A Modified K-Nearest Neighbor Algorithm Using Feature Optimization

Rashmi Agrawal
Faculty of Computer Applications, Manav Rachna International University
rashmi.sandeep.goel@gmail.com

***Abstract* - A classification technique is an organized approach for building classification model from given input dataset. The learning algorithm of each technique is employed to build a model used to find the relationship between attribute set and class label of the given input data. Presence of irrelevant information in the data set reduces the speed and quality of learning. The technique of feature selection reduces the amount of data needed and execution time and it also improves the accuracy for prediction in the classification problem. In this paper we have modified K- Nearest Neighbor algorithm with relevant feature selection which selects the relevant features and removes irrelevant features of the dataset automatically.**

Keywords- Classification, K Nearest Neighbor, Feature Space, Filter Approach, Wrapper Approach, Relevant Feature Selection, Covariance, Covariance Matrix

I. INTRODUCTION

The classification task in data mining is commonly referred as supervised learning in which specified set of classes are known and training sample objects are assigned with appropriate class. The aim of classification technique is to build a model with best generalization capability. For the supervised learning task, we represent data as a table of training samples, also known as instances, which are described by features and their classes. Features are also called as attributes. Generally, we require two sets of data tables, training dataset and test dataset. We train the classification algorithm on training dataset and test the algorithm using test dataset. Various classification techniques used are Decision Tree, Naïve Bayes, and Nearest Neighbor. K- Nearest Neighbor is one of simple and well known classification technique in which distance is measured between input point and all other records of the dataset. The class label of the K-Nearest Neighbor is the class label for input point.

It is known that if too much irrelevant information is present in the training or test data, the learning and prediction becomes more difficult and inaccurate. The process of identification of relevant features and removal of irrelevant features from the data is known as feature selection, which is also known as attribute selection, or subset selection or variable selection. The technique of feature selection gives the advantage of reduction in the amount of data needed, reduced execution time and improved accuracy for prediction in classification problem.

In this paper, we propose a modified k-nearest neighbor algorithm with relevant feature selection (RFS-KNN) which selects the relevant features and removes irrelevant features of dataset automatically.

II. RELATED WORK

As an improvement to KNN, Dudani [1] introduced distance-weighted KNN (WKNN) algorithm. However, WKNN does not produce satisfactory results due to the existence of outliers, particularly in small sample size dataset.

On the basis of WKNN by Dudani, a new distance-weighted k-nearest neighbor rule (DWKNN) was given by Gou [2] using dual distance-weighted function. They employed the dual distance-weights neighbors to find out the class of the object. Simply majority voting for KNN may not be effective if the neighbors vary widely with their distances. In DWKNN, the original weight is multiplied with a new weight to determine the dual weight. This method reduces the weight of nearest neighbors and provides too much weight to the outliers as compared to the WKNN and thus improves the classification performance. However, DWKNN is not effective with irregular class distribution. Zuo and Zhang [9] defined the weighted KNN rule as a constrained optimization problem and contributed a kernel difference-weighted k- nearest neighbor (KDF-KNN).

Identification of relevant features and removal of other irrelevant features has been an interesting problem in the area of machine learning. In 1994, Langley Pat [4] studied this problem and described it in the form of heuristic search. He presented the task of feature selection as a search problem having a subset of possible features in each state. The four issues were addressed-

a) To determine forward selection or backward selection
b) Adding and removing features at each decision point.
c) Strategy to evaluate alternative subset of attributes (filter method or wrapper method).
d) Criteria for halting the search.

Feature selection is of extreme importance to enhance the speed of learning and to improve the quality. Kira and Render [5] presented a new feature selection algorithm RELIEF which uses a statistical method and does not include heuristic search. This algorithm takes the assumption that scale of every feature is either nominal or numerical. A function is used to update the feature weight vector for every sample and the average feature weight called Relevance is determined. But this algorithm is valid only when the relevance is large for relevant features and small for other features.

Elena Marchiori [6] investigated a decomposition of RELIEF into the class dependent feature weight terms. They showed that when complementary characteristics of a feature in different classes are added, they neutralize each other otherwise they may give different weight contributions. Consequently, relevance of some features for a single class may not be detected.

In feature selection techniques, generally a search strategy is incorporated [7] to explore the space of subsets of features which includes methods for finding starting point and generating candidate subsets and evaluation criteria to compare the suitability of candidates. This evaluation scheme can be divided into two broad categories-

a) Filter Approach- In this approach, the irrelevant features are removed from the set of features before applying the learning objective.
b) Wrapper Method- In this method, the learning algorithm is used to select the features from the feature set.

The limitation of filter approach is that features are considered in isolation. Therefore two strongly correlated features either may be ignored or may be redundant. Wrapper method overcomes the limitations of filter approach as classifier is itself wrapped in feature selection process. It is done either through forward selection or through backward selection. The forward selection starts with no features and each feature is added at a step. The backward selection process starts by considering all features initially and irrelevant features are removed at every step.

In 2001, Das [8] examined the pros and cons of filter and wrapper methods used in feature selection and proposed a new hybrid algorithm. This algorithm was based on the concept of boosting from computational learning theory. He presented Boosted Decision Stumps for Feature Selection (BDSFS) algorithm by using AdaBoost to bridge the gap and giving more informed filter method. The algorithm used boosted decision stumps as the weak learners. The algorithm was, however, not well suited for multi-class datasets.

Feature Space

In a feature selection algorithm, searching space of feature subsets in limited time is necessary. For this the existing feature selection algorithm uses the forward or backward selection technique. In the forward selection, the algorithm starts with an empty feature set and adds relevant features at each step, whereas in the backward selection algorithm, the algorithm starts with a full feature set and deletes irrelevant features at each step. Fig 1 and 2 show the forward selection of a 3 feature dataset.

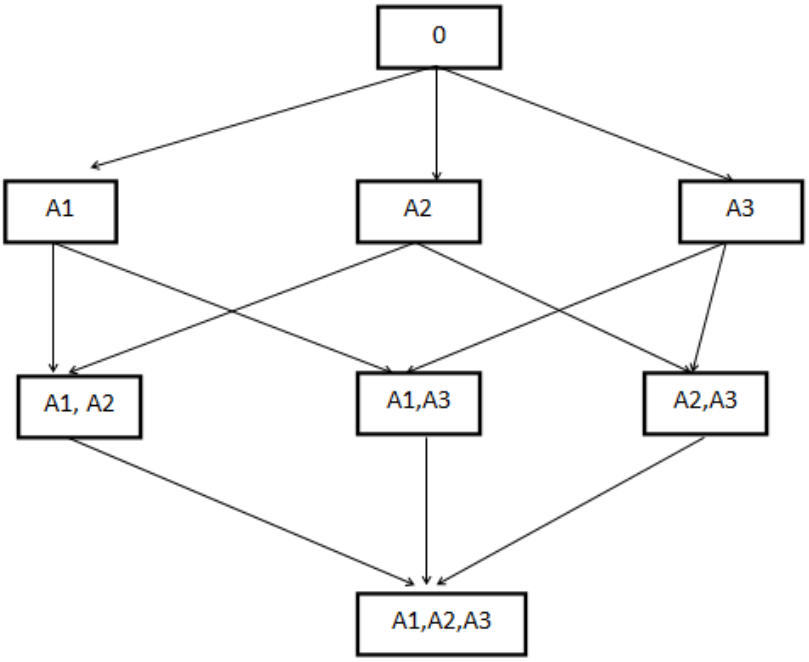

Fig 1: Feature subset space using forward selection

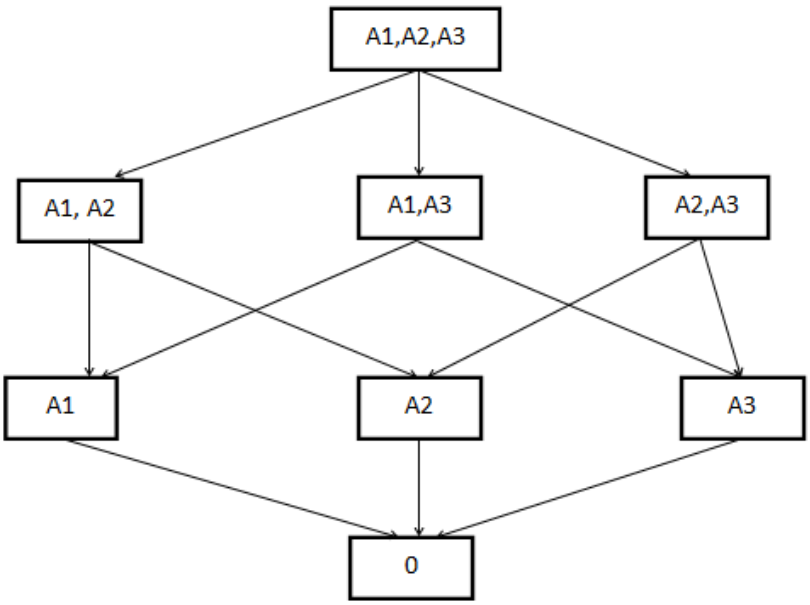

Fig 2: Feature subset space using backward selection

In our proposed feature selection technique to generate the feature space, we start with backward selection with all features along with the output class to generate a feature subset with 2n possible subsets. Fig 3 shows the feature subset space using proposed approach.

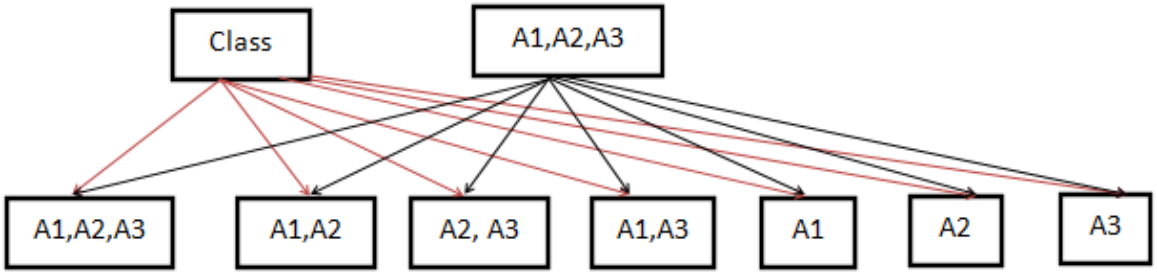

Fig 3: Feature subset space using proposed approach

Bupa is a medical research database of blood tests which is thought to be sensitive to liver disorders arising from excessive alcohol consumption. Each record of Bupa dataset represents the record of a single male individual. Features used in this dataset are as under:-

A1	mcv	Mean Corpusclar Volume
A2	alkphos	Alkaline Phosphotase
A3	sgpt	Alamine Aminotransferase
A4	sgot	Aspartate Aminotransferase
A5	gammogt	Gamma Glutamyl Tanspeptidase
A6	drinks	Number of half-pint equivalents of alcoholic beverages drunk per day
Class	selector	output class

Fig 4 represents the feature subset space using RFS for Bupa dataset

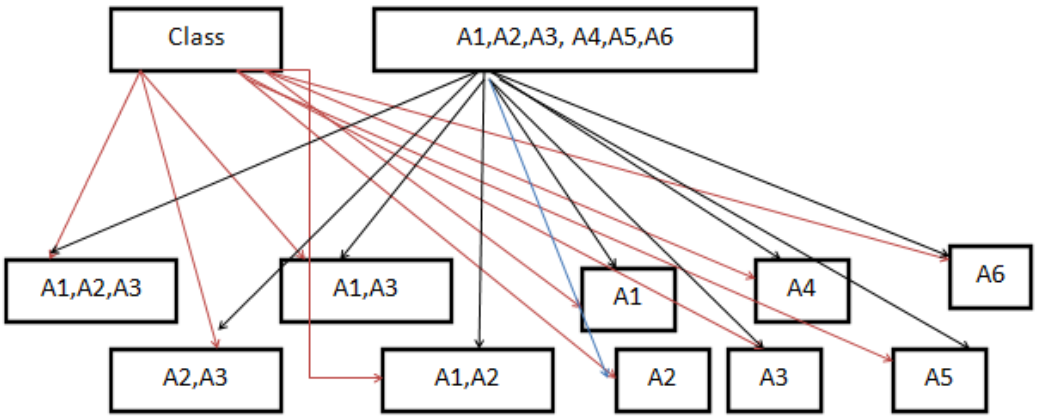

Fig 4: Feature subset space using RFS for Bupa dataset

III. Feature Selection Algorithm

Description of Modified k-Nearest Nearest Neighbor Algorithm for Relevant Feature Selection (RFS-KNN)

We develop a modified k- Nearest Neighbor Algorithm for Relevant Feature Selection (RFS-KNN). The algorithm does not require any input on the number of features to be selected and hence adopts a filter approach. The algorithm works on the concept that if two features are highly correlated (either positively or negatively), their importance in predicting the class label is negligible and so these features are irrelevant in classification. On the other hand, if features are highly correlated with class label, they take a prime role in predicting the class label. Also, if the variance of a feature is less it means the population will exhibit almost same characteristics and if it is more, that means it will exhibit different characteristics. The figure 5 represents the framework of RFS-KNN. In the RFS-KNN, relevant features from a dataset are selected using the RFS module and then the dataset with selected features is passed to KNN algorithm for prediction.

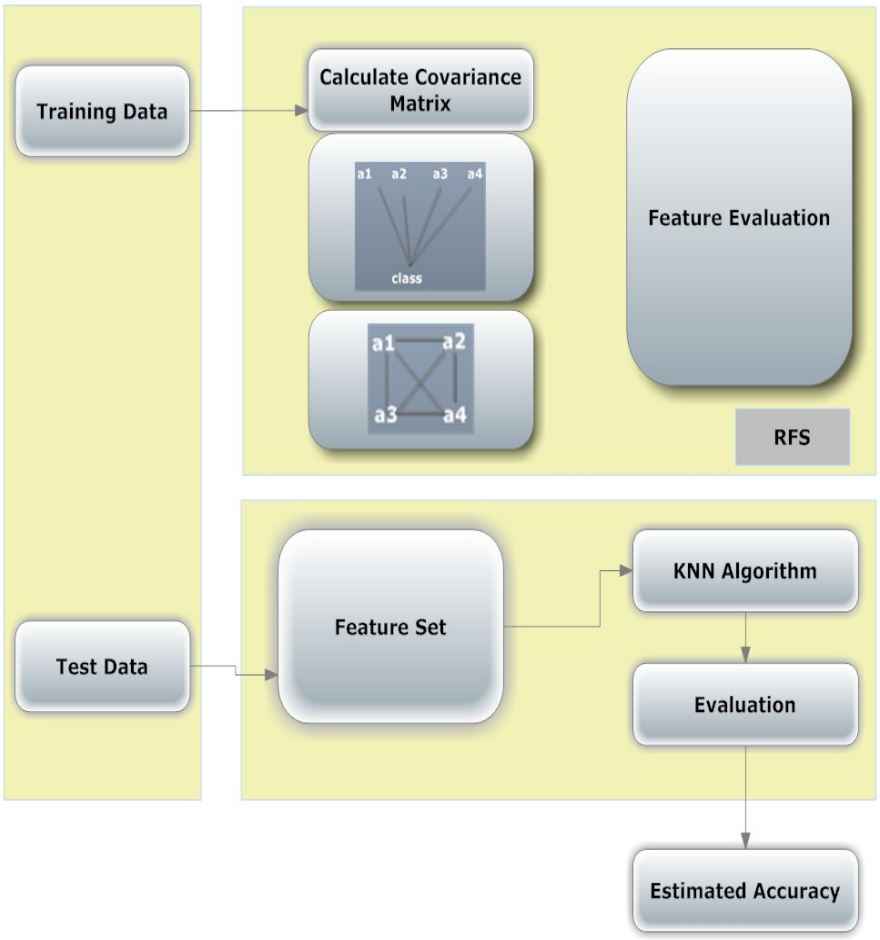

Fig 5 : Framework of Relevant Feature Selection KNN (RFS-KNN) Algorithm

Based on the facts discussed above, we build the variance covariance matrix of all features including class label of training data set. The following table shows the variance-covariance matrix of the Bupa dataset having six features. The values at the main diagonal (shown in the bold face) represent variance of the features.

TABLE 1: Variance-Covariance matrix of Bupa dataset

	A1	**A2**	**A3**	**A4**	**A5**	**A6**	**Class**
A1	**15.56533**	0.40427	7.70553	3.41859	26.58191	3.50465	-0.16055
A2	0.40427	**294.93847**	29.58121	22.99432	99.43678	6.58170	-0.61912
A3	7.70553	29.58121	**348.58854**	119.01427	371.88663	15.10548	-0.22643
A4	3.41859	22.99432	119.01427	**83.93156**	163.94010	8.96761	0.97402
A5	26.58191	99.43678	371.88663	163.94010	**1498.98131**	49.77327	2.88583
A6	3.50465	6.58170	15.10548	8.96761	49.77327	**10.11532**	0.04180
Class	-0.16055	-0.61912	-0.22643	0.97402	2.88583	0.04180	0.24701

From the variance-covariance matrix, first we find the variance of a feature at diagonals and then covariance between features and after that covariance of features and output class is analyzed. We know that if the variance of feature is less than a certain threshold value (ƛv), the feature will not exhibit variation in the sample and may be considered as a constant value. We find such feature as irrelevant and remove it from feature subset. In this paper throughout the experiment, we have taken ƛv as 0.0001.

In the next step, we analyze the covariance between features excluding the class label which is represented by last row and last column in variance-covariance matrix. If covariance is high among the features, they are less involved in prediction, hence we find them little irrelevant and remove from feature subset and put in a temporary subset. For this we set a threshold (ƛc) and the features with covariance more than the threshold ƛc are moved to a temporary subset. Here in this paper, throughout the experiment, we have taken ƛc as 0.30. We also analyze the covariance between a feature and class and if it is more than a threshold value (ƛcc), we select them as highly relevant and put them in feature space. Simultaneously, we compare these highly relevant features with the features of temporary dataset having high covariance among features and if similar feature is found we move that feature from temporary dataset to feature subset. Thus the feature subset contains only relevant features. . We take ƛcc as 0.30. After the selection of relevant features, using RFS module, KNN is applied on the training dataset and test dataset to predict the accuracy.

To achieve this mathematically, we used sets A, B and C to store feature sets. The set A contains the feature having covariance more than a threshold value ƛc with each other excluding class label, whereas set B contains the features having covariance more than ƛcc on class label. Set C contains the features having variance below ƛv . Set U represents the universal set of all features.

Then the set I given by

$I = (A \setminus B) \cup C$

will determine set of irrelevant features.

and set of relevant features 'R' is given as

$R = U \setminus I$

Algorithm for RFS-KNN

Declare A, B, C , U, IR, R,F as array

Comments:

// The Collection A will contain the feature having covariance more than a threshold value ƛc

// Collection B will contain the features having covariance more than a ƛcc

// Collection C will contain the features having variance below ƛv

// Collection U contains all feature labels, Collection IR will contain irrelevant feature labels and collection R will contain relevant feature labels

// Collection F contains all feature labels of the data set

Initialize array A, B C, IR and R as empty and array U with full feature labels.
Cov[m,n] = Get_Covariance_Matrix(Training data)

> Where Get_Covariance_Matrix is a user defined function used to calculate covariance matrix of the given data with m features including class label.

```
/*  This loop is used to populate collection C with feature labels /*
For i= 1 to m-1
  Start loop
          For j= 1 to n-1
           Start loop
              If i==j
                 Then
                    If cov[i,j]<= λv
                             Then
                         Add F[i] to collection C
                            End If
                End if
            End Loop
   End Loop
/*  This loop is used to populate  collection A with feature labels /*
For i= 1 to m-1
  Start loop
          For j= 1 to n-1
           Start loop
              If i !=j
                 Then
                    If cov[i,j]> λc
                             Then
                         Add F[i] to collection A
                            End If
                End if
            End Loop
   End Loop
/* This loop is used to populate collection B with feature labels /*
Set i:=m
For j= 1 to n-1
Start loop
 If cov[i,j]> λcc
     Then
          Add F[i] to collection B
End if
End Loop
```

> Here we apply SET DIFFERENCE operation in place of the minus operator

Set IR:=((Collection A –Colletion B) UNION Collection C)
Set R:= (Collection U-Collection IR)

/* collection R now contains the relevant feature labels. The RFS-KNN algorithm only selects R features from the dataset and traditional KNN is applied thereafter*/

The k-nearest neighbor classification algorithm

1. let k be the number of nearest neighbors and D be the set of training examples
2. for each test example z=(x', y') do
3. compute d(x',x) , the distance between z and every example, (x,y) ∈ D
4. select Dz ⊆ D, the set of k closest training examples to z.
5. y' = argmax xi, yi ∈ Dz I (v=yi)
6. end for

IV. EVALUATION OF RFS-KNN

To evaluate the RFS-KNN, we used five data sets from UCI machine learning repository [3]. These data sets have been chosen due to their predominance in the literature. A short description of these data sets is given below:-

Glass data set:- This dataset was given by USA Forensic Science Service in which 7 types of glasses are defined in terms of their oxide content. The study of classification of types of glass was motivated by criminological investigation. At the crime scene, the left glass can be used as an evidence of the crime if it is correctly identified. This dataset contains 10 features.

Mushroom: This dataset was given by Audubon Society Field Guide. This dataset is used for classification of mushrooms as poisonous or edible. This classification task is done on the basis of 22 nominal attributes describing characteristics of mushrooms such as cap shape, odour and give gill spacing. This is a large data set.

Bupa: Bupa dataset was donated by BUPA Medical Research Ltd. This is a multivariate dataset with 345 instances and 6 attributes. The dataset is used for classification on the basis of blood tests which are thought to be sensitive to liver disorders.

E.Coli:- This dataset contains protein localization sites. This is a small dataset with 336 numbers of instances and 7 attributes. Attributes are real in nature.

Pima: Pima dataset was given by National Institute of Diabetes and Digestive and Kidney diseases. This dataset having 8 features is used to classify, whether a person has not diabetes (tested- negative) or the person has diabetes (tested-positive).

A short description of these datasets is given below in the table 2-

TABLE 2: Description of dataset used in experiment

S.no.	Data Set	No of class	Training Sample	Test Sample
1	Glass	7	136	78
2	Mushroom	2	300	200
3	Bupa	2	200	145
4	Ecoli	5	206	121
5	Pima	2	400	368

We applied RFS-KNN on these 5 datasets and results have been summarized in table 4. For this experiment, we set λv as 0.0001, λc as 0.30 and λcc as 0.30. We selected relevant features 'R' and irrelevant features 'IR' and applied KNN algorithm on the training and test dataset with 'R' feature. The table 3 shows the 'IR' and 'R' features along with the total number of features of the datasets used in the experiment.

TABLE 3: Relevant and Irrelevant Features

S.no.	Data Set	Total Number of Features	IR	R
1	Glass	9	3	6
2	Mushroom	21	12	9
3	Bupa	6	3	3
4	Ecoli	7	1	6
5	Pima	8	1	7

As an illustration, we apply RFS-KNN on the Bupa dataset and after evaluating the variance-covariance matrix given in table 1, we generate the elements of set A,B, C, U, R and IR as follows-

U= { A1, A2, A3, A4, A5, A6} ,

A= { A1, A2, A3, A4, A5, A6} ,

B= { A2,A4, A5} ,

C={ } ,

I= (A \ B) ∪ C = { A1, A3, A6},

R= U – I = {A2, A4, A5}.

It follows that the features A2, A4 and A5 are relevant and applying the KNN algorithm on Bupa dataset with these relevant features we achieve 59.5 % accuracy as compared to 52.5 % accuracy without feature selection (see table 8).

The confusion matrix of Bupa dataset without feature selection and with feature selection using RFS-KNN is shown in table 4, 5, 6 and 7.

TABLE 4: Confusion Matrix per true class of Bupa dataset without feature selection

True Class	**1**	51.7%	48.3%
	2	46.9%	53.1%
	Confusion Matrix (Per True Class)	**1**	**2**
	Predicted Class		

TABLE 5: Overall Confusion Matrix of Bupa dataset without feature selection

True Class	**1**	22.5 %	21 %
	2	26.5 %	30.0%
	Confusion Matrix (Overall)	**1**	**2**
	Predicted Class		

TABLE 6: Confusion Matrix per true class of Bupa dataset with feature selection

True Class	**1**	67.5%	32.5 %
	2	48.7 %	51.3 %
	Confusion Matrix (Per True Class)	**1**	**2**
	Predicted Class		

TABLE 7: Overall Confusion Matrix of Bupa dataset with feature selection

True Class	1	30.5 %	15 %
	2	27.5 %	29 %
	Confusion Matrix (Overall)	1	2
	Predicted Class		

The comparative results of five datasets with traditional KNN and RFS-KNN has been shown in the table below-

TABLE 8: Accuracy of RFS-KNN on five datasets

S.no.	Data Set	Without feature selection (accuracy in %)	RFS-KNN (accuracy in %)
1	Glass	70	70.5
2	Mushroom	100	100
3	Bupa	52.5	59.5
4	Ecoli	79.6	84.3
5	Pima	64.5	69

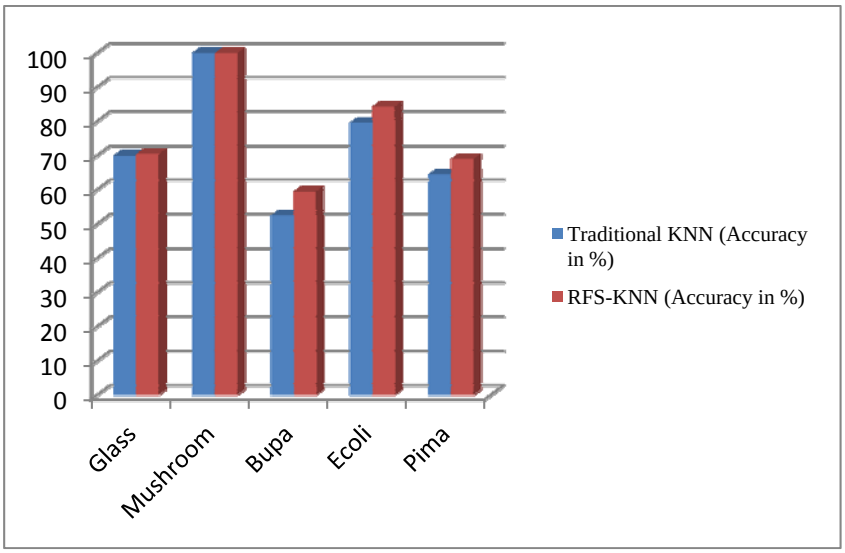

Fig 6 : Accuracy of RFS-KNN

The results reported in table 8 show that our algorithm RFS-KNN produces better results as compared to the traditional KNN.

V. CONCLUSION

Presence of irrelevant information in the dataset makes the learning and prediction process difficult and inaccurate. In this paper, we have developed a modified K-Nearest Neighbor algorithm with relevant feature selection (RFS-KNN) to select the relevant features and remove irrelevant features from the dataset automatically. The implementation of the algorithm on five datasets taken from UCI machine learning repository proved that our algorithm RFS-KNN gives better results in terms of accuracy and also reduces the amount of data used in prediction process thereby reducing execution-time.

ACKNOWLEDGEMENT

My thanks are due to my supervisor Dr Babu Ram, Professor, Manav Rachna International University, Faridabad for his constant guidance and support.

REFERENCES

[1] Dudani, Sahibsingh A. "The distance-weighted k-nearest-neighbor rule." Systems, Man and Cybernetics, IEEE Transactions on 4 (1976): 325-327.
[2] Gou, J., et al. "A new distance-weighted k-nearest neighbor classifier." J. Inf. Comput. Sci 9 (2012): 1429-1436.
[3] http://archive.ics.uci.edu/ml/.
[4] Selection of relevant features in machine learning, Proceedings AAAI Fall Symposium on Relevance, New Orleans, LA (1994), pp. 140–144
[5] Kira, Kenji, and Larry A. Rendell. "A practical approach to feature selection."Proceedings of the ninth international workshop on Machine learning. 1992.
[6] Marchiori, Elena. "Class dependent feature weighting and k-nearest neighbor classification." Pattern Recognition in Bioinformatics. Springer Berlin Heidelberg, 2013. 69-78.
[7] Cunningham, Padraig, and Sarah Jane Delany. "k-Nearest neighbour classifiers." Multiple Classifier Systems (2007): 1-17.
[8] Das, Sanmay. "Filters, wrappers and a boosting-based hybrid for feature selection." ICML. Vol. 1. 2001.
[9] Zuo, Wangmeng, David Zhang, and Kuanquan Wang. "On kernel difference-weighted k-nearest neighbor classification." Pattern Analysis and Applications11.3-4 (2008): 247-257.

AUTHOR PROFILE

Rashmi Agrawal Author is working as Associate Professor in Manav Rachna International University. She has more than 15 years of teaching experience in the area of Computer Applications. Her area of expertise includes Data Mining and Artificial Intelligence. She is actively involved in research and published many research papers in various national and international journals of repute. She is the life member of CSI.