Increasing Efficiency of Microarray Analysis by PCA and Machine Learning Methods

**Jing Sun**1, **Kalpdrum Passi**1, **Chakresh Jain**2

1Department of Mathematics and Computer Science, Laurentian University, Sudbury, Ontario, Canada

2Department of Biotechnology, Jaypee Institute of Information Technology, Noida, India

Abstract - *Principal Component Analysis (PCA) is widely used method for dimensionality reduction. However, it has not been studied much as a feature selection method to increase the efficiency of the classifiers on microarray data analysis. In this study, we assessed the performance of four classifiers on the microarray datasets of colon and leukemia cancer before and after applying PCA as a feature selection method. Different thresholds were used with 10-fold cross validation. Significant improvement was observed in the performance of the well-known machine learning classifiers on microarray datasets of colon and leukemia after applying PCA*.

Keywords - Principal component analysis; Support Vector Machine; Random Forest; Neural Network; K-Nearest-Neighbor; Feature selection.

# **Introduction**

The gene expression profiling techniques by DNA microarrays provide the analysis of large amount of genes [1]. The amount of gene expression data of microarray has grown exponentially. It is of great importance to find the key gene expression which can best describe the phenotypic trait [2]. The microarray dataset usually has a large number of genes in small number of experiments which collectively raise the issue of “curse of dimensionality” [17]. To find the key gene expression, one way is to use feature selection methods. In this paper we use Principal Component Analysis (PCA) for feature selection and apply four well-known machine learning methods, Support Vector Machine (SVM), Neural Network (NN), K-Nearest-Neighbor (KNN) and Random Forest algorithms to validate and compare the performance of Principal Component Analysis. In the first set of experiments presented in this paper, the performance of the four machine learning techniques (SVM, NN, KNN, Random Forest) is compared on the colon and leukemia microarray datasets. The second set of experiments compares the performance of these machine learning algorithms by applying PCA method on the same datasets.

The main contribution of this research is to show that PCA used as a feature selection method can improve the performance of the four well known machine learning algorithms on microarray datasets.

The paper is organized as follows. We first state our experiment environment in Section II. Secondly, literature review is given in Section III. Methodology is explained in Section IV; results and discussion are given in Section V. Section VI presents the conclusions of this study and states future prospects and limitation of this work.

# **Dataset and Tools**

## **Dataset**

The datasets used in this comparison study are the “Colon cancer dataset” and the “Leukemia cancer dataset”. They can be accessed from various sources. The colon dataset describes a colon cancer study [3] in which gene expression levels were measured on 40 normal tissues and 22 tumor tissues. The leukemia dataset consists of 38 bone marrow samples obtained from acute leukemia patients and 34 normal samples [4]. There are 2000 attributes corresponding to 2000 different genes for each tissue. The leukemia dataset has 7129 attributes corresponding to 7129 different genes for each tissue. The datasets were already normalized to mean zero and variance at the time of downloading.

## **Tools**

In our experiment, we use Weka as the main testing tools which is a collection of machine learning algorithms for data mining tasks. All the results in this paper have been obtained from Weka 3-7-13-oracle-jvm on Mac OS 10.11.1. In Weka and R, we can access implementation of PCA from its library. Some of the plots and tables have been obtained from R. The version of R used in the experiments is 3.2.2.

# **Literature Review**

Disease classification is the primary issue of microarray research. We can see from [5] that most of the previous analysis and reporting focused on outcome-related gene finding, class discovery and supervised prediction. Most of the studies focus on Hematologic malignancies, Lung and pleura [6][7], Breast [8][9], Hepato-digestive system, etc. As far as the authors are aware, few studies have been carried out that investigate the effect of using PCA with a number of machine learning algorithms.

Tom Howley [10] states the usefulness of PCA for reducing dimensionality and improving the performance of a variety of machine learning methods. Previous work mostly focuses on some specific machine learning algorithms [11][12].

PCA is a classical statistical method for transforming attributes of a dataset into a new dataset of uncorrelated attributes called principal components. PCA can be used as a dimensionality reduction method. The goal of this research is to determine if PCA can be used to improve the performance of machine learning algorithms in the classification of colon and leukemia datasets.

PCA is an exploratory multivariate statistical technique for simplifying complex data sets [13]. Given m observations on n variables, the goal of PCA is to reduce the dimensionality of the data matrix by finding r new variables, where r < n. Termed principal components, these r new variables together account for as much of the variance in the original n variables as possible while remaining mutually uncorrelated and orthogonal. Each principal component is a linear combination of the original variables, and so it is often possible to ascribe meaning to what the components represent. Principal Components Analysis has been used in a wide range of biomedical problems, including the analysis of microarray data in search of outlier genes [14] as well as the analysis of other types of expression data [15].

# **Methodology**

There are three parts in our experiments: feature selection by PCA, cross-validation comparison and ratio comparison. We apply all the three parts on the colon and leukemia datasets.

## **4.1 Feature selection**

Principal Component Analysis (PCA) is a multivariate technique that analyzes a data table in which observations are described by several inter-correlated quantitative dependent variables. Its goal is to extract the important information from the table, to represent it as a set of new orthogonal variables called principal components, and to display the pattern of similarity of the observations and of the variables as points in maps. The quality of the PCA model can be evaluated using cross-validation techniques. Mathematically, PCA depends upon the eigen-decomposition of positive semi-definite matrices and upon the singular value decomposition (SVD) of rectangular matrices [16]. The PCA viewpoint requires that one compute the eigenvalues and eigenvectors of the covariance matrix, which is the product , where is the data matrix. Since the covariance matrix is symmetric, the matrix is diagonalizable, and the eigenvectors can be normalized such that they are orthonormal:

(1)

On the other hand, applying SVD to the data matrix X as follows:

(2)

and attempting to construct the covariance matrix from this decomposition gives

(3)

(4)

and since V is an orthogonal matrix(,

(5)

and the correspondence is easily seen.

For each experiment, we need the original dataset and the new dataset obtained by applying the PCA. Proportion of variance is an important value in PCA which gives the main idea of how much variance this new attribute covered. Our selection uses this value to be the threshold and we choose different thresholds for selecting new subsets of data from the original one. We then obtain different datasets with threshold values of 95%, 90%, …, 50%.

## **4.2 Cross-Validation in Principle Component Analysis**

10-fold cross validation was applied for each classifier on all the datasets. The performance was compared for correctly classified instances and area under the ROC (Receiver Operating Characteristic) curve (AUC).

## **4.3 Ratio Comparison**

The process of Ratio Comparison in PCA is that we split the dataset in different ratios of training set and test set. The four classifiers are applied on the datasets and use the test set to validate the model. The results are compared by correctly classified instances and AUC.

# **Results and Discussion**

## **5.1 Principal Component Analysis Dataset List**

We applied PCA on the colon and leukemia datasets. The variance table returned by PCA is listed in Table 1 and Table 2.

**Table 1. Colon Dataset** **Thresholds and Attribute Selection**

|  |  |  |
| --- | --- | --- |
| Colon Dataset Thresholds | Cumulative Proportion | Attributes Selected |
| 100%(Raw) | 100% | 2001 |
| 95% | 95.013% | 45 |
| 90% | 90.520% | 35 |
| 85% | 85.677% | 27 |
| 80% | 80.006% | 20 |
| 75% | 75.545% | 16 |
| 70% | 71.429% | 13 |
| 65% | 66.004% | 10 |
| 60% | 61.154% | 8 |
| 55% | 57.701% | 7 |
| 50% | 53.180% | 6 |

The experiment is based on the 11 datasets shown in Table 1 and Table 2. The 100% dataset threshold means we use the raw data as input for the experiments. The 95% to 50% datasets are chosen by PCA method.

**Table 2. Leukemia Dataset Thresholds and Attributes Selection**

|  |  |  |
| --- | --- | --- |
| Dataset Thresholds | Cumulative Proportion | Attributes Selected |
| 100%(Raw) | 100% | 7130 |
| 95% | 95.192% | 59 |
| 90% | 90.244% | 49 |
| 85% | 85.560% | 41 |
| 80% | 80.232% | 33 |
| 75% | 75.638% | 27 |
| 70% | 70.261% | 21 |
| 65% | 65.997% | 17 |
| 60% | 60.570% | 13 |
| 55% | 55.557% | 10 |
| 50% | 51.440% | 8 |

## **5.2 10-folds Cross Validation Results**

The results are listed in Table 3. The accuracy (correctly classified instances) is given by:

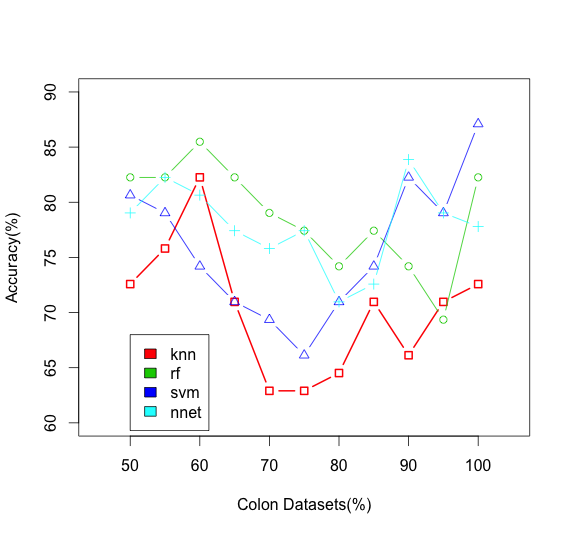
(6)

where TP indicates the True Positive instances, TN indicates the True Negative instances and N is the total number of instances in the test set.

Table 3 shows the accuracy and Area Under ROC curve (AUC) for the four classifiers on the colon dataset. K-nearest-neighbor and Random Forest algorithms shows the highest accuracy for a threshold of 60% by PCA. SVM algorithm shows the highest accuracy for the raw data and for 90% threshold by PCA. Neural Network algorithm shows the highest accuracy for a threshold of 90% by PCA. All the four classifiers show improvement in accuracy for some threshold value of PCA as compared to raw data except for SVM. Figure 1 shows the results of the 10-folds cross validation for colon dataset.

**Table 3. 10-Folds Cross Validation For Colon Dataset**

|  |  |  |  |
| --- | --- | --- | --- |
| Data Mining Methods | Dataset | Accuracy | ROC Area(auc) |
| KNN | Raw | 72.5806% | 0.699 |
| 95% | 70.9677% | 0.680 |
| 90% | 66.129% | 0.648 |
| 85% | 70.9677% | 0.728 |
| 80% | 64.5161% | 0.619 |
| 75% | 62.9032% | 0.581 |
| 70% | 62.9032% | 0.592 |
| 65% | 70.9677% | 0.706 |
| 60% | 82.2581% | 0.815 |
| 55% | 75.8065% | 0.752 |
| 50% | 72.5806% | 0.744 |
| Random Forest | Raw | 82.2581% | 0.885 |
| 95% | 69.3548% | 0.879 |
| 90% | 74.1935% | 0.877 |
| 85% | 77.4194% | 0.840 |
| 80% | 74.1935% | 0.812 |
| 75% | 77.4194% | 0.845 |
| 70% | 79.0323% | 0.855 |
| 65% | 82.2581% | 0.873 |
| 60% | 85.4839% | 0.892 |
| 55% | 82.2581% | 0.881 |
| 50% | 82.2581% | 0.872 |
| SVM | Raw | 87.0968% | 0.886 |
| 95% | 79.0323% | 0.868 |
| 90% | 82.2581% | 0.893 |
| 85% | 74.1935% | 0.805 |
| 80% | 70.9677% | 0.797 |
| 75% | 66.129% | 0.759 |
| 70% | 69.3548% | 0.723 |
| 65% | 70.9677% | 0.830 |
| 60% | 74.1935% | 0.903 |
| 55% | 79.0323% | 0.869 |
| 50% | 80.6452% | 0.881 |
| Neural Network | Raw | 77.8% | 0.857 |
| 95% | 79.0323% | 0.851 |
| 90% | 83.871% | 0.895 |
| 85% | 72.5806% | 0.819 |
| 80% | 70.9677% | 0.777 |
| 75% | 77.4194% | 0.845 |
| 70% | 75.8065% | 0.786 |
| 65% | 77.4194% | 0.805 |
| 60% | 80.6452% | 0.843 |
| 55% | 82.2581% | 0.834 |
| 50% | 79.0323% | 0.826 |



***Figure 1. 10-fold cross validation results for colon dataset***

Table 4 shows the accuracy and Area Under ROC curve (AUC) for the four classifiers on the leukemia dataset. K-nearest-neighbor shows the highest accuracy for a threshold of 70% by PCA. Random Forest shows the highest accuracy for a threshold of 65% and 70% by PCA. SVM shows the highest accuracy for the raw data and next highest accuracy for a threshold of 95% by PCA. Neural Network shows the highest accuracy for a threshold of 60% and 50% by PCA. Figure 2 shows the results of the 10-folds cross validation for leukemia dataset.

## **5.3 Ratio Validation Results**

The second method we use for this experiment is that we split the dataset to a training set and test set by different ratio in 90%:10%,80%:20%,70%:30% and 60%:40%. All the result applied to the data which preprocessed by PCA. We show the results in Table 5 and Table 6.

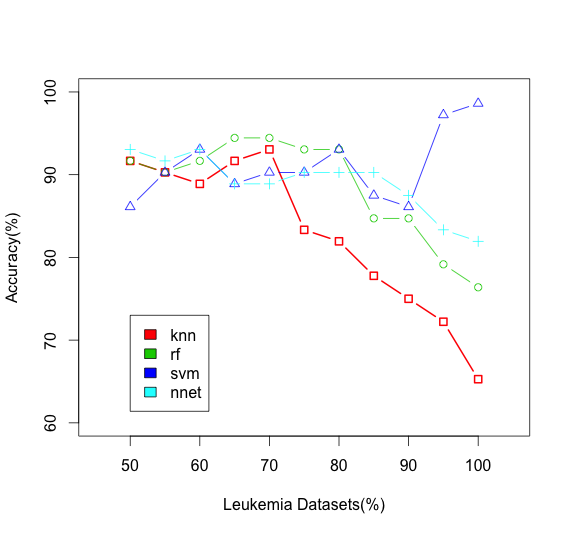
Figures 3 to 6 show the accuracy for the four algorithms for different ratios of training and test datasets. Further discussion is given below.

**Table 4. 10-Folds Cross Validation For Leukemia Dataset**

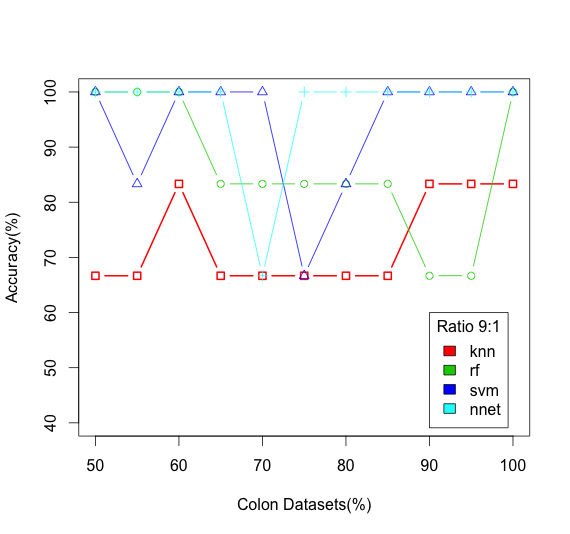
|  |  |  |  |
| --- | --- | --- | --- |
| Data Mining Methods | Dataset | accuracy | ROC Area (auc) |
| KNN | Raw | 65.2778% | 0.505 |
| 95% | 72.2222% | 0.656 |
| 90% | 75% | 0.667 |
| 85% | 77.7778% | 0.719 |
| 80% | 81.9444% | 0.811 |
| 75% | 83.3333% | 0.829 |
| 70% | 93.0556% | 0.911 |
| 65% | 91.6667% | 0.887 |
| 60% | 88.8889% | 0.861 |
| 55% | 90.2778% | 0.874 |
| 50% | 91.6667% | 0.874 |
| Random Forest | Raw | 76.3889% | 0.889 |
| 95% | 79.1667% | 0.918 |
| 90% | 84.7222% | 0.945 |
| 85% | 84.7222% | 0.952 |
| 80% | 93.0556% | 0.963 |
| 75% | 93.0556% | 0.958 |
| 70% | 94.4444% | 0.978 |
| 65% | 94.4444% | 0.974 |
| 60% | 91.6667% | 0.963 |
| 55% | 90.2778% | 0.968 |
| 50% | 91.6667% | 0.969 |
| SVM | Raw | 98.6111% | 0.998 |
| 95% | 97.2222% | 0.995 |
| 90% | 86.1111% | 0.969 |
| 85% | 87.5% | 0.959 |
| 80% | 93.0556% | 0.968 |
| 75% | 90.2778% | 0.977 |
| 70% | 90.2778% | 0.974 |
| 65% | 88.8889% | 0.963 |
| 60% | 93.0556% | 0.969 |
| 55% | 90.2778% | 0.933 |
| 50% | 86.1111% | 0.962 |
| Neural Network | Raw | 81.9444% | 0.865 |
| 95% | 83.3333% | 0.877 |
| 90% | 87.5% | 0.917 |
| 85% | 90.2778% | 0.934 |
| 80% | 90.2778% | 0.970 |
| 75% | 90.2778% | 0.977 |
| 70% | 88.8889% | 0.980 |
| 65% | 88.8889% | 0.971 |
| 60% | 93.0556% | 0.971 |
| 55% | 91.6667% | 0.951 |
| 50% | 93.0556% | 0.974 |

## **5.4 Discussion**

In Table 3 for the colon dataset, we observe that K-nearest-neighbor algorithm gives the highest accuracy of 82.3% for a threshold of 60% by PCA as compared to 72.6% for the raw data, an increase of 9.7% in accuracy by applying PCA. Random Forest algorithm gives the highest accuracy of 85.5% for a threshold of 60% by PCA as compared to 82.3% for the raw data, an increase of 3.3% in accuracy. Neural Network algorithm gives the highest accuracy of 83.9% for a threshold of 90% by PCA as compared to 77.8% for the raw data, an increase of 6.1% in accuracy. However, in SVM, the highest accuracy was observed as 87% for the raw data and 82% accuracy for a threshold of 90% by PCA, a decrease of 5% in accuracy.



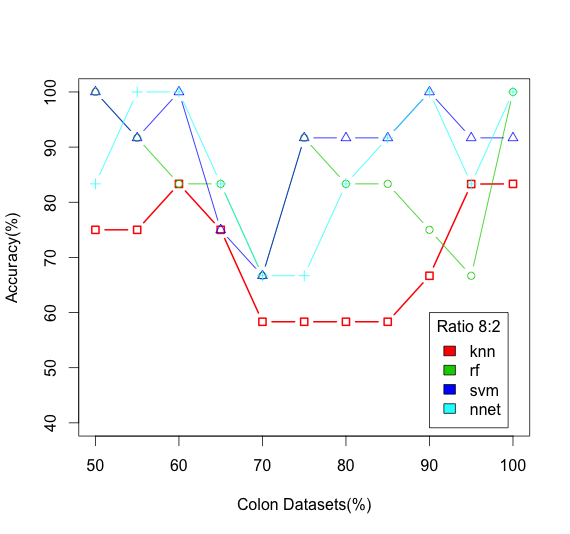
***Figure 2. 10-fold cross validation results for leukemia dataset***



***Figure 3. Accuracy comparison in Ratio 9:1***

In Table 4 for the leukemia dataset, we observe that K-nearest-neighbor algorithm gives the highest accuracy of 93% for a threshold of 70% by PCA as compared to 65% for the raw data, an increase of 28% in accuracy by applying PCA. Random Forest algorithm gives the highest accuracy of 94.4% for a threshold of 65% and 70% by PCA as compared to 76.4% for the raw data, an increase of 18% in accuracy. Neural Network algorithm gives the highest accuracy of 93% for a threshold of 50% and 60% by PCA as compared to 81.9% for the raw data, an increase of 11% in accuracy. However, in SVM, the highest accuracy was observed as 98.6% for the raw data as compared to 97.2% for a threshold of 95% by PCA, a decrease of 3.6% in accuracy.

SVM was tested for four different kernels – linear, polynomial, radial basis function and sigmoid function. The linear kernel gave the best results. For the exception of SVM, all other algorithms increased the accuracy of classification by applying PCA. Greater increase in accuracy was observed in leukemia dataset than the colon dataset. Figures 1 and 2 show the results for 10-fold cross validation for the colon and leukemia datasets, respectively.



***Figure 4. Accuracy comparison in ratio 8:2***

Tables 5 and 6 show the accuracy and AUC of the colon and leukemia datasets respectively for raw data and different thresholds of PCA and by taking different ratios of training and test data.

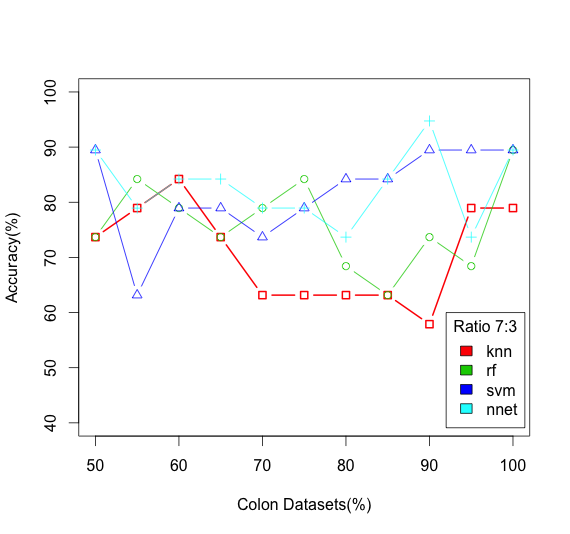
In the colon dataset, we observe that highest accuracy is achieved for the training to test ratio of 9:1 and second highest accuracy for the ratio 8:2 for all the algorithms with and without using PCA.

In training to test ratio 9:1, Random Forest, SVM and Neural Network algorithms give an accuracy of 100% whereas k-nearest-neighbor gives an accuracy of 83.3% for the raw data. PCA maintains the accuracy of 100% at the threshold of 50% and 60% for Random Forest, SVM and Neural Network and maintains the accuracy of 83.3% at the threshold of 60%, 90% and 95% for the k-nearest-neighbor algorithm. Figure 3 shows the comparison of accuracy for the four algorithms for the ratio 9:1.

In training to test ratio 8:2, Random Forest and Neural Network algorithms give an accuracy of 100%, SVM has an accuracy of 91.6% and KNN has an accuracy of 83.3% for the raw data. PCA maintains the accuracy of 100% at thresholds of 55% and 60% for Neural Network and at the threshold of 50% for Random Forest. PCA increased the accuracy of SVM from 91.6% to 100% at the thresholds of 50%, 60% and 90%. PCA maintains the accuracy of KNN at 83.3% at a threshold of 60%. Figure 4 shows the comparison of accuracy for the four algorithms for the ratio 8:2.

In training to test ratio 7:3, PCA increases the accuracy of KNN from 78.95% to 84.21% at a threshold of 60% and increases the accuracy of Neural Network from 89.5% to 94.7% at a threshold of 90%. PCA maintains the accuracy of SVM at 89.5% at the thresholds of 50%, 90% and 95%. However, the accuracy of Random Forest is decreased from 89.5% to 84.2% at thresholds of 55% and 80%. Figure 5 shows the comparison of accuracy for the four algorithms for the ratio 7:3.

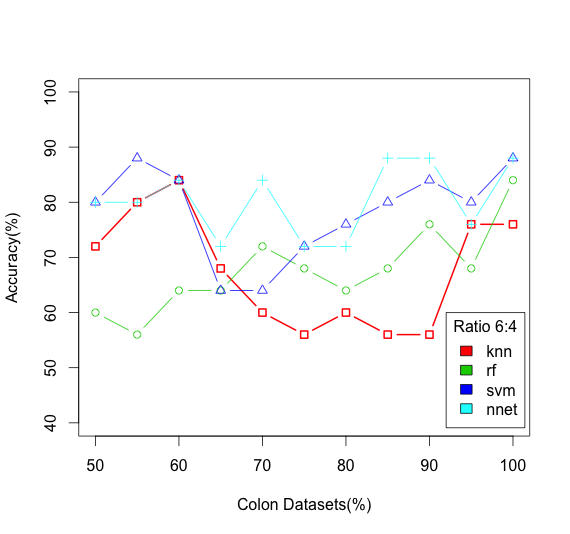
In training to test ratio of 6:4, PCA increases the accuracy of KNN from 76% to 84% at a threshold of 60%. PCA maintains the accuracy of SVM and Neural Network at 88% at a threshold of 55% for SVM and at 85% and 90% for Neural Network. However, the accuracy of Random Forest decreased from 88% to 76% at a threshold of 95%. Figure 6 shows the comparison of accuracy for the four algorithms for the ratio 6:4.



***Figure 5. Accuracy comparison in Ratio 7:3***

Overall, PCA either maintains the accuracy of all the four algorithms or increases the accuracy except for Random Forest at ratios of 7:3 and 6:4.

For the leukemia dataset experiment, we observe from Table 6 that the highest accuracy is achieved for the training to test ratio of 9:1.



**Figure 6. Accuracy comparison in Ratio 6:4**

In training to test ratio of 9:1, PCA increased the accuracy of KNN from 28.6% to 100 at the thresholds of 50%, 55% and 70%, an increase of 71.4% in accuracy. PCA increased the accuracy of Random Forest from 14.3% to 100% at threshold of 50%, 55%, 60%, 65%, 70%, 75%, an increase of 85.7% in accuracy. PCA increased the accuracy of Neural Network from 71.4% to 100% at threshold of 50%, 55% and 70%, an increase of 28.6% in accuracy. PCA maintains the accuracy of SVM at 100% at a threshold of 50%, 60%, 65%, 70%, 85%, 90% and 95%.

In training to test ratio of 8:2, PCA increased the accuracy of KNN from 35.7% to 100% at a threshold of 70%, an increase of 64.3% in accuracy. PCA increased the accuracy of Random Forest from 28.6% to 92.9% at threshold of 50%, 55%, 60% and 65%, an increase of 64.3% in accuracy. PCA increased the accuracy of Neural Network from 42.9% to 100% at thresholds of 50%, 55%, 60%, and 65%, an increase of 57% in accuracy. PCA maintains the accuracy of 100% for SVM at a threshold of 95%.

In training to test ratio of 7:3, PCA increased the accuracy of KNN from 54.5% to 95.5% at threshold of 65% and 70%, an increase of 41%. PCA increased the accuracy of Random Forest from 59% to 95.5% at thresholds of 50%, 55%, 60%, 65%, and 70%, an increase of 36.5% in accuracy. PCA increased the accuracy of Neural Network from 63.6% to 100% at thresholds of 50%, 60%, and 65%, an increase of 36.4% in accuracy. PCA maintains the accuracy of SVM at 100% at a threshold of 70%.

In training to test ratio of 6:4, PCA increased the accuracy of KNN from 58.6% to 93% at a threshold of 70%, an increase of 34.4% in accuracy. PCA increased the accuracy of Random Forest from 55% to 96.5% at thresholds of 50% and 65%, an increase of 41.5% in accuracy. PCA increased the accuracy of Neural Network from 68.9% to 100% at thresholds of 50%, 55%, 60% and 65%, an increase of 31% in accuracy. However, the accuracy of SVM decreased from 100% to 96.5% at thresholds of 50%, 55%, 70% and 95%.

From the two datasets that PCA increases the accuracy of the four classifiers at different thresholds. There are significant improvements in the accuracy for leukemia dataset.

# **Conclusions**

In this paper, we applied the Principle Component Analysis (PCA) on colon dataset and the leukemia dataset and we compared the accuracy for four different classifiers. Support Vector Machine and Neural Network gave the best performance among the four methods. The experiments included 10-fold cross validation and different training to test ratios of 9:1, 8:2, 7:3 and 6:4.

PCA increased the accuracy of the four classifiers for the colon and leukemia datasets. However, it was observed that there were significant improvements in the performance of most of the classifiers with 10-folds cross validation. The improvements were more significant for the leukemia dataset. In the case of different training to test ratios, PCA maintained the accuracy of the classifiers or increased the accuracy for the colon dataset. However, PCA increased the accuracy of the classifiers significantly for the leukemia dataset.

PCA was selected as a feature selection method to test for increase in accuracy of classifiers on test datasets. The results were promising and it gives us further incentive to test the accuracy of the classifiers with other feature selection algorithms in future.

##### **References**

1. A. Richard and Young, “Biomedical Discovery with DNA Arrays,” Cell, vol. 102, pp. 9-15, 2000.
2. Liu, Hsi-Che, Pei-Chen Peng, Tzung-Chien Hsieh, Ting-Chi Yeh, Chih-Jen Lin, Chien-Yu Chen, Jen-Yin Hou, Lee-Yung Shih, and Der-Cherng Liang. "Comparison of Feature Selection Methods for Cross-Laboratory Microarray Analysis." *IEEE/ACM Trans. Comput. Biol. and Bioinf. IEEE/ACM Transactions on Computational Biology and Bioinformatics* 10.3 (2013): 593-604. Web.
3. Alon, U., N. Barkai, D. A. Notterman, K. Gish, S. Ybarra, D. Mack, and A. J. Levine. "Broad Patterns of Gene Expression Revealed by Clustering Analysis of Tumor and Normal Colon Tissues Probed by Oligonucleotide Arrays." *Proceedings of the National Academy of Sciences* 96.12 (1999): 6745-750. Web.
4. Golub, T. R. "Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring." Science 286.5439 (1999): 531-37. Web.
5. Dupuy, A., and R. M. Simon. "Critical Review of Published Microarray Studies for Cancer Outcome and Guidelines on Statistical Analysis and Reporting." *JNCI Journal of the National Cancer Institute* 99.2 (2007): 147-57. Web.
6. Subramanian, J., and R. Simon. "Gene Expression-Based Prognostic Signatures in Lung Cancer: Ready for Clinical Use?" *JNCI Journal of the National Cancer Institute* 102.7 (2010): 464-74. Web.
7. Sun, Z., D. A. Wigle, and P. Yang. "Non-Overlapping and Non-Cell-Type-Specific Gene Expression Signatures Predict Lung Cancer Survival." *Journal of Clinical Oncology* 26.6 (2008): 877-83. Web.
8. Pusztai L, Symmans FW, Hortobagyi GN. Development of pharmacogenomic markers to select preoperative chemotherapy for breast cancer. Breast Cancer2005;12:73–85.
9. Verderio, P. "Assessing the Clinical Relevance of Oncogenic Pathways in Neoadjuvant Breast Cancer." *Journal of Clinical Oncology* 30.16 (2012): 1912-915. Web.
10. Howley, Tom, Michael G. Madden, Marie-Louise O’Connell, and Alan G. Ryder. "The Effect of Principal Component Analysis on Machine Learning Accuracy with High Dimensional Spectral Data." *Applications and Innovations in Intelligent Systems XIII* (2006): 209-22. Web.
11. O'connell, Marie-Louise, Tom Howley, Alan G. Ryder, Marc N. Leger, and Michael G. Madden. "Classification of a Target Analyte in Solid Mixtures Using Principal Component Analysis, Support Vector Machines, and Raman Spectroscopy." Opto-Ireland 2005: Optical Sensing and Spectroscopy (2005): n. pag. Web.
12. Petránek, S. "Neural Network Based Principal Components Analysis for EEG Pre-processing and Analysis." *Electroencephalography and Clinical Neurophysiology* 103.1 (1997): 115. Web.
13. Raychaudhuri, Soumya, Joshua M. Stuart, and Russ B. Altman. "Principal Components Analysis To Summarize Microarray Experiments: Application To Sporulation Time Series." Biocomputing 2000 (1999): n. pag. Web.
14. Hilsenbeck, S. G., W. E. Friedrichs, R. Schiff, P. O'connell, R. K. Hansen, C. K. Osborne, and S. A. W. Fuqua. "Statistical Analysis of Array Expression Data as Applied to the Problem of Tamoxifen Resistance." JNCI Journal of the National Cancer Institute 91.5 (1999): 453-59. Web.
15. Vohradsky, Jiří, Xin-Ming Li, and Charles J. Thompson. "Identification of Procaryotic Developmental Stages by Statistical Analyses of Two-dimensional Gel Patterns." Electrophoresis 18.8 (1997): 1418-428. Web.
16. Abdi, H. and Williams, L. J. (2010), Principal component analysis. WIREs Comp Stat, 2: 433–459. doi: 10.1002/wics.101
17. Antoniadis, A., S. Lambert-Lacroix, S., Leblanc, F., Effective dimension reduction methods for tumor classification using gene expression data Bioinformatics (2003) 19 (5): 563-570

doi:10.1093/bioinformatics/btg062

**Table 5. Ratio Validation Results For Colon Dataset**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | 9:1 | | 8:2 | | 7:3 | | 6:4 | |
| Method | accuracy | auc | accuracy | auc | accuracy | auc | accuracy | auc |
| Raw | Knn | 83.33% | 0.75 | 83.33% | 0.75 | 78.95% | 0.683 | 76% | 0.7 |
| Svm | 100% | 1 | 91.667% | 1 | 89.47% | 0.9 | 88% | 0.93 |
| Rf | 100% | 1 | 100% | 1 | 89.47% | 0.892 | 84% | 0.9 |
| Nnet | 100% | 0.85 | 100% | 0.916 | 89.47% | 0.935 | 88% | 0.893 |
| 95% | Knn | 83.33% | 0.875 | 83.3% | 0.875 | 78.95% | 0.867 | 76% | 0.775 |
| Svm | 100% | 1 | 91.67% | 1 | 89.47% | 0.917 | 80% | 0.91 |
| Rf | 66.7% | 0.875 | 66.67% | 0.797 | 68.42% | 0.725 | 68% | 0.77 |
| Nnet | 100% | 1 | 83.3% | 0.875 | 73.68% | 0.800 | 76% | 0.81 |
| 90% | Knn | 83.33% | 0.875 | 66.67% | 0.688 | 57.89% | 0.642 | 56% | 0.575 |
| Svm | 100% | 1 | 100% | 1 | 89.47% | 0.95 | 84% | 0.92 |
| Rf | 66.67% | 1 | 75% | 0.734 | 73.68% | 0.833 | 76% | 0.845 |
| Nnet | 100% | 1 | 100% | 1 | 94.74% | 0.983 | 88% | 0.91 |
| 85% | Knn | 66.67% | 0.625 | 58.33% | 0.563 | 63.16% | 0.675 | 56% | 0.575 |
| Svm | 100% | 1 | 91.67% | 1 | 84.21% | 0.95 | 80% | 0.910 |
| Rf | 83.33% | 1 | 83.33% | 0.875 | 63.16% | 0.783 | 68% | 0.82 |
| Nnet | 100% | 1 | 91.67% | 1 | 84.21% | 0.967 | 88% | 0.88 |
| 80% | Knn | 66.67% | 0.5 | 58.33% | 0.438 | 63.16% | 0.4 | 60% | 0.45 |
| Svm | 83.33% | 1 | 91.67% | 0.938 | 84.21% | 0.850 | 76% | 0.84 |
| Rf | 83.33% | 1 | 83.33% | 0.969 | 68.42% | 0.817 | 64% | 0.83 |
| Nnet | 100% | 1 | 83.33% | 0.938 | 73.68% | 0.833 | 72% | 0.77 |
| 75% | Knn | 66.67% | 0.5 | 58.33% | 0.438 | 63.16% | 0.4 | 56% | 0.35 |
| Svm | 66.67% | 0.75 | 91.67% | 1 | 78.95% | 0.842 | 72% | 0.87 |
| Rf | 83.33% | 1 | 91.67% | 1 | 84.21% | 0.833 | 68% | 0.8 |
| Nnet | 100% | 1 | 66.67% | 0.906 | 78.95% | 0.817 | 72% | 0.78 |
| 70% | Knn | 66.67% | 0.5 | 58.33% | 0.438 | 63.16% | 0.4 | 60% | 0.375 |
| Svm | 100% | 1 | 66.67% | 0.875 | 73.68% | 0.875 | 64% | 0.835 |
| Rf | 83.33% | 0.875 | 66.67% | 0.906 | 78.95% | 0.85 | 72% | 0.94 |
| Nnet | 66.67% | 1 | 66.67% | 0.906 | 78.95% | 0.833 | 84% | 0.92 |
| 65% | Knn | 66.67% | 0.5 | 75% | 0.625 | 73.68% | 0.558 | 68% | 0.575 |
| Svm | 100% | 1 | 75% | 0.938 | 78.95% | 0.858 | 64% | 0.81 |
| Rf | 83.33% | 1 | 83.33% | 0.938 | 73.68% | 0.933 | 64% | 0.835 |
| Nnet | 100% | 1 | 83.33% | 1 | 84.21% | 0.917 | 72% | 0.92 |
| 60% | Knn | 83.33% | 0.75 | 83.33% | 0.75 | 84.21% | 0.808 | 84% | 0.825 |
| Svm | 100% | 1 | 100% | 1 | 78.95% | 0.933 | 84% | 0.95 |
| Rf | 100% | 1 | 83.33% | 1 | 78.95% | 0.942 | 64% | 0.955 |
| Nnet | 100% | 1 | 100% | 1 | 84.21% | 0.933 | 84% | 0.98 |
| 55% | Knn | 66.67% | 0.625 | 75% | 0.688 | 78.95% | 0.775 | 80% | 0.8 |
| Svm | 83.33% | 1 | 91.67% | 1 | 63.16% | 0.958 | 88% | 0.95 |
| Rf | 100% | 1 | 91.67% | 1 | 84.21% | 0.917 | 56% | 0.96 |
| Nnet | 100% | 1 | 100% | 1 | 78.94% | 0.9 | 80% | 0.94 |
| 50% | Knn | 66.67% | 0.625 | 75% | 0.688 | 73.68% | 0.65 | 72% | 0.675 |
| Svm | 100% | 1 | 100% | 1 | 89.47% | 0.9 | 80% | 0.88 |
| Rf | 100% | 1 | 100% | 1 | 73.68% | 0.908 | 60% | 0.95 |
| Nnet | 100% | 1 | 83.33% | 1 | 89.47% | 0.883 | 80% | 0.92 |

**Table 6. Ratio Validation Results For Leukemia Dataset**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | 9:1 | | 8:2 | | 7:3 | | 6:4 | |
| Method | accuracy | auc | accuracy | auc | accuracy | auc | accuracy | auc |
| Raw | Knn | 28.5714% | 0.583 | 35.7143% | 0.550 | 54.5455% | 0.545 | 58.6207% | 0.546 |
| Svm | 100% | 1 | 100% | 1 | 100% | 1 | 100% | 1 |
| Rf | 14.2857% | 1 | 28.5714% | 1 | 59.0909% | 0.888 | 55.1724% | 0.798 |
| Nnet | 71.4286% | 1 | 42.8571% | 0.9 | 63.6364% | 0.983 | 68.9655% | 0.870 |
| 95% | Knn | 71.4286% | 0.833 | 78.5714% | 0.775 | 81.8182% | 0.818 | 72.4138% | 0.714 |
| Svm | 100% | 1 | 100% | 1 | 95.4545% | 1 | 96.5517% | 1 |
| Rf | 57.1429% | 1 | 35.7143% | 0.975 | 59.0909% | 0.921 | 65.5172% | 0.825 |
| Nnet | 71.4286% | 1 | 42.8571% | 0.825 | 72.7273% | 0.818 | 72.4138% | 0.731 |
| 90% | Knn | 42.8571% | 0.667 | 50% | 0.650 | 54.5455% | 0.545 | 58.6207% | 0.538 |
| Svm | 85.7143% | 1 | 85.7143% | 1 | 90.9091% | 0.975 | 89.6552% | 0.976 |
| Rf | 42.8571% | 1 | 50% | 1 | 59.0909% | 0.893 | 65.5172% | 0.873 |
| Nnet | 71.4286% | 1 | 64.2857% | 0.9 | 68.1818% | 0.851 | 72.4138% | 0.793 |
| 85% | Knn | 42.8571% | 0.667 | 50% | 0.650 | 54.5455% | 0.545 | 62.069% | 0.584 |
| Svm | 71.4286% | 1 | 71.4286% | 0.975 | 86.3636% | 0.975 | 89.6552% | 0.976 |
| Rf | 85.7143% | 1 | 57.1429% | 1 | 63.6364 | 1 | 68.9655% | 0.962 |
| Nnet | 57.1429% | 1 | 71.4286% | 0.925 | 72.7273% | 0.901 | 79.3103% | 0.870 |
| 80% | Knn | 57.1429% | 0.750 | 57.1429% | 0.7 | 68.1818% | 0.682 | 72.4138% | 0.7 |
| Svm | 71.4286% | 1 | 71.4286% | 1 | 90.9091% | 0.942 | 89.6552% | 0.962 |
| Rf | 85.7143% | 1 | 71.4286% | 1 | 81.8182% | 0.992 | 72.4138% | 0.988 |
| Nnet | 57.1429% | 1 | 85.7143% | 1 | 90.9091% | 0.934 | 72.4138% | 0.861 |
| 75% | Knn | 85.7143% | 0.917 | 92.8571% | 0.95 | 81.8182% | 0.818 | 75.8621% | 0.752 |
| Svm | 71.4286% | 1 | 85.7143% | 0.975 | 86.3636% | 0.934 | 89.6552% | 0.942 |
| Rf | 100% | 1 | 64.2857% | 1 | 77.2727% | 0.996 | 79.3103% | 0.981 |
| Nnet | 71.4286% | 1 | 85.7143% | 0.975 | 81.8182% | 0.967 | 79.3103% | 0.875 |
| 70% | Knn | 100% | 1 | 100% | 1 | 95.4545% | 0.955 | 93.1034% | 0.923 |
| Svm | 100% | 1 | 85.7143% | 1 | 100% | 1 | 96.5517% | 1 |
| Rf | 100% | 1 | 78.5714% | 1 | 95.4545% | 1 | 89.6552% | 0.955 |
| Nnet | 100% | 1 | 85.7143% | 1 | 90.9091% | 1 | 89.6552% | 0.976 |
| 65% | Knn | 85.7143% | 0.917 | 92.8571% | 0.95 | 95.4545% | 0.955 | 89.6552% | 0.892 |
| Svm | 85.7143% | 1 | 85.7143% | 1 | 95.4545% | 1 | 96.5517% | 1 |
| Rf | 100% | 1 | 92.8571% | 1 | 95.4545% | 1 | 96.5517% | 1 |
| Nnet | 85.7143% | 1 | 100% | 1 | 100% | 1 | 100% | 1 |
| 60% | Knn | 85.7143% | 0.917 | 85.7143% | 0.9 | 90.9091% | 0.909 | 89.6552% | 0.892 |
| Svm | 100% | 1 | 78.5714% | 1 | 90.9091% | 1 | 93.1034% | 1 |
| Rf | 100% | 1 | 92.8571% | 1 | 95.4545% | 1 | 89.6552% | 1 |
| Nnet | 100% | 1 | 100% | 1 | 100% | 1 | 100% | 1 |
| 55% | Knn | 100% | 1 | 71.4286% | 0.8 | 77.2727% | 0.773 | 82.7586% | 0.815 |
| Svm | 100% | 1 | 92.8571% | 0.975 | 95.4545% | 0.992 | 96.5517% | 0.990 |
| Rf | 100% | 1 | 92.8571% | 1 | 95.4545% | 1 | 93.1034% | 1 |
| Nnet | 100% | 1 | 100% | 1 | 95.4545% | 1 | 100% | 1 |
| 50% | Knn | 100% | 1 | 78.5714% | 0.850 | 81.8182% | 0.818 | 86.2069% | 0.853 |
| Svm | 85.7143% | 1 | 92.8571% | 1 | 95.4545% | 1 | 96.5517% | 1 |
| Rf | 100% | 1 | 92.8571% | 1 | 95.4545% | 0.996 | 96.5517% | 1 |
| Nnet | 100% | 1 | 100% | 1 | 100% | 1 | 100% | 1 |