Journal of Computer and Communications, 2024, 12, 211-230
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

An Application of Machine Learning to Thalassemia Diagnosis

Sitan Liu[1,2]

[1]School of Mathematics and Statistics, Guilin University of Technology, Guilin, China
[2]Guangxi Colleges and Universities Key Laboratory of Applied Statistics, Guilin, China
Email: 1597413075@qq.com

How to cite this paper: Liu, S.T. (2024) An Application of Machine Learning to Thalassemia Diagnosis. *Journal of Computer and Communications*, **12**, 211-230.
https://doi.org/10.4236/jcc.2024.122013

Received: January 21, 2024
Accepted: February 26, 2024
Published: February 29, 2024

Abstract

Mediterranean anemia is a genetic disease that currently relies heavily on expert clinical experience to determine whether patients are affected. This method is overly reliant on expert experience and is not precise enough. This paper proposes two modeling methods to predict whether patients have Mediterranean anemia. The first method involves using Principal Component Analysis (PCA) to reduce the dimensionality of the data, followed by logistic regression modeling (PCA-LR) on the reduced dataset. The second method involves building a Partial Least Squares Regression (PLS) model. Experimental results show that the prediction accuracy of the PCA-LR model is 87.5% ($degree = 2$, $\lambda = 40$), and the prediction accuracy of the PLS model is 92.5% ($ncomp = 4$), indicating good predictive performance of the models.

Keywords

Multicollinearity, Statistical Analysis Models, Data Mining, PCA-LR, PLS

1. Introduction

Thalassemia, a hereditary chronic hemolytic disease, is caused by the deficiency or mutation of globin genes that impede the synthesis of hemoglobin [1]. It was first discovered and named by Thomas Cooley and Pear Lee, Italian researchers, along the coast of the Mediterranean Sea in 1925. According to the statistical data of the World Health Organization (WHO) in 2008, about 300,000 to 400,000 thalassemia patients are born worldwide each year, accounting for 17% of the global population as carriers of thalassemia genes [2]. Approximately 18.7% of beta-thalassemia major neonates require regular blood transfusions to sustain life, and about 10% of affected children die in the neonatal period. The mortality rate of children under five years old is as high as 3.4%, posing a significant threat

to people's health [3].

As a monogenic hereditary disease, thalassemia is widely distributed in parts of Africa, the Middle East, and Asia. However, there are significant variations in the screening programs for thalassemia due to differences in the level of medical development in different countries and regions [4]. Common screening methods include single-factor analysis and combined screening. Additionally, even with the same screening protocol, each country may set different thresholds for blood parameters based on regional influences [5]. For example, the common hematological parameter, Mean Corpuscular Volume (MCV), has a threshold of 80 FL in the Yunnan region of China, while it is set at 82 FL in other regions, resulting in regional variations in screening outcomes [6].

Research on screening for thalassemia patients can be divided into three stages. In the first stage, due to the underdevelopment of the medical field, screening methods mainly relied on medical tests or post-onset blood parameter screening, lacking systematic mathematical data collection and analysis methods [7]. The second stage introduced the application of statistical methods, where screening methods for thalassemia were primarily based on the statistical results of certain indicators [8], such as MCV, MCH, and HbA2. However, this stage still remained at the stage of manual screening and statistical analysis, thus increasing the possibility of misdiagnosis and risk index to some extent. In the third stage, screening methods based on machine learning models gradually emerged. For example, Yi-Kai used Support Vector Machine (SVM) to differentiate between beta-thalassemia and non-beta-thalassemia microcytic anemia [6]. However, the data studied by Yi-Kai did not start from the perspective of gene detection, but from the perspective of blood, resulting in a lower algorithm accuracy rate.

Thalassemia is a prevalent and debilitating genetic disorder in the local population, with the severity of symptoms increasing with the accumulation of gene deletions [9]. Individuals with severe thalassemia have a short lifespan, and if identified and addressed during early pregnancy, measures can be taken to control the birth of children with severe thalassemia, reducing unnecessary suffering and loss. Therefore, considering the characteristics of existing machine learning algorithms and techniques, this study proposes the construction of a warning model for thalassemia screening based on machine learning algorithms, as well as further research on risk factors. This has significant academic and practical implications.

2. Materials and Methods

The data used in this study were sourced from real clinical records at a hospital in the Guangxi Zhuang Autonomous Region, China. The dataset consists of a total of 60 individuals' genetic samples, with each sample containing 110 different genes, resulting in a total of 110 observed indicators or variables. Prior to conducting data analysis, strict privacy protection measures were taken. All personally identifiable information that could identify patient identities was removed, and the data underwent de-identification procedures.

Due to the relatively small sample size and high dimensionality of the data used in this study, issues such as data sparsity and distance calculation pose significant challenges for all machine learning methods [10]. This is commonly referred to as the "*curse of dimensionality*" [11].

In this context, dimension reduction is considered an important approach [12]. It involves a mathematical transformation that converts the original high-dimensional attribute space into a lower-dimensional "*subspace*" to identify more suitable observed variables for modeling. Dimension reduction effectively reduces the data's dimensionality, improves model training efficiency, and better addresses the curse of dimensionality. Next, we will provide a detailed introduction to the dimension reduction method and machine learning model used in this article.

2.1. Principal Component Analysis

Principal Component Analysis (abbreviated as PCA) is a widely used data dimensionality reduction algorithm [13]. Its main idea is to map the original n-dimensional features onto a new k-dimensional space, which is composed of entirely new orthogonal features, also known as principal components.

The process of PCA involves sequentially searching for a set of mutually orthogonal axes, where the selection of these new axes is closely related to the data itself [14]. The first new axis chosen is the direction of maximum variance in the original data. The second new axis is then selected as the direction of maximum variance in the plane orthogonal to the first axis. The third axis is selected as the direction of maximum variance in the plane orthogonal to the first two axes, and so on, until we obtain n such axes.

By following this approach, most of the variance is captured by the first k axes, while the remaining axes contain almost no variance. Therefore, we can ignore the remaining axes and only retain the first k axes that contain the majority of the variance [14] [15]. In practice, this means keeping the feature dimensions that capture the significant variance and disregarding the ones with negligible variance, thus achieving dimensionality reduction of the data features.

The algorithmic steps of PCA are shown in **Algorithm 1**.

Algorithm 1. PCA.

Input: The sample set $D=\{x_1,x_2,\cdots,x_m\}$;

The dimension of the low-dimensional subspace is denoted as d'.

Output: The projection matrix W.

1: Centering all samples: $x_i \leftarrow x_i - \frac{1}{m}\sum_{i=1}^{m} x_i$.

2: Calculate the covariance matrix XX^{T}.

3: Perform eigenvalue decomposition on the covariance matrix XX^{T}.

4: Take the eigenvectors corresponding to the d' largest eigenvalues $W=w_1,w_2,\cdots,w_{d'}$.

2.2. Partial Least Squares Regression

Partial Least Squares Regression (abbreviated as PLS) is a commonly used statistical analysis method for finding the relationship between independent and dependent variables [16]. It combines the characteristics of principal component analysis and canonical correlation analysis, as well as linear regression analysis. PLS Regression can effectively handle problems such as multicollinearity and small sample size among independent variables.

In the process of PLS, a new space is created by projecting the independent and dependent variables onto it. This new space is characterized by principal components, which are new variables obtained through linear transformations of the original independent variables. The method is effective at handling issues such as multicollinearity and small sample size among independent variables. The parameters of PLS are estimated by minimizing the sum of squared residuals, resulting in the establishment of a linear regression model.

In comparison to traditional multiple linear regression models, PLS exhibits the following distinctive features [17]:

1) When there is severe multicollinearity among independent variables, traditional regression models may encounter issues. However, PLS can handle regression modeling in such cases and reduce the impact of collinearity among independent variables on the results.

2) In situations where the number of data points is fewer than the number of variables, traditional regression analysis methods may suffer from overfitting problems. On the other hand, PLS can perform regression modeling under such conditions, improving the stability and reliability of the model.

3) The regression coefficients in PLS are more interpretable for each independent variable, facilitating a better understanding of the relationship between the independent and dependent variables.

In summary, PLS performs well in addressing challenges such as multicollinearity and small sample size, while also providing more interpretable regression coefficients [18].

2.3. Logistic Regression

Logistic Regression (abbreviated as LR) is a classical statistical learning method commonly used to solve binary classification problems [19]. It predicts the probability of a sample belonging to a certain category by establishing a LR model [20]. For example, it can be used to predict the likelihood of a user purchasing a certain product, a patient having a particular disease, or a user clicking on a certain advertisement.

The LR model is based on the concept of linear regression [20], but with a special function transformation known as the logistic or sigmoid function. This conversion maps the output to probability values between 0 and 1. The function that LR aims to fit is as follows:

$$h_\theta(x) = \theta^{\mathrm{T}} x = \sum_{i=0}^{n} \theta_i x_i = \theta_0 + \theta_1 x_1 + \cdots + \theta_n x_n \tag{1}$$

The cost function of LR is represented by the following equation:

$$J(\theta) = -y\log(h_\theta(x)) - (1-y)\log(1-h_\theta(x)) \tag{2}$$

The dependent variable y can only take values of 0 or 1, while it is difficult for the independent variable x to truly reach positive or negative infinity. Therefore, the range of $h_\theta(x)$ being (0, 1), without truly reaching the two endpoint values, suggests that the cost function can be considered as a conditional function:

$$J(\theta) = \begin{cases} -\log(h_\theta(x)) & y=1 \\ -\log(1-h_\theta(x)) & y=0 \end{cases} \tag{3}$$

To find the minimum points of the cost function, we need to equate the partial derivatives to zero. Here, we take the partial derivative of $J(\theta)$:

$$\frac{\partial J(\theta)}{\partial \theta_j} = \frac{1}{m}\sum_{i=1}^{m}\left(h_\theta\left(x^{(i)}\right) - y^{(i)}\right)x_j^{(i)} \tag{4}$$

In classification problems, overfitting can easily occur when the sample size is too small. To achieve reasonable fitting results, there are two methods: The first method is reducing the number of parameters or limiting the coefficient values within a certain range [21]. Regularization is an example of the second method. At this point, the cost function can be rewritten as:

$$\begin{aligned} J(\theta) = &-\frac{1}{m}\sum_{i=1}^{m}\left[\left(1-y^{(i)}\right)\log\left(1-h_\theta\left(x^{(i)}\right)\right)\right] \\ &-\frac{1}{m}\sum_{i=1}^{m}y^{(i)}\log\left(h_\theta\left(x^{(i)}\right)\right) + \frac{\lambda}{2m}\sum_{j=1}^{n}\theta_j^2 \end{aligned} \tag{5}$$

This can limit the size of θ. It should be noted that when λ is too large, θ can become too small, resulting in underfitting. When λ is too small or even zero, θ can become too large, resulting in overfitting. Therefore, it is important to adjust the appropriate value of λ.

3. Modeling and Result

3.1. Experimental Environment

This study was conducted on the Windows 11 operating system using MATLAB version 2023a. The computer was equipped with an Intel® Core TM i7-12700H processor and 24GB of RAM.

To prevent overfitting, this study randomly divided the data into training and testing sets in a 7:3 ratio. The training set was used to train the model using leave-one-out cross validation [22], and the performance of the model was evaluated on the testing set to validate its predictive effectiveness. Next, we will provide a detailed introduction to the modeling process.

3.2. Modeling

3.2.1. PCA-LR

1) PCA Dimension

Due to the high dimensionality (110 dimensions) and relatively small size of

the data used in this study, it is necessary to perform dimensionality reduction before modeling. By calculating the contribution rate of each principal component, we can determine how many principal components to retain while minimizing information loss.

As shown in Figure 1, as the number of variables increases, the cumulative contribution rate of the first n principal components gradually increases. When the number of variables reaches 14, the cumulative contribution rate of the first 14 principal components has already reached 81.25%. Generally, when the cumulative contribution rate exceeds 80%, a sufficient amount of sample information has been extracted. Therefore, we use the first 14 principal components for modeling analysis.

Visualizing the distribution of different types of patients on the first two principal components, as shown in Figure 2. By observing the distribution in the figure, it can be seen that the majority of patients with thalassemia are concentrated on the right side of the plot, while normal patients are mainly distributed on the left side. This indicates that there is a certain difference in spatial distribution between normal patients and thalassemia patients on the first two principal components.

2) PCA-LR Modeling

The reduced-dimensional data from the previous section is used as the new independent variable, and the patient's condition is input as the response variable into a logistic regression model for modeling.

In the process of logistic regression modeling, two important hyper parameters need to be considered: the highest degree of interaction term (*degree*) and

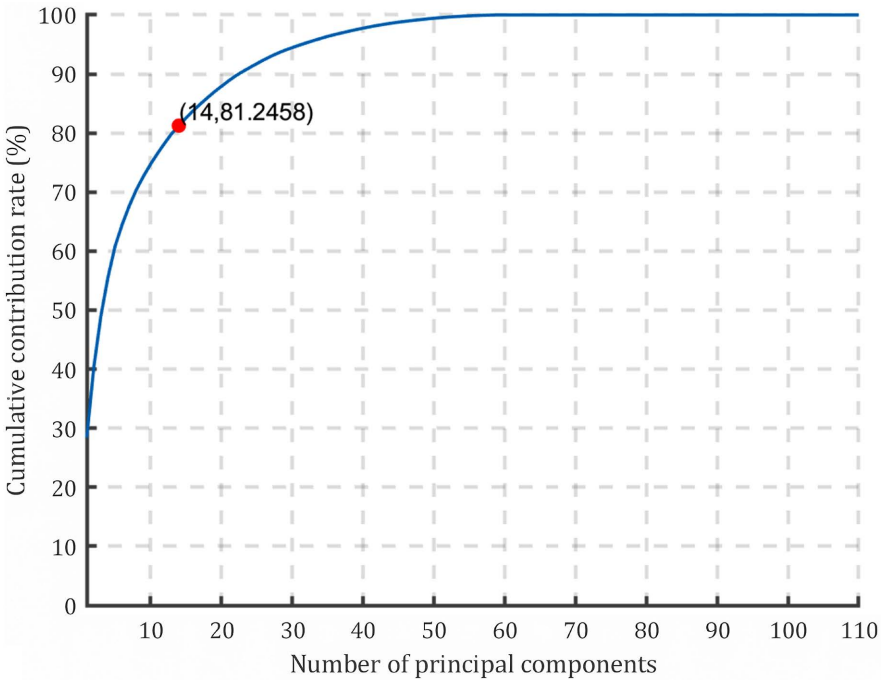

Figure 1. PCA contribution rate curve.

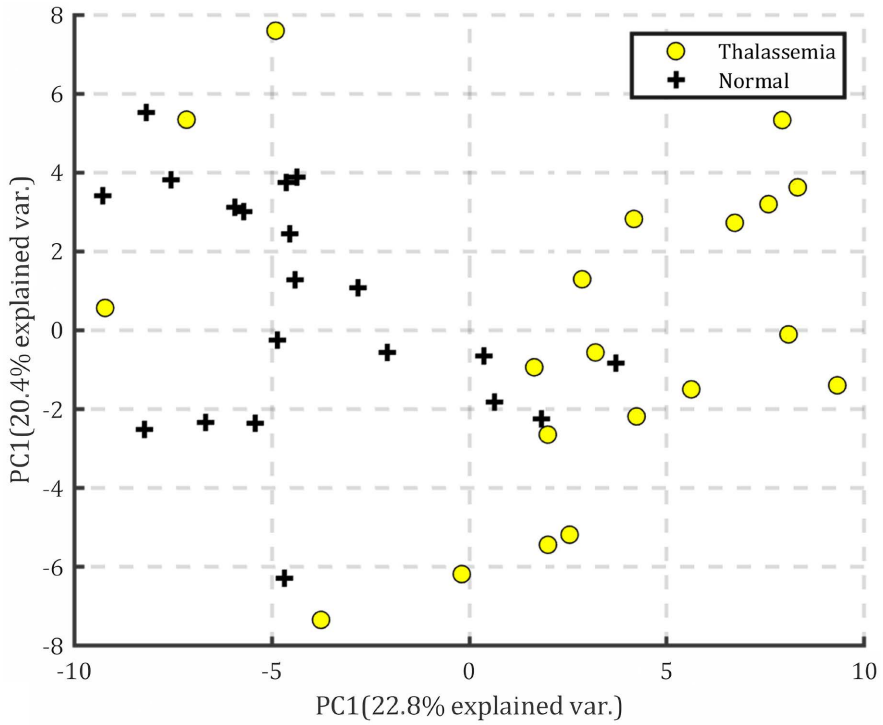

Figure 2. Distribution map of different patients.

the regularization coefficient λ. This section aims to optimize these two parameters and evaluate the model's accuracy on the test set.

We will observe the performance of the model in terms of accuracy based on different values of the regularization coefficient λ and the highest degree of interaction term (*degree*) ranging from 1 to 4.

Figure 3 shows the test accuracy curve of the LR model with *degree* = 1 and λ ranging from 1 to 10^5. It can be seen from the figure that the model's prediction accuracy remains unchanged at 80% as λ increases, and the accuracy is stable. Figure 4 shows the boundary curve (red line in the figure) of the model trained under the first two principal components when the *degree* value is 1 and the regularization coefficient λ is 50. After verification, it was found that the model's boundary remained unchanged regardless of the value of λ, which is also the reason why the accuracy remained unchanged.

Figure 5 shows the test accuracy curve for different values of λ ranging from 1 to 2000 when *degree* is set to 2. As λ increases, the accuracy first decreases and then stabilizes. When $\lambda \approx 40$, the test accuracy reaches its highest point at 87.5%. At λ around 400, the accuracy is 82.5%. When $\lambda \geq 1200$, the accuracy is 75%.

Figures 6-8 respectively show the decision boundaries for λ values of 40, 400, and 1200 when *degree* is set to 2. As λ increases, the decision boundary becomes more and more "elliptical" as can be seen from the graphs, resulting in a batch of misclassified samples and decreasing classification accuracy towards the end.

Figure 9 shows the test accuracy curve for λ values ranging from 1 to 3000

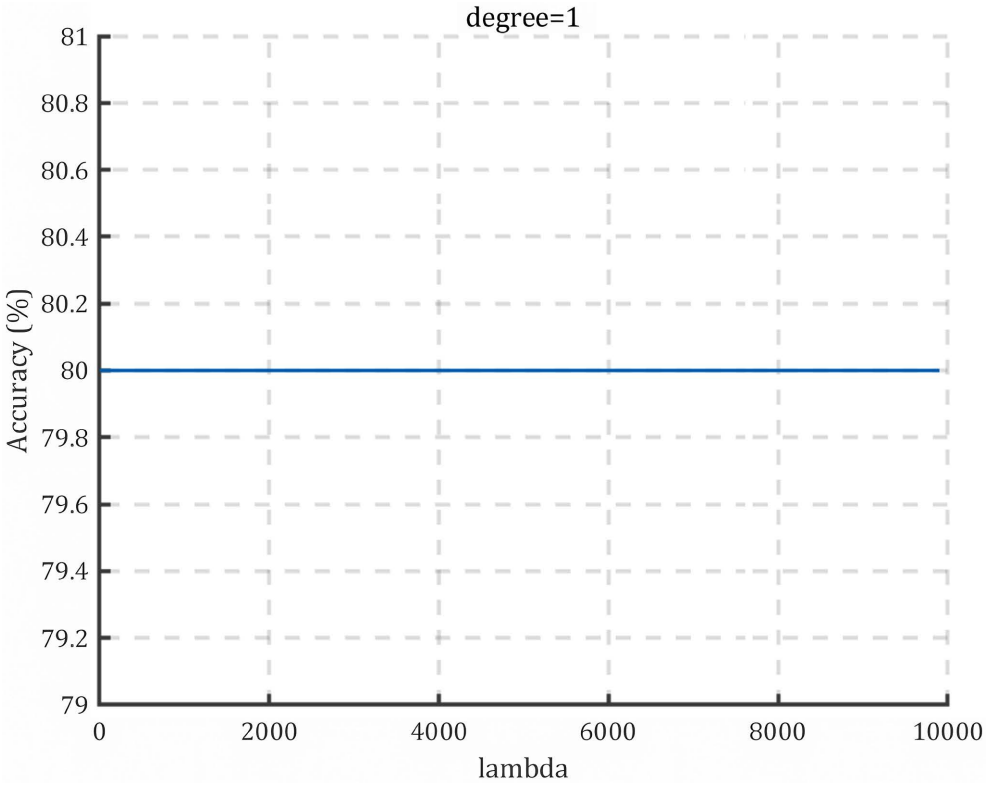

Figure 3. Accuracy curve (*degree* = 1).

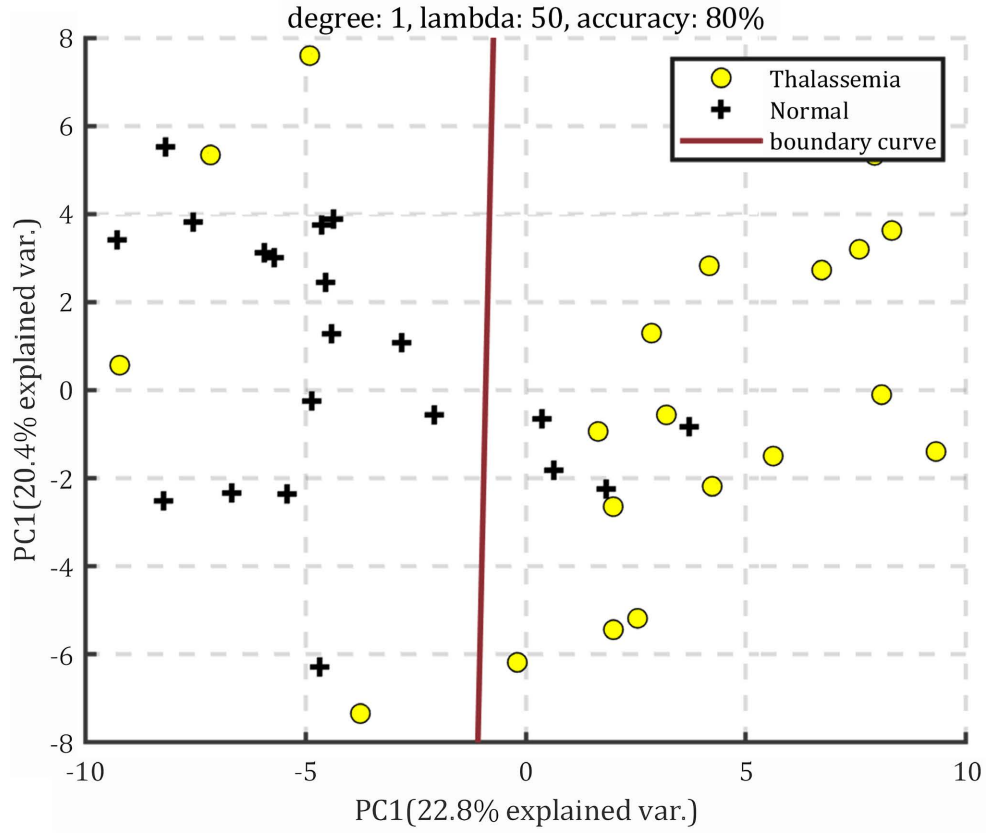

Figure 4. Boundary visualization plot (*degree* = 1, *lambda* = 50).

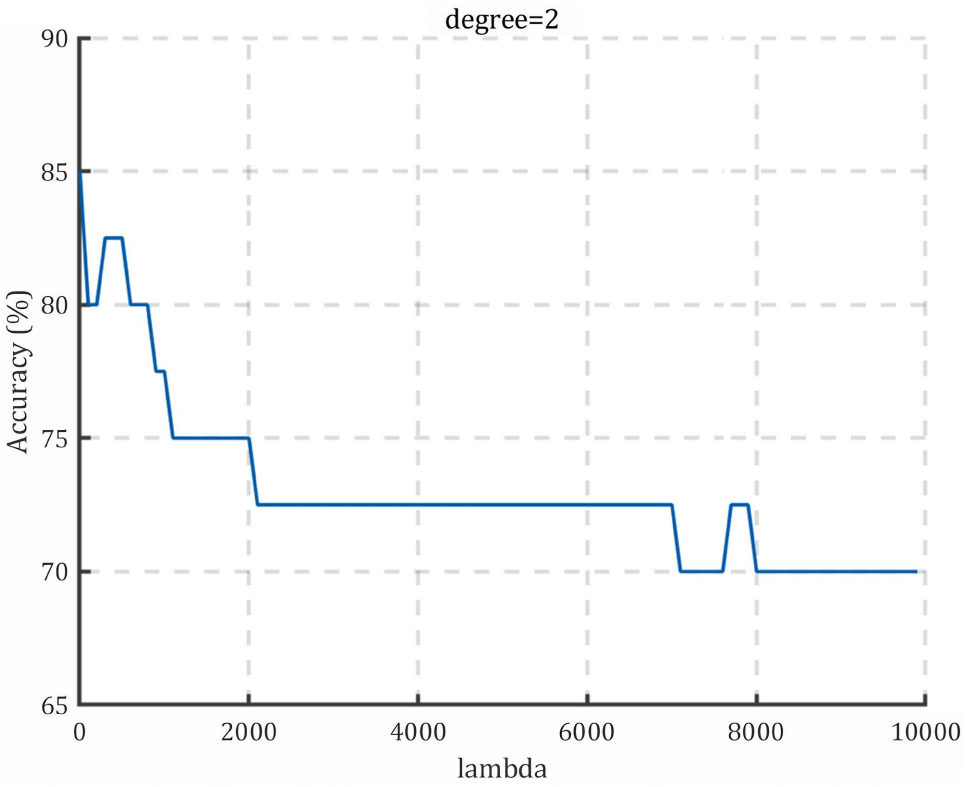

Figure 5. Accuracy curve (*degree* = 2).

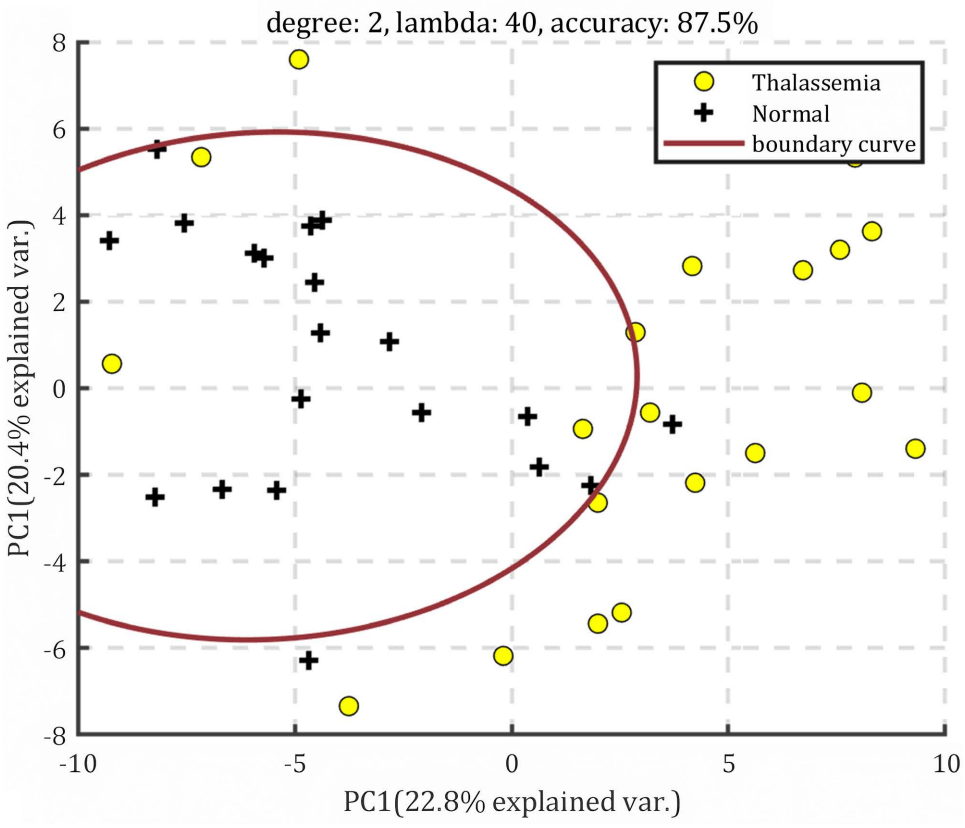

Figure 6. Boundary visualization plot (*degree* = 2, *lambda* = 40).

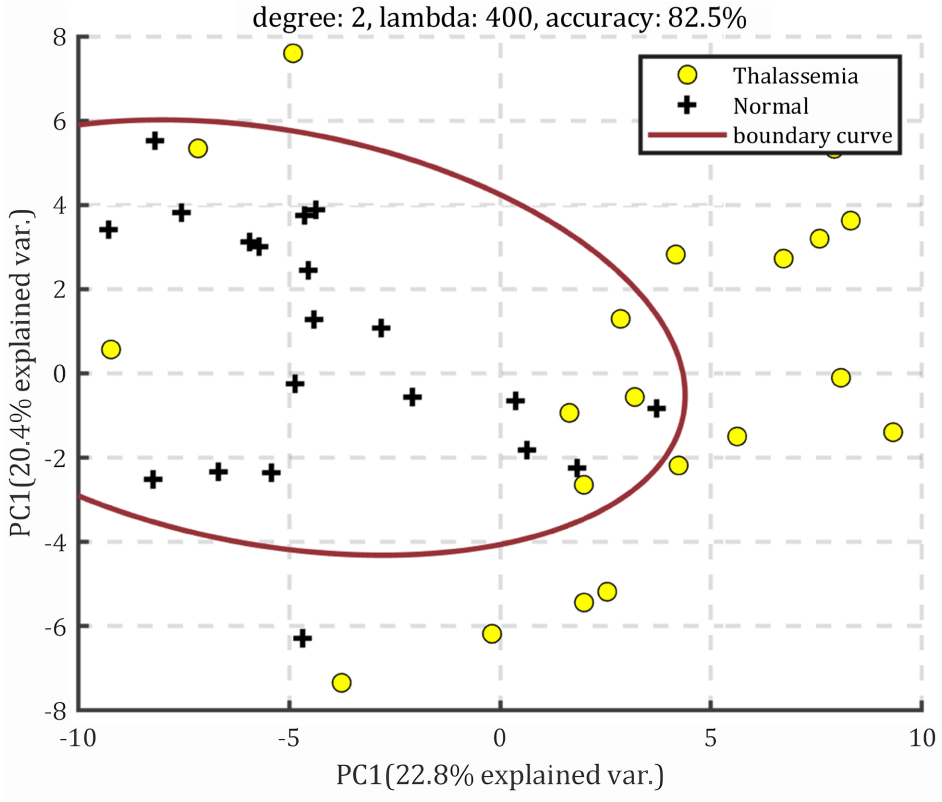

Figure 7. Boundary visualization plot (*degree* = 2, *lambda* = 400).

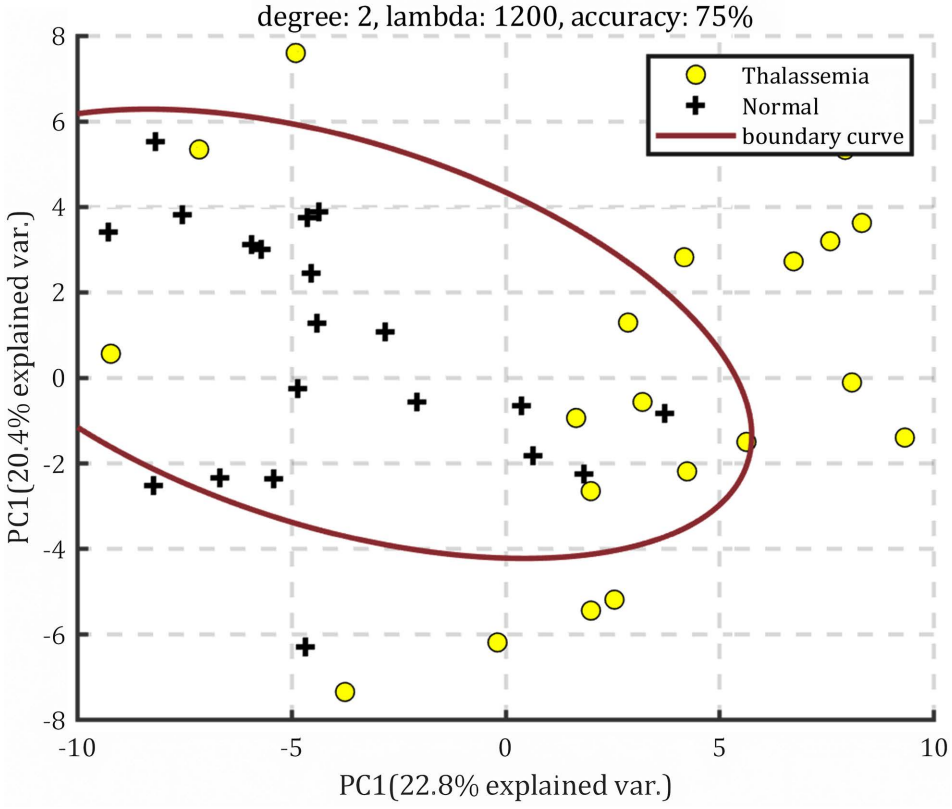

Figure 8. Boundary visualization plot (*degree* = 2, *lambda* = 1200).

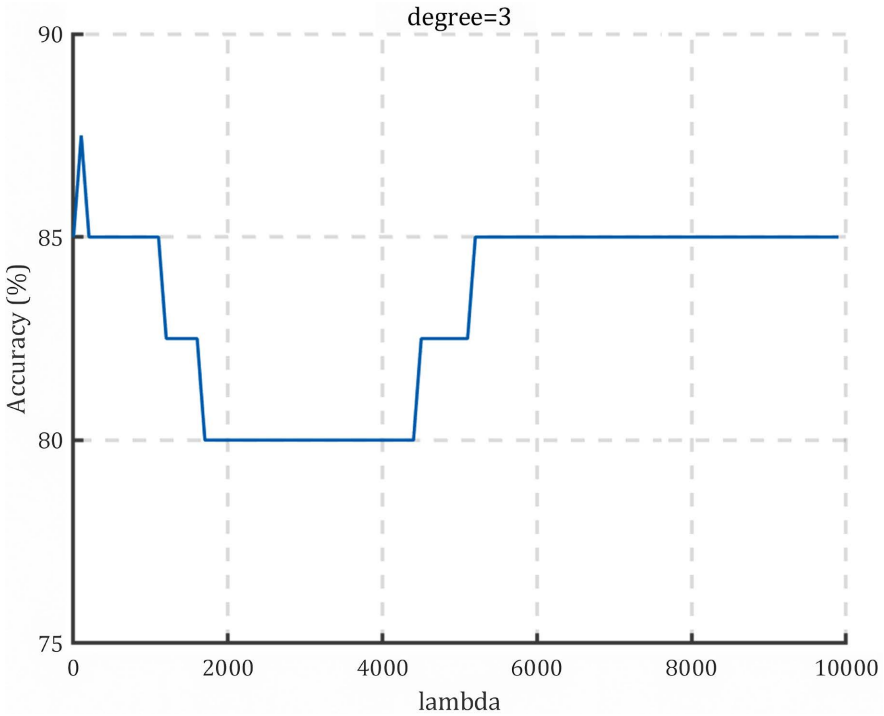

Figure 9. Accuracy curve (*degree* = 3).

when *degree* is set to 3. From the graph, it can be observed that as λ increases, the accuracy initially increases and then decreases. The highest test sample accuracy of 87.5% is achieved at around $\lambda = 50$. At approximately $\lambda = 1200$, the accuracy is 82.5%. For $\lambda \geq 1700$, the accuracy is 80%.

Figures 10-12 respectively illustrate the decision boundaries for λ values of 50, 1200, and 1700 when *degree* is set to 3. From the graphs, it can be seen that as λ increases, the decision boundary becomes more and more insensitive to the points on the boundary. This leads to a batch of misclassified samples and decreasing classification accuracy towards the end.

Figure 13 shows the test accuracy curve for λ values ranging from 1 to 10000 when *degree* is set to 4. From the graph, it can be observed that as λ increases, the accuracy initially increases and then decreases. The highest test sample accuracy of 87.5% is achieved at around $\lambda = 3000$. At approximately $\lambda = 100$, the accuracy is 85%. For $\lambda \geq 6000$, the accuracy is 85%.

Figures 14-16 respectively illustrate the decision boundaries for λ values of 100, 3000, and 6000 when *degree* is set to 4. From the graphs, it can be seen that as λ increases, the decision boundary becomes more and more curved, resulting in a poorer classification of the points on the boundary and decreasing classification accuracy.

3.2.2. PLS Modeling

The plsregress function in MATLAB can be used to implement PLS, with the following syntax:

[Xloadings, Yloadings, betaPLS, PCTVAR] = plsregress(X, y, dims)

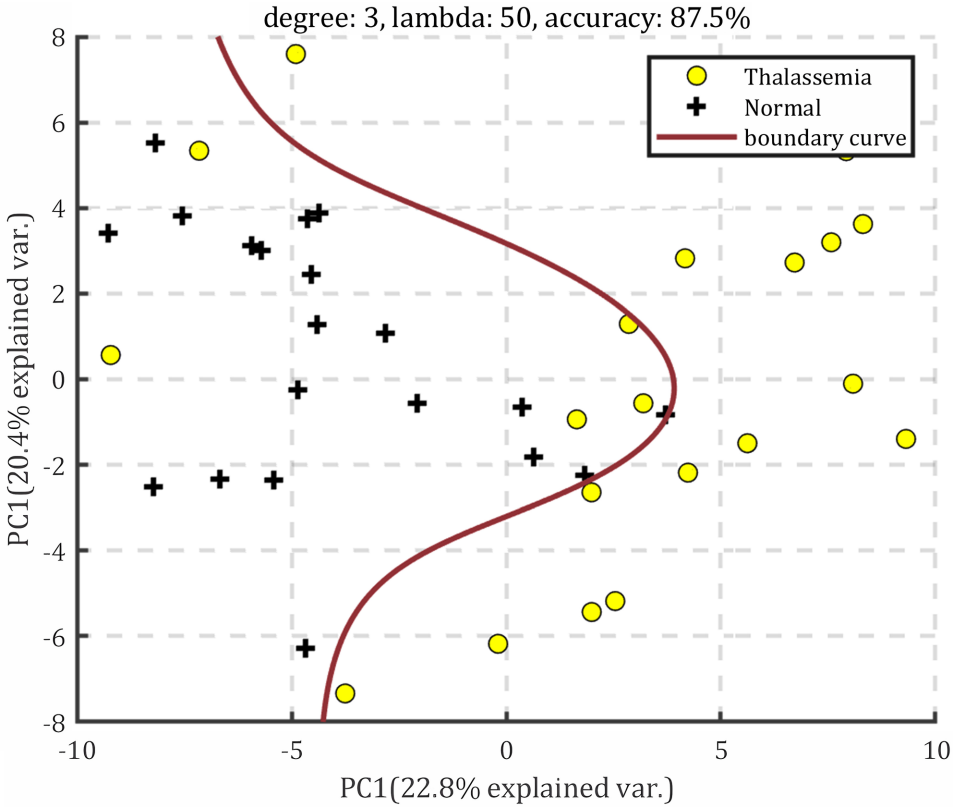

Figure 10. Boundary visualization plot (*degree* = 3, *lambda* = 50).

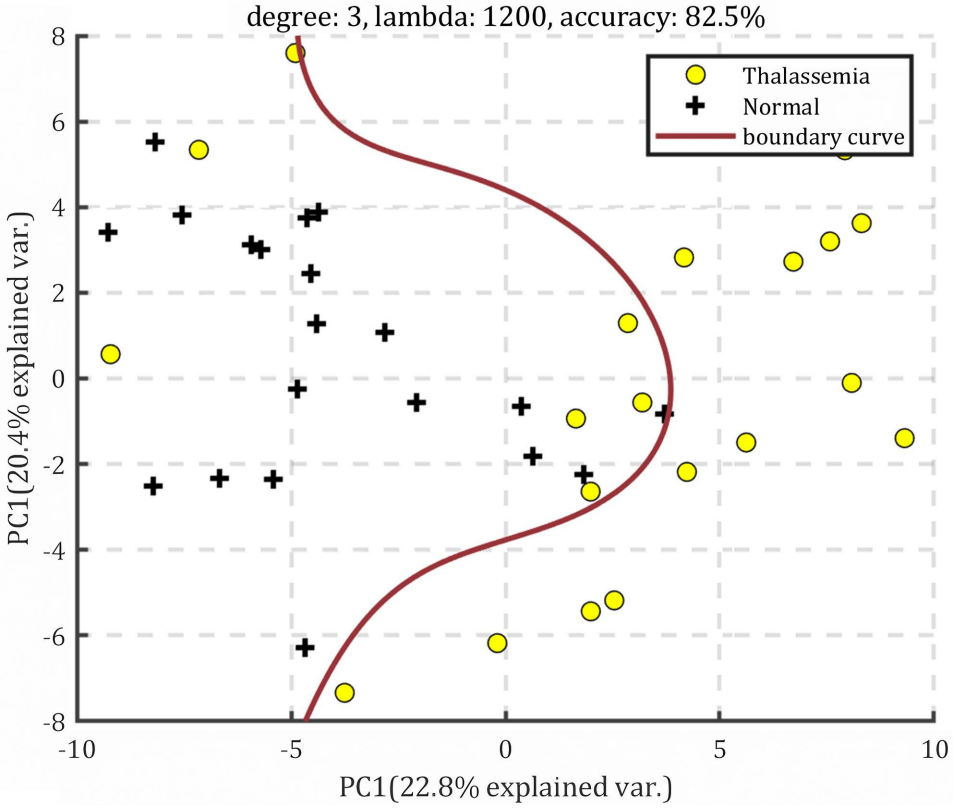

Figure 11. Boundary visualization plot (*degree* = 3, *lambda* = 1200).

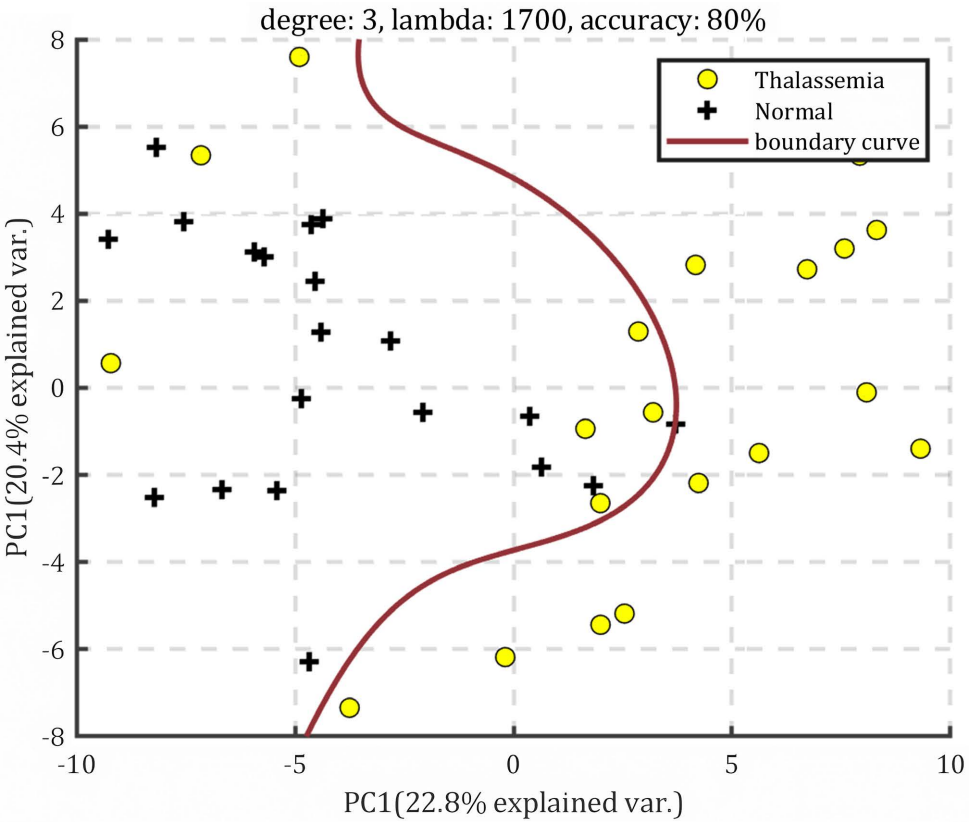

Figure 12. Boundary visualization plot (*degree* = 3, *lambda* = 1700).

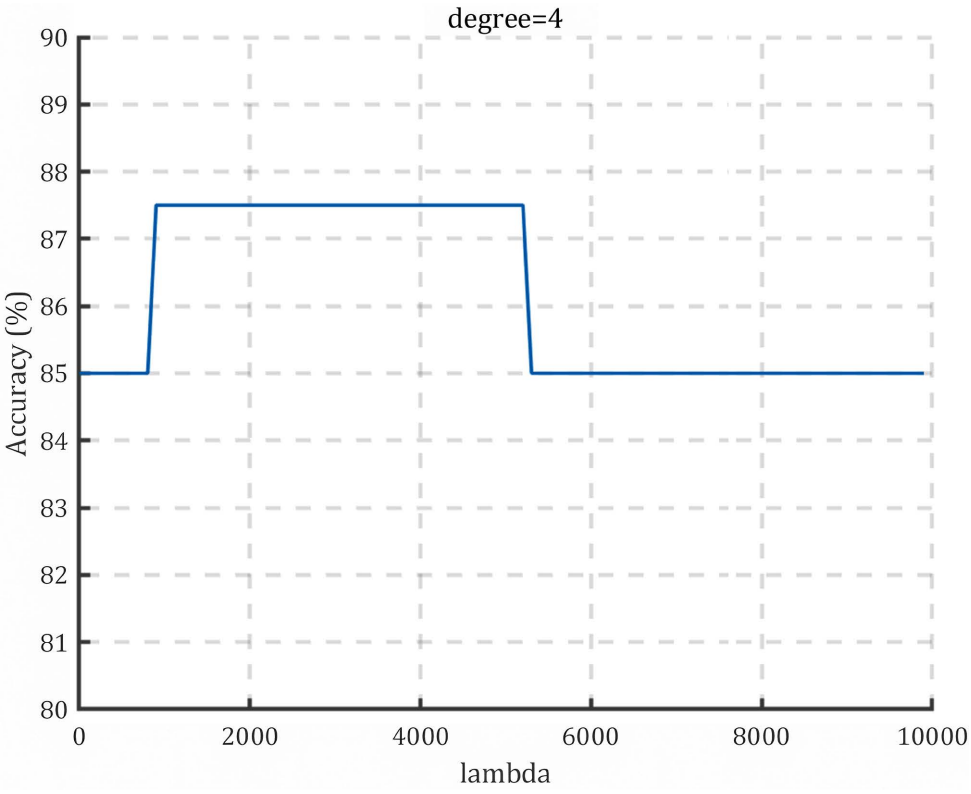

Figure 13. Accuracy curve (*degree* = 4).

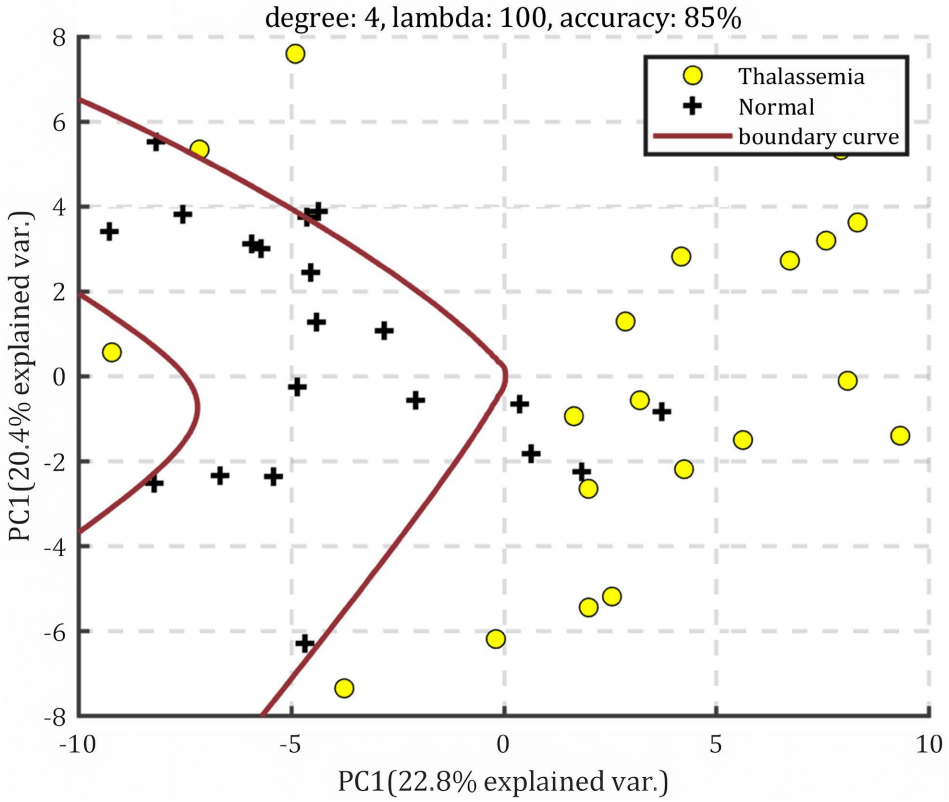

Figure 14. Boundary visualization plot (*degree* = 4, *lambda* = 100).

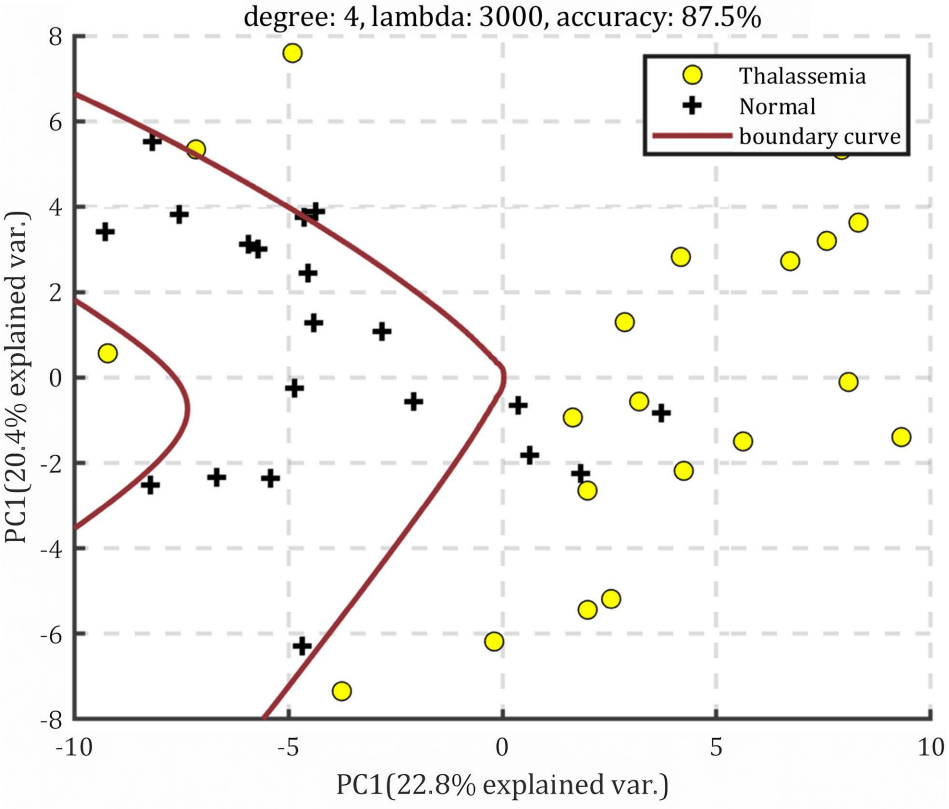

Figure 15. Boundary visualization plot (*degree* = 4, *lambda* = 3000).

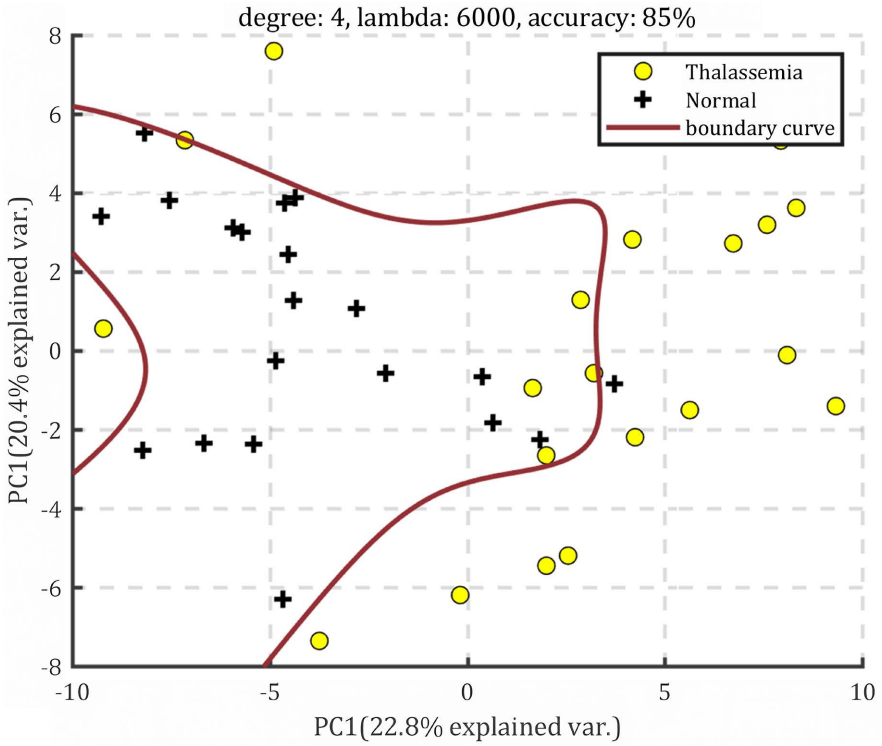

Figure 16. Boundary visualization plot (*degree* = 4, *lambda* = 6000).

In this syntax, dims represent the number of components selected for analysis. PCTVAR can be used to calculate the proportion of dependent variable *y* explained by the first *i* components.

Figure 17 illustrates the proportions of the dependent variable explained by the first 30 components. From the graph, it can be seen that using 10 or more principal components explains over 90% of the variation in the dependent variable.

Due to the large number of variables and severe multicollinearity among them in the data used in this study, not all variables are suitable for modeling. Therefore, we need to select the most important variables for modeling and prediction. Some studies have suggested that Variable Importance in Projection (abbreviated as VIP) can be used to select predictive variables [23], where variables with a VIP score greater than 1 are considered important for predicting the PLS regression model.

The VIP value distributions of different variables are shown in Figure 18, where the red crosses correspond to variables with VIP values greater than 1 and the blue dots correspond to variables with VIP values less than 1. The 30 variables selected by VIP value screening are listed in Table 1.

Using these 30 variables as independent variables and whether a patient has Mediterranean disease as the dependent variable, they are inputted into the PLS model. By gradually increasing the dimensionality ("dims" parameter), the performance of the PLS model can be observed to change, as shown in Figure 19. It can be seen that the model performs best when 4 components are selected, with a corresponding cross-validation accuracy of 92.5%.

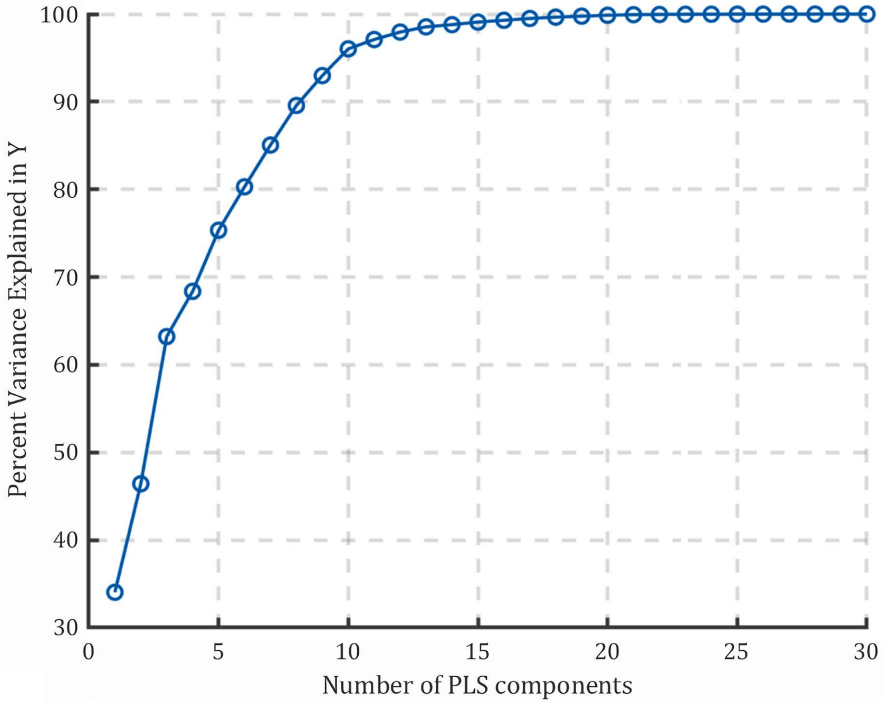

Figure 17. PLS component interpretation.

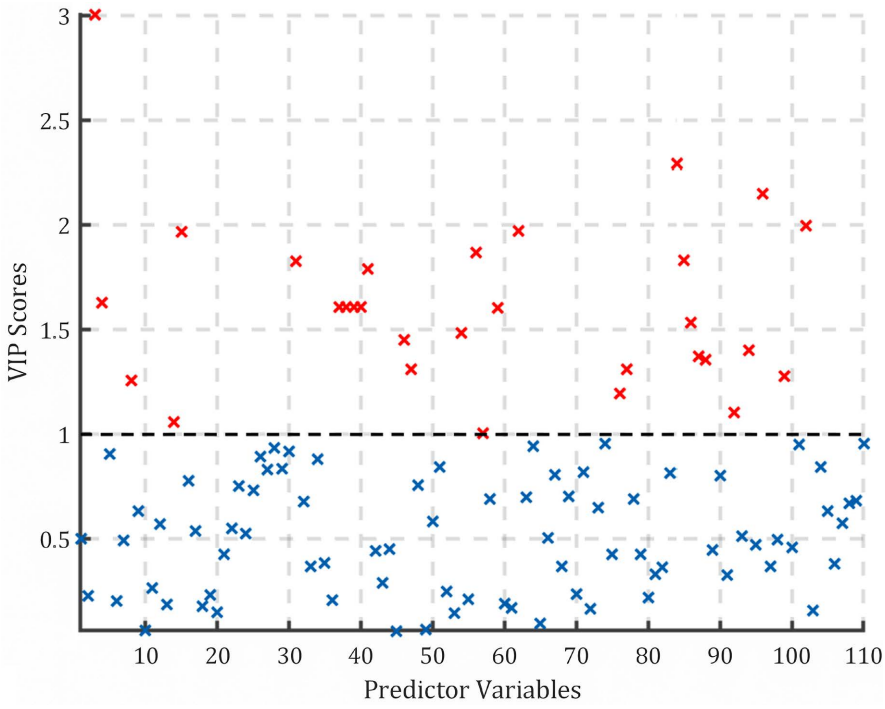

Figure 18. Independent variable VIP values.

3.3. Results Comparison

The accuracy of PCA-LR under different degree conditions and the accuracy of the PLS model are presented in Table 2. It can be observed that the Partial Least Squares Regression (PLS) model demonstrates the highest accuracy among the

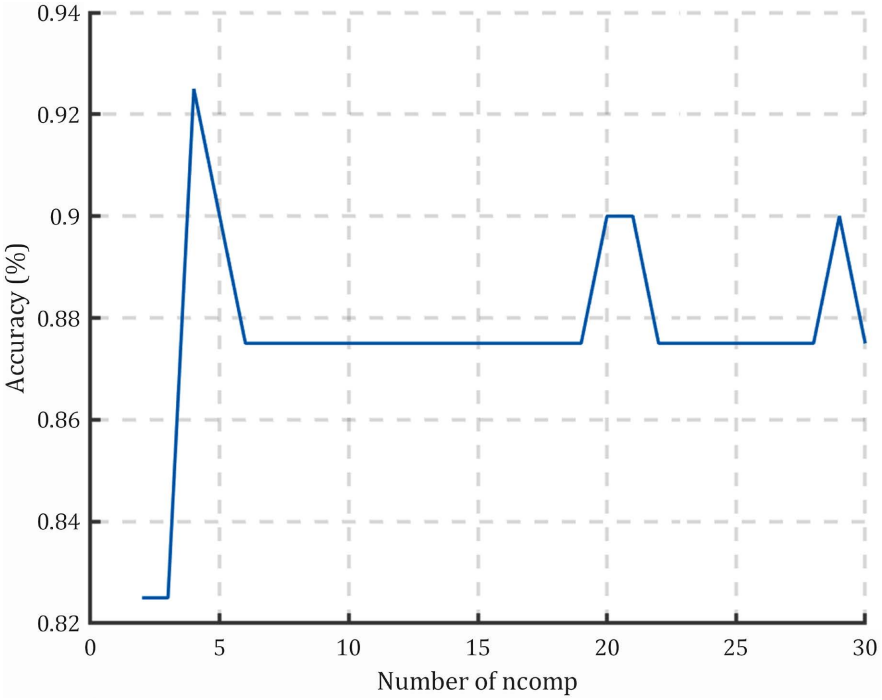

Figure 19. Performance curve of the PLS model.

Table 1. Variable VIP value table.

Variable name	VIP value	Variable name	VIP value	Variable name	VIP value
Gene 3	3.0072	Gene 41	1.7901	Gene 84	2.2934
Gene 4	1.6282	Gene 46	1.4531	Gene 85	1.8321
Gene 8	1.2569	Gene 47	1.3125	Gene 86	1.5362
Gene 14	1.0576	Gene 54	1.4850	Gene 87	1.3751
Gene 15	1.9677	Gene 56	1.8709	Gene 88	1.3577
Gene 31	1.8291	Gene 57	1.0054	Gene 92	1.1042
Gene 37	1.6068	Gene 59	1.6047	Gene 94	1.4029
Gene 38	1.6068	Gene 62	1.9706	Gene 96	2.1503
Gene 39	1.6068	Gene 76	1.1938	Gene 99	1.2770
Gene 40	1.6068	Gene 77	1.3094	Gene 102	1.9950

Table 2. Accuracy comparison of different models.

Model	Best accuracy
PCA-LR (*degree* = 1)	80%
PCA-LR (*degree* = 2)	87.5%
PCA-LR (*degree* = 3)	87.5%
PCA-LR (*degree* = 4)	87.5%
PLS	92.5%

compared models, reaching 92.5%. This indicates that for the dataset under discussion, the PLS model is more effective in capturing the underlying patterns

and relationships.

The PCA-LR model shows an interesting trend with changes in the polynomial degree used. When moving from a first-degree polynomial to a second-degree polynomial, there is a significant increase in accuracy. This suggests that introducing non-linear transformations (by increasing the polynomial degree) can significantly enhance the model's ability to better fit the data. However, when the polynomial degree increases from 2 to 3 and 4, there is no further improvement in accuracy (remaining at 87.5%). This plateau effect indicates that beyond a certain level of complexity (in this case, *degree* = 2), increasing model complexity does not necessarily equate to better performance. This may be because the model has already captured most of the variance in the data with a second-degree polynomial, and additional degrees only add complexity without improving the model's predictive capability.

4. Conclusions

PLS is an excellent modeling algorithm that is suitable for small samples and high-dimensional data, and it can also handle multicollinearity issues. In this study, we utilized the PLS model for the discrimination and diagnosis of Mediterranean anemia in the Guangxi region. To ensure the reliability of the model, we employed leave-one-out cross-validation and split validation set methods for modeling analysis. The results show that the model established has a high accuracy rate, demonstrating the effectiveness of this method.

Due to the limitations of the sample data, this paper did not explore its application in the diagnosis of different subtypes and clinical stages of Mediterranean anemia, which is a direction for future research. Additionally, research on how to integrate this algorithm into existing medical systems or mobile health applications to enhance its practicality and convenience can also be considered.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Cao, A. and Galanello, R. (2010) Beta-Thalassemia. *Genetics in Medicine*, **12**, 61-76. https://doi.org/10.1097/GIM.0b013e3181cd68ed

[2] Saleem, M., Aslam, W., Lali, M.I.U., *et al.* (2023) Predicting Thalassemia Using Feature Selection Techniques: A Comparative Analysis. *Diagnostics*, **13**, Article 3441. https://doi.org/10.3390/diagnostics13223441

[3] Ferih, K., Elsayed, B., Elshoeibi, A.M., *et al.* (2023) Applications of Artificial Intelligence in Thalassemia: A Comprehensive Review. *Diagnostics*, **13**, Article 1551. https://doi.org/10.3390/diagnostics13091551

[4] Singh, A., Mora, J. and Panepinto, J.A. (2018) Identification of Patients with Hemoglobin SS/Sβ^0 Thalassemia Disease and Pain Crises within Electronic Health Records. *Blood Advances*, **2**, 1172-1179.

https://doi.org/10.1182/bloodadvances.2018017541

[5] Das, R., Saleh, S., Nielsen, I., *et al.* (2022) Performance Analysis of Machine Learning Algorithms and Screening Formulae for β-Thalassemia Trait Screening of Indian Antenatal Women. *International Journal of Medical Informatics*, **167**, Article ID: 104866. https://doi.org/10.1016/j.ijmedinf.2022.104866

[6] Fu, Y.K., Liu, H.M., Lee, L.H., *et al.* (2021) The TVGH-NYCU Thal-Classifier: Development of a Machine-Learning Classifier for Differentiating Thalassemia and Non-Thalassemia Patients. *Diagnostics*, **11**, Article 1725. https://doi.org/10.3390/diagnostics11091725

[7] Angelucci, E., Muretto, P., Lucarelli, G., *et al.* (1997) Phlebotomy to Reduce Iron Overload in Patients Cured of Thalassemia by Bone Marrow Transplantation. *Blood*, **90**, 994-998. https://doi.org/10.1182/blood.V90.3.994

[8] Xie, F., Ye, L., Chang, J.C., *et al.* (2014) Seamless Gene Correction of β-Thalassemia Mutations in Patient-Specific iPSCs Using CRISPR/Cas9 and *piggyBac*. *Genome Research*, **24**, 1526-1533. https://doi.org/10.1101/gr.173427.114

[9] Ren, Z., Sun, G., Zhang, Q., *et al.* (2023) LC-MS/MS-Based Absolute Quantitation of Hemoglobin Subunits from Dried Blood Spots Reveals Novel Biomarkers for α-Thalassemia Silent Carriers. *Analytical Chemistry*, **95**, 9244-9251. https://doi.org/10.1021/acs.analchem.3c00895

[10] Giraldo, L.F., Lozano, F. and Quijano, N. (2011) Foraging Theory for Dimensionality Reduction of Clustered Data. *Machine Learning*, **82**, 71-90. https://doi.org/10.1007/s10994-009-5156-0

[11] Abdelmoula, W.M., Stopka, S.A., Randall, E.C., *et al.* (2022) massNet: Integrated Processing and Classification of Spatially Resolved Mass Spectrometry Data Using Deep Learning for Rapid Tumor Delineation. *Bioinformatics*, **38**, 2015-2021. https://doi.org/10.1093/bioinformatics/btac032

[12] Zhou, C., Li, Y., Wu, W., *et al.* (2023) Preparation and Performance Analysis of a Dimension-Controlled Nano-Drag-Reducing Agent for Low-Permeability Reservoirs. *Energy and Fuels*, **37**, 3908-3917. https://doi.org/10.1021/acs.energyfuels.3c00077

[13] Luo, L., He, G., Chen, C., *et al.* (2022) Adaptive Data Dimensionality Reduction for Chemical Process Modeling Based on the Information Criterion Related to Data Association and Redundancy. *Industrial & Engineering Chemistry Research*, **61**, 1148-1166. https://doi.org/10.1021/acs.iecr.1c04926

[14] Chabriel, G., Kleinsteuber, M., Moreau, E., *et al.* (2014) Joint Matrices Decompositions and Blind Source Separation: A Survey of Methods, Identification, and Applications. *IEEE Signal Processing Magazine*, **31**, 34-43. https://doi.org/10.1109/MSP.2014.2298045

[15] Kanavaki, A., Spengos, K., Moraki, M., *et al.* (2017) Serum Levels of S100b and NSE Proteins in Patients with Non-Transfusion-Dependent Thalassemia as Biomarkers of Brain Ischemia and Cerebral Vasculopathy. *International Journal of Molecular Sciences*, **18**, Article 2724. https://doi.org/10.3390/ijms18122724

[16] Yin, S., Zhu, X. and Kaynak, O. (2015) Improved PLS Focused on Key-Performance-Indicator-Related Fault Diagnosis. *IEEE Transactions on Industrial Electronics*, **62**, 1651-1658. https://doi.org/10.1109/TIE.2014.2345331

[17] Wold, S., Kettaneh, N. and Tjessem, K. (2015) Hierarchical Multiblock PLS and PC Models for Easier Model Interpretation and as an Alternative to Variable Selection. *Journal of Chemometrics*, **10**, 463-482. https://doi.org/10.1002/(SICI)1099-128X(199609)10:5/6%3C463::AID-CEM445%3E3.0.CO;2-L

[18] You, L.X. and Chen, J.H. (2022) Autogenerated Multilocal PLS Models without Pre-Classification for Quality Monitoring of Nonlinear Processes with Unevenly Distributed Data. *Industrial & Engineering Chemistry Research*, **61**, 5898-5913. https://doi.org/10.1021/acs.iecr.1c04461

[19] Betül, Ç., Ayyldz, H. and Tuncer, T. (2020) Discrimination of *β*-Thalassemia and Iron Deficiency Anemia through Extreme Learning Machine and Regularized Extreme Learning Machine Based Decision Support System. *Medical Hypotheses*, **138**, Article ID: 109611. https://doi.org/10.1016/j.mehy.2020.109611

[20] Saraf, S.L., Akingbola, T.S., Shah, B.N., *et al*. (2016) Genetic Modifiers Identify a High Risk Group for Stroke in Three Independent Cohorts of Sickle Cell Anemia Patients. *Blood*, **128**, 1015. https://doi.org/10.1182/blood.V128.22.1015.1015

[21] Paokanta, P., Ceccarelli, M., Harnpornchai, N., *et al*. (2012) Rule Induction for Screening Thalassemia Using Machine Learning Techniques: C5.0 and CART. *ICIC Express Letters*, **6**, 301-306.

[22] Paokanta, P., Ceccarelli, M. and Srichairatanakool, S. (2010) The Effeciency of Data Types for Classification Performance of Machine Learning Techniques for Screening *β*-Thalassemia. 2010 3*rd International Symposium on Applied Sciences in Biomedical and Communication Technologies* (*ISABEL* 2010), Rome, 7-10 November 2010, 1-4. https://doi.org/10.1109/ISABEL.2010.5702769

[23] Ergon, R. (2004) Informative PLS Score-Loading Plots for Process Understanding. *Journal of Process Control*, **14**, 889-897. https://doi.org/10.1016/j.jprocont.2004.02.004