ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 1, Issue 2, December 2012

An Efficient Classification Mechanism Using Machine Learning Techniques For Attack Detection From Large Dataset

Vineet Richhariya[1],Dr. J.L.Rana[2],Dr. R.K.Pandey[3],Dr. R.C.Jain[4]

Dept. of CSE, LNCT, Bhopal, Madhya Pradesh,India[1]

Dept. of CSE, RRIST,Bhopal, Madhya Pradesh,India[2]

Dept. of CSE, BU UIT,Bhopal, Madhya Pradesh,India[3]

Dept. of CSE,SATI, Bhopal, Madhya Pradesh ,India[4]

Abstract: Recent internet based communication technology has an important part in our life. Cyber based communication and networks connections are very huge not just in the terms of size, but also in the terms of changing the services offered and the mobility of users that make them more vulnerable to various kinds of complex attacks. Security is the main issue of networking, as malicious activities perform in the network by inside and outside users. There are number of intrusions present in the network. There are number of strategy, which have been developed in order to detect malicious activities. But a single algorithm does not correctly classify the malicious activity. In this paper, we have used machine learning approaches based on K-mean clustering and Naive Bayesian, to efficiently detect the intrusions present in the network. These algorithms have resulted in improved Precision, and reduce the false positive rate in order to provide better performance as compared to some exiting research works.

Keywords: Intrusion Detection System, Machine Learning, Native Bayesian, K-means clustering, False Positive rate, Kddcup1999 Dataset.

I. INTRODUCTION

Internet and local area networks are expanding at an amazing rate in recent years, not just in the terms of size, but also in the terms of changing the services offered and the mobility of users that make them more vulnerable to various kinds of complex attacks. While we are benefiting from the convenience that new technology has brought us , computer systems are exposed to increasing number and complexity of security threats. Of particular importance, thus, is the ability of applying rapidly new network security policies in order to detect and react as quickly as possible to the occurring attacks. Different techniques have been developed and deployed to protect computer systems against network attacks (anti-virus software, firewall, message encryption, secured network protocols, password protection). Despite all the efforts, it is impossible to have a completely secured system. Therefore, intrusion detection[2] is becoming an increasingly important technique that monitors network traffic and identifies network intrusions such as anomalous network behaviors, unauthorized network access, or malicious attacks to computer systems. Intrusion detection systems are typically classified with respect to placement as: host based or network based [1]. A host based IDS will monitor resources such as system logs, file systems and disk resources; whereas a network based intrusion detection system monitors the data passing through the network. . Intrusion detection system was first introduce by James P.Anderson in 1980 and Dr. Dorothy Denning in 1987[3]. Various data mining techniques like decision Tree , Bayesian (NB) , Rule based, Fuzzy logic, support vector machine (SVM) are being used for intrusion detection by the researchers. There are two types of attacks in intrusion detection system. (1) Passive attack: Unauthorized way to access the data without any security policies. (2)Active attack: Unauthorized way to modify the IT resources. some attack are identified by the intrusion detection system [3].(1) it detect the cracking password and access the data.(2)it detect the flooding in the network.(3)it also detect the packet listening (3)it detect the information altering and deletion. Many challenges in the intrusion detection system: (1) Main challenges in intrusion detection system is alarm or alert. The alarms are how to maintain. The systems are containing the number of alert than intrusion detection system are occur the problem to determine which one alert are false positive. (2) Most of the time system are generate the false positive. (3)Attackers understand the weakness of the system and it attacks the system and access the data. Data mining techniques are the result of a long process of research and product development. This evolution began when business data was first stored on computers, continued with improvements in data access, and more recently, generated technologies that allow users to navigate through their data in real time. Data mining [9] takes this evolutionary process beyond retrospective data access and navigation to prospective and proactive

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 1, Issue 2, December 2012

information delivery. Data mining is ready for application in the business community because it is supported by three technologies that are now sufficiently mature: (1)Massive data collection (2)Powerful multiprocessor computers (3)Data mining algorithms.

II. RELATED WORK

Classification defines the task of data analysis, where a model or a classifier is constructed to predict categorical labels. Classification can be described as a supervised learning algorithm in the machine learning process. In classification a given set of records is divided into training and test datasets. The training dataset is used in building a classification model, while test data is used in validating the data models. Data classification is a two step process. In the first step a classifier is built describing a predetermined set of data classes or concepts. This is the learning step or training step. In the second step the model is used for classification. For supervised learning for intrusion detection, there are mainly supervised neural network (NN)-based approaches [4], [5] and support vector machine (SVM)-based approaches [6], [7].

a) Neural Network-based approaches*:* Author [8] proposed an NN for distinguishing between intrusions and normal behaviors. They unify the coding of categorical fields and the coding of character string fields in order to map the network data to an NN. Rapaka *et al.* [9] use execution numbers of system calls in a host machine as the features of network behaviors to train the NN. Zhang [5] propose an approach for intrusion detection using hierarchical NNs. Han and Cho [10] use evolutionary NNs to detect intrusions.

b) Support Vector machine based approaches*:* Mukkamala *et al.* [11], [12] use SVMs to distinguish between normal network behaviors and intrusions and further identify important features for intrusion detection. Mill and Inoue [13] propose the TreeSVM and ArraySVM for solving the problem of inefficiency of the sequential minimal optimization algorithm for the large set of training data in intrusion detection. Zhang and Shen [7] propose an approach for online training of SVMs for real-time intrusion detection based on an improved text categorization model. Aside from the aforementioned supervised-learning-based approaches for intrusion detection, decision tree [15] and discriminant analysis [16] are applied to detect intrusions. Comparisons between different classifiers and fusion of multiple classifiers for intrusion detection are studied in [17] and [18].

c) Unsupervised Learning-Based Approaches: Supervised learning methods for intrusion detection can only detect known intrusions. Unsupervised learning methods can detect the intrusions that have not been previously learned. Examples of unsupervised learning for intrusion detection include *K*-means-based approaches and self-organizing feature map (SOM)-based approaches [20].

d) K-means-based approaches: In [19] authors proposed a *K*-means-based clustering algorithm, which is named *Y* -means, for intrusion detection. Xian *et al.* [21] combine the fuzzy *K*-means method and a clonal selection algorithm to detect intrusions. Jiang *et al.* [22] uses the incremental clustering algorithm that is an modification of the *K*-means algorithm to detect intrusions. All of these techniques improve the detection rate for intrusion detection but no able to solve the class dominance problem of k-mean clustering So for removing this problem we are proposing new mechanism which removes the class dominance problem along with the no class problem. In class dominance problem low instance classes (i.e. R2L and U2R) are dominated by high instances classes. In no class problem some of the clusters are assigned to no class.

III. PROPOSED ALGORITHMS

A. *K-MEANS WITH NAÏVE BAYESIAN*

The *k*-means algorithm is a algorithm, which belongs to the partition method. The *k*-means[16]algorithm initializes *k* clusters by arbitrarily selecting one object to represent each cluster. Each of the remaining objects are assigned to a cluster and the clustering criterion is used to calculate the cluster mean. These means are used as the new cluster points and each object is reassigned to the cluster that it is most similar to. This continues until there is no longer a change when the clusters are recalculated. Given a set of observations ($\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_n$), where each observation is a *d*-dimensional real vector, *k*-means clustering aims to partition the *n* observations into *k* sets ($k \le n$) $\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$ so as to minimize the within-cluster sum of squares (WCSS) :

$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2$$

where $\boldsymbol{\mu}_i$ is the mean of points in S_i.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 1, Issue 2, December 2012

The most common algorithm uses an iterative refinement technique. Due to its ubiquity it is often called the ***k*-means algorithm**; it is also referred to as **Lloyd's algorithm**, particularly in the computer science community.

B. *Naive Bayesian*

The Naïve Bayes method is based on the work of Thomas Bayes (1702-1761). In Bayesian classification [23], we have a hypothesis that the given data belongs to a particular class. We, then calculate the probability for the hypothesis to be true. This is among the most practical approaches for certain types of problems. The approach requires only one scan of the whole data. Also, if at some stage there are additional training data, then each training example can incrementally increase/decrease the probability that a hypothesis is correct. A Bayesian network is used to model a domain containing uncertainty. This classifier is designed for use when features are independent of one another within each class, but it appears to work well in practice even when that independence assumption is not valid. It classifies data in two steps: A)Training step: Using the training samples, the method estimates the parameters of a probability distribution, assuming features are conditionally independent given the class. B) Prediction step: For any unseen test sample, the classifier computes the posterior probability of that sample belonging to each class. The method then classifies the test sample according the largest posterior probability. The class-conditional independence assumption greatly simplifies the training step since you can calculate the one-dimensional class-conditional density for each feature individually. In general the class-conditional independence between features is not true. This assumption of class independence allows the Naive Bayes classifier [4],[5] to better estimate the parameters required for accurate classification while using less training data than many other classifiers. This makes it particularly effective for datasets containing many predictors or features. Naive Bayes classification is based on estimating P(X|Y), the probability or probability density of features X given class Y.

Principal of Bayesian Algorithms:

$$P(H \mid X) = \frac{P(X|H)P(H)}{P(X)}$$

According to Bayesian theorem : X is data sample whose class label is unknown . The X data sample and H Hypothesis are used to determine the class label. Naive Bayesian classifiers use the bayes theorem[23] to classify the new instance of data. Each instance is a set of attribute values described by a vector, X = (x1, x2 . . . , xn). Considering m classes, the sample X is assigned to the class Ci if and only if

$$P(X|Ci)\ P(Ci) > P(X|Cj)\ P(Cj)$$

The sample data belongs to the class with maximum posterior probability for the sample. For categorical data P(Xk|Ci) is calculated as the ratio of frequency of value X(k) for attribute Ak and the total number of samples in the training set. For continuous valued attributes a Gaussian distribution can be assumed without any loss of generality. In naive Bayesian approach the attributes are assumed to be conditionally independent. In spite of this assumption, naive bayesian classifiers give satisfactory results because focus is on identifying the classes for the instances, not the exact probabilities. Application like spam mail classification, text classification can use naïve bayesian classifiers. Theoretically, Bayesian classifiers are least prone to errors. The limitation is the requirement of the prior probabilities. The amount of probability information required is exponential in terms of number of attribute, number of classes and the maximum cardinality of attributes. The algorithm first searches for the multiple copies of the same example in the training data D, if found then keeps only one unique example in the training data D (suppose all attribute values of two examples are equal then the examples are similar). Then the algorithm disc reties the continuous attributes in the training data D by finding each adjacent pair of continuous attribute values that are not classified into the same class value for that continuous attribute. Next the algorithm estimates the prior P(Cj) and conditional P(X|Cj) probabilities in the training data D. The prior probability P(Cj) for each class is estimated by counting how often each class occurs in the training data D. For each attribute Ai the number of occurrences of each attribute value X can be counted to determine P(Ai). Also, the Conditional probability P(X|Cj) for each attribute values X can be estimated by counting how often each attribute value occurs in the class in the training data D. Then the algorithm classifies all the examples in the training data D with these prior P(Cj) and conditional P(X|Cj) probabilities. For classifying the examples, the prior and conditional probabilities are used to make the prediction. This is done by combining the effects of the different attribute values. When the prior probabilities are not known. Then prior probability are calculated by P (Ci) =Si/S. here Si is Number of training sample or S is total no of sample. If not known the assume the prior probability is equally

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 1, Issue 2, December 2012

P(C1)=P(C2)=……..=P(Cn). With increase in number of classes or attributes, the space and computational complexity of Bayesian classifiers increases exponentially.

Proposed Hybrid NB with K-Means based algorithms for intrusion detection.

INPUT : M is a subset of KDDCUP 1999 dataset
OUTPUT: Detection of attack from test dataset
Learning algorithm:
1.Define the number of cluster on the basis of similarity. (K>=n)
Where K = no. of cluster and n = no. of data sample.
2. Compute the centroids of each cluster.
3. Calculate the prior probability using

P (Ki)=Si/S

Where Ki=No. of Cluster label in Kddcup database.
Si=No. of Training Sample for the each cluster.
S=Total number of Sample.
4. Calculate the posterior probability
Here we are calculate the posterior probability t

M=P (H/X)

$P(H/X) = P(X/H) * P(H) / P(X)$

H=Hypothesis , X= Kddcup data sample.

The Algorithm makes the assumption that all the input attributes are independent, such as one attribute doesn't affect the other in deciding whether or not a condition will met in the database.
5. From the dataset ,for each attribute Ai the number of occurrences of each attribute value X can be counted to determine P(Ai).
P(A/C)=P(Ai)/P(Ki) Here
Ai=No. of attribute are used in the M
Ki=No. of clusters in the M, Where i=0,1,2,3,4,
6. Classifying all the training examples using these prior and conditional probabilities,

$$\text{Posterior} = \frac{\text{Likelihood x Prior}}{\text{Evidence}}$$

$$P(H \mid X) = \frac{P(X|H)P(H)}{P(X)}$$

Then P(Ki) would be the probability that the how many clusters in the Dataset, and P(X|K) is the probability that the given condition (input) in the dataset, given that the condition in the dataset. P(X) is just the probability of a condition appearing in database.

7. Now Calculate the maximum posterior probability of the tuple from dataset.

8. Again classify all training examples in M using updated probability values.

IV. EVALUATION RESULT OF THE PROPOSED WORK

Our experimental work of network intrusion detections carried out using proposed techniques over KDDCup'99 data set[25] . This data set is benchmarked for doing the research in the field of IDS. It consists of 41 features of network transaction and a associated class label. We have evaluated our approaches and compared the results with the results obtained by other researchers.We evaluate the performance of our system by estimating the Precision Rate , the false positive rate and accuracy. The detection rate is the number of attacks detected by the system divided by the number of attacks present in the data set. The false positive rate is the number of normal connections that are misclassified as attacks divided by the number of normal connections in the data set. In order to evaluate the performance of proposed learning algorithm, we applied 5-class classification using KDD99 network intrusion detection benchmark dataset. All experiments were performed using an Intel Core 2 Duo Processor 2.5 GHz processor (2 MB Cache, 600 MHz FSB) with 2vGB of RAM. The detection rates (DR), false positives (FP) and accuracy are used to estimate the performance of IDS which are given as below :

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 1, Issue 2, December 2012

$$\text{Precision Rate} = \frac{TP}{TP+FP} * 100 \quad \text{---------- (1)}$$

$$\text{False Positive Rate} = \frac{FP}{TN+FP} \quad \text{----------(2)}$$

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} * 100 \quad \text{----------(3)}$$

Table 1 Parameters for performance estimation of intrusion detection system

Parameters	Definition
True Positive (TP) or Detection Rate (DR)	Attack occur and alarm raised
False Positive (FP)	No attack but alarm raised
True Negative (TN)	No attack and no alarm
False Negative (FN)	Attack occur but no alarm

Here TP=True Positive, TN=True Negative, FP=False Positive, FN=False Negative.

.

The simulated attacks were classified, according to the actions and goals of the attacker. In network all the attacks type fall into one of the following four main categories:

1. Denials-of Service (DoS) attacks : Denials-of Service (DoS) attacks have the goal of limiting or denying services provided to the user, computer or network
2. Probing or Surveillance attacks :Probing or Surveillance attacks have the goal of gaining knowledge of the existence or configuration of a computer system or network.
3. User-to-Root (U2R) attacks : User-to-Root (U2R) attacks have the goal of gaining root or super-user access on a particular computer or system on which the attacker previously had user level access.
4 Remote-to-Local(R2L) attack : Remote-to-Local(R2L) attack is an attack in which a user sends packets to a machine over the internet, which the user does not have access to in order to expose the machine vulnerabilities and exploit privileges which a local user would have on the computer.

Experimental results are as shown in the following subsection.

A. **Result for Naïve Bayes:**

After operating of proposed NB based approach following CONFUSION MATRIX IS OBTAINED.

TABLE 2: Confusion Matrix for NAÏVE BAYES

Actual	Predicted Normal	Predicted DOS	Predicted Probing	Predicted U2R	Predicted R2L
Normal	39809	15113	1671	3117	883
DOS	6531	164935	58213	174	0
Probing	110	35	4017	4	0
U2R	30	639	232	3679	150
R2L	137	10075	556	449	605

B. **Result for KMNB:**

In the following table the confusion matrix of our proposed hybrid learning algorithm i.e. Combination of supervised and unsupervised learning is shown.

ISSN: 2319 – 8753

International Journal of Innovative Research in Science, Engineering and Technology
Vol. 1, Issue 2, December 2012

Table 3: Confusion Matrix for Proposed technique

Actual	Predicted Normal	Predicted DOS	Predicted Probing	Predicted U2R	Predicted R2L
Normal	40630	6471	3	33	118
DOS	3224	165008	71	195	133
Probing	1094	58025	4082	188	535
U2R	3658	349	10	4172	360
R2L	11989	0	0	7	10676

C. **Performance Comparison**:

Result are given in the Table 4 to compare the classifier, we have record precision rate and false positive rate of each algorithms. The best algorithms for the given attak category. Proposed work is compared with other algorithm the BPNSQAQ, ID3 and Naïve bays algorithm. So the result for KMNB algorithm is improved for other algorithm.

Performance comparison in terms of Precision rate:

Here we are comparing with all classifier to our proposed work on the Precision Parameter. Proposed KMNB provides better results from all other techniques in all attack class type except in DOS type , where the obtained result is comparable with BPNSQAQ method.

Table 4. Comparision the precision rate among ID3, BPNSQAQ, NB and Proposed techniques

Algorithms	DOS	Probing	U2R	R2L
BPNSQAQ	99.1	75.8	49.2	76.5
ID3	92.41	94.75	41.86	77.94
Proposed Naïve Bayes	94.6	78.6	54	40.6
Proposed KMNB work	98.7	98.4	99.2	98.9

Performance comparison in terms of False Positive rate:

Here we are comparing our proposed work with all other techniques on the False positive Parameter. It is found that FPR is very less in our method from other techniques in case of Probing, U2R & R2L attack types but very near with BPNSQAQ in case of DOS attack type.

Table5 : Comparison of FPR among various techniques

Algorithms	DOS	Probing	U2R	R2L
BPNSQAQ	0.009	0.242	0.908	0.235
ID3	0.0458	0.0006	0.00038	0.000229
Proposed Naïve Based method	0.2	0.04	0.07	0.02
Proposed Machine Learning (KMNB)	0.013	0.00007	0.00008	0.0002

So, from the obtained experimental results it can be understand that the proposed Machine learning approach is providing efficient results in terms of FPR as well as precision rate.

V. CONCLUSION

In modern society, the security of computer networks becomes an increasingly vital issue to be solved. Traditional intrusion detection techniques lack extensibility in face of changing network as well as adaptability in face of unknown attack type. In this paper, a modified detection algorithms based on naive bayes and unsupervised method k-means is used. The sum of the Euclidean distance of the points from the respective cluster centers is adopted as the similarity metric. After finding the optimal cluster center modified NB is applied to detect the attacks. Experiment results obtained proves the effectiveness of the mechanism. Results are compared on the basis of Precision Rate ,False Positive Rate and are found better among some the existing research work. In future the proposed methodology can be tested with reduced dimensionality i.e. after selecting best features from the KDD dataset , better classification results can be obtained.

REFERENCES

[1] Priyanka J Pathak and Snehlata S Dongre, "Intrusion Detection through Ensemble Classification Approach", *IJCA Proceedings on 2nd National Conference on Information and Communication Technology* NCICT (1):11-15, November 2011.
[2] James P.Anderson, *"Computer security threat monitoring and surveillance"*, Technical Report 98-17,james P.Anderson Co.,Fort Washington,Pennsylvania, USA,April 1980.
[3] Dorothy E.Denning,"*An intrusion detection model",IEEE Transaction on Software Engineering*,SE-13(2),1987,pp.222-232.
[4] Y.-H. Liu, D.-X. Tian, and A.-M. Wang, "Annids: Intrusion detection system based on artificial neural network," in *Proc. Int. Conf. Mach Learn. Cybern.*, Nov. 2003, vol. 3, pp. 1337–1342.
[5] C. Zhang, J. Jiang, and M. Kamel, "Intrusion detection using hierarchical neural networks," *Pattern Recognit. Lett.*, vol. 26, no. 6, pp. 779–791, May 2005.
[6] P. Hong and R. E. Schapire, "An intrusion detection method based on rough set and SVM algorithm," in *Proc. Int. Conf. Commun., Circuits Syst.*, Jun. 2004, vol. 2, pp. 1127–1130.
[7] Z. Zhang and H. Shen, "Online training of SVMs for real-time intrusion detection," in *Proc. Int. Conf. Adv. Inf. Netw. Appl.*, 2004, vol. 1, pp. 568–573.
[8] J. M. Bonifacio, Jr., A. M. Cansian, A. C. P. L. F. De Carvalho, and E. S. Moreira, "Neural networks applied in intrusion detection systems," in *Proc. IEEE Int. Joint Conf. Neural Netw.*, 1998, vol. 1, pp. 205–210.
[9] A. Rapaka, A. Novokhodko, and D. Wunsch, "Intrusion detection using radial basis function network on sequences of system calls," in *Proc. Int. Joint Conf. Neural Netw.*, 2003, vol. 3, pp. 1820–1825.
[10] S. J. Han and S. B. Cho, "Evolutionary neural networks for anomaly detection based on the behavior of a program," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 36, no. 3, pp. 559–570, Jun. 2006.
[11] S. Mukkamala, G. Janoski, and A. H. Sung, "Intrusion detection using neural networks and support vector machines," in *Proc. Int. Joint Conf. Neural Netw.*, 2002, vol. 2, pp. 1702–1707.
[12] A. H. Sung and S. Mukkamala, "Identifying important features for intrusion detection using support vector machines and neural networks," in *Proc. Symp. Appl. Internet*, 2003, pp. 209–216.
[13] J. Mill and A. Inoue, "Support vector classifiers and network intrusion detection," in *Proc. Int. Conf. Fuzzy Syst.*, 2004, vol. 1, pp. 407–410.
[14] Scarfone, Karen, Mell, Peter.,"Guide to Intrusion Detection and Prevention Systems (IDPS)". Computer Security Resource Center (National Institute of Standards and Technology) (800-94). Retrieved 1 January 2010.
[15] T. Abbes, A. Bouhoula, and M. Rusinowitch, "Protocol analysis in intrusion detection using decision tree," in *Proc. Int. Conf. Inf. Technol.: Coding Comput.*, 2004, vol. 1, pp. 404–408.
[16] M. Asaka, T. Onabura, T. Inoue, and S. Goto, "Remote attack detection method in IDA: MLSI-based intrusion detection using discriminant analysis," in *Proc. Symp. Appl. Internet*, 2002, pp. 64–73.
[17] S. Mukkamala, A. H. Sung, and A. Abraham, "Intrusion detection using an ensemble of intelligent paradigms," *J. Netw. Comput. Appl.*, vol. 28, no. 2, pp. 167–182, Apr. 2005.
[18] S. Mukkamala and A. H. Sung, "A comparative study of techniques for intrusion detection," in *Proc. Int. Conf. Tools Artif. Intell.*, 2003, pp. 570–577.
[19] W. Lee and S. J. Stolfo, "A framework for constructing features and models for intrusion detection systems," *ACM Trans. Inf. Syst. Secur.*, vol. 3, no. 4, pp. 227–261, Nov. 2000.
[20] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, " On the capability of an SOM based intrusion detection system," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2003, vol. 3, pp. 1808–1813.
[21] J. Xian, F. Lang, and X. Tang, "A novel intrusion detection method based on clonal selection clustering algorithm," in *Proc. Int. Conf. Mach. Learn. Cybern.*, 2005, vol. 6, pp. 3905–3910.
[22] S. Jiang, X. Song, H.Wang, J. Han, and Q. Li, "A clustering-based method for unsupervised intrusion detections," *Pattern Recognit. Lett.*, vol. 27, no. 7, pp. 802–810, May 2006.
[23] John, G.H., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: Proc. of the 11th Conf. on Uncertainty in Artificial Intelligence (1995).
[24] Cheeseman, P., & Stutz, J. (1996). Bayesian clustersification (AutoClusters): Theory and results. In *Advances in knowledge discovery and data mining*, 153–180. Menlo Park,CA: AAAI Press.
[25] The KDD Archive. KDD99 cup dataset, 1999. http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html