IJAMML 7:1-2 (2017) 39-55 September and December 2017
ISSN: 2394-2258
Available at http://scientificadvances.co.in
DOI: http://dx.doi.org/10.18642/ijamml_7100121858

NAÏVE BAYES CLASSIFICATION USING KERNEL INDEPENDENT COMPONENT ANALYSIS

Abdullah Bashir Musa and Fuad Abedalrazeq Mussallum

College of Computer Sciences and Information Technology, Department of Computer Sciences, University of Dammam, Dammam, Kingdom of Saudi Arabia

Deanship of Preparatory and Support Studies, Department of Basic Sciences, University of Dammam, Dammam, Kingdom of Saudi Arabia

Abstract

Naïve Bayes is one of the most well-known efficient learning algorithms. It has been used extensively in a number of different areas of data mining and machine learning. It is known to have outstanding classification performance, which competitive with modern methods such as support vector machines (SVM). Naïve Bayes classifier is used for building on independent features. Although the assumption of independence on which naïve Bayes is based is rarely satisfied in practice, these features can be modified to make the independent assumption hold, which result in improving naïve Bayes classifier. A number of current techniques have been used for that modification. Typically these methods find a set of uncorrelated/independent components of the features. Kernel independent component analysis (KICA) is a popular example of such methods. This paper proposes an application of kernel independent component analysis (KICA) with naïve Bayes for classification. Kernel independent component analysis (KICA) is a kernel version of independent component analysis (ICA). Applying KICA yields new reduced independent features space on which the naïve Bayes classifier can be built. For assessing the classification performance of this technique, this method is compared with independent component analysis (ICA) and kernel

*Corresponding author.
E-mail address: abhamad@uod.edu.sa (Abdullah Bashir Musa).

2010 Mathematics Subject Classification: 68-XX.
Submitted by Jianqiang Gao.
Received July 17, 2017; Revised October 5, 2017

principle component analysis (KPCA) as a kernel approach. A number of performance measures have been used in the comparison; accuracy, sensitivity, specificity, and precision, F-score, the area under receiver operating characteristic curve (AUC) and the receiver operating characteristic (ROC) analysis.

Keywords: naïve Bayes, classification, kernel independent component analysis (KICA), independent component analysis (ICA), kernel principle component analysis (KPCA).

---

1. Introduction

Naïve Bayes classifier is well-known and popular algorithm in statistic and machine learning that have been found to perform surprisingly well [7, 8, 9]. Naïve Bayes has been extensively applied in statistic, data mining, machine learning, and pattern recognition areas of research that is due to its simplicity, elegance, interpretability, and robustness. Even though, it is considered as a classic and simple it is always efficient and effective learning method. It categorized as one of the 10 tops algorithms in data mining in terms of usage and classification performance, it has been widely used in areas such as text classification and spam filtering [2, 22], it may be the best possible classifier in any particular application if its conditions satisfied [21, 23, 24].

Classification is a classical issue in machine learning and data mining; for a given set of instances where each instance is belong to specific class. The aim of the classification is to construct a method which allow to assign future instances to a class, given only the vectors of instances, this type of classification is called supervised classification, and many techniques have been developed for that [25, 26]; naive Bayes method is the most famous supervised classification method.

In naïve Bayes classifier, for classifying new examples Bayes' rule is used to select the class that is more likely than any other to have produced the example. The naïve Bayes classifier is the simplest such model as it assumes that all the attributes of the instances are independent of each other in the class; this is known as the naïve Bayes assumption.

Unfortunately, when building naïve Bayes model the independence assumption on which it is based almost never hold in its practical application in most real-world tasks. Naïve Bayes often performs classification very well in the cases that satisfy the independence assumption; the parameters for each attribute can be learned separately because of the independence assumption, this greatly simplifies learning, particularly when the number of attributes is large [3]. Even though Domingos & Pazzani [4] have shown that the violation of the independence assumption does not have serious effects on the performance of naïve Bayes classification, other research such that presented by Zhang [1] which demonstrated that naïve Bayes can perform poorly when its features are not independent, he shows that dependence distribution essentially effected the classification performance of naïve Bayes, i.e., the matters of how the local dependence of a node distributes in each class, evenly or not, and how the local dependencies of all nodes work together, consistently (supporting a certain classification) or not (cancelling each other out), play a decisive role in the classification.

From a statistical point of view, the assumption of class-conditional independence greatly simplifies estimation by its marginalization of class-conditional densities and the independence assumption is important when applying naïve Bayes classifier [5].

Up to point, independence assumption on which naïve Bayes classifications are based almost never holds for natural data sets. This issue has been the subject of a great deal of research. This research can be categorized in three main types: (1) attempt to produce better classification via relaxing the independence assumption, (2) the modification of the feature sets to make the independence assumption truer, and (3) attempts to explain why the independence assumption is not really necessary in the final analysis [6].

Recently, kernel techniques have been combined with many classification method to deal with non-linearity problem and to improve the classification performance [25, 27, 30-32].

In regards to the modification of the features, some studies have been done such as LIWEI, FAN [10] who applied ICA with naïve Bayes, his study showed that using of ICA with naïve Bayes significantly improved naïve Bayes classification performance. Also Bressan and Vitria [5] provided a framework for the use of class conditional independent component analysis (CC-ICA) and they also confirmed that using of CC-ICA increases naïve Bayes classification performance.

Although ICA and CC-ICA are effective with naïve Bayes, they still restricted to linear approaches and their affects with nonlinear problems are limited. In case of nonlinear problems KICA is an effective solution.

The additional advantage of the use of KICA as kernel approach is that it transfers the original $\boldsymbol{n} \times \boldsymbol{p}$ features matrix into $\boldsymbol{p} \times \boldsymbol{p}$ kernel space, that is very useful when the number of features ($\boldsymbol{P}$) is relative large to the number of instances ($\boldsymbol{n}$) as in many applications such as in the field of medical studies.

In logistic regression (LR) the features are required to be uncorrelated where they should be independent in naïve Bayes. KICA has been applied with logistic regression [27] and resulted in increasing the classification performance; consequently it supposed to increase naïve Bayes classification performance also.

To the best of current knowledge, effect of KICA has not yet been assessed with naïve Bayes classifier, so, in this paper KICA [16] with Gaussian kernel (RBF) as a nonlinear approach is investigated with naïve Bayes. For the assessing the performance of KICA, a comprehensive statistical comparison between KICA, ICA, and KPCA has been conducted; several machine learning measures are used.

2. Naïve Bayes Classifier

For a given X $(\boldsymbol{n} \times \boldsymbol{p})$ training data where n is the number of training data and p the number of features, these training data need to be classified according to a given class C. Naïve Bayes classifier can be used to classify multi classes however, this study is focused only on classifying binary class (0/1). Naïve Bayes classifier is dependent on the so called Bayes theorem [14] which can be explained as follows:

$$p(C = c_k / X = x) = \frac{p(C = c_k) * p(X = x / C = c_k)}{p(X = x)}, \tag{1}$$

where C_k represented the class where $k = 1, 2$ and X represented the training instances. The denominator in (1) can be described as follows:

$$p(X = x) = p(c_1)p(X = x / C = c_1) + p(c_2)p(X = x / C = c_2), \tag{2}$$

where $p(X = x / C = c_k)$ is defined as conditional probability that the instance x is belong to class $c_k (k = 1 \text{ or } 2)$, when $p(C = c_k / X = x)$ is exactly known for a classification problem.

Classification can be done in an optimal way for a wide variety of effectiveness measures [14, 15]. Since Equation (2) is invariant across classes, it has no effect on classification, so it can be omitted and Equation (1) can be written as follows:

$$p(C = c_k / X = x) \alpha \; p(C = c_k) * p(X = x / C = c_k). \tag{3}$$

Now suppose we assume for each variable x_j that its outcome is independent of the outcome of all other variables given class C_k. In this case, we can obtain the so-called naïve Bayes classifier as follows:

$$p(C = c_k / X = x) \alpha \prod_{j=1}^{n} p(C = c_k) * p(X_j / C = c_k), \tag{4}$$

where $p(X_j / C = c_k)$ is often called the likelihood of the data x_j given C_k.

Given an unseen test instance, the learner is asked to predict its class according to the evidence provided by the training data according to the following:

$$\phi(x) = \text{sgn } p(C_k/X),$$

where

$$\text{sgn } p(C_k/X) = \begin{cases} +1 \text{ if } p(C = c_k/X = x) > 0.5, \\ -1 \text{ if } p(C = c_k/X = x) \leq 0.5. \end{cases} \quad (5)$$

3. Independent Component Analysis

ICA [11, 12, 25] is a somewhat new computational statistical technique for data analysis. ICA originated from the signal-processing community, where it was developed as a powerful procedure for blind source separation [11]. The goal of ICA is to find representation of non-Gaussian data so those components are statistically independent or as independent as possible [12]. The basic ICA model for feature transformation can be written as:

$$s_t = ux_t, \quad (6)$$

where x_t is $n \times p$ matrix represents the observed feature vectors, s_t is $n \times p$ matrix represent the new independent estimated vectors for classification purpose, u is called the $n \times n$ de-mixing matrix is used to find an entirely new coordinate system of statistically independent non-Gaussian directions, with the first IC direction being the most non-Gaussian. The algorithm works iteratively and determines the most non-Gaussian direction first. Based on this direction it finds the next most non-Gaussian direction which is independent from the first, etc. For $n \times p$ dimensional data vectors, it determines up to $n \times p$ dimensional independent vectors, so it projects the feature vectors representing the original data into independent components, u must be estimated from the data. There are many algorithms have been developed for performing ICA, among them the fixed-point algorithm is a popular one. The fixed point fast ICA algorithm presented by Hyvarinen and Oja [13] is used in

this paper. In fast ICA, PCA is used to perform the whitening before estimating the independent components vectors; the original input vectors will be transformed to a set of new uncorrelated vectors with zero means and unity variance. After the process of data whitening is finished, the fixed point-algorithm is performed to estimate the transformation matrix and independent components. A measure of the dependence between random variables is called mutual information. Minimizing the mutual information between the components is equivalent to maximizing their negentropy. The negentropy in the fast ICA can be approximately expressed as follows:

$$J_G(s_{t(i)}) \approx [E\{G(s_{t(i)}) - E\{G(V)\}]^2, \tag{7}$$

where G is practically any non-quadratic function, V is a Gaussian variable with zero mean and unit variance and μ_i is n-dimensional vector, comprising one of the rows of the matrix u. There are many functions can be used as G [11]. Substituting $s_{t(i)} = u_i T_{x_t}$ in Equation (7) obtaining the following optimization problem:

$$Maximize \sum J_G(\mu_i) = [E\{G(\mu_i^T x) - E\{G(V)\}]^2 \tag{8}$$

Subject to

$$E\{(\mu_i^T x)^2\} = 1, \quad i = 1, 2, \ldots, n. \tag{9}$$

One new independent component can be estimated by solving this optimization problem through the fast ICA algorithm and based on this the whole reduced independent components matrix s_t^* can be estimated. In this paper, fast ICA with skew is used to compute the independent components.

4. Kernel Independent Component Analysis (KICA)

Kernel principle component analysis (KICA) [28, 29] is the kernel version of ICA; for a given training data x, suppose this training data is mapped to new feature space F through $\Phi(x)$, $F = \Phi(x)$ some nonlinear

mapping. Where, $k(x_i, x_j) = \Phi^T(x_i)\Phi(x_j)$ is a Mercer's kernel that allows the calculation of the dot product in this space without explicitly knowing the nonlinear mapping. In this nonlinear space, the centering and whitening that have been discussed in the previous section of ICA is obtained as follows:

For the centering task the data $\Phi^T(x_i)$, where $i = 1, 2, \ldots, k$, should be transformed to

$$\Phi^*(x_i) = \Phi(x_i) - E(\Phi(x_i), \tag{10}$$

where

$$E(\Phi^*(x_i)) = 0.$$

For the whitening in this space, the task here is to find a transformation matrix Q satisfy that the covariance matrix of the $(\Phi(x_i) = Q(\Phi^*(x_i)$ data is unit matrix.

For $Z \in x$ arbitrary vector the KICA transformation can be obtained as: $Z^* = W^*Q\Phi(Z)$, where W^* denotes the orthogonal transformation matrix that can be obtained as described for ICA, while Q is the matrix obtained from kernel centering and whitening. In this paper, kernel-ICA with Gaussian kernel is used.

5. Materials and Methods

5.1. The data sets

The data sets that are used in this study are composed of five numerical features. These data sets are downloaded from the UCI repository (http://archive.ics.uci.edu/ml/datasets.html). Table 1 give a numerical summary of the data sets.

Table1. Summary of the data sets

Data set	Data size	Number of features
Diabetes	768	8
Ionosphere	351	34
Heart	270	13
Breast cancer	569	30
Sonar	208	60

5.2. Statistical comparison methods

The statistical analysis is an essential part of any comparison, and the comparison' conclusion cannot be generalized unless the proper statistical tests are used. Since the nonparametric tests don't required any assumption regarding the data unlike the parametric 'tests, Wilcoxon signed-rank test is used in this paper. The Wilcoxon signed-ranks test [17]; it ranks the differences in performance measurement of the two algorithms for each data set, ignoring the signs, and compares the ranks for the positive and the negative differences. The differences are ranked according to their absolute values; average ranks are assigned in case of ties. Let R+ be the sum of ranks for the data set, in which the second algorithm outperformed the first. Let R– be the sum of ranks for the opposite. Ranks of $di = 0$ are split evenly among the two sums. Let $T = \min(\mathrm{R}+, \mathrm{R}-)$, then the test statistics is computed as follows:

$$Z = \frac{T - \frac{1}{4}(N(N+1))}{\sqrt{\frac{1}{24}(N(N+1)(2N+1)}}. \tag{11}$$

The statistic in Equation (11) is approximately follows normal distribution. Case of $Z > Z_{(\alpha/2)}$ indicating that there is statistical significant different between the two methods.

5.3. Experimental set up

The kernel ICA with Gaussian kernel (RBF), ICA with skew and KPCA with Gaussian kernel (RBF) are used for modifying and obtaining new features. Naïve Bayes classifier with 10 fold cross-validation methods is built on these new features. The classification performance of naïve Bayes on KICA is compared to its classification performance on KPCA and ICA. To avoid the biasness Gaussian kernel (RBF) is used with both kernel methods.

All the performance measures: accuracy, sensitivity, specificity, F-score, precision, and AUC are computed for each classifier and method eventually the average is used. The ROC analysis is computed after each cross validation for each method, however, due to the limitation of this paper, the highest ROC curves for Sonar, Ionosphere, Heart and Breast cancer data sets are used. The Wilcoxon signed-ranks test is used for the comparing KICA with KPCA and with KICA, the test was applied only to AUC as since is it the most powerful machine learning measure [20].

6. Results and Discussions

The kernel –ica version available at http://people.kyb.tuebingen.mpg.de/arthur/fastkica.htm and the Statistical Pattern Recognition Toolbox for MATLAB (stprtool) version 2.11 [18] are used for implementing KICA and KPCA, respectively. For the application of ICA, the fast- ICA-2.5 39 [19] software package is used. All methods have been applied under matlab (7.8.0347- R2009a) interface. The ROCs for the different methods are obtained by using spss.16.0 (SPSS Inc., Chicago, IL, USA).

6.1. Results

The results of the building of naïve Bayes on the features obtained by KICA, ICA, and KPCA are shown in Table 2, Table 3, and Table 4, respectively, where each value represent the average for the associated measure

Table 2. The results of the performance measures for naïve Bayes –KICA

Data set	**The performance measures**					
	Accuracy	Sensitivity	Specificity	F-score	Precision	AUC
Diabetes	0.767	0.865	0.589	0.650	0.701	0.827
Ionosphere	0.953	0.940	0.964	0.963	0.961	0.986
Heart	0.840	0.885	0.791	0.803	0.829	0.904
Breast cancer	0.969	0.970	0.969	0.960	0.952	0.994
Sonar	0.890	0.881	0.912	0.892	0.887	0.947

Table 3. The results of the performance measures for naïve Bayes –ICA

Data set	**The performance measures**					
	Accuracy	Sensitivity	Specificity	F-score	Precision	AUC
Diabetes	0.762	0.867	0.562	0.622	0.703	0.815
Ionosphere	0.877	0.763	0.942	0.904	0.872	0.89
Heart	0.799	0.536	0.894	0.871	0.859	0.841
Breast cancer	0.752	0.891	0.523	0.609	0.750	0.728
Sonar	0.650	0.617	0.687	0.668	0.671	0.726

Table 4. The results of the performance measures for naïve Bayes –KPCA

Data set	**The performance measures**					
	Accuracy	Sensitivity	Specificity	F-score	Precision	AUC
Diabetes	0.760	0.858	0.582	0.643	0.694	0.820
Ionosphere	0. 946	0.933	0.957	0.956	0.954	0.979
Heart	0.833	0.878	0.784	0.796	0.822	0.897
Breast cancer	0.962	0.963	0.962	0.953	0.945	0.987
Sonar	0.883	0.874	0.905	0.885	0.880	0.940

6.2. Discussions

Although several metrics are used in this study AUC has been recommended for assessing performance of machine learning algorithms [20, 30] accordingly, Wilcoxon test is used to test whether there is statistical significant difference between the classification performance of KICA and ICA and between KICA and KPCA. The values of Wilcoxon signed-ranks test for comparing KICA to ICA and KICA to KPCA are 0.04 and 0.025, respectively; this indicated that there is significant difference of KICA to ICA and KPCA.

Concerning the ROC curves of sonar, ionosphere, heart and breast cancer those are depicted in Figure 1, Figure 2, Figure 3, and Figure 4, respectively, it's clear that KICA curve is highest compare to ICA and KPCA. The relation among the RUC curves of KICA, ICA, and KPCA is depicted in Figure 5, it demonstrated that performance of KICA is higher than the performance of ICA and KPCA on most datasets. Consequently, KICA technique is performing better than both ICA and KPCA techniques.

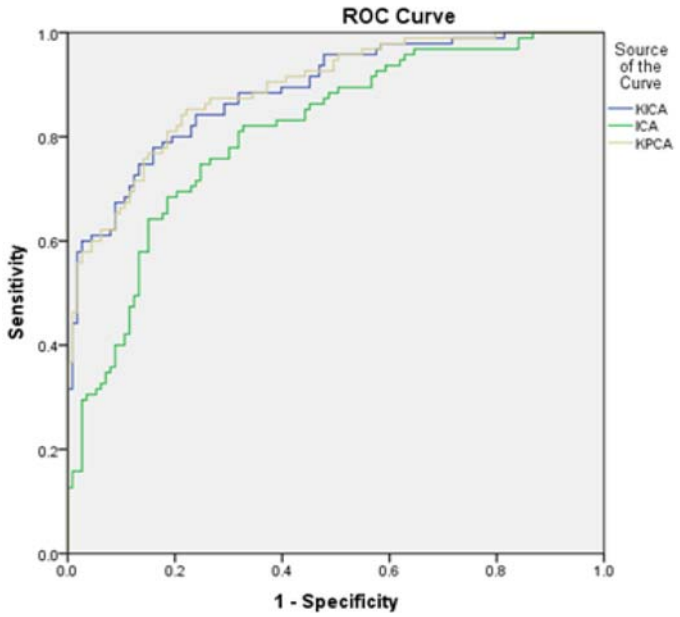

Figure 1. The ROC curve for sonar.

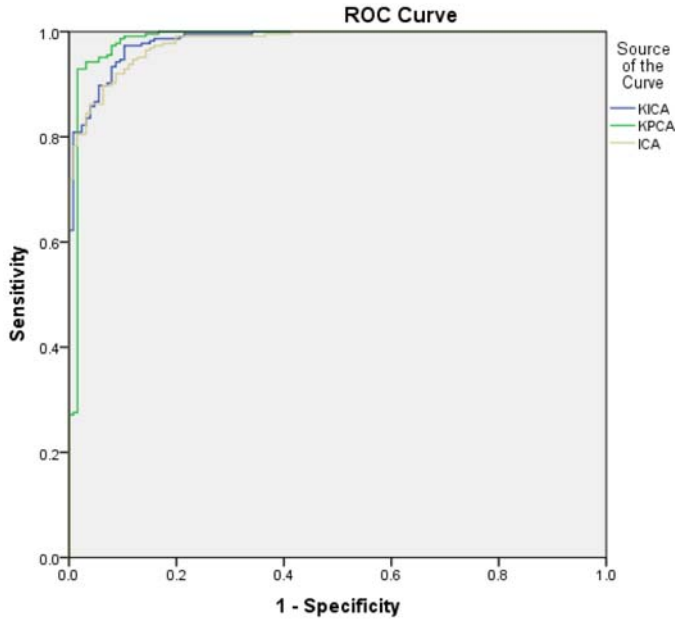

Figure 2. The ROC curve for ionosphere.

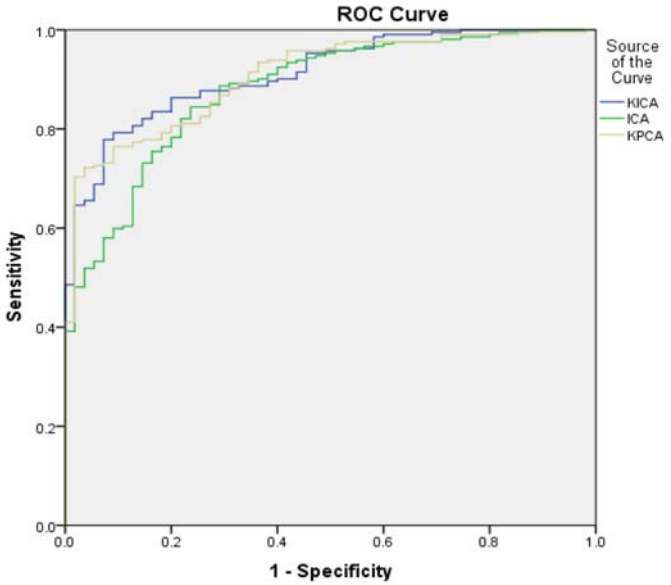

Figure 3. The ROC curve for heart.

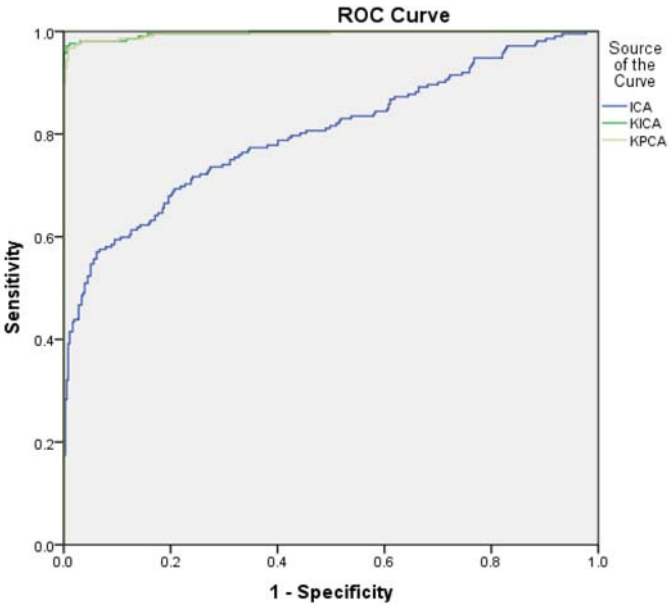

Figure 4. The ROC curve for breast cancer.

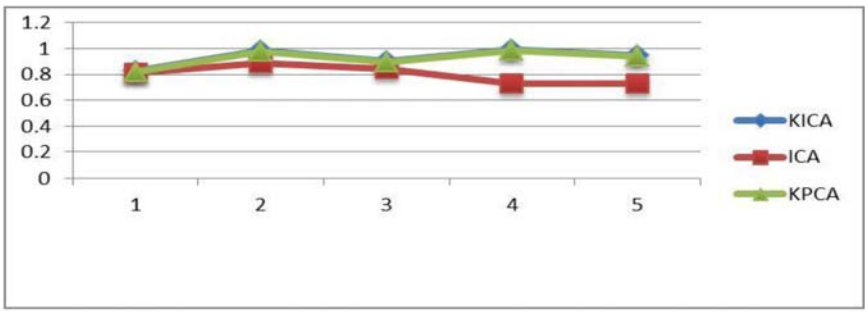

Figure 5. The relationship among RUCs of KICA, ICA & KPCA for the data sets.

7. Conclusion

Since building naïve Bayes classifier required independent features; for producing those independent features from the original features, this paper, proposed the application of KICA to yield independent features that naïve Bayes classifier can be built on. For assessing this technique KICA is compared to ICA and KPCA. The experiment demonstrated that the classification performance of naïve Bayes can be improved by using KICA. Moreover, KICA performing better than KPCA, that because KPCA yields uncorrected features in kernel space while KICA yields independent features in kernel space which satisfied the independence assumption naïve Bayes based on. From the results, it can be indicated that KICA is the best in compare to ICA and KPCA.

Acknowledgement

I wish to thank master degree students of the departments of statistic and computer Sciences at faculty of mathematics and computer sciences for their encouragement, useful discussions, and interest. Especial thanks goes to Dr. James Chambers for his efforts in improving the paper editing.

References

[1] Harry Zhang, The optimality of naive Bayes, AA 1(2) (2004), 3.

[2] Zengchang Qin, Naive Bayes classification given probability estimation trees, Machine Learning and Applications 2006, ICMLA'06, 5th International Conference on. IEEE, 2006.

[3] Andrew McCallum and Kamal Nigam, A comparison of event models for naive Bayes text classification, AAAI-98 Workshop on Learning for Text Categorization 752 (1998).

[4] Pedro Domingos and Michael Pazzani, On the optimality of the simple Bayesian classifier under zero-one loss, Machine Learning 29(2-3) (1997), 103-130.

[5] Marco Bressan and Jordi Vitria, On the selection and classification of independent features, Pattern Analysis and Machine Intelligence, IEEE Transactions on 25(10) (2003), 1312-1317.

[6] David D. Lewis, Naïve (Bayes) at forty: The independence assumption in information retrieval, Machine Learning: ECML-98, Springer Berlin Heidelberg (1998), 4-15.

[7] Nir Friedman, Dan Geiger and Moises Goldszmidt, Bayesian network classifier, Machine Learning 29 (1997), 131-163.

[8] Mehran Sahami, Learning Limited Dependence Bayesian Classifier, KDD 96 (1996).

[9] Pat Langley, Wayne Iba and Kevin Thompson, An analysis of Bayesian classifier, AAAI 90 (1992).

[10] Fan Liwei, Independent component analysis for naive Bayes classification, Diss. 2010.

[11] A. Hyvärinen, J. Karhunen and E. Oja, Independent Component Analysis, Wiley, New York, 2001.

[12] A. Hyvarinen and E. Oja, Independent component analysis: Algorithms and applications, Neural Net. 13 (2000), 411-430.

[13] A. Hyvarinen and E. Oja, A fast fixed-point algorithm for independent component analysis, Neural Computation 9(7) (1997), 483-1492.

[14] Richard O. Duda and Peter E. Hart, Pattern Classification and Scene Analysis, Vol. 3, Wiley, New York, 1973.

[15] David D. Wis, Evaluating and optimizing autonomous text classification systems, Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval, ACM 1995.

[16] Hao Shen, Stefanie Jegelka and Arthur Gretton, Fast kernel ICA using anapproximate newton method, International Conference on Artificial Intelligence and Statistics, 2007.

[17] J. Demsar, Statistical comparisons of classifier over multiple data sets, J. Mach. Learn. Res. 7 (2006), 1-3.

[18] Laurens van der Maaten, Statistical Pattern Recognition Toolbox for Matlab (stprtool) version 2.11, version 0.7.2b (2010).

[19] Hugo Gavert, Jarmo Hurri, Jaakko Sarela and Aapo Hyvarinen, Fast ICA for Matlab 7.x and 6.x, Version 2.5 (2005).

[20] A. P. Bradley, The use of the area under the ROC curve in the evaluation of machine learning algorithms, Pattern Recognition 30 (1997), 1145-1159.

[21] Xindong Wu et al., Top 10 algorithms in data mining, Knowledge and Information Systems 14(1) (2008), 1-37.

[22] Xindong Wu, and Vipin Kumar, eds, The Top Ten Algorithms in Data Mining, CRC Press, 2009.

[23] Lee H. Dicker, and Sihai D. Zhao, High-dimensional classification via nonparametric empirical Bayes and maximum likelihood inference, Biometrika (2016), asv067.

[24] David J. Hand, and Keming Yu, Idiot's Bayes-not so stupid after all?. International Statistical Review 69(3) (2001), 385-398.

[25] Abdallah Bashir Musa, A comparison of ℓ1-regularizion, PCA, KPCA and ICA for dimensionality reduction in logistic regression, International Journal of Machine Learning and Cybernetics 5(6) (2014), 861-873.

[26] Abdallah Bashir Musa, Logistic regression classification for uncertain data, Research Journal of Mathematical and Statistical Sciences-ISSN 2320 (2014), 6047.

[27] Abdallah Bashir Musa, Gene expression data classification with kernel independent component analysis, Research Journal of Mathematical and Statistical Sciences ISSN 2320: 6047.

[28] Xin Jin et al., Kernel independent component analysis for gene expression data clustering, Independent Component Analysis and Blind Signal Separation, Springer, Berlin Heidelberg (2006), 454-461.

[29] R. Francis Bach and Michael I. Jordan, Kernel independent component analysis, The Journal of Machine Learning Research 3 (2003), 1-48.

[30] Abdallah Bashir Musa, Comparative study on classification performance between support vector machine and logistic regression, International Journal of Machine Learning and Cybernetics 4(1) (2013), 13-24.

[31] J. Q. Gao, L. Y. Fan, L. Li et al., A practical application of kernel based fuzzy discriminant analysis, International Journal of Applied Mathematics and Computer Science 23(4) (2013), 887-903.

[32] J. Gao and L. Fan, Kernel-based weighted discriminant analysis with QR decomposition and its application face recognition, WSEAS Transactions on Mathematics 10(10) (2011), 358-367.

■