FSEFST:Feature Selection and Extraction using Feature Subset Technique in High Dimensional Data

Radhika K R, Pushpa C N, Thriveni J, Venugopal K R

Abstract- Dimensionality reduction is one of the pre-processing phases required when large amount of data is available. Feature selection and Feature Extraction are one of the methods used to reduce the dimensionality. Till now these methods were using separately so the resultant feature contains original or transformed data. An efficient algorithm for Feature Selection and Extraction using Feature Subset Technique in High Dimensional Data (FSEFST) has been proposed in order to select and extract the efficient features by using feature subset method where it will have both original and transformed data. The results prove that the suggested method is better as compared with the existing algorithm

Keywords : Dimensionality reduction, Feature Extraction, Feature Selection, Feature Subset, High Dimensional Data.

I. INTRODUCTION

Present data which is in digital form and storage technology has contributed to the development of high dimensional data. The dimensionality problem can be solved by a feature selection approach which removes irrelevant and redundant features [1]. The major challenges faced by the researchers in high dimensional data are the selection of subset feature and classification [2]. Methods like Filter, wrapper and embedding are used to get more efficient and informative features. Using the symmetrical uncertainty method the predominant features can be predicted. The core idea of a feature selection or extraction algorithms is to eliminate the irrelevant features by selecting a subset of original features.

Different algorithms for feature selection can be used, they are classified into Supervised, Unsupervised and Semi supervised algorithms [3]. In case of Supervised method important features can be obtained by finding the correlation with the class labels, unsupervised methods find the features by exploiting data variance or by distribution. Whereas semi supervised methods extract the features by considering the trivial amount of label data as added information in order to get the efficient features [4].

Revised Manuscript Received on December 30, 2019.
* Correspondence Author
Radhika K R*, Department of CSE, Bangalore University,Bangalore, India,. Email:radhika@bmsit.in
Pushpa C N, Department of CSE, UVCE, Bangalore University, Bangalore, India.
Dr.Thriveni J, Department of CSE, UVCE, Bangalore University, Bangalore, India.
Dr. Venugopal K R,Vice-Chancellor, Bangalore University, Bangalore, India.

The High dimensional data can be transformed into low dimensional by Feature Extraction. Feature Extraction methods are also characterised into supervised and unsupervised methods.

The familiar feature extraction methods are Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA). The PCA and LDA methods will find reduced set in original or converted but not both. An approach is developed here where both original and transformed features can be obtained by considering the suitable threshold.

Motivation: The selection of subset of features from the original data by removing the data which are not relevant is called feature selection. Feature selection can be done by using supervised, unsupervised and semi-supervised algorithms. Feature extraction is to convert the high dimensional data to low dimensional data by linearly or non-linearly. The most common linear abstraction methods are Principal Component Analysis (PCA) and Linear Discrimination Method (LDM).

Contribution: The reduced set which will be obtained is from original or transformed features but not from both. The proposed algorithms bridge the gap between these two. In the proposed algorithms projection error is calculated by considering two features and redundancy removal is also based on these features. In order to have the most efficient calculation of projection error the previous and current features can be used to get constant threshold. An algorithms as proposed to select efficient number from two different combinations of features.

Organization: The paper organization is as follows: Related work is given in Section II, Section III gives the proposed Methodology, Proposed algorithms are explained in Section IV Performance Analysis and Conclusion are given in Section V and VI respectively.

II. RELATED WORK

Sreevani et al., [5] discuss about the popular approaches for dimensionality reduction, they are feature selection and feature extraction. They proposed a framework called Minimum Projection error Minimum Redundancy (MPeMR) to obtain original and converted features in the reduced set. The proposed method is compared with advanced feature selection and extraction methods. Results are compared with supervised methods.

Smita Chormunge et al., [6] discuss about IFSA (Information gain based Feature Selection Algorithm) to get an ideal feature subset in less time and progress the computational performance of learning algorithms.

The above mentioned algorithm executes in two fold first the filter method is executed on the dataset and second, obtain minor feature subset by using the information gain method. Abeer Alzubaidi et al., [7] introduced three phases of feature filtering methods first phase removes noisy features, second phase is feature selection which is based on multivariate machine learning technique and the third method retrieve the filtered features using Genetic algorithm. Augusto Destrero et al., [8] introduced the benefits of adopting a regularized approach with L_1 or $L_1 - L_2$ penalties to extract features in high dimensional data in two different applications like microarray analysis in computational biology and object detection in computer vision.

D. Asir Antony Gnana Singh et al., [8] emphasized on how Machine learning can be used to build the learning model to obtain efficient features in less time, which also increases the accuracy in learning process. Priyanka Jindal et al., [9] has given the analysis of some present feature selection and extraction techniques and has addressed the advantages and challenges of these algorithms. Bharat Singh et al., [10] applied the filter method to categorise the features based on score and measured the performance by applying on four Data Mining classification algorithms. The proposed method has shown the improvement in precision of selected features. Hoang Vu Nguyen et al., [11] introduced efficient feature extraction method which improves the detection accuracy when applied on two detection techniques.

Yue WU et al., [12] applied maxHeap based approach instead of OFS (Online Feature Selection) to extract more efficient features compared to existing batch learning methods. Agnieszka Wosiak et al., [13] used SVMREF (Support Vector Machine-Recursive Feature Elimination) technique to classify both binary and multiclass dimensional set for a specific application.

ZHAO Zhongwen et al., [14] employed PCA and SVM to reduce dimension, classify the data and project the classified data to two-dimensional features. Manikandan G et al., [15] proposed a method to discover the optimal threshold value and feature subsets are given to the classifier to obtain maximum accuracy. Ding et al., [16] discuss about the novel feature selection framework for generally minimizing the feature redundancy to increase the ranking score for the given feature, which can originate from any supervised or unsupervised methods. W. Sheng et al., [17] proposed a technique to refine feature selection and to obtain the cluster centers prearranged in the chromosomes called local search operations.

D. Cai et al., [18] discussed about unsupervised feature selection to select those features which are having multi-cluster structure where the data can be preserved by the method called Multi-Cluster Feature Selection (MCFS). Z. Xu et al., [19] proposed a novel discriminative semi supervised feature selection method in order to maximize the classification margin to differentiate between labelled and unlabelled data. K. Benabdeslem et al., [20] describes about two important processes to provide an effective selection of semi-features in semi-supervised environment. P. Mitra et al., [21] introduced a method called maximum information compression index, which describes how feature selection and extraction can be improvised with an entropy measure.

Dash M et al., [22] recommended a filter technique that is independent of any clustering algorithm. The method has entropy calculation that is low if clusters are distinct and high clusters are not distinct. Song Q et al., [23] proposed a FAST algorithm which makes use of the Minimum Spanning Tree (MST) clustering means to find the efficient features. Different clusters will have relatively independent Features and FAST algorithm has a high possibility of generating a subset of useful and independent features. R Kohavi et al., [24] discussed the problems faced by learning algorithms while selecting the suitable subset features and explained how to overcome these problems using wrapper method to search for optimal feature subsets. J Liu et al., [25] discussed about regularized regression model called L2 1-norm model for combined feature selection from multiple tasks.

III. PROPOSED METHODOLOGY

A Proposed framework depends on non-projection error to generate compound features and considers all original features as candidate features. Some of the definitions are given below for the better understanding.

Definition 1: (Semi-feature). A grouping of suitable subset of original feature is called semi-feature.

Definition 2: (Feature Set (FS)). The combination of original and semi-features together is called Feature Set.

Definition 3: (Compound Feature Generation (CFG)) Generating the features among the selected original and combination of features with orthogonality.

The normal procedure in any feature extraction involves finding the eigen value and eigen vector by framing criterion function. Highest eigenvectors comprise most of the information. Least eigenvalue means the little information alongside its resultant eigenvector.

Definition 4: (Error Ratio) Given a subset of features $\{v_1, v_2 \ldots\ldots v_k)$ and a special feature extracted method D. Let $\lambda_1, \lambda_2 \ldots\ldots \lambda_k$ be the resulting eigenvalues with $\lambda_1 > \lambda_2 > \ldots\ldots \lambda_k$. Since the smallest eigenvalue λ_1 gives the data along the k^{th} section, which is considered as the total number of error presented while reducing the dimension from k to k-1. As λ_k is arbitrary, still normalization can be done by sum of all eigenvalus $\lambda_1 + \lambda_2 \ldots\ldots + \lambda_k$. This ratio is called Error Ratio for the set of features $v_1 \ldots\ldots v_k$, when dimension is reduced from k to k-1. It is known that the efficiency of the learning algorithm could be ruined when the features which are considered comprise redundant to each other if they have correlated values.

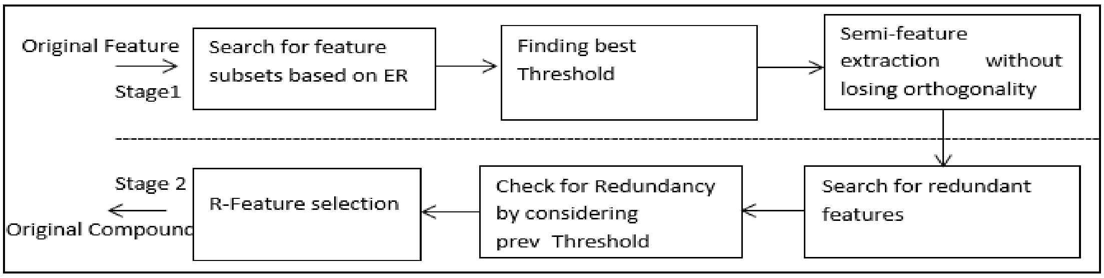
The Error Ratio is calculated for each separate pair of existing features. If the error for a pair (v_i, v_j) is fewer than the threshold T_1 which is already defined then extraction of semi feature can be done. Then select the second feature set check for the threshold T_1 if the threshold of this pair is better than the previous pair, the first pair to cross threshold will be best. Store the threshold for next comparison also store the index of v_i and v_j. Calculate the eigen component for v_i and v_j which will have the best pair. Now add the transformed feature into the semi feature set. Then delete the original feature from the respective set.

Now from the reduce function check for the reduced set by considering the threshold T_2, if the reduced set is better than the previous threshold then update prevthreshold.

Suggested technique primarily considers all features $f_1, f_2, \ldots . . f_D$ as candidate features, and two stages as to be performed in each iteration.

The stage one includes extraction of semi features and stage second is for removing the redundant data. In stage one , for each individual pair of available features, the measure of classification Error check Ratio is calculated, if the error for a pair of u_i, u_j is less than a threshold T_1 the extraction of the semi feature is performed on u_i, u_j. The proposed algorithm is better than the existing algorithm by initializing prevthreshold for every iteration to get best pair. When the second pair is selected from the set check with the previous threshold in order to get a better pair than the previous one.

IV. PROPOSED ALGORITHM



The stages of finding the compound feature generation is given in two stages which are shown in Figure. 1. The first stage involves searching for features based on (Error Ratio) ER then find the best threshold by considering the best pair to from the given set of values. The semi feature extraction can be done without losing the orthogonality. In the second stage search for the redundant features by considering prevthreshold and get the relevant features.

TABLE I. Notations used in the algorithm

Symbols	*Definitions*
N	Number of Original Features
T_1, T_2	Predefined Threshold values
D	Set of semi-features
$ER_{i,j}$	Error rate
O	Original Feature set $\{f_1, f_2, \ldots f_n\}$

The Notations used in algorithm are given in Table I. Table II gives an algorithm to find the features and the best Threshold. Initially the set D will be empty and the selected semi-features are stored in D by considering the pair V_i and V_j and if the considered pair threshold is less than T_1 then check for the prevThreshold. If it is less than PrevThreshold then the pair which considered the best pair.

TABLE II. Algorithm to search for features based on (Error Ratio) ER and to get best Threshold

```
Begin
Step 1: for each vj (≠ vi ) ∈ O ∪ D d
            if Red ( vi > vj ) > T2 then
        Red ( vi > vj ) > prevThreshold then
        prevThreshold ← Red (vi, vj)
        vi ← i
        vj ← j
      end if
   end for
Step 7: u' ← R-Feature of { i, vj }
Step 8: u'' ← weak feature of { i, vj }
Step 9: if u'' ∈ O then
        O ← O − { '' }
  else
            D ← D − { '' }
        if (Dim == |O ∪ D|) then
        flag ← FALSE
        else
        flag ← TRUE
    end if
     end if

Step 10: return C ← O ∪ D
end
```

TABLE III. Algorithm for Adding transformed Features to Semi-feature set

```
Begin
    Step 1: dk ← first eigen component vi & vj
    Step 2: D ← D ∪ { k }
    Step 3: k ← k + 1
    Step 4: if vi, vj ∈ O then
            O ← O − { i, vj }
              else if vi ∈ O, vj ∈ D then
                  O ← O − { i }
             D ← D − { j }
                          else if  vi ∈ D, vj ∈ O
           then
                               O ← O − { j }
                               D ← D − { i }
                         end if
              end if
           end if
        else
    D ← D − { i, vj }
           for each vi ∈ O ∪ D do
                   prevThreshold ← T2
           end for
end
```

From the algorithm given in Table III the transformed features are converted into semi features and semi features are extracted on v_i and v_j and stored in D and the prevThreshold is considered for the next iteration.

TABLE IV. Algorithm for Checking Redundancy

```
Begin
Step 1. Consider Set of semi-features D (Initially Empty).
Step 2. for a pair (vi , vj) where vi ∈ O ∪ D
        PrevThreshold = T1
            for each vj (≠vi ) ∈ O ∪ D do
              if ERi,j < T1 then
                if ERi,j < PrevThreshold
                   PrevThreshold ← ERi,j
                   vi ← i
                   vj ← j
             end if
           end if
        end for
    end for
end
```

From the available features (v_i,v_j) , if the redundancy between (v_i,v_j) amongst the pair is better than a threshold T_2 then R-feature is calculated by using Ranking Criterion of both the features. From the algorithm given in the Table IV one of the weak features are rejected to get the reduced set.

V. PERFORMANCE ANALYSIS

For the purpose of evaluation of performance of the framework, three different datasets have been considered whose details are given in Table V.

TABLE V. Data Description

Dataset	Instances	Attributes	Classes
Abalone	4177	8	3
Ecoli	332	7	6
Pageblocks	5473	10	5

Here advanced feature selection and extraction approach such as MPEMR is considered. The two versions of proposed algorithm is considered to obtain by passing two different parameters: projection error in rder to check for classification accuracy and redundancy algorithms. These algorithms takes parameters (Non-projection error, redundancy, R-Feature) as (LDA, RHO, RHO). These methods require two thresholds which are given in table as [(t_{11}, t_{12}), (t_{21}, t_{22})] where t_{i1} is projection error threshold and t_{i2} is redundancy threshold.

In the proposed framework, the accuracy rate for the classification has been considered as the evaluation metric. The Table VI demonstrates the accuracy rates for the classification of the KNN (K=1, 3 and 5) and SVM classifiers for the supervised dimensionality reduction algorithm by considering 3 data sets. The performance of the suggested method is compared for the classification and Redundancy accuracy while making clusters. The types of classification algorithms considered are 1) KNN (K = 1, 3, 5), and 2) Linear SVM with the parameter 'd' set to 1 in all the tests, are used to categorize data sets after dimensionality reduction.

TABLE VI. Ecoli Dataset for Classification Accuracy

EColi (d=5),T[(0.1, 0.7), (0.15, 0.85)]		
Algorithm	**Classification Accuracy**	
	MPEMR	**FSEFST**
KNN1	0.8089	**0.8121**
KNN3	0.8165	**0.849**
KNN5	0.8292	**0.8553**
SVM	0.8548	**0.8727**

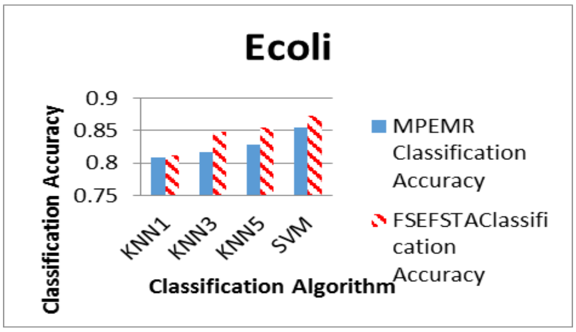


Figure 2. Classification Accuracy Comparison with Ecoli Data set

The results in the Table VI give the comparison of existing and the proposed methods for finding the classification accuracy in order to maintain the orthogonality with the reduced dimension d and with two threshold values T. The comparisons are done with two types of classification algorithms, i.e., knn (k=1, 3, 5) and Linear SVM. The KNN classifier as it tries to find the nearest data points based on their classification and the SVM is a learning algorithm that analyses data for classification in supervised learning methods. Knn classifier calculates the k nearest data points and uses these datapoints to determine to which class particular data point belongs. Figure 2 shows the comparison with classification accuracy with KNN and SVM methods which are having 80 to 85 percent with MPEMR methods, whereas the proposed method is having 81 to 87 percent of accuracy with classification in order to maintain the orthogonality. The T_{11}, T_{12}, T_{21}, T_{22}, T_{31}, T_{32} where T_{i1}, T_{i2} are the threshold values for the i^{th} method. For each data set, the reduced dimensionality (d) is also provided. In all the cases the FSEFST algorithm gives more accuracy as the threshold is changing in every iteration in order to get the best pair of features.

TABLE VII. Ecoli Dataset for Redundancy checking

EColi (d=5),T[(0.1, 0.7), (0.15, 0.85)]		
Algorithm	**Redundancy Accuracy**	
	MPEMR	**FSEFST**
KNN1	0.76688	0.7668
KNN3	0.78816	0.7881
KNN5	0.81429	0.8142
SVM	0.8244	0.8244

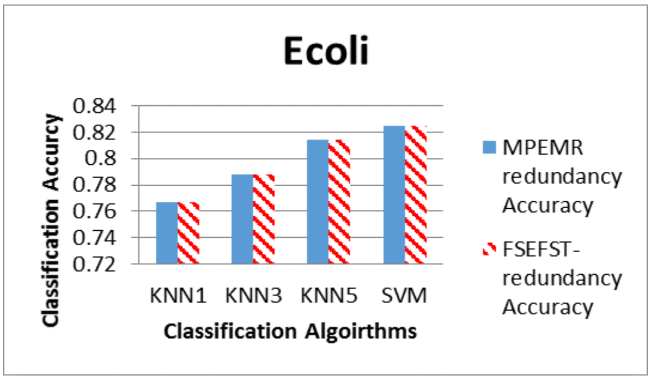


Figure 3. Redundancy Checking Comparison with Ecoli Data set

In Table VII the Ecoli dataset is considered in checking the redundancy by considering the reduced dimensionality d and two thresholds T. The knn is the first choice for the classification study when there is little or no prior knowledge about the distribution of data. Figure 3 which shows the comparison of the proposed algorithm is having the same accuracy as that of the existing methods because for each dataset there is a combination of T_1 and T_2 which controls the level of redundancy in this case the redundancy checking remains same in some cases the smaller value of T_1, the reduced set contains no semi features to extract and check for the redundancy.

TABLE VIII. Pageblocks Dataset for Classification Accuracy.

Pageblocks (d=6), T[(0.08, 0.95), (0.1, 0.75)]		
Algorithm	Classification Accuracy	
	MPEMR	FSEFST
KNN1	0.9632	**0.964**
KNN3	0.9674	**0.9685**
KNN5	0.966	0.9665
SVM	**0.953**	0.9523

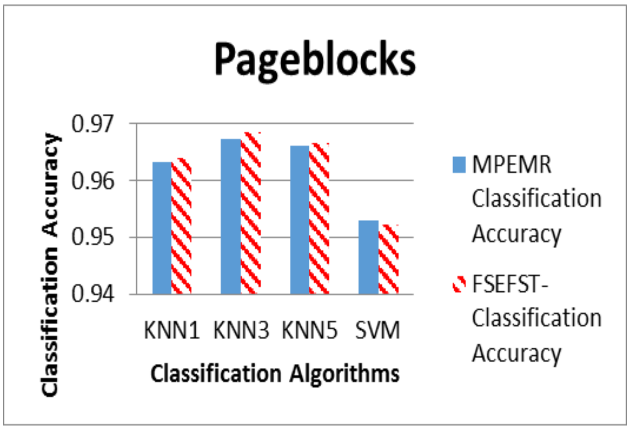


Figure 4. Classification Accuracy Comparison with Pageblocks Data set

The Table VIII and Figure 4 give the comparison done by considering the page blocks dataset in order to compare the accuracy for error checking. The MPEMR method gives the 96.3 percent of accuracy for KNN1 to 96.6 percent for KNN3 whereas the proposed system is having accuracy with KNN1 and KNN3 is 96.4 percent to 96.6 percent, a small improvement in the error classification accuracy, whereas with the SVM classification method the accuracy is same as the the existing method as compare to other classifiers SVM scales relatively well to high dimensional data. But chooseing appropriate kernel function is the challenging in case of SVM.

TABLE IX. Pageblocks Dataset for Redundacy checking

Pageblocks (d=6), T[(0.08, 0.95), (0.1, 0.75)]		
Algorithm	Redundancy Accuracy	
	MPEMR	FSEFST
KNN1	0.9179	0.9645
KNN3	0.9198	**0.9685**
KNN5	0.9102	**0.9671**
SVM	0.8991	0.9514

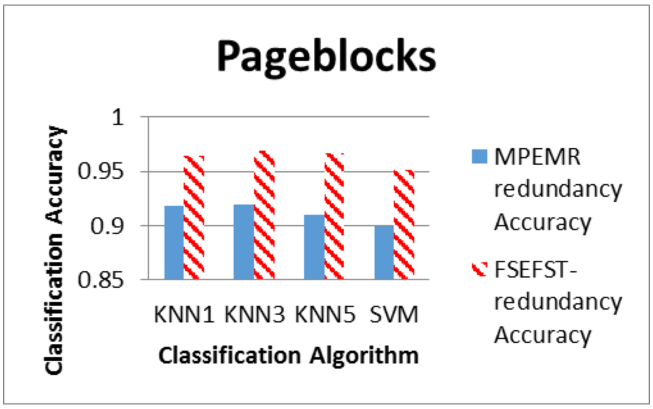


Figure 5. Redundancy Accuracy Comparison with Pageblocks Data set

The Table IX explains about the comparison done by considering the pageblocks dataset for checking the redundancy. In case of SVM once the boundary is established, most of the data is redundant. All it needs is a core set of points which can help to identify and set the boundary. Knn algorithm is based on feature similarity so it is more efficient to find the redundancy in the given dataset. Figure 5 shows that in all cases the proposed method as done much better redundancy checking than the existing method as the existing method is having 89 to 91 percent for almost for all the methods whereas the proposed system is having 96 percent for all the methods with redundancy checking. In MPEMR method the features and removal of redundancy is based on two features like (2,2), whereas in FSEFST method the features are like combination of previous and the next pair of sets (2,3). So FSEFST method is having better performance compared to the existing method.

TABLE X. Abalone Dataset for Classification Accuracy

Abalone (d=2),T[(0.01, 0.95), (0.001, 0.5)]		
Algorithm	Classification Accuracy	
	MPEMR	FSEFST
KNN1	**0.4819**	0.4462
KNN3	**0.4967**	0.4759
KNN5	**0.519**	0.4852
SVM	0.5353	0.5252

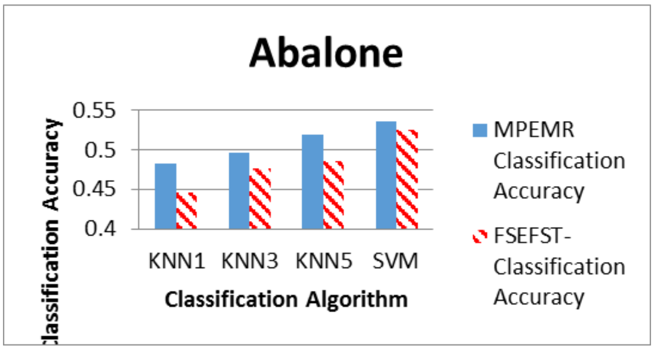


Figure 6. Classification Accuracy Comparison with Abalone Data set

In Table X the Abalone dataset is taken for c classification, comparison, Figure 6 shows that the existing method is more efficient than the proposed system for classification accuracy as the knn prediction stage is slow and also sensitive to irrelevant features and also with the increasing size of the data set.

TABLE XI. Abalone Dataset for Redundancy checking.

Abalone (d=2),T[(0.01, 0.95), (0.001, 0.5)]		
Algorithm	**Redundancy Accuracy**	
	MPEMR	**FSEFST**
KNN1	0.4632	0.4711
KNN3	0.4856	0.4932
KNN5	0.4938	0.5073
SVM	0.5353	0.541

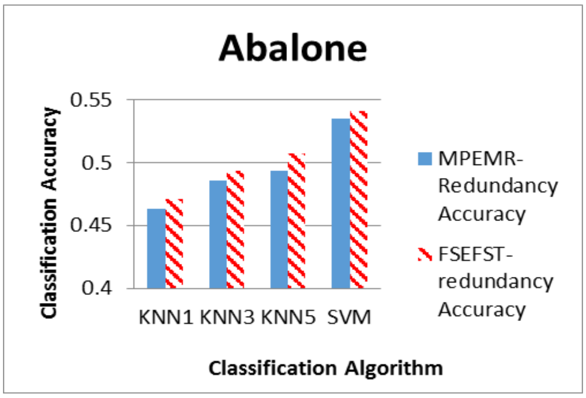


Figure 7. Redundancy Accuracy Comparison with Abalone Data set

The Table XI and Figure 7 show that the with the proposed method for the redundancy check is having 1 percent of improvement for all the methods. Where as in FSEFST method the features are like combination of previous and the next pair of sets (2,3). So FSEFST method is having better performance compared to the existing method.

VI. CONCLUSION

The Dimensionality reduction can be achieved by getting the reduced set from the combination of both original and the reduced set of the features. In order to get the reduced set a framework is proposed which generates the compound features by having minimum projection error and minimum redundancy. As the proposed approach gives the reduced set with the combination of both original and features in the condensed set, the results are compared with existing feature selection and extraction methods. The proposed methods is showing the improvement in finding the projection error and redundancy check.

REFERENCES

1. H. Liu and H. Motoda, "Computational Methods of Feature Selection", Boca Raton, FL, USA, CRC Press, 2007.
2. R O Duda, P E Hart and D G Stork, "Pattern Classification", Hoboken, NJ, USA, Wiley, 1999.
3. M. Robonik, Sikonja and I. Kononenka, "Theroretical and Empirical analysis of relief and rrelieff", Machine Learning, vol. 53, no. ½, pp. 23-69, 2003.
4. F Nie, S. Xiang, Y. Jia, C. Zhang and S. Yan, "Trace Ratio Criterion for Feature Selection", In Proceedings of 23rd National Conference in Artificial Intelligence, 2008, vol. 2, pp. 671-678.
5. Sreevani and C A. Murthy, "Bridging Feature Selection and Extraction: Compound Feature Generation", IEEE Transaction on Knowledge and Data Engineering, vol. 29, No. 4, April 2017.
6. Smita Chormunge and Sudarson Jena, "Efficient Feature Subset Selection algorithm for High Dimensional Data", International Journal of Electrical and Computer Engineering (IJECE), vol. 6, No. 4
7. Abeer Alzubaidi and Georgina Cosma, "Efficient Feature Selection Algorithm for High Dimensional Data", International Journal of Electrical and Computer Engineering (IJECE), Vol. 6, No. 4, August 2016, pp. 1880-1888.
8. D. Asir Antony Gnana Singh, S. Appavu and E. Jebamalar Leavline, "Literature Review on Feature Selection Methods for High-Dimensional Data", International Journal of Computer Applications, vol. 136, No.1, February 2016.
9. D. Sheela Jeyarani and A. Pethalakshmi, "Optimized Feature Selection Algorithm for High Dimensional Data", Indian Journal of Science and Technology, vol. 9, August 2016.
10. Bharat Singh, Nidhi Kushwaha and Om Prakash Vyas, " A Feature Subset Selection Technique for High Dimensional Data using Symmetric Uncertainty", Journal of Data Analysis and Information Processing, pp. 95-105, 2014.
11. Hoang Vu Nguyen and Vivekanand Gopalkrishnan, "Feature Extraction for Outlier Detection in High Dimensional Spaces, In Proceedings of Fourth Workshop on Feature Selection in Data Mining JMLR : Workshop and Conference vol. 10, pp. 66-75.
12. Yue WU, Steven C H Hoi, "Large Scale Online Feature Selection for Ultrahigh Dimensional Sparse Data", ACM Transactions Knowledge Discovery from Data, 2017.
13. Agnieszka Wosiak, Agata Dziomdziora, "Feature Selection and Classification Pairwise Combinations for High-Dimensional Tumour Biomedical Datasets", Schedae Informaticae, vol. 24, pp. 53-62, 2015.
14. Zhao Zhongwen, GUO Hunanghuang, "Visualization Study of High-dimensional Data Classification Based on PCA-SVM", In proceedings of IEEE Second International Conference on Data Science in Cyberpace, 2017.
15. Manikandan G, Susi E and Abirami S, "Feature Selection on High Dimensional Data using Wrapper Based Subset Selection", In Proceedings of Second International Conference on Recent Trends and Challenges in Computational Models, pp. 320-325, 2017.
16. Ding, Hanuchuan Peng and Fuhui Long, "Feature Selection based on Mutual Information Criteria Max-Dependency, Max-Relevance and Min-Redundency", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 27, 2005.
17. W. Sheng, x. Liu and M. Fairhurst, "A Niching Memetic Algorithm for Simultaneous Clustering and Feature Selection", IEEE Transactions on Knowledge and data Engineering, vol. 20, pp. 868-879, Jul 2008.
18. D. Cai, C. Zhang and X. He, "Unsupervised Feature Selection for Multi-Cluster Data", In Proceedings of 16th ACM SIGKDD International Conference Knowledge Discovery Data Mining, pp. 333-342, 2005.
19. Z. Xu, L. King, M R-T Lya and R Jin, "Discriminative Semisupervised Feature Selection via Manifold Regularization", IEEE Transaction Neural Networks, vol. 21, No. 7, pp. 1033-1047, Jul 2010.
20. K. Benabdeslem and M. Hindawi, "Efficient Semi-Supervised Feature Selection Constraint, Relevance and Redundancy", IEEE Transaction Knowledge Data Engineering, vol. 26, No. 5, pp. 1131-1143, May 2014.
21. P. Mitra, C Murthy and S K Pal, "Unsupervised feature selection using feature similarity", IEEE Transaction pattern analysis Machine Intelligence, vol. 24, no. 3, pp. 301-312, March 2002.
22. Dash M, K Choi, P Scheuermann and H. Liu, "Feature Selection for Clustering a Filter Solution", In Proceedings of 2nd International Conference Data Mining, pp. 115-122, 2002.
23. Song Q, J Ni and G Wang, " A Fast Clustering based Feature Subset Selection Algorithm for High Dimensional Data", IEEE Transaction on Knowledge and Data Engineering, vol. 25, pp. 1-14, No. 1, Jan 2013.
24. R Kohavi and G H John, "Wrappers for Feature Subset Selection", Artificial Intelligence, vol. 97, No. 1, pp. 273-324, 1997.
25. J Liu, S Ji and I Ye, "Multi-task Features Learning via Efficient L2, L-noorm Minimization", In Proceedings of 20th International Conference Machine Learning, vol. 3, pp. 856-863, 2003.

AUTHORS PROFILE

Radhika K R has completed Bachelor of Engineering in Computer Science and Engineering from Bangalore University. M.Tech in Computer Science and Engineering from Visvesvaraya Technological University. She has 17 Years of teaching Experience. Presently she is working as Assistant Professor in Department of CSE, BMSIT&M and pursuing her PhD in Computer Science and Engineering, UVCE, Bangalore University under the Guidance of Dr. Thriveni J. She has published 6 papers in International conferences and International Journal.

Dr. Pushpa C N has completed Bachelor of Engineering in Computer Science and Engineering from Bangalore University, Master of Technology in VLSI Design and Embedded Systems from Visvesvaraya Technological University and PhD in Computer Science and Engineering from Bangalore University. Presently she is working as Assistant Professor in the Department of Computer Science and Engineering at University Visvesvaraya College of Engineering, Bangalore. She has 18 years of teaching experience. She has presented and published 25 research papers in the International Conferences and Journals. Her research interest includes Web mining, Personalized Web Search, Semantic Web and Data Analytics.

Dr.Thriveni J has completed Bachelor of Engineering, Masters of Engineering and Doctoral Degree in Computer Science and Engineering. She has 4 ears of industrial experience and 24 years of teaching experience. Currently she is Professor in the Dept. of CSE, University Visvesvaraya College of Engineering, Bangalore. She has over 100 research papers to her credit. She has produced five doctorate students and guiding 06 Ph.D Students. Her research interests include Networks, Web Mining, IOT, Cloud Computing and Security.

Dr. K. R. Venugopal received the Bachelor of Engineering degree from the University Visvesvaraya College of Engineering (UVCE), the master's degree in computer science and automation from the Indian Institute of Science Bangalore, and the Ph.D. degree in economics from Bangalore University and in computer science from the Indian Institute of Technology Madras. He is currently the Vice Chancellor, Bangalore University, Bangalore. He has served as the Principal with the University Visvesvaraya College of Engineering, Bangalore University, Bengaluru for more than one decade. He has a distinguished academic career and has degrees in Electronics, Economics, Law, Business Finance, Public Relations, Communications, Industrial Relations, Computer Science, and Journalism. He has authored and edited 72 books on Computer Science and Economics, which include Petrodollar and the World Economy, C Aptitude, Mastering C, Microprocessor Programming, Mastering C++, and Digital Circuits and Systems. He has filed 101 patents. During his three decades of service at UVCE he has over 800 research papers to his credit. His research interests include computer networks, wireless sensor networks, parallel and distributed systems, digital signal processing, and data mining. He is a Fellow of IEEE and a ACM Distinguished Educator.