Journal of Data Science 22 (2), 259–279
April 2024
DOI: 10.6339/24-JDS1127
Statistical Data Science

Interaction Selection and Prediction Performance in High-Dimensional Data: A Comparative Study of Statistical and Tree-Based Methods

Chinedu J. Nzekwe[1,*], Seongtae Kim[1], and Sayed A. Mostafa[1]

[1]*Department of Mathematics and Statistics, North Carolina Agricultural and Technical State University, Greensboro, NC, 27411, USA*

Abstract

Predictive modeling often ignores interaction effects among predictors in high-dimensional data because of analytical and computational challenges. Research in interaction selection has been galvanized along with methodological and computational advances. In this study, we aim to investigate the performance of two types of predictive algorithms that can perform interaction selection. Specifically, we compare the predictive performance and interaction selection accuracy of both penalty-based and tree-based predictive algorithms. Penalty-based algorithms included in our comparative study are the regularization path algorithm under the marginality principle (RAMP), the least absolute shrinkage selector operator (LASSO), the smoothed clipped absolute deviance (SCAD), and the minimax concave penalty (MCP). The tree-based algorithms considered are random forest (RF) and iterative random forest (iRF). We evaluate the effectiveness of these algorithms under various regression and classification models with varying structures and dimensions. We assess predictive performance using the mean squared error for regression and accuracy, sensitivity, specificity, balanced accuracy, and F1 score for classification. We use interaction coverage to judge the algorithm's efficacy for interaction selection. Our findings reveal that the effectiveness of the selected algorithms varies depending on the number of predictors (data dimension) and the structure of the data-generating model, i.e., linear or nonlinear, hierarchical or non-hierarchical. There were at least one or more scenarios that favored each of the algorithms included in this study. However, from the general pattern, we are able to recommend one or more specific algorithm(s) for some specific scenarios. Our analysis helps clarify each algorithm's strengths and limitations, offering guidance to researchers and data analysts in choosing an appropriate algorithm for their predictive modeling task based on their data structure.

Keywords *interaction selection; iRF; LASSO; predictive modeling; RAMP; RF*

1 Introduction

Modern data collection technology has made it possible for researchers in many different scientific domains to access high-dimensional and ultra-high-dimensional data at relatively low costs. Sparsity is common in high-dimensional data, especially in genomics, proteomics, biomedical imaging, tumor classification, image analysis, signal processing, and finance (Evans, 2006; Manolio and Collins, 2007; Kooperberg and Leblanc, 2008; Cordell et al., 2009). Variable selection plays a critical role in high-dimensional data modeling. Variable selection can increase estima-

*Corresponding author. Email: cjnzekwe@ncat.edu.

Received August 1, 2023; Accepted March 26, 2024

tion accuracy and model interpretability with sparsity by successfully identifying the subset of significant predictors. Over the last decades, numerous statistical and machine learning-based methods have been proposed for high-dimensional data, especially when the number of predictors (p) surpasses the number of observations (N) (Fan and Li, 2006; Donoho, 2000; Fan and Lv, 2010).

Conventional methods such as stepwise selection, forward and backward selection, and subset selection combined with AIC and BIC were devised to address variable selection tasks successfully. However, these methods face computational challenges as the number of predictors (p) grows. For example, subset selection becomes exceptionally demanding when p exceeds 20 due to the potential creation of 2^p subset models. Additionally, forward and backward selection criteria encounter difficulties in backtracking variable addition or removal orders. Researchers have proposed different families of variable selection methods in the high-dimensional setting, some of which are penalty-driven algorithms. This type of penalty-driven method simultaneously achieves variable selection and parameter estimation in various types of regression models. For instance, the least absolute shrinkage selector operator (LASSO) performs variable selection via L_1-norm penalization on the regression coefficients (Tibshirani, 1996). LASSO is inclined to set some coefficients to zero while promoting sparsity, which can lead to the exclusion of relevant variables, potentially overlooking important features. In situations involving correlated predictors or when the number of predictors surpasses the number of observations, LASSO may exhibit instability in variable selection. Other penalty-driven variable selection algorithms are Smoothed Clipped Absolute Deviance (SCAD) (Fan and Li, 2001) and Minimax Concave Penalty (MCP) (Zhang, 2010). Both algorithms incorporate a nonconvex penalty function to alleviate excessive penalization of large coefficient values in regression. The penalty-driven methods depend on selecting appropriate tuning parameter values, which can be non-trivial.

While the aforementioned variable selection methods have been successfully applied in regression modeling, they primarily focus on selecting main effects. When we consider two-way interactions between main effects, the dimension of predictors significantly increases from p to $p + p(p-1)/1$. Conventional variable selection methods may not be well-suited for this expanded dimensionality. However, penalty-driven variable selection methods may be more suitable for addressing interaction selection in high-dimensional data, *e.g.*, additive LASSO (Hastie and Tibshirani, 1990). Bien et al. (2013) introduced LASSO with hierarchical interaction selection. Zhao et al. (2009) proposed a composite absolute penalty function and showed that it can perform two-way interaction selection. Yuan et al. (2009) proposed a structured two-way interaction selection and estimation procedure for parametric models. Choi et al. (2010) introduced the parametric interaction selection method under the heredity assumption. Hao et al. (2018) pointed out that these approaches perform well and exhibit oracle properties when the number of predictors is relatively small, typically a few hundred or less. However, when p is much greater than the sample size, these techniques become impractical due to the computational challenges associated with handling the entire $\mathcal{O}(p^2) \times N$ design matrix and solving intricate constrained optimization problems. More recently, interaction screening methods have been proposed to deal with interaction selection in high dimensional data (Hao and Zhang (2014, 2017); Kong et al. (2017)). While these methods are computationally efficient, their selection performance is slightly subpar due to forward selection and the marginal relationship between predictors and response. Subsequently, Hao et al. (2018) proposed the Regularization path Algorithm under the Marginality Principle (RAMP), designed to select main effects and interactions simultaneously under certain statistical principles defined in Section 2. The RAMP algorithm considers two-way hierarchical multiplicative interactions within penalized linear or logistic regression, which

means that the method has some limitations for higher-order, nonlinear interactions.

Alternatively, tree-based algorithms have the potential to detect and select higher-order non-multiplicative interactions in high-dimensional data while sacrificing the selection process's interpretability. This tree-based approach enables the use of several ensemble methods for interaction selection and model construction, such as bagging (Breiman, 1996) and random subspace sampling (Tin Kam Ho, 1998; Breiman, 2001). As outlined by (Kotsiantis and Kanellopoulos, 2012), these ensemble techniques effectively achieve robust modeling in environments with high noise levels. The bagging technique involves training a multitude of models on different segments of the dataset, which significantly reduces variance and boosts the robustness of the overall ensemble. Additionally, random forest (RF) (Breiman, 2001; Liaw and Wiener, 2002), integrating random subspace sampling with bagging, enhances the ensemble's performance. This integration promotes model de-correlation, thereby elevating the ensemble's ability to generalize effectively in the presence of noise and outliers. However, these tree-based ensemble methods were not originally designed to capture interaction selection in ultrahigh dimensional data. The random intersection trees (RIT), proposed by Shah and Meinshausen (2014), is an algorithm built on the modification of RF that aids the search for stable, high-order non-multiplicative interactions. The iterative random forest (iRF), developed by Basu et al. (2018), is an extension of the RIT that can capture non-hierarchical, higher-order interactions.

This study aims to compare the predictive performance of two approaches for interaction selection: penalty-driven algorithms, which mainly capture two-way multiplicative interactions in regression, and tree-based ensemble algorithms, which excel at selecting higher-order non-multiplicative interactions. While both penalized regression and tree-based ensemble methods are popular supervised learning techniques, there is a research gap comparing their predictive performance in high and ultra-high-dimensional data, especially in the presence of interactions. We perform comparative simulation studies and empirical data analyses to address this gap.

The rest of the paper is organized as follows. Section 2 describes the variable selection algorithms considered in our comparative analyses. Section 3 summarizes the settings and results of extensive simulation experiments comparing the predictive and variable selection performance of penalized regression methods and ensemble tree-based methods. Section 4 compares the two sets of methods on a real high-dimensional data set. We conclude the paper with a discussion of the main results in Section 5.

2 Methods

This section describes selected predictive modeling algorithms that perform interaction selection: LASSO, SCAD, MCP, RAMP, and iRF. Let p denote the number of predictors, and let N denote the sample size of the training data. A linear regression model with p main effects and their two-way multiplicative interactions is defined as follows:

$$Y = \beta_0 + \sum_{j=1}^{p} \beta_j X_j + \sum_{k=1}^{p-1} \sum_{\ell=k+1}^{p} \beta_{k,\ell} X_k X_\ell + \varepsilon, \tag{1}$$

where ε is the independently and identically distributed error term with $E(\varepsilon) = 0$ and $E(\varepsilon^2) = \sigma^2$. In matrix notation, we can rewrite (1) as follows:

$$\mathbf{y} = \beta_0 \mathbf{1}_N + \mathbf{Z}\boldsymbol{\beta} + \boldsymbol{\varepsilon}, \tag{2}$$

where $\mathbf{y} = (Y_1, Y_2, \ldots, Y_N)^T$ represents the response vector, and $\boldsymbol{\varepsilon} = (\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N)^T$ denotes the corresponding error vector. The augmented coefficient vector is denoted as $\boldsymbol{\beta} = (\boldsymbol{\beta}_1^T, \boldsymbol{\beta}_2^T)^T$ where $\boldsymbol{\beta}_1 = (\beta_1, \beta_2, \ldots, \beta_p)^T$ represents the main effects coefficients vector, and $\boldsymbol{\beta}_2 = (\beta_{1,2}, \ldots, \beta_{p-1,p})^T = (\beta_{p+1}, \ldots, \beta_q)^T$ corresponds to the vector of interaction coefficients with $q = p + p(p-1)/2$. Each X_i is standardized. The augmented design matrix is given by $\mathbf{Z} = (\mathbf{Z}_1\ \mathbf{Z}_2)$ where $\mathbf{Z}_1$ and $\mathbf{Z}_2$ represent specific components of the matrix.

$$\mathbf{Z}_1 = \begin{pmatrix} X_{11} & X_{12} & \ldots & X_{1p} \\ X_{21} & X_{22} & \ldots & X_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \ldots & X_{Np} \end{pmatrix} \text{ and } \mathbf{Z}_2 = \begin{pmatrix} X_{11}X_{12} & \ldots & X_{1(p-1)}X_{1p} \\ X_{21}X_{22} & \ldots & X_{2(p-1)}X_{2p} \\ \vdots & \ddots & \vdots \\ X_{N1}X_{N2} & \ldots & X_{N(p-1)}X_{Np} \end{pmatrix}.$$

In statistical modeling, interaction terms involve two fundamental principles for interaction selection: the *hierarchical* principle and the *marginality* principle.

1. The *hierarchical* principle states that when interaction effects are selected, we also include the main effects irrespective of their significance (McCullagh, 2002; Yuan et al., 2009; Zhao et al., 2009; Choi et al., 2010).
2. Conversely, the *marginality* principle states that when main effects are selected, we consider the inclusion of interaction effects among the selected main effects (Nelder, 1977; Chipman et al., 1997; McCullagh, 2002; Zhao et al., 2009; Choi et al., 2010).
   - Strong heredity rule: If both predictors X_k and X_ℓ are selected, then we include the two-way multiplicative interaction of them.
   - Weak heredity rule: If either X_k or X_ℓ is selected, then we include the two-way multiplicative interaction of them.

2.1 Least Absolute Shrinkage Selection Operator

The least absolute shrinkage selector operator (LASSO) is a penalized regression for variable selection proposed by Tibshirani (1996). LASSO solves the following minimization problem:

$$\underset{\beta_0, \boldsymbol{\beta}}{\arg\min} \sum_{i=1}^{N} \left(Y_i - \beta_0 - (\mathbf{Z}\boldsymbol{\beta})_i\right)^2 + \lambda \sum_{j=1}^{q} |\beta_j|, \tag{3}$$

where $(\mathbf{Z}\boldsymbol{\beta})_i$ is the i-th row of $\mathbf{Z}\boldsymbol{\beta}$. In (3), we aim to estimate $\boldsymbol{\beta}$ instead of $\boldsymbol{\beta}_1$, which are the same for the SCAD and MCP objective functions in (4) and (5). These three penalized regression methods do not follow the statistical principles regarding interaction selection described above. Under the LASSO penalty, the regression coefficients of unimportant predictors are shrunk towards zero, with some forced to be exactly zero via a sufficiently large λ. When $\lambda = 0$, optimum $\hat{\boldsymbol{\beta}}$ is equal to the ordinary least squares (OLS) solution. One of the significant limitations of this augmented design LASSO is that it considers interaction terms while ignoring the main effects and vice versa. Further, LASSO can select only a maximum of N variables when $p > N$. Moreover, in the presence of strongly correlated variables, LASSO may choose only one of these variables at random and ignore the others. LASSO excels in high-dimensional settings by creating sparse models and reducing coefficients to zero for effective feature selection. However, it often falls short in identifying interaction terms, particularly when main effects are weak, due to its propensity to select only one variable from groups of correlated predictors.

2.2 Smoothly Clipped Absolute Deviance

The Smoothly Clipped Absolute Deviation (SCAD) penalty, proposed by Fan and Li (2001), is a nonconvex regularization technique used in regression models to prevent overfitting by penalizing large coefficients. Unlike L_1 regularization, which imposes a fixed penalty regardless of the β size, SCAD has a unique property that can be advantageous in scenarios where overly penalizing large β is a concern. The key feature of SCAD that makes it useful for not overly penalizing large $\beta's$ is its *flattening* effect for $\beta's$ beyond a certain threshold. The SCAD method solves the following optimization problem:

$$\underset{\beta_0,\boldsymbol{\beta}}{\arg\min} \sum_{i=1}^{N} \left(Y_i - \beta_0 - (\mathbf{Z}\boldsymbol{\beta})_i\right)^2 + \sum_{j=1}^{q} P_\lambda^{SCAD}(\beta_j), \tag{4}$$

where

$$P_\lambda^{SCAD}(\beta_j) = \begin{cases} \lambda|\beta_j|, & \text{if } |\beta_j| \leqslant 2\lambda \\ -\left(\frac{|\beta_j|^2 - 2a\lambda|\beta_j| + \lambda^2}{2(a-1)}\right), & \text{if } \lambda \leqslant |\beta_j| \leqslant a\lambda \\ \frac{(a+1)\lambda^2}{2}, & \text{if } |\beta_j| > a\lambda, \end{cases}$$

where λ is the regularization parameter, and a is a predefined constant (often set to 3.7 for SCAD). For small β, the SCAD penalty maintains LASSO's penalization rate (and bias), but as the absolute value $|\beta_j|$ increases, the rate of penalization gradually decreases. The SCAD penalty is continuously differentiable on $(-\infty, 0) \cup (0, \infty)$, but it is singular at zero with zero derivatives outside the range $[-a\lambda, a\lambda]$. Small coefficients are set to zero as a result of this, while a few additional coefficients tend to approach zero while large coefficients are retained.

2.3 Minimax Concave Penalty

The Minimax Concave Penalty (MCP), proposed by (Zhang, 2010), is another popular nonconvex penalty method to achieve variable selection and parameter estimation in high-dimensional data. MCP, similar to SCAD, starts by penalizing coefficients at the same rate as the LASSO, gradually reducing the rate toward zero as the absolute value of the coefficient increases. However, unlike SCAD, MCP immediately relaxes the penalization rate, while SCAD maintains a stable rate before reducing it. Although the unbiasedness and selection criteria we impose on the penalty function prohibit the use of fully convex penalties, MCP achieves sparse convexity to a great extent by constraining the maximum concavity (Zhang, 2010). The augmented design SCAD and MCP have a similar issue to the LASSO in the interaction selection in the sense that they do not consider the hierarchical and marginality principles. It should be noted that while MCP is not entirely convex, it still provides a substantial level of sparsity. Mathematically, the MCP optimization problem is formulated as follows:

$$\underset{\beta_0,\boldsymbol{\beta}}{\arg\min} \sum_{i=1}^{N} \left(Y_i - \beta_0 - (\mathbf{Z}\boldsymbol{\beta})_i\right)^2 + \sum_{j=1}^{q} P_\lambda^{MCP}(\beta_j), \tag{5}$$

where

$$P_\lambda^{MCP}(\beta_j) = \begin{cases} \lambda|\beta_j| - \frac{\beta_j^2}{2a}, & \text{if } |\beta_j| \leqslant a\lambda \\ \frac{1}{2}a\lambda^2, & \text{if } |\beta_j| > a\lambda. \end{cases}$$

SCAD and MCP are known for their oracle properties, meaning that, under certain conditions, they can correctly identify non-zero coefficients that achieve asymptotic normality as sample size increases. On the other hand, LASSO may not possess this property without modifications. Adaptive LASSO (Zou, 2006) was proposed to overcome this issue in LASSO. In practical terms, the choice between LASSO, SCAD, and MCP often comes down to the specific data scenario, the size of the dataset, the computational resources available, and the degree of bias one is willing to accept in the final model. LASSO is typically favored for its simplicity and computational efficiency, while SCAD and MCP are often chosen for their ability to reduce bias and handle complex data structures more effectively (Breheny and Huang, 2015).

2.4 Regularization Path Algorithm Under the Marginality Principle

The RAMP algorithm, proposed by Hao et al. (2018), uses the marginality principle to preserve a model's hierarchical structure following variable selection. The RAMP algorithm can be applied to both linear regression and logistic regression. Consider the interaction model in (1) and define the linear main effects index set $\mathcal{M} = \{1, 2, \ldots, p\}$ and the second-order interaction index set $\mathcal{I} = \{(j,k) : 1 \leqslant j < k \leqslant p\}$. The objective function of the RAMP algorithm can be written as follows:

$$\frac{1}{2N} \sum_{i=1}^{N} \left(Y_i - \beta_0 - \mathbf{X}_i^T \boldsymbol{\beta}_{\mathcal{M}} - (\mathbf{X}_i^T)^{\circ 2}_{\mathcal{M}_{\ell-1}} \boldsymbol{\beta}^{\circ 2}_{\mathcal{M}_{\ell-1}} \right)^2 + \lambda_\ell ||\boldsymbol{\beta}_{\mathcal{H}^c_{\ell-1}}||_1 + \lambda_\ell ||\boldsymbol{\beta}_{\mathcal{M}^{\circ 2}_{\ell-1}}||_1, \tag{6}$$

where the penalty is imposed on the candidate interaction effects. The set $\mathcal{M}_{\ell-1}$ is the active main effects set at iteration $\ell-1$. The set $\mathcal{H}_{\ell-1}$ is defined as the parent (main effects) set corresponding to $\mathcal{I}_{\ell-1}$, the active interaction set at iteration $\ell-1$. More precisely, $\mathcal{H}_{\ell-1}$ comprises the main effects which have at least one associated interaction effect (referred to as a "child") in $\mathcal{I}_{\ell-1}$. Conversely, $\mathcal{H}^c_{\ell-1}$ representing the complement of $\mathcal{H}_{\ell-1}$, consist of those main effects in $\mathcal{M}$ that are not linked by the strong heredity constraint to $\mathcal{H}_{\ell-1}$, i.e., $\mathcal{H}^c_{\ell-1} = \mathcal{M} - \mathcal{H}_{\ell-1}$. Throughout the solution path, $\mathcal{M}_\ell, \mathcal{I}_\ell$ and $\mathcal{H}_\ell$ are iteratively updated until all the main effects and the interaction effects are selected and meet the hierarchy and marginality principles. The RAMP algorithm allows for three penalty functions (LASSO, SCAD, and MCP) and two heredity rules (weak and strong). The RAMP algorithm is given in Algorithm 1.

Algorithm 1 RAMP (Hao et al., 2018).

1: *Initialization*: Set $\lambda_{max} = N^{-1} \max |\mathbf{X}^T \mathbf{y}|$ and $\lambda_{min} = \zeta \lambda_{max}$ with some small $\zeta > 0$.
2: Generate an exponentially decaying sequence $\lambda_{max} = \lambda_1 > \lambda_2 > \cdots > \lambda_{\mathcal{L}} = \lambda_{min}$.
3: Initialize the main effect set $\mathcal{M}_0 = \emptyset$ and interaction effect set $\mathcal{I}_0 = \emptyset$.
4: Path-building: Repeat the following steps for $\ell = 1, 2, \ldots, \mathcal{L}$. Given $\mathcal{M}_{\ell-1}$, $I_{\ell-1}$, $\mathcal{H}_{\ell-1}$, add the possible interactions among main effects in $\mathcal{M}_{\ell-1}$ to the current model.
5: With respect to $(\beta_0, \beta^T_{\mathcal{M}}, \beta^T_{\mathcal{M}^{\circ 2}_{\ell-1}})^T$, we minimize (6).

2.5 Random Forest (RF)

RF is an ensemble machine learning algorithm that constructs multiple decision trees for classification or regression, enhancing prediction accuracy and reducing overfitting by aggregating the outputs from various trees (Breiman, 2001). Each tree in the RF is built using a random subset of data and features, contributing to algorithm diversity. Predictions are made through majority

voting in classification tasks or averaging in regression. Noted for its robustness and effectiveness across different data types, RF can be computationally intensive and less interpretable. The RF algorithm is given in Algorithm 2.

Algorithm 2 Breiman's random forest (Hastie et al., 2004).

Input: Training set $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^{N}$, where $\mathbf{X} = (X_1, X_2, \ldots, X_p)$ is p-predictor vector and Y is the response.

1: Generate $\mathcal{B}$ different bootstrap samples, each with N observations, from the original training dataset.

2: For each bootstrap sample numbered b (where b ranges from 1 to $\mathcal{B}$), construct a decision tree $\hat{f}_b$. This is done by iteratively performing the following steps for each terminal node of the tree until the size of the node is reduced to or below a pre-set minimum, n_{min}

- Randomly choose m out of the total p feature variables.
- Identify and execute the optimal split from among the selected m features.
- Divide the current node into two child nodes.

Output: the ensemble of trees $\hat{f}_1, \hat{f}_2, \hat{f}_3, \ldots, \hat{f}_{\mathcal{B}}$ is used to produce the final prediction for new points as follows:

$$\begin{cases} \frac{1}{\mathcal{B}} \sum_{b=1}^{\mathcal{B}} \hat{f}_b(X) & ; \quad \text{for regression,} \\ \text{Majority vote} = \left\{ \hat{f}_b(X) \right\}_1^{\mathcal{B}} & ; \quad \text{for classification.} \end{cases}$$

A crucial aspect of the RF algorithm involves using a subset of features randomly selected from the entire feature space considered for each tree using a bootstrapped sample to train decorrelated trees. This random subset of features mainly differentiates RF from bagging. While selecting a random subset of features is simple, determining the most effective feature for splitting is more complex. This selection uses different metrics depending on whether the RF is used for classification or regression. The Gini impurity is used to measure the likelihood of incorrect classification at a node by considering the proportion of misclassified observations, thereby guiding the algorithm towards the optimal split for classification trees. For regression trees, the best split is determined by minimizing the total sum of squared deviances between the actual and predicted values (residuals). This approach helps in making accurate and efficient splits for regression tasks. Moreover, before each splitting event, $m \leqslant p$ input variables are randomly chosen as potential candidates. The default setting for m in classification tasks is $\sqrt{p}$, with a minimum node size of one. For regression, m is typically set at $p/3$, and the minimum node size is determined to be five (Hastie et al., 2004).

2.6 iterative Random Forests (iRF)

An alternative to the multiplicative interaction method, such as RAMP, includes a popular tree-based interaction method (Breiman et al., 1984; Breiman, 1996, 2001; Meinshausen, 2010). In particular, RF achieves robust and accurate prediction performance while mitigating overfitting and leveraging high-order interactions. The iterative random forest (iRF) is an extension of the traditional RF to model stable, predictive high-order interactions for classification or regression tasks. The iRF algorithm allows for the iterative refinement of feature importance scores by repeatedly building random forests on reduced feature sets. This iterative process helps identify

and prioritize the most informative features for the task. It can be particularly useful when dealing with high-dimensional datasets with many features, as it can improve algorithm performance and reduce overfitting by focusing on the most relevant features.

The iRF algorithm is a set of feature-weighted ensemble decision trees created to find stable high-order interaction terms. We summarize important steps of the iRF algorithm for a binary classification problem as in Basu et al. (2018). Let $U_i \in \{0, 1\}$ be a binary response, and let $\mathcal{I}_i \subseteq \{1, 2, \ldots, p\}$ be the feature-index subset for $i = 1, 2, \ldots, N$, which are used for identifying features in leaf nodes of RF trees. Let $\mathcal{D} = \{(X_i, Y_i)\}_{i=1}^{N}$ be the training data, where $\mathbf{X} = (X_1, X_2, \ldots, X_p)$ are predictors, and $Y \in \{0, 1\}$ is a binary response. The algorithm detects higher-order interactions via three steps. In the first step, adaptive regularization is applied to the RF fitting through iterative feature re-weighting, which determines the relative importance of features. In the second step, the feature-reweighted RF is utilized to identify prevalent interactions within the RF using a generalized version of the random intersection tree (RIT) method, as proposed by Shah and Meinshausen (2014). More specifically, the computationally efficient RIT algorithm searches order-s interactions that appear with higher frequency in a given class, which leads to the construction of informative interaction set $\mathcal{S} \subseteq \{1, 2, \ldots, p\}$ in class $C \in \{0, 1\}$. The prevalence of an interaction for binary classification is defined as

$$P(\mathcal{S}|U = C) := \frac{\sum_{i=1}^{N} I(\mathcal{S} \subseteq \mathcal{I}_i)}{\sum_{i=1}^{N} I(U_i = C)},$$

where $\mathcal{S} \subseteq \{1, 2, \ldots, p\}$, $I(\cdot)$ denotes an indicator function, and $0 \leqslant sta(\mathcal{S}) \leqslant 1$. The bagged stability score is the proportion of times an interaction appears as an output of RIT in B bootstrap samples, which is defined as

$$sta(\mathcal{S}) = \frac{1}{B} \sum_{b=1}^{B} I(\mathcal{S} \in \mathcal{S}_{(b)}).$$

In our simulation and empirical studies, we identified the interactions with $sta(\mathcal{S}) \geqslant 0.7$. Order-$s$ interaction terms can be addressed and identified through Algorithm 3.

Using bootstrap sampling, the RIT algorithm searches the interaction set $\mathcal{S}$ satisfying the following conditions (see **Algorithm S1** in Supplementary Material):

$$P(\mathcal{S}|U = 0) \leqslant \pi_0 \quad \text{and} \quad P(\mathcal{S}|U = 1) \geqslant \pi_1,$$

where $0 \leqslant \pi_0 < \pi_1 \leqslant 1$. In the last step, bagging evaluates the stability of recovered interactions from bootstrap perturbations.

The iterative random forests offer a dynamic approach to capturing and adapting to interactions in evolving data. Tracking variable importance, employing dynamic feature selection and engineering, using visualization tools, and considering ensemble learning techniques contribute to the algorithm's ability to identify and respond to interaction effects over multiple iterations.

2.7 Prediction Performance Assessment Metrics

Our comparative study will evaluate the prediction performance of the selected methods for regression and classification using the following assessment metrics.

Algorithm 3 iterative random forests (Basu et al., 2018).

Input: $\mathcal{D}, \mathcal{C} \in \{0,1\},\ \mathcal{B}, K\ \ w^{(1)} \leftarrow (1/p, \ldots, 1/p)$

for $k \leftarrow 1$ to K **do**
  Fit $RF(w^{(k)})$ on $\mathcal{D}$ ▷ Iterative reweighted RF
  $w^{(k+1)} \leftarrow$ Gini importance of $RF(w^{(k)})$
end for

for $b \leftarrow 1$ to $\mathcal{B}$ **do**
  Generate bootstrap samples $\mathcal{D}_{(b)}$ of the form $\{X_{b(i)}, Y_{b(i)}\}$ from $\mathcal{D}$
  Fit $RF(w^{(K)})$ on $\mathcal{D}_{(b)}$ ▷ Generalized RIT (through $RF(w^{(K)})$)
  $\mathcal{R}_{(b)} \leftarrow \left\{(\mathcal{I}_i, \mathcal{U}_i) : X_{b(i)} \text{ falls in leaf node } \mathrm{i_t} \text{ of tree t}\right\}$
  $\mathcal{S}_{(b)} \leftarrow RIT(\mathcal{R}_{(b)}, \mathcal{C})$
end for

for $\mathcal{S} \in \cup_{b=1}^{\mathcal{B}} \mathcal{S}_{(b)}$ **do**
  $sta(\mathcal{S}) = \frac{1}{\mathcal{B}} \sum_{b=1}^{\mathcal{B}} I\left[\mathcal{S} \in \mathcal{S}_{(b)}\right]$ ▷ Bagged stability score
end for

Output:

(i) The interaction stability scores: $\left\{\mathcal{S}, sta(\mathcal{S})\right\}_{\mathcal{S} \in \cup_{b=1}^{\mathcal{B}} \mathcal{S}_{(b)}}$

(ii) The trained iRF algorithm: $\left\{RF(w^{(K)})\right\}$.

Regression case: We used the mean squared prediction error (MSE) as the metric for assessing the predictive performance in the regression case, i.e., continuous response:

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^{N} \left(Y_i - \hat{Y}_i\right)^2,$$

where Y_i is the actual response value, $\hat{Y}_i$ is the predicted value, and N is the size of the training/testing set.

Classification case: We use the following metrics:

- Sensitivity (or Recall): measures the proportion of actual positives (true positive cases) correctly identified by the model.

$$\text{Sensitivity (SENS)} = \frac{\text{True Positive (TP)}}{\text{True Positives (TP)} + \text{False Negative (FN)}}$$

- Specificity: measures the proportion of actual negatives (true negative cases) correctly identified by the model.

$$\text{Specificity (SPEC)} = \frac{\text{True Negative (TN)}}{\text{True Negative (TN)} + \text{False Positives (FP)}}$$

- Accuracy: is the proportion of true results (both true positives and true negatives) in the total cases as a measure of the overall correctness of the model.

$$\text{Accuracy (Acc)} = \frac{\text{True Positive (TP)} + \text{True Negative (TN)}}{\text{Total Number of Cases}}$$

- Balanced Accuracy: is the average of sensitivity and specificity, and often used for imbalanced response.

$$\text{Balanced Accuracy (BalAcc)} = \frac{\text{Sensitivity} + \text{Specificity}}{2}$$

- F1-Score: is the harmonic mean of precision and recall, providing a balance between them.

$$\text{F1-score} = 2 \times \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}},$$

  where the precision is defined as TP/(TP+FP).

Finally, we assess variable selection performance using the proportion of accurately captured interaction terms (coverage rate for interaction terms). It is worth noting that higher sensitivity, specificity, F-1 score, and balanced accuracy indicate better predictive performance, whereas lower MSE indicates improved predictive performance.

3 Simulations

In this section, we introduce various simulation experiments and summarize their results to compare the performance of penalty-driven algorithms such as LASSO, SCAD, MCP, and RAMP with two tree-based ensemble algorithms, RF and iRF, under both the regression and classification scenarios. As mentioned in the previous section, while all of the methods are capable of capturing interaction terms in the model, LASSO, SCAD, MCP, and RF were not primarily designed for this purpose, whereas RAMP and iRF were specifically designed for capturing certain types of interactions. We separately consider regression and classification models in the following subsections. The *glmnet* package was used for LASSO implementation (Friedman et al., 2010), the *ncvreg* package was used for SCAD and MCP implementation (Breheny and Huang, 2011), the *randomForest* package was used for RF implementation (Liaw and Wiener, 2002), the *RAMP* package was used for RAMP implementation (Feng et al., 2020), and the *iRF* package was used for RF implementation (Basu and Kumbier, 2018). The tuning parameters for the selected algorithms were determined using the default options provided by each package except for iRF, where the tree depth was set to 20 instead of the default value of 5. We measure predictive performance using the mean squared prediction error (MSE) for the regression case and using accuracy, sensitivity, specificity, F-1 score, and balanced accuracy for the classification case.

3.1 Regression Case

We used four different simulation scenarios for regression-based response cases. The simulation scenarios include: 1) a perfect hierarchical structured model with a multiplicative two-way interaction term; 2) a non-hierarchical model (no main effect term); 3) a nonlinear model that does not conform to obeying the marginality principle and; 4) a nonlinear model that obeys the marginality principle (hierarchical structure). Specifically, we consider the following 4 models:

1. Linear hierarchical model: $Y = \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_{1,2} X_1 X_2 + \varepsilon$,
2. Linear non-hierarchical model: $Y = \beta_3 X_3 + \beta_{1,2} X_1 X_2 + \varepsilon$,
3. Nonlinear non-hierarchical model: $Y = \beta_3 X_3 + \beta_{1,2} I(X_1 > 0.5) I(X_2 > 0.5) + \varepsilon$,
4. Nonlinear hierarchical model: $Y = \beta_1 X_1 + \beta_2 X_2 + \beta_{1,2} X_1 X_2 + \beta_{3,4} I(X_3 > 0.5) I(X_4 > 0.5) + \varepsilon$,

where the parameter vector $\{\beta_1, \beta_2, \beta_3, \beta_{1,2}, \beta_{3,4}\} = \{0.2, 0.3, 0.4, 0.3, 0.3\}$. In all 4 models, the predictors were generated from a multivariate normal distribution with a zero mean vector and

Table 1: Interaction selection coverage of six algorithms under four regression models.

		LASSO	SCAD	MCP	RAMP*	RF	iRF
Models	p	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)
	25	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	0.710 (.083)
Linear	100	**1.000 (.000)**	0.998 (.025)	0.990 (.049)	0.995 (.035)	**1.000 (.000)**	0.597 (.122)
Hierarchical	500	**1.000 (.000)**	0.998 (.025)	0.993 (.043)	0.995 (.035)	**1.000 (.000)**	0.553 (.162)
	1000	**1.000 (.000)**	0.993 (.043)	0.985 (.060)	0.992 (.050)	**1.000 (.000)**	0.477 (.207)
Linear	25	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	0.680 (.241)	**1.000 (.000)**	0.999 (.013)
Non-	100	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	0.570 (.174)	**1.000 (.000)**	**1.000 (.000)**
hierarchical	500	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	0.545 (.144)	**1.000 (.000)**	**1.000 (.000)**
	1000	0.936 (.017)	0.944 (.035)	0.943 (.032)	0.572 (.305)	**0.988 (.011)**	0.397 (.142)
Nonlinear	25	0.425 (.229)	0.425 (.206)	0.425 (.210)	0.360 (.226)	0.300 (.130)	**0.488 (.197)**
Non-	100	0.525 (.110)	0.520 (.099)	0.520 (.099)	0.500 (.124)	**0.857 (.202)**	0.778 (.132)
hierarchical	500	0.500 (.000)	0.500 (.000)	0.500 (.241)	0.500 (.234)	**0.823 (.234)**	0.791 (.302)
	1000	0.399 (.010)	0.458 (.021)	0.364 (.018)	0.319 (.250)	**0.789 (.011)**	0.757 (.118)
	25	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	0.563 (.216)	0.955 (.097)	0.851 (.112)
Nonlinear	100	**1.000 (.000)**	**1.000 (.000)**	0.997 (.033)	0.656 (.295)	0.945 (.104)	0.880 (.213)
Hierarchical	500	**1.000 (.000)**	**1.000 (.000)**	**1.000 (.000)**	0.942 (.160)	0.913 (.135)	0.901 (.235)
	1000	0.982 (.021)	0.998 (.024)	0.933 (.037)	0.612 (.116)	**0.999 (.007)**	0.991 (.110)

*RAMP denotes the RAMP algorithm with weak heredity and LASSO penalty.

a covariance matrix with standard deviation $\sigma = 1$, non-zero covariance among $\{X_1 \ldots, X_5\}$, and zero covariance among all other predictors as shown below (Jain and Xu, 2021):

$$\Sigma = \begin{bmatrix} 1.0 & 0.3 & 0.3 & 0.6 & 0.6 & 0 & \ldots & 0 \\ 0.3 & 1.0 & 0.3 & 0.2 & 0.1 & 0 & \ldots & 0 \\ 0.3 & 0.3 & 1.0 & 0.2 & 0.1 & 0 & \ldots & 0 \\ 0.6 & 0.2 & 0.2 & 1.0 & 0.1 & 0 & \ldots & 0 \\ 0.6 & 0.1 & 0.1 & 0.1 & 1.0 & 0 & \ldots & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & 0 & 0 & 0 & \ldots & 1.0 \end{bmatrix}.$$

Models 1 and 2 appeared in Jain and Xu (2021), whereas models 3 and 4 are similar to the models used by Van der Laan et al. (2007). In all 4 models, the errors, ε, were generated from the normal distribution with mean zero and standard deviation $\sigma = 0.5$. $I(A)$ is the usual indicator function which takes 1 if event A holds and zero otherwise. Under each scenario, we varied the number of predictors, $p = \{25, 100, 500, 1000\}$. In all scenarios, the number of simulation iterations was fixed at $iter = 100$, and the full data sample size was $n = 500$. The full data was partitioned into a 60% training set and a 40% test set in each simulation iteration.

Two evaluation metrics were used to compare the various algorithms under study. The first metric focuses on the efficacy of the algorithm in interaction selection and was defined as the rate at which the true interaction terms present in the underlying model are correctly identified (coverage rate of interaction terms). High coverage means that the algorithms capture most of the interaction terms affecting the outcome. The second metric, taken to be the training and test MSE, was used to assess the algorithm's predictive performance. The results of our simulations for the regression case are presented in Tables 1-2.

In Table 1, RF shows a better interaction selection coverage across the four models except for the nonlinear non-hierarchical case when $p = 25$, where iRF performs better than RF

Table 2: Average MSE for seven algorithms under four regression models with varying number of predictors.

		Linear hier.	Linear non-hier.	Nonlinear non-hier.	Nonlinear hier.
p	Method	Train / Test	Train / Test	Train / Test	Train / Test
	LASSO	0.231 / 0.269	0.240 / 0.262	0.236 / 0.259	0.231 / 0.276
	SCAD	0.231 / 0.257	**0.242 / 0.252**	0.236 / 0.259	0.235 / 0.268
	MCP	0.237 / 0.257	**0.245 / 0.252**	0.237 / 0.259	0.243 / 0.266
25	RAMP-W*	**0.246 / 0.252**	0.301 / 0.314	0.247 / 0.256	0.253 / 0.263
	RAMP-S*	**0.246 / 0.251**	0.334 / 0.341	0.247 / 0.255	0.252 / 0.263
	RF	0.059 / 0.356	0.058 / 0.349	**0.042 / 0.254**	**0.055 / 0.230**
	iRF	0.053 / 0.321	0.055 / 0.339	0.042 / 0.267	0.049 / 0.279
	LASSO	0.219 / 0.286	0.227 / 0.274	0.239 / 0.273	0.228 / 0.288
	SCAD	0.205 / 0.268	**0.228 / 0.258**	0.246 / 0.262	0.218 / 0.272
	MCP	0.229 / 0.265	**0.238 / 0.257**	0.250 / 0.262	0.236 / 0.268
100	RAMP-W	**0.244 / 0.258**	0.313 / 0.335	0.257 / 0.261	0.253 / 0.263
	RAMP-S	**0.244 / 0.258**	0.341 / 0.348	0.257 / 0.261	0.253 / 0.263
	RF	0.059 / 0.384	0.057 / 0.367	**0.045 / 0.260**	**0.055 / 0.261**
	iRF	0.051 / 0.347	0.053 / 0.354	0.041 / 0.281	0.047 / 0.289
	LASSO	0.204 / 0.294	0.211 / 0.281	0.234 / 0.282	0.209 / 0.308
	SCAD	0.140 / 0.270	**0.187 / 0.259**	0.230 / 0.267	0.140 / 0.272
	MCP	0.213 / 0.263	**0.225 / 0.256**	0.245 / 0.266	0.230 / 0.272
500	RAMP-W	**0.244 / 0.247**	0.323 / 0.341	0.256 / 0.264	0.253 / 0.260
	RAMP-S	**0.244 / 0.247**	0.342 / 0.352	0.256 / 0.264	0.253 / 0.260
	RF	0.062 / 0.408	0.057 / 0.385	0.045 / 0.262	0.055 / 0.276
	iRF	0.053 / 0.370	0.050 / 0.371	**0.039 / 0.258**	**0.046 / 0.254**
	LASSO	0.206 / 0.307	0.212 / 0.288	0.238 / 0.280	0.214 / 0.311
	SCAD	0.106 / 0.277	**0.163 / 0.261**	0.221 / 0.266	0.118 / 0.285
	MCP	0.211 / 0.271	**0.222 / 0.258**	0.243 / 0.264	0.220 / 0.279
1000	RAMP-W	**0.247 / 0.254**	0.324 / 0.343	0.260 / 0.261	0.251 / 0.266
	RAMP-S	**0.246 / 0.255**	0.341 / 0.348	0.260 / 0.261	0.252 / 0.269
	RF	0.064 / 0.417	0.057 / 0.385	0.046 / 0.294	0.056 / 0.276
	iRF	0.054 / 0.386	0.050 / 0.368	**0.039 / 0.255**	**0.046 / 0.243**

*RAMP-S and RAMP-W represent the RAMP with strong heredity and RAMP with weak heredity, respectively. The results in bold font demonstrate the best performance in the test data.

and other methods. The RAMP algorithm shows better selection coverage than iRF for the linear hierarchical model. Moreover, the table reveals that other non-hierarchical, penalty-based algorithms nearly perfectly identify interaction terms. Although ensemble methods such as iRF are potent for nonlinear models, they can be computationally intensive, particularly when the predictor count p greatly exceeds the number of observations N. Notably, these methods are not limited to two-way interactions; they can identify and select higher-order interactions.

In terms of predictive accuracy, both the weak and strong RAMP rules demonstrate steady performance across various numbers of predictors (p), with Mean Squared Error (MSE) ranging

Table 3: Conditional probabilities $\pi := P(Y = 1|X)$ for classification models.

Model	**Balanced Case**	**Imbalanced Case**
Linear hierarchical	$\left(1+\exp\left[-(0.5+A)\right]\right)^{-1}$	$\left(1+\exp\left[-(-1+A)\right]\right)^{-1}$
Linear non-hierarchical	$\left(1+\exp\left[-(0.5+B)\right]\right)^{-1}$	$\left(1+\exp\left[-(-1+B)\right]\right)^{-1}$
Nonlinear non-hierarchical	$\left(1+\exp\left[-(0.5+C)\right]\right)^{-1}$	$\left(1+\exp\left[-(-1+C)\right]\right)^{-1}$
Nonlinear hierarchical	$\left(1+\exp\left[-(0.5+D)\right]\right)^{-1}$	$\left(1+\exp\left[-(-1+D)\right]\right)^{-1}$

from 0.244 to 0.264. This consistent performance is noted across all models, with the exception of the linear non-hierarchical model. Additionally, unlike some alternative approaches, the RAMP rules do not exhibit overfitting issues, as indicated in Table 2. In linear hierarchical models, RAMP-W and RAMP-S consistently demonstrate the lowest test MSE across all p values. For linear non-hierarchical models, SCAD and MCP also tend to perform well, but SCAD and MCP show competitive performance on test sets, especially as p increases.

In nonlinear non-hierarchical and hierarchical models, iRF exhibits the best test MSE at higher p values ($p = \{500, 1000\}$), while RF shows better test MSE for the small p cases ($p = \{25, 100\}$). There is an overall trend where simpler methods like LASSO tend to perform less effectively as the complexity of the model (nonlinearity) and dimensionality (p) increase. The advanced ensemble methods (RF and iRF) demonstrate resilience to the increase in p for the nonlinear (non-hierarchical and hierarchical) model, maintaining low MSE values and suggesting their suitability for complex, high-dimensional data modeling. It is also notable that the hierarchical model types do not consistently outperform non-hierarchical ones, as might be expected, which could be due to the specific nature of the data or the interactions being modeled.

3.2 Classification Case

In the classification context, we simulated data considering balanced and imbalanced responses for four models. Set

$$
\begin{aligned}
A &:= \beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3 + \beta_{1,2} X_1 X_2 + \varepsilon, \\
B &:= \beta_3 X_3 + \beta_{1,2} X_1 X_2 + \varepsilon, \\
C &:= \beta_3 X_3 + \beta_{1,2} I(X_1 > 0.5) I(X_2 > 0.5) + \varepsilon, \text{ and} \\
D &:= \beta_1 X_1 + \beta_2 X_2 + \beta_{1,2} X_1 X_2 + \beta_{3,4} I(X_3 > 0.5) I(X_4 > 0.5) + \varepsilon.
\end{aligned}
$$

The binary response variable Y for the classification simulations was generated using the conditional probability defined in Table 3. The predictor variables and the error term for the classification simulations are generated in the same manner as described in the regression case in Section 3.1. Similar to the regression case, the full data in each simulation iteration was split into a 60% training set and a 40% test set. We again use the rate of correctly identifying interaction terms as our interaction selection performance metric and report the results in Table 4. For predictive performance, we calculated five metrics for both the training and testing sets. The five performance metrics comprised accuracy, balanced accuracy, sensitivity, specificity, and F1-score. The predictive performance results for the classification scenarios are summarized in Figures 1-4.

Table 4: Interaction selection coverage of six algorithms under four classification models with a balanced response.

		LASSO	SCAD	MCP	RAMP*	RF	iRF
Method	p	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)	Mean (SD)
	25	0.455 (.321)	0.528 (.243)	0.430 (.207)	0.468 (.263)	**0.753 (.275)**	0.477 (.223)
Linear	100	0.268 (.249)	0.345 (.197)	0.277 (.159)	0.423 (.218)	**0.677 (.253)**	0.415 (.235)
Hierarchical	500	0.125 (.179)	0.175 (.186)	0.163 (.179)	0.423 (.218)	**0.447 (.260)**	NA (NA)
	1000	0.160 (.212)	0.145 (.167)	0.199 (.435)	0.503 (.276)	**0.615 (.153)**	NA (NA)
	25	0.305 (.389)	0.410 (.398)	0.335 (.363)	0.350 (.230)	0.580 (.496)	**0.631 (.409)**
Linear	100	0.200 (.333)	0.265 (.351)	0.195 (.317)	0.305 (.245)	0.560 (.498)	**0.968 (.125)**
Non-hierarchical	500	0.115 (.234)	0.110 (.208)	0.080 (.184)	0.326 (.135)	**0.400 (.492)**	NA (NA)
	1000	0.207 (.119)	0.200 (.170)	0.202 (.165)	0.198 (.201)	**0.603 (.101)**	NA (NA)
	25	0.057 (.143)	0.060 (.145)	0.023 (.085)	0.040 (.136)	0.340 (.206)	**0.538 (.253)**
Nonlinear	100	0.026 (.090)	0.016 (.073)	0.010 (.057)	0.033 (.111)	**0.326 (.211)**	0.114 (.282)
Non-hierarchical	500	0.000 (.000)	0.000 (.000)	0.000 (.000)	0.000 (.000)	**0.173 (.186)**	NA (NA)
	1000	0.000 (.000)	0.000 (.000)	0.000 (.000)	0.000 (.000)	**0.119 (.213)**	NA (NA)
	25	0.233 (.308)	0.313 (.287)	0.223 (.250)	0.333 (.255)	0.327 (.250)	**0.635 (.299)**
Nonlinear	100	0.090 (.211)	0.133 (.211)	0.093 (.171)	0.196 (.246)	0.215 (.222)	**0.333 (.492)**
Hierarchical	500	0.056 (.134)	0.050 (.128)	0.026 (.090)	0.183 (.314)	**0.207 (.209)**	NA (NA)
	1000	0.087 (.211)	0.062 (.221)	0.058 (.206)	0.110 (.351)	**0.188 (.300)**	NA (NA)

*RAMP denotes the RAMP algorithm with weak heredity and LASSO penalty.

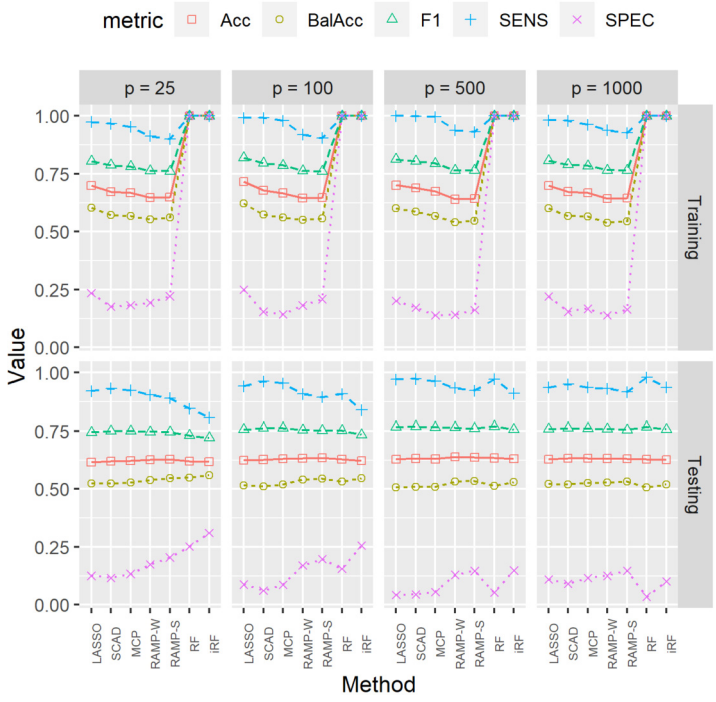

Figure 1: Classification performance for 7 algorithms under a balanced linear hierarchical model. RAMP-W and RAMP-S denote the RAMP algorithm with LASSO penalty under the weak and strong heredity rules, respectively.

We first analyze the interaction selection results. Overall, RF demonstrates superior performance over the RAMP and other penalty-based interaction selection methods. iRF also showed

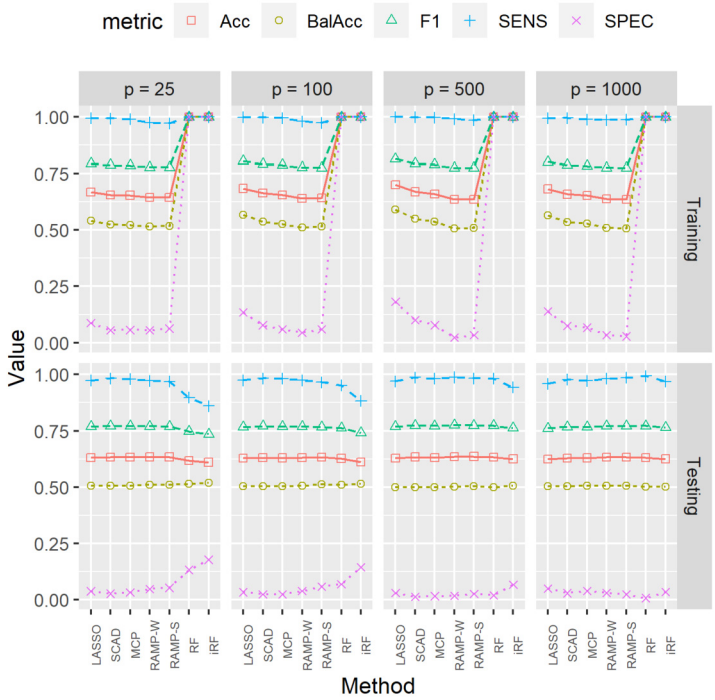

Figure 2: Classification performance for 7 algorithms under a balanced linear non-hierarchical model.

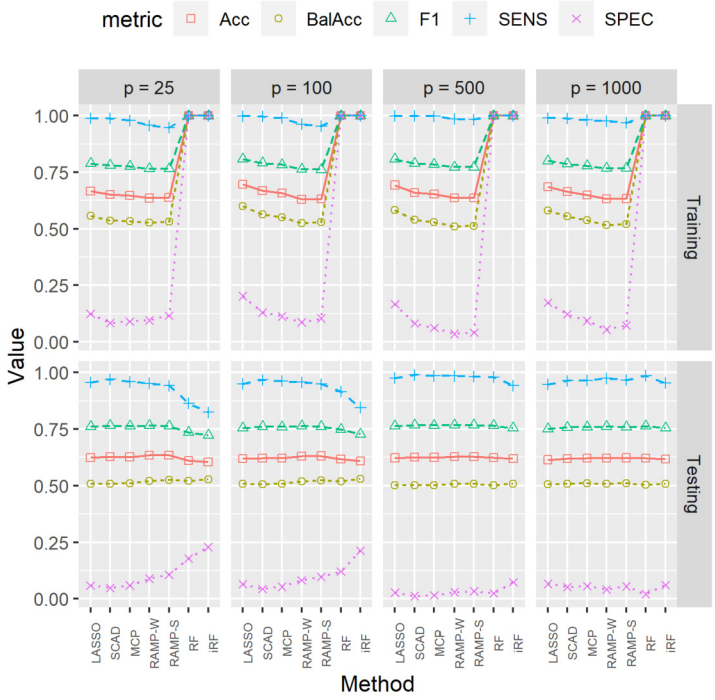

Figure 3: Classification performance for 7 algorithms under a balanced nonlinear non-hierarchical model.

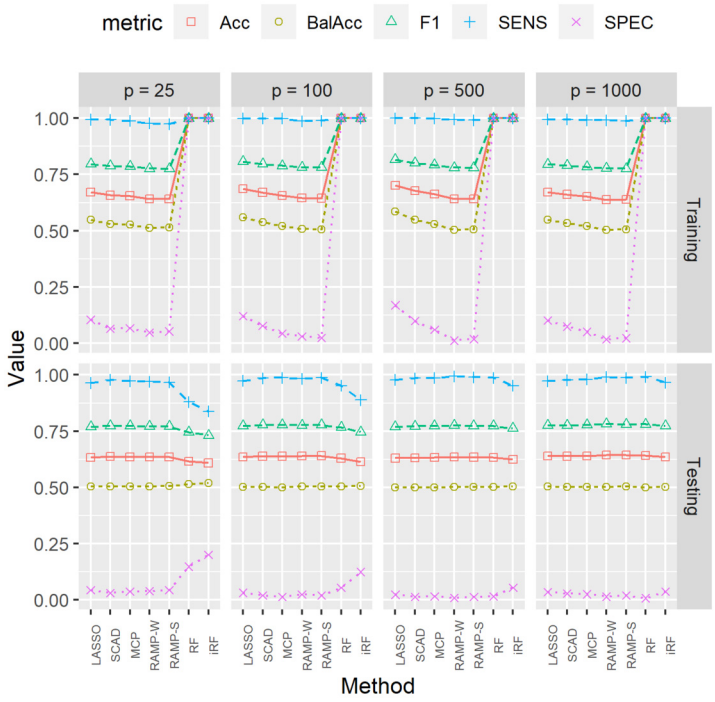

Figure 4: Classification performance for 7 algorithms under a balanced nonlinear hierarchical model.

good coverage results when it didn't suffer from convergence issues, i.e. when $p = \{25, 100\}$. For large values of p (500 and 1000), iRF failed as indicated by the NA's in Table 4. This is likely due to the computational burden associated with such extensive interaction selection for large p cases. Finally, the penalty-based algorithms, LASSO, SCAD, MCP, and RAMP, showed their best performance under the linear hierarchical model, especially for small $p = 25$, but they suffered under other models, especially the nonlinear non-hierarchical model.

We now turn our attention to the predictive performance results in Figures 1-4. Analyzing the training set curves reveals that the RF and iRF algorithms attain robust performance in the metrics, regardless of the number of predictors (p). Among the other five algorithms, LASSO shows slight superiority across the various performance metrics. The four algorithms, SCAD, MCP, RAMP-W, and RAMP-S, have similar performance in terms of accuracy, balanced accuracy, F1 score, and sensitivity. RAMP-W and RAMP-S have lower training specificity than SCAD and MCP under three of the four scenarios considered, especially as p increases.

For the test set results, all algorithms underperform across the various metrics, but they all maintain reasonable performance in terms of sensitivity, F1 score, and accuracy. Despite the big drop in its specificity, iRF maintains the highest specificity rates across all models and dimensions, except for $p = 1000$, where RAMP-S and RAMP-W show competitively higher specificity rates. Another noteworthy observation is how the RF's testing specificity drops for higher dimensions, $p = 500, 1000$. The dramatic change in the performance of the two tree-based algorithms (RF and iRF) when transitioning from training to testing data hints at the potential challenge of overfitting for these algorithms. We also examined the case of imbalanced response for the same four classification models. The results were quite similar to the balanced response case. Thus, these results were deferred to the supplementary material section.

4 Real Data Application

Breast cancer is the most frequent cancer among women, accounting for about one-fifth of all malignancies diagnosed in women worldwide. It is the second most common cause of death from cancer among women. In this section, we compare the performance of the predictive modeling algorithms described in earlier sections using the bcTCGA breast cancer gene expression data from The Cancer Genome Atlas Program (TCGA) of the National Cancer Institute's Center for Cancer Genomics: bcTCGA. The bcTCGA data contain information on 17, 323 gene expressions from 536 women with breast cancer. The BRCA1 gene, a key discovery in breast cancer research, along with the BRCA2 gene (identified a year later), are the core causes for 70% of breast cancer cases. Due to BRCA1 interactions with other genes, identifying genes that interact with it is crucial for further research (Deng and Brodie, 2000; Antoniou et al., 2003; Kuchenbaecker et al., 2017). The selected techniques are particularly valuable in this context for reducing the vast number of potential gene interactions to a more manageable set that is significant for cancer development. These interactions can then be further studied in a laboratory setting and targeted for cancer treatment development. In our regression analysis, the BRCA1 gene expression level serves as the response variable, with the other 17322 genes serving as predictors. This scenario is a typical example of the $p \gg n$ scenario. A two-fold cross-validation approach was used where we repeatedly (100 times) split the dataset into training and testing sets at a ratio of $6:4$. In each iteration, we computed the training and testing MSE for each of the seven predictive algorithms under study. Table 5 displays the average training and testing MSE for the seven algorithms. SCAD outperforms other algorithms with the minimal test MSE observed and higher standard deviation. The RAMP algorithm demonstrates a marginally better test MSE of 0.200 compared to iRF's 0.220. Both RF and iRF exhibit signs of overfitting.

For the classification case, we use the Wisconsin breast cancer dataset available at the UCI Machine Learning Repository (Wolberg et al., 1995). The dataset consists of 569 observations on 30 predictors and one binary response variable representing tumor *diagnosis* with 62.742% benign (non-cancerous) cases and 37.258% malignant (cancerous) cases. The features, which describe characteristics of the cell nuclei, are computed from a digitized image of a fine needle aspirate of a breast mass. The features included radius, texture, perimeter, area, smoothness, etc. This is a typical classification problem that can be handled by any of the predictive modeling algorithms studied here. Similar to the regression case, we used a two-fold cross-validation approach by repeatedly (100 times) splitting the dataset into training and testing sets at a ratio of $6:4$. The predictive performance results averaged over 100 iterations are summarized in Table 6.

Table 5: Average MSE for seven algorithms on the breast cancer (bcTCGA) data with a continuous response.

Method	Train (SD)	Test (SD)
LASSO	0.206 (0.012)	0.214 (0.119)
SCAD	**0.206 (0.101)**	**0.182 (0.215)**
MCP	0.207 (0.011)	0.191 (0.112)
RAMP-W	0.196 (0.022)	0.200 (0.101)
RAMP-S	0.178 (0.042)	0.212 (0.108)
RF	0.038 (0.051)	0.300 (0.121)
iRF	0.039 (0.097)	0.220 (0.186)

Table 6: Classification performance of six algorithms for the Wisconsin breast cancer data.

	LASSO	SCAD	MCP	RAMP**	RF	iRF
Metric*	Train/Test	Train/Test	Train/Test	Train/Test	Train/Test	Train/Test
ACC	0.951/0.943	0.970/0.959	0.966/0.957	**0.974/0.961**	0.999/0.956	1.000/0.947
SENS	0.920/0.905	0.937/0.921	0.934/0.920	**0.959/0.940**	0.998/0.932	1.000/0.924
SPEC	0.969/0.965	0.989/0.960	**0.984/0.979**	0.982/0.973	1.000/0.970	1.000/0.961
F1	0.933/0.921	**0.959/0.949**	0.954/0.940	0.964/0.947	0.999/0.940	1.000/0.928
B.ACC	0.944/0.935	0.963/0.954	0.960/0.949	**0.971/0.957**	0.999/0.951	1.000/0.942

*ACC, SENS, SPEC, F1, B.ACC denote average accuracy, sensitivity, specificity, F1-score, and balanced accuracy, respectively. **RAMP represents RAMP with the weak heredity rule.

As highlighted in the table, RAMP achieved the highest scores in test accuracy, balanced accuracy, and sensitivity. MCP excelled in specificity, and SCAD led in F1 score performance. Notably, SCAD's test accuracy, balanced accuracy, and F1 score results were closely comparable to those of RAMP. Finally, it is readily seen that the tree-based algorithms (RF and iRF) suffered overfitting issues, with the test set results being significantly lower than the training set ones.

5 Discussion

This study compared two predictive modeling approaches that enable interaction selection: the penalty-based approach (LASSO, SCAD, MCP, and RAMP) and the tree-based approach (RF and iRF). Our findings from extensive simulations and real data applications revealed that RAMP had superior predictive performance under linear hierarchical regression scenarios, exhibiting lower MSE across varying dimensions (different values of p). iRF showed signs of overfitting in prediction performance in most regression and classification scenarios, but it was found to be best-suited for nonlinear hierarchical/non-hierarchical large p regression scenarios ($p = \{500, 1000\}$). Similarly, RF suffered from overfitting but was best suited for the nonlinear hierarchical and non-hierarchical small p regression scenarios ($p = \{25, 100\}$). SCAD and MCP performed superiorly under linear non-hierarchical scenarios.

While all of the six techniques under comparison are capable of capturing interactions, RAMP and iRF are specifically designed to capture interactions and, hence, were expected to have superior interaction selection performance. Overall, our interaction selection results revealed mixed results across most scenarios under both regression and classification tasks. For regression tasks, non-hierarchical penalty-based algorithms such as LASSO, SCAD, and MCP also demonstrated superior interaction selection performance compared to RAMP (in most scenarios) and iRF (except for non-linear non-hierarchical models). iRF outperformed RAMP in selecting interactions for regression tasks except under linear hierarchical regression models where RAMP was superior to iRF. In classification tasks, RAMP had superior interaction selection performance to the other penalty-based algorithms, while iRF did not deliver stable results, especially for the large p scenarios.

Taking the above results collectively, we were not be able to declare one algorithm as a clear winner across all or even the majority of scenarios. There were at least one or more scenarios that favored each of the algorithms included in this comparative study. However, from the general pattern, we can recommend 1) the use of RF and iRF for regression/classification tasks

involving nonlinear models, 2) the use of RAMP for regression/classification tasks involving linear hierarchical models, and 3) the use of SCAD or MCP for regression tasks involving linear non-hierarchical models. Future work shall expand this comparative study to include nonparametric interaction selection algorithms with and without hierarchical structures (e.g., Dong and Wu, 2022). Such extension would allow for a better understanding of their potential advantages.

Supplementary Material

The supplementary material includes the following: (1) README: a brief explanation of the supplementary material; (2) application datasets; (3) code files; and (4) the description of the RIT algorithm and additional simulation results.

References

Antoniou A, Pharoah P, Narod S, Risch H, Eyfjörd J, Hopper J, et al. (2003). Average risks of breast and ovarian cancer associated with brca1 or brca2 mutations detected in case series unselected for family history: A combined analysis of 22 studies. *American Journal of Human Genetics*, 72: 1117–1130. https://doi.org/10.1086/375033

Basu S, Kumbier K (2018). *iRF: Iterative Random Forests*. R package version 3.0.0.

Basu S, Kumbier K, Brown JB, Yu B (2018). Iterative random forests to discover predictive and stable high-order interactions. *Proceedings of the National Academy of Sciences*, 115(8): 1943–1948. https://doi.org/10.1073/pnas.1711236115

Bien J, Taylor J, Tibshirani R (2013). A lasso for hierarchical interactions. *The Annals of Statistics*, 41(3): 1111–1141.

Breheny P, Huang J (2011). Coordinate descent algorithms for nonconvex penalized regression, with applications to biological feature selection. *Annals of Applied Statistics*, 5(1): 232–253.

Breheny P, Huang J (2015). Group descent algorithms for nonconvex penalized linear and logistic regression models with grouped predictors. *Statistics and Computing*, 25(2): 173–187. https://doi.org/10.1007/s11222-013-9424-2

Breiman L (1996). Bagging predictors. *Machine Learning*, 24(2): 123–140.

Breiman L (2001). Random forests. *Machine Learning*, 45(1): 5–32. https://doi.org/10.1023/A:1010933404324

Breiman L, Friedman JH, Olshen RA, Stone CJ (1984). Classification and regression trees. *Biometrics*, 40: 874. https://doi.org/10.2307/2530946

Chipman H, Hamada M, Wu CF (1997). A Bayesian variable-selection approach for analyzing designed experiments with complex aliasing. *Technometrics*, 39(4): 372–381. https://doi.org/10.1080/00401706.1997.10485156

Choi N, Li W, Zhu J (2010). Variable selection with the strong heredity constraint and its oracle property. *Journal of the American Statistical Association*, 105: 354–364. https://doi.org/10.1198/jasa.2010.tm08281

Cordell D, Drangert JO, White S (2009). The story of phosphorus: Global food security and food for thought. *Global Environmental Change*, 19: 292–305. https://doi.org/10.1016/j.gloenvcha.2008.10.009

Deng CX, Brodie SG (2000). Roles of brca1 and its interacting proteins. *BioEssays*, 22(8): 728–737. https://doi.org/10.1002/1521-1878(200008)22:8<728::AID-BIES6>3.0.CO;2-B

Dong Y, Wu Y (2022). Nonparametric interaction selection. *Statistica Sinica*, 32: 1563–1582.

Donoho D (2000). High-dimensional data analysis: The curses and blessings of dimensionality. *AMS Math Challenges Lecture*, 1(2000): 1–32.

Evans JD (2006). Beepath: An ordered quantitative-PCR array for exploring honey bee immunity and disease. *Journal of Invertebrate Pathology*, 93(2): 135–139. https://doi.org/10.1016/j.jip.2006.04.004

Fan J, Li R (2001). Variable selection via nonconcave penalized likelihood and its oracle properties. *Journal of the American Statistical Association*, 96(456): 1348–1360. https://doi.org/10.1198/016214501753382273

Fan J, Li R (2006). Statistical Challenges with High Dimensionality: Feature Selection in Knowledge Discovery. In M Sanz-Solé, J Soria, JL Varona & J Verdera. *Proc. Madrid Int. Congress of Mathematicians*, 3: 595–622.

Fan J, Lv J (2010). A selective overview of variable selection in high dimensional feature space. *Statistica Sinica*, 20: 101–148.

Feng Y, Hao N, Helen Zhang H (2020). *RAMP: Regularized Generalized Linear Models with Interaction Effects*. R package version 2.0.2.

Friedman J, Tibshirani R, Hastie T (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of Statistical Software*, 33(1): 1–22. https://doi.org/10.18637/jss.v033.i01

Hao N, Feng Y, Zhang HH (2018). Model selection for high-dimensional quadratic regression via regularization. *Journal of the American Statistical Association*, 113(522): 615–625. https://doi.org/10.1080/01621459.2016.1264956

Hao N, Zhang HH (2014). Interaction screening for ultrahigh-dimensional data. *Journal of the American Statistical Association*, 109(507): 1285–1301. https://doi.org/10.1080/01621459.2014.881741

Hao N, Zhang HH (2017). A note on high-dimensional linear regression with interactions. *American Statistician*, 71(4): 291–297. https://doi.org/10.1080/00031305.2016.1264311

Hastie T, Tibshirani R (1990). Exploring the nature of covariate effects in the proportional hazards model. *Biometrics*, 46(4): 1005–1016. https://doi.org/10.2307/2532444

Hastie T, Tibshirani R, Friedman J, Franklin J (2004). The elements of statistical learning: Data mining, inference, and prediction. *The Mathematical Intelligencer*, 27: 83–85.

Jain R, Xu W (2021). HDSI: High dimensional selection with interactions algorithm on feature selection and testing. *PLoS ONE*, 16(2): e0246159. https://doi.org/10.1371/journal.pone.0246159

Kong Y, Li D, Fan Y, Lv J (2017). Interaction pursuit in high-dimensional multi-response regression via distance correlation. *ArXiv:Methodology*.

Kooperberg C, Leblanc M (2008). Increasing the power of identifying gene-gene interactions in genome-wide association studies. *Genetic Epidemiology*, 32: 255–263. https://doi.org/10.1002/gepi.20300

Kotsiantis S, Kanellopoulos D (2012). Combining bagging, boosting, and random subspace ensembles for regression problems. *International Journal of Innovative Computing, Information & Control: IJICIC*. 3953–3961.

Kuchenbaecker K, Hopper J, Barnes D, Phillips KA, Mooij T, Roos-Blom MJ, et al. (2017). Risks of breast, ovarian, and contralateral breast cancer for brca1 and brca2 mutation carriers. *JAMA*, 317: 2402. https://doi.org/10.1001/jama.2017.7112

Liaw A, Wiener M (2002). Classification and regression by randomforest. *R News*, 2(3): 18–22.

Manolio TA, Collins FS (2007). Genes, environment, health, and disease. *Human Heredity*, 63(2):

63–66. https://doi.org/10.1159/000099178

McCullagh P (2002). What is a statistical model? *The Annals of Statistics*, 30(5): 1225–1267. https://doi.org/10.1214/aos/1035844977

Meinshausen N (2010). Node harvest. *Annals of Applied Statistics*, 4(4): 2049–2072. https://doi.org/10.1214/10-AOAS367

Nelder JA (1977). A reformulation of linear models. *Journal of the Royal Statistical Society. Series A. General*, 140(1): 48–77. https://doi.org/10.2307/2344517

Shah RD, Meinshausen N (2014). Random intersection trees. *Journal of Machine Learning Research*, 15(1): 629–654.

Tibshirani R (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B, Methodological*, 58(1): 267–288. https://doi.org/10.1111/j.2517-6161.1996.tb02080.x

Tin Kam Ho (1998). The random subspace method for constructing decision forests. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(8): 832–844. https://doi.org/10.1109/34.709601

Van der Laan MJ, Polley EC, Hubbard AE (2007). Super learner. *Statistical Applications in Genetics and Molecular Biology*, 6(2007): 25.

Wolberg W, Mangasarian O, Street N, Street W (1995). *Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository*. DOI. https://doi.org/10.24432/C5DW2B

Yuan M, Joseph VR, Zou H (2009). Structured variable selection and estimation. *Annals of Applied Statistics*, 3(4): 1738–1757. https://doi.org/10.1214/09-AOAS254

Zhang CH (2010). Nearly unbiased variable selection under minimax concave penalty. *The Annals of Statistics*, 38(2): 894–942. https://doi.org/10.1214/09-AOS729

Zhao P, Rocha G, Yu B (2009). The composite absolute penalties family for grouped and hierarchical variable selection. *The Annals of Statistics*, 37, No. 6A: 3468–3497. https://doi.org/10.1214/07-AOS584

Zou H (2006). The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, 101(476): 1418–1429. https://doi.org/10.1198/016214506000000735