Ridge Redundancy Analysis for High-Dimensional Omics Data

Hayato Yoshioka[1,2], Julie Aubert [2], Hiroyoshi Iwata [1], Tristan Mary-Huard [2,3]*

[1] *Department of Agricultural and Environmental Biology, Graduate School of Agricultural and Life Science, The University of Tokyo, Tokyo, Japan* [2] *Université Paris-Saclay, AgroParisTech, INRAE, UMR MIA Paris-Saclay, 91120, Palaiseau, France* [3] *Université Paris-Saclay, INRAE, CNRS, AgroParisTech, GQE - Le Moulon, 91190, Gif-sur-Yvette, France.*

* Corresponding Author: tristan.mary-huard@agroparistech.fr

Abstract

Motivation Redundancy Analysis (RDA) is a popular multivariate approach for modeling the relationships between two sets of variables. In omics studies, RDA provides a flexible framework for investigating associations between high-dimensional molecular data. However, omics datasets often exhibit strong multicollinearity and suffer from the "large p, small n" problem, where the number of predictor variables p (features) exceeds the number of samples n. Ridge RDA addresses multicollinearity by introducing a ridge penalty in addition to the rank restriction. However, its application to high-dimensional omics data remains challenging due to i) difficulties in manipulating the large coefficient matrix of dimensions $p \times q$, where q represents the number of response variables, and ii) the need to select the rank and the regularization parameter jointly as the two of them influence the performance of the method.

Results We propose an efficient computational framework for ridge RDA that overcomes these challenges by leveraging the Singular Value Decomposition (SVD) of the predictor matrix X. This approach eliminates the need for direct covariance matrix inversion, improving computational efficiency. Furthermore, we introduce a novel strategy that reduces the memory burden associated with the storage of the coefficient matrix from pq to $(p+q)r$, with r the chosen reduced rank dimensionality. Our method enables an efficient grid search to jointly select the ridge penalty λ and r via cross-validation. The proposed framework is implemented in the R package `rrda`, providing a practical and scalable solution for the analysis of high-dimensional omics data.

Availability The source code for the R package `rrda`, along with the data used for implementation and analysis, is available on GitHub: `https://github.com/Yoska393/rrda`

1 Introduction

Redundancy Analysis (RDA; van den Wollenberg, 1977) is a widely used multivariate statistical method for investigating relationships between two sets of variables (Reinsel & Velu, 1998). Also referred to as reduced-rank regression (Anderson, 1951) and principal components of instrumental variables (Rao, 1964), RDA is a flexible technique for modeling multiple response variables as functions of multiple predictor variables. Unlike Canonical Correlation Analysis (Hotelling, 1936), which maximizes the correlation between two variable sets, RDA aims to optimize the predictability of one set from the other.

RDA is commonly applied in ecological studies, where the response matrix Y of dimensions $n \times q$ typically represents community composition, while the predictor matrix X of dimensions $n \times p$ contains environmental variables explaining the variation in Y. The method is based on the multivariate linear model $Y = XB + E$, where B is the coefficient matrix subject to the rank constraint $\text{rank}(B) < r$, ensuring dimensionality reduction while preserving essential predictive information. It is beneficial when the coefficient matrix is not of full rank, as its reduced-rank structure efficiently captures and exploits redundant information. This constraint facilitates the identification of dominant patterns in the data, making RDA an effective tool for ecological and environmental analyses.

In recent years, RDA has been increasingly applied in various omics studies. The field of multi-omics has emerged as a powerful approach for deciphering complex biological pathways (Buescher & Driggers, 2016). Notable applications include gene expression analysis (Wen et al., 2023), DNA methylation studies (Ruiz-Arenas & González, 2017), and metabolomic research (Hilafu et al., 2020). However, omics data are typically high-dimensional and characterized by highly correlated variables, leading to challenges related to multicollinearity.

To mitigate multicollinearity, regularized redundancy analysis (Takane & Hwang, 2007), also referred to as reduced-rank ridge regression (Mukherjee & Zhu, 2011), is commonly employed in omics research. By introducing a regularization term, ridge regression reduces the impact of collinearity by shrinking the coefficients of correlated predictors, thus preventing overfitting and enhancing generalization performance. This regularization improves stability and generalization to new datasets, making it a valuable tool for analyzing complex biological systems.

Although ridge regression effectively addresses multicollinearity, applying ridge RDA to high-dimensional omics data remains challenging. A primary issue is the "large p, small n" problem, where the number of predictor variables (p) in the matrix X exceeds the number of samples (n). This imbalance complicates conventional RDA, as it prevents the straightforward inversion of the covariance matrix of X (Huang et al., 2017). Compared to ridge regression, the ridge RDA approach raises new challenges, as both the number of response variables q and the number of predictors p may be large (e.g. $p, q > 10^5$), representing some significant computational and storage challenges for the coefficient matrix B of size $(p \times q)$. Despite these limitations, no formal statistical framework has been developed to efficiently manage the computational burden and memory requirements for ridge RDA, particularly in high-dimensional omics research.

To address these challenges, we propose a novel approach for computing the regularization path and optimizing data storage for ridge RDA. This method eliminates the need for direct inversion of the covariance matrix by introducing the Singular Value Decomposition (SVD) of the predictor matrix X, thus improving computational efficiency in high-dimensional settings. Furthermore, we introduce a new decomposition of the coefficient matrix B that reduces its storage in memory from pq to $(p+q)r$, making the approaches amenable to large-dimension settings. These computational enhancements open the way to efficient parameter tuning, where the ridge regularization parameter λ and the rank (r) are jointly selected through a cross-validation (CV) process. Our method is implemented in the R package `rrda`.

This article introduces an efficient ridge RDA framework designed for high-dimensional omics data. We begin by providing a theoretical foundation that demonstrates how our approach improves computational efficiency in ridge RDA analysis. Next, we conduct simulation studies to compare the performance of our method with classical approaches across a range of dataset sizes. Finally, the method is illustrated through several applications to high-dimensional omics datasets of varying sizes and types, including DNA copy number, gene expression, metabolomics, microbiome, and methylation-expression data. These applications demonstrate the efficiency of the approach to effectively handle high-dimensional data.

2 Materials and Methods

2.1 Ridge Redundancy Analysis

Let Y be an $n \times q$ matrix of response variables, and let X be an $n \times p$ matrix of predictor variables. We assume that the columns of matrices Y and X are centered to have means of 0.

$$Y = XB + E$$

where B is the matrix of regression coefficients, and E is the error matrix. The ridge RDA aims to obtain the coefficient B under rank and ridge restrictions. To achieve this, we define the objective function for a ridge regression model:

$$\phi_\lambda(B) = \|Y - XB\|^2 + \lambda\|B\|^2$$

where $\|A\|^2 = \mathrm{tr}\,(A'A)$, which denotes the Frobenius norm, and λ is a positive value of the ridge penalty. The rank of B is under restriction

$$\mathrm{rank}(B) \leq r$$

To separate the term involving Y from B, we can rewrite the expression as:

$$\phi_\lambda(B) = \|Y\|^2_{Q_X(\lambda)} + \|\hat{B}(\lambda) - B\|^2_{X'X+\lambda P_{X'}} \tag{1}$$

where $\|A\|^2_M = \mathrm{tr}\,(A'MA)$, and

$$\hat{B}(\lambda) = \left(X'X + \lambda P_{X'}\right)^{-1} X'Y \tag{2}$$

$$P_X(\lambda) = X\left(X'X + \lambda P_{X'}\right)^{-1} X' \tag{3}$$

$$Q_X(\lambda) = I - P_X(\lambda) \tag{4}$$

where $P_{X'} = X'\left(XX'\right)^{-1} X$.

$\hat{B}(\lambda)$ represents the ridge estimate of B without rank restriction. It is important to note that the first term on the right-hand side of (1) is independent of B, effectively eliminating Y. To obtain the ridge and rank-restrained coefficient $\hat{B}(\lambda, r)$, we minimize the second term, which can be done using the Generalized Singular Value Decomposition (GSVD) method (Cailliez & Pages, 1976; Greenacre, 1984). See the Appendix for details on the process and the definition of GSVD.

$$\mathcal{GSVD}(\hat{B}(\lambda))_{LL',I} = U_{\hat{B}(\lambda)} D_{\hat{B}(\lambda)} V'_{\hat{B}(\lambda)} \tag{5}$$

Here, $LL' = X'X + \lambda P_{X'}$, which is the row metric of GSVD, and the column metric is identity matrix I. The ridge-penalized-r-rank estimate $\hat{B}(\lambda, r)$ is then obtained by retaining only the portions of GSVD components corresponding to the first r dominant singular values.

$$\hat{B}(\lambda, r) = U^{[r]}_{\hat{B}(\lambda)} D^{[r]}_{\hat{B}(\lambda)} V^{[r]\prime}_{\hat{B}(\lambda)} \tag{6}$$

Here, $U^{[r]}_{\hat{B}(\lambda)}$ is a $p \times r$ matrix, $D^{[r]}_{\hat{B}(\lambda)}$ is a $r \times r$ diagonal matrix $V^{[r]}_{\hat{B}(\lambda)}$ is a $q \times r$ matrix. By using those components, we can choose the best rank r so that it minimizes the objective function under a ridge restriction.

The equation involves the inversion of $X'X + \lambda P_{X'}$ in (2), which is of size $p \times p$, posing significant computational challenges. Additionally, the coefficient matrix B, with dimensions $p \times q$, presents both computational and storage difficulties. Furthermore, model parameter tuning requires repeating this procedure for each combination of the ridge penalty (λ) and rank constraint (r), which can present significant challenges in high-dimensional settings.

2.2 Adaptation to the High-dimensional Setting

We extend the ridge RDA framework to high-dimensional settings (where $p, q \gg n$). To handle this efficiently, we introduce a computational shortcut that reduces the dimensionality, following Hastie et al. (2001, Section 18.3.5). Specifically, we first perform the singular value decomposition (SVD) of X as

$$X = UDV' \tag{7}$$

Note that this SVD is performed only once for all parameters λ and r. Using (7), the equation (3) and (4) are rewritten as follows,

$$P_X(\lambda) = UD^2\left(D^2 + \lambda I\right)^{-1} U' \tag{8}$$

$$Q_X(\lambda) = U(\lambda(D^2 + \lambda I)^{-1})U' \tag{9}$$

where we assume $\lambda > 0$. The equation (8) and (9) must be obtained for each λ. From (2), the ridge estimator $\hat{B}(\lambda)$ is given by:

$$\hat{B}(\lambda) = VD\left(D^2 + \lambda I\right)^{-1} U'Y$$

Row metric LL' can be expressed as follows:

$$LL' = V(D^2 + \lambda I)V' \tag{10}$$

Thus, the GSVD of matrix $\hat{B}(\lambda)$ is donated as

$$\mathcal{GSVD}(\hat{B}(\lambda))_{LL',I} = V(D^2 + \lambda I)^{-1/2}\hat{U}_\lambda \hat{D}_\lambda \hat{V}'_\lambda$$

Here, the $\hat{U}_\lambda$, $\hat{D}_\lambda$, and $\hat{V}'_\lambda$ represent a parameter λ- dependent SVD components of the matrix $(D^2 + \lambda I)^{-1/2}DU'Y$ as follows:

$$\mathcal{SVD}\left((D^2 + \lambda I)^{-1/2}DU'Y\right) = \hat{U}_\lambda \hat{D}_\lambda \hat{V}'_\lambda \tag{11}$$

Thus, those values need to be updated for each value of λ. Referring to (5), the GSVD of $\hat{B}(\lambda)$ is expressed as:

$$\begin{aligned} U_{\hat{B}(\lambda)} &= V(D^2 + \lambda I)^{-1/2}\hat{U}_\lambda, \\ D_{\hat{B}(\lambda)} &= \hat{D}_\lambda, \\ V_{\hat{B}(\lambda)} &= \hat{V}_\lambda \end{aligned}$$

For the r-rank estimate $\hat{B}(\lambda, r)$, we adhere to the formulation presented in (6). Here, we benefit from a computationally efficient procedure.

$$\hat{B}(\lambda, r) = V(D^2 + \lambda I)^{-1/2}\hat{U}_\lambda^{[r]}\hat{D}_\lambda^{[r]}\hat{V}_\lambda^{[r]\prime} \tag{12}$$

Here, equation (12) corresponds to (6), while the computation is efficiently facilitated in high-dimensional settings (where $p, q \gg n$). This formulation leverages the SVD to avoid the direct inversion of the $X'X + \lambda P_{X'}$ matrix, which is of size $p \times p$, thereby significantly reducing computational complexity.

Note that $(D^2 + \lambda I)^{-1/2}$ is the inverse square root of a diagonal matrix. Thus, the computation reduces to simple element-wise operations on the diagonal entries, given by $(d_i^2 + \lambda)^{-1/2}$, where d_i denotes the i-th singular value of X. This leads to a highly efficient and low-cost computation.

The SVD of X is computed only once across all parameters in (7), thereby eliminating the need for repeated matrix inversion for each λ. While the second SVD must be computed for each value of λ in (11), this computation is performed on the $n \times q$ matrix, which is computationally less expensive than dealing with the $p \times q$ matrix B.

Furthermore, in omics studies, the rank r is often smaller than both n and p, enabling efficient SVD computation using an optimized algorithm from the R package `RSpectra` (Qiu & Mei, 2024, version 0.16-2), which computes only the top singular values.

In conventional methods, where the number of predictors p and q exceeds the number of observations n, the computational cost typically increases substantially. However, with this approach, computations are performed in an n-dimensional space rather than a p-dimensional one, reducing the computational burden, particularly in high-dimensional settings. See the Appendix for details on the derivation of the equations and computational algorithms.

2.3 Data storing for high dimensional matrices

In omics studies, the ridge RDA coefficient matrix, denoted as $\hat{B}(\lambda, r)$, is often challenging to store in computation due to its large size as a $p \times q$ matrix. The size can make it computationally infeasible to perform processes involving $\hat{B}(\lambda, r)$. To address this issue, we propose an efficient storage method. From (6), for $i = 1, \ldots, r$, we define the matrices F and G as

$$\hat{B}(\lambda, r) = \sum_{i=1}^{r} (F_{.i} G'_{.i})$$

where

$$F_{.i} = U^{[i]}_{\hat{B}(\lambda)} D^{[i]}_{\hat{B}(\lambda)}, \quad G_{.i} = V^{[i]}_{\hat{B}(\lambda)}$$

Here, $F_{.i}$ and $G_{.i}$ denote the i-th columns of the matrices F and G, respectively. Specifically, $F_{.i}$ is the product of the i-th column of the matrix $U_{\hat{B}(\lambda)}$ and the diagonal matrix $D_{\hat{B}(\lambda)}$ corresponding to the i-th eigenvalue, while $G_{.i}$ is the i-th column of the matrix $V_{\hat{B}(\lambda)}$. This decomposition significantly reduces storage requirements, as F is a $p \times r$ matrix and G is a $q \times r$ matrix, both of which are much smaller than the full $p \times q$ matrix $\hat{B}(\lambda, r)$, particularly in high-dimensional settings.

The function `rrda.fit` returns the coefficient matrix $\hat{B}(\lambda, r)$ in either a component form, represented by the matrices F and G with a total storage size of $(p + q)r$, or as a full $p \times q$ matrix for all combinations of λ and r. From a data storage perspective, the component form is preferable due to its reduced memory requirements.

In the function `rrda.pred`, the matrices F and G are utilized efficiently to avoid direct computation of $\hat{B}(\lambda, r)$ in its full matrix form. Instead, predictions are computed as follows:

$$\hat{Y}_r = X \left(\sum_{i=1}^{r} (F_{.i} G'_{.i}) \right) = \sum_{i=1}^{r} (X F_{.i} G'_{.i}) = \sum_{i=1}^{r} (\tilde{X}_{.i} G'_{.i})$$

where $\tilde{X} = XF$, and r denotes the rank of $\hat{B}(\lambda, r)$. This formulation allows for an efficient computation of the predictions by iteratively computing vector products at each step. By adopting this approach, these functions provide a storage-efficient representation of $\hat{B}(\lambda, r)$ for predictions as well.

2.4 Cross validation for parameter tuning

To determine the optimal rank r and ridge penalty λ, it is crucial to evaluate a range of values for both parameters. This process can be extremely challenging or even infeasible, particularly in high-dimensional settings. However, the computational shortcut and efficient storing outlined earlier prove useful in reducing these challenges, making it possible to perform a grid search via cross-validation. This approach enables the calculation of the regularization path for different values of λ and r, thereby streamlining the parameter selection process.

J-fold cross-validation method is used to choose r and λ at the same time, as follows

$$\epsilon(\lambda, r) = \frac{1}{N}\frac{1}{J}\sum_{j=1}^{J} \|Y^{(j)} - X^{(j)}\hat{B}(\lambda, r)^{(-j)}\|^2$$

Here, the matrix of estimated ridge RDA coefficients for the subset except for j^{th} sub-samples is represented as $\hat{B}(\lambda, r)^{(-j)}$. Mean Squared Error (MSE) was calculated for the jth subset of cases.

In practice, we choose the value of λ associated with the grid search of λ through $\epsilon(\lambda, r)$. For the range of λ, we referred to the algorithm of `glmnet` (Friedman et al., 2010) as follows

$$\lambda_{\max} = \frac{\max_{j\in\{1,2,\ldots,p\}} \sqrt{\sum_{k=1}^{q}\left(\sum_{i=1}^{n}(x_{ij}\cdot y_{ik})\right)^2}}{N \times 10^{-3}} \tag{13}$$

Then, we define $\lambda_{min} = 10^{-4}\lambda_{max}$, and the sequence λ is generated based on the range.

However, the (13) poses computational challenges in high-dimensional settings. Thus, to efficiently find a range of λ, a variable sampling method was implemented. When the number of variables (p or q) is larger than v, the model function performs sampling of v variables from each matrix. The sampled matrices, X and Y, are used to determine the range of λ, and the subsequent cross-validation is performed on the full matrices X and Y. The default value of v is 1000. This sampling method achieved high accuracy in a simulation while significantly reducing computation time and memory usage. The simulation details and results are shown in Figure 1S.

Furthermore, to define robust parameters, we extend the `glmnet` algorithm by incorporating a robust ridge parameter λ that provides a more regularized model, ensuring that the mean squared error (MSE) is within one standard error of the smallest MSE. This approach is referred to as the "one standard error rule" (Friedman et al., 2010). We apply this rule to the dimensionality parameter (r) as well, selecting the smallest r and the largest λ within the one standard error range. Alternatively, other information criteria, such as the Akaike information criterion (AIC, Akaike 1974) and the Bayesian information criterion (BIC, Schwarz 1978), are commonly used and were compared in this study.

Additionally, the functions are designed for parallel computing depending on `future` package (H. Bengtsson et al., 2021), which facilitates the computation for multiple parameters λ and rank. See Appendix for the details of parallel computing.

2.5 Simulation data

In this section, we describe two simulation data generation models to demonstrate the capabilities of various sizes of matrices. We developed two types of data-generating processes, including (1) the latent space model and (2) the matrix factorization model. The choice of model depends on research interests. In brief, the latent space model is optimized for high-dimensional settings, while the matrix factorization model is useful for obtaining the true B value.

Model 1: Latent Space Model

This model is effective for generating rank-restricted data in high-dimensional settings because it avoids the need to estimate a full coefficient matrix $B \in \mathbb{R}^{p \times q}$.

Instead, both the predictor and response variables are generated from a shared latent space. Let $H \in \mathbb{R}^{n \times k}$ be a latent variable matrix, where n is the number of observations and k is the dimension of the latent space. The entries of H are independently drawn from a standard normal distribution:

$$H_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

Let $\theta_x \in \mathbb{R}^{k \times p}$ and $\theta_y \in \mathbb{R}^{k \times q}$ be coefficient matrices, with their entries independently drawn from standard multivariate normal distributions:

$$(\theta_x)_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1), \quad (\theta_y)_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

The observed predictor matrix $X \in \mathbb{R}^{n \times p}$ and response matrix $Y \in \mathbb{R}^{n \times q}$ are then generated as:

$$X = H\theta_x + E_x$$

$$Y = H\theta_y + E_y$$

where

$$(E_x)_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_x^2), \quad (E_y)_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma_y^2)$$

Let $E_x \in \mathbb{R}^{n \times p}$ and $E_y \in \mathbb{R}^{n \times q}$ be independent noise matrices drawn from the following parameter:

$$\sigma_x = \sqrt{\frac{\|H\theta_x\|^2}{\|\tilde{E}_x\|^2 \cdot s_x}}, \quad \sigma_y = \sqrt{\frac{\|H\theta_y\|^2}{\|\tilde{E}_y\|^2 \cdot s_y}}$$

and

$$(\tilde{E}_x)_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1), \quad (\tilde{E}_y)_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

Let $\tilde{E}_x \in \mathbb{R}^{n \times p}$ and $\tilde{E}_y \in \mathbb{R}^{n \times q}$. In this context, s_x and s_y denote the signal-to-noise ratio for X and Y, respectively. We tested values $s_x = s_y = 1, 5, 10$, and the results presented in the following sections correspond to $s_x = s_y = 5$. Similar trends were observed across the other tested values. For details, see the Appendix.

Here, X and Y are influenced by the shared latent structure H, while the noise matrices E_x and E_y introduce variability not explained by the latent space.

Model 2: Matrix Factorization Model

This model is useful in scenarios where we aim to generate rank-restricted data while preserving access to the true coefficient matrix B. This is particularly beneficial for validating prediction performance. However, computational cost increases in high-dimensional settings due to the growing size of B.

The simulation model employs a matrix factorization approach to characterize the relationship between predictor and response variables. Let $X \in \mathbb{R}^{n \times p}$ be the predictor matrix, where n is the number of observations and p is the number of predictor variables. The entries of X are sampled from a multivariate normal distribution:

$$X_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

Next, we generate two coefficient matrices $A \in \mathbb{R}^{p \times k}$ and $C \in \mathbb{R}^{k \times q}$, where k is the number of latent factors and q is the number of response variables. Both A and C are drawn from standard normal distributions:

$$A_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1), \quad C_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

The columns of A and C are then normalized to form A^* and C^*, respectively. The coefficient matrix B, representing the linear relationship between X and the response matrix Y, is constructed as:

$$B = A^* C^*$$

The observed response matrix $Y \in \mathbb{R}^{n \times q}$ is generated as a linear combination of X and B, with added noise:

$$Y = XB + E = XA^*C^* + E$$
$$E_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, \sigma^2)$$

Here, $\tilde{E}_{ij} \in \mathbb{R}^{n \times q}$ is an independent noise matrix, and σ is chosen to control the signal-to-noise ratio (SNR) as follows:

$$\sigma = \sqrt{\frac{\|XB\|^2}{\|\tilde{E}\|^2 \cdot s}}$$

where

$$\tilde{E}_{ij} \overset{\text{i.i.d.}}{\sim} \mathcal{N}(0, 1)$$

In this context, s denotes the signal-to-noise ratio. We tested values $s = 1, 5, 10$, and the results presented in the following sections correspond to $s = 5$. Similar trends were observed across the other tested values. For details, see the Appendix.

This simulation captures a structured relationship between the predictors and responses through the low-rank matrix B, which reflects multiplied latent factors. The inclusion of the noise matrix E introduces random perturbations, enabling assessment of the model's ability to handle structured data under varying noise conditions.

2.6 Simulation scenarios

To evaluate the proposed efficient ridge redundancy analysis (`rrda`) across various matrices, we assess the performance through three simulation scenarios: rank estimation in matrices (Scenario 1), computational efficiency in high-dimensional data (Scenario 2), and prediction accuracy assessment (Scenario 3). See the Appendix for details on the computational resources (Table 1S).

Scenario 1

To test the rank estimation for high-dimension data, we conducted a simulation study with randomly generated parameters over 100 iterations. In each iteration, the parameters n, p, q, and k were sampled from predefined ranges under three different matrix size settings: *Large*, *Small*, and *Mixed*.

In the *Large* setting, the number of samples n ranged from 50 to 100, while the matrix dimensions p and q ranged from 100 to 1000. In the *Small* setting, n was between 100 and 200, with p and q varying from 20 to 100. The *Mixed* setting combined a broad range of matrix sizes, with n ranging from 50 to 200 and p, q ranging from 20 to 1000. In all settings, the true rank k was randomly selected from 1 to 10.

For each parameter set, we generated data using two models: Model 1 (Latent Space Model) and Model 2 (Matrix Factorization Model). Then, we tested the accuracy of the rank estimations.

Scenario 2

To assess the computational capability of our model for super high-dimensional data, we evaluated the computation time of the `rrda.fit` function and compared it with the conventional function `rrs.fit` from the R package `rrpack` (version 0.1-13, Chen & Wang, 2022). The `rrpack` package implements reduced-rank ridge regression but does not incorporate the efficient computational shortcuts required for high-dimensional settings. For the simulations, data were generated based on model 1 (latent space model). We fixed the parameters $n = 100$ and $k = 5$, and tested several values for p and q, specifically $p = q = 10^2, 10^3, 10^4$, 10^5, and 10^6.

Scenario 3

To further elucidate the performance of various methods, we conducted the following random-splitting procedure 100 times. We set the parameters as fixed values: $n = 100$, $p = 200$, $q = 200$, and $k = 5$, using both Model 1 and Model 2. The data were randomly split into a training set ($n_{\text{train}} = 90$) and a test set ($n_{\text{test}} = 10$). Within the training set, 5-fold cross-validation was performed to tune parameters. The mean squared prediction error (MSPE) was then evaluated on the test set and calculated as follows:

$$\text{MSPE} = 100 \times \left\| Y_{\text{test}} - X_{\text{test}} \hat{B}(\lambda, r) \right\|^2 / q n_{\text{test}}$$

Alongside CV via `rrda`, we compared the mean squared prediction error (MSPE) across various methods, including classical model selection criteria such as AIC and BIC, as well as the Generalized Information Criterion (GIC), Generalized Cross-Validation (GCV), and the Bayesian Information Criterion with Penalty (BICP) (Table 2S). These methods are implemented in the R package `rrpack` (Chen & Wang, 2022). We also included the recently

proposed "Stability Approach to Regularization Selection for Reduced-Rank Regression" (StARS-RRR; Wen et al., 2023) in our comparison, alongside our proposed `rrda` method. The predicted dimensionality r and the ridge parameter λ were also examined.

To evaluate different parameter tuning strategies, we compared the following six settings: (1) **CV min**, which uses the best set of parameters selected by cross-validation; (2) **Rank.1se**, which selects the rank using the one-standard-error rule while keeping λ the same as in CV min; (3) **$\boldsymbol{\lambda}$.1se**, which selects $\boldsymbol{\lambda}$ using the one-standard-error rule while keeping the rank the same as in CV min; (4) **Full Rank**, a full-rank multivariate model using the $\boldsymbol{\lambda}$ from CV min; (5) **$\boldsymbol{\lambda}$0,** which uses the rank estimated by CV min without applying $\boldsymbol{\lambda}$ regularization; and (6) **Full Rank + $\boldsymbol{\lambda}$0**, a standard multivariate linear model without rank restriction or regularization.

2.7 Application data

We used three publicly available high-dimensional omics datasets to evaluate the proposed efficient ridge redundancy analysis (`rrda`) method: a breast cancer dataset, a soybean multi-omics dataset, and a methylation-expression dataset.

The breast cancer dataset includes gene expression data as explanatory variables (X) and DNA copy number variations (CNVs) as responses (Y), measured from $n = 89$ patients. The aim was to study the relationships between CNVs and gene expression profiles across different patients. This dataset, previously explored in studies by Chen et al. (2013) and Wen et al. (2023), is available via the R package `PMA` version 1.2-4 (Witten et al., 2009). We analyzed two versions of this breast dataset: one limited to chromosome 13, where $X \in \mathbb{R}^{89\times 319}$ and $Y \in \mathbb{R}^{89\times 58}$, and another using whole-genome data, where $X \in \mathbb{R}^{89\times 19,672}$ and $Y \in \mathbb{R}^{89\times 2,149}$.

The soybean multi-omics dataset consists of $n = 179$ samples, where microbiome species (or taxonomic units) data serve as explanatory variables $X \in \mathbb{R}^{179\times 4771}$, and metabolome data serve as responses $Y \in \mathbb{R}^{179\times 253}$, measured from different soybean accessions under drought conditions. The aim was to study the relationships between microbiome and metabolome profiles across different accessions. This dataset was previously analyzed using a clustering method I-SVVS (Integrative Stochastic Variational Variable Selection) proposed by Dang et al. (2023, bioRxiv).

The methylation-expression dataset contains data from $n = 20$ samples, with DNA methylation as input $X \in \mathbb{R}^{20\times 392,277}$, and gene expression as output $Y \in \mathbb{R}^{20\times 67,528}$. The aim was to study the relationships between methylation and gene expression profiles across different patients. This dataset is available via the R package `MEAL` version 1.34.0 (Ruiz-Arenas and González 2024). The dataset is designed to evaluate the relationship between epigenetic regulation and transcriptional activity across the genome. Here, our method demonstrates how DNA methylation features can be used to explain variation in gene expression profiles. Its extreme dimensionality ($p \gg n$ and $q \gg n$) presents a challenging test case for multivariate prediction methods.

3 Results

3.1 Simulation Results

This section evaluates the performance of the proposed efficient `rrda` procedure across different matrix sizes. We begin with a brief demonstration of how our `rrda` package visualizes the regularization path, illustrating the behavior of rank and penalty parameter selection. Following this, we assess the performance through three simulation scenarios described in methods.

Visualization

We illustrate how our `rrda` visualizes the regularization path using a controlled simulation setting with parameters $n = 100$, $p = 200$, $q = 200$, and $k = 5$, employing both model 1 and model 2 (Figure 1). We demonstrate that the optimal rank and ridge penalty λ can be estimated through cross-validation (Table 1) to minimize the mean squared error (MSE). For selecting robust parameters, we apply the "one-standard-error" rule (Friedman et al., 2010), where the largest λ and the smallest rank are chosen such that the MSE remains within one standard error of the minimum.

Scenario 1

To test the accuracy of rank estimation, we applied `rrda` to simulated datasets generated under random parameter settings (Table 2). The results show consistently high correlations between the true and estimated ranks across all settings. The `rrda` achieved a correlation of 0.995 in the Large setting and 0.994 in the Small setting for Model 1 (Latent Space Model), and near-perfect accuracy (1.00) in the Small setting for Model 2 (Matrix Factorization Model). These findings suggest that `rrda` accurately recovers the true rank structure, both in small- and high-dimensional data, thereby demonstrating the scalability of the proposed `rrda` procedure. Even under higher noise levels, `rrda` maintained strong performance (see Appendix for details).

Scenario 2

In this scenario, we evaluated the computational performance of our method on extremely high-dimensional data. The results indicate that our approach is computationally more efficient than the conventional method, particularly for larger matrices (Table 3). Notably, only our method was able to handle extremely high-dimensional cases ($p = q \geq 10^4$). Details of the computational environment are provided in the Appendix. For example, when $p = 10^6$, the function `rrs.fit` in the R package `rrpack` required approximately 7450.6 GB of memory, rendering the computation practically infeasible, whereas our approach successfully completed the task.

Scenario 3

To further elucidate the performance of various methods, we evaluated the mean squared prediction error (MSPE) using a random-splitting procedure repeated 100 times (Table 4).

AIC, BIC, and StARS-RRR tend to overestimate the rank. In contrast, methods such as GCV, GIC, and BICP tend to recommend more appropriate ranks. Notably, CV with `rrda` achieved the lowest MSPE among all methods, demonstrating its effectiveness in tuning both rank and the ridge parameter λ. Even under higher noise levels, `rrda` maintained strong performance (see Appendix for details).

We also evaluated several parameter settings (Table 5). In simulation model 1, the CV min model and the Rank.1se model performed best in terms of MSPE, with no significant difference between them. In simulation model 2, the CV min model achieved the best MSPE score. These results demonstrate that CV min effectively selects parameters. Even under higher noise levels, similar prediction patterns were observed (see Appendix for details). Therefore, we employ the CV min model for applications in this article. If users prefer more robust parameters for their purposes, the one-standard-error rule is also recommended.

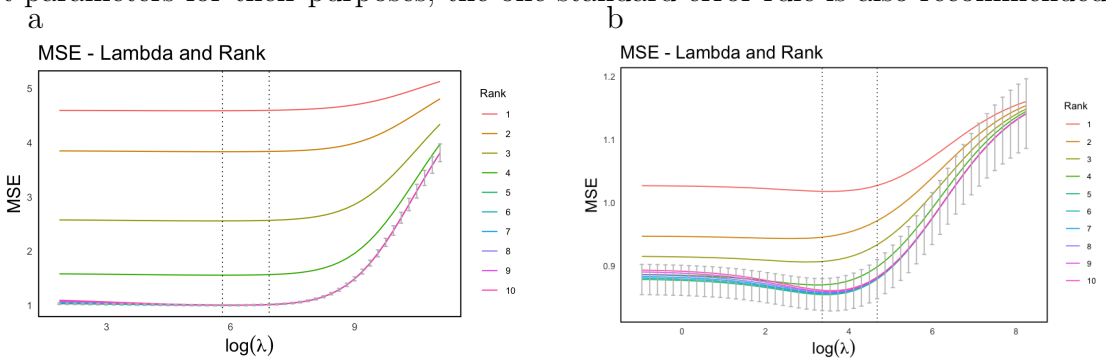

Figure 1: Visualization of regularization path with our efficient `rrda` package. Plot of mean squared error for all parameters. The subsampling was performed on training data (n=90) and test data (n=10). Plots present the mean squared value for all combinations of ridge penalty λ and rank of B. The left vertical line corresponds to the mean squared error, while the right vertical line is the largest value of lambda such that the error is within one standard error of the minimum, which is plotted as the error bar. (a) Latent Space Model (b) Matrix Factorization Model

Table 1: The table of MSE in the setting of $n = 100$, $p = 200$, $q = 200$, and $k = 5$. CV finds the best MSE at each rank of the λ set. Evaluated for Model 1 (Latent Space Model) and Model 2 (Matrix Factorization Model).

	model1		model2	
Rank	MSE	SE	MSE	SE
1	4.5858	0.2282	0.9899	0.0497
2	3.8312	0.1901	0.8954	0.0357
3	2.5561	0.1317	0.8569	0.0424
4	1.5565	0.0840	0.8084	0.0394
5	**1.0016**	**0.0131**	**0.7827**	**0.0375**
6	1.0031	0.0134	0.7847	0.0373
7	1.0036	0.0133	0.7863	0.0370
8	1.0049	0.0132	0.7883	0.0371
9	1.0043	0.0131	0.7910	0.0369
10	1.0056	0.0131	0.7937	0.0369

Table 2: Scenario 1. Rank estimation. In 100 iterations, matrices X and Y are generated from random parameters in the table range. Simulations were performed for three groups of matrix sizes (Large, Small, Mixed). Ranks are estimated by CV (5 folds). The table shows the correlation between true and estimated ranks for Model 1 (Latent Space Model) and Model 2 (Matrix Factorization Model).

	Parameter Range				COR-Rank	
	n	p	q	k	Model 1	Model 2
Large	50-100	100-1000	100-1000	1-10	.995	.971
Small	100-200	20-100	20-100	1-10	.994	1.00
Mixed	50-200	20-1000	20-1000	1-10	.994	.993

Table 3: Scenario 2. Benchmark of Computation Time (rrda vs. rrpack); Data were generated using Model 1: the latent space model. The computation time (in seconds) was recorded. The variable p and q represents the numbers of variables in the X and Y matrices. The mean and standard error (SE) of the computation time (in seconds) are reported. "N/A" indicates that the computation was not converged. For $p = q = 10^6$, the required memory for `rrpack` was 7450.6 GB. For details on the session information, see Appendix.

	rrpack (Chen et al.)		rrda (Yoshioka et al.)	
p, q	Mean (sec.)	SE	Mean (sec.)	SE
10^2	0.01	0.00	0.01	0.00
10^3	3.42	0.01	0.03	0.00
10^4	N/A	N/A	0.16	0.00
10^5	N/A	N/A	4.37	0.00
10^6	N/A	N/A	45.66	0.01

Table 4: Scenario 3. The `rrda` evaluation for simulation data. The Latent Space Model and the Matrix Factorization Model were tested. The estimations of rank, MSPE, λ, and Computation time of parameter tuning for random-splitted samples (100 iterations), and tested on the test data by MSPE for 100 times. Comparison of model fit with various tuning parameters. CV (5-fold) is performed for the `rrda` model. The computation times given here are measured using sequential executions without parallel computing. Mean values are presented with standard error in the parentheses.

Method	AIC	BIC	GIC	BICP	GCV	StARS	rrda (CV)
Model 1: Latent Space Model							
Rank	89.00 (0.00)	5.00 (0.00)	5.00 (0.00)	5.00 (0.00)	5.00 (0.00)	89.00 (0.00)	5.00 (0.00)
λ	107 (0.82)	1.33e+05 (491.18)	4.1e+05 (1453.05)	3.31e+05 (1190.04)	2.85e+04 (101.33)	107 (0.82)	250 (2.41)
MSPE	116.89 (0.38)	450.74 (6.28)	510.44 (7.26)	502.86 (7.12)	278.75 (3.39)	116.89 (0.38)	**100.23** (0.32)
Computation Time (sec)	0.395 (0.005)	0.389 (0.004)	0.394 (0.005)	0.389 (0.004)	0.389 (0.004)	2.21 (0.013)	3.66 (0.009)
Model 2: Matrix Factorization Model							
Rank	89.00 (0.00)	5.00 (0.00)	5.00 (0.00)	5.00 (0.00)	5.00 (0.00)	89.00 (0.00)	4.93 (0.03)
λ	10.5 (0.10)	1.27e+04 (45.18)	4e+04 (133.68)	3.21e+04 (111.67)	2.72e+03 (10.00)	10.5 (0.10)	18.9 (0.82)
MSPE	90.92 (0.87)	116.89 (1.55)	117.89 (1.57)	117.77 (1.57)	112.31 (1.46)	90.92 (0.87)	**80.55** (0.88)
Computation Time (sec)	0.389 (0.005)	0.392 (0.007)	0.39 (0.006)	0.397 (0.008)	0.429 (0.011)	2.22 (0.017)	3.63 (0.013)

Table 5: Scenario 3. Parameter selections for Model 1 (Latent Space Model) and Model 2 (the Matrix Factorization Model). Random-Splitting (100 iterations) for the following rank and lambda parameters. The parameters were selected as follows: CV min—the rank corresponding to the smallest MSE; Rank.1se—the rank selected by the one-standard-error rule; λ.1se—the regularization parameter λ selected by the one-standard-error rule; Full Rank—the ridge regression model without a rank constraint; $\lambda 0$—the reduced-rank model without a ridge penalty; Full Rank+$\lambda 0$—the multivariate model without a ridge penalty or rank constraint. Mean values are presented with standard error in the parentheses.

	CV min	Rank.1se	λ.1se	Full Rank	$\lambda 0$	Full Rank+$\lambda 0$
Model 1: Latent Space Model						
Rank	5.00 (0.00)	5.00 (0.00)	5.00 (0.00)	90.00 (0.00)	5.00 (0.00)	90.00 (0.00)
λ	250.19 (2.41)	250.19 (2.41)	757.13 (11.20)	250.19 (2.41)	0.00 (0.00)	0.00 (0.00)
MSPE	**100.23** (0.32)	**100.23** (0.32)	100.80 (0.33)	107.40 (0.37)	103.58 (0.35)	178.26 (0.69)
Model 2: Matrix Factorization Model						
Rank	4.93 (0.03)	4.10 (0.05)	4.93 (0.03)	90.00 (0.00)	4.93 (0.03)	90.00 (0.00)
λ	18.89 (0.82)	18.89 (0.82)	99.04 (3.74)	18.89 (0.82)	0.00 (0.00)	0.00 (0.00)
MSPE	**80.55** (0.88)	83.30 (1.03)	84.65 (1.01)	89.57 (0.89)	84.04 (0.95)	99.18 (0.94)

3.2 Application Results

Across diverse high-dimensional applications, we evaluated `rrda` in terms of predictive performance and computational scalability. The following case studies illustrate the motivation behind each analysis, how our `rrda` approach compares with existing approaches, and the biological insights it enables.

Breast Cancer data

First, we used a breast cancer dataset ($n = 89$) with gene expression as explanatory variables and CNV profiles as responses (Witten et al., 2009). Two versions were analyzed: a chromosome 13 subset ($X \in \mathbb{R}^{89\times 319},\ Y \in \mathbb{R}^{89\times 58}$) and a whole-genome version ($X \in \mathbb{R}^{89\times 19,672},\ Y \in \mathbb{R}^{89\times 2,149}$).

Following the simulation, we evaluated MSPE in this data, comparing classical criteria methods (Table 2S), StARS-RRR (Wen et al., 2023), and our efficient `rrda` method. To assess performance stability, we performed a random-splitting procedure 100 times, partitioning the dataset into a training set ($n_{\text{train}} = 79$) and a test set ($n_{\text{test}} = 10$). In our `rrda` formula, the CV was performed within the training data for parameter tuning.

We first used the chromosome 13 subset ($p = 319,\ q = 58$) as a small-scale example. The `rrda` successfully selected both the regularization parameter λ and the model rank (Figure 2, Table 6), achieving the lowest MSPE among competing methods. Although classical criteria (e.g., AIC: 0.07 s) were faster, `rrda` (2.13 s) was substantially more efficient than StARS-RRR (32.34 s), enabling a broader search over candidate parameters and better cross-validation performance. Unlike StARS, which first selects the rank and then determines λ, `rrda` jointly optimizes both λ and the rank. This joint selection contributes to `rrda`'s superior search efficiency and improved predictive performance.

We then applied `rrda` to the whole genome dataset ($p = 19{,}672,\ q = 2{,}149$) to assess its scalability. It again achieved the lowest MSPE and identified an intermediate rank (35), contrasting with the higher ranks selected by AIC/GCV and lower ranks by BIC/GIC. Despite the increased dimensionality, `rrda` completed the analysis in 49.65 seconds, much faster than StARS-RRR (368.12 s). The computation time of `rrda` was longer than that of classic methods such as AIC in the single-core setting. However, `rrda` was also evaluated in a parallel computing environment. In parallel setting, the computation for whole-genome breast cancer data was even faster than that of the classic methods (Table 6S).

Since our efficient `rrda` method is designed for high-dimensional data and enables integration of information across the genome, we further investigated its performance at the chromosome level. In Figure 3, we compared two approaches: the Chr–Chr model, where training and testing were performed on each chromosome separately, and the All–Chr model, where the model was trained on the whole genome and tested on individual chromosomes. The results indicated that both approaches performed comparably. Notably, the All–Chr model demonstrated the ability to capture diverse patterns across chromosomes, even without explicitly using chromosome labels (Figure 2S).

Soybean Multi-omics data

We next demonstrate the performance of our efficient `rrda` method using the soybean multi-omics dataset (Dang et al., 2023, *BioRxiv*). In this dataset, $X \in \mathbb{R}^{179\times 4771}$ corresponds to the microbiome data, and $Y \in \mathbb{R}^{179\times 253}$ represents the metabolome data. Figure 4 indicates that `rrda` effectively selected both the regularization parameter λ and the rank for this dataset.

We conducted a random-splitting procedure 100 times. In each split, the dataset was randomly divided into a training set of size $n_{\text{train}} = 159$ and a test set of size $n_{\text{test}} = 20$. The estimated parameters and MSPE are summarized in Table 7. AIC and StARS selected the highest rank, while BIC, GIC, and BICP tended to select smaller ranks. The `rrda` method selected an intermediate rank of 11 and achieved the lowest MSPE among all tested methods, suggesting favorable predictive performance.

Consistent with the clustering-based I-SVVS model proposed by Dang et al. (2023, bioRxiv), our `rrda` approach identified key microbial and metabolic variables associated with plant-microbe community (Figure 5). These included the *Chitinophagaceae* family within the order *Chitinophagales*, which is known to contribute to fungal suppression through chitinase activity (Venessa et al., 2013, Victor et al. 2019), and *Hyphomicrobiaceae* familiy from the order *Rhizobiales* is well known for its nitrogen-fixing capabilities (Javad et al., 2015; Qin et al., 2020). Among the metabolites, glutamic acid and aspartic acid were selected, consistent with previous findings highlighting their roles in promoting beneficial microbial recruitment and enhancing plant health (Elisa et al. 2015, Hongwei et al. 2019).

In addition to the findings of Dang et al. (2023, bioRxiv), our efficient `rrda` also identified the other microbes as well. The order of *Gaiellales*, which is involved in rhizosphere nutrient cycling (Wang et al., 2024). Furthermore, daidzein—a soybean-derived secondary metabolite known to function as a key signaling molecule secreted by soybean roots to initiate symbiosis with rhizobia (Sugiyama et al., 2017)—was also selected.

Methylation and Expression data

To validate the efficiency of `rrda` in a high-dimensional context, we demonstrate the performance using the methylation-expression dataset (Ruiz-Arenas and González, 2017). In this dataset, $X \in \mathbb{R}^{20\times 392,277}$ corresponds to the DNA methylation data, and $Y \in \mathbb{R}^{20\times 67,528}$ represents the gene expression data.

The data were split into training ($n = 19$) and test ($n = 1$) sets using a leave-one-out approach. Parameter tuning was performed for each training set via cross-validation. As a result, our efficient `rrda` method successfully selected both the regularization parameter λ and the model rank (Figure 6). The proposed `rrda` approach performs well on this highly dimensional dataset due to its scalability. In contrast, conventional methods (e.g., `rrpack`, `StARS`) were unable to handle matrices of this size, even on a computational server. Therefore, our approach was the only feasible method for carrying out the analysis. Details of the computational resources used are provided in the Appendix (Table 1S).

Notably, a `rrda` analysis integrating both DNA methylation and gene expression data

simultaneously has not been previously conducted on this dataset, likely due to the extreme high dimensionality of the combined matrix.

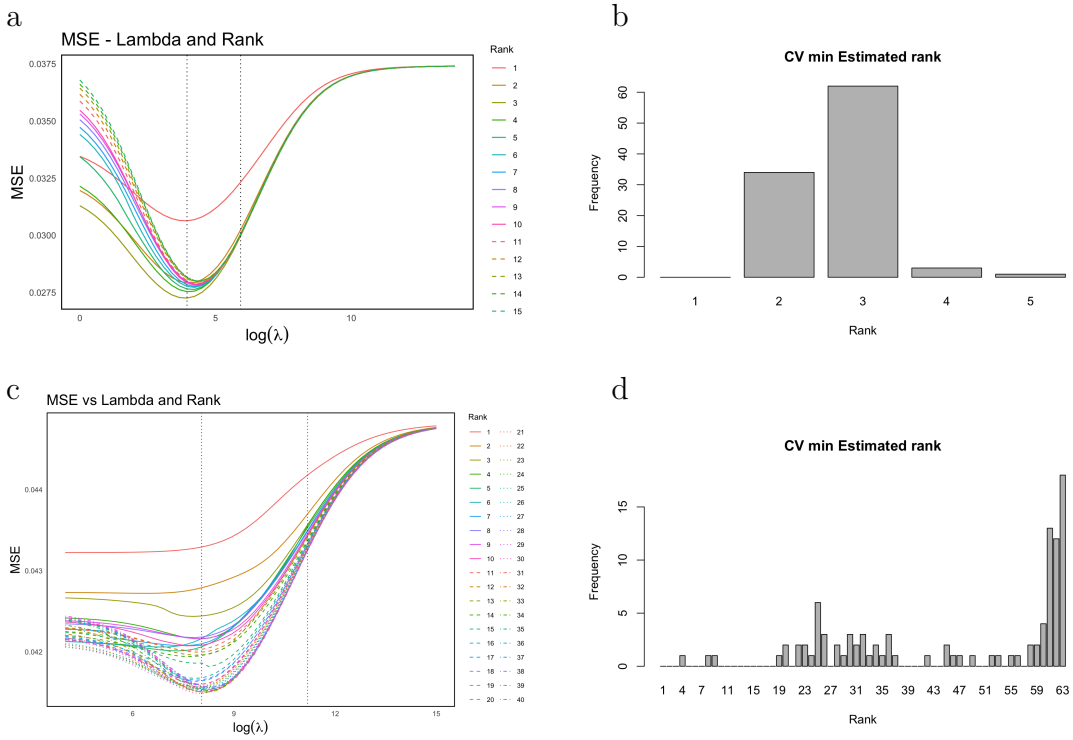

Figure 2: Regularization path and rank estimation in breast cancer data. For the chromosome 13 subset, the gene expression (p=319) and DNA copy number (q = 58), and the number of patients (n = 89). For the Whole Genome, p=19672 and q=2149. The data was randomly split into training data (n = 79) and test data (n = 10). The best rank and ridge penalty λ was estimated on the training data via CV (5-folds). For chromosome 13 subset, (a) Plot for parameter tuning (b) The estimated rank for each iteration. For the whole genome, (c) Plot for parameter tuning (d) The estimated rank for each iteration

Table 6: The `rrda` evaluation in breast cancer data. Rank estimations for full data. The estimations of rank, MSPE, λ, and Computation time of parameter tuning for random-splitted samples (100 iterations), and tested on the test data by MSPE for 100 times.. Comparison of model fit with various tuning parameters. CV (5-fold) is performed for the rrda model. The computation times given here are measured using sequential executions without parallel computing. Mean values are presented with standard error in the parentheses.

Method	AIC	BIC	GIC	BICP	GCV	StARS	rrda (CV)
The Chromosome 13 subset							
Rank	56.95 (0.0219)	37.26 (2.6433)	1.19 (0.0394)	1.82 (0.0386)	24.47 (0.1283)	2.40 (0.0865)	2.71 (0.0574)
λ	7.36e-04 (2.32e-05)	18.93 (2.6532)	344.48 (8.6255)	196.03 (6.5157)	0.296 (4.68e-03)	95.28 (10.73)	53.74 (1.5359)
MSPE	4.37 (0.1015)	3.79 (0.1367)	3.04 (0.1111)	2.86 (0.1126)	4.20 (0.1001)	2.83 (0.0976)	**2.70** (0.0881)
Computation Time (sec)	0.065 (0.0012)	0.062 (0.0012)	0.067 (0.0023)	0.064 (0.0019)	0.065 (0.0019)	32.34 (0.1995)	2.130 (0.0053)
Whole Genome (Chromosome 1-23)							
Rank	78.00 (0)	1.07 (0.0256)	0.00 (0)	0.00 (0)	67.86 (0.1564)	78.00 (0)	46.71 (1.7336)
λ	10.37 (0.2204)	23900.71 (287.6330)	89743.22 (487.3246)	89743.22 (487.3246)	41.11 (0.4320)	10.37 (0.2204)	3619.11 (104.6320)
MSPE	4.21 (0.0823)	4.44 (0.0952)	4.54 (0.0967)	4.54 (0.0967)	4.20 (0.0824)	4.21 (0.0823)	**4.11** (0.0868)
Computation Time (sec)	24.02 (0.4519)	20.61 (0.0235)	20.59 (0.0229)	20.54 (0.0208)	63.22 (18.8047)	368.12 (0.2619)	49.65 (0.0595)

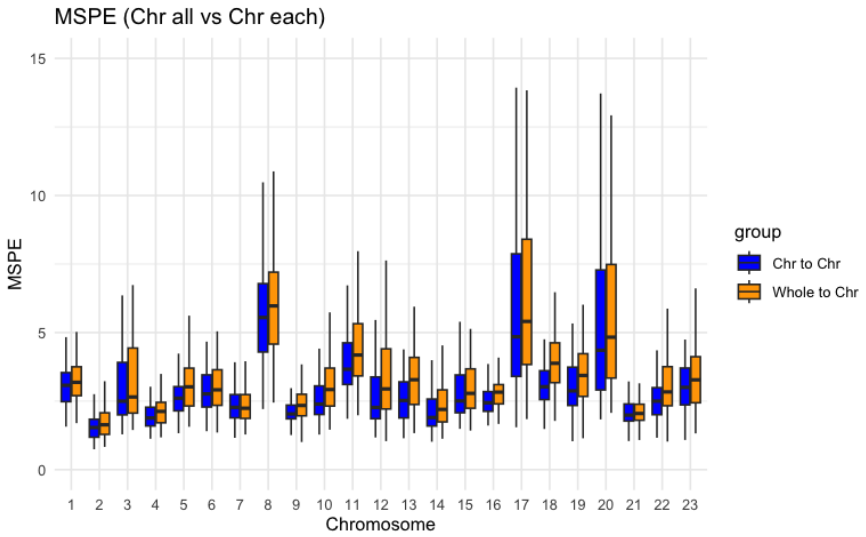

Figure 3: Breast Cancer: Boxplot for MSPE for each chromosome. "Chr to Chr" vs "Whole to Chr" model. In the "Chr to Chr" model, the model was trained and tested on each chromosome. In the "Whole to Chr" model, the model was trained on all chromosomes and tested on each chromosome.

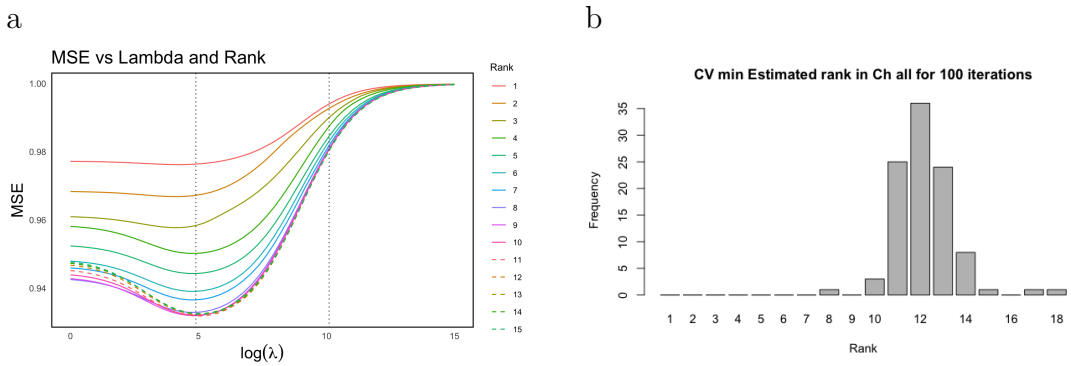

Figure 4: Regularization path and rank estimation in soybean microbiome and metabolome data, where it includes the microbe taxonomic traits (p=4771) and metabolome (q=253) for the number of samples (n=179). The data was randomly split into training data (n = 159) and test data (n = 20). The best rank and ridge penalty λ was estimated on the training data via CV (5-folds). (a) Plot for parameter tuning (b) The estimated rank for each iteration.

Table 7: The `rrda` evaluation in Soybean dataset. The estimations of rank, MSPE, λ, and Computation time for parameter tuning with random-splitting samples (100 iterations), and tested on the test data by MSPE for 100 times.. Comparison of model fit with various tuning parameters. CV (5-fold) is performed for the rrda model. The rank is estimated to be 0 when no variables are selected in the model. The computation times given here are measured using sequential executions without parallel computing. Mean values are presented with standard error in the parentheses.

Method	AIC	BIC	GIC	BICP	GCV	StARS	rrda (CV)
Performed on Random-Splitting Data							
Rank	158.00 (0.0000)	3.00 (0.0000)	0.00 (0.0000)	0.00 (0.0000)	43.44 (0.3846)	158.00 (0.0000)	11.24 (0.1327)
λ	4.76 (0.0863)	1.01e+05 (500.922)	3.66e+05 (2607.586)	3.66e+05 (2607.586)	3.48e+03 (41.9056)	4.76 (0.0863)	3.09e+02 (9.7968)
MSPE	100.47 (1.6038)	100.51 (1.5455)	100.92 (1.5495)	100.92 (1.5495)	95.52 (1.5379)	100.47 (1.6038)	**94.15** (1.5572)
Computation Time (sec)	1.326 (0.0060)	1.123 (0.0056)	1.123 (0.0055)	1.107 (0.0054)	1.151 (0.0059)	27.957 (0.0373)	10.991 (0.0220)

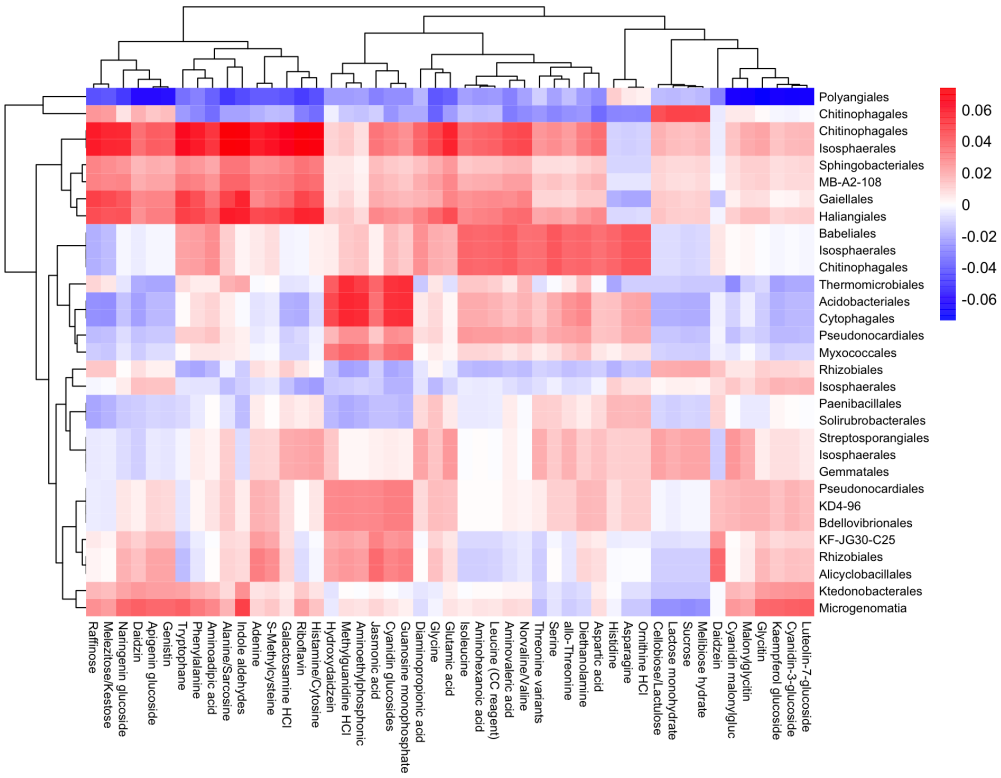

Figure 5: Heatmap for coefficients of microbiome and metabolome in soybean dataset. The top features were selected after parameter tuning and fitting.

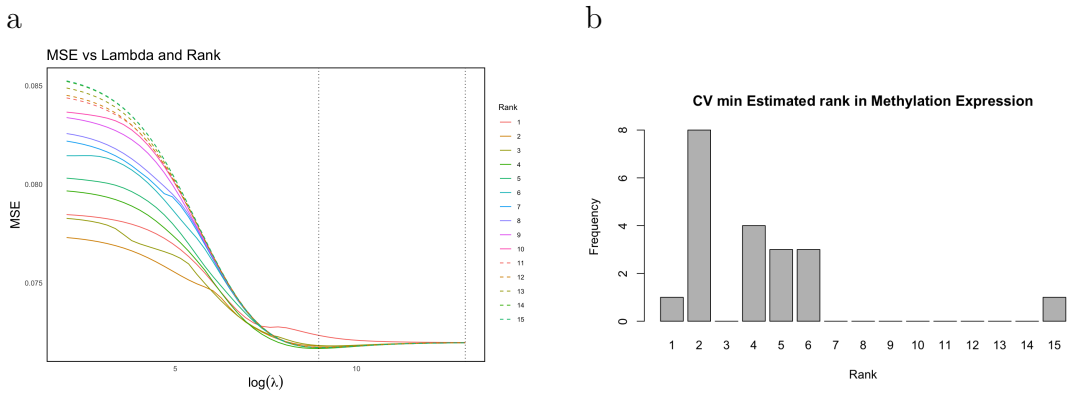

Figure 6: Regularization path and rank estimation in methylation and gene expression, where it includes DNA methylation (p = 392277), gene expression (q = 67528) for the number of patients (n = 20). The data was split into training data (n = 1) and test data (n = 19) as the leave-one-out approach. The best rank and ridge penalty λ was estimated on the training data via CV (5-folds), and tested on the test data. (a) Plot for parameter tuning (b) The estimated rank for each splitting iteration

4 Discussion

4.1 Efficient Computation and Storing for High-Dimensional Data

Our efficient `rrda` approach demonstrates a significant improvement in computational efficiency relative to conventional ones, especially when applied to high-dimensional data. Our approach enables cross-validation (CV) for parameter tuning, resulting in the most accurate predictions across various scenarios through simulation and application. This advantage is significant when dealing with high-dimensional data, where the number of variables (p and q) exceeds a lot the number of samples (n), a challenge commonly encountered in fields like genomics and other omics studies. Traditional methods often require the inversion of large covariance matrices, which becomes increasingly time-consuming as data size grows. Our method addresses these challenges by incorporating an efficient regularization path into the ridge RDA process, choosing the λ and rank and rank (r) at the same time, substantially enabling extremely high-dimensional computation.

4.2 Choice of Dimensionality r and Ridge Penalty λ

In high-dimensional omics studies, encountering many canonical axes is common (Legendre et al., 2011). Through simulation and application, our `rrda` method has demonstrated its ability to efficiently handle high-dimensional omics data and identify the optimal robust parameters r (dimensionality) and λ (ridge penalty) that minimize the mean squared error (MSE). Compared to traditional methods, our approach provides more consistent and accurate parameter estimates, particularly in high-dimensional settings. We also applied the "one standard error rule" to improve the selection of robust parameters r and λ for a more parsimonious modeling. However, the results indicated that this rule can be subjective to the characteristics of the data. For instance, it may lead to the selection of overly simplified models in certain cases. In our method, we provide users with flexible options to balance precision and robustness based on their specific preferences or analytical goals. Another promising approach, such as permutation testing, could be explored for further refining parameter selection and warrants further investigation.

4.3 Comparison with Existing Ridge RDA Approaches

Our results highlight the practical advantages of `rrda` over existing ridge RDA approaches, particularly in high-dimensional contexts. While ridge RDA is commonly applied to such data, existing methods do not fully utilize singular value decomposition (SVD) to obtain closed-form solutions. For example, Sparse RDA (Csala et al., 2017) depends on the covariance structure of X, which makes its computational performance sensitive to sparsity and limits its scalability. In contrast, `rrda` efficiently computes the entire regularization path by leveraging SVD, enabling a closed-form solution that is not only faster but also more scalable to ultra high-dimensional settings. Additionally, our approach reduces memory demands by avoiding explicit construction of large covariance matrices, making it especially suitable for

datasets with extreme $p, q \gg n$ dimensions. These properties collectively contribute to the improved speed, scalability, and storage efficiency.

4.4 Practical Application to Omic Studies

In our application to breast cancer data, The results illustrate that the All-Chr model is capable of learning from genome-wide information without requiring the segmentation of individual chromosomes (Figure 3). Remarkably, it still captures chromosome-specific patterns, as shown in Figure 2S. This highlights the model's ability to extract biologically meaningful structure even from broadly defined genomic data, demonstrating its robustness in practical scenarios where detailed genomic mapping may not be available. Furthermore, our `rrda` approach proves uniquely capable in extremely high-dimensional settings—such as the methylation-expression dataset—where existing methods fail due to computational or statistical limitations. Traditional techniques are often infeasible when both the number of features (p and q) far exceeds the number of samples (n). In contrast, `rrda` scales efficiently and maintains predictive performance, enabling integrated analysis across different omics layers. These results emphasize the potential of `rrda` as a powerful tool for high-dimensional genomic and multi-omics studies. It offers a scalable and effective solution for complex datasets that have traditionally posed challenges for conventional methods.

5 Conclusion

In this article, we proposed a novel method for ridge RDA tailored to high-dimensional omics data, particularly addressing the challenges posed by the "large p, small n" problem. By incorporating an efficient regularization path within the ridge RDA framework, our approach circumvents the direct inversion of the covariance matrix, making it computationally feasible even for large and complex datasets commonly found in omics science. We demonstrated the efficiency of our method through simulation studies and applications, highlighting its superior performance, especially in scenarios involving large matrices. Our method provides flexibility in determining the best rank and ridge penalty λ through CV, enhancing its practicality for researchers working with high-dimensional data. The implementation of this method in the R package `rrda` provides a valuable tool for the genomics and other omics community, offering both theoretical foundations and practical solutions for modern multivariate analysis. Future work could explore further optimizations to delve deeper into the selection of dimensionality in the analysis.

Acknowledgements

We would like to thank Dr. Yushiro Fuji and Dr. Masami Hirai Yokota (RIKEN, Japan) for metabolomic analysis, and Dr. Yasunori Ichihashi (RIKEN, Japan) for microbial analysis. We extend our sincere gratitude to the following researchers for their invaluable contribu-

tions to data collection as part of the CREST project; Akito Kaga (NARO, Japan), Yusuke Toda, Hideki Takanashi, Yoshihiro Ohmori, Motoyuki Ishimori, Toru Fujiwara, (University of Tokyo, Japan), Megumi Narukawa, Erika Usui, Kie Kumaishi, Takumi Satou, Shungo Kobori (RIKEN, Japan), Hirokazu Takahashi, Mikio Nakazono (Nagoya University, Japan), Mai Tsuda (Tsukuba University, Japan), Yuji Yamasaki and Hisashi Tsujimoto (Tottori University, Japan). We thank Yutaka Yamada (RIKEN CSRS) for informatics support. We are grateful to the technical staff at the Arid Land Research Center, Tottori University, and Izumi Higashida.

Funding

This study was supported by the Japan Society for the Promotion of Science (JSPS) KAKENHI Grant Number JP 23KJ0506 and Bourses du Gouvernement Français (BGF). This work was also supported by Japan Science and Technology (JST) Core Research for Evolutional Science and Technology (CREST; JPMJCR16O2), JSPS KAKENHI (JP22K21352), and the JST ALCA-Next Program 751 (JPMJAN23D1), Japan.

References

Akaike, H. (1974), "A New Look at the Statistical Model Identification," IEEE Transactions on Automatic Control, 19, 716–723. [975]

Anderson, T.W., 1951. Estimating linear restrictions on regression coe±cients for multivariate normal distributions. Annals of Mathematical Statistics, 22, 327-351.

Buescher, J. M., and Driggers, E. M. (2016). Integration of omics: More than the sum of its parts. Cancer and Metabolism, 4(1), 4.

Cailliez, F., Pages, J.P., 1976. Introduction a l'Analyse des Donnees, Societe de Mathematique Appliquees et de Sciences Humaines, Paris.

Canhong Wen, Qin Wang and Yuan Jiang (2023) Stability Approach to Regularization Selection for Reduced-Rank Regression, Journal of Computational and Graphical Statistics, 32:3, 974-984

Chen K and Wang W (2022). rrpack: Reduced-Rank Regression. R package version 0.1-13, ¡https://CRAN.R-project.org/package=rrpack¿

Dang, T., Fuji, Y., Kumaishi, K., Usui, E., Kobori, S., Sato, T., Toda, Y., Sakurai, K., Yamasaki, Y., Tsujimoto, H. and Hirai, M.Y., 2023. An integrative framework of stochastic variational variable selection for joint analysis of multi-omics microbiome data. bioRxiv, pp.2023-08.

Elisa Gamalero and Bernard R. Glick. Bacterial modulation of plant ethylene levels. Plant Physiology, 169(1):13–22, 2015.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010). Regularization Paths for Generalized

Linear Models via Coordinate Descent. *Journal of Statistical Software*, *33*(1), 1–22.

Greenacre, M.J., 1984. Theory and Applications of Correspondence Analysis, Academic Press, London.

Hastie, T., Tibshirani, R., and Friedman, J. The Elements of Statistical Learning. Springer Series in Statistics Springer (2001) New York Inc., New York, NY, USA

Henrik Bengtsson, A Unifying Framework for Parallel and Distributed Processing in R using Futures, The R Journal (2021) 13:2, pages 208-227, doi:10.32614/RJ-2021-048

Hilafu, H., Safo, S.E., and Haine, L. Sparse reduced-rank regression for integrating omics data. BMC Bioinformatics 21, 283 (2020).

Hotelling, H., Relations Between Two Sets ofVariates. Biometrika, 28, 321-377 (1936).

Huang, S., Chaudhary, K., and Garmire, L. X. (2017). More is better: Recent progress in multi-omics data integration methods. Frontiers in Genetics, 8, 84.

Javad Hamedi and Fatemeh Mohammadipanah. Biotechnological application and taxonomical distribution of plant growth promoting actinobacteria. *Journal of Industrial Microbiology and Biotechnology*, 42(2):157–171, 2015.

Legendre P, Oksanen J, ter Braak CJF (2011) Testing the significance of canonical axes in redundancy analysis. Methods Ecol Evol 2(3):269–277.

Liu, H., Khan, M. Y., Carvalhais, L. C., Delgado-Baquerizo, M., Yan, L., Crawford, M., Dennis, P. G., Singh, B., and Schenk, P. M. (2019). Soil amendments with ethylene precursor alleviate negative impacts of salinity on soil microbial properties and productivity. Scientific Reports, 9(1):6892.

Reinsel, G.C., Velu, R.P., 1998. Multivariate Reduced-rank Regression, Springer, New York

Mukherjee, A. and Zhu, J. (2011). Reduced rank ridge regression and its kernel extensions. Statistical analysis and data mining: the ASA data science journal, 4(6):612–622.

Qin Han, Qun Ma, Yong Chen, Bing Tian, Lanxi Xu, Yang Bai, Wenfeng Chen, and Xia Li. Variation in rhizosphere microbial communities and its association with the symbiotic efficiency of rhizobia in soybean. The ISME Journal, 14(8):1915–1928, 2020.

Qiu Y, Mei J (2024). RSpectra: Solvers for Large-Scale Eigenvalue and SVD Problems. R package version 0.16-2,

Rao, C. R. (1964) The use and interpretation of principal component analysis in applied research. Sankhya, A 26, 329–359.

R Core Team (2023). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Ruiz-Arenas, C., González, J.R. Redundancy analysis allows improved detection of methylation changes in large genomic regions. BMC Bioinformatics 18, 553 (2017).

Ruiz-Arenas C, Gonzalez J (2024). MEAL: Perform methylation analysis. R package version 1.34.0. MEAL: Perform methylation analysis. doi:10.18129/B9.bioc.MEAL

Schwarz, G. (1978), "Estimating the Dimension of a Model," Annals of Statistics, 6, 461–464. [975]

Sugiyama, A., Ueda, Y., Zushi, T., Takase, H., Yazaki, K. 2014. Changes in the bacterial community of soybean rhizospheres during growth in the field. PLOS ONE, 9:e100709

Takane, Y and Hunter M A, (2001). Constrained Principal Component Analysis: A Comprehensive Theory.

Takane, Y. and Hwang, H. (2005) On a test of dimensionality in redundancy analysis. Psychometrika, 70, 271–281.

Takane, Y and Hwang, H. (2007) Regularized linear and kernel redundancy analysis", *Comput. Statist. Data Anal.*, vol. 52, no. 1, pp. 394-405,

Van den Wollenberg, A.L., 1977. Redundancy analysis: An alternative for canonical analysis. Psychometrika, 42, 207-219.

Vanessa L. Bailey, Sarah J. Fansler, James C. Stegen, and Lee Ann McCue. Linking microbial community structure to beta-glucosidic function in soil aggregates. The ISME Journal, 7(10):2044–2053, 2013.

Víctor J. Carrión, Juan Perez-Jaramillo, Viviane Cordovez, Vittorio Tracanna, Mattias De Hollander, Daniel Ruiz-Buck, Lucas W. Mendes, Wilfred F.J. van Ijcken, Ruth Gomez-Exposito, Somayah S. Elsayed, et al. Pathogen-induced activation of disease-suppressive functions in the endophytic root microbiome. *Science*, 366(6465):606–612, 2019.

Walker S (2014). pr2: Predictive R-squared. R package version 0.1/r212, ¡https://R-Forge.R-project.org/projects/multitable/¿.

Wang, M., Ge, A.H., Ma, X. et al. Dynamic root microbiome sustains soybean productivity under unbalanced fertilization. *Nature Communications*, 15:1668, 2024.

Wen, C., Wang, Q., and Jiang, Y. (2022). Stability Approach to Regularization Selection for Reduced-Rank Regression. Journal of Computational and Graphical Statistics, 32(3), 974–984.

Wickham H. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York; 2016.

Witten D, Tibshirani R, Gross S, Narasimhan B (2024). PMA: Penalized Multivariate Analysis. R package version 1.2-4,