative dimensionality reduction," *Pattern Recognit.*, vol. 111, p. 107664, 2021, doi: 10.1016/j.patcog.2020.107664.

[7] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2nd ed)*, 2nd ed., no. April. New York: John Wiley & Sons, 2000.

[8] A. Sharma and K. K. Paliwal, "Linear discriminant analysis for the small sample size problem: an overview," *Int. J. Mach. Learn. Cybern.*, vol. 6, no. 3, pp. 443–454, 2015, doi: 10.1007/s13042-013-0226-9.

[9] I. Husein and R. Widyasari, "Algorithm Symmetric 2-DLDA for Recognizing Handwritten Capital Letters," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 21, no. 2, pp. 389–402, 2022, doi: 10.30812/matrik.v21i2.1254.

[10] S. Benkhaira and A. Layeb, "Face recognition using RLDA method based on mutated cuckoo search algorithm to extract optimal features," *Int. J. Appl. Metaheuristic Comput.*, vol. 11, no. 2, pp. 118–133, 2020, doi: 10.4018/IJAMC.2020040106.

[11] M. I. Afjal, M. N. I. Mondal, and M. Al Mamun, "Segmented Linear Discriminant Analysis for Hyperspectral Image Classification," *12th Int. Conf. Electr. Comput. Eng. ICECE 2022*, no. December, pp. 204–207, 2022, doi: 10.1109/ICECE57408.2022.10088677.

[12] J. Ye and T. Xiong, "Computational and theoretical analysis of null space and orthogonal linear discriminant analysis," *J. Mach. Learn. Res.*, vol. 7, pp. 1183–1204, 2006.

[13] A. A. Joseph *et al.*, "Online Person Identification based on Multitask Learning," *Int. J. Integr. Eng.*, vol. 13, no. 2, pp. 119–126, 2021, doi: 10.30880/ijie.2021.13.02.014.

[14] J. Liu, X. Cai, and M. Niranjan, "GO-LDA : Generalised Optimal Linear Discriminant Analysis," pp. 1–15, 2023.

[15] R. Huang, Q. Liu, H. Lu, and S. Ma, "Solving the small sample size problem of LDA," *Proc. - Int. Conf. Pattern Recognit.*, vol. 16, no. 3, pp. 29–32, 2002, doi: 10.1109/icpr.2002.1047787.

[16] D. Chu and G. S. Thye, "A new and fast implementation for null space based linear discriminant analysis," *Pattern Recognit.*, vol. 43, no. 4, pp. 1373–1379, 2010, doi: 10.1016/j.patcog.2009.10.004.

[17] G. F. Lu and Y. Wang, "Feature extraction using a fast null space based linear discriminant analysis algorithm," *Inf. Sci. (Ny).*, vol. 193, pp. 72–80, 2012, doi: 10.1016/j.ins.2012.01.015.

[18] A. Sharma and K. K. Paliwal, "A new perspective to null linear discriminant analysis method and its fast implementation using random matrix multiplication with scatter matrices," *Pattern Recognit.*, vol. 45, no. 6, pp. 2205–2213, 2012, doi: 10.1016/j.patcog.2011.11.018.

[19] G. F. Lu and W. Zheng, "Complexity-reduced implementations of complete and null-space-based linear discriminant analysis," *Neural Networks*, vol. 46, pp. 165–171, 2013, doi: 10.1016/j.neunet.2013.05.010.

[20] A. Zaib, T. Ballal, S. Khattak, and T. Y. Al-Naffouri, "A doubly regularized linear discriminant analysis classifier with automatic parameter selection," *IEEE Access*, vol. 9, pp. 51343–51354, 2021, doi: 10.1109/ACCESS.2021.3068611.

[21] L. F. Chen, H. Y. M. Liao, M. T. Ko, J. C. Lin, and G. J. Yu, "New LDA-based face recognition system which can solve the small sample size problem," *Pattern Recognit.*, vol. 33, no. 10, pp. 1713–1726, 2000, doi: 10.1016/S0031-3203(99)00139-9.

[22] B. H. S. Utami, T. Trisnawati, R. Pratiwi, and M. Gumanti, "Robust Singular Value Decomposition Method on Minor Outlier Data," *J. Varian*, vol. 4, no. 1, pp. 19–24, 2020, doi: 10.30812/varian.v4i1.857.

[23] M. Jurek and M. Katzfuss, "Hierarchical sparse Cholesky decomposition with applications to high-dimensional spatio-temporal filtering," *Stat. Comput.*, vol. 32, no. 1, pp. 1–29, 2022, doi: 10.1007/s11222-021-10077-9.

[24] I. A. Olajide, "Examination of QR Decomposition and the Singular Value Decomposition Methods," *J. Multidiscip. Eng. Sci. Stud.*, vol. 7, no. 4, 2021.

[25] F. Konietschke, K. Schwab, and M. Pauly, "Small sample sizes: A big data problem in high-dimensional data analysis," *Stat. Methods Med. Res.*, vol. 30, no. 3, pp. 687–701, 2021, doi: 10.1177/0962280220970228.

[26] P. Kokol, M. Kokol, and S. Zagoranski, "Machine learning on small size samples: A synthetic knowledge synthesis," *Sci. Prog.*, vol. 105, no. 1, pp. 1–16, 2022, doi: 10.1177/00368504211029777.

[27] I. Markoulidakis, I. Rallis, I. Georgoulas, G. Kopsiaftis, A. Doulamis, and N. Doulamis, “Multiclass Confusion Matrix Reduction Method and Its Application on Net Promoter Score Classification Problem,” *Technologies*, vol. 9, no. 4, 2021, doi: 10.3390/technologies9040081.

[28] R. Sistem, “JURNAL RESTI Analysis And Classification of Customer Churn Using Machine Learning,” vol. 5, no. 158, pp. 1253–1259, 2023.