Journal Title Here, 2025, 1–26
doi: DOI HERE
Advance Access Publication Date: Day Month Year
Paper

Ridge Redundancy Analysis for High-Dimensional Omics Data

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

[1]Department of Agricultural and Environmental Biology, Graduate School of Agricultural and Life Science, The University of Tokyo, Tokyo, 113-0032, Japan, [2]Université Paris-Saclay, AgroParisTech, INRAE, UMR MIA Paris-Saclay, Palaiseau, 91120, France and [3]Université Paris-Saclay, INRAE, CNRS, AgroParisTech, GQE - Le Moulon, Gif-sur-Yvette, 91190, France

*Corresponding author. tristan.mary-huard@agroparistech.fr

FOR PUBLISHER ONLY Received on Date Month Year; revised on Date Month Year; accepted on Date Month Year

Abstract

Motivation: Redundancy Analysis (RDA) is a popular reduced rank regression 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. Despite these advantages, 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 jointly select the rank and the regularization parameter that both 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 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 R package `rrda` is available on CRAN: `https://CRAN.R-project.org/package=rrda`. Scripts and data used for the analysis are 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 and 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 $\text{rank}(B) \leq r$, ensuring dimensionality reduction. Rank restriction is beneficial as it efficiently captures and exploits redundant information, and facilitates the identification of dominant patterns in the data (Capblancq and Forester, 2021).

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 and Driggers, 2016). Notable applications include gene expression analysis (Wen et al., 2023), DNA methylation studies (Ruiz-Arenas and González, 2017), and metabolomic research (Hilafu

© The Author 2025. Published by Oxford University Press. All rights reserved. For permissions, please e-mail: journals.permissions@oup.com

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 and Hwang, 2007), also referred to as reduced-rank ridge regression (Mukherjee and Zhu, 2011), has been considered 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 (Wen et al., 2023).

In classical ridge regression, 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), a challenge commonly encountered in fields like genomics and other omics studies (Zhao et al., 2019). Compared to ridge regression, the ridge RDA approach raises new challenges, as both the number of predictor variables p and the number of response variables q may be large (e.g. $p, q > 10^5$), representing 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. 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`, publicly available on CRAN.

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. 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, metabolome, 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. The relation between X and Y is given by

$$Y = XB + E \tag{1}$$

where B is the matrix of regression coefficients, and E is an error matrix. The ridge RDA aims to estimate B under both rank and ridge restrictions. The corresponding optimization problem is

$$\hat{B}(\lambda, r) = \arg\min_B \left\{\|Y - XB\|^2 + \lambda\|B\|^2\right\} \quad u.c. \quad \text{rank}(B) \leq r$$

where $\|A\|^2 = \text{tr}(A'A)$ is the Frobenius norm and λ is the ridge regularization parameter. As shown in Takane and Hwang (2007), the previous optimization problem can be reformulated as

$$\begin{aligned} \hat{B}(\lambda, r) &= \arg\min_B \left\{\|Y\|^2_{Q_X(\lambda)} + \|\hat{B}(\lambda) - B\|^2_{X'X+\lambda P_{X'}}\right\} \\ &\quad \text{u.c.} \quad \text{rank}(B) \leq r \\ &= \arg\min_B \|\hat{B}(\lambda) - B\|^2_{X'X+\lambda P_{X'}} \\ &\quad \text{u.c.} \quad \text{rank}(B) \leq r \end{aligned} \tag{2}$$

where $\|A\|^2_M = \text{tr}\,(A'MA)$, $P_{X'} = X'\,(XX')^{-1}\,X$, and $Q_X(\lambda) = I - X\,(X'X + \lambda P_{X'})^{-1}\,X'$. Matrix $\hat{B}(\lambda) = (X'X + \lambda P_{X'})^{-1}\,X'Y$ represents the ridge estimator of B without rank restriction. The solution of (2) can be obtained by considering the Generalized Singular Value Decomposition (GSVD, Cailliez and Pages 1976; Greenacre 1984) of $\hat{B}(\lambda)$ with respect to row and column metric matrices $X'X + \lambda P_{X'}$ and I. See Appendix B for the definition of GSVD. This decomposition yields

$$\hat{B}(\lambda) = U_{\hat{B}(\lambda)} D_{\hat{B}(\lambda)} V'_{\hat{B}(\lambda)} \tag{3}$$

where $U_{\hat{B}(\lambda)}$ satisfies $U'_{\hat{B}(\lambda)}(X'X + \lambda P_{X'})U_{\hat{B}(\lambda)} = I$, $V_{\hat{B}}$ satisfies $V'_{\hat{B}(\lambda)} V_{\hat{B}(\lambda)} = I$, and $D_{\hat{B}(\lambda)}$ is a diagonal matrix. The ridge-penalized-r-rank estimate $\hat{B}(\lambda, r)$ is then obtained by retaining only the GSVD components corresponding to the first r dominant singular values:

$$\hat{B}(\lambda, r) = U^{[r]}_{\hat{B}(\lambda)} D^{[r]}_{\hat{B}(\lambda)} V^{[r]\prime}_{\hat{B}(\lambda)} \tag{4}$$

where $U^{[r]}_{\hat{B}(\lambda)}$ 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.

Computing the unconstrained ridge estimator requires the inversion of $X'X + \lambda P_{X'}$, which is a large matrix of size $p \times p$. Estimate $\hat{B}(\lambda, r)$ is also a large matrix (of dimensions $p \times q$), and presents both computational and storage difficulties. Lastly, model parameter tuning requires repeating the inference procedure for each combination of the ridge (λ) and rank (r) constraints.

2.2. Adaptation to the high-dimensional setting

We now focus on ridge RDA in high-dimensional settings where $p, q \gg n$. Similar to what is classically performed in the ridge regression setting (see Hastie et al. 2001, Section 18.3.5) we first perform a Singular Value Decomposition (SVD) of X as

$$X = UDV' \tag{5}$$

The different quantities appearing in Equation (2) can then be rewritten as follows:

$$\begin{aligned} \hat{B}(\lambda) &= VD\left(D^2 + \lambda I\right)^{-1} U'Y \\ X'X + \lambda P_{X'} &= V(D^2 + \lambda I)V' \end{aligned} \tag{6}$$

where we assume $\lambda > 0$. Note that this first SVD is performed only once, as it does not depend on parameters λ and r.

A second SVD is required for the computation of the GSVD of matrix $\hat{B}(\lambda)$. Indeed, one has

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

where $\hat{U}_\lambda, \hat{D}_\lambda \hat{V}_\lambda$ are the SVD components of matrix $M_\lambda = (D^2+\lambda I)^{-1/2}DU'Y$. As this matrix depends on λ, this second SVD must be computed for each specific value of λ. However, the computation of the matrix M_λ itself for different values of λ can be factorized as M_λ is the product of a diagonal matrix and a constant matrix $U'Y$ that does not depend on λ.

The final expression of the $\hat{B}(\lambda, r)$ is then

$$\hat{B}(\lambda, r) = V(D^2+\lambda I)^{-1/2}\hat{U}_\lambda^{[r]}\hat{D}_\lambda^{[r]}\hat{V}_\lambda^{[r]\prime} \tag{7}$$

To summarize, the proposed formulation i) leverages the SVD of X to avoid the direct inversion of the $X'X + \lambda P_{X'}$ matrix, which is of size $p \times p$, thereby significantly reducing computational cost, ii) reduces the computation of the GSVD that follows to the computation and storage of diagonal matrices $(D^2+\lambda I)^{-1/2}$ combined with the SVD of a $n \times q$ matrix M_λ, offering efficiency over handling the $p \times q$ matrix B. See Appendix B and F for details.

2.3. Efficient storage of the high-dimensional coefficient matrix

In omics studies, where both p and q may be large, the direct storage of the ridge RDA coefficient $p \times q$ matrix may become challenging or even infeasible. In the Methylation application of Section 3.2.3, one has $p = 392,277$ and $q = 67,528$, leading to a coefficient matrix with $> 10^{10}$ elements that would require a memory storage of ~ 100 GB. To address this issue, we propose an efficient storage method. From (4), define

$$F_i = U_{\hat{B}(\lambda)}^{[i]} D_{\hat{B}(\lambda)}^{[i]}, \quad G_i = V_{\hat{B}(\lambda)}^{[i]}, \quad i = 1, \ldots, r.$$

One has

$$\hat{B}(\lambda, r) = \sum_{i=1}^{r} F_i G_i' \tag{8}$$

Moreover, the computation of any new prediction from the ridge RDA estimator can be obtained as

$$X_{new}\hat{B}(\lambda, r) = \sum_{i=1}^{r} (X_{new}F_i)G_i' \tag{9}$$

which does not require the computation of $\hat{B}(\lambda, r)$. Consequently, the storage of the coefficient matrix can be replaced by the storage of the two matrices $F = [F_1, ..., F_r]$ and $G = [G_1, ..., G_r]$ of size $p \times r$ and $q \times r$, respectively. This alternative representation drastically reduces the memory usage, moving, e.g., from 100Go to 175Mo in the methylation application when considering $r = 100$.

The computation and storage of the $\hat{B}(\lambda, r)$ and of its usage for prediction are implemented in the R package `rrda` available on CRAN. The `rrda.fit` function computes the coefficient matrix either in a component form, i.e., using the F and G matrices, or directly as a full $p \times q$ matrix, for all combinations of λ and r. The `rrda.pred` function performs prediction using $\hat{B}(\lambda, r)$ directly or following (9) whenever the component form is used.

In addition, the low-rank representation of the coefficient matrix also enables the extraction of compact submatrices for visualization by selecting variables with the largest row-wise ℓ_2-norm of F and G on the r-dimensional projections, as presented in the following applications. See Appendix E for details.

2.4. Parameter tuning

Based on the computational and storage shortcuts introduced in the previous section, we consider V-fold CV method to jointly select r and λ. The criterion to optimize is

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

where (v) stands for the vth subset of the data, and $\hat{B}(\lambda, r)^{(-v)}$ is the ridge RDA coefficient matrix estimated from all subsets except the vth one.

In practice, a sequence of values for λ is considered within a range $[\lambda_{\min}, \lambda_{\max}]$ that must be provided by the user. Following the choice of the `glmnet` procedure (Friedman et al. 2010), the by-default value of the interval bounds can be set at

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

and $\lambda_{min} = 10^{-4}\lambda_{max}$, respectively.

As the computational cost of (10) is $\mathcal{O}(npq)$, a random subsampling of the variables sampling method was implemented whenever $\max(p, q) > m$, with a default value of $m = 10^3$. In our experiments, this sampling approach significantly reduced the computation time without hampering accuracy (Figure 1S). Furthermore, we extended the `glmnet` "one-standard-error" rule (Friedman et al., 2010) to the selection of both the ridge penalty coefficient λ and the rank parameter r, selecting the largest λ and smallest r within one-standard-error of the minimum mean square error (MSE).

Selecting the optimal (r,λ) combination through CV can be performed using the `rrda.cv` function of the R package `rrda` that also benefits from parallel computing, see Appendix F.

Alternatives to CV for model selection include information criteria. In what follows, we benchmarked the CV approach to five criteria that have been proposed in the context of reduced rank regression: Akaike Information Criterion (AIC; Akaike 1974), Bayesian Information Criterion (BIC; Schwarz 1978), Generalized Information Criterion (GIC; Fan and Tang 2013), Generalized Cross-Validation (GCV; Golub et al. 1979), and Bayesian Information Criterion with Penalty (BICP; An et al. 2008). Although GCV was originally introduced as an approximation to CV, here we regard it as a type of information criterion. The mathematical expressions of the different criteria are provided in Table 1.

2.5. Competing methods

Two competing reduced-rank regression approaches were considered. The first is the Adaptive Nuclear Norm procedure (ANN, Chen et al., 2013). The ANN implements rank reduction via singular vector sparsity by thresholding the singular values of B, yielding a set of candidate ranks $\mathcal{R} \subseteq \{1, \ldots, \min\{n, p, q\}\}$. The optimal rank is then selected by minimizing an information criterion over $\mathcal{R}$. The ANN procedure is implemented in the `rrr` function of the R package `rrpack` (version 0.1-13; Chen and Wang 2022). The choice of the regularization parameter can be performed

using the information criteria of Table 1, or alternatively using the Stability Approach to Regularization Selection for Reduced-Rank Regression (`StARS`, Wen et al., 2023). In our benchmark analysis, both information criteria and `StARS` were used for the ANN parameter tuning, see Appendix D for details. Hereafter, those criteria-based ANN are denoted as `ANN.AIC`, `ANN.BIC`, etc.

Secondly, the sparse RDA (`sRDA`, Csala et al., 2017) was considered. The `sRDA` proposes Redundancy Analysis with Elastic Net penalty to obtain parsimonious solutions in high-dimensional omics datasets. It can perform ridge RDA when the Elastic Net penalty is tuned toward its ℓ_2 component. `sRDA` defines the number of latent variables within a partial least squares regression framework, and is therefore not directly compatible with `rrda` in terms of formulation. Although not fully equivalent, it serves as a recent RDA method for comparison in terms of predictive accuracy and computational performance, as presented in later sections. See Appendix C for details of `sRDA`.

2.6. Simulation data

In this subsection, we consider two data generation models to illustrate the performance of Ridge RDA.

Model 1: Latent space model

In Model 1, 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 also independently drawn from standard 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:

$$X = H\theta_x + \sigma_x E_x, \qquad Y = H\theta_y + \sigma_y E_y$$

where

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

The scaling parameters σ_x and σ_y are chosen afterwards in order to ensure some chosen signal-to-noise ratio s_x, and s_y, i.e.,

$$\sigma_x = \sqrt{\frac{\|H\theta_x\|^2}{\|E_x\|^2 \cdot s_x}}, \quad \sigma_y = \sqrt{\frac{\|H\theta_y\|^2}{\|E_y\|^2 \cdot s_y}}.$$

We tested the signal-to-noise (s2n) values $s_x = s_y = 0.1, 1, 10$, the results presented in the following sections corresponding to $s_x = s_y = 1$.

This first setting does not explicitly rely on a computational matrix B to which the ridge RDA estimator should be compared.

Model 2: Matrix factorization model

Let $X \in \mathbb{R}^{n\times p}$ be the predictor matrix, with entries sampled from a standard normal distribution:

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

and $A \in \mathbb{R}^{p\times k}$ and $C \in \mathbb{R}^{k\times q}$ be two coefficient matrices, with k the true latent dimension. 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, and the coefficient matrix B is constructed as:

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

Lastly, the observed response matrix $Y \in \mathbb{R}^{n\times q}$ is generated as

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

where the standard error σ is chosen to ensure a chosen signal-to-noise ratio so that

$$\sigma = \sqrt{\frac{\sum (XB)^2}{\sum E^2 \cdot s}},$$

We considered the signal-to-noise (s2n) values $s = 0.1, 1, 10$.

This simulation setting corresponds to equation (1), where the true B matrix is obtained as the product of two low-rank matrices. It can be used to evaluate the performance of the Ridge RDA estimator in terms of i) recovering the latent dimension of the coefficient matrix, ii) extremely high-dimensional computing, and iii) prediction accuracy on new data.

2.7. Simulation scenarios

The performance of our ridge redundancy analysis procedure, hereafter referred to as RRDA, was evaluated in 3 different criteria: computational performance (Scenario 1), rank estimation (Scenario 2), and prediction performance (Scenario 3).

Scenario 1: Computational performance

To assess the computational efficiency of the newly implemented procedure in high-dimension settings, we evaluated the computation time of our `rrda.fit` function from the R package `rrda` and compared it with the conventional `rrs.fit` function from the R package `rrpack`. While `rrs.fit` is not designed for high-dimensional settings, it is the only available alternative implementation of RRDA. For both `rrs.fit` and `rrda.fit`, we fixed parameters: $\lambda = 1$ and $r = 5$. We also included `sRDA` (Csala et al., 2017) as a non-sparse setting (nonzero $= p$) with $\lambda = 1$ and the number of latent variables set to 5. For the simulations, data were generated based on Model 1, with the parameters $n = 100$ and $k = 5$, and tested several values $p = q \in \{10^2, 10^3, 10^4, 10^5, 10^6\}$. The computation time for a single model fitting was recorded 5 times. See Table 1S for session details.

Scenario 2: Rank estimation

Rank estimation performance was evaluated on Models 1 and 2. The matrix dimensions were set at $p{=}q \in 50, 100, 200, 500$, and the number of samples set at $n \in 100, 200$. The true rank k was chosen from $2, 5, 10$. For each setting, simulation replicates were conducted 20 times. We compared the performance of the following procedures: information criteria applied to RRDA, information criteria applied to ANN, and `StARS` approach. A 5-fold CV was applied to either classic non-ridge RDA (`c-rda.cv`) for selecting rank, and to RRDA (`rrda.cv`) for selecting both the rank and the ridge penalty parameter.

Scenario 3: Prediction performance

The predictive performance of the different parameter tuning methods was evaluated through external cross-validation. We fixed

the parameters at $n = 100$, $p = 200$, $q = 200$, 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 ($n_{\text{test}} = 10$). This train/test split was repeated 100 times. The mean squared prediction error (MSPE) was then evaluated on each test set as follows:

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

Here, the error is normalized by the number of responses q. The average MSPE across splits was compared in the information criterion-based ANN methods, `StARS`, `c-rda.cv` ($\lambda = 0$), and `rrda.cv` ($\lambda > 0$). The 5-fold CV for parameter tuning was performed within the training set. For ANN, which does not incorporate the ridge penalty in its framework, we applied a reduced-rank regression model with a selected rank.

2.8. Application to high-dimensional omics data

Three publicly available omics datasets were considered for the evaluation of the RRDA procedure (see Appendix A for details):

- The Breast Cancer dataset (Chin et al., 2006) for investigating the relationship between gene expression profiles and copy number variations (CNVs) in patients. Two versions were analyzed: a chromosome 13 subset ($X \in \mathbb{R}^{89\times 319}$, $Y \in \mathbb{R}^{89\times 58}$) (also used in Wen et al. 2023), and a whole-genome set ($X \in \mathbb{R}^{89\times 19,672}$, $Y \in \mathbb{R}^{89\times 2,149}$).

- The Soybean dataset of (Dang et al. 2025), for investigating the relationship between microbiome and metabolome in the plant rhizosphere, where $X \in \mathbb{R}^{179\times 4771}$ represents microbiome data, and $Y \in \mathbb{R}^{179\times 253}$ represents metabolome data.

- The Cancer Genome Atlas (TCGA) methylation and expression dataset (Colaprico et al., 2016) to evaluate the association between DNA methylation and gene expression. We define $X \in \mathbb{R}^{20\times 392,277}$ as DNA methylation and $Y \in \mathbb{R}^{20\times 67,528}$ as gene expression.

For each application, the dataset was randomly split into training and test sets to evaluate predictive and computational performance, with criteria described in Section 2.4 and 5-fold CV.

Additionally, following parameter tuning via 5-fold CV, $\hat{B}(\hat{\lambda}, \hat{r})$ was visualized as a heatmap to highlight the association of the top 30 variables in the feature interactions on the r-dimensional space.

3. Results

3.1. Simulation Results

Visualization

We first illustrate the parameter selection strategy implemented in the `rrda` R package, using simulation models 1 and 2 with parameters $n = 100$, $p = q = 200$, and $k = 5$. Similarly to the classical output in ridge regression, the regularization path can be visualized through a graph representing the cross-validated mean square error (CV-MSE) as a function of the (log) value of the regularization parameter λ, as shown in Figure 1. Note that each curve corresponds to a candidate rank value r. The optimal combination $\hat{\lambda}, \hat{r}$ is then obtained as the location of the minimum among all curves, corresponding to the left vertical dotted line on the graph. The λ value corresponding to the one-standard-error rule is also displayed (right vertical dotted line). In what follows, only the results obtained with $\hat{\lambda}, \hat{r}$ are displayed, as the one-standard-error rule did not show any improvement (Table 9S).

Computational performance

Table 2 displays the computational times (obtained on a single CPU core) of 3 different implementations of the Ridge RDA, corresponding to R packages `sRDA`, `rrpack`, and `rrda`, respectively. The computational time corresponds to a single fit of the model with a rank set to 5 and a predetermined lambda value 1. For mild values of $(p, q) = 10^3$, `rrda` achieves a $\times 100$ speed-up over its best competitor. For higher values of $(p, q) \geq 10^4$, the computational times become prohibitive for both `sRDA` and `rrpack` while remaining lower than 5 seconds for `rrda`. At $(p, q) = 10^6$, conventional implementations in `rrpack` and `sRDA` require over 7,450 GB of memory, rendering computation infeasible. In contrast, `rrda` performed the task in 45.66 seconds. See Table 1S for session details.

Rank estimation

Figure 2 displays the behavior of classic criteria or CV-MSE (Y-axis) across a rank grid (X-axis), with the estimated rank indicated in red and the true rank in blue. The parameters were set to $n = 100$, $p = q \in \{50, 100, 200\}$, $k = 5$, and $s2n = 1$ (mid-noise) with the best λ selected for each rank. As shown, all of the tested classic criteria (AIC, BIC, GIC, BICP, GCV) severely overestimated the rank due to overfitting, as indicated by the sum of squared errors (SSE) converging to zero. In contrast, `rrda.cv` provided a reliable rank estimation. Similar patterns were observed across broad parameter settings (Tables 2S, 3S, and 4S).

Next, we evaluated the scenario where information criteria are applied to ANN (information criterion-based ANN), comparing `StARS` with RRDA-based CV. Although those information criterion-based ANN methods were more stable than the classic grid search of information criteria, overfitting was still observed. Especially when $n = p = q = 100$ or 200, with `ANN.BIC`, `ANN.BICP`, `ANN.GIC`, and `ANN.GCV` showed extreme overestimation of the rank (Figure 2S).

For a detailed comparison between information criteria-based ANN methods and RRDA-based CV, the results of the parameter setting $n = 100$, $p = q = 200$ are presented in Table 3. `ANN.AIC` and `StARS` tended to overestimate the rank, while `ANN.BIC`, `ANN.GIC`, and `ANN.BICP` often underestimated it. Especially in high-noise settings ($s2n = 0.1$), `ANN.BIC`, `ANN.GIC`, and `ANN.BICP` returned an estimated rank of 0, indicating failure to detect any signal. A broader landscape across matrix sizes is provided in Tables 5S, 6S, and 7S. In low-dimensional settings, `ANN.AIC` and `StARS` yielded relatively more accurate rank estimates, consistent with Wen et al. (2023). In contrast, in high-dimensional settings, both `ANN.AIC` and `StARS` consistently estimated the rank at its maximum value. See also Appendix D for details.

RRDA-based CV methods were the most reliable compared with other methods. Notably, `rrda.cv` outperformed `c-rda.cv`, especially in high-noise settings ($s2n = 0.1$), highlighting the role of ridge regularization in accurate rank estimation.

Prediction performance

Table 4 displays the rank estimation and the prediction accuracy (computed as MSPE using data random-splitting) on Scenario 3. The results show that the `rrda.cv` gave the lowest MSPE among classical criteria and `StARS`. The `rrda.cv` formula clearly outperformed `c-rda.cv` by incorporating the ridge penalty. Similar trends were observed under various noise conditions (Table 8S). The RRDA also outperformed a standard ridge regression and a

multivariate linear regression without regularization (Table 9S). Using our newly implemented procedure, the computation time of CV was less than 5 seconds for all settings.

3.2. Application Results

3.2.1. Breast cancer data

We first evaluated the performance of `rrda` on the chromosome 13 subset ($X \in \mathbb{R}^{89\times 319}$, $Y \in \mathbb{R}^{89\times 58}$), using 100 random-splitting of the initial dataset ($n_{\text{train}} = 79, n_{\text{test}} = 10$). We also considered the criterion-based ANN methods and `StARS`, as described in Wen et al. (2023). Contrary to their findings, `c-rda.cv` outperformed `StARS` in rank estimation and prediction accuracy. Across all tested methods, the `rrda.cv` yielded the best prediction accuracy (Table 10S).

The `rrda` was then applied to the whole-genome dataset ($X \in \mathbb{R}^{89\times 19,672}$, $Y \in \mathbb{R}^{89\times 2,149}$) for scalability assessment. Among all methods, `rrda.cv` yielded the lowest MSPE. The `rrda` procedure completed inference in $\sim$ 50 seconds using a single core and $\sim$ 6 seconds using parallel computing (Table 12S). In comparison, `StARS` completed inference in $\sim$ 6 minutes, all other criteria requiring 20 seconds or more each.

After performing CV using the `rrda.cv` function, we obtained the estimated coefficient matrix $\hat{B}(\hat{\lambda}, \hat{r})$ with the selected parameter set $(\hat{r}, \hat{\lambda}) = (36,\ 3{,}611)$. This result was visualized as a heatmap to highlight CNV–gene expression interactions (Figure 3S). The model identified several genes, such as *CEACAM5* and *SCGB2A2*, which are known to be overexpressed in breast cancer carcinomas (Sjödin et al., 2008; Bechmann et al., 2020). These genes showed strong correlations with clusters of CNVs located on chromosomes 11 and 17.

RRDA was further compared to sparse RDA (`sRDA`; Csala et al. 2017) on the whole-genome dataset. Due to the high computational burden of `sRDA`, we limited its evaluation to 5 random splits (train/test = 79/10). Hyperparameter selection for both methods was conducted via a 5-fold CV within each training split. The parameter grid was designed to balance computational feasibility with effective coverage (see Figure 4S). For `sRDA`, parameters were tuned over the predefined grid to maximize the sum of absolute correlations, as proposed by Csala et al. (2017). We also tested MSE-based tuning, which yielded similar performance. As summarized in Table 13S, `rrda` consistently achieved lower MSPE and significantly faster computation. While `sRDA` with Elastic Net required over 24 hours to complete five splits, `rrda` completed in under 2 minutes on a single core.

Additionally, we investigated the prediction performance at the chromosome-level in the whole-genome scenario. Two approaches were compared: (i) *Chr-Chr*, where gene expression values from each chromosome were used to predict CNV values on the same chromosome; and (ii) *All-Chr*, where gene expression values from all chromosomes were used to predict CNV values on each individual chromosome. MSPE was evaluated via 100 random-splitting (train/test = 79/10). The results indicate similar performance for both approaches (Figure 3a). Alternatively, the chromosome-level scenario was tested in the inverted setting, i.e., using CNV as predictors and gene expression as response variables to evaluate how CNV affects gene expression. Figure 3b indicates that the *All-Chr* approach yielded more accurate predictions than the *Chr-Chr* approach in the inverted setting.

3.2.2. Soybean multi-omics data

For the Soybean dataset ($X \in \mathbb{R}^{179\times 4771}$, $Y \in \mathbb{R}^{179\times 253}$), we conducted 100 random-splitting ($n_{\text{train}} = 159, n_{\text{test}} = 20$). Over splittings CV mostly selected between 10 and 12 (Figure 4), and RRDA yielded the lowest MSPE among all tested methods (Table 11S).

Similar to the Breast Cancer application, a low-dimensional representation of the estimated coefficient matrix $\hat{B}(\hat{\lambda}, \hat{r})$ is displayed in Figure 4c. Consistent with the analysis of Dang et al. (2025), `rrda` method identified the *Chitinophagaceae* family (Carrión et al., 2019) as a key microbial variable. In the metabolome, tryptophan and tyrosine were correlated with microbiome, in line with their reported roles in plant–microbe community (Sugiyama et al., 2014).

The `rrda` method identified additional microbe–metabolite associations. *Paenibacillaceae* showed positive correlations with soybean-derived aglycones (daidzein, genistein) and negative correlations with their glycosylated forms (daidzin, genistin). This pattern reflects the known ability of *Paenibacillus* to convert glycosides into bioactive aglycones (Park et al., 2013), highlighting their role in shaping the rhizosphere metabolome.

3.2.3. TCGA methylation and expression data

The TCGA dataset ($X \in \mathbb{R}^{20\times 392,277}$, $Y \in \mathbb{R}^{20\times 67,528}$) was tested using a leave-one-out (LOO) approach. Parameter tuning was performed for each training set through 5-fold CV. Note that alternative approaches (e.g., `rrpack`, `StARS`) required approximately 100 GB of memory, and were unable to handle matrices of this size, even on a computing server. `sRDA` was also unable to operate under a non-sparse setting (i.e., nonzero=p). Therefore, our approach was the only feasible method for conducting the RRDA on this dataset. The computation time for parameter tuning (5-fold CV) using the `rrda.cv` function was 16 seconds using parallel computing (Table 12S).

The average estimated rank was 5.10 ± 1.01, and the regularization parameter was $\lambda = 6.17 \times 10^3 \pm 1.17 \times 10^3$, with an MSPE of $7.27 \times 10^{-2} \pm 0.95 \times 10^{-2}$ across LOO evaluations. All values are reported as mean $\pm$ standard error.

The top 30 features of methylation sites and gene expression in the predicted low-dimensional coefficient matrix $\hat{B}(\hat{\lambda}, \hat{r})$ (Figure 5S) highlights a group of methylation sites on chromosome X (e.g., `cg21983484`, `cg01742836`) that are strongly associated with specific gene expression (`TC0X001155.hg.1`, `TC0X002125.hg.1`) on the chromosome X. The observation is consistent with previously reported regulation, with e.g. `TC0X001155.hg.1` corresponding to a non-coding RNA known as *XIST*, which is known to play a critical role in X chromosome inactivation (Lee et al., 2019).

4. Discussion

The novel implementation of the Ridge Redundancy Analysis method, `rrda`, effectively addresses the computational challenges of high-dimensional settings and makes the procedure compatible with applications to large-scale datasets, including omics-integrative approaches as illustrated here.

Beyond scalability, the computational efficiency of our `rrda` procedure opened the way to efficient resampling strategies for the joint tuning of the regularization parameter λ and the rank r. Our study revealed the limitations of some classical information

criteria for RRDA parameter tuning in high-dimensional settings. Indeed, whenever $p, q > n$ rank values larger than r can reduce the sum of squared errors to zero, resulting in an overfit that is not compensated by the penalty term (Figure 2) and leads to systematic overestimation of the rank. Although the ANN grid search partly mitigated this issue, its performance remained sensitive to dimensionality (Figure 2S). In both simulations and real data experiments, the K-fold CV of `rrda` led to improved rank estimation and yielded the best prediction performance compared to alternative selection criteria and competing models (Table 9S).

In our experiments, we observed a clear overestimation of the rank when using `StARS` in high-dimensional settings (Tables 5S, 6S, and 7S) that was not reported in the seminal paper. This overestimation may be partially explained by the implementation of `StARS` provided by Wen et al. (2023). In the formula, `StARS` is first applied with the smallest threshold τ - i.e., with the largest rank - and then moves to higher values of τ until stability is reached via a sub-sampling method. However, in high-dimensional settings, when $r = n_{\text{sample}}$ (i.e., when the rank equals the sample size), the ANN model fits perfectly. This corresponds to the first "stable" rank estimate reached, which stops the grid search procedure and results in an overestimation of the rank (Figure 2S).

The various applications to real datasets illustrated the practical advantages of `rrda` compared to existing RDA methods. In the Breast Cancer application, `rrda` consistently outperformed `sRDA` both in terms of accuracy and computational efficiency (Table 13S). Although sparsity has often been advocated as leading to more interpretable models, it has been acknowledged that the identified support of variables with non-zero coefficients may be quite unstable under subsampling or slight changes of the experimental conditions, especially in high-dimensional settings (see e.g. Meinshausen and Bühlmann 2010; Baldassarre et al. 2017). In comparison, the `rrda` coefficient matrix in such settings was estimated to be relatively low-dimensional, resulting in a limited number of components to interpret, and facilitated graphical representations, hence getting around the classical limitation of non-rank-restricted ridge procedures.

Our analysis suggests that integrating whole-genome data (*All-Chr*) can improve prediction accuracy in certain settings. In particular, when predicting gene expression from CNVs, global information was more effectively captured within the `rrda` framework, leading to improved performance (Figure 3). This improvement may be due to the trans-regulatory nature of CNVs, making genome-wide integration beneficial. More generally, having access to efficient and scalable whole-genome prediction procedures may be advantageous in scenarios where detailed genomic mapping is unavailable (Meuwissen et al., 2001; Bermingham et al., 2015).

Furthermore, the ability of `rrda` to offer a compact low-dimensional representation of feature associations was demonstrated in all the applications, where it revealed biologically meaningful structures across diverse omics layers (Figures 4c, 3S, and 5S). These structures, visualized through interpretable heatmaps, were obtained with minimal computational cost.

The RRDA framework we developed provides practical tools for multivariate data analysis in genomics and related fields, and is available as the R package `rrda` on CRAN.

Competing interests

No competing interest is declared.

Author contributions statement

H.Y., J.A., and T.M.-H. conceived the study. H.Y., J.A., and T.M.-H. analysed the data. H.Y., J.A., H.I., and T.M.-H. wrote and reviewed the manuscript.

Acknowledgements

This work was supported by JSPS-KAKENHI (JP23KJ0506, JP22K21352), JST-CREST (JPMJCR16O2), JST-ALCA-Next (JPMJAN23D1), and JST-Mirai (JPMJMI120C7). The authors thank the researchers who contributed to the soybean data collection: Y. Fuji, M. H. Yokota, Y. Ichihashi, and all the other collaborators and technical staff.

References

H. Akaike. A new look at the statistical model identification. *IEEE Trans Autom Control*, 19:716–723, 1974.

H. An, D. Huang, Q. Yao, and C.-H. Zhang. Stepwise searching for feature variables in high-dimensional linear regression. 2008.

T. Anderson. Estimating linear restrictions on regression coefficients for multivariate normal distributions. *Ann Math Stat*, 22:327–351, 1951.

L. Baldassarre, M. Pontil, and J. Mourão-Miranda. Sparsity is better with stability: Combining accuracy and stability for model selection in brain decoding. *Front Neurosci*, 2017.

M. B. Bechmann, A. V. Brydholm, V. L. Codony, J. Kim, and R. Villadsen. Heterogeneity of ceacam5 in breast cancer. *Oncotarget*, 11(43):3886–3899, 2020. ISSN 1949-2553.

H. Bengtsson. A unifying framework for parallel and distributed processing in r using futures. *The R Journal*, 2021.

M. L. Bermingham, R. Pong-Wong, A. Spiliopoulou, C. Hayward, I. Rudan, and et al. Application of high-dimensional feature selection: evaluation for genomic prediction in man. *Scientific Reports*, 5(1):10312, 2015.

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

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

T. Capblancq and B. R. Forester. Redundancy analysis: A swiss army knife for landscape genomics. *Methods in Ecology and Evolution*, 12(12):2298–2309, 2021.

V. J. Carrión, J. Perez-Jaramillo, V. Cordovez, V. Tracanna, D. Hollander, and et al. Pathogen-induced activation of disease-suppressive functions in the endophytic root microbiome. *Science*, 2019.

K. Chen and W. Wang. rrpack: Reduced-rank regression, 2022. R package version 0.1-13.

K. Chen, H. Dong, and K.-S. Chan. Reduced rank regression via adaptive nuclear norm penalization. *Biometrika*, 100(4): 901–920, 2013.

K. Chin, S. DeVries, J. Fridlyand, P. T. Spellman, and R. ett al. Genomic and transcriptional aberrations linked to breast cancer pathophysiologies. *Cancer Cell*, 10(6):529–541, 2006.

A. Colaprico, T. C. Silva, C. Olsen, L. Garofano, C. Cava, D. Garolini, and et al. Tcgabiolinks: an r/bioconductor package for integrative analysis of tcga data. *Nucleic Acids Research*, 2016.

A. Csala, F. P. J. M. Voorbraak, A. H. Zwinderman, and M. H. Hof. Sparse redundancy analysis of high-dimensional genetic and genomic data. *Bioinformatics*, 33(20):3228–3234, 2017.

T. Dang, Y. Fuji, K. Kumaishi, E. Usui, S. Kobori, and et al. I-svvs: integrative stochastic variational variable selection to explore joint patterns of multi-omics microbiome data. *Briefings in Bioinformatics*, 26(3):bbaf132, 05 2025. ISSN 1477-4054.

Y. Fan and C. Y. Tang. Tuning parameter selection in high dimensional penalized likelihood. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 75(3): 531–552, 2013.

J. H. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1):1–22, 2010.

G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

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

T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer, New York, 2001.

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

H. Hotelling. Relations between two sets of variates. *Biometrika*, 28:321–377, 1936.

H. J. Lee, R. Gopalappa, H. Sunwoo, S. W. Choi, and S. e. a. Ramakrishna. En bloc and segmental deletions of human xist reveal x chromosome inactivation-involving rna elements. *Nucleic Acids Research*, 47(8):3875–3887, 2019.

N. Meinshausen and P. Bühlmann. Stability selection. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 72(4):417–473, 2010.

T. H. E. Meuwissen, B. J. Hayes, and M. E. Goddard. Prediction of total genetic value using genome-wide dense marker maps. *Genetics*, 157(4):1819–1829, 04 2001. ISSN 1943-2631.

A. Mukherjee and J. Zhu. Reduced rank ridge regression and its kernel extensions. *Statistical Analysis and Data Mining*, 4(6): 612–622, 2011.

D.-J. Park, Y.-S. Lee, and Y.-L. Choi. Characterization of a cold-active beta-glucosidase from paenibacillus xylanilyticus kj-03 capable of hydrolyzing isoflavones daidzin and genistin. *The Protein Journal*, 32(7):579–584, 2013.

Y. Qiu and J. Mei. Rspectra: Solvers for large-scale eigenvalue and svd problems, 2024. R package version 0.16-2.

R Core Team. R: A language and environment for statistical computing, 2023.

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

G. Reinsel and R. Velu. *Multivariate Reduced-rank Regression*. Springer, New York, 1998.

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

G. Schwarz. Estimating the dimension of a model. *Annals of Statistics*, 6:461–464, 1978.

A. Sjödin, I. Ljuslinder, R. Henriksson, and H. Hedman. Mammaglobin and lipophilin b expression in breast tumors and their lack of effect on breast cancer cell proliferation. *Anticancer Research*, 28(3A):1493–1498, 2008. ISSN 0250-7005.

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

Y. Takane and M. A. Hunter. Constrained principal component analysis: A comprehensive theory. *Br J Math Stat Psychol*, 2001.

Y. Takane and H. Hwang. Regularized linear and kernel redundancy analysis. *Comput Stat Data Anal*, 52, 2007.

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

D. Vaughan and D. M. furrr: Apply mapping functions in parallel using futures, 2022. R package version 0.3.1.

P. Waldmann, G. Mészáros, B. Gredler, C. Fuerst, and J. Sölkner. Evaluation of the lasso and the elastic net in genome-wide association studies. *Frontiers in Genetics*, 4:270, 2013.

C. Wen, Q. Wang, and Y. Jiang. Stability approach to regularization selection for reduced-rank regression. *Journal of Computational and Graphical Statistics*, 32(3):974–984, 2023.

H. Wickham. ggplot2: Elegant graphics for data analysis, 2016.

D. M. Witten, R. Tibshirani, and T. Hastie. A penalized matrix decomposition, with applications to sparse principal components and canonical correlation analysis. *Biostatistics*, 2009.

Y. Zhao, C. Chang, and Q. Long. Knowledge-guided statistical learning methods for analysis of high-dimensional -omics data in precision oncology. *JCO Precision Oncology*, 2019.

H. Zou and T. Hastie. Regularization and variable selection via the elastic net. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 67(2):301–320, 03 2005.

Figures and Tables

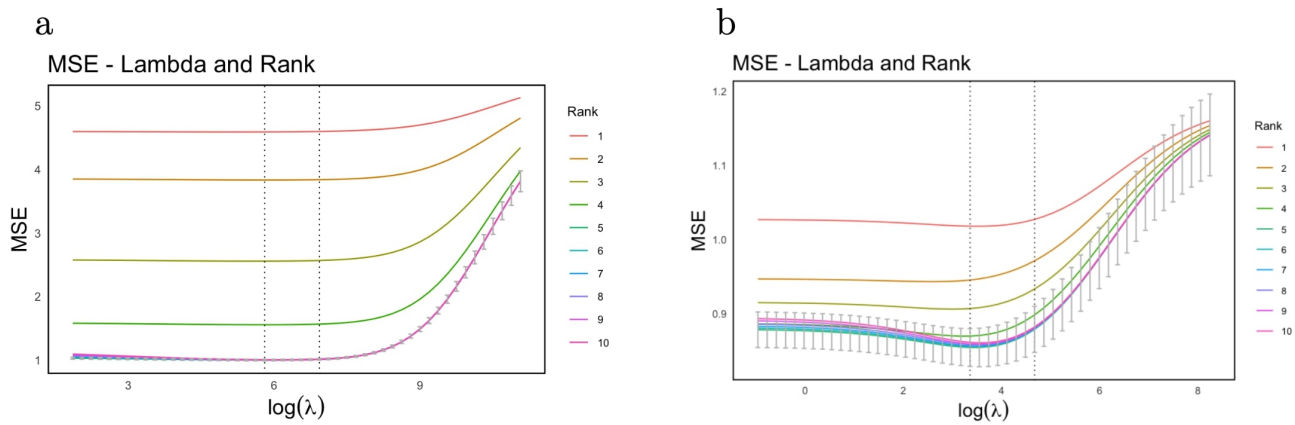

Fig. 1: Visualization of the regularization path using the **rrda** package. The plot shows the cross-validated mean squared error (CV-MSE) from 5-fold CV for all combinations of ridge penalty λ and rank. The left vertical dotted line indicates the λ value that minimizes the CV-MSE, while the right vertical dotted line corresponds to the λ selected by the one-standard-error rule. The one-standard-error is depicted as error bars. (a) Model 1: Latent Space Model; (b) Model 2: Matrix Factorization Model.

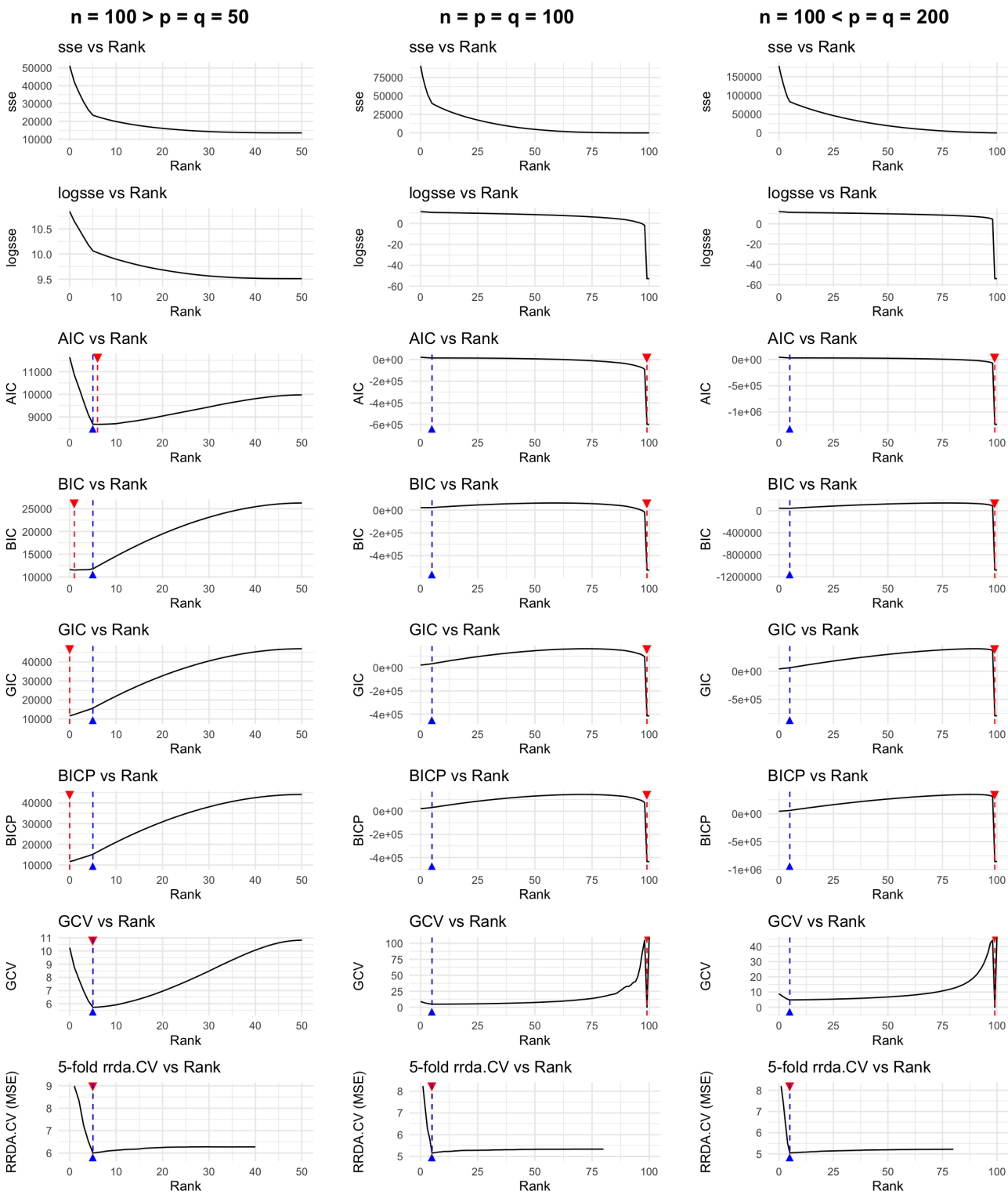

Fig. 2: Rank and classic information criteria for each setting ($n = 100 > p = q = 50$, $n = p = q = 100$, $n = 100 < p = q = 200$). Tested under Simulation Model 1 (mid-noise: s2n = 1). The grid search of both rank and λ are tested for SSE ($\|Y - \hat{Y}\|_F^2$), log(SSE), Selection methods of the AIC, BIC, GIC, BICP, GCV, and `rrda.cv`. For each rank, the best λ is selected. The x-axis indicates the rank, and the y-axis shows the corresponding selection criterion to be minimized. In each figure, the true rank is indicated by a blue upward triangle and dashed vertical line, while the estimated rank selected by the method is indicated by a red downward triangle and dashed vertical line.

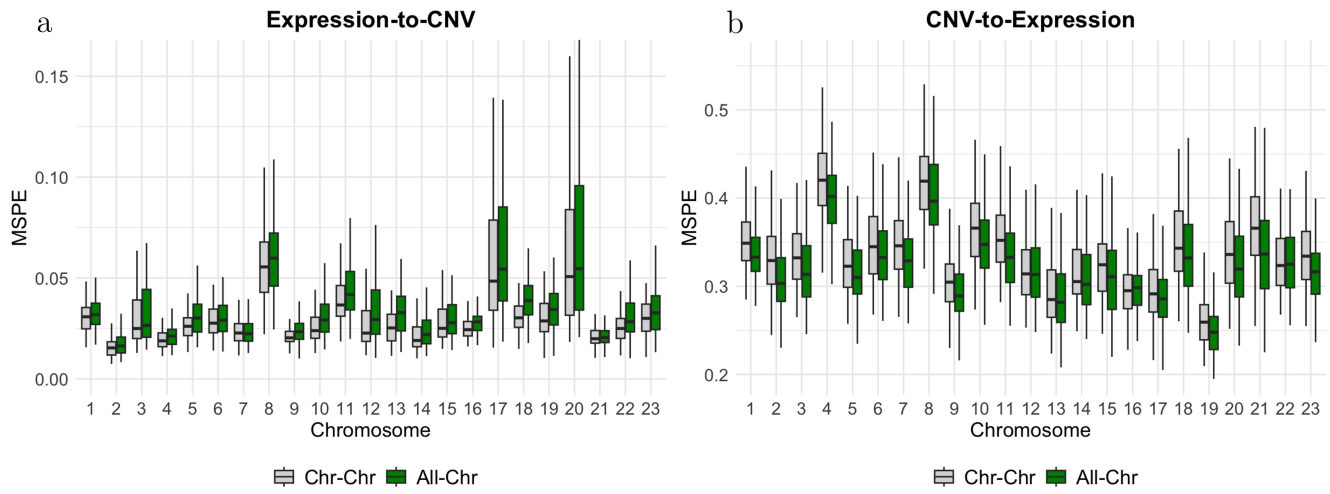

Fig. 3: Breast Cancer: MSPE for Chromosome-wise Prediction Performance. MSPE is shown for each chromosome under two modeling strategies: *Chr-Chr* (gray), where the model was trained using data from the corresponding chromosome only, and *All-Chr* (green), where the model was trained using data from the entire genome. Two prediction settings were evaluated: (a) Expression-to-CNV, where gene expression data were used to predict CNV values, and (b) CNV-to-Expression, where CNV data were used to predict gene expression values.

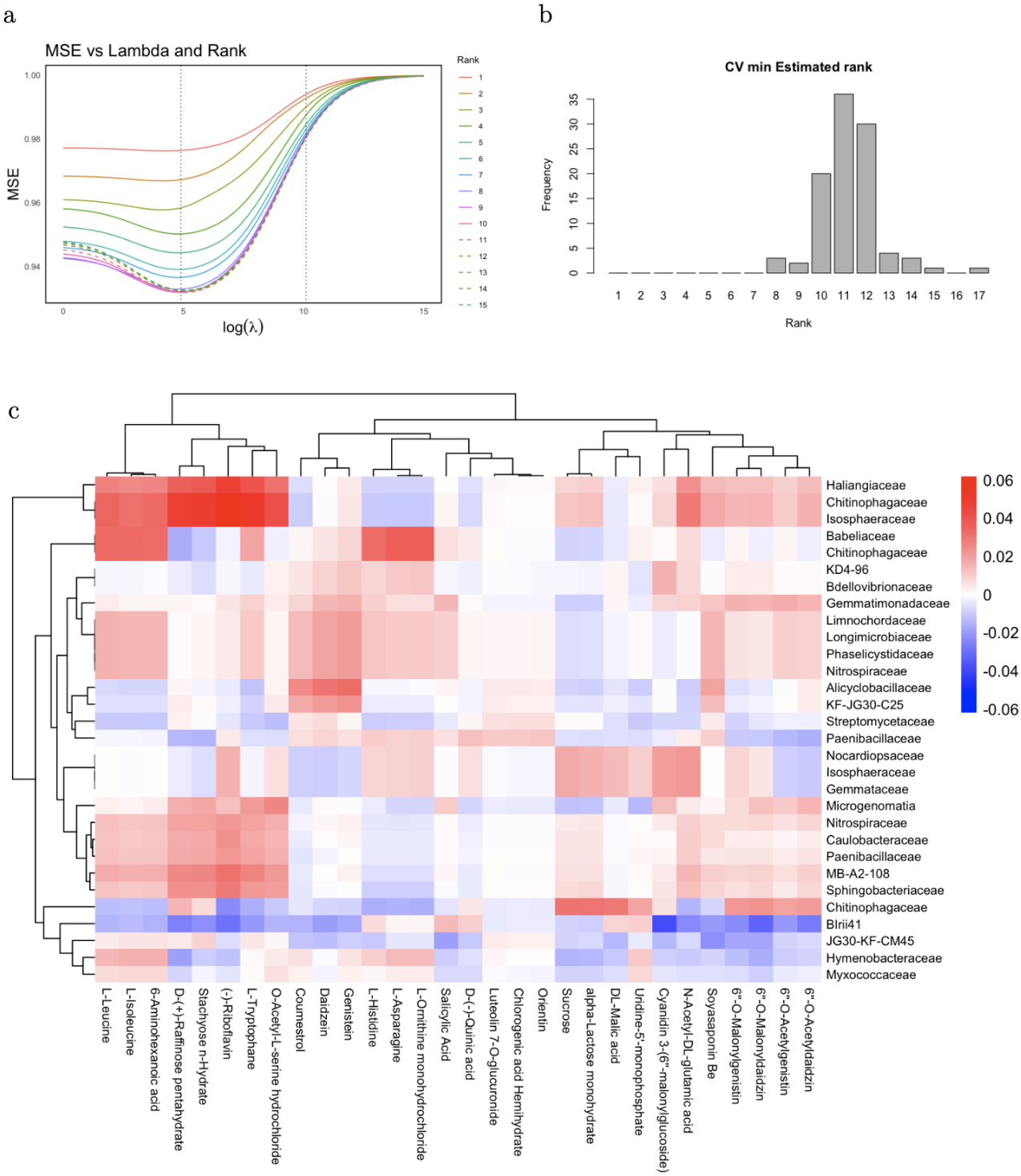

Fig. 4: Application of `rrda` in Soybean multi-omics data, where it includes the microbe taxonomic traits (p=4771) and metabolome (q=253) for the number of samples (n=179). The data was randomly split into training data (n = 159) and test data (n = 20). (a) Plot for parameter tuning (b) The best rank estimated across 100 random-splitting model evaluation. For each training was done via CV (5-folds) (c) Heatmap of rrda coefficients for selected microbiome features (each row) and metabolome features (each column) in the Soybean dataset. The top 30 variables were presented based on the ℓ_2-norm in the low-dimensional space. Based on parameter tuning (5-fold CV), with $rank = 11$ and $\lambda = 335.74$, the model was fitted and the resulting coefficient matrix was obtained.

Table 1. The evaluation criteria based methods for tuning parameter selection. Here, r_x denotes the rank of the explanatory variable matrix X, q is the number of response variables, p the number of explanatory variables, n the sample size, and $\hat{B}(r,\lambda)$ the estimator depending on rank r and penalty parameter λ.

Criterion	**Formula**
AIC	$nq \log\left(\frac{\|Y - X\hat{B}(r,\lambda)\|^2}{nq}\right) + 2\, r(r_x + q - r)$
BIC	$nq \log\left(\frac{\|Y - X\hat{B}(r,\lambda)\|^2}{nq}\right) + \log(nq)\, r(r_x + q - r)$
GIC	$nq \log\left(\frac{\|Y - X\hat{B}(r,\lambda)\|^2}{nq}\right) + \log(nq)\log(pq)\, r(r_x + q - r)$
BICP	$nq \log\left(\frac{\|Y - X\hat{B}(r,\lambda)\|^2}{nq}\right) + 2\log(pq)\, r(r_x + q - r)$
GCV	$\dfrac{nq\|Y - X\hat{B}(r,\lambda)\|^2}{\left[nq - r(r_x + q - r)\right]^2}$

Table 2. Scenario 1. Benchmark of computation time for the fitting functions (`sRDA`*, `rrpack` (using function `rrs.fit`), and `rrda`). The mean and standard error of the computation time in seconds are reported. "$> 10^5$" indicates that the computation did not converge within the time limit (10^5 seconds). "*Out of Memory*" indicates that the computation failed due to exceeding the memory limit on the computational server (see Table 1S). *`sRDA` is configured in a non-sparse setting of the Elastic Net to approximate a ridge prediction model.

	`sRDA`*		`rrpack`		`rrda`	
p, q	Mean Time (s)	Std. Error	Mean Time (s)	Std. Error	Mean Time (s)	Std. Error
10^2	4.85	0.47	0.01	0.00	0.01	0.00
10^3	2.38×10^3	141.09	3.42	0.01	0.03	0.00
10^4	$> 10^5$	–	$> 10^5$	–	0.16	0.00
10^5	$> 10^5$	–	$> 10^5$	–	4.37	0.00
10^6	*Out of Memory*	–	*Out of Memory*	–	45.66	0.01

Table 3. Scenario 2: Rank estimation with signal-to-noise ratio (s2n) = 0.1 (high-noise), 1 (mid-noise), 10 (low-noise). The results correspond to the setting with $n = 100$, $p = q = 200$. The estimated rank values for each true rank ($k = 2, 5, 10$) are averaged across iterations, shown with standard errors in parentheses.

			ANN						**RDA**	
s2n	**Model**	**rank (k)**	**AIC**	**BIC**	**GIC**	**BICP**	**GCV**	**StARS**	**c-rda.cv**	**rrda.cv**
0.1	Model 1	2	99.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	2.40 (0.18)	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)
		5	99.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	6.95 (0.25)	99.00 (0.00)	1.25 (0.10)	**5.15** (0.08)
		10	99.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	7.45 (0.29)	99.00 (0.00)	1.00 (0.00)	**9.45** (0.59)
	Model 2	2	99.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	2.35 (0.15)	99.00 (0.00)	1.60 (0.11)	**1.95** (0.05)
		5	99.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	6.95 (0.26)	99.00 (0.00)	1.25 (0.12)	**4.25** (0.29)
		10	99.00 (0.00)	0.00 (0.00)	0.00 (0.00)	0.00 (0.00)	7.00 (0.35)	99.00 (0.00)	1.00 (0.00)	**8.00** (0.90)
1	Model 1	2	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)	**2.00** (0.00)	2.30 (0.15)	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)
		5	99.00 (0.00)	4.95 (0.05)	0.00 (0.00)	0.00 (0.00)	5.05 (0.05)	99.00 (0.00)	**5.00** (0.00)	**5.00** (0.00)
		10	99.00 (0.00)	0.70 (0.18)	0.00 (0.00)	0.00 (0.00)	10.30 (0.16)	99.00 (0.00)	**10.00** (0.00)	**10.00** (0.00)
	Model 2	2	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)	**2.00** (0.00)	2.30 (0.18)	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)
		5	99.00 (0.00)	**5.00** (0.00)	0.00 (0.00)	0.00 (0.00)	5.05 (0.05)	99.00 (0.00)	4.90 (0.07)	**5.00** (0.00)
		10	99.00 (0.00)	1.05 (0.18)	0.00 (0.00)	0.00 (0.00)	10.35 (0.18)	99.00 (0.00)	9.45 (0.15)	**9.65** (0.15)
10	Model 1	2	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)	**2.00** (0.00)	2.25 (0.16)	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)
		5	99.00 (0.00)	**5.00** (0.00)	**5.00** (0.00)	**5.00** (0.00)	5.10 (0.10)	99.00 (0.00)	**5.00** (0.00)	5.05 (0.05)
		10	99.00 (0.00)	**10.00** (0.00)	0.00 (0.00)	0.05 (0.05)	**10.00** (0.00)	99.00 (0.00)	**10.00** (0.00)	10.05 (0.05)
	Model 2	2	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)	**2.00** (0.00)	2.40 (0.15)	99.00 (0.00)	**2.00** (0.00)	**2.00** (0.00)
		5	99.00 (0.00)	**5.00** (0.00)	**5.00** (0.00)	**5.00** (0.00)	5.40 (0.20)	99.00 (0.00)	**5.00** (0.00)	**5.00** (0.00)
		10	99.00 (0.00)	**10.00** (0.00)	0.00 (0.00)	0.10 (0.07)	**10.00** (0.00)	99.00 (0.00)	**10.00** (0.00)	**10.00** (0.00)

Table 4. Scenario 3. The `rrda` evaluation for simulation data. The signal-to-noise (s2n) = 1. 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. Mean values are presented with standard error in the parentheses. For session details, see Table 1S

	ANN						**RDA**	
	AIC	**BIC**	**GIC**	**BICP**	**GCV**	**StARS**	**c-rda.cv**	**rrda.cv**
Model 1: Latent Space Model								
Rank	89.00 (0.00)	4.58 (0.05)	0.00 (0.00)	0.00 (0.00)	5.00 (0.00)	89.00 (0.00)	5.00 (0.00)	5.00 (0.00)
λ	- -	- -	- -	- -	- -	- -	- -	888 (7.61)
MSPE	8.93 (3.41e-2)	5.47 (3.96e-2)	9.06 (7.78e-2)	9.06 (7.78e-2)	5.18 (1.72e-2)	8.93 (3.41e-2)	5.18 (1.72e-2)	**5.00** (1.57e-2)
Model 2: Matrix Factorization Model								
Rank	89.00 (0.00)	4.09 (0.03)	0.00 (0.00)	0.11 (0.03)	5.00 (0.00)	89.00 (0.00)	4.78 (0.04)	4.94 (0.02)
λ	- -	- -	- -	- -	- -	- -	- -	28.2 (1.00)
MSPE	2.62 (1.64e-2)	1.69 (9.90e-3)	1.97 (1.59e-2)	1.97 (1.58e-2)	6.13 (4.44)	2.62 (1.64e-2)	6.13 (4.44)	**1.62** (9.30e-3)