Semi-Supervised Kernel Discriminative Low-Rank Ridge Regression for Data Classification

Qi Zhu
School of Computer Science and Technology, Hangzhou Dianzi University, China
zhuqi7@hdu.edu.cn

Yong Peng
School of Computer Science and Technology, Hangzhou Dianzi University, China
yongpeng@hdu.edu.cn

Abstract: *Regression problem is currently a popular research topic in the field of machine learning. However, most existing research directly performs linear classification on the data after simple preprocessing, or performs classification after feature selection. It usually does not take into account the characteristics of the sample itself, especially for data that is linearly inseparable in original dimensional space and often produces unsatisfactory classification performance. Furthermore, the method of simply mapping the data into a kernel space using kernel trick before classification makes data classification more complex. It also results in unsatisfactory classification performance. In this paper, a simple yet effective semi-supervised Kernel discriminative Low-Rank Ridge Regression (KLRRR) model is proposed for data classification, which unifies kernel trick and discriminant subspace projection together. Specifically, the data is first mapped into kernel space to deal with the linear inseparability problem in original dimensional space, and then the projection matrix in the least square regression is decomposed into the product of two factor matrices to complete the joint discriminant subspace projection and regression. Experiments on 12 benchmark data sets show that the proposed KLRRR model greatly improves the classification performance in comparison with some state-of-the-arts.*

Keywords: *Discriminative subspace, low-rank regression, kernel space, ridge regression, semi-supervised classification*.

Received February 16, 2024; accepted August 26, 2024
https://doi.org/10.34028/iajit/21/5/3

1. Introduction

Regression problem has always been a popular research topic in the machine learning community. Various regression models have been applied to many aspects, such as image recognition [33], biometric information identification [21], medical image analysis [18], data mining [32] and some other fields [17, 39]. Regression analysis is a statistical method for analyzing data [9]. The purpose is attempting to model the relationship between predictors and the response by fitting a linear equation to the observed data. Generally speaking, through regression analysis we can estimate the conditional values of the dependent variables using known independent variables. Regression is mainly divided into the following types: linear regression [16], logistic regression [14], polynomial regression [22], support vector machine regression [31], decision tree regression [34], forest random regression [4], etc., [19, 30]. Among them, linear regression is the most widely used one in the field of statistics as an analysis method. It determines the quantity relationship of interdependence between several variables. Essentially, relying on linear correlation between variables to create dependence, which is the theoretical basis of the linear regression model. Due to its simplicity and good regression effect, linear regression [28] and its improvements have attracted more and more research enthusiasm in recent years [23].

Generally, a linear regression model contains the following components: the feature matrix, the regression target (i.e., the label matrix), and projection matrix [16]. The model learns the projection matrix [27] to fit the feature matrix and the corresponding label matrix, and then predicts the label of unlabeled data based on the learned projection matrix. However, the data set might be linearly inseparable in original dimensional space [15]. In this case, using linear models for regression tasks will generate unsatisfactory results, and for the complex data set, relying on a single projection matrix to reflect the correspondence between samples and labels is often difficult to obtain better regression results.

Existing studies have proposed two main solutions to solve the problem of linear inseparability of samples in original dimensional space. One is to construct a nonlinear regression model [6, 10], and the other is to the use kernel trick [7, 13]. For the first approach, according to whether a nonlinear model can be transformed into a linear model, methods for solving nonlinear regression models can be divided into the two categories. One is transforming a nonlinear model into a linear model [1]. The other is solving the nonlinear model directly [12]. For the second approach, namely kernel trick, it allows us to non-linearly map data from an original feature space to a kernel space. Kernel trick might enhance the linear separability of data in the kernel space. However, the data often become more

complex after implicit mapping. Relying solely on the projection matrix learned in subsequent linear models to represent the relationship between samples matrix and labels matrix can be overly simplistic. To solve the issue, many methods perform dimensionality reduction on the data before executing the regression task [11, 36]. These approaches extract regression features of data that have a greater impact on the task, thereby reduce the complexity of the data. However, they divide feature selection and regression into two stages inevitably. It breaks the connection between them. Only a few methods consider combining the two parts [24].

This paper proposes a simple and effective semi-supervised Kernel discriminative Low-Rank Ridge Regression (KLRRR) model for data classification. It unifies kernel trick and low-rank structure together, and can widely perform regression tasks on various data. It uses kernel trick to map the data into kernel space to solve the problem of linear inseparability of original dimensional data. Furthermore, mathematically, it replaces the single projection matrix in least squares regression by using the product of two matrices. When performing regression tasks, one factor matrix serves as the discriminant subspace projection to make the data easier to separate, the other factor matrix serves to construct the connection between the data matrix and the corresponding labels matrix in the discriminant subspace. We implement KLRRR model under a semi-supervised paradigm, where soft label matrices of unlabeled samples are jointly estimated to facilitate discriminative subspace identification. We conducted a large number of experiments on 12 benchmark data sets and demonstrated the effectiveness of our proposed model in terms of recognition accuracy, function convergence, and model robustness.

Compared with existing researches, the main contributions of this paper are as follows:

- A new machine learning model KLRRR is proposed, which unifies the kernel trick and the discriminant subspace together. On one hand, the kernel trick tries to solve the data nonlinear separability problem in original dimensional space. On the other hand, KLRRR replaces the single projection matrix in least squares with the product of two factor matrices, which respectively perform discriminant subspace exploration and the establishment of a bridge between data and labels matrices. This solves the issues of increased complexity caused by mapping data into kernel spaces and the inefficiency of a single projection matrix. It leads to a robust connection between the data and the regression target.
- The KLRRR model is implemented within the semi-supervised paradigm. The direct benefit of this approach is that it effectively guides the learning of discriminative subspace projection matrix and the projection matrix between data matrix and labels matrix through soft label matrix estimation of unlabeled sample. Additionally, the semi-supervised paradigm has more practical significance in real-world applications.

The remainder of this paper is structured as follows. In section 2, we introduce the related regression models to this work. The formulation and optimization of the KLRRR model are introduced in detail in section 3. Experiments are performed in section 4 to investigate the effectiveness of the KLRRR model. Section 5 summarizes the full paper and raises our future works.

Notations. In this article, we use Greek letters, such as λ and σ, to represent the parameters in the model. Matrices and vectors in the model are, respectively, denoted by italicized boldface uppercase and italicized lowercase letters. The F-norm of matrix $\boldsymbol{M}\in\mathbb{R}^{m\times n}$ is defined as $\|\boldsymbol{M}\|_F = \left(\sum_{i=1}^{m}\sum_{j=1}^{n} m_{ij}\right)^{1/2}$, m^i is the i-th row of $\boldsymbol{M}$. In particular, 1_n denotes an all-one column vector, where the subscript n indicates its length.

2. Related Works

In this section, we introduce some related works, including kernel regression models and low-rank regression models.

2.1. Kernel Regression Models

Feng *et al*. [8] proposed an innovative model, named Center-based Weighted Kernel Linear Regression (CWKLR). It is designed for object and face recognition. This model is inspired by regression models such as linear regression classification and kernel linear regression classification. The CWKLR model uses the center of each category to form the kernel matrix and test vector. Subsequently, by using the Tikhonov matrix to calculate the coefficients of a weighted projection, CWKLR provides an effective tool for classification tasks. The center-based kernel matrix introduces non-linear information between the training set and class centers, thus promoting more accurate classification. At the same time, the new test vectors also introduces nonlinear information between predicted samples and class centers, allowing the model to effectively model complex relationships.

Sahoo *et al*. [29] proposed an approach termed Online Multiple Kernel Regression (OMKR). This approach requires to get kernel-based regressors in a scalable online manner. Notably, it dynamically explores a diverse pool of multiple kernels, strategically avoiding the pitfalls associated with adhering to a single, suboptimal kernel. It serves as a remedy for the inherent limitations of manual or heuristic kernel selection. The proposed scheme elegantly determines the optimal kernel-based regressor for each different kernels in real time, and simultaneously identify efficient ways to combine multiple kernel regressors. Additionally, a family of OMKR models tailored for

regression were introduced, with a particular focus on their applications in time series prediction. To address the challenge of scalability posed by large data sets, the scheme devises innovative approaches. These strategies are instrumental in mitigating issues arising from an unbounded proliferation of support vectors. Overall, this scheme marks a significant advancement in the field of online kernel regression. It is applicable to various regression scenarios and has scalability for processing large data sets.

2.2. Low-Rank Regression Models

Zhang *et al*. [38] proposed a low-rank-sparse subspace representation for robust regression, termed LRS-RR. They consider that some challenging issues in learning robust regression models for high-dimensional corrupted data. Their approach simultaneously performs low-rank sparse subspace recovery and regression optimization. It exhibits fast convergence, low complexity, and good performance in handling high-dimensional corrupted data through low-rank structures.

Zhang *et al*. [37] proposed to adopt a low-rank structure to obtain the global representation and intrinsic structure of the residual and coefficient matrices. Which ignores the assumption of data being independently and identically distributed. This method introduces a non-convex and non-smooth low-rank regression model guided by the power exponential distribution of extended matrix variables. By incorporating a decomposition strategy into the regression coefficient matrices and utilizing the Schatten-*p* norm with three different *p* values, computational efficiency is enhanced. By introducing auxiliary variables and using singular values, the subproblem is efficiently solved by this method. At the same time, a closed-form solution is obtained by using the proposed multivariable alternating multiplier direction method. The optimization model exhibits good local convergence, computational complexity, and superior performance.

3. Method

In this section, we formulate the objective function of KLRRR model and then introduce the solution and optimization method of this model. In addition, we also perform an analysis on the convergence and time complexity of the KLRRR model's loss function.

3.1. Model Optimization

In semi-supervised learning, we are usually given $\boldsymbol{X}=[\boldsymbol{X}_l, \boldsymbol{X}_u]\in\mathbb{R}^{d\times n}$ to represent a centered data collection matrix, namely feature matrix. This matrix contains l labeled and u unlabeled samples. $\boldsymbol{Y}_l\in\mathbb{R}^{l\times c}$ is a matrix indicating the labels of the corresponding samples, in the form of one-hot encoding. In particular, if the sample $\boldsymbol{x}_i|_{i=1}^{l}$ belongs the j-th class and $\boldsymbol{y}^j\in\mathbb{R}^{1\times c}$ is the i-th row of $\boldsymbol{Y}_l$, then the j-th element in $\boldsymbol{y}^j$ is one and the remaining elements of $\boldsymbol{y}^j$ are zeros. $\boldsymbol{Y}_u$ represents an unknown label matrix corresponding to the unlabeled samples, and $\boldsymbol{Y}=[\boldsymbol{Y}_l;\ \boldsymbol{Y}_u]\in\mathbb{R}^{n\times c}$ is the complete label matrix corresponding to $\boldsymbol{X}$. Meanwhile, d is the dimensionality of samples, c is the number of classes, and $n=l+u$ is the total number of labeled and unlabeled samples. Our purpose is to obtain $\boldsymbol{Y}_u\in\mathbb{R}^{u\times c}$ as accurate as possible given $\boldsymbol{X}$ and $\boldsymbol{Y}_l$.

Usually, connection between feature matrix and the corresponding label matrix is directly built by a single projection matrix. For example, if the F-norm regularization is applied to semi-supervised linear regression, we have the following objective,

$$\begin{aligned}&\min_{W,Y_u}\|\boldsymbol{Y}-\boldsymbol{X}^T\boldsymbol{W}\|_F^2+\lambda\|\boldsymbol{W}\|_F^2,\\&s.t.\ \boldsymbol{Y}_u\geq 0,\boldsymbol{Y}_u\mathbf{1}_c=\mathbf{1}_u.\end{aligned}\tag{1}$$

where $1_c\in\mathbb{R}^c$ and $1_u\in\mathbb{R}^u$ are column vectors with all elements equal to 1. The second constraint requires the summation of row elements in $\boldsymbol{Y}_u$ to be 1. At the same time, considering the nonnegativity of $\boldsymbol{Y}_u$, the elements in each row of $\boldsymbol{Y}_u$ can be considered as the probability of a sample belonging to different classes. Therefore, we can determine the class of a sample by examining the position of the maximal value in each row of $\boldsymbol{Y}_u$. For instance, if the fifth row of $\boldsymbol{Y}_u$ is [0.1, 0.3, 0.6], then the fifth unlabeled sample should be classified into the fifth class.

However, the samples may be linearly inseparable in the original data space, which might be necessary to map them into a kernel space to get better discriminative ability. Considering that the samples are mapped into the kernel space, the linear separability of samples in kernel space can be achieved, we firstly propose the following objective,

$$\begin{aligned}&\min_{W,Y_u}\|\boldsymbol{Y}-\phi^T(\boldsymbol{X})\boldsymbol{W}\|_F^2+\lambda\|\boldsymbol{W}\|_F^2,\\&s.t.\boldsymbol{Y}_u\geq 0,\boldsymbol{Y}_u\mathbf{1}_c=\mathbf{1}_u\end{aligned}\tag{2}$$

where $\phi:R^d\rightarrow\mathcal{H}$ defines a kernel mapping function, λ is the regularization parameter and kernel matrix $\boldsymbol{K}$ can be expressed as the following equation:

$$\boldsymbol{K}_{ij}=\phi^T(\boldsymbol{x_i})\phi(\boldsymbol{x_j})\tag{3}$$

where $\mathcal{H}$ is a reproducing kernel Hilbert space [35].

Further, due to the complexity of data in the kernel space, it is too difficult to establish the connection between the data matrix and the label matrix solely by relying on the projection matrix. It might not capture the kernel space data properties well. A feasible solution is to first project the sample matrix into a discriminant subspace to enhance its separability, and then map this subspace data representation to the corresponding label matrix. To this end, we propose a new model termed KLRRR to seamlessly unify discriminant subspace exploration and semi-supervised recognition together on the basis of kernel space. It is expected to effectively alleviate the data linear inseparability limitation in the original data space. At the same time, it enables the

projection matrix to accurately capture the property relationships between the data matrix and the label matrix in the kernel space.

Supposing that $\boldsymbol{A}\in\mathbb{R}^{n\times s}$, $\boldsymbol{B}\in\mathbb{R}^{s\times c}$, $s<\min(n,c)$, we replace $\boldsymbol{W}$ in Equation (2) with $\phi(\boldsymbol{X})\boldsymbol{AB}$ to get the objective function of our KLRRR model finally. It can be expressed as the following equation

$$\begin{aligned}\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{Y}_u} \|\boldsymbol{Y}-\boldsymbol{KAB}\|_F^2+\lambda\|\phi(\boldsymbol{X})\boldsymbol{AB}\|_F^2,\\ s.t.\,\boldsymbol{Y}_u\geq 0,\boldsymbol{Y}_u\mathbf{1}_c=\mathbf{1}_u.\end{aligned} \tag{4}$$

Based on the definition of *F*-norm, Equation (4) is rewritten to

$$\begin{aligned}\min_{\boldsymbol{A},\boldsymbol{B},\boldsymbol{Y}_u} \|\boldsymbol{Y}-\boldsymbol{KAB}\|_F^2+\lambda\mathrm{Tr}(\boldsymbol{B}^T\boldsymbol{A}^T\boldsymbol{KAB}),\\ s.t.\ \boldsymbol{Y}_u\geq 0,\boldsymbol{Y}_u\mathbf{1}_c=\mathbf{1}_u.\end{aligned} \tag{5}$$

3.2. Model Optimization

There are three variables, i.e., $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{Y}_u$, in the KLRRR model objective function in Equation (5). We propose to solve them in an alternating update manner. Denoting $\Omega(\boldsymbol{A},\ \boldsymbol{B},\ \boldsymbol{Y}u)=\|\boldsymbol{Y}-\boldsymbol{KAB}\|_F^2+\lambda\mathrm{Tr}(\boldsymbol{B}^T\boldsymbol{A}^T\boldsymbol{KAB})$, next, we solve each of the variables within the model.

1. Updating $\boldsymbol{B}$ with $\boldsymbol{A}$ and $\boldsymbol{Y}_u$ fixed. Taking the derivative of $\Omega(\boldsymbol{A},\boldsymbol{B},\boldsymbol{Y}u)$ w.r.t. $\boldsymbol{B}$, we have,

$$\frac{\partial\Omega(\boldsymbol{A},\boldsymbol{B},\boldsymbol{Y}u)}{\partial\boldsymbol{B}}=-2\boldsymbol{A}^T\boldsymbol{K}^T\boldsymbol{Y}+2\boldsymbol{A}^T\boldsymbol{K}^T\boldsymbol{KAB}+2\lambda\,\boldsymbol{A}^T\boldsymbol{KAB}. \tag{6}$$

Setting Equation (6) to zero, we have,

$$\boldsymbol{B}=(\boldsymbol{A}^T(\boldsymbol{K}^T+\lambda\boldsymbol{I})\boldsymbol{KA})^{-1}\boldsymbol{A}^T\boldsymbol{K}^T\boldsymbol{Y} \tag{7}$$

where, $\boldsymbol{I}\in\mathbb{R}^{n\times n}$ is identity matrix.

2. Updating $\boldsymbol{A}$ with $\boldsymbol{B}$ and $\boldsymbol{Y}_u$ fixed. Substituting Equation (7) back into Equation (5), we implement the sub-objective function through variable $\boldsymbol{A}$ as

$$\boldsymbol{A}^*=\underset{\boldsymbol{A}}{\mathrm{argmax}}\{\mathrm{Tr}((\boldsymbol{A}^T(\boldsymbol{K}^T+\lambda\boldsymbol{I})\boldsymbol{KA})^{-1}\boldsymbol{A}^T\boldsymbol{K}^T\boldsymbol{YY}^T\boldsymbol{KA}) \tag{8}$$

Assuming that $\boldsymbol{S}_t=(\boldsymbol{K}^T+\lambda\,\boldsymbol{I})\boldsymbol{K}$, $\boldsymbol{S}_b=\boldsymbol{K}^T\boldsymbol{YY}^T\boldsymbol{K}$, the solution of Equation (8) can be written as,

$$\boldsymbol{A}^*=\underset{\boldsymbol{A}}{\mathrm{argmax}}\{\mathrm{Tr}\,((\boldsymbol{A}^T\boldsymbol{S}_t\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{S}_b\boldsymbol{A})\}. \tag{9}$$

The top s eigenvectors of $\boldsymbol{S}_t^{-1}$, $\boldsymbol{S}_b$ corresponding to the largest s eigenvalues is global optimal solution to $\boldsymbol{A}$. This is a problem of discriminant subspace projection. After we get the optimal solution $\boldsymbol{A}$ with $\boldsymbol{Y}_u$ fixed, Equation (4) is re-write as,

$$\min_{\boldsymbol{B}}\|\boldsymbol{Y}-(\boldsymbol{KA})\boldsymbol{B}\|_F^2+\lambda\|\phi(\boldsymbol{X})\boldsymbol{AB}\|_F^2 \tag{10}$$

the $\boldsymbol{B}$ is doing regularized regression. The optimal solution is given by Equation (7). Thus, with $\boldsymbol{Y}_u$ fixed, the KLRRR of Equation (4) is equivalent to performing ridge regression in a discriminant subspace.

3. Updating $\boldsymbol{Y}_u$ with $\boldsymbol{A}$ and $\boldsymbol{B}$ fixed. We assume $\boldsymbol{Q}\triangleq\boldsymbol{KAB}$, we can get $\boldsymbol{Y}_u$ by solving the following subobjective function,

$$\min_{\boldsymbol{Y}_u}\|\boldsymbol{Y}_u-\boldsymbol{Q}\|_F^2\ s.t.\,\boldsymbol{Y}_u\geq 0,\boldsymbol{Y}_u\mathbf{1}_c=\mathbf{1}_u. \tag{11}$$

Denoting $\boldsymbol{y}^i|_{i=1}^u$ as the i-th row of $\boldsymbol{Y}_u$, by solving the above objective function row by row we have the following equation:

$$\min_{\boldsymbol{y}_i}\|\boldsymbol{y}^i-\boldsymbol{q}^i\|_2^2, s.t.\,\boldsymbol{y}^i\geqslant 0,\boldsymbol{y}^i\mathbf{1}_c=1 \tag{12}$$

which specifies the definition form of the Euclidean distance on the simplex constraint [25], In fact, there is a standard solution method for this problem. In summary, the optimal solution is obtained based on the Lagrange multipliers method combined with the Karush-Kuhn-Tucker (KKT) conditions according to the constraints. The detailed optimization method for Equation (12) is provided as follows.

To simplify the notations, we replace the transpose of $\boldsymbol{y}^i$ and $\boldsymbol{q}^i$ with $\boldsymbol{y}_i$ and $\boldsymbol{q}_i$. Then, the corresponding Lagrangian function of Equation (12) is,

$$\mathcal{L}(\boldsymbol{y}_i,\eta,\boldsymbol{\beta})=\|\boldsymbol{y}_i-\boldsymbol{q}_i\|_2^2-\eta(\boldsymbol{y}_i^T1_c-1)-\boldsymbol{\beta}^T\boldsymbol{y}_i \tag{13}$$

where, η and $\boldsymbol{\beta}\in\mathbb{R}^c$ are Lagrange multipliers in scalar and vector forms, respectively. Next, we provide a method to determine the two Lagrangian multipliers. Assume that the optimal solution to problem Equation (12) is y_i^*, and the corresponding Lagrange multiplier is η^* and $\boldsymbol{\beta}^*$. Then, by using KKT criteria, we have the following equations and inequalities,

$$\forall j, y_{ij}^*-q_{ij}-\eta^*-\beta_j^*=0, \tag{14}$$

$$\forall j, y_{ij}^*\geqslant 0, \tag{15}$$

$$\forall j, \beta_j^*\geqslant 0, \tag{16}$$

$$\forall j, y_{ij}^*\,\beta_j^*=0, \tag{17}$$

where y_{ij}^* is the j-th scalar element of vector y_i^*.

a) Solving η^*. Equation (14) can be equivalent to the following vector form as,

$$\boldsymbol{y}_i^*-\boldsymbol{q}_i-\eta^*\mathbf{1}_c-\boldsymbol{\beta}^*=\mathbf{0}. \tag{18}$$

Taking into account the constraints $\boldsymbol{y}_i^*\boldsymbol{I}c=1$, Equation (18) can be rewritten as the following equation,

$$\eta^*=\frac{1-\mathbf{1}_c^T\boldsymbol{q}_i-\mathbf{1}_c^T\boldsymbol{\beta}^*}{c}. \tag{19}$$

b) Solving $\boldsymbol{y}_i^*$. Putting Equation (19) into Equation (18), we have the following equation,

$$\boldsymbol{y}_i^*=\boldsymbol{q}_i-\frac{\mathbf{1}_c\mathbf{1}_c^T}{c}\boldsymbol{q}_i+\frac{1}{c}\mathbf{1}_c-\frac{1_c^T\boldsymbol{\beta}^*}{c}\mathbf{1}_c+\boldsymbol{\beta}^*. \tag{20}$$

Denote $\bar{\beta}^*=(\mathbf{1}_c^T\boldsymbol{\beta}^*/c)$ and $\boldsymbol{g}=\boldsymbol{q}_i-(\mathbf{1}_c\mathbf{1}_c^T/c)\boldsymbol{q}_i+(1/c)\mathbf{1}_c$. Equation (20) can be rewritten as the following

$$\boldsymbol{y}_i^*=\mathbf{g}+\boldsymbol{\beta}^*-\bar{\beta}^*\mathbf{1}_c. \tag{21}$$

Therefore, considering the situation of each element in Equation (21), for each j=1,⋯, c, we have

$$y_{ij}^*=\mathrm{g}_j+\beta_j^*-\bar{\beta}^*. \tag{22}$$

According to Equations (15), (16), (17) and (22), we have $\mathrm{g}_j+\beta_j^*-\bar{\beta}^*=\left(\mathrm{g}_j-\bar{\beta}^*\right)_+$, where ($f(\cdot)$)+=max($f(\cdot)$), 0). Therefore, we can get following formula

$$y_{ij}^{*} = \left(g_j - \bar{\beta}^{*}\right)_{+}. \tag{23}$$

Now, If we can determine the optimal $\bar{\beta}^{*}$, the optimal solution $\boldsymbol{y}_i^{*}$ can be obtained simply from Equation (23).

c) Solving $\boldsymbol{\beta}^{*}$. Equation (22) can be rewritten as $\beta_j^{*} = y_{ij}^{*} + \bar{\beta}^{*} - \mathrm{g}_j$ such that $\beta_j^{*} = (\bar{\beta}^{*} - \mathrm{g}_j)_{+}$. With the above analysis, we can get the calculation formula of $\bar{\beta}^{*}$ as,

$$\bar{\beta}^{*} = \frac{1}{c}\sum_{j=1}^{c}(\bar{\beta}^{*} - \mathrm{g}_j)_{+}. \tag{24}$$

Taking into account constraints $\boldsymbol{y}_i^{T}\mathbf{1}_c = 1$ and Equation (23), we define the following function

$$f(\bar{\beta}) = \sum_{j=1}^{c}(\mathrm{g}_j - \bar{\beta})_{+} - 1. \tag{25}$$

The optimal $\bar{\beta}^{*}$ obtained should satisfy $f(\bar{\beta}^{*})=0$. When Equation (25) equals zero, we can used Newton method to obtain the optimal solution as

$$\bar{\beta}^{(m+1)} = \bar{\beta}^{(m)} - \frac{f(\bar{\beta}^{(m)})}{f'(\bar{\beta}^{(m)})}. \tag{26}$$

We know that $f(\bar{\beta})$ is a monotonically increasing piecewise function. When $\mathrm{g}_j \leqslant \bar{\beta}$, $f(\bar{\beta}) = \sum_{j=1}^{c}\mathrm{g}_j - \bar{\beta} - 1.$, and we have $f'(\bar{\beta}) = -1$. When $\mathrm{g}_j \leqslant \bar{\beta}$, $f(\bar{\beta}) = -1$ and its derivative $f'(\bar{\beta}) = 0$. By enumerating the number of positive values in $(\mathrm{g}_j - \bar{\beta})|_{j=1}^{c}$, we can get $f'(\bar{\beta})$.

In summary, Algorithm (1) gives the optimization process of Equation (12).

Algorithm 1: Procedure to Solve Function of Equation (12).

Input: vector $\boldsymbol{q}_i \in \mathbb{R}^{c}$;
Output: vector $\boldsymbol{y}_i \in \mathbb{R}^{c}$.
1: obtaining the number of sample classes c;// The value of c can be determined by calculating the maximal value among the elements in $\boldsymbol{Y}_l$.
2: Computer $\mathbf{g} = \boldsymbol{q}_i - \frac{\mathbf{1}_c\mathbf{1}_c^{T}}{c}\boldsymbol{q}_i + \frac{1}{c}\mathbf{1}_c$;
3: Obtain the root $\bar{\beta}^{*}$ *of Equation (25) by using Newton method;*
// Specifically, $\bar{\beta}^{(m+1)} = \bar{\beta}^{(m)} - \frac{f(\bar{\beta}^{(m)})}{f'(\bar{\beta}^{(m)})}$
4: Obtain the optimal solution $y_{ij}^{*} = (\mathrm{g}_j - \bar{\beta}^{*})$ *+ for j=1,⋯, c;*

We summarize the entire optimization solution steps of the objective function of Equation (10) in Algorithm (2).

Algorithm 2: Optimization Procedure to KLRRR.

Input: data matrix $\boldsymbol{X} \in \mathbb{R}^{d\times n}$, *label matrix* $\boldsymbol{Y}_l \in \mathbb{R}^{l\times c}$, *low-rank parameter s, regularization parameter* λ, *iter=50;*
Output: projection matrices $\boldsymbol{A} \in \mathbb{R}^{d\times s}$ *and* $\boldsymbol{B} \in \mathbb{R}^{s\times c}$, *and label matrix* $\boldsymbol{Y}_u \in \mathbb{R}^{u\times c}$.
1: Initialize r=0, $\boldsymbol{Y}_u^{(r)}$;
2: Calculation of the Gaussian kernel matrix $\boldsymbol{K}$;
3: while not objective function converged and r< iter do
4: Calculate $\boldsymbol{B}^{(r+1)}$ *by Equation* (7);
5: Calculate $\boldsymbol{S}^{(r+1)} = (\boldsymbol{S}_t^{r})^{-1}\boldsymbol{S}_b^{(r)}$
6: Perform eigenvalue decomposition of matrix $\boldsymbol{S}^{(r+1)}$
7: Calculate $\boldsymbol{A}^{(r+1)}$ *by Equation* (8); *// $\boldsymbol{A}^{(r+1)}$=top s largest eigenvalues of* $\boldsymbol{S}^{(r+1)}$.
8: Calculate , $\boldsymbol{Y}_u^{(r+1)}$ *by (12);*
9: Update $\boldsymbol{S}_b$ *base on* $\boldsymbol{Y}^{(r+1)}$;
10: r=r+1 ;
11: end while

3.3. Model Property Analysis

Algorithm convergence analysis. We will prove the convergence of our proposed Algorithm (2).

- Proof: In the *t*-th iteration, we have the following equation

$$\begin{aligned} < \boldsymbol{A}^{(t+1)}, \boldsymbol{B}^{(t+1)}, \boldsymbol{Y}^{(t+1)} >_u \geq \operatorname{argmin} & \left\| \boldsymbol{Y}^{(t)} - \boldsymbol{K}\boldsymbol{A}^{(t)}\boldsymbol{B}^{(t)} \right\|_F^2 \\ & + \lambda \mathrm{Tr}\left(\boldsymbol{B}^{(t)T} - \boldsymbol{A}^{(t)T}\boldsymbol{K}\boldsymbol{A}^{(t)}\boldsymbol{B}^{(t)}\right). \end{aligned} \tag{27}$$

Bringing $\boldsymbol{A}^{(t+1)}$, $\boldsymbol{B}^{(t+1)}$, $\boldsymbol{Y}_u^{(t+1)}$ obtained by *t*+1 iterations into the Equation (5), we can get the following inequality

$$\begin{aligned} & \left\| \boldsymbol{Y}^{(t+1)} - \boldsymbol{K}\boldsymbol{A}^{(t+1)}\boldsymbol{B}^{(t+1)} \right\|_F^2 \\ & + \lambda \mathrm{Tr}\left(\boldsymbol{B}^{(t+1)T}\boldsymbol{A}^{(t+1)}\boldsymbol{K}\boldsymbol{A}^{(t+1)}\boldsymbol{B}^{(t)+1}\right) \\ & \leqslant \\ & \left\| \boldsymbol{Y}^{(t)} - \boldsymbol{K}\boldsymbol{A}^{(t)}\boldsymbol{B}^{(t)} \right\|_F^2 \\ & + \lambda \mathrm{Tr}\left(\boldsymbol{B}^{(t)T} - \boldsymbol{A}^{(t)T}\boldsymbol{K}\boldsymbol{A}^{(t)}\boldsymbol{B}^{(t)}\right) \end{aligned} \tag{28}$$

Specifically, variables $\boldsymbol{A}$ and $\boldsymbol{B}$ use gradient method to update, and variable $\boldsymbol{Y}_u$ is updated by using the Lagrange multiplier method with analytical multipliers. It shows that the updating of $\boldsymbol{Y}_u$ is also based on the decreasing of loss function decreasing. From this we draw the conclusion that Algorithm (2) will monotonically reduce the loss function values.

- **Computational Complexity Analysis**.

We analyze the computational complexity of Algorithm (2) using big $\mathcal{O}$ notation. Clearly, the most computationally expensive step is the update of the variable $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{Y}_u$. The complexity of updating variable $\boldsymbol{A}$ is $\mathcal{O}(n^3)$, the complexity of updating variable $\boldsymbol{B}$ is $\mathcal{O}(cn^2+c^2n+c^3)$), the complexity of updating variable $\boldsymbol{Y}_u$ is $\mathcal{O}(u(c^2+c))$. Overall, the time complexity of Algorithm (2) is $\mathcal{O}(r(n^3+n^2c+nc^2+c^3))$, where r is the number of iterations. In our KLRRR model, the solution for $\boldsymbol{Y}_u$ is performed row-by-row, with each row's solution being independent of the others. Therefore, parallel computing can be employed to solve for $\boldsymbol{Y}_u$.

4. Experiments

In this section, we compare the proposed KLRRR model and its out-of-sample prediction version performance with the semi-supervised Kernel space Full-Rank Ridge Regression (KFRRR) model base on Equation (4) where $\boldsymbol{AB}$ is replaced by $\boldsymbol{C}$, Semi-supervised Low-Rank Ridge Regression (SLRRR) base on [2] and all of their out-of-sample prediction version across 12 standard data sets. Besides, we compare our model with a popular semi-supervised model Rescaled Linear Square Regression (RLSR) [3] and its out-of-sample prediction version in

classification accuracy and standard deviation. We also evaluate the convergence, parameter sensitivity and running time of our model on the benchmark data sets under different kernel functions. Regarding the choice of kernel function in our experiment, we take the Gaussian kernel function as an example. Of course, our model is not limited to a specific kernel function, the effectiveness of our model with other kernel functions will be further demonstrated in the subsequent discussion section. The form of Gaussian kernel function [35] is as follows

$$\boldsymbol{K}_{ij} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{\sigma^2}\right). \tag{29}$$

Among them, σ is the bandwidth parameter of Gaussian distribution.

4.1. Data Set Descriptions

The 12 benchmark data sets from real-world were used in the experiments. Ecoli is a biological data set used for predicting the location of protein sites. Glass is a data set defined by the content of oxides, containing six categories. Jaffe is a face data set that includes six categories. Binalpha36 consists of 20×16 binary digits ranging from '0' to '9' and uppercase letters 'A' to 'Z', with 39 examples for each category. Vehicle is a data set that describes the general overview of vehicles, with four categories. Umist face data set contains 20 individuals with a total of 575 images. COIL20 is an object data set. Yale is a data set containing 165 grayscale GIF images from 15 individuals. ORL_32x32 is an image data set captured under various times, lighting conditions, and facial details. Auto is a data set that includes the feature specifications of cars, specified insurance risk levels, and standardized loss in use. The "Control" data set is composed of synthetically generated control charts from six categories. Dermatology is a data set of clinical pathological used to determine the types of psoriasis. We present the main characteristics of each data in Table 1.

Table 1. Main characteristics of the 12 benchmark data sets.

Data sets	# Sample	# Dimensions	# Class
ecoli	366	343	8
glass	241	9	6
jaffe	212	177	7
binalpha36	1404	320	36
vehicle	846	18	4
umist	575	644	20
COIL20	1440	1024	20
Yale	165	1024	15
ORL_32x32	400	1024	40
auto	205	25	6
control	600	60	6
dermatology	366	34	6

4.2. Experimental Settings

For each of the 12 benchmark data sets, we randomly divide the samples into 5 parts, each part has a similar amount of data. Using five-fold cross validation, in each round, we use four parts as labeled data and the other parts as unlabeled data. In particular, in the out-of-sample prediction evaluation, the unlabeled data are further randomly divided into two parts of equal proportion, one part of them is used as the test set. The average classification accuracy and the standard deviation of five rounds is reported for different methods. In all experiments, We adaptively adjust the value of the regularization parameter λ within the range of $\{10^r: r\in\{-8, -7, \cdots, 8\}\}$. In addition, for KLRRR and the out of sample version (KLRRR-OS) of KLRRR, the different low-rank parameter s is in range of [1, c) and the parameters σ is in rank of $\{2^r: r\in\{-10, -9, \cdots, 10\}\}$.For Semi-supervised Low Rank Linear Regression and the Out of Sample version (SLRRR-OS) of SLRRR, the adjustment range for parameters s is the same as KLRRR. For Kernel Full Rank Ridge Regression and the Out of Sample version (KFRRR-OS) of it, the adjustment range for parameters σ is the same as KLRRR. The maximal number of iterations for each variable update is set to 50. When the difference between the loss functions of two adjacent iterations is less than 10^{-5}, the iteration is terminated.

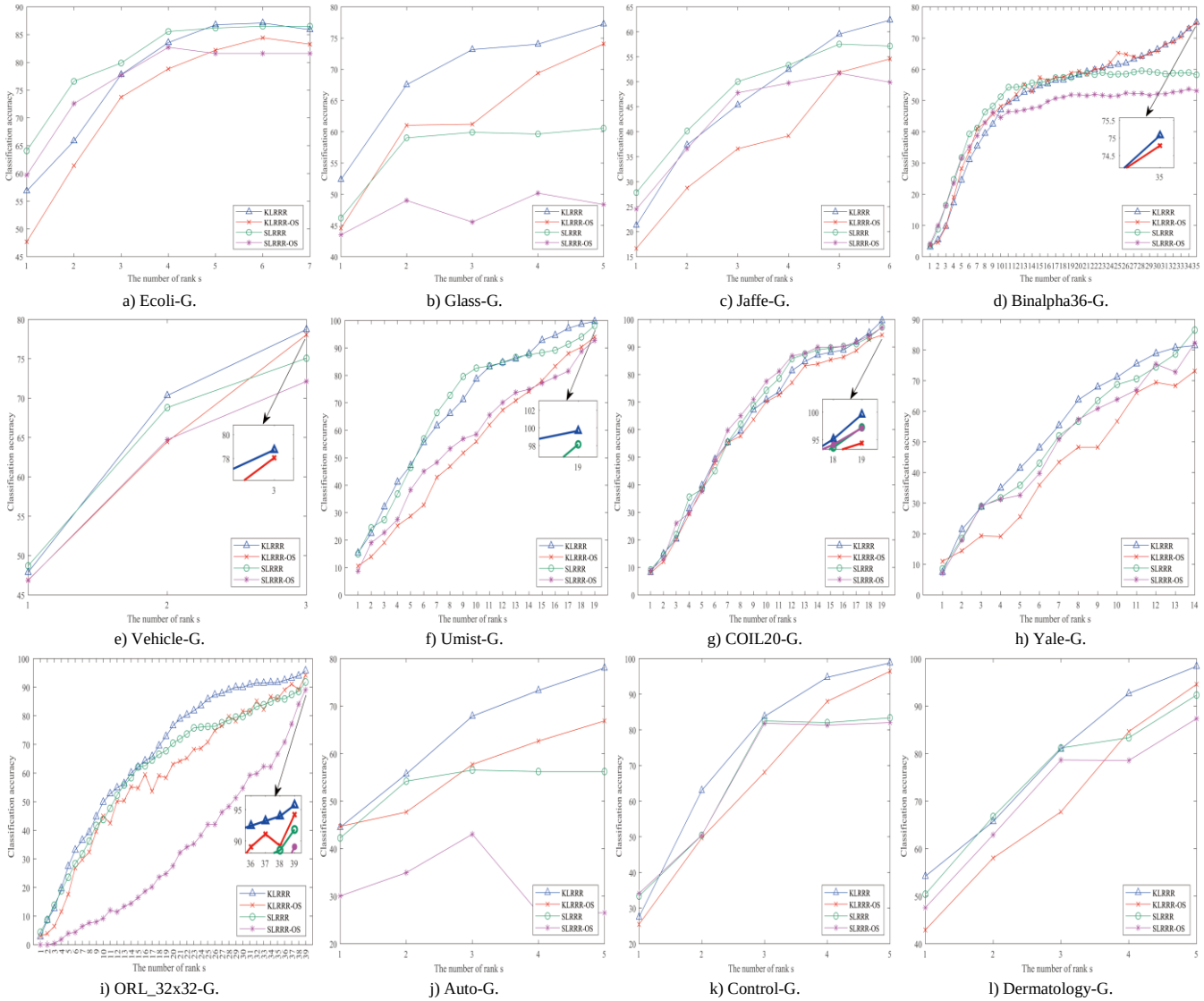
4.3. Recognition Results and Analysis

The classification accuracies and the standard deviations of the five-fold cross-validation of our proposed model KLRRR, the comparison methods LRRR, KFRRR, RLSR and their out-of-sample prediction versions are shown in Table 2, where each row represents a certain model, and each column represents a benchmark data set. The maximal classification accuracy in each case is highlighted by bold, and the second largest classification accuracy is highlighted in underline. From the table we see that our proposed KLRRR achieves the maximal classification accuracy and demonstrated good robustness in 11 out of the 12 benchmark data sets. In terms of classification accuracy and robustness, the effectiveness of kernel trick and low-rank methods is demonstrated. In addition, the results in Table 2 also show that our proposed method is superior to the existing and popular semi-supervised classification method RLSR in terms of classification accuracy. Figure 1 shows the comparison of the average classification accuracies of kernel model and non-kernel version on the 12 benchmark data sets under different low- rank constraints and low-rank conditions, namely KLRRR, SLRRR and their out-of-sample prediction versions. It is obvious from the figure that when the low-rank parameter s falls within an appropriate range, the classification accuracy of the kernel model is higher, which demonstrates the effectiveness and advantages of the kernel trick. Furthermore, our proposed method always has the highest classification accuracy on 11 out of the 12 benchmark data sets. Figure 2 shows the comparison of classification accuracy between low-rank

methods and full-rank methods in kernel space on the 12 benchmark data sets, namely KLRRR, KFRRR and their out-of-sample prediction versions. It can also be concluded from Figure 2 that when the low-rank parameter s falls within an appropriate range, the classification accuracy under the low-rank method is higher, which demonstrates the effectiveness and advantages of low-rank. Overall, our proposed method demonstrates excellent classification accuracy and robustness on the 12 benchmark data sets.

Table 2. Average classification accuracies (%) and variance of five-fold cross validation on Gaussian kernel.

	ecoli	**glass**	**jaffe**	**binalpha36**	**vehicle**	**umist**
SLRRR-OS	83.86±8.20	56.54±12.27	55.36±11.23	55.97±8.26	72.13±8.30	92.75±4.10
SLRRR	86.81±5.09	61.94±11.63	59.48±3.51	60.40±3.57	75.06±2.79	98.13±1.23
KFRRR-OS	80.29±8.54	68.61±10.35	54.31±7.72	71.93±3.26	74.29±5.88	92.11±4.51
KFRRR	85.24±3.74	76.02±14.15	61.44±8.24	73.08±1.92	75.41±2.39	98.81±1.74
RLSR-OS	63.66±8.66	56.49±10.29	51.87±11.49	57.41±7.24	72.24±3.98	94.16±4.05
RLSR	82.86±5.44	63.27±14.89	60.91±5.80	60.18±4.65	77.41±2.98	98.47±1.12
KLRRR-OS	85.02±6.95	74.98±10.62	55.50±9.93	74.78±2.93	78.06±6.14	93.97±1.53
KLRRR	**87.38±3.27**	**77.78±6.00**	**62.35±6.25**	**75.07±1.55**	**78.72±6.28**	**99.66±0.75**
	COIL20	**Yale**	**ORL_32x32**	**auto**	**control**	**dermatology**
SLRRR-OS	97.10±1.02	82.28±6.33	89.07±10.33	44.00±12.52	82.59±7.48	87.37±10.91
SLRRR	97.23±1.06	**86.53±3.68**	92.07±8.03	58.53±8.15	83.37±5.20	92.32±6.03
KFRRR-OS	98.75±1.25	72.39±11.91	91.18±6.00	73.49±9.62	93.33±3.56	97.34±2.04
KFRRR	98.89±0.76	78.92±6.27	92.97±2.57	73.86±6.07	94.43±1.72	97.52±2.43
RLSR-OS	95.58±1.42	70.16±14.39	85.43±5.02	44.80±14.23	82.68±8.15	86.54±7.04
RLSR	96.94±0.39	86.24±5.52	94.53±1.74	59.55±8.03	82.71±5.59	90.96±8.24
KLRRR-OS	94.63±1.53	74.47±8.74	94.20±3.52	73.13±12.78	96.72±2.47	94.58±8.38
KLRRR	**99.58±0.76**	82.01±6.81	**95.72±2.42**	**78.02±7.22**	**98.80±1.05**	**98.36±2.25**

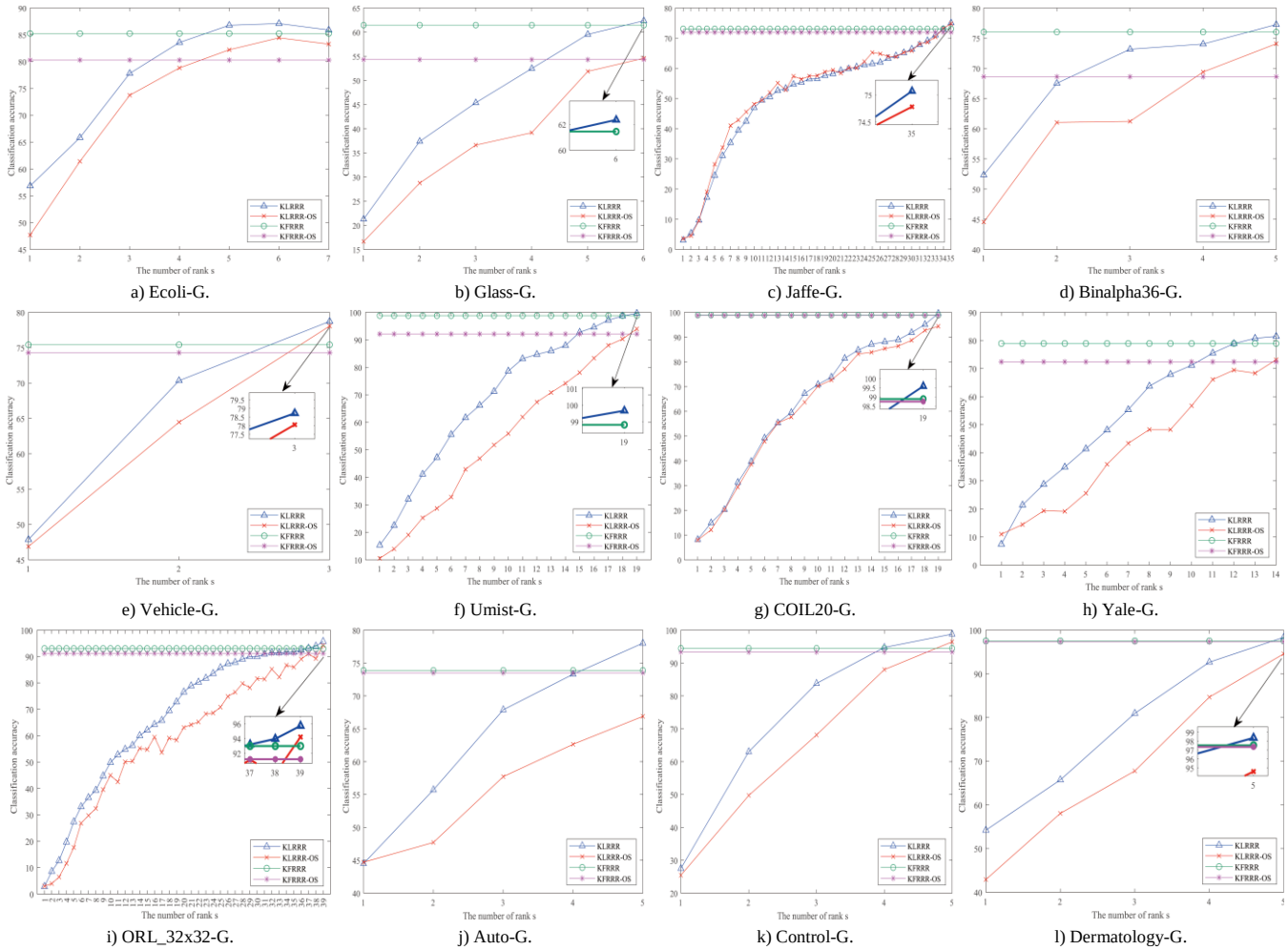


a) Ecoli-G. b) Glass-G. c) Jaffe-G. d) Binalpha36-G.

e) Vehicle-G. f) Umist-G. g) COIL20-G. h) Yale-G.

i) ORL_32x32-G. j) Auto-G. k) Control-G. l) Dermatology-G.

Figure 1. The comparison of the average classification accuracies (%) of kernel space and non-kernel space on 12 benchmark data sets under different low-rank conditions on Gaussian kernel.



a) Ecoli-G. b) Glass-G. c) Jaffe-G. d) Binalpha36-G.

e) Vehicle-G. f) Umist-G. g) COIL20-G. h) Yale-G.

i) ORL_32x32-G. j) Auto-G. k) Control-G. l) Dermatology-G.

Figure 2. The comparison of classification accuracies (%) between low-rank methods and full-rank methods in kernel space on 12 benchmark data sets on Gaussian kernel.

4.4. Discussion

In this section, we first discuss the flexibility and versatility of the KLRRR model under different kernel functions. Then, we experimentally show the convergence properties of the model. Finally, we present the model's performance sensitivity to the regularization parameter and the hyperparameters within the kernel functions.

a) Model Flexibility and Versatility. To demonstrate the model flexibility and versatility in different scenarios, we also conducted experiments using other kernel functions, i.e., the linear kernel and the polynomial kernel. The mathematical expression of the linear kernel is [5]:

$$\boldsymbol{K}_{ij} = \boldsymbol{x}_i^T \boldsymbol{x}_j \tag{30}$$

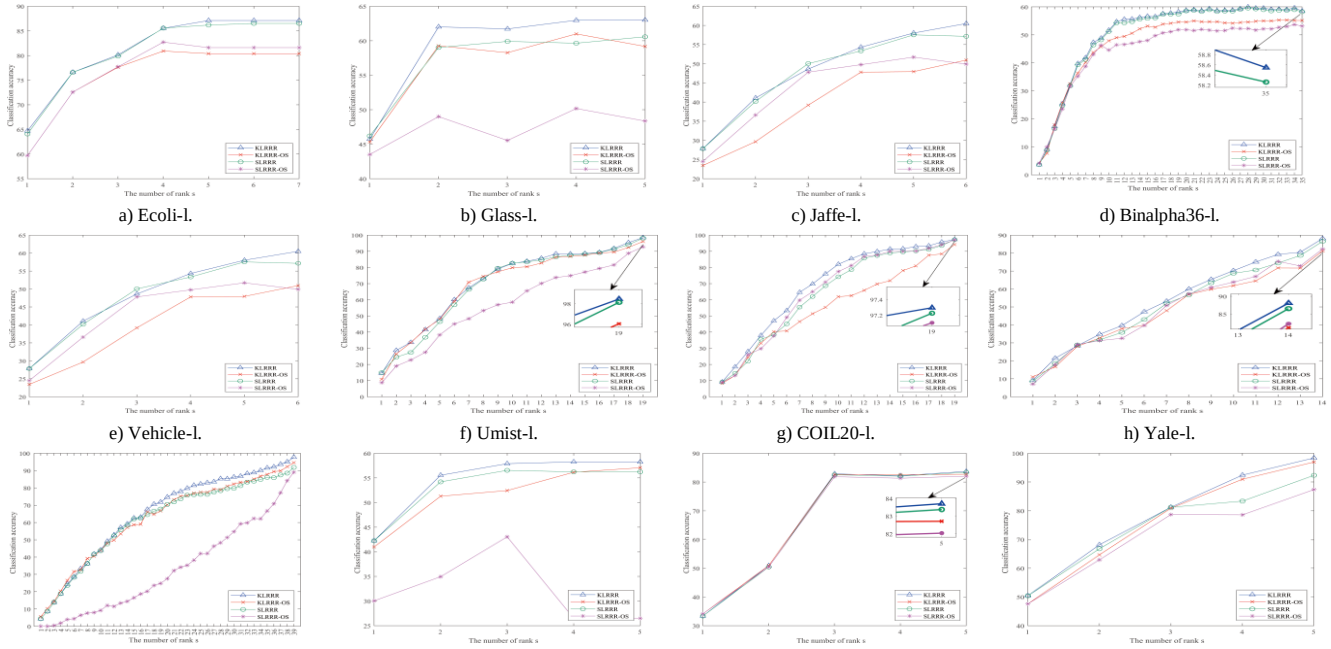
The expression for the polynomial kernel is,

$$\boldsymbol{K}_{ij} = (a\boldsymbol{x}_i\, \boldsymbol{x}_j^T + c)^d \tag{31}$$

In Equation (31), a, c and d are the hyperparameters of the polynomial kernel. Inspired by existing studies, we set a and c to 1 [20, 26], respectively. The optimal value for d is searched within the range $[1, 2, \cdots, 5]$. In the case of using linear kernel, the average classification accuracy and standard deviation of the 12 data sets across the 8 methods are presented in Table 3. From which we observe, our method achieved the highest classification accuracy on 11 out of the 12 data sets. Besides, our model also demonstrated lower standard deviations, indicating its stability and robustness. Figure 3 shows the comparison of average classification accuracy between kernel and non-kernel models under different low-rank constraints and conditions across the 12 benchmark data sets. Figure 4 provides us with the comparison of classification accuracy between low-rank methods and full-rank methods in kernel space. When using the polynomial kernel, the average classification accuracies and standard deviations of the 12 data sets across the 8 methods are presented in Table 4, where our method achieved the highest classification accuracy on 8 out of the 12 data sets. Our method also has a smaller standard deviation value. Figure 5 shows the comparison of average classification accuracy between kernel and non-kernel models under different low-rank constraints and conditions. Figure 6 corresponds to the comparison of classification accuracy between low-rank methods and full-rank methods in kernel space.

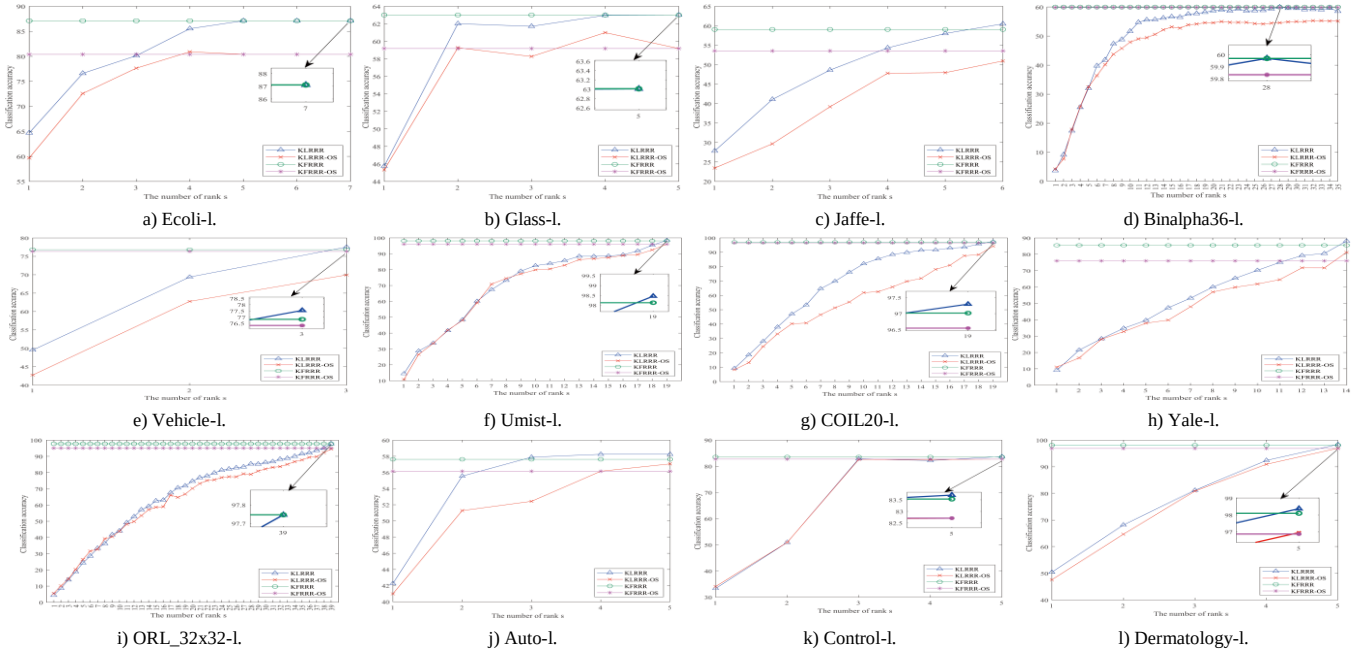
Table 3. Average classification accuracies (%) and variance of five-fold cross validation on linear kernel.

	ecoli	glass	jaffe	binalpha36	vehicle	umist
SLRRR-OS	83.86±8.20	56.54±12.27	55.36±11.23	55.97±8.26	72.13±8.30	92.75±4.10
SLRRR	86.81±5.09	61.94±11.63	59.48±3.51	60.40±3.57	75.06±2.79	98.13±1.23
KFRRR-OS	80.39±9.23	59.18±17.41	53.54±11.21	59.83±5.68	76.33±4.15	96.04±1.58
KFRRR	87.11±4.89	63.01±10.32	59.05±6.07	59.96±4.62	76.82±2.99	98.13±1.10
RLSR-OS	63.66±8.66	56.49±10.29	51.87±11.49	57.41±7.24	72.24±3.98	94.16±4.05
RLSR	82.86±5.44	63.27±14.89	60.91±5.80	60.18±4.65	77.41±2.98	**98.47±1.12**
KLRRR-OS	82.06±10.72	62.90±18.60	52.78±12.36	56.84±6.68	70.67±7.30	96.40±1.87
KLRRR	**87.40±4.59**	**66.46±13.11**	**62.35±5.56**	**60.61±3.63**	**77.54±2.80**	98.46±0.95
	COIL20	**Yale**	**ORL_32x32**	**auto**	**control**	**dermatology**
SLRRR-OS	97.10±1.02	82.28±6.33	89.07±10.33	44.00±12.52	82.59±7.48	87.37±10.91
SLRRR	97.23±1.06	86.53±3.68	92.07±8.03	58.53±8.15	83.37±5.20	92.32±6.03
KFRRR-OS	96.55±0.97	75.97±13.34	95.09±2.47	56.15±8.64	82.71±7.33	96.86±2.25
KFRRR	97.02±0.99	85.51±5.34	97.74±1.02	57.63±7.28	83.53±5.12	98.10±2.28
RLSR-OS	95.58±1.42	70.16±14.39	85.43±5.02	44.80±14.23	82.68±8.15	86.54±7.04
RLSR	96.94±0.39	86.24±5.52	94.53±1.74	59.55±8.03	82.71±5.59	90.96±8.24
KLRRR-OS	96.00±0.97	82.53±8.09	94.58±4.11	59.73±11.62	83.27±7.26	96.94±2.65
KLRRR	**97.43±0.90**	**88.08±5.96**	**97.98±0.73**	**60.50±9.05**	**83.70±5.05**	**98.37±1.77**



a) Ecoli-l. b) Glass-l. c) Jaffe-l. d) Binalpha36-l.
e) Vehicle-l. f) Umist-l. g) COIL20-l. h) Yale-l.
i) ORL_32x32-l. j) Auto-l. k) Control-l. l) Dermatology-l.

Figure 3. The comparison of the average classification accuracies (%) of kernel space and non-kernel space on 12 benchmark data sets under different low-rank conditions on linear kernel.

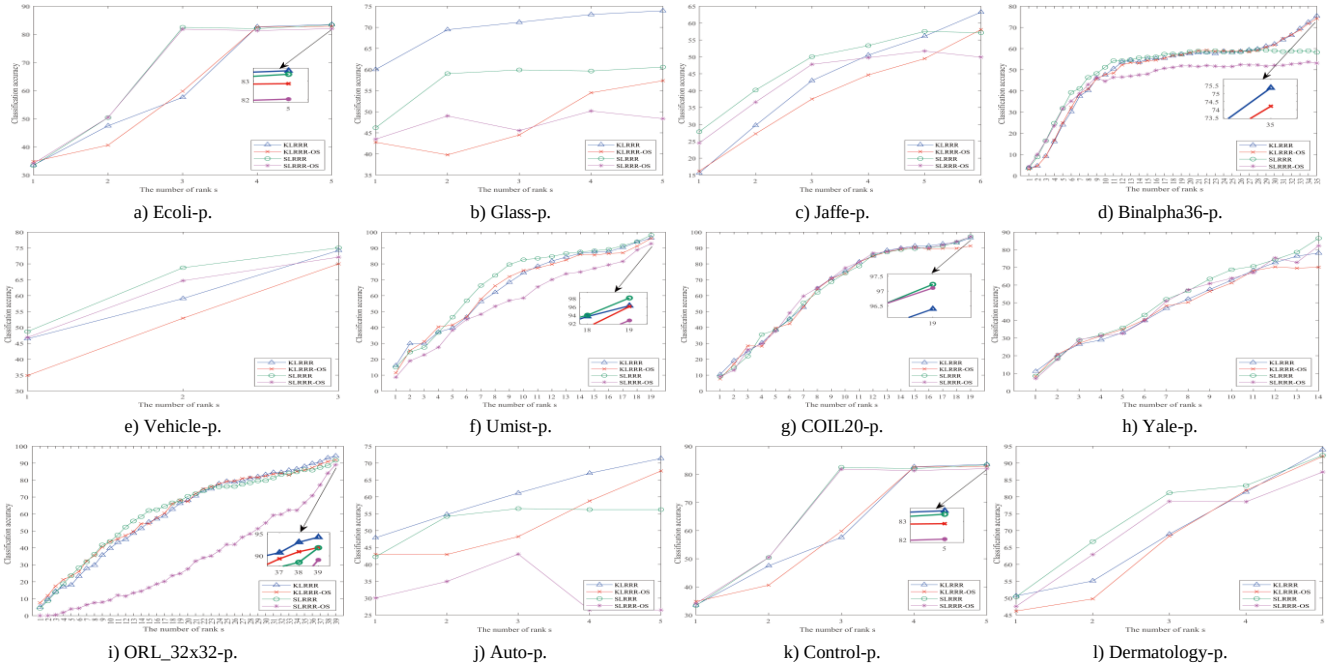


a) Ecoli-l. b) Glass-l. c) Jaffe-l. d) Binalpha36-l.
e) Vehicle-l. f) Umist-l. g) COIL20-l. h) Yale-l.
i) ORL_32x32-l. j) Auto-l. k) Control-l. l) Dermatology-l.

Figure 4. The comparison of classification accuracies (%) between low-rank methods and full-rank methods in kernel space on 12 benchmark data sets on linear kernel.

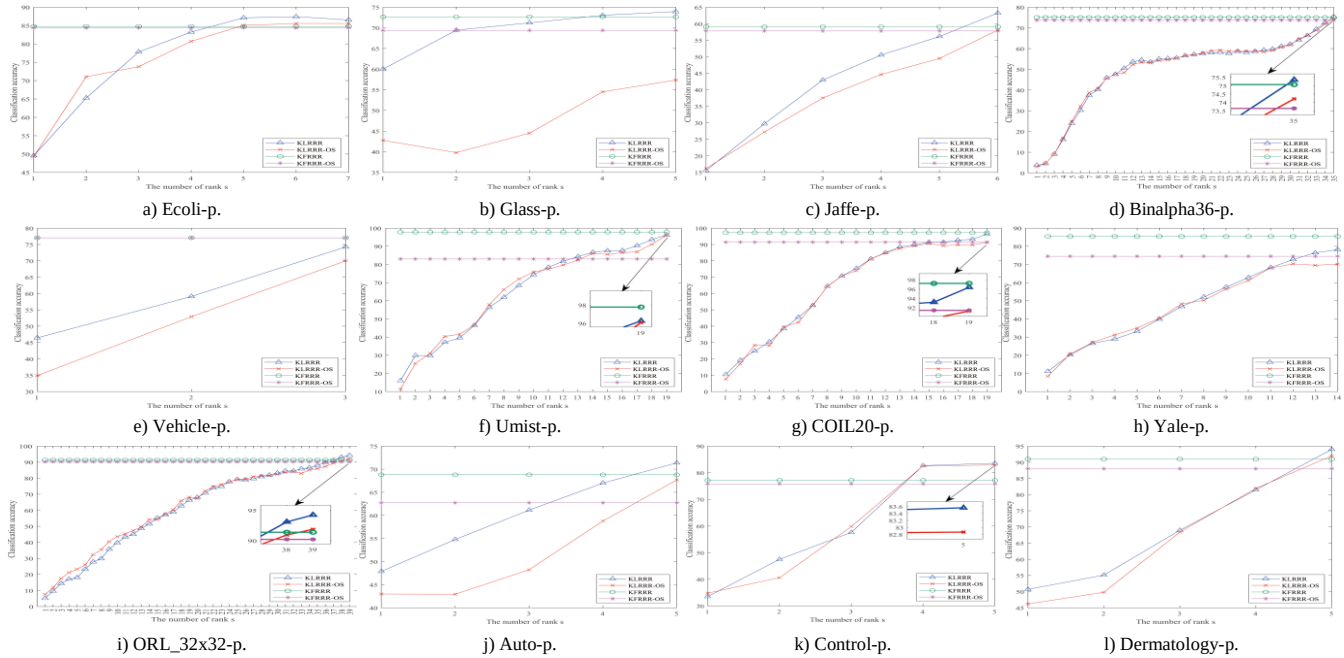
Table 4. Average classification accuracies (%) and variance of five-fold cross validation on polynomial kernel.

	ecoli	**glass**	**jaffe**	**binalpha36**	**vehicle**	**umist**
SLRRR-OS	83.86±8.20	56.54±12.27	55.36±11.23	55.97±8.26	72.13±8.30	92.75±4.10
SLRRR	86.81±5.09	61.94±11.63	59.48±3.51	60.40±3.57	75.06±2.79	98.13±1.23
KFRRR-OS	84.42±5.86	69.39±8.70	57.80±10.07	73.65±3.71	77.05±4.66	83.01±2.27
KFRRR	84.76±3.57	72.66±10.51	59.05±6.11	75.07±1.23	77.06±2.82	97.78±1.43
RLSR-OS	63.66±8.66	56.49±10.29	51.87±11.49	57.41±7.24	72.24±3.98	94.16±4.05
RLSR	82.86±5.44	63.27±14.89	60.91±5.80	60.18±4.65	**77.41±2.98**	**98.47±1.12**
KLRRR-OS	86.15±5.45	60.21±14.05	58.09±3.63	74.21±2.93	70.01±7.73	96.08±2.88
KLRRR	**87.99±3.17**	**73.93±6.15**	**63.25±5.54**	**75.35±1.93**	74.24±5.39	96.27±2.71
	COIL20	**Yale**	**ORL_32x32**	**auto**	**control**	**dermatology**
SLRRR-OS	97.10±1.02	82.28±6.33	89.07±10.33	44.00±12.52	82.59±7.48	87.37±10.91
SLRRR	**97.23±1.06**	**86.53±3.68**	92.07±8.03	58.53±8.15	83.37±5.20	92.32±6.03
KFRRR-OS	91.45±1.43	74.53±13.02	90.20±4.04	62.72±12.52	75.82±5.71	88.01±6.81
KFRRR	**97.23±1.17**	85.47±4.16	91.38±3.91	68.80±6.40	77.17±7.34	90.98±8.48
RLSR-OS	95.58±1.42	70.16±14.39	85.43±5.02	44.80±14.23	82.68±8.15	86.54±7.04
RLSR	96.94±0.39	86.24±5.52	94.53±1.74	59.55±8.03	82.71±5.59	90.96±8.24
KLRRR-OS	92.27±2.46	73.81±9.28	92.76±4.10	68.59±5.25	83.56±7.11	91.99±5.69
KLRRR	96.39±2.39	78.73±6.08	**94.71±1.75**	**71.38±6.43**	**83.73±6.04**	**93.99±3.46**



a) Ecoli-p. b) Glass-p. c) Jaffe-p. d) Binalpha36-p.
e) Vehicle-p. f) Umist-p. g) COIL20-p. h) Yale-p.
i) ORL_32x32-p. j) Auto-p. k) Control-p. l) Dermatology-p.

Figure 5. The comparison of the average classification accuracies (%) of kernel space and non-kernel space on 12 benchmark data sets under different low-rank conditions on polynomial kernel.

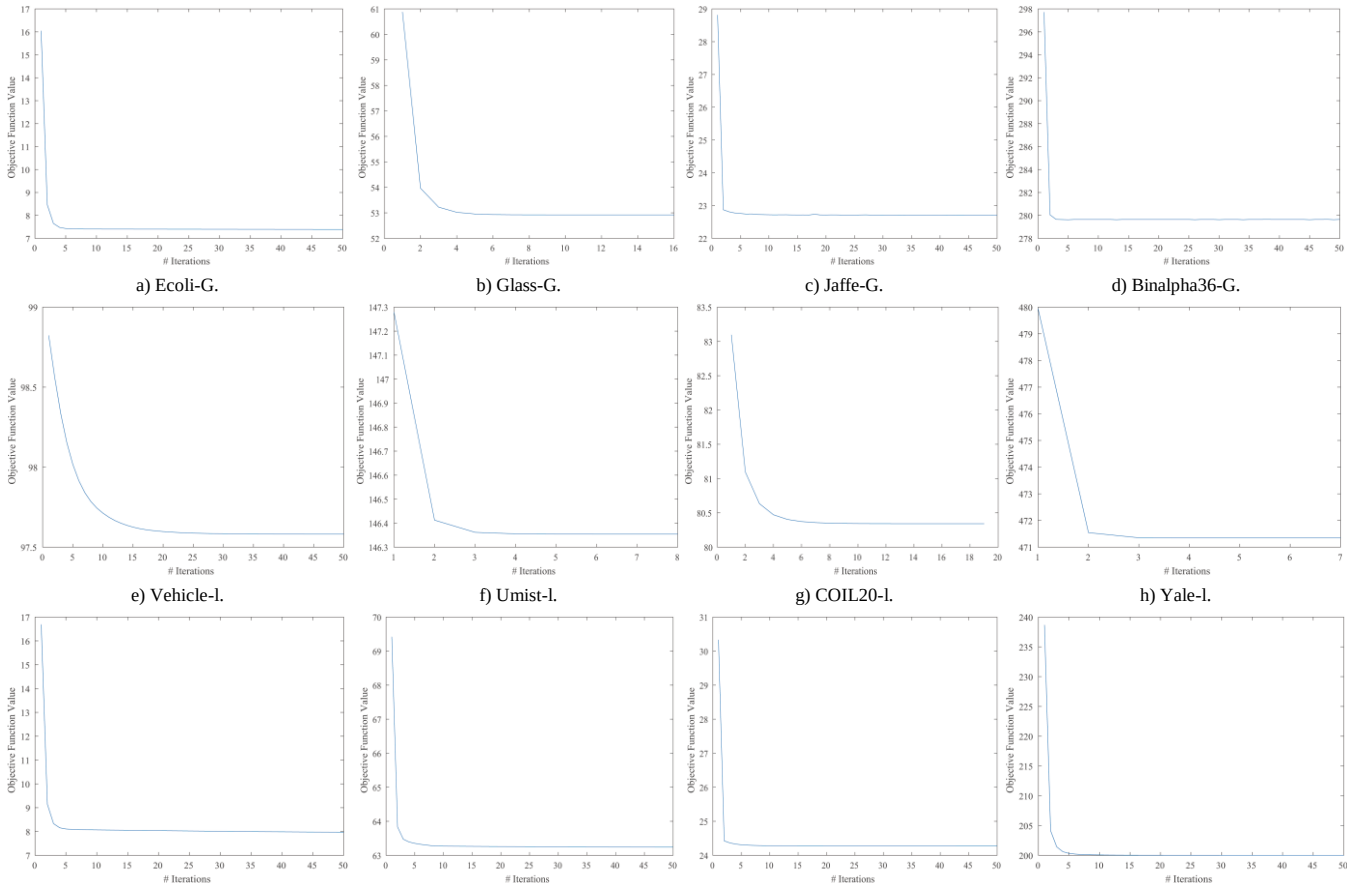


a) Ecoli-p. b) Glass-p. c) Jaffe-p. d) Binalpha36-p.
e) Vehicle-p. f) Umist-p. g) COIL20-p. h) Yale-p.
i) ORL_32x32-p. j) Auto-p. k) Control-p. l) Dermatology-p.

Figure 6. The comparison of classification accuracies (%) between low-rank methods and full-rank methods in kernel space on 12 benchmark data sets on polynomial kernel.

b) Convergence. In section 3, we have theoretically proved that the optimization process described in Algorithm (1) would converge. To more intuitively display the convergence performance of the loss function in the experiment, we present the decreasing of the KLRRR objective function values on some data sets, where the Gaussian kernel, linear kernel, and polynomial kernel are respectively corresponding to Figure 7-a), (d), (e), (h), (i), and (l). On each benchmark data set, the values of the loss function decrease as the iteration increases. From the figure, we draw the conclusion that Algorithm (2) definitely converge within 50 iterations on the 12 benchmark data sets. Obviously, our model converges within 15 iterations on most of the data sets. This also demonstrates the superiority of our algorithm.

c) Parameter Sensitivity. We also presented the parameter sensitivity for some data sets under different kernel functions. In the case of the Gaussian kernel, Figure 8-a) and (d) of shows how the classification accuracy of the model on the data sets varies in terms of the regularization parameter λ and the standard deviation σ of the Gaussian kernel function. In the case of the linear kernel, Figure 8-e) and (h) of shows how the classification accuracy of the model on the data sets varies in terms of the regularization parameter λ. In the case of the polynomial kernel, Figure 8-i) and (l) of shows how the classification accuracy of the model on the data sets varies in terms of the regularization parameter λ. Since KLRRR aims to achieve higher classification accuracy, we focused on the parameter range where the model achieved high accuracy to explore the model's sensitivity to parameters. It can be observed that, our model consistently achieves high classification accuracy within a large selectable space for parameters λ and σ. The model is not highly sensitive to the two parameters, demonstrating its robustness. When using Gaussian kernel function. Especially for the glass and umist data sets, the model performs well on these two data sets. When the highest accuracy is achieved, changes in the two parameters have almost no impact on the classification accuracy of the model. Similarly, when achieving higher accuracy, the regularization parameters and kernel hyperparameters in the case of linear kernels and polynomial kernels have little impact on the accuracy. Since the selection of regularization parameter and kernel hyperparameter is based on experience and trial, the smaller the impact of these two parameters on the accuracy, the better the performance of the model. Figure 8 shows model robustness our proposed model.



a) Ecoli-G. b) Glass-G. c) Jaffe-G. d) Binalpha36-G.

e) Vehicle-l. f) Umist-l. g) COIL20-l. h) Yale-l.

i) ORL_32x32-p. j) Auto-p. k) Control-p. l) Dermatology-p.

Figure 7. The convergence performance of the loss function on 4 benchmark data sets. (a)-(d) correspond to the Gaussian kernel, (e)-(h) correspond to the linear kernel, (i)-(l) correspond to the polynomial kernel.

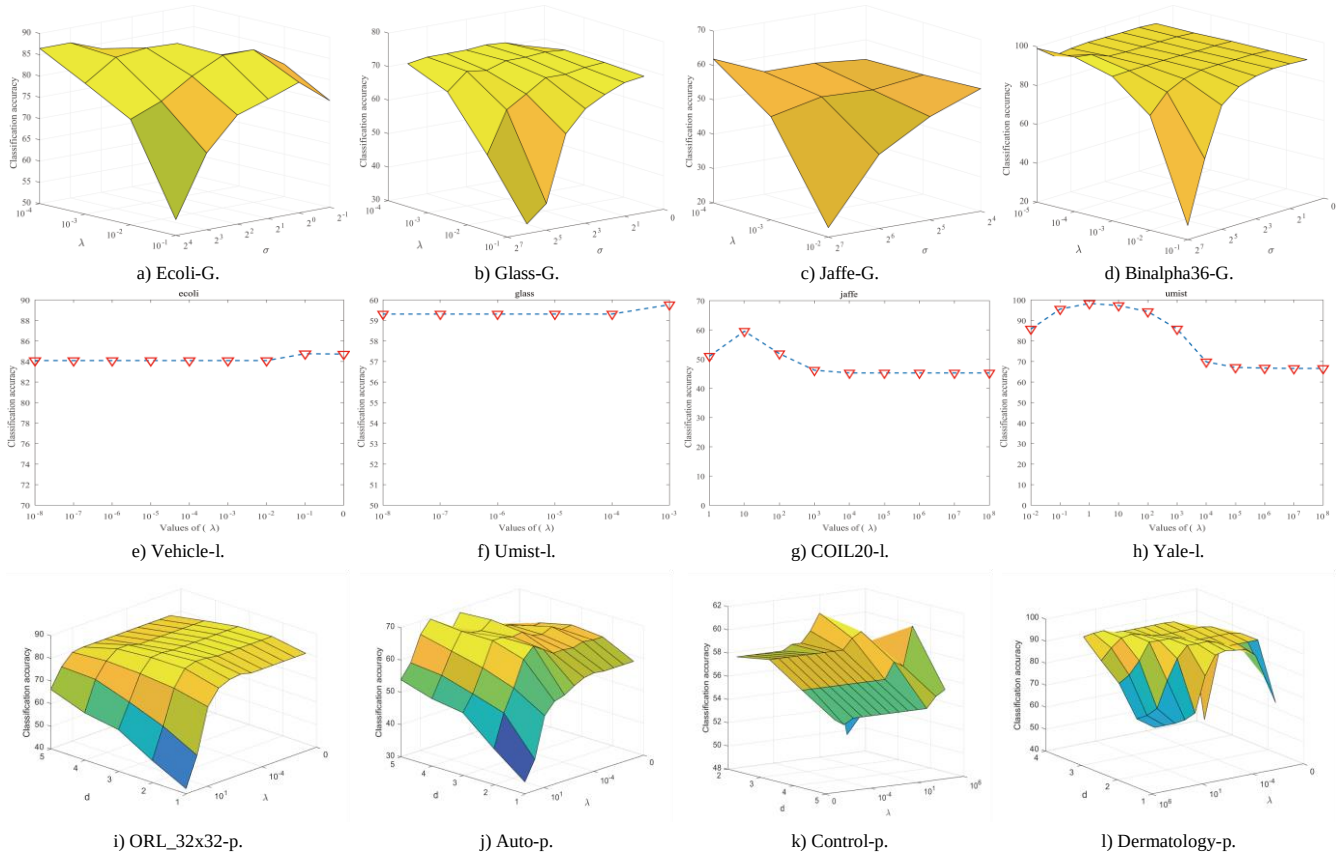
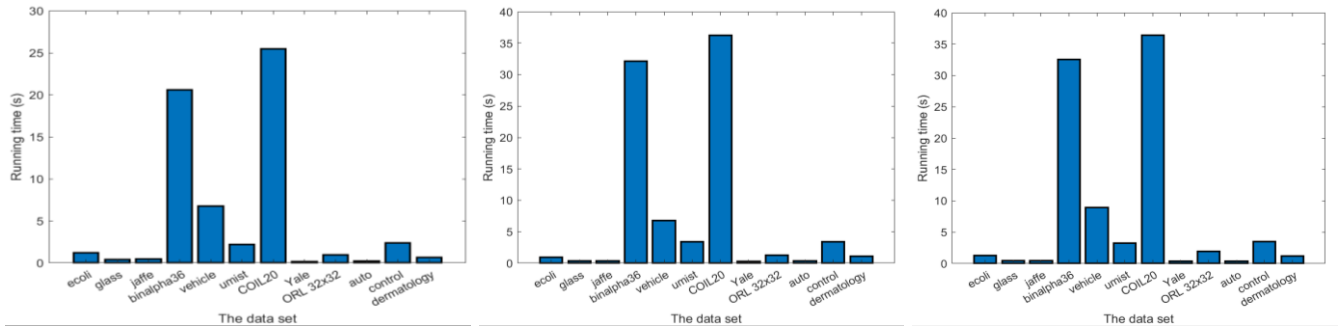


Figure 8. The parameter sensitivities of part of benchmark data sets. (a)-(d) correspond to the Gaussian kernel, (e)-(h) correspond to the linear kernel., (i)-(l) correspond to the polynomial kernel.

d) Running Time. To demonstrate the scalability of KLRRR, Figure 9 shows the running time of KLRRR model with Gaussian, linear, and polynomial kernels on the 12 benchmark data sets. In this figure, the horizontal axis represents the data set, and the vertical axis represents the running time for each data set under its respective optimal parameters. The platform used for these experiments was "Windows 10 64-bit + Intel(R) Core(TM) i5-7300 CPU @ 2.50 GHz+16 GB DDR4 2400 MHz+MATLAB 2023b". According to Table 1 and the results of Figure 9, overall, the computational efficiency of our algorithm is still quite fast, and the effect of sample dimensions on running time is relatively minor. This indicates that our model has an advantage in classifying high-dimensional data. In conclusion, our KLRRR model generally demonstrates ideal operational efficiency.



a) Gaussian kernel. b) Linear kernel. b) Polynomial kernel.

Figure 9. The running time of KLRR model under different kernel function on 12 benchmark data sets.

5. Conclusions

In this paper, we propose a joint kernel trick and low-rank semi-supervised regression model, called KLRRR, to achieve discriminative subspace learning and data classification. The advantages of KLRRR are summarized as follows:

1. It effectively solves the problem of linear inseparability of data in its current dimension.
2. It integrates the discriminant subspace projection into the regression model, solving the problem that a single projection matrix struggles to accurately construct the relationship between complex data and its labels.

3. It is implemented in a semi-supervised paradigm. The immediate benefit is that by estimating the soft labels of unlabeled samples, it can effectively guide discriminative subspace identification.

Experimental results showed that KLRRR performs well in improving classification accuracy, and the model is robust to different combinations of parameters. In addition, the model has a fast convergence speed. In the future, we will focus on the following two aspects: one is combining discriminative low-rank regression with feature selection of data. We aim to achieve automatic feature selection by embedding feature weighting factors. The other is improving the learning speed of the model.

6. Acknowledgment

This work was supported by National Key Research and Development Program of China (2023YFE0114900), MoE Key Laboratory of Embedded System and Services Computing (ESSCKF2024-11), and Zhejiang Province College Students' Science and Technology Innovation Activity Plan (New Talent Program) (2023R407071).

References

[1] Bunke O., Droge B., and Polzehl J., "Model Selection, Transformations and Variance Estimation in Nonlinear Regression," *Statistics*, vol. 33, no. 3, pp. 197-240, 1999. DOI:10.1080/02331889908802692

[2] Cai X., Ding C., Nie F., and Huang H., "On the Equivalent of Low-Rank Regressions and Linear Discriminant Analysis Based Regressions," *in Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Chicago, pp. 1124-1132, 2013. https://doi.org/10.1145/2487575.2487701

[3] Chen X., Yuan G., Nie F., and Huang J., "Semi-Supervised Feature Selection via Rescaled Linear Regression," *in Proceedings of the 26th International Joint Conference on Artificial Intelligence*, Melbourne, pp. 1525-1531, 2017. DOI:10.24963/ijcai.2017/211

[4] Coulston J., Blinn C., Thomas V., and Wynne R., "Approximating Prediction Uncertainty for Random Forest Regression Models," *Photogrammetric Engineering and Remote Sensing*, vol. 82, no. 3, pp. 189-197, 2016. DOI:10.14358/PERS.82.3.189

[5] Daemen A. and De Moor B., "Development of a Kernel Function for Clinical Data," *in Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, Minneapolis, pp. 5913-5917, 2009. DOI:10.1109/IEMBS.2009.5334847

[6] Dahmani M. and Guerti M., "Recurrence Quantification Analysis of Glottal Signal as non Linear Tool for Pathological Voice Assessment and Classification," *The International Arab Journal of Information Technology*, vol. 17, no. 6, pp. 857-866, 2020. DOI:10.34028/iajit/17/6/4

[7] Elbashir M. and Wang J., "Kernel Logistic Regression Algorithm for Large-Scale Data Classification," *The International Arab Journal of Information Technology*, vol. 12, no. 5, pp. 465-472, 2015. https://www.iajit.org/portal/PDF/Vol%2012,%20No.%205/6059.pdf

[8] Feng Q., Yuan C., Huang J., and Li W., "Center-Based Weighted Kernel Linear Regression for Image Classification," *in Proceedings of the IEEE International Conference on Image Processing*, Quebec City, pp. 3630-3634, 2015. DOI:10.1109/ICIP.2015.7351481

[9] Fitzmaurice G., "Regression," *Diagnostic Histopathology*, vol. 22, no. 7, pp. 271-278, 2016. DOI:10.1016/j.mpdhp.2016.06.004

[10] Frank I., "Modern Nonlinear Regression Methods," *Chemometrics and Intelligent Laboratory Systems*, vol. 27, no. 1, pp. 1-19, 1995. DOI: 10.1016/0169-7439(95)80003-R

[11] Fukumizu K., Bach F., and Jordan M., "Kernel Dimension Reduction in Regression," *The Annals of Statistics*, vol. 37, no. 4, pp. 1871-1905, 2009. DOI: 10.1214/08-AOS637

[12] Hartley H. and Booker A., "Nonlinear Least Squares Estimation," *The Annals of Mathematical Statistics*, vol. 36, no. 2, pp. 638-650, 1965. DOI:10.1214/aoms/1177700171

[13] Kutateladze V., "The Kernel Trick for Nonlinear Factor Modeling," *International Journal of Forecasting*, vol. 38, no. 1, pp. 165-177, 2022. DOI:10.1016/j.ijforecast.2021.05.002

[14] LaValley M., "Logistic Regression," *Circulation*, vol. 117, no. 18, pp. 2395-2399, 2008. DOI:10.1161/CIRCULATIONAHA.106.682658

[15] Lu H., Meng Y., Yan K., and Gao Z., "Kernel Principal Component Analysis Combining Rotation Forest Method for Linearly Inseparable Data," *Cognitive Systems Research*, vol. 53, pp. 111-122, 2019. DOI:10.1016/j.cogsys.2018.01.006

[16] Lunt M., "Introduction to Statistical Modelling: Linear Regression," *Rheumatology*, vol. 54, no. 7, pp. 1137-1140, 2015. DOI:10.1093/rheumatology/ket146

[17] Marill K., "Advanced Statistics: Linear Regression, Part II: Multiple Linear Regression," *Academic Emergency Medicine*, vol. 11, no. 1, pp. 94-102, 2004. DOI:10.1197/j.aem.2003.09.006

[18] Maulud D. and Abdulazeez A., "A Review on Linear Regression Comprehensive in Machine Learning," *Journal of Applied Science and*

Technology Trends, vol. 1, no. 2, pp. 140-147, 2020. DOI:10.38094/jastt1457

[19] Meer P., Mintz D., Rosenfeld A., and Kim D., "Robust Regression Methods for Computer Vision: A Review," *International Journal of Computer Vision*, vol. 6, no. 1, pp. 59-70, 1991. DOI:10.1007/BF00127126

[20] Moghaddam V. and Hamidzadeh J., "New Hermite Orthogonal Polynomial Kernel and Combined Kernels in Support Vector Machine Classifier," *Pattern Recognition*, vol. 60, pp. 921-935, 2016. DOI:10.1016/j.patcog.2016.07.004

[21] Naseem I., Togneri R., and Bennamoun M., "Linear Regression for Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 11, pp. 2106-2112, 2010. DOI:10.1109/TPAMI.2010.128

[22] Ostertagova E., "Modelling Using Polynomial Regression," *Procedia Engineering*, vol. 48, pp. 500-506, 2012. DOI:10.1016/j.proeng.2012.09.545

[23] Peng Y., Ke J., Liu S., Li J., and Lei T., "An Improvement to Linear Regression Classification for Face Recognition," *International Journal of Machine Learning and Cybernetics*, vol. 10, no. 9, pp. 2229-2243, 2019. https://link.springer.com/article/10.1007/s13042-018-0862-1

[24] Peng Y., Zhang Y., Kong W., Nie F., Lu B., and Cichocki A., "S3LRR: A Unified Model for Joint Discriminative Subspace Identification and Semisupervised EEG Emotion Recognition," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 1-13, 2022. DOI:10.1109/TIM.2022.3165741

[25] Peng Y., Zhu X., Nie F., Kong W., and Ge Y., "Fuzzy Graph Clustering," *Information Sciences*, vol. 571, pp. 38-49, 2021. DOI:10.1016/j.ins.2021.04.058

[26] Prajapati G. and Patle A., "On Performing Classification Using SVM with Radial Basis and Polynomial Kernel Functions," *in Proceedings of the 3rd International Conference on Emerging Trends in Engineering and Technology*, Goa, pp. 512-515, 2010. DOI: 10.1109/ICETET.2010.134

[27] Rajan K. and Murugesan V., "Hyperspectral Image Compression Based on DWT and TD with ALS Method," *The International Arab Journal of Information Technology*, vol. 13, no. 4, pp. 435-442, 2016. https://iajit.org/PDF/vol.13,%20no.4/7162.pdf

[28] Rochefort-Maranda G., "Simplicity and Model Selection," *European Journal for Philosophy of Science*, vol. 6, no. 2, pp. 261-279, 2016. DOI:10.1007/s13194-016-0137-1

[29] Sahoo D., Hoi S., and Li B., "Large Scale Online Multiple Kernel Regression with Application to Time-Series Prediction," *ACM Transactions on Knowledge Discovery from Data*, vol. 13, no. 1, pp. 1-33, 2019. DOI:10.1145/3299875

[30] Stulp F. and Sigaud O., "Many Regression Algorithms, one Unified Model: A Review," *Neural Networks*, vol. 69, pp. 60-79, 2015. DOI:10.1016/j.neunet.2015.05.005

[31] Tong H., Chen D., and Peng L., "Analysis of Support Vector Machines Regression," *Foundations of Computational Mathematics*, vol. 9, no. 2, pp. 243-257, 2009. DOI:10.1007/s10208-008-9026-0

[32] Wahyudi T. and Arroufu D., "Implementation of Data Mining Prediction Delivery Time Using Linear Regression Algorithm," *Journal of Applied Science and Technology Trends*, vol. 4, no. 1, pp. 84-92, 2022. DOI:10.37385/jaets.v4i1.918

[33] Wang W., Fang L., and Zhang W., "Robust Double Relaxed Regression for Image Classification," *Signal Processing*, vol. 203, pp. 108796, 2023. DOI:10.1016/j.sigpro.2022.108796

[34] Xu M., Watanachaturaporn P., Varshney P., and Arora M., "Decision Tree Regression for Soft Classification of Remote Sensing Data," *Remote Sensing of Environment*, vol. 97, no. 3, pp. 322-336, 2005. DOI:10.1016/j.rse.2005.05.008

[35] Yang Y., Yang Y., Shen H., Zhang Y., Du X., and Zhou X., "Discriminative Nonnegative Spectral Clustering with Out-of-Sample Extension," *IEEE Transactions on Knowledge and Data Engineering*, vol. 25, no. 8, pp. 1760-1771, 2013. DOI: 10.1109/TKDE.2012.118

[36] Yildiz K., Camurcu Y., and Dogan B., "Comparison of Dimension Reduction Techniques on High Dimensional Datasets," *The International Arab Journal of Information Technology*, vol. 15, no. 2, pp. 256-362, 2018. https://www.iajit.org/PDF/March%202018%2C%20No.%202/9699.pdf

[37] Zhang H., Yang J., Qian J., Gao G., Lan X., Zha Z., and Wen B., "Efficient Image Classification via Structured Low-Rank Matrix Factorization Regression," *IEEE Transactions on Information Forensics and Security*, vol. 19, pp. 1496-1509, 2024. DOI:10.1109/TIFS.2023.3337717

[38] Zhang Y., Shi D., Gao J., and Cheng D., "Low-Rank-Sparse Subspace Representation for Robust Regression," *in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, pp. 2972-2981, 2017. DOI:10.1109/CVPR.2017.317

[39] Zhao L., Chen Y., and Schaffner D., "Comparison of Logistic Regression and Linear Regression in Modeling Percentage Data," *Applied and Environmental Microbiology*, vol. 67, no. 5, pp. 2129-2135, 2001. DOI: 10.1128/AEM.67.5.2129-2135.2001

Qi Zhu received the B.S. degree from the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China, in 2021, where he is currently pursuing the M.S. degree. His research interests include Machine Learning, Pattern Recognition, and Data Processing.

Yong Peng (senior member, IEEE) received the Ph.D. degree from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2015. He is currently a Full Professor with the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China. His main research interests include Machine Learning, Pattern Recognition, and EEG-based Brain-Computer Interfaces. He has authored/co-authored 62 SCI/SSCI indexed journal papers such as IEEE TAFFC/TII/TNSRE/TCSVT/TMM/TCDS/TIM/TETC I. Dr. Peng was the recipient of the President Prize from the Chinese Academy of Sciences in 2009, the Third Prize from the Chinese Institute of Electronics in 2018 and the First Prize from the Zhejiang Graduate Education Association in 2022.