Accepted Manuscript

Classification performance improvement using random subset feature selection algorithm for data mining

Lakshmi Padmaja Dhyaram, B. Vishnuvardhan

PII: S2214-5796(17)30179-X
DOI: https://doi.org/10.1016/j.bdr.2018.02.007
Reference: BDR 94

To appear in: *Big Data Research*

Received date: 24 July 2017
Revised date: 21 January 2018
Accepted date: 23 February 2018

Please cite this article in press as: L.P. Dhyaram, B. Vishnuvardhan, Classification performance improvement using random subset feature selection algorithm for data mining, *Big Data Res.* (2018), https://doi.org/10.1016/j.bdr.2018.02.007

Classification Performance Improvement Using Random Subset Feature Selection Algorithm for Data Mining

Lakshmi Padmaja Dhyaram,Department Of IT,Associate Professor,Anurag Group Of Institutions,Hyderabad :email *lakshmipadmajait@cvsr.ac.in*

Dr.B Vishnuvardhan,Professor,JNTUH,Hyderabad ,email MAILVISHNUVARDHAN@GMAIL.COM

Abstract

This study focuses on feature subset selection from high dimensionality databases and presents modification to the existing Random Subset Feature Selection(RSFS) algorithm for the random selection of feature subsets and for improving stability. A standard k-nearest-neighbor (kNN) classifier is used for classification. The RSFS algorithm is used for reducing the dimensionality of a data set by selecting useful novel features. It is based on the random forest algorithm. The current implementation suffers from poor dimensionality reduction and low stability when the database is very large. In this study, an attempt is made to improve the existing algorithm's performance for dimensionality reduction and increase its stability. The proposed algorithm was applied to scientific data to test its performance. With 10 fold cross-validation and modifying the algorithm classification accuracy is improved. The applications of the improved algorithm are presented and discussed in detail. From the results it is concluded that the improved algorithm is superior in reducing the dimensionality and improving the classification accuracy when used with a simple kNN classifier. The data sets are selected from public repository. The datasets are scientific in nature and mostly used in cancer detection. From the results it is concluded that the algorithm is highly recommended for dimensionality reduction while extracting relevant data from scientific datasets.

Keywords: Random Forest, Subset Feature Selection, Dimensionality Reduction, Scientific Data, Stability

1. Introduction

Data mining, the extraction of useful hidden features from large databases, is an effective new innovation with incredible potential to help organizations, focus on developing business strategies. The tools, developed for mining data, anticipate future patterns and practices, permitting organizations to make proactive, learning-driven choices. Many data mining tools can address business challenges more effectively than can traditional query or report-based tools. The performance of traditional tool's is very poor because of the large quantities of data involved. However, large quantities

of data might sometimes result in poor performance in data analytics applications as well.

Most data mining algorithms are implemented column-wise, which makes them become slower as the number of features increases. When the quantity of collected data is very large, mining for relevant data is a challenge. This is known as the "curse Of dimensionality"[1, 2, 3, 4]. Hence, there is a need for reducing the dimensionality of data without compromising the intrinsic geometric properties. Several methods have been developed, as shown in Figure (1), to address the challenge. Especially in the fields of bio-medical engineering, drug testing, cancer research, the data quantities involved are huge, and collecting them is very expensive. The data generated from experiments, in the above-mentioned fields are popularly known as scientific data. Such scientific data are tend to be noisy and sparse in nature [5, 6]. Because of this, standard data mining tools often do not perform efficiently when applied to scientific data. In this paper, an attempt is made to improve the existing random subset feature selection(RSFS) algorithm for better dimensionality reduction when applied on scientific data.

Scientific data sets result from extensive research in fields such as cancer research, bio-informatics, medical diagnosis, genetic engineering and weather studies. These data sets are sparse in nature. For example, cancer, also called malignancy, is an abnormal growth of cells. For cancer treatment chemotherapy, radiation, and/or surgery may be required according to the severity of the disease. In this study, we attempted to reduce the number of features to aid in the detection of cancer, leading to time savings and saved lives. In this paper, we propose a dimensionality reduction on features when applied to cancer data sets. We describe and evaluate our approach in 4 phases: (1) improvement of the random subset feature selection(RSFS) algorithm, (2)the two-sample t-test to ascertain whether the difference between the existing and proposed algorithms is significant, (3) a box plot comparing proposed algorithm's performance with that of the existing algorithm when datasets are from two classes are of a multi-class labeled type, and (4)stability enhancement for a stable feature subset. This paper is organized into 6 sections: 1. Introduction(this section), 2. Dimensionality Reduction Techniques Related Work, 3. About the existing RSFS, 4. Proposed RSFS Algorithm(present work), 5. Experiments and Results, and 6. Conclusion.

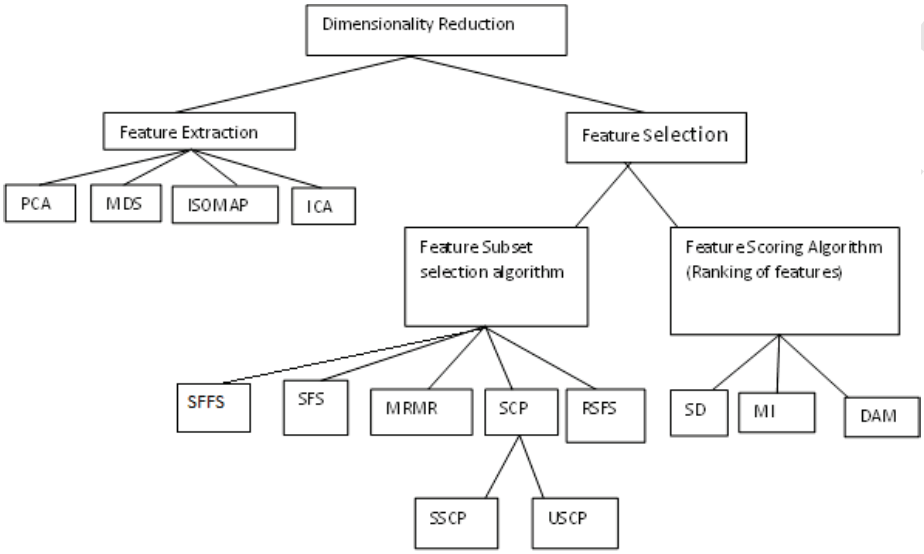

Figure 1: Taxonomy of techniques for dimensionality reduction

2. Dimensionality Reduction Techniques Related work

The dimensionality reduction techniques(shown in Figure (1)) are explained in the following sections.

2.1. Feature Extraction and Feature Selection Techniques

Feature extraction is a dimensionality reduction technique to build a new set of features from the original feature set.

The Principal component analysis(PCA)[7] is widely used in exploratory data analysis for building predictive models: it is a statistical procedure that converts a large data set with correlated features into a set of linearly uncorrelated *principal components*. The Multi-Dimensional Scaling(MDS)data visualization technique[8] is used to understand the level of similarity between individual cases of a data set. Independent component analysis(ICA)[9], which is widely used in signal analysis, is a computational method to separate, a multivariate signal into individual components.

Feature selection is a process for identifying and removing redundant and irrelevant features and increasing classification accuracy. The feature selection method[10, 11] selects a subset from the original feature set without impairing its knowledge. Feature subset selection methods are classified into four types:(1) embedded methods (2) wrapper methods (3) filter methods, and (4) hybrid methods. The feature selection methods are further divided into subset selection methods and feature scoring methods.

2.1.1. Feature Subset Selection Methods

Feature subset selection(FSS) is an important step in the data mining process[12, 13, 14] to select the relevant feature subset from a large data set before classification. This process is important in the scientific research areas such as genomics, bioinformatics, metabonomics, near-infrared and Raman spectroscopy [15, 16, 17], fields space in which the data sets are very large and sparse or noisy in nature. The purpose of feature subset selection is to improve performance and select effective predictors by addressing the dimensionality. Several methods, are employed to address this challenge in the field of data mining: Two popular approaches are Feature Selection(FS) and the other is Feature Extraction(FE)[18, 19, 20, 21]. It has been observed that for scientific data mining FS is superior to FE[15, 20].

There are many important feature subset selection methods, such as sequential forward selection(SFS)[22], sequential floating forward selection (SFFS)[23], minimum redundancy and maximum relevancy method(MRMR[24]) and the random subset feature selection (RSFS)method [25]. In SFS the features are selected by iteratively updating the feature importance, however this some times results in, sub-optimal output. To eliminate this problem, SFFS was developed, in which iterations are continued until they converge. The MRMR method selects the features based on their relevancy:features that are independent are selected, and irrelevant features are eliminated. The above methods perform with greater degrees of accuracy when the data set is small; however, when the data set is large, the results are unsatisfactory.

The random subset feature selection (RSFS)algorithm [26, 25, 27, 28, 29, 30] is used for selecting relevant features from large data sets. Each feature is evaluated and selected based on its usefulness. Because of the randomization technique, the selected subset of features is always optimal.

The set covering problem(SCP)[31] is a popular method for subset selection that is used in various applications; the supervised set covering problem(SSCP) or unsupervised set covering problem(USCP) are subsets of the SCP for finding the optimal set for classification.

2.1.2. Feature Scoring Methods

The goal of the statistical dependency(SD) method[32] is to ascertain whether a feature is statistically dependent on the label; if all the features are independent, then the SD value is minimized. Mutual information(MI) [18] is used for calculating the mutual dependence between two features. Both the SD and MI methods are used for scoring and ranking of the features. The distribution alignment and matching(DAM)method[33] is a fully unsupervised method used for selecting the features by comparing their respective distributions. This method is very effective if the data are already analyzed.

This paper presents an improvement to the existing RSFS algorithm and evaluates its performance with that of existing one. An attempt is made to select the most useful and relevant features from a large data set. Unlike other Greedy heuristic algorithms, this algorithm is not susceptible to local maxima[10]. However, it is susceptible to overfitting when proper precautions, such as cross-validation and data set splitting, are not followed in the training and testing data sets. The algorithm learns a good subset of features from the total data set for data mining classification tasks and demonstrates reasonable performance on independent test sets. This study conducted experiments using various scientific data sets, and the results are discussed in section 5.

2.2. Challenges in Feature Subset Selection on High Dimensional Scientific Data

Selection of the most important and relevant features from high dimensional scientific data[34, 35], for the classification task is a challenge currently faced by many data mining professionals. Ideally the best subset will contain those features providing complete information about the data and adding or subtracting information should not improve or degrade the performance [20, 36, 37]. Since the resulting data set is small than the original high dimensional dataset, it is easier to perform classification on that data for further analysis [38, 36].

Although it may sound simple, there are two fundamental challenges in finding an optimal feature subset from a huge scientific dataset:(1)exhaustive searching and (2)over fitting. For a dataset of n features there exists 2^n subsets. As the number of features increases, the number of subsets increase exponentially, and searching for a relevant subset is exhaustive and time consuming and sometimes computationally impossible. Similarly, when the dataset is huge and the total number of features is greater than the size of training sample, the model shows very good performance on the training set but not on the test or validation data[39, 24, 40, 41].

In order to overcome the exhaustive search issue, several feature subset selection methods[42] are used to select a good subset. Several heuristic methods are used, such as incrementally adding the next best feature to the current feature subset known as *forward selection*[22]; removing irrelevant features[43], known as *backward elimination*; or iteratively carrying out both to avoid feature nesting[23]. In some cases, a correlation measure of feature importance can be used to select the most important features based on the rank, which eliminates the need for iterative searching of the best subsets.

The overfitting problem can be addressed by splitting the data set into training and testing sets(two-thirds and one-third, respectively) and testing the quality of the selected subset using the independent test set[38]. Although this technique can help to estimate the amount of overfitting, it may not help in selecting the optimal feature subset. However, in reality, the overfitting can either by using the comprehensive

dataset or by using an algorithm that is not prone to "false" optima for a specific training dataset.

In the current work, an improved random subset feature selection method is presented that reduces the dimensionality of the dataset without compromising the classification accuracy. The proposed method also has improved performance compared with the existing algorithm.

2.3. Classifier

An instance based standard k-nearest-neighbor(kNN) classifier is used to determine the class membership. In all the experiments, the counts of the different class labels within of k samples were normalized to account for uneven distribution.

2.4. Data Sets

Shown in Figure(2), the datasets were collected from various repositories such as UCI-machine learning repository, www.featureselection.asu.edu, and www.broadinstitute.org [44]. The majority of the datasets are from cancer research studies.

Data set	Original features	Instances	Type of Dataset	Data set	Original features	Instances	Type of Dataset
GLI_85	22283	85	binary	PCMAC	3289	1943	binary
lungcancer_32_149	12533	149	multiclass	warpPIE10P	2420	210	binary
CLL_SUB_111	11340	111	multiclass	warpAR10P	2400	130	binary
orlraws10P	10304	100	binary	Colon	2000	62	binary
Arcene	10000	200	binary	Colon Cancer	2000	62	binary
pixraw10P	10000	100	binary	Data_nfold_random_opticalpen	1343	61	multiclass
nci9	9712	60	binary	ORL	1024	400	binary
Carcinom	9182	174	multiclass	Yale	1024	165	binary
ALLAML	7129	72	binary	brain_images	957	80	binary
Leukemia	7070	72	binary	Lung_discrete	325	73	multiclass
Prostate_GE	5966	102	binary	dermatology_data	279	452	multiclass
T0X_171	5749	171	multiclass	alcohol	52	65	multiclass
BASEHOCK	4862	1993	binary	dermatology_data	34	33	binary
GLIOMA	4434	50	multiclass	wine	13	178	multiclass
RELATHE	4322	1427	binary	Iris	4	150	multiclass
Lymphoma	4026	96	multiclass	Ovarian Cancer	4000	216	binary
lung	3312	203	multiclass				

Figure 2: Data sets

2.5. Stability

A feature selection algorithm is regarded as having high stability if it produces a consistent feature subset when new training samples are added or when some training samples are removed[45, 46, 47, 48]. If an algorithm produces a different result when there is any change in the training data, then that algorithm is unreliable for feature selection. Examples of instabilities are given in [48]. In Dunne et al. [46], wrapper techniques were used to study stability measures and introduced with possible solutions for addressing the problem. Various measures were established in[45, 49, 47]

Table 1: Important notations used in the paper

Important notations used in the paper	
Notation	Description
$\bar{x}_i$	Normalized value of the feature
μ	Mean value of the feature
σ	Standard deviation of feature
P	Full set of features(columns)
N	Samples(rows)
$\bar{Y}_i$	Subset (selected features or columns)
f_p	Feature
r_p	Relevancy of feature f_p
d_q	Dummy feature
r_q	Relevancy of dummy feature d_q
I	Number Of dummy features
S_i	Feature subset
c_i	Criterion function
E(c)	Expected criterion
r_{rand}	Relevancy Of dummy feature
θ	Threshold value
m	Subset selected for processing$\sqrt{P}$
g_q	Relevancy Of dummy feature d_q

for evaluating different subsets obtained using a certain number of iterations. Using these measures, a more robust subset can be found for different datasets. In Yang and Mao[48], a multi-criterion fusion algorithm was developed using a combining of multiple classifiers to improve the accuracy.

The feature selection algorithm can alone provide the relevancy[50] of each feature for classification, but it cannot provide a stable subset on each iteration as it is dependent on the classifier. In addition, a stable feature set cannot be provided on the basis of an appropriate feature selection algorithm. However, it can aid in selecting the relevant features when the latter is coupled with a classifier. In our approach, a post-processing technique based on distance measure is implemented after the classifier output is collected. This results in a stable feature subset. The procedure explained in detail in the section that follow.

3. About the Existing RSFS Algorithm

The RSFS algorithm has three tasks, namely (1)pre-processing (2) random subset feature selection, and (3)classification (4)challenges

3.1. Pre-processing

Data transformation is an important task for improving the performance of a data mining algorithm.There are several transformation techniques, such as min-max,Z score, and decimal scaling. Of these three methods Z score is the most powerful[51, 52]. Based on the work ofAl Shalabi et al. [51], the Z score transformation is a widely used pre-processing technique and it is applied here as well. Before initiating the actual processing, all features are normalized to have a zero mean and unit standard deviation:

$$\bar{x} = (x_j - \mu_j)/\sigma_j$$

where x_j is the original value of the feature; μ_j and σ_j are the mean and standard deviation, respectively, of the feature x_j; and $\bar{x}$ is the normalized value of the feature. The values measured across the respective data sets (training, development, and test sets) are normalized to improve the performance.

3.2. Existing Random Subset Feature Selection(RSFS) Algorithm

The aim of the random subset feature selection algorithm is to identify the best possible feature subset, from a large data set,in terms of its usefulness in a classification task; this process is shown in Figure (3). The feature selection is carried out iteratively. A simple kNN classifier is used to classify randomly selected subsets. In each iteration, the relevancy is fine tuned based on the feature membership. In the other feature subset selection algorithms, such as sequential forward selection [53], the *relevance* of a feature is computed by its presence or absence in the subset, whereas RSFS selects the features based on the average usefulness of the feature in the context of other features. Because of randomness, the RSFS is not prone to converging to local maxima[25, 50, 54].

In RSFS, as the number of iterations increases, more features are selected and classified using the kNN classifier[55, 56].

3.3. Classification

During the classification, kNN classifiers are used to classify the random subsets generated by the random forest algorithm[57, 58, 59, 56, 60, 61]. Since kNN is stable, re-sampling is not necessary for kNN. Each kNN classifier classifies a test point by the majority, or weighted majority class, of its k nearest neighbors. The final classification in each case is determined by a majority vote of random kNN classifications. This can be similar to voting by majority [62].

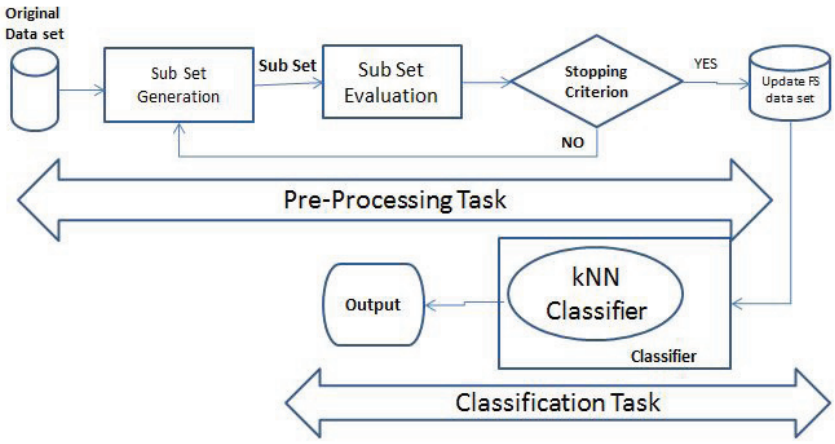

Figure 3: Feature subset selection and classification

3.4. Challenges

Several challenges are encountered with the existing methods of feature subset selection methods resulting a need for improved one. The challenges are mentioned below

The feature's utility,in filter based methods, is computed as the correlation between feature and class label. This is carried out by ranking the features in descending order and selection the top scores. This method cannot capture the interaction between the features. This results in suboptimal results and it is very difficult to decide the optimal subset.

Using Wrapper methods, the feature subset selection relatively more accurate when compared to the filter method but the feature subsets that are overly specific to the used classifier.[63].

Both Filter and Wrapper methods uses search strategy for identifying the subsets. Most widely used method by researchers are Genetic Algorithm(GA) and Support Vector Machine(SVM) for classification. But the GA is cannot assure the global maxima most likely to stuck at local maxima. The convergence tine is high some times leading to overfitting. This algorithm gives better results if the labels are binary. The SVM's are memory intensive and difficult to tune as the performance depends on picking the right kernel. The scalability is a challenge.Currently in industry Random Forest Classifiers are preferred over SVMs

The existing RSFS algorithm works well as long as the data set has a moderate

to high number of features. However, when the number of features in the data set exceeds 9000, the classification accuracy starts to decline, and reliability becomes very poor(see Figure (9))[64, 65]. As a result, there is a dire need of modification to the existing algorithm to enhance the performance and improve reliability for cases in which there are more than 10,000 features.

4. Proposed RSFS Algorithm

After carefully considering the challenges in the above mentioned section the proposed algorithm is mentioned here

In order to achieve improved accuracy and reliability for high dimensional scientific data sets, the modified RSFS algorithm is presented as follows.

Given a data set that contains N samples in rows and P features as columns, let Y_j be the columns selected in the previous iteration. Each *true* feature f_p from the full set of features P has a relevance value $r_p \in$ [-∞,∞] associated with it. A set of dummy features $d_q \in I$ with related relevancies $r_q$are also defined. During each iteration, the algorithm operates as described below(see Figure (4)).

4.1. Improvement to the RSFS Algorithm

The main advantages to observe in proposed(improved) RSFS algorithm are, which are not there in exiting algorithm

1. Separate training and testing datasets are given as input for realistic accuracy in improved RSFS algorithm.
2. Consistency of the final condensed subset is high.
3. Relatively lesser number of iterations will be taken for achieving the same result.
4. The benefits can be observed better for large data sets.
5. The number of features are reduced in proposed algorithm compared to existing algorithm.

We now discuss the proposed algorithm. The improved algorithm begins by normalizing the data of a given dataset of size D. The dataset is split into two parts two-thirds for the purpose of training and one-third for testing. From the training dataset, the algorithm randomly generates subsetsS_i of features($f_(p)$); each subsets size is equal to$\sqrt{(No.of features)}$ [62, 66]. The relevance of each generated subset is calculated as the difference between performance criteria $P(c)$ and expected criteria $E(c)$. The

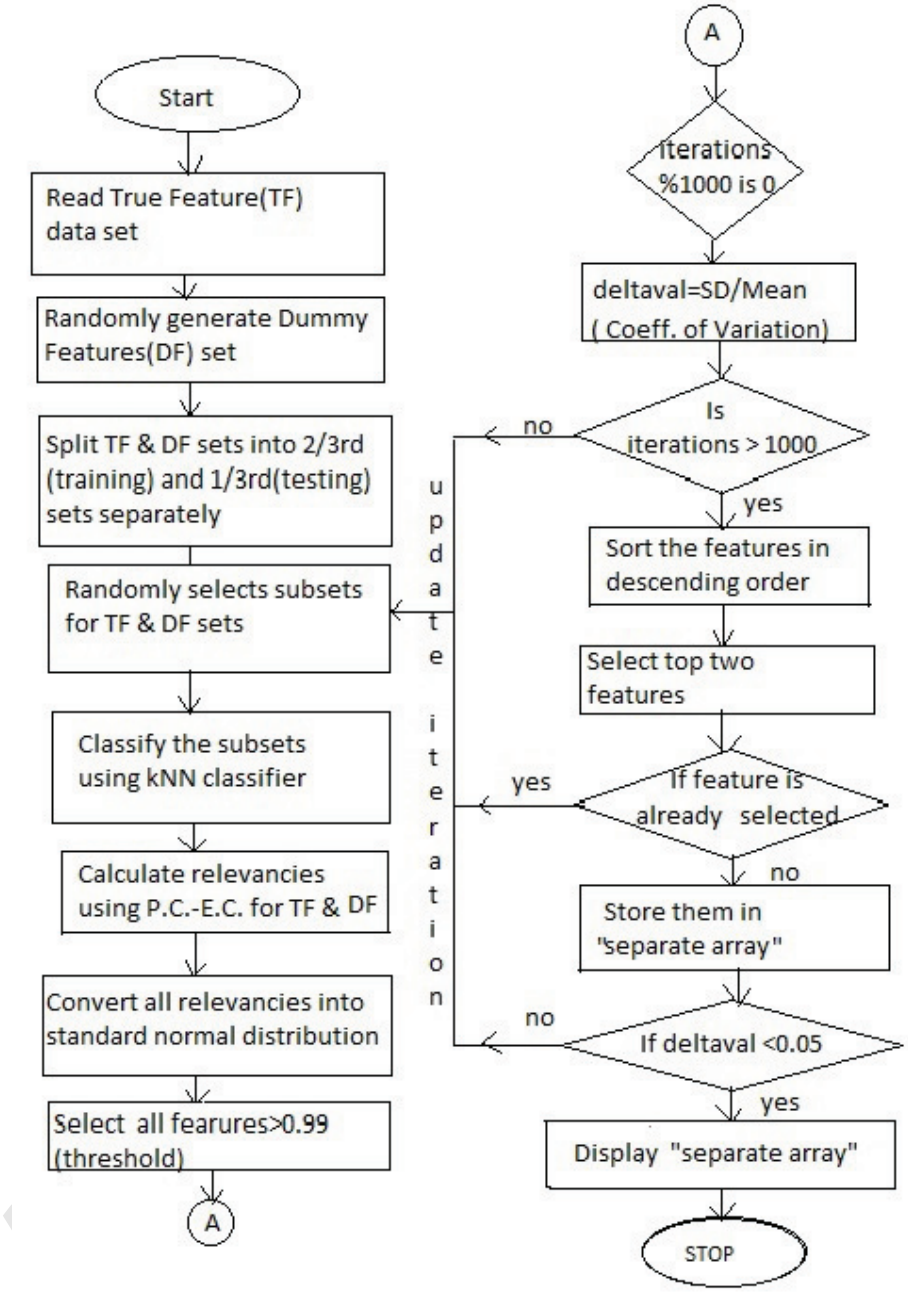

Figure 4: Flow of proposed algorithm

expected criterion is the average of all accuracies of all labels across all iterations, calculated as

$$E(c) = \sum \frac{(Correctly- \ Classified)}{(Correctly- \ Classified) + (Wrongly- \ Classified)} \times \frac{100}{n} \quad (1)$$

where n is the number of class labels in the dataset.

The performance criterion is the average of all accuracies of all labels for the current iteration. In the first iteration, the performance criterion and expected criterion are the same. In every iteration, the relevance of each true feature is updated. The relevancy column matrix is converted into a probability column matrix using the normal cumulative function(*normcdf()*) as a transfer function. This transformation is required because of the non availability of a global relevance value for each type of dataset. This transformation helps us to convert the relevancy values to probability values to establish a common understanding[67]. A set of dummy features d_q is generated, similar to S_i, to calculate the shape parameter and simulate the random walk process. As mentioned above, the relevancies of the dummy features are updated in each iteration. The dummy features‘ relevancies are useful as a baseline to set the threshold for the stopping criterion. The "best"subset is determined as the probability(relevancy of (feature $(r_p) >$ dummy feature $(r_q)) >$ (threshold value). The *threshold value* is user set threshold for probability.(in the current work the arithmetic mean of Expected criterion across all previous trials is used). The threshold value decides the time of convergence and shall be carefully selected. If the threshold value is very high then the no. of iterations will increase where as if the threshold value is less then the classification accuracy will decrease. Hence the value shall be selected based on nature of problem. The relevancy of dummy features(r_q) are modeled as normal distribution.

In our experiment, we selected the threshold value as 0.9(user defined threshold) for Forest data set explanation due to space constraint in the paper, for the remaining data sets 0.99 is threshold value and datasets were used as listed in Figure 2. Most of the datasets were collected through cancer research, and the cost of collection is very high. Dimensionality reduction on these datasets considerably helps the community for pre-screening the cancer condition in the early stages, as number of features(confirming tests)is reduced. The sampling process was repeated 20,000 times before the final reduced feature set was selected. The kNN classifier was selected as the criterion function, and the number of selected neighbors k was 2 (see [62]).

A prerequisite required for an improved RSFS algorithm is splitting the total feature set into separate training and testing datasets respectively to eliminate the bias.

4.2. Pseudo-code for the Improved RSFS Algorithm

Step 1: Normalize all the features to have zero mean and unit standard deviation for better performance.

Step 2: Perform the following operations on the training set, until the threshold value is reached.

Step 3: Randomly pick a subset S_j of m features, where $f_p(\ |S_j|{=}m_i\ p \in (1,|P|))$, from the full feature set P by sampling.

Step 4: Perform kNN classification on the reduced data set using S_j, and measure the value of a desired criterion function c_j.

Step 5: Update the relevancies r_p of all selected features f_p as

$$r'_p \leftarrow r_p + c_j - E(c) \tag{2}$$

where r'_p is the updated current relevance values of the feature vector, c_j is the value of the performance criterion function for the jth iteration, and $E(c)$ is the expected criterion function. In the current work, this corresponds to the average of the c_j values across all previous iterations.

Step 6: The relevancies of all the features are converted into cumulative normal distribution(to know the list of features, which satisfy the threshold criteria).

Step 7: Select all features which are $\geq$0.99 probability.

Step 8: The relevancies of such selected features are termed as weights.

Step 9: All selected features are sorted, in descending order based on their weights.

Step 10: After sorting, select the top two features, and store them in a separate array.

Step 11: If the feature is already in the top two features stored, in separate array, it will not be selected in the current iteration to avoid the redundancy.

Step 12: Repeat the process from step 3 until the threshold value is reached.

Step 13: All such top two features, which are stored in a separate array is the final output of the improved RSFS algorithm.

In parallel, a similar process is carried out on dummy features by always selecting a random subset of m dummy features and then updating the relevance values r_{rand} according to equation (2). The dummy features are never used in the classification task, but their relevancies are considered across all the trials. In the same way, the relevance g_q of any dummy feature d_q represent a random walk process that has no impact on the classification[68].

The best subset, as the final goal, is selected by satisfying the following equation

$$p(r_k > r_{rand}) \geq \theta \tag{3}$$

for all $f_k \in B$, P and $B \subset P$, where r_{rand} is the relevance of the dummy feature and θ is a user defined threshold for computing the probability. The random baseline level r_{rand} is modeled as

$$p(r_k > r_{rand}) = \frac{1}{\sigma_g \sqrt{2\pi}} \int_{-\infty}^{r_k} \frac{\exp(-(x-\mu_g)^2)}{2\sigma_g^2} dx \tag{4}$$

where μ_g and $\sigma_g$are the mean and standard deviation respectively, of the dummy feature relevancies r_g.

In the current study unweighted average recall(UAR) was used as the criterion function $E(c)$ in equation (2), and the probability was set to θ =0.99. The features selected in each iteration were set to m=round($\sqrt{|P|}$)[62]. Similarly,$\sqrt{I}$ random features were selected in each iteration, and the relevancies were updated in each iteration. This process was repeated according to the condition given in equation (3) before the final subset was selected. The kNN classifier was used for the classification task [62].

5. Experiments and Results

To demonstrate our claim, we illustrate with an example, of how the improved RSFS algorithm operates. A simple data set "Forest" is used for purposes of explanation. The data set has 27 features and 523 instances with a 4-way class label.

5.1. Example

Using this example, we explain the pre-processing, processing and results.

5.1.1. Pre-processing

1. Initially, the dataset is randomly divided into two parts in a ratio of one-third to two-thirds. These corresponds to the test and training matrices respectively.
2. Both the training and test matrices are normalized column-wise for the entire dataset.
3. The size of the subset is selected as 5 since the number of features is 27. The subset size is round($\sqrt{(No.of features)}$).

5.1.2. Processing

First Iteration

1. A subset of five features, feature numbers{ 20,20,20,10,11}, is randomly selected by algorithm.
2. The features are classified using the kNN classifier.

3. After first iteration, the classifier output(correctly classified and wrongly classified labels) is as shown below.

Label	Correct	Wrong
1	33	21
2	39	9
3	23	14
4	47	12

4. The performance criterion(P.C.) and expected criterion(E.C.) are equal at the first iteration.

5. The performance criterion is the average of all accuracies of all labels for the current iteration, = 71.0461%

6. The expected criterion is the average of all accuracies of all labels across all iteration upto the current iteration. = 71.0461% ((33 / (33+21)) + (39 / (39 + 9)) + (23/ (23 + 14)) + (47/ (47 + 12)))*100/4 = 71.0461%.

7. Similarly, the value of the expected criterion is calculated.

8. The relevance of the randomly selected features is equal to performance criterion – expected criterion = 0.

Second Iteration

1. The randomly selected feature numbers in the subset are { 2,23,27,14,15}.

Label	Correct	Wrong	Total Correct	Total Wrong
1	35	19	68	40
2	33	15	72	24
3	28	9	51	23
4	30	29	77	41

2. Performance criterion = 2.599 /4 * 100 = 64.975%.

3. Expected criterion = ((68/(68 + 40)) + (72/(72+ 24)) + (51/(51+23)) + (77/(77+41)))* 100 /4 = 67.987%.

4. Relevance of the features (randomly) selected (P.C. – E.C.) = 64.975 – 67.987 = (previous relevance)+ - 3.0120.

Third Iteration

1. The randomly selected feature numbers in the subset are { 12,19,26,7,18}.

Label	Correct	Wrong	Total Correct	Total Wrong
1	42	12	110	52
2	24	24	96	48
3	31	6	82	29
4	45	14	122	55

2. Performance criterion = 71.958%.

3. Expected criterion = 69.3421%.

4. Relevance of the features (randomly) selected (P.C. – E.C.) = +2.6161.

5. Relevance of the features (randomly selected) = (previous iteration relevance) + 2.6161.

6. The relevancy matrix of all the features of the dataset at the end of third iteration is shown in Table 2.

Table 2: Relevancy matrix of features after third iteration

2(a)

Feature No.	Relevancy
1	0
2	-3.0210
3	0
4	0
5	0
6	0
7	2.6161
8	0
9	0
10	0
11	0
12	2.6161
13	-3.0210

2(b) (continued)

Feature No.	Relevancy
14	-3.0210
15	-3.0210
16	0
17	0
18	2.6161
19	2.6161
20	0
21	0
22	0
23	-3.0210
24	0
25	0
26	2.6161
27	-3.0210

Similarly, the dummy features are processed and their relevancies are calculated & updated in each iteration. The dummy features are not used in classification, but their relevancies are accumulated across trials. However, the relevance of a dummy feature is useful as a threshold value for selection of the relevant features.

In every iteration, the mean and standard deviation of the dummy relevance are taken as the shaping parameters from the normal distribution fit of the calculated relevancies. The feature relevance is compared with the dummy feature relevance, and the feature having a relevance value greater than the dummy feature relevance value is selected.

Relevancy values are converted to probability values(Table 3) using *normcdf()* as the transfer function. This will eliminate the need for global relevance values for different data types.

Table 3: Probability matrix of features after third iteration

3(a)

Feature No.	Probability
1	0.5171
2	0.0462
3	0.5171
4	0.5171
5	0.5171
6	0.5171
7	**0.9379**
8	0.5171
9	0.5171
10	0.5171
11	0.5171
12	**0.9379**
13	0.5171

3(b) (continued)

Feature No.	Probability
14	0.0462
15	0.0462
16	0.5171
17	0.5171
18	**0.9379**
19	**0.9379**
20	0.5171
21	0.5171
22	0.5171
23	0.0462
24	0.5171
25	0.5171
26	**0.9379**
27	0.0462

Post processing Since the relevancy values are not constant globally, they are converted into probability values using the *normcdf* function. Features having a probability of greater than 0.9 are selected, in that iteration. These are stored in a separate array.

After every 1000 iterations, the coefficient of variation (deltaval) of the final set is used as the termination condition for the algorithm. This value detects the addition of features to the final feature set. If deltaval is less than 0.05, which means that there are no new features added to the final set, the execution loop is terminated; otherwise,it continues for another 1000 iterations.

The selected features are sorted in descending order based on their weight, and the top two features are removed and stored in a separate array(final features). The execution loop continues to select the next best features. After completion of the

iterations, the best subset of features is available in the final feature array.

Final Iteration

1. The randomly selected feature numbers in the subset are {25,6,15,12,13}.

Label	Correct	Wrong	Total Correct	Total Wrong
1	73	32	116341	93554
2	34	4	56357	19605
3	36	10	67451	24503
4	85	51	150424	121440

2. Performance criterion = 74.9396%.
3. Expected criterion = ((68/(68 + 40)) + (72/(72+ 24)) + (51/(51+23)) + (77/(77+41)))* 100 /4 = 64.5757%.
4. Relevance of the features (randomly) selected (P.C. – E.C.) = previous relevance + 10.3639

Table 4: Relevancy matrix of features after Final iteration

4(a)

Feature No.	Relevancy
1	776.9105
2	733.3913
3	740.1225
4	138.5532
5	647.2156
6	353.0844
7	-388.1886
8	-386.6553
9	-0.2404
10	727.5735
11	738.1689
12	309.5596
13	278.8731

4(b) (continued)

Feature No.	Relevancy
14	590.62493
15	519.0916
16	-531.4572
17	-438.8865
18	106.5135
19	-2035.7649
20	-2144.5181
21	-1909.2496
22	-1917.7052
23	-2039.0581
24	-1793.7886
25	-1520.1799
26	-2166.6891
27	-2528.5155

Table 5: Probability matrix of features after final iteration

5(a)

Feature No.	Probability
1	**1**
2	**1**
3	**1**
4	0.99
5	**1**
6	0.99
7	0.00
8	0.00
9	0.94
10	**1**
11	**1**
12	0.99
13	0.99

5(b) (continued)

Feature No.	Probability
14	**1**
15	**1**
16	0.14
17	0.20
18	0.99
19	0.93
20	0.51
21	0.51
22	0.51
23	0.04
24	0.51
25	0.51
26	0.93
27	0.04

From the example it can be seen that the original 27 features have been reduced to eight features(1,2,3,5,10,11,14,15) whose probability values are greater than 0.99. The dimensionality reduction has been carried out successfully.

Experiments were similarly conducted on all 31 datasets listed in Figure (2). The results are tabulated in Figure (5)and Figure (6).

5.2. kNN Classification

The k-nearest neighbor classification rule was used in the current study. The algorithm was tested using samples in the training set as a reference. This is conceptually simple and easy to implement. When the classifier is trained with sufficient data, it can easily classify complex patterns in datasets. However, the classier suffers from the curse of dimensionality [69]. The optimum value of k was calculated based on 10 fold cross validation. Owing to the splitting of the data into training and testing sets and the performing of cross validation, the overfitting problem is minimized[70, 47, 71].

5.3. Feature Subset Selection Results

Using the above mentioned methods, the experiments were conducted using various data sets with the kNN classifier. The results are tabulated in Figures(5) and Figure (6). The comparison of the results, shown in Figure (7), indicates that the data reduction performance of the modified algorithm is superior to that of the RSFS algorithm, while similar accuracy levels are maintained.

Data set	Original features	Instances	Type of Dataset	Original Accuracy	RSFS		MRSFS	
					Feautres selected	Accuracy	Features Selected	Acccuracy
GLI_85	22283	85	binary	68.97	1797	75.86	79	79.31
lungcancer_3	12533	149	multiclass	96.88	50	81.25	94	84.38
CLL_SUB_1	11340	111	multiclass	63.16	58	65.79	107	86.84
orlraws10P	10304	100	binary	91.18	166	97.06	80	85.29
Arcene	10000	200	binary	83.82	562	88.24	91	89.71
pixraw10P	10000	100	binary	94.12	100	91.18	80	88.24
nci9	9712	60	binary	33.33	202	66.67	98	61.9
Carcinom	9182	174	multiclass	91.53	270	98.31	57	91.53
Prostate_GE	5966	102	binary	74.29	1435	88.57	90	94.29
TOX_171	5749	171	multiclass	56.9	304	70.69	97	72.41
BASEHOCK	4862	1993	binary	81.35	1007	86.77	46	87.97
GLIOMA	4434	50	multiclass	83.33	95	88.89	67	88.89
RELATHE	4322	1427	binary	77.36	163	76.1	39	77.57
Lymphoma	4026	96	multiclass	81.82	272	90.91	88	78.79
Ovarian Canc	4000	216	binary	87.84	913	93.24	82	94.59
lung	3312	203	multiclass	85.51	476	92.75	83	88.41

Figure 5: Results

Data set	Original features	Instances	Type of Dataset	Original Accuracy	RSFS		MRSFS	
					Features selected	Accuracy	Features Selected	Acccuracy
PCMAC	3289	1943	binary	71.96	398	74.42	33	74.73
warpPIE10I	2420	210	binary	83.1	367	95.77	69	94.37
ColonCanc	2000	62	binary	50	17	59.09	8	59.09
Data_nfold	1343	61	multiclass	99.34	23	96.7	6	79.52
ORL	1024	400	binary	84.33	123	88.81	37	77.61
Yale	1024	165	binary	57.14	128	66.07	40	53.57
brain_imag	957	80	binary	64.29	83	71.43	35	75
Isolet	617	7797	binary	86.76	91	87.91	24	59.88
madelon	500	4400	binary	72	21	87.33	10	87.21
Lung_discr	325	73	multiclass	88	47	92	21	92
dermatolog	279	452	multiclass	5.92	54	37.5	13	33.55
alcohol	52	65	multiclass	78.26	16	86.96	5	91.3
dermatolog	34	33	binary	66.67	8	91.67	4	58.33
wine	13	178	multiclass	61	5	63.33	2	83.33
Iris	4	150	multiclass	96.08	2	98.04	2	98.04

Figure 6: Results

The performance was measured using the average recall both weighted and un-weighted. The un-weighted is the average classification accuracy of the classes, and

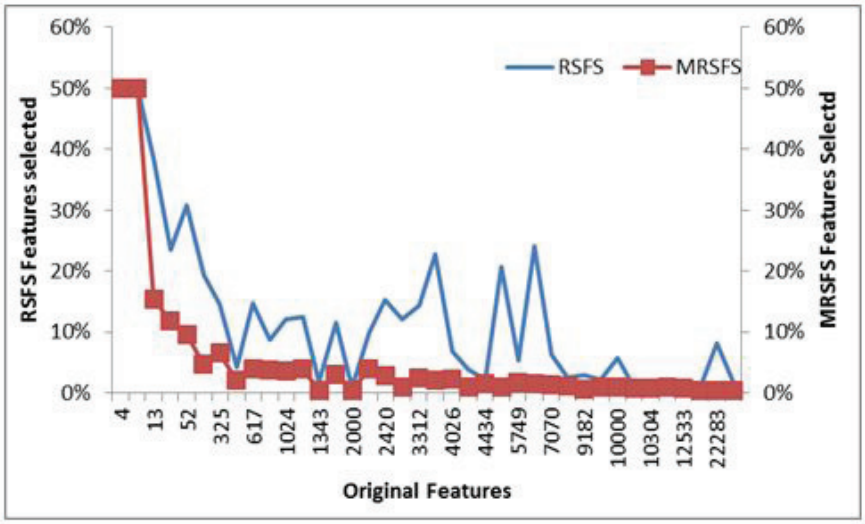

Figure 7: Comparison of results of existing algorithm(RSFS) and improved algorithm(MRSFS)

the weighted is the ratio of correct to incorrect classification of samples. The results are tabulated in Figures (5) and (6). From the results, as shown in Figure (7), it can be seen that the modified algorithm is more efficient in addressing the curse of dimensionality. This is also evident from Figures (5) and (6).

5.4. Stability Enhancement

In our method, we use the Hamming distance method to detect the change in the final subset. The stability is plotted as, the final subset size versus the number of iterations. The change in the final subset size is calculated using the following equation:

$$d_{st} = (No.\ of\ (x_{sj} \neq y_{tj})/n) \tag{5}$$

where an m by n data matrix X, is the current iteration output, which is m_x (1-by-n) row vectors x_1, x_2, ..., x_{mx}, and an my-by-n data matrix Y is the, previous iteration output, which is m_y (1-by-n) row vectors y_1, y_2, ...,y_{my}. The distances between the vectors x_s and y_t are calculated as shown in Yun et al.[34].

From the result in Figure(8), it can be seen that the proposed algorithm produces a subset that is more stable than that produced by the existing one.

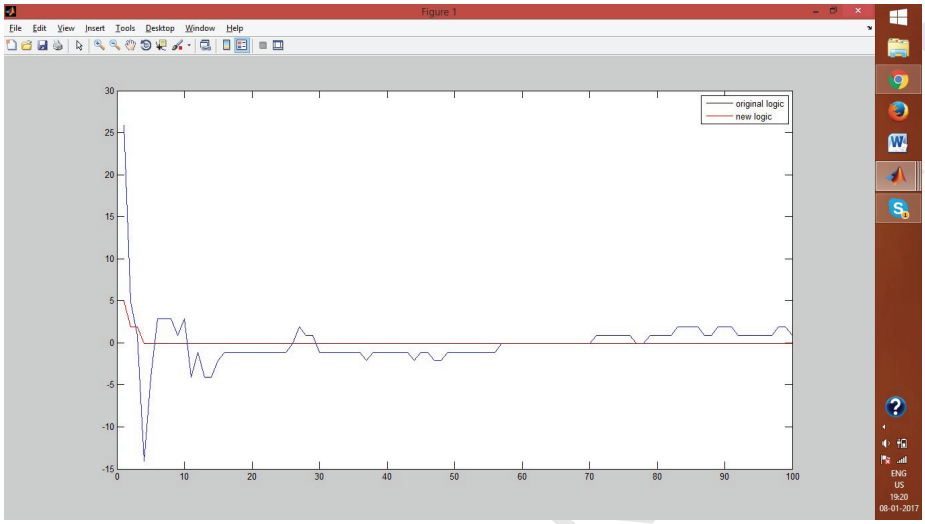

Figure 8: Stability of the algorithms

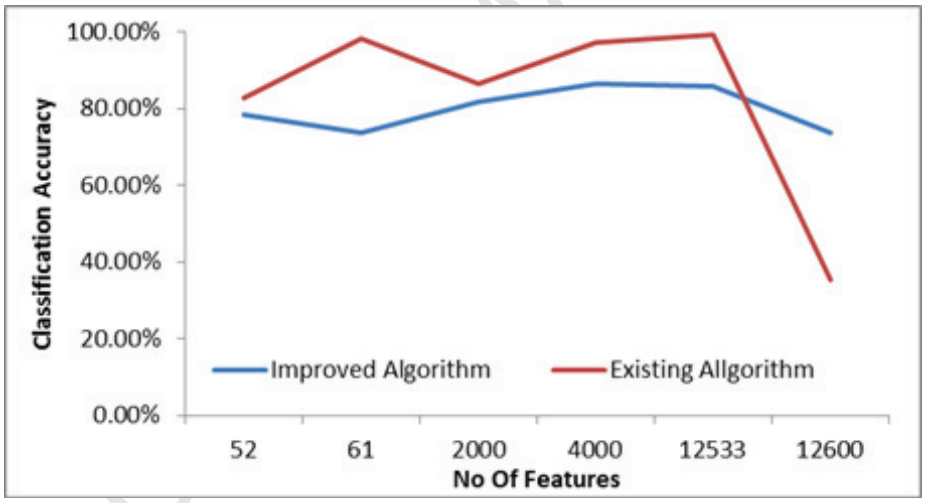

Figure 9: Comparison of classification accuracy of existing(RSFS) and improved algorithm(MRSFS)

5.5. Two Sample t Test

The two-sample t-test is one of the most commonly used hypothesis tests. It is applied to ascertain whether the average difference between the output of two algorithms is significant or if it is due to random chance. It helps in answering questions such as whether the dimensionality reduction is improved after a new algorithm has been implemented.

From the experimental results, it is evident that the hypothesis is proved as the two algorithms‘ performance results are different. To validate this, a statistical test was applied on the number of features selected by each algorithm and tested with the two sample t-test.

The two sample t-Test was conducted on the datasets generated by the RSFS algorithm(existing algorithm) and the improved RSFS algorithm. The results, shown in Figure(10), indicate that the modified algorithm selects fewer features than the existing algorithm, and the result is statistically significant[57]. This means that the improved algorithm is more efficient in terms of dimensionality reduction [72, 73, 74].

Two-Sample T-Test and CI: RSFS Compression, Mod RSFS Compression

```
Two-sample T for RSFS Compression vs Mod RSFS Compression

                       N    Mean   StDev  SE Mean
RSFS Compression      57  0.0867  0.0812    0.011
Mod RSFS Compression  57  0.0269  0.0381   0.0050


Difference = mu (RSFS Compression) - mu (Mod RSFS Compression)
Estimate for difference:  0.0598
95% lower bound for difference:  0.0400
T-Test of difference = 0 (vs >): T-Value = 5.03  P-Value = 0.000  DF = 79
```

Figure 10: Two sample t-Test results for existing RSFS and improved(MRSFS)algorithm

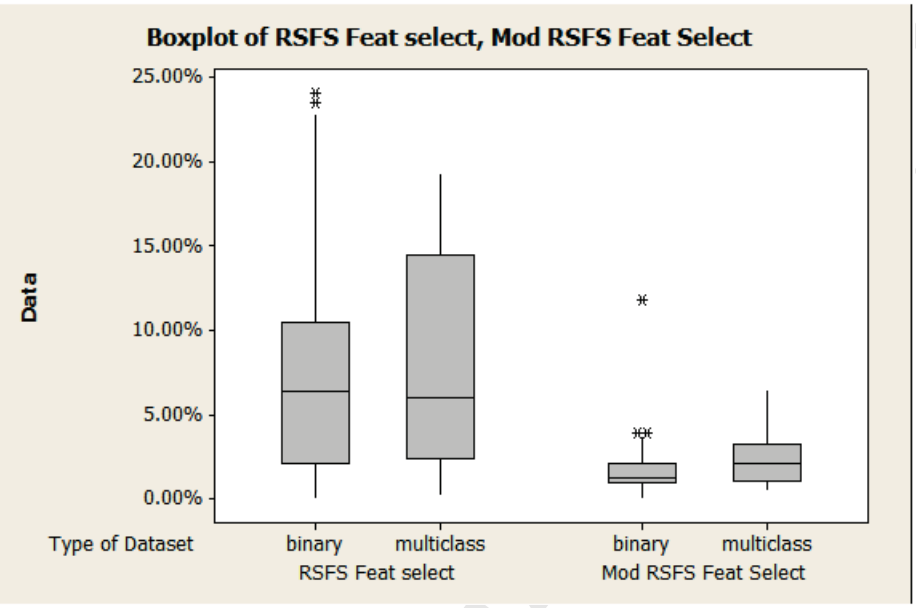

Figure 11: Box plot for performance of the two algorithms(existing RSFS and improved (MRSFS)) for dimensionality reduction

To understand the influence of two-label and multi-label class datasets on the feature subset selection performance, the experimental results were plotted on a box plot as shown in Figure (11). From the box plot it is evident that the improved algorithm has better performance than the existing algorithm.

6. Conclusion

Based on the above results, it is observed that the improved version of the RSFS algorithm is more effective in reducing the dimensionality of the scientific datasets than the existing algorithm and does not compromise the accuracy. The two sample t-Test shows that data compression is improved in the modified algorithm, and this is also evident from the box plot shown in Figure (11).

In the current study, the improved algorithm was iteratively applied on the training data set to enhance the classification accuracy. In the final result, it is evident that the informative features are better chosen by the classifier even for the high dimensional datasets. In every iteration the selected features are compared with final subset array to discard the duplicate features. Features are not present in the final array are selected and updated. In every iteration, the training data set is modified by omitting the strong two features which are selected in the previous iteration. This enables to converge the solution faster and more relevant features are selected[58, 75, 76].

Although the modification was successfully implemented on high dimensional scientific data[77, 78], to fully address the curse of dimensionality, more study is required to understand the behavior of the solution when applied to sparse datasets. It is also required to undertake future study, on multi-class data with a combination of classifiers.

In future a detailed comparative study is required using deferent types of classifiers such as Support Vector machine (SVM),Artificial Neural Network (ANN) etc to understand the performance of the algorithm.

7. Conflict of Interest Statement

The authors declare that there is no conflict of interest regarding the publication of this paper.

8. References

[1] R. Bellman, The theory of dynamic programming, Technical Report, RAND CORP SANTA MONICA CA, 1954.

[2] L. Chen, Curse of Dimensionality, Springer US, Boston, MA, pp. 545–546.

[3] G. V. Trunk, A problem of dimensionality: A simple example, IEEE Transactions on pattern analysis and machine intelligence (1979) 306–307.

[4] M. Houle, H.-P. Kriegel, P. Kröger, E. Schubert, A. Zimek, Can shared-neighbor distances defeat the curse of dimensionality?, in: Scientific and Statistical Database Management, Springer 2010, pp. 482–500.

[5] A. Sisto, C. Kamath, Ensemble Feature Selection in Scientific Data Analysis, Technical Report, Lawrence Livermore National Laboratory (LLNL), Livermore, CA, 2013.

[6] N. M. Adams, Scientific data mining: a practical perspective chandrika kamath, siam, 2009, Statistics in Medicine 30 (2011) 799–799.

[7] H. Abdi, L. J. Williams, Principal component analysis, Wiley Interdisciplinary Reviews: Computational Statistics 2 (2010) 433–459.

[8] J. B. Kruskal, Multidimensional scaling by optimizing goodness of fit to a nonmetric hypothesis, Psychometrika 29 (1964) 1–27.

[9] J. V. Stone, Independent component analysis, Wiley Online Library, 2004.

[10] M. Dash, H. Liu, Feature selection for classification, Intelligent data analysis 1 (1997) 131–156.

[11] Y. Saeys, T. Abeel, Y. Van de Peer, Robust feature selection using ensemble feature selection techniques, in: Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Springer 2008, pp. 313–325.

[12] B. J. Read, Data mining and science? Knowledge discovery in science as opposed to business, Citeseer, 1999.

[13] R. L. Grossman, C. Kamath, P. Kegelmeyer, V. Kumar, R. Namburu, Data mining for scientific and engineering applications, volume 2, Springer Science & Business Media, 2013.

[14] D. L. Padmaja, B. Vishnuvardhan, Comparative study of feature subset selection methods for dimensionality reduction on scientific data, in: Advanced Computing (IACC), 2016 IEEE 6th International Conference on, IEEE, pp. 31–34.

[15] Chandrika Kamath, Scientific Data Mining A Practical Perspective, SIAM, 1ed edition, 2009.

[16] M. M. Gaber, Scientific data mining and knowledge discovery, Springer, 2009.

[17] U. Fayyad, D. Haussler, P. Stolorz, Mining scientific data, Communications of the ACM 39 (1996) 51–57.

[18] H. Peng, F. Long, C. Ding, Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Transactions on pattern analysis and machine intelligence 27 (2005) 1226–1238.

[19] L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, Journal of machine learning research 5 (2004) 1205–1224.

[20] R. Kohavi, G. H. John, Wrappers for feature subset selection, Artificial intelligence 97 (1997) 273–324.

[21] Z. M. Hira, D. F. Gillies, A review of feature selection and feature extraction methods applied on microarray data, Advances in bioinformatics 2015 (2015).

[22] A. W. Whitney, A direct method of nonparametric measurement selection, IEEE Transactions on Computers 100 (1971) 1100–1103.

[23] P. Pudil, J. Novovičová, J. Kittler, Floating search methods in feature selection, Pattern recognition letters 15 (1994) 1119–1125.

[24] C. Ding, H. Peng, Minimum redundancy feature selection from microarray gene expression data, Journal of bioinformatics and computational biology 3 (2005) 185–205.

[25] O. Räsänen, J. Pohjalainen, Random subset feature selection in automatic recognition of developmental disorders, affective states, and level of conflict from speech., in: INTERSPEECH 2016, pp. 210–214.

[26] J. Pohjalainen, S. Kadioglu, O. Räsänen, Feature selection for speaker traits., in: INTERSPEECH 2012, p. 270 273.

[27] B. Arguello, et al., A survey of feature selection methods: algorithms and software, Ph.D. thesis, 2015.

[28] G. Chandrashekar, F. Sahin, A survey on feature selection methods, Computers & Electrical Engineering 40 (2014) 16–28.

[29] R. B. Pereira, A. Plastino, B. Zadrozny, L. H. Merschmann, Categorizing feature selection methods for multi-label classification, Artificial Intelligence Review 49 (2018) 57–78.

[30] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, A review of feature selection methods on synthetic data, Knowledge and information systems 34 (2013) 483–519.

[31] T. H. Cormen, Introduction to algorithms, MIT press, 2009.

[32] H. Yamada, Y. Matsumoto, Statistical dependency analysis with support vector machines, in: Proceedings of IWPT, volume 3, pp. 195–206.

[33] N. Hu, R. B. Dannenberg, G. Tzanetakis, Polyphonic audio matching and alignment for music retrieval, Computer Science Department (2003) 521.

[34] Y.-H. Yun, W.-T. Wang, M.-L. Tan, Y.-Z. Liang, H.-D. Li, D.-S. Cao, H.-M. Lu, Q.-S. Xu, A strategy that iteratively retains informative variables for selecting optimal variable subset in multivariate calibration, Analytica Chimica Acta 807 (2014) 36–43.

[35] K. S. Mark J. Embrecht, Boleslaw Szymanski, Computationally Intelligent Hybrid Systems: The Fusion of Soft Computingand Hard Computing, in: Introduction to Scientific Data Mining: Direct Kernel Methods & Applications, Wiley, New York, 2005, pp. 317 – 365.

[36] J. Pohjalainen, O. Räsänen, S. Kadioglu, Feature selection methods and their combinations in high-dimensional classification of speaker likability, intelligibility and personality traits, Computer Speech & Language 29 (2015) 145–171.

[37] L. Yu, C. Ding, S. Loscalzo, Stable feature selection via dense feature groups, in: Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, ACM, pp. 803–811.

[38] J. Reunanen, Overfitting in making comparisons between variable selection methods, Journal of Machine Learning Research 3 (2003) 1371–1382.

[39] S. Li, E. J. Harner, D. A. Adjeroh, Random knn feature selection-a fast and stable alternative to random forests, BMC bioinformatics 12 (2011) 1.

[40] L.-J. Zhang, Z.-J. Li, H.-W. Chen, J. Wen, Minimum redundancy gene selection based on grey relational analysis, in: Sixth IEEE International Conference on Data Mining-Workshops (ICDMW'06), IEEE 2006, pp. 120–124.

[41] J. Han, J. Pei, M. Kamber, Data mining: concepts and techniques, Elsevier, 2011.

[42] J. N. Qinbao Song, G. Wang, A Fast Clustering-Based Feature subset Selection Algorithm for Hihg Dimensional Data, in: IEEE Transactions on Knowledge and Data Engineering, volume 25, IEEE, 2013.

[43] G. H. John, R. Kohavi, K. Pfleger, et al., Irrelevant features and the subset selection problem, in: Machine learning: proceedings of the eleventh international conference 1994, pp. 121–129.

[44] M. Lichman, Uci machine learning repository, 2013, URL http://archive. ics. uci. edu/ml 114 (2015).

[45] A.-C. Haury, P. Gestraud, J.-P. Vert, The influence of feature selection methods on accuracy, stability and interpretability of molecular signatures, PloS one 6 (2011) e28210.

[46] K. Dunne, P. Cunningham, F. Azuaje, Solutions to instability problems with sequential wrapper-based approaches to feature selection, Journal of Machine Learning Research (2002) 1–22.

[47] P. Somol, J. Novovicova, Evaluating stability and comparing output of feature selectors that optimize feature subset cardinality, IEEE Transactions on Pattern Analysis and Machine Intelligence 32 (2010) 1921–1939.

[48] F. Yang, K. Mao, Robust feature selection for microarray data based on multicriterion fusion, IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 8 (2011) 1080–1092.

[49] A. Kalousis, J. Prados, M. Hilario, Stability of feature selection algorithms: a study on high-dimensional spaces, Knowledge and information systems 12 (2007) 95–116.

[50] A. L. Blum, P. Langley, Selection of relevant features and examples in machine learning, Artificial intelligence 97 (1997) 245–271.

[51] L. Al Shalabi, Z. Shaaban, B. Kasasbeh, Data mining: A preprocessing engine, Journal of Computer Science 2 (2006) 735–739.

[52] I. B. Mohamad, D. Usman, Standardization and its effects on k-means clustering algorithm, Research Journal of Applied Sciences, Engineering and Technology 6 (2013) 3299–3303.

[53] A. Marcano-Cedeno, J. Quintanilla-Domínguez, M. Cortina-Januchs, D. Andina, Feature selection using sequential forward selection and classification applying artificial metaplasticity neural network, in: IECON 2010-36th Annual Conference on IEEE Industrial Electronics Society, IEEE, pp. 2845–2850.

[54] D. L. Padmaja, B. Vishnuvardhan, Influence of data geometry in random subset feature selection (2017).

[55] B. Xu, J. Z. Huang, G. Williams, Y. Ye, Hybrid weighted random forests for classifying very high-dimensional data, International Journal of Data Warehousing and Mining 8 (2012) 44–63.

[56] V. Sazonau, Implementation and evaluation of a random forest machine learning algorithm, University of Manchenster (2012) 9.

[57] L. Breiman, et al., Statistical modeling: The two cultures (with comments and a rejoinder by the author), Statistical Science 16 (2001) 199–231.

[58] C. Vens, F. Costa, Random forest based feature induction, in: 2011 IEEE 11th International Conference on Data Mining, IEEE, pp. 744–753.

[59] J. Rogers, S. Gunn, Identifying feature relevance using a random forest, in: Subspace, Latent Structure and Feature Selection, Springer, 2006, pp. 173–184.

[60] F. Livingston, Implementation of breiman's random forest machine learning algorithm ece591q machine learning journal paper, 2005.

[61] T.-T. Nguyen, J. Z. Huang, T. T. Nguyen, Unbiased feature selection in learning random forests for high-dimensional data, The Scientific World Journal 2015 (2015).

[62] Li, S, Harner, J., Adjeroh, D, Random kNN feature selection – a fast and stable alternative to random forests. BMC Bioinformatics (2011).

[63] C. Liu, W. Wang, Q. Zhao, X. Shen, M. Konan, A new feature selection method based on a validity index of feature subset, Pattern Recognition Letters 92 (2017) 1–8.

[64] D. L. Padmaja, B. Vishnuvardhan, Comparative study of feature subset selection methods for dimensionality reduction on scientific data, in: 2016 IEEE 6th International Conference on Advanced Computing (IACC), pp. 31–34.

[65] V. Kumar, S. Minz, Feature selection: A literature review, Smart CR 4 (2014) 211–229.

[66] P. Cichosz, Data Mining Algorithms: Explained Using R, John Wiley & Sons, 2014.

[67] D. Agarwal, E. Gabrilovich, R. Hall, V. Josifovski, R. Khanna, Translating relevance scores to probabilities for contextual advertising, in: Proceedings of the 18th ACM conference on Information and knowledge management, ACM, pp. 1899–1902.

[68] L. C. Molina, L. Belanche, À. Nebot, Feature selection algorithms: a survey and experimental evaluation, in: Data Mining, 2002. ICDM 2003. Proceedings. 2002 IEEE International Conference on, IEEE, pp. 306–313.

[69] R. O. Duda, P. E. Hart, D. G. Stork, Pattern classification. 2nd, Edition. New York (2001) 55.

[70] J. Reunanen, et al., Overfitting in feature selection: Pitfalls and solutions (2012).

[71] M. Stone, Cross-validatory choice and assessment of statistical predictions, Journal of the Royal Statistical Society. Series B (Methodological) (1974) 111–147.

[72] T. Chai, R. R. Draxler, Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature, Geoscientific Model Development 7 (2014) 1247–1250.

[73] M. H. Doughabadi, H. Bahrami, F. Kolahan, Evaluating the effects of parameters setting on the performance of genetic algorithm using regression modeling and statistical analysis, Industrial Engineering (2011) 61–68.

[74] D. L. Padmaja, B. Vishnuvardhan, Survey of dimensionality reduction and mining techniques on scientific data, International Journal of Computer Science & Engineering Technology 1 (2014) 1062–1066.

[75] M. E. Farmer, S. Bapna, A. K. Jain, Large scale feature selection using modified random mutation hill climbing, in: Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on, volume 2, IEEE, pp. 287–290.

[76] H. Alshamlan, G. Badr, Y. Alohali, mrmr-abc: A hybrid gene selection algorithm for cancer classification using microarray gene expression profiling, BioMed research international 2015 (2015).

[77] N. Ramakrishnan, A. Y. Grama, Mining scientific data, Advances in computers 55 (2002) 119–VIII.

[78] E. Maltseva, C. Pizzuti, D. Talia, Mining high-dimensional scientific data sets using singular value decomposition, in: Data Mining for Scientific and Engineering Applications, Springer, 2001, pp. 425–438.