Information Sciences 574 (2021) 590–605

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

RHDSI: A novel dimensionality reduction based algorithm on high dimensional feature selection with interactions

Rahi Jain [a], Wei Xu [a,b,*]

[a] Biostatistics Department, Princess Margaret Cancer Research Centre, Toronto, Ontario, Canada
[b] Dalla Lana School of Public Health, University of Toronto, Toronto, Ontario, Canada

ARTICLE INFO

Article history:
Received 29 December 2020
Received in revised form 8 June 2021
Accepted 26 June 2021
Available online 6 July 2021

Keywords:
Interaction terms
High-dimensional data
Feature selection
Dimensionality reduction
Machine learning
Regression

ABSTRACT

Classical statistical learning techniques struggle to perform feature selection in high-dimensional data that includes interaction effects i.e., when independent feature/s influence the effect of another feature on study outcome. Methods like penalized regression and sparse partial least squares regression can help, but penalization restricts the handling of interaction terms. This study proposes a novel Dimensionality Reduction based algorithm on High Dimensional feature Selection with Interactions (RHDSI), a new feature selection method that integrates dimensionality reduction and machine learning. The method can handle high-dimensional data, incorporate interaction terms and perform statistically-interpretable feature selection; and enables existing classical statistical techniques to work on high-dimensional data. RHDSI performs feature selection in three steps. The first step is a coarse feature selection through dimensionality reduction and statistical modeling on multiple resampled datasets and features, along with their interaction terms. The second step uses pooled results for unsupervised statistical learning-based feature refinement. Finally, supervised statistical learning-based feature selection is performed on the refined feature set to identify the final features with interactions. We evaluate the performance of this algorithm on simulated data and real studies. RHDSI shows better or par performance compared to standard feature selection algorithms like LASSO, subset selection, and sparse PLS.

1. Introduction

Data analysis still relies heavily on classical statistical models. However, these classical models are reaching their limitations with an increase in feature dimension p, as datasets get larger and more complex. Well-documented issues such as false discovery rate [1], degrees of freedom, the singularity of predictors in a model [2], and computational complexity [1] arise when p is large, and especially when p is greater than the sample size (n). Increasingly, it is also necessary to consider interaction terms in analysis models [3] and this can cause an exponential increase in p.

Two of the conventional approaches used to handle high dimensionality data are feature selection (FS) and dimensionality reduction (DR). They aim to find the final predictor space, q_{ff}, for building a model such that $q_{ff} < p$. The critical difference between these approaches is that FS selects a subset of p features as k predictors while DR linearly (or non-linearly) transforms p features into q_{ff}predictors [4].

* Corresponding author at: Biostatistics Department, Princess Margaret Cancer Research Centre, Toronto, Ontario, Canada.
E-mail addresses: Rahi.Jain@uhnresearch.ca (R. Jain), wei.xu@uhnres.utoronto.ca (W. Xu).

https://doi.org/10.1016/j.ins.2021.06.096
0020-0255/© 2021 The Author(s). Published by Elsevier Inc.

1.1. Feature selection

Literature provides different strategies for performing FS. A common approach is to use domain experts to select features based on their domain knowledge [5,6]. Another is to use statistical models for FS: specific techniques may focus on an intrinsic property of features [7,8], the statistical significance of features [6,8,9] or the feature importance in model performance [3,10,11]. These statistical models can also incorporate domain knowledge in FS [12,13].

More recently, machine learning (ML) techniques have been used to handle FS in high dimensional datasets. Random Forest is a common ML technique that focuses on selecting features based on their importance in model performance [14]. ML techniques often use ensemble methods like bagging [15], random subspace sampling [16] or both [17] for model-building as they help create robust models for noisy datasets [18]. Other techniques in the literature, such as BoLASSO [19], regRSM [20] and random LASSO [21], integrate ensemble methods with classical statistical techniques.

These current FS techniques all have certain limitations: Domain expert-based FS is subjective and challenging for a feature set with high dimensions and/or interaction terms; statistics-based approaches struggle to handle datasets with $p > n$; the subset selection approach can work only if $p << n$; and LASSO regression cannot select features more than n [11]. Further, even in datasets with $p < n$, these methods may not help discriminate between target and noise variables [21]. Ensemble methods make ML-based techniques computationally intensive for high-dimensional data [21].

1.2. Dimensionality reduction

DR approaches can be classified into unsupervised dimensionality reduction and supervised dimensionality reduction methods. Unsupervised DR techniques transform the original feature set into a reduced feature space without using the outcome variable. Principal Component Analysis is one commonly used unsupervised DR technique. Supervised DR techniques consider the relationship between the outcome variable and original input features while creating a reduced feature space from the existing feature set. Partial Least Squares (PLS) is a common supervised DR technique [22].

One fundamental limitation of DR techniques is that they lose the original feature set in the DR output, preventing downstream analysis using the original features. While supervised DR tries to preserve the internal relationships between the outcome variable and the original feature set, unsupervised DR cannot. Thus, features obtained from unsupervised DR may not be suitable for interpretation studies. However, some approaches, like principal components regression [23] and partial least squares regression [24], can always identify the effect of the original feature set on the outcome variable from the transformed feature set. In those cases, we get the original feature set with no feature selection.

1.3. Combined feature selection and dimensionality reduction

The methodologies using a combination of FS and DR strategy are also available in the literature. One approach is using FS and DR strategies in sequence [25] by first performing a coarse Restricted Boltzmann machines-based FS, followed by DR on the reduced feature set with partial least squares regression, to give a low-dimension transformed feature set. This approach cannot perform complete FS on the original feature set. A more recent method, sparse partial least squares regression (SPLS), focuses on performing FS and DR simultaneously [26]. In this technique, either the partial least squares step or the regression step performs the penalization. However, no guideline is available for how to handle interaction terms with this method.

The above techniques do not handle interaction effects adequately. For example, LASSO-based methods of FS only keep significant terms in the model. This can be an issue in scenarios with non-significant marginal features, as these methods may select only interaction features. Thus, we need a feature selection method which can reduce the feature space containing both marginal (p) and interaction terms (χ) to smaller feature space (q_{ff}) for building a parsimonious model with outcome, y with minimum error.

$$y = f(q_{ff})|q_{ff} \in (p \cup \chi) \tag{1}$$

$$\min \varphi(y, f(q_{ff}))$$

Where, f represents model building function and φ represents the error function. This paper focuses on a new method for high dimensional feature selection that considers interaction effects during FS and DR. Our proposed strategy, named dimensionality Reduction based algorithm on High Dimensional feature Selection with Interactions (RHDSI) algorithm, combines bootstrapping and random subspace sampling with supervised and unsupervised dimensionality reduction-based regression and an in-built ability to handle interaction terms.

To our knowledge, this is the first methodology paper on this topic. RHDSI is unique as it combines four different feature selection principles, i.e., dimensionality reduction, statistical feature selection, unsupervised learning, and supervised learning, in a single feature selection pipeline. The pipeline leverages the feature selection capability of existing statistical and machine learning techniques in a modular fashion that can be tailored to specific research needs. RHDSI reduces the feature space a model has to process by generating interaction terms post-sampling and identifies both the marginal and the interaction terms in the dataset. Further, RHDSI transforms simple rule-based statistical techniques into a machine learning framework by adding randomness to the dataset to create multiple models and performing a pooled analysis.

The organization of the paper is as follows. The Methodology Section explains and contextualizes the RHDSI algorithm. The Simulation Studies and Real Studies section evaluates the performance of RHDSI in simulations and with real studies, against existing FS methodologies. Finally, the Conclusion and Discussion Section summarizes our work and highlights future directions.

2. Methodology

Some of the existing algorithms, such as LASSO and Partial Least Squares are described in the Supplementary file (Supplementary Section 1). The standard linear statistical models for FS, such as subset selection and penalized regression (SPLS and LASSO), have limited effectiveness in dealing with interaction terms. Furthermore, these models are only effective when $p < n$ and fail or have limited functionality when $p \not< n$. Additionally, statistical significance for the selected features may not be known. The combination of FS and DR techniques addresses many of these challenges, but handling interaction terms is difficult for L1-penalty based FS, especially when interpretability of models is required.

The RHDSI approach is a new methodology for combining FS and DR techniques to perform feature selection which addresses the above limitations. Fig. 1 provides a graphical representation of the architecture of RHDSI. It is a three-step procedure that first performs coarse FS using a combination of FS and DR techniques, followed by unsupervised statistical learning-based refined FS and ending with supervised statistical learning-based FS in the third step.

The coarse FS step first uses bootstrapping and random subspace sampling to generate random samples and features, respectively. The algorithm then generates interaction terms in each sample, followed by dimensionality reduction, statistical modeling on the reduced dimension, and feature recovery to get feature estimates. Finally, it pools the results from all samples to estimate and select statistically significant features. The refined FS step reduces features selected in the coarse FS step using unsupervised statistical learning. The final FS step performs supervised statistical learning on the further reduced features from the refined FS step and selects the final features. At the end of this process, we can estimate the final coefficients of selected features using an appropriate statistical learning technique. We describe the proposed method below for more details.

2.1. Coarse feature selection

RHDSI initiates with Coarse FS step as described below.

2.1.1. Sampling and interaction effects

The algorithm generates random sample datasets, $1, 2, \cdots, B$, from the original dataset of sample size, n, and feature space, p. Each sample dataset is of the same sample size, n, by randomly sampling from the original dataset with replacement following the bootstrap sampling algorithm. The bootstrapping dataset has feature space, $q\ (q < p)$, by randomly sampling the p features without replacement using random subspace sampling. All interaction terms, $\chi = \bigcup_{k=2}^{\omega} \binom{q}{k}$, up to$\omega$ level for the q sampled features are created and incorporated into the sample feature set to create the final sample feature set, $q^{*}(= q \cup \chi)$. We discuss the choice of hyperparameter q in Section 2.3.4. Sampling creates datasets with smaller feature spaces than the original and disrupts multicollinearity between features, enabling the use of existing statistical techniques on high-dimensional data.

2.1.2. Modeling

Separate models are prepared for each dataset in B. The modeling is performed in three parts, namely dimensionality reduction, statistical modeling and feature recovery. Dimensionality reduction reduces q^{*}features in a sample to a smaller number of latent dimensions.

$$Lf_r = f\left(w_{jr}, x_j\right) \tag{2}$$

where, Lf_r is r latent dimension created from q^{*} features in a sample, w_{jr} is the weight given to the j feature for r latent dimension and x_j is the data for j feature. Function f determines the feature weights and changes with dimensionality reduction techniques. The number of latent dimensions (Lf) is the hyperparameter that needs to be pre-defined. The two-step reduction in dataset dimension by sampling and latent features generation allows the incorporation of interaction terms in the statistical model without increasing the feature space.

The algorithm prepares a statistical model to determine the performance of latent dimensions on the outcome. The statistical model, $y = g(u_r Lf_r)$, uses the outcome variable y and latent dimensions $Lf_r$to estimate the coefficients of the latent dimensions u_r. The function g will depend on the statistical model used. Finally, the estimated coefficient of original features, $\widehat{\beta}_i$ for i dataset in B, is determined in the feature recovery phase. Feature recovery transforms coefficients of the latent factor-based model back to original features. These coefficients can provide feature performance in the model. In regression-based models, features with coefficient value zero or close to zero will have no or limited influence on the outcome.

$$y = h\left(\widehat{\beta}_{ij} x_j\right) \tag{3}$$

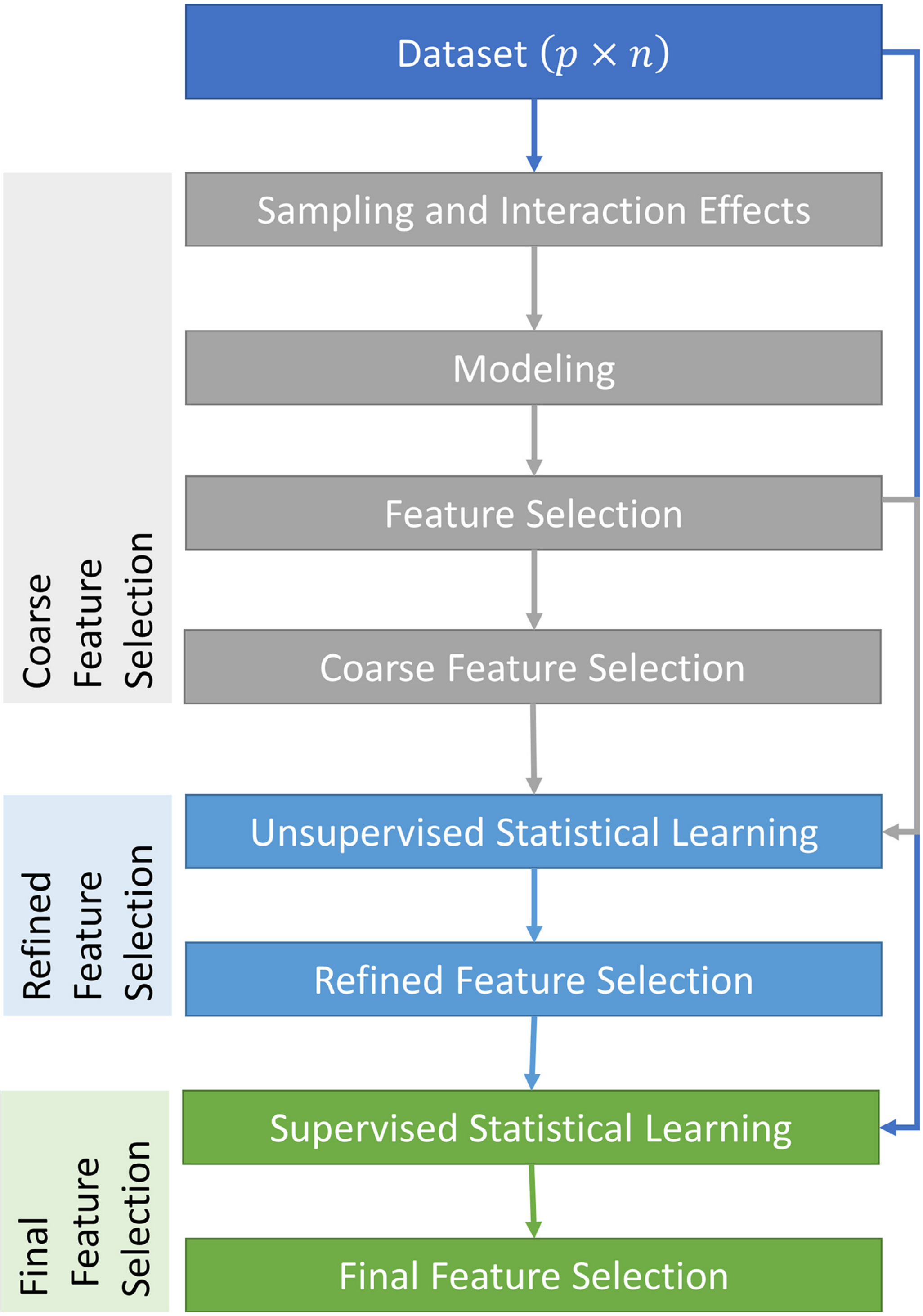
Dataset $(p \times n)$
Sampling and Interaction Effects
Modeling
Feature Selection
Coarse Feature Selection
Unsupervised Statistical Learning
Refined Feature Selection
Supervised Statistical Learning
Final Feature Selection
Coarse Feature Selection
Refined Feature Selection
Final Feature Selection

Fig. 1. Graphical representation of RHDSI methodology. p: number of features in the dataset, n: sample size of the dataset.

$$g(u_r f(w_{jr}, x_j)) = h\left(\widehat{\beta}_{ij} x_j\right) \tag{4}$$

In cases where functions f, g and h linearly combine variables, $\widehat{\beta}_i$ can be estimated from Eq. (6). In the current study, for simplicity, we only consider linear combinations.

$$\widehat{\beta}_{ij} = \sum_{s=1}^{r} u_s w_{js} \tag{5}$$

Where, $\widehat{\beta}_{ij}$ is the $\widehat{\beta}_i$ value of j feature. u_s and w_{js} are estimated coefficient and feature weight respectively for s latent dimension.

2.1.3. Feature selection

The study assumes that the target features should influence the outcome; hence, they should have non-zero coefficient values when averaged across the models. Conversely, noise features should not influence the outcome; hence, they should have a zero coefficient value when averaged across the models. After modeling, j feature pooled coefficient estimate, $\widehat{\beta}_{pool_j} = \sum_{i=1}^{B} \widehat{\beta}_{ij}/b_j$, is obtained by averaging the coefficient estimates $\widehat{\beta}_{ij}$ of j feature over the number of samples containing j feature, b_j. The study uses b_j as the denominator instead of B to reduce the bias, since, with B, features not selected in the dataset are treated as a feature with a zero coefficient rather than as a missing value.

Our algorithm selects features with significant non-zero $\widehat{\beta}_{pool}$and uses them as inputs for the Refined FS step. To estimate the non-zero statistical significance of $\widehat{\beta}_{pool}$, the algorithm calculates the confidence interval (CI) of $\widehat{\beta}_{pool}$. It selects features without zero in their $\widehat{\beta}_{pool}$confidence interval. The confidence level (CL) influences the selection or rejection of target features and changes with the study dataset. A hyperparameter, namely CL, is used to estimate the z-value for the confidence interval.

$$CI_j = \ \widehat{\beta}_{pool_j} \pm \ z_{CL} * \sigma_{\ \widehat{\beta}_j} / \sqrt{b_j} \tag{6}$$

where, CI_j is the confidence interval of $\widehat{\beta}_{pool}$ for j feature, $z$is the z-score at CLhyperparameter and $\sigma_{\widehat{\beta}_j}$ is the standard deviation of the coefficient estimates for j feature. If selected features contain interaction terms without their marginal features, then the algorithm selects its marginal features with their $\widehat{\beta}_{pool}$ as the input for Refined FS.

2.1.4. Number of bootstrap datasets B

The initial coarse feature selection depends on the pooling of the feature coefficient from different models. Therefore, it is critical to determine the minimum number of feature samples needed to perform the statistical interpretation of the pooled estimate of a feature coefficient. RHDSI assumes that the population mean coefficient value of each noise feature is zero. This is used as a reference value to compare the pooled estimate of corresponding feature coefficients. The minimum sample size of the coefficient estimates, $L = 8/\Delta^2$, is obtained using Lehr's equation one-sample case [27] based on given effect size Δ, which is suggested to be 0.2, 0.5 and 0.8 for 'small', 'medium' and 'large' effect sizes respectively [28]. Since sampling is random, the feature probability of getting selected in q features is$\rho = q/p$. For features with interaction terms, $\rho = \chi/\sum_{k=2}^{\omega}\binom{p}{k}$, we do not consider marginal features, as they would be selected with their respective interaction terms.

Since each sample in B is random, we treat each sample as an independent trial to select a feature. The probability of selecting a feature L times, $Pr(X = L) = \binom{B}{L}\rho^L(1-\rho)^{B-L}$, in B trials is the probability mass function of a binomial distribution. L is the minimum sample size for a feature, so B is calculated using the cumulative distribution function. Eq. (9) estimates B for a value of ρ and L,

$$\Pr(X \geq L) = 1 - \sum_{m=0}^{L-1}\Pr(X = m) \tag{7}$$

which can be rearranged to create Eq. (10).

$$B \geq f(\Pr(X \geq L), L) \tag{8}$$

2.2. Refined feature selection

In the second phase of the RHDSI algorithm, the features selected in the coarse FS step, q_c, are evaluated using unsupervised statistical learning techniques to obtain a refined set of features. This step enables an additional reduction in noise features by assuming a smaller coefficient of variation (CV) for target features than noise features. The CV of q_c features, $CV_{j_c} = \sigma_{\widehat{\beta}_{j_c}} / \widehat{\beta}_{pool_{j_c}}$, is used for unsupervised statistical learning, whereCV_{j_c} is the coefficient of variation, $\sigma_{\widehat{\beta}_{j_c}}$ is the standard deviation and $\widehat{\beta}_{pool_{j_c}}$ is the pooled coefficient estimate of j_c feature coefficient estimate in q_c features. The cut-off value for CV depends on the dataset. We use clustering, unsupervised statistical learning, approach to identify the potential cut-off points. One can use different clustering approaches (centroid-based, density-based, model-based or connectivity-based) [29] to cluster the CV. In this study, we use K-means clustering, a commonly used centroid-based clustering algorithm, to partition CV into clusters with the nearest mean value. This approach provides local rather than global optima [30] and is sensitive to high-dimensional data [31] and outliers [32]. The current study avoids issues relating to high-dimensional data by using only low-dimensional data (i.e., performance metrics, CV) for clustering. Further, as a result of the coarse feature selection step removing noisy features from the data, the risk of outliers is also reduced.

Among the K-means clusters, we select the cluster with the smallest mean *CV* value (C_s) and use its features (q_{cu}) for the final feature selection step. In the current study, the algorithm pre-defines the number of clusters equal to two. The C_s may not always contain the target features, as K-means clustering may get stuck in local optima. Thus, a hyperparameter, namely cluster size (*cs*), is used to define the desirable number of features in C_s. The algorithm adds or removes features from C_s to make the number of features in C_s equal to *cs*. These features must have *CV* greater than the features remaining in C_s but less than the features not in C_s.

2.3. Final feature selection

Features selected after coarse FS and unsupervised refinement, q_{cu}, undergo a final filtering process to produce the final features, q_{ff}. This filtering process uses supervised statistical learning techniques like stepwise regression, LASSO and elastic net. If the supervised FS algorithm selects interaction terms without marginal features, it will also add these unselected marginal features to the final feature set. Pseudo Algorithm summarizes the complete RHDSI algorithm.

Pseudo Algorithm: RHDSI

```
Input: Feature data X (p × n)
Target feature Y (1 × n)
Number of bootstrap replicates $B$
Number of randomly sampled features for each bootstrap replicate $q$;
Number of latent factors $Lf$
Confidence Level $CL$
Cluster size $cs$
Coarse selected features list $q_c = \{empty\}$
Refined selected features list $q_{cu} = \{empty\}$
Final selected features list $q_{ff} = \{empty\}$
Output: Final Feature set $q_{ff}$
Begin:
# Step I: Coarse Feature Selection
for $i = 1$ to $B$ do
Generate bootstrap samples $\left(X^i, Y^i \in R^{n\times(p+1)}\right)$
Generate $q$ random features from $p$
Generate all interaction terms, $\chi$, for $q$ features
Compute final feature space, $q^* = q \cup \chi$
Generate latent dimensions $Lf$ from features $\left(q^{*i} \in R^{n\times q^*}\right)$ with weight $w_{ijr}|r = \{1, \cdots, Lf\}, j = \{1, \cdots, q^*\}$
Compute ordinary least squares (or other statistical regression modeling technique) estimate $\widehat{u}_i$ from $\left(Lf^i, Y^i\right)$
Compute $q^*$ coefficient estimate $\widehat{\beta}_{ij} = \left\{\sum_{r=1}^{Lf} u_{ir} w_{ijr} | j = \{1, \cdots, q^*\}\right\}$
end for
for $j = 1$ to $(1, \cdots, p, \cdots, \sum_{k=2}^{\omega} \binom{p}{k})$
Compute pooled coefficient estimate $\widehat{\beta}_{pool_j} = \sum_{i=1}^{B} \widehat{\beta}_{ij}/b_j$, where $b_j$ is the number of bootstrap samples containing the $j$ feature
Compute standard deviation of $\widehat{\beta}_{pool_j}$ $\sigma_{\widehat{\beta}_j}$
Compute Z score for confidence level $(CL) z_{CL}$
Compute confidence interval of pooled coefficient estimate, $CI_j = \widehat{\beta}_{pool_j} \pm \left(z_{CL} \times \sigma_{\widehat{\beta}_j}/\sqrt{b_j}\right)$
if $0 < CI_j$ or $0 > CI_j$
Add $j$ to $q_c$ feature list
end for
Add missing marginal features for selected interaction terms in $q_c$
# Step II: Refined Feature Selection
for $j = 1$ to $q_c$
Compute the coefficient of variation $CV_j = \sigma_{\widehat{\beta}_j}/\widehat{\beta}_{pool_j}$
end for
Generate two clusters $(C_1, C_2)$ from $CV$of $q_c$ features using clustering technique like K-means
```

(*continued on next page*)

```
Compute mean $CV$ of $C_1$ $mC_1$ = MEAN($C_1$)
Compute mean $CV$ of $C_2$ $mC_2$ = MEAN($C_2$)
**if** $mC_1$ = MINIMUM($mC_1$,$mC_2$)
selectedcluster$C_s = C_1$
**else if** $mC_2$ = MINIMUM($mC_1$,$mC_2$)
selectedcluster$C_s = C_2$
**if** LENGTH($C_s$) < $cs$
Add more features to $C_s$ such that $\{CV_{ns} > CV_{af} > CV_{C_s}\}$, where $CV_{ns}$ is $CV$of unselected feature, $CV_{af}$ is $CV$of added
   feature and $CV_{C_s}$ is $CV$of features in $C_s$..Add $C_s$ features to $q_{cu}$
# Step III: Final Feature Selection
Compute supervised learning (like penalized, stepwise regression) estimate $\widehat{w}_{ffs}$ from $\left(X, Y \in R^{n\times(q_{cu}+1)}\right)$
**for** j = 1 **to** $q_{cu}$
**if** $\widehat{w}_{ffs_j} \neq 0$
Add $j$ to $q_{ff}$ feature list
Add missing marginal features for selected interaction terms in $q_{ff}$to get final feature selection
**End**
```

2.4. Feature estimation

Once the final features have been selected, these features can be fed into an appropriate statistical model to estimate the feature coefficients. The selection of statistical technique for feature estimation is flexible, and is best determined by the user based on their problem statement. The current study uses bootstrapping-based ordinary least squares regression to obtain feature coefficient estimates. We prepared ensemble models of ordinary least square regression using the final selected features and the bootstrap datasets generated during coarse FS. Each feature coefficient, β^{SF} is the pooled average of the coefficient value obtained from each model.

$$\beta_j^{SF} = \sum_{i=1}^{B} \beta_{ij}^{SF} / B \tag{9}$$

where β_j^{SF} represents the coefficient estimate of j selected feature.

2.5. Hyperparameter selection

RHDSI uses four hyperparameters, namely number of features in a sample (q), number of latent factors (Lf), confidence level (CL) and cluster size (cs). The value of each hyperparameter depends on the study data. Therefore, it is vital to perform the hyperparameter optimization before performing the FS. In smaller studies, a grid search or trial-and-error approach could perform hyperparameter optimization. However, larger studies need an optimization algorithm.

The Genetic Algorithm (GA) is a metaheuristic optimization technique known for providing good approximations of optimal values for many optimization problems [33]. This approach takes inspiration from the natural selection process. In brief (Supplementary Fig. 1), we create multiple combinations of hyperparameter values as initial samples (*parent population*) and evaluate their performance (*fitness*). The algorithm generates more samples (*offspring population*) from initial samples (*parent population*) based on their performance. It obtains convergence by recombining the *parent population* parameters (*crossover*), while diversity in samples is obtained by randomly changing the parameter values (*mutation*). The process is iterative and stops after achieving desirable performance. The current study uses GA for optimizing the hyperparameters on the training dataset. RHDSI is performed on each combination of hyperparameter (*parent* or *offspring population*). Root mean square error (*RMSE*) is the fitness function determined from the feature estimation step of RHDSI. The robustness of the hyperparameter combination is ensured by performing cross-validation and estimating *RMSE* on the left-out sample. The mean of *RMSE* is the *fitness* value. The study uses the five-fold data split and three trials for cross-validation.

3. Simulation studies

We evaluate the performance of the proposed method in simulated studies and compare it with other feature selection methods. RHDSI can be applied to any order of interaction terms; in this study, we use two-way interactions for demonstration. The regression model, $y = \beta_0 + \sum_{i=1}^{p} \beta_i x_i + \frac{1}{2}\sum_{i\neq j,i=1,j=1}^{i=p,j=p} \beta_{ij} x_{ij} + \epsilon$ provides the outcome variable of the simulated data. $\varepsilon\ N(0,\sigma^2)$, $x_i\ N(0,1)$ and $\{x_{ij}\}$ represents the pairwise interactions between features $\{(x_1,x_2),(x_1,x_3),\cdots,(x_{p-1},x_p)\}$. β value will be non zero for true features and zero for noise features. We create four scenarios for generating different datasets from multivariate normal distributions containing feature coefficients with non-zero and zero values (Table 1). The multicollinearity is added in the model using the covariance matrix for features $\{x_1, \cdots, x_p\}$. Scenario I and II uses following covariance matrix, where first five features are collinear:

$$\begin{bmatrix} x_1x_1 & x_1x_2 & . & . & x_1x_5 & . & . & x_1x_p \\ x_2x_1 & x_2x_2 & . & . & x_2x_5 & . & . & . \\ . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . \\ x_5x_1 & x_5x_2 & . & . & x_5x_5 & . & . & . \\ x_6x_1 & x_6x_2 & . & . & x_6x_5 & . & . & . \\ . & . & . & . & . & . & . & . \\ x_px_1 & . & . & . & . & . & . & x_px_p \end{bmatrix} = \begin{bmatrix} 1.0 & 0.3 & 0.3 & 0.6 & 0.6 & 0.0 & . & 0.0 \\ 0.3 & 1.0 & 0.3 & 0.2 & 0.1 & 0.0 & . & 0.0 \\ 0.3 & 0.3 & 1.0 & 0.2 & 0.1 & 0.0 & . & 0.0 \\ 0.6 & 0.2 & 0.2 & 1.0 & 0.1 & 0.0 & . & 0.0 \\ 0.6 & 0.1 & 0.1 & 0.1 & 1.0 & 0.0 & . & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & . & 0.0 \\ . & . & . & . & . & . & . & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & . & 1.0 \end{bmatrix}$$

Scenario III and IV use the following covariance matrix, where all features are collinear:

$$\begin{vmatrix} x_1x_1 & . & . & x_1x_p \\ . & . & . & . \\ . & . & . & . \\ x_px_1 & . & . & x_px_p \end{vmatrix} = \begin{vmatrix} 1.0 & . & . & 0.5 \\ . & . & . & . \\ . & . & . & . \\ 0.5 & . & . & 1.0 \end{vmatrix}$$

The RHDSI approach uses a linear regression technique for statistical modeling in coarse FS and a stepwise regression technique for supervised statistical learning in final FS. PLS performs dimensionality reduction. We refer to this technique as the RHDSI technique.

3.1. GA based hyperparameters

Hyperparameters *q*, *LF*, *CL* and *cs*, are optimized for each scenario and trial through five-fold cross-validation with three iterations on the training data. During the optimization process, we cap the maximum value of *LF* at 10 for computation purposes as the maximum target marginal features used in any scenario is 10. In the case of *cs*, we use another variable, additional feature ($Af = cs - C_s$) in GA optimization and test it in the range of 0 to 10. The hyperparameter combination with the best RMSE performance is selected.

Supplementary Table 1 provides detailed results for each scenario. The optimal values of hyperparameters varied with the scenario. The mean optimal values of *q*, *LF* and *cs* are 10 (5–15), 8 (2–10) and 5 (1–9), respectively. With *CL*, optimal values of mean and range varied with scenarios and seemed to be influenced by the target feature to noise feature ratio. The *CL* threshold became more stringent with an increase in target to noise feature ratio. The results suggest that optimal parameters of *q*, *LF* and *cs* are more stable and robust across scenarios than *CL*. The stability of optimal values reduces the search domain for GA optimization. However, an extensive range of optimal values for all parameters in most scenarios does not eliminate the need for hyperparameter optimization.

3.2. Computation time estimation

We used the Scenario 1 dataset to provide computation time of the RHDSI algorithm on a system with processor Intel® Core(TM) i7-8750H CPU@2.20 GHz with 16 GB RAM on a Windows 10 64-bit operating system. The CPU time for a single trial is 0.08 s.

3.3. RHDSI comparison with standard methods

We compare RHDSI performance with existing standard methods, namely LASSO, adaptive LASSO (ALASSO), group LASSO (GLASSO), forward subset selection, elastic net (Enet), adaptive elastic net (AEnet) and sparse PLS (SPLS) in ten trials. The current study uses inbuilt packages in statistical language R for determining the performance of existing methods. LASSO, ALASSO, Enet and AEnet were performed using the *glmnet* package [34] with ridge LASSO providing weights for ALASSO

Table 1
Description of the simulation data.

Scenario	β(Non-Zero coefficients)	p	Sample Size (n)		σ
			Train	Test	
I	$\{\beta_1, \beta_2, \beta_3, \beta_{12}\} = \{0.2, 0.3, 0.4, 0.3\}$	25	500	500	0.25
II		25	100	500	0.25
III	$\{\beta_i, \beta_{ij} \vert i = \{1, \cdots, 10\}, j = i+1, j < 11\}$ $=\{0.5, -0.5, 0.5, -0.5, \cdots, 0.5\}$	15	500	500	0.25
IV		15	100	500	0.25

and AEnet [35]. GLASSO, forward selection and SPLS were performed using *glinternet* [36], *MASS* [37] and *spls* [38] packages, respectively. Other than *glinternet* [39], no other existing package considers interactions terms automatically, hence all possible two-way interaction pairs were created and entered into the model. RHDSI algorithm implementation is conducted using R (source code link: https://github.com/rahijaingithub/HDSI.DR).

We evaluated the performance of different methods based on two criteria. The first criterion focuses on the ability of different methods to discriminate between target and noise features. It measures selection of true features and rejection of noise features. The second criterion focuses on the predictive performance of different methods. It measures different metrics, namely root mean square error (RMSE), mean square error (MSE), mean absolute error (MAE) and mean absolute percentage error (MAPE) between the estimated outcome and actual outcome of the test data.

Table 2 is a summary of FS performance which shows that existing methods have selected the target features in most cases. However, all these methods have also selected some noise features. The proposed method, RHDSI, successfully selects all the target features and, while it also selects some noise features, the selection frequency of noise features is much less than selection frequency of the target features, and lower than other methods tested. Further, RHDSI may be better than the other methods at selecting interaction features as observed in Scenario III and IV. This improved performance is likely because RHDSI combines many different FS and DR strategies, which reduce noise variables, compared to other methods that use a single FS strategy. The difference in performance is especially evident in high-dimensional data with a large target to sample ratio (Scenarios II, III and IV).

In Scenario IV, multicollinearity is very high and penalty-based techniques (LASSO and Enet methods and SPLS) struggle to separate the marginal target features from noise features. These approaches have identified more than twice the interaction terms as RHDSI and the majority of these are noise. The non-penalized technique (Forward) does not have this issue but struggles to identify the target features. By generating multiple datasets with different feature combinations, RHDSI is able to disrupt multicollinearity. This allows the same feature to be tested on multiple datasets with different feature combinations, improving the detection of target features and reducing noise.

Fig. 2 shows feature selection frequency, i.e., the percentage of trials in which a feature is selected, for all tested methods. With the exception of ALASSO, existing methods could not consistently select target features at a higher frequency than noise features. Forward subset selection could not select even one target feature consistently in Scenario IV, which has a small sample size and a sample size to target feature ratio of five. Most other existing methods have higher selection frequencies for target features compared to noise features in Scenarios I and II which drops in Scenarios III and IV. Only ALASSO and RHDSI could perform consistently in all scenarios, but among the two, RHDSI selects fewer noise features as compared to ALASSO (Table 2). The results suggest that the proposed method performs at par or better than the existing methods.

Fig. 3 and Supplementary Table 2 depicts the performance of different methods in terms of prediction outcome. RMSE, MSE, MAE and MAPE performance of the tested methods suggest that RHDSI method consistently performs better than the existing methods. However, the prediction performance of different models is not drastically different. These findings suggest that the proposed method may provide at par prediction performance or better compared to existing methods. Overall, the proposed method could expand the capability of powerful existing techniques like non-penalized regression, which are limited by dimensionality, to operate in high-dimensional settings.

Table 2
Feature selection performance of different FS methods in simulated scenarios.

Scenario	Trials	Performance (Number of Features Selected)	Target Features	Existing models							RHDSI
				ALASSO	*Forward*	*GLASSO*	*LASSO*	*SPLS*	Enet	AEnet	RHDSI
				Mean (Range)							
I	10	Marginal ($p = 25$)	3	3 (3–5)	4 (3–6)	23 (5–25)	4 (3–5)	3 (3–5)	4 (3–5)	3 (3–5)	4 (3–8)
		Interaction ($\chi = 300$)	1	**1 (1–2)**	1 (1–3)	80 (1–104)	**1 (1–2)**	**1 (1–2)**	**1 (1–2)**	**1 (1–2)**	2 (1–4)
II	10	Marginal ($p = 25$)	3	6 (3–23)	4 (3–7)	23 (5–25)	9 (3–21)	4 (3–10)	10 (3–19)	10 (3–22)	**4 (3–7)**
		Interaction ($\chi = 300$)	1	4 (1–23)	2 (1–7)	54 (1–67)	6 (1–19)	1 (1–4)	6 (1–17)	6 (1–20)	**1 (1–2)**
III	10	Marginal ($p = 15$)	10	13 (10–15)	11 (10–15)	15 (15–15)	14 (13–15)	14 (10–15)	13 (10–15)	14 (13–15)	**11 (10–15)**
		Interaction ($\chi = 105$)	9	12 (9–17)	11 (9–12)	35 (11–44)	21 (10–27)	20 (10–28)	14 (9–19)	22 (11–31)	**10 (9–12)**
IV	10	Marginal ($p = 15$)	10	15 (15–15)	10 (0–15)	15 (14–15)	15 (15–15)	14 (12–15)	15 (15–15)	15 (15–15)	**13 (11–15)**
		Interaction ($\chi = 105$)	9	31 (20–41)	12 (0–18)	40 (22–51)	33 (18–49)	36 (16–102)	32 (24–41)	34 (21–44)	**14 (9–21)**

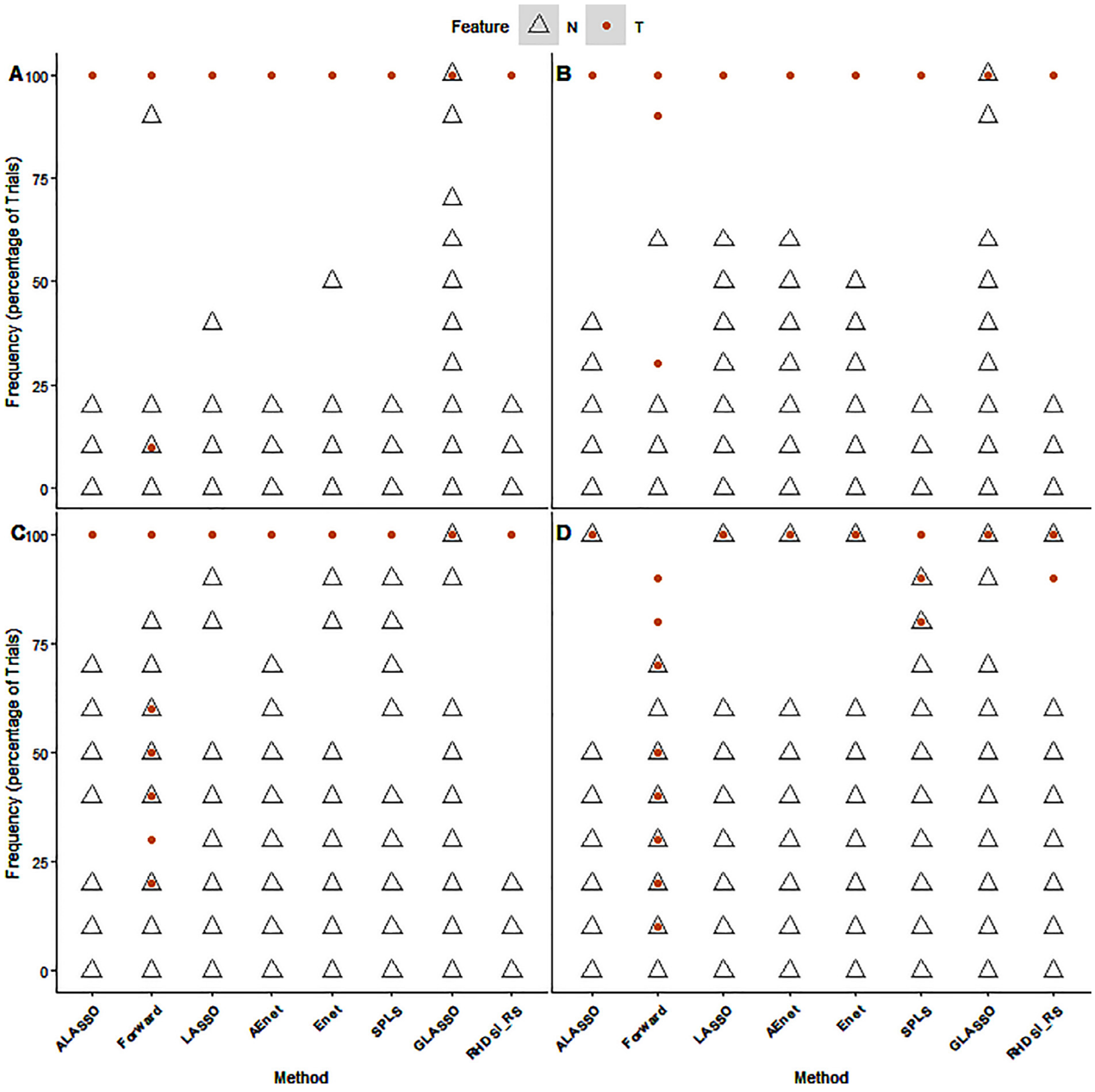

Fig. 2. Performance of different methods in selecting target features. A: Scenario I, B: Scenario II, C: Scenario III, D: Scenario IV, N: Noise Features and T: Target Features.

4. Real data studies

The current study also uses four real datasets to compare the proposed approach and existing methods. Dataset A is the Community Health Status Indicators (CHSI) dataset with data for 3141 US counties on 578 features related to non-communicable diseases [40]. Datasets B and C are National Social Life, Health and Aging Project (NSHAP) datasets with data for 4377 residents on 1470 features [41], and 3005 residents on 820 features [42] for health and wellbeing of aged Americans. Dataset D is a DNA methylation (*CpG loci focused on 5′ promoter regions*) dataset with data for 108 patients on 27,579 features [43–44].

For simplicity, we removed features with too many missing values, high correlatation with other features, low sample variation, categorical data, or contextual data from the datasets, and dropped samples with missing values. We use "percentage of unhealthy days" (Dataset A), "height" (Dataset B and Dataset C) and "age" (Dataset D) as the outcomes. We split the data into training data (n_{tr}) and test data (n_{te}) (Supplementary Table 3). Dataset A (n = 1156), Dataset B (n = 1292) and Data-

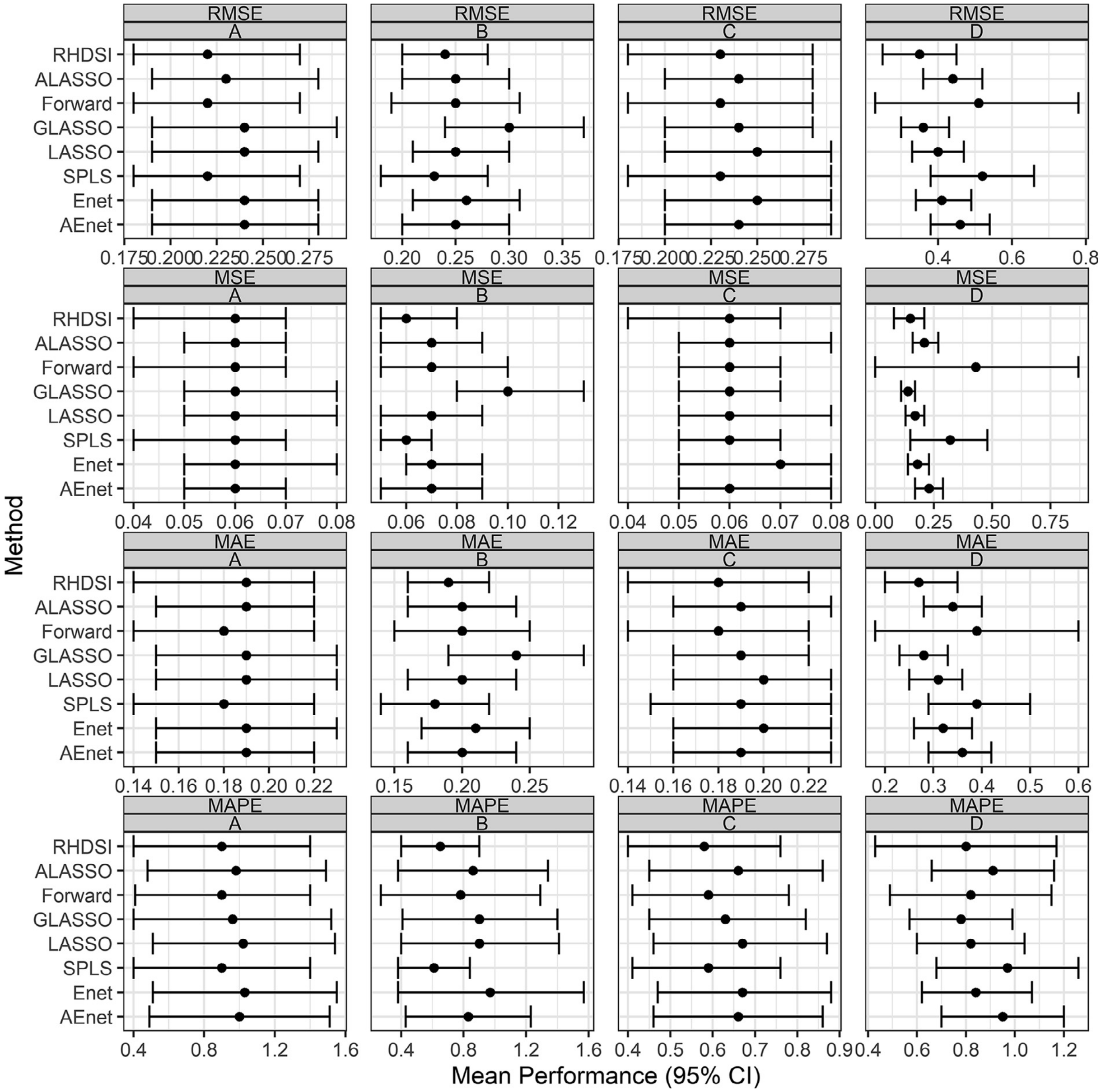

Fig. 3. Average performance (RMSE, MSE, MAE, MAPE) of different methods over ten trials on the simulated datasets for test data. A: Scenario I, B: Scenario II, C: Scenario III and D: Scenario IV.

set C (n = 1511) use 20% of the data for training while Dataset D (n = 108) uses 80% of data for training. In each dataset, all the possible two interaction terms are added in the feature selection model.

The hyperparameters were optimized as described above (Supplementary Table 4). The hyperparameters q (8–18), LF (4–9) and Af (3–8) for Datasets B, C and D are in a similar optimal range as identified in the simulated dataset, which supports the robustness of the hyperparameter search region. Dataset A has a different q region (9–31) than the q region from simulated studies(6–15), likely because Dataset A ($p = 55$) has a larger feature space than the simulated studies ($p = 25$). As predicted through simulations, CL is not robust across the four datasets. These results support the hyperparameter search region proposed in the simulated studies for up to 25 marginal features.

RHDSI was performed on all four test Datasets with optimized hyperparameters. The process is shown in Fig. 4 using Scenario D as an example. We compare the performance of methods using the mean predictive performance of the test data over ten trials. RHDSI estimates this parameter using regression without bootstrapping to reduce computation time, as the results with simulated data suggested similar RHDSI performance for regression with and without bootstrapping. The current study

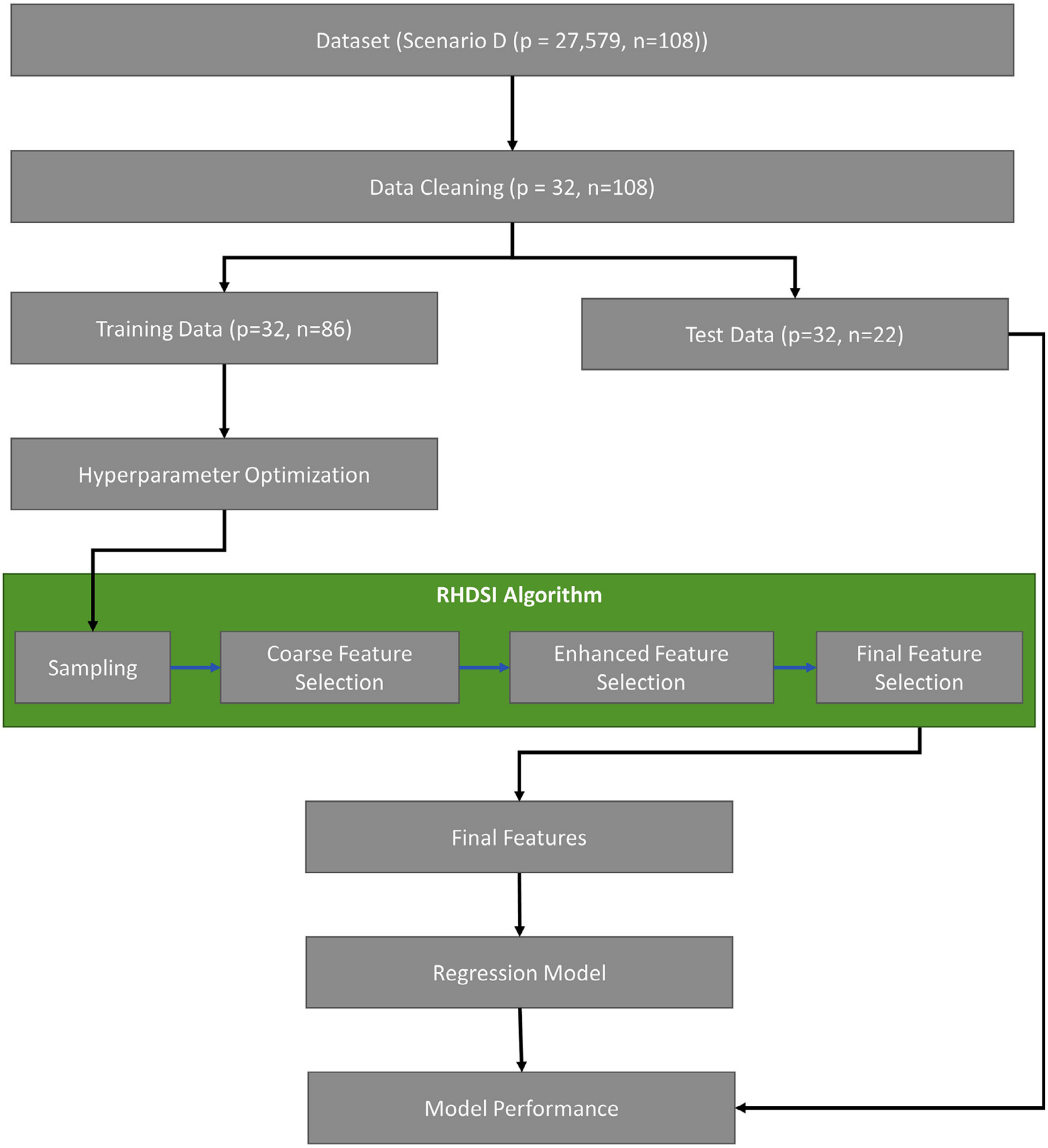
Dataset (Scenario D (p = 27,579, n=108))
Data Cleaning (p = 32, n=108)
Training Data (p=32, n=86)
Test Data (p=32, n=22)
Hyperparameter Optimization
RHDSI Algorithm
Sampling
Coarse Feature Selection
Enhanced Feature Selection
Final Feature Selection
Final Features
Regression Model
Model Performance

Fig. 4. Real data analysis process explained using Scenario D as an example.

uses adaptive ridge regression for model building as some scenarios do not have sufficient samples to prepare simple regression.

Table 3 summarizes the feature selection results. The average number of marginal and interaction features selected by the proposed method, RHDSI, is similar to or less than those selected by the other existing methods. Compared with the proposed method, the Forward method is more parsimonious; target features underestimation could cause parsimony since the Forward method showed poor target feature selection in simulation studies (Table 2). The selection of any interaction term in the final model suggests that the real dataset has feature/s which could influence the effect of other features on study outcome. The features selected in each scenario is provided in Supplementary file (Supplementary Tables 5 and 6). In Scenario D, interaction effect between methylation of Beta-galactoside alpha-2,6-sialyltransferase 1 (ST6GAL1) and Glutathione S-Transferase Mu 5 (GSTM5) genes (β = 9.26 (7.95–10.56)) is found in 70 percent of the trials for RHDSI algorithm. This interaction effect suggests that hypermethylation of ST6GAL1 with age is accelerated with hypermethylation of GSTM5. This

Table 3
Feature Selection Performance of different methods on the real datasets.

Scenario	Trials	Performance Parameter (Number of Features Selected)	Standard							RHDSI
			ALASSO	*Forward*	*GLASSO*	*LASSO*	*SPLS*	*Enet*	*AEnet*	RHDSI
			Mean (Range)							
A	10	Marginal (p = 55)	7 (4-9)	5 (4-6)	51 (46-53)	7 (4-12)	13 (4-30)	11 (6-18)	12 (5-28)	21 (9-32)
		Interaction (χ = 1485)	1 (0-4)	1 (0-2)	153 (144-164)	1 (0-3)	20 (0-72)	3 (1-9)	5 (0-27)	12 (3-20)
B	10	Marginal (p = 19)	6 (1-17)	4 (2-6)	19 (19-19)	5 (1-12)	8 (1-19)	6 (1-13)	7 (1-17)	8 (2-13)
		Interaction (χ = 171)	2 (0-11)	1 (1-3)	94 (84-106)	2 (0-5)	34 (0-156)	2 (0-6)	3 (0-11)	5 (1-8)
C	10	Marginal (p = 26)	8 (1-14)	5 (4-7)	26 (26-26)	6 (1-22)	3 (1-16)	7 (1-21)	10 (1-20)	13 (9-20)
		Interaction (χ = 325)	2 (0-5)	1 (0-3)	154 (136-165)	3 (0-20)	1 (0-10)	2 (0-14)	3 (0-10)	8 (3-18)
D	10	Marginal (p = 32)	20 (10-27)	8 (6-10)	30 (29-31)	21 (18-26)	32 (30-32)	28 (20-32)	26 (20-32)	18 (10-24)
		Interaction (χ = 496)	5 (0-14)	4 (2-6)	56 (54-59)	6 (2-17)	125 (92-170)	33 (1-63)	18 (3-45)	9 (3-16)

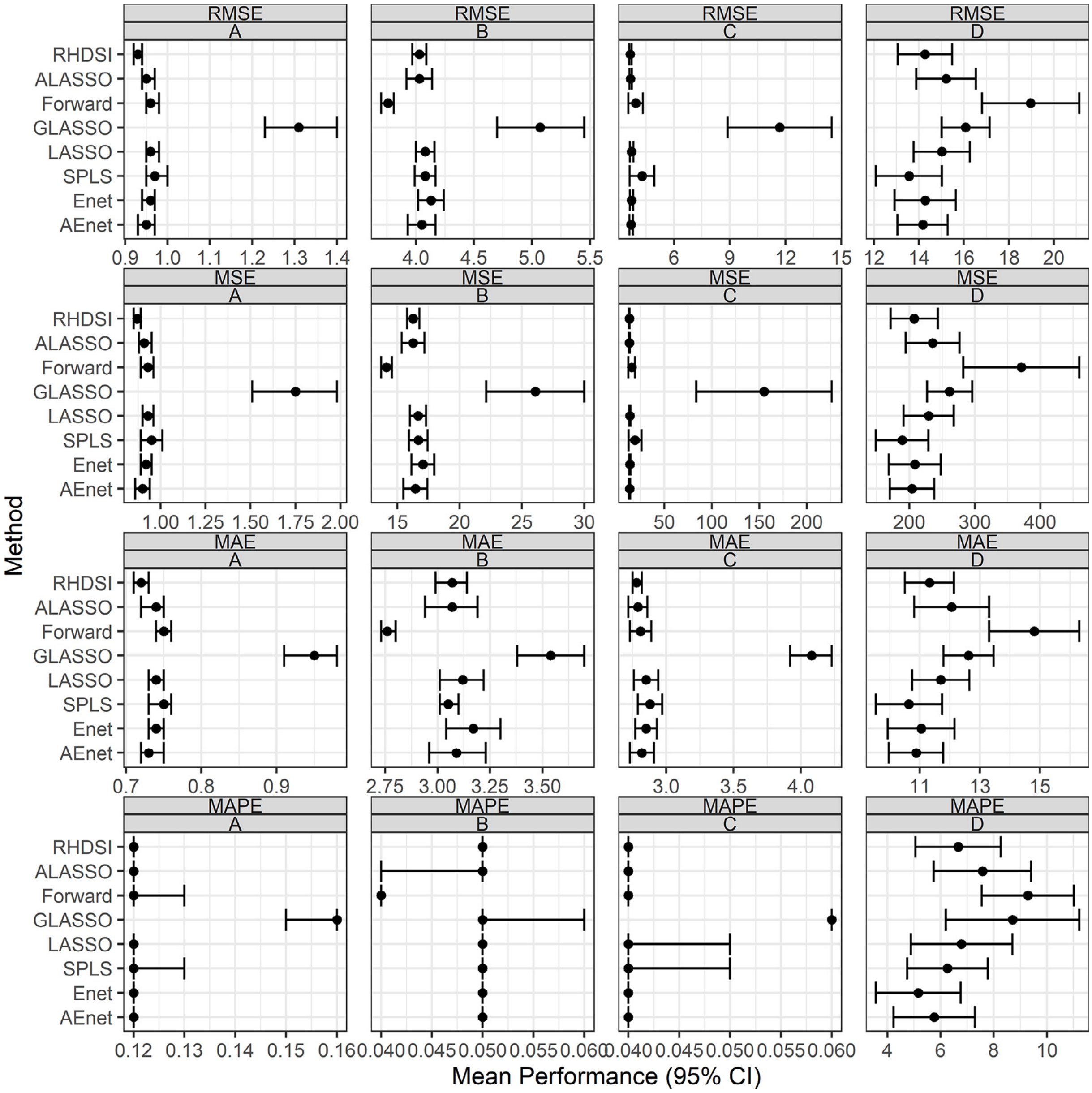

Fig. 5. Average performance (RMSE, MSE, MAE, MAPE) of different methods over ten trials on the real datasets for test data. A: Scenario A, B: Scenario B, C: Scenario C and D: Scenario D.

could be a relevant interaction as ST6GAL1 gene is involved in protein glycosylation and its hypomethylation may provide protection from gliomas [45]. Thus, RHDSI could help in identifying the interactions between features which could provide better models.

In terms of prediction performance (Fig. 5 and Supplementary Table 7), different performance metrics, i.e., RMSE, MSE, MAE and MAPE, showcase that the proposed method is at par or better than existing methods. The fact that RHDSI has similar prediction performance to existing methods suggests that the proposed method might perform better feature selection than the existing methods. Importantly, we found that the performance of both existing and proposed methods varied with datasets. The performance of Forward and GLASSO is more volatile as compared to other methods. The proposed method shows a more robust performance with different datasets as compared to existing methods.

5. Conclusion and discussion

This study proposes RHDSI, an innovative combinatorial statistical strategy to select features and interaction terms in high-dimensional datasets. This method is modular: it can incorportate many existing statistical modeling techniques at various stages, which provides the user freedom to adapt the method to their specific research requirements. The power of RHDSI is that it mitigates challenges associated with high-dimensional data, such as when the feature space is larger than the sample size, so that existing techniques can be employed in scenarios where they would otherwise fail.

Furthermore, RHDSI improves the selection efficiency of interaction terms and minimizes the impact of multicollinearity. RHDSI achieves these advantages by preparing and pooling multiple models for many smaller datasets generated from the original dataset. Our implementation of this strategy on simulated and real-world data shows that RHDSI could outperform existing techniques for feature selection and prediction. The successful performance of RHDSI with real data demonstrates its practical relevance.

This study opens various avenues to explore in future research. The current study focused on the ability of RHDSI to handle interaction terms, so we performed tests on limited data types. Future studies could focus on determining the robustness of RHDSI using different data types, such as temporal, categorical or time-to-event data. One limitation of this study is that it tests RHDSI applicability with techniques that estimate the outcome from a feature space using a linear combination function. Testing RHDSI with non-linear combination function techniques could be explored.

The coarse feature selection step in RHDSI depends on the regression coefficients of features, which prevents the use of statistical learning techniques that do not provide feature coefficients. Thus, future research could focus on exploring different pooling criteria to increase the number of techniques that could benefit RHDSI. Future work could test many other techniques like GLASSO, SPLS, principal component regression and elastic net in the RHDSI framework. Finally, the current study uses only a single parameter, i.e., CV, to perform unsupervised refinement by clustering. Future work could explore additional parameters like the coefficient of features and model performance of features for clustering.

Funding

This work was supported by Natural Sciences and Engineering Research Council of Canada (NSERC Grant RGPIN-2017-06672) and Prostate Cancer Canada (Translation Acceleration Grant 2018). The funding sources had no involvement in study design, in the collection, analysis and interpretation of data; in the writing of the report; and in the decision to submit the article for publication.

CRediT authorship contribution statement

Rahi Jain: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Writing - review & editing, Supervision, Project administration. **Wei Xu:** Conceptualization, Methodology, Validation, Writing - review & editing, Supervision, Project administration, Funding acquisition.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

Not applicable.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ins.2021.06.096.

References

[1] Fan J, Li R. Statistical challenges with high dimensionality : feature selection in knowledge discovery. Proceedings of the International Congress of Mathematicians Madrid, August 22–30, 2006. Madrid; 2007. pp. 595–622

[2] J.Y. Liu, W. Zhong, R.Z. Li, A selective overview of feature screening for ultrahigh-dimensional data, Sci. China Math. 58 (10) (2015) 1–22.

[3] P. Tavallali, M. Razavi, S. Brady, T.R. Singh, A non-linear data mining parameter selection algorithm for continuous variables, PLoS One 12 (11) (2017) e0187676, https://doi.org/10.1371/journal.pone.0187676.

[4] H. Motoda, H. Liu, Feature selection, extraction and construction, Commun IICM. 5 (2002) 67–72.

[5] S. Walter, H. Tiemeier, Variable selection: Current practice in epidemiological studies, Eur. J. Epidemiol. 24 (12) (2009) 733–736.

[6] G. Heinze, C. Wallisch, D. Dunkler, Variable selection – A review and recommendations for the practicing statistician, Biometrical J. 60 (3) (2018) 431–449.

[7] G. Heinze, D. Dunkler, Five myths about variable selection, Transpl. Int. 30 (1) (2017) 6–10.

[8] J.R. Donoghue, Univariate screening measures for cluster analysis, Multivariate Behav Res. 30 (3) (1995) 385–427.

[9] L.D.D. Desboulets, A review on variable selection in regression analysis, Econometrics 6 (2018) 1–23.

[10] O. Morozova, O. Levina, A. Uusküla, R. Heimer, Comparison of subset selection methods in linear regression in the context of health-related quality of life and substance abuse in Russia, BMC Med. Res. Methodol. 15 (2015) 1–17.

[11] F. Emmert-streib, M. Dehmer, High-dimensional LASSO-based computational regression models: Regularisation, shrinkage, and selection, Mach. Learn. Knowl Extr. 1 (2019) 359–383.

[12] T.J. Mitchell, J.J. Beauchamp, Bayesian variable selection in linear regression, J. Am. Stat. Assoc. 83 (404) (1988) 1023–1032.

[13] G. Zycinski, A. Barla, M. Squillario, T. Sanavia, B. Di Camillo, A. Verri, Knowledge Driven Variable Selection (KDVS) – A new approach to enrichment analysis of gene signatures obtained from high-throughput data, Source Code Biol. Med. 8 (2013) 1–14.

[14] A. Liaw, M. Wiener, Classification and Regression by randomForest, R News. 2 (2002) 18–22.

[15] L. Breiman, Bagging predictors, Mach Learn. 24 (2) (1996) 123–140.

[16] T.K. Ho, The random subspace method for constructing decision forests, IEEE Trans. Pattern Anal. Mach. Learn. 20 (1998) 832–844.

[17] L. Breiman, Random forests, Mach. Learn. 45 (2001) 5–32.

[18] S. Kotsiantis, D. Kanellopoulos, Combining bagging, boosting and random subspace ensembles for regression problems, Int. J. Innov. Comput. Inf. Control. 8 (2012) 3953–3961.

[19] Bach FR. Bolasso : Model Consistent Lasso Estimation through the Bootstrap. Proceedings of the 25th International Conference on Machine Learning. Helsinki; 2008. pp. 33–40.

[20] P. Teisseyre, R.A. Kłopotek, J. Mielniczuk, Random Subspace Method for high-dimensional regression with the R package regRSM, Comput. Stat. 31 (3) (2016) 943–972.

[21] B.S. Wang, B. Nan, S. Rosset, J. Zhu, Random lasso, Ann. Appl Stat. 5 (2011) 468–485.

[22] P.R. Shakya, Y.A. Melaku, A. Page, T.K. Gill, Association between dietary patterns and adult depression symptoms based on principal component analysis, reduced-rank regression and partial least-squares. 39 (9) (2020) 2811–2823.

[23] H. Artigue, G. Smith, Z. Lu, The principal problem with principal components regression, Cogent Math. Stat. 6 (1) (2019) 1622190, https://doi.org/10.1080/25742558.2019.1622190.

[24] T. Mehmood, S. Sæbø, K.H. Liland, Comparison of variable selection methods in partial least squares regression, J. Chemom. 34 (2020) 1–14.

[25] Sutawika LA, Wasito I. Restricted Boltzmann machines for unsupervised feature selection with partial least square feature extractor for microarray datasets. 2017 International Conference on Advanced Computer Science and Information Systems (ICACSIS). Bali; 2017. pp. 257–260.

[26] C. Broc, B. Calvo, B. Liquet, Penalized Partial Least Square applied to structured data, Arab. J. Math. 9 (2) (2020) 329–344.

[27] R. Lehr, Sixteen S-squared over D-squared: A relation for crude sample size estimates, Stat. Med. 11 (8) (1992) 1099–1102.

[28] J. Cohen, Statistical power analysis for the behavioral sciences, 2nd ed., Lawrence Earlbaum Associates, Hillsdale, New Jersey, 1988, pp. 24–27.

[29] V. Mehta, S. Bawa, J. Singh, Analytical review of clustering techniques and proximity measures, Artif Intell Rev. 53 (8) (2020) 5995–6023.

[30] D. Steinley, K-means clustering: A half-century synthesis, Br. J. Math. Stat. Psychol. 59 (2006) 1–34.

[31] P. Fränti, S. Sieranoja, K-means properties on six clustering benchmark datasets, Appl. Intell. 48 (12) (2018) 4743–4759.

[32] Z. Friggstad, K. Khodamoradi, M. Rezapour, M.R. Salavatipour, Approximation schemes for clustering with outliers, ACM Trans. Algorithms. 15 (2018) 398–414.

[33] M. Abdel-Basset, L. Abdel-Fatah, A.K. Sangaiah, Metaheuristic algorithms: A comprehensive review, in: Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications, Academic Press, New York, USA, 2018, pp. 185–231.

[34] J.H. Friedman, T. Hastie, R. Tibshirani, Regularization paths for generalized linear models via coordinate descent, J Stat Softw. 33 (2010) 1–22.

[35] H. Zou, The adaptive lasso and its oracle properties, J. Am. Stat. Assoc. 101 (476) (2006) 1418–1429.

[36] Lim M, Hastie T. glinternet: Learning Interactions via Hierarchical Group-Lasso Regularization. R Package version 109. 2019.

[37] W.N. Venables, B.D. Ripley, Modern Applied Statistics with S, Springer, Fourth. New York, 2002.

[38] Chung D, Chun H, Keleş S. Package' spls'. 2019 [cited 22 Sep 2020]. Available: https://cran.r-project.org/web/packages/spls/spls.pdf

[39] M. Lim, T.J. Hastie, Learning interactions through hierarchical group-lasso regularisation, J. Comput. Graph Stat. 24 (2015) 627–654.

[40] Centers for Disease Control and Prevention. Community Health Status Indicators (CHSI) to Combat Obesity, Heart Disease and Cancer. In: Healthdata.gov [Internet]. 2012 [cited 6 Aug 2020]. Available: https://healthdata.gov/dataset/community-health-status-indicators-chsi-combat-obesity-heart-disease-and-cancer

[41] Waite L, Cagney K, Dale W, Hawkley L, Huang E, Lauderdale D, et al. National Social Life, Health and Aging Project (NSHAP): Wave 3, [United States], 2015-2016 (ICPSR 36873). In: Inter-university Consortium for Political and Social Research [Internet]. 2019 [cited 22 Sep 2020]. Available: 10.3886/ICPSR36873.v4

[42] Waite LJ, Laumann EO, Levinson WS, Lindau ST, 'O'Muircheartaigh CA. National Social Life, Health, and Aging Project (NSHAP): Wave 1, [United States], 2005-2006 (ICPSR 20541). In: Inter-university Consortium for Political and Social Research [Internet]. 2019 [cited 22 Sep 2020]. Available: 10.3886/ICPSR20541.v9

[43] S. Numata, T. Ye, T. Hyde, X. Guitart-Navarro, R. Tao, M. Wininger, C. Colantuoni, D. Weinberger, J. Kleinman, B. Lipska, DNA methylation signatures in development and aging of the human prefrontal cortex, Am. J. Hum. Genet. 90 (2) (2012) 260–272.

[44] Akalin A. compGenomRData, In: Github [Internet]. 2019 [cited 22 February 2021]. Available: https://github.com/compgenomr/compGenomRData/blob/master/inst/extdata/CpGmeth2Age.rds.

[45] R.A. Kroes, J.R. Moskel, The role of DNA methylation in ST6Gal1 expression in gliomas, Glycobiology 26 (2016) 1271–1283.

RJ is a postdoctoral fellow at Princess Margaret Cancer Centre whose work on high dimensional data focuses on developing methodologies to perform feature selection and handle missing data. The interest is in designing methodologies that integrate statistical and machine learning principles. He is PhD from Indian Institute of Technology Bombay with many journals and conference publications in data science, public health and environment.

WX is a Clinician Scientist at Princess Margaret Cancer Centre, and an Associate Professor of Biostatistics at Dalla Lana School of Public Health, University of Toronto. His research interests focus on statistical and machine learning, bioinformatics, cancer big data, statistical genetics, clinical trial design and analysis, predictive model construction, and personalized medicine development. So far, he has published over 360 peer-reviewed papers in high impact journals on statistics, bioinformatics, medical science, and human genetics. His H-index is 61 with more than 19,500 citations.