Chapter - 4
Machine Learning based Feature Discrimination for Discerning Nominal and Numeric Variables

Authors

J. Anitha
Associate Professor, Department of CSE, Malla Reddy Engineering College, Hyderabad, Telangana, India

K. Srikanth
Ph.D. Research Scholar, Department of CS & SE, Andhra University, Visakhapatnam, Andhra Pradesh, India

B. Manasa
Ph.D. Research Scholar, Department of CSE, GITAM University, Visakhapatnam, Andhra Pradesh, India

Chapter - 4

Machine Learning based Feature Discrimination for Discerning Nominal and Numeric Variables

J. Anitha, K. Srikanth and B. Manasa

Abstract

In the preceding, we discussed the proposed model of replacing the distance function of Non-Linear Dimension Reduction by Gower and thereafter with its weighted variant that handle the mixed data efficiently, and at the same time prevent the attributes dominating others. The scatter plots showed that the proposed models reported superior performance. In the last section, the problems of the proposed model were discussed and one of them was the dependence of manual expertise for classifying a variable as nominal or numeric (once the categorical values have been replaced by numeric equivalents). In this work, a method of automatic feature type prediction based on Machine Learning (ML) approach is proposed. The experimental results on the created data show that the proposed model based on a binary SVM classifier was able to correctly predict the class of feature for 98.2% of instances correctly.

Keywords: machine learning, SVM, ANN

1. Introduction

In mixed data analysis, we are often encountered with the situations where it is imperative to assert the variables as qualitative and quantitative given the fact that the two have to be dealt differently, while calculating the similarity or difference of the entities composed of such variables. In such cases, it is often left to the expertise of adroit to manually categorize the variables as being nominal or numeric. An improper categorization of the variable is going to result in erroneous (dis) similarity and thereby negatively impact the final decision i.e., classification or regression. In this work, we propose an ML based classification approach for discriminating between the variables.

Many ML algorithms have the intrinsic property of dealing with varied data differently. Even though the ML algorithms deal with the attributes

differently based on their type, there is no way for the algorithms to know about the type of the variable and in that case, it is the responsibility of the user to signal the learning algorithm about the type of variable. Till now there is no well-defined mechanism to determine the type of variable, rather, it entirely depends on the user experience. Any research effort that intending the learning to be fully autonomous should, therefore, contain a mechanism with the sole responsibility of determining the type of each variable. Subsequently, the learning algorithm will go ahead with their normal working procedure. This way the learning can be fully automatic with the responsibility of user's type assignment being cut down. This part of the research work is dedicated to find a well-defined type assignment mechanism. Although the data can be of various types this work only considers to differentiated between qualitative and quantitative data as these two types are most popular in ML fraternity.

Before going forward and discussing the procedure to predict the type for data variables, we need to understand different dynamics of the data like what forms the data takes, what are the most popular forms of data, how to convert from one form to another, etc. The following few subsections present enough information in this regard. Thereafter a discussion about various Entropy of variables will follow.

2. Types of data

Concerning statistical investigation, data can be thought of as a collection of various snippets of information or facts, commonly known as variables. A variable is an identifiable bit of data containing one or more values. Those values can appear as a number or text (which could be converted into a number). Of course, data can be gathered in a various ways; however, the typology of the result can be without much of a stretch, normally recognized by a straightforward test. On the off chance that we need to quantify an amount identified with a particular event, we gather numbers that recognize quantitative factors. In the event that we need to portray the quality of an observed phenomenon, we cannot quantify it and we are collecting qualitative variables.

As already mentioned that the attributes used to portray real-life objects may be of various kinds. The typically used types of variables are given below, Interval-Scaled Variables: These variables are specified roughly on a linear scale. A variable of this type captures and represents intervals or differences between values. Some common examples of these variables are weight, height, latitude, longitude and temperature. Nominal Variables:

These variables distinguish an instance from another by having distinct names or values of attributes. Some common examples are employee ID, fingerprint, zip codes and gender. Binary Variables: Variables of this type are very much popular in the field of computer science, as computers can store only two values. The variables of this type can take on either of the two values 0 or 1, where the values depict the presence of a particular character.

Categorical variables: Also known as qualitative variables, can take on one of the values from a fixed and finite set. The presence of more than two states differentiates categorical variables from the binary variables, for example, the protocol type, service, intrusion classes in case of KDD99 data-set. Ordinal Variables: These variables also take more than two states as is the case with categorical variables, but with the meaningful ordering of the classes. Some common examples are grades obtained in an examination (e.g., A+, A, B+, B, C+, etc.) and height (e.g., tall, medium and short). Ratio-Scaled Variables: These variables are positive measurements on a non-linear scale such as an exponential scale. Here both differences and ratios are meaningful. Examples are the temperature in Kelvin, length, time and counts.

3. Nominal to numeric conversion

As it is clear by now that the data of interest takes many different formats like quantitative, qualitative, Date Time etc. As we have already mentioned that most of the ML algorithms are designed in a way that they able to deal efficiently with the data expressed as numbers. So, before feeding any of the qualitative data to the ML algorithm the categorical symbols should be expressed as numbers. Over the years a set of methods has evolved to solve this problem (Davis and Clark, 2011). The simplest of all is to simply discard the qualitative variables, but doing so will lead to the elimination of the information of interest. Below we mention some of the approaches that ML practitioners have used over the years. Dummy Coding Dummy coding is a commonly used method for converting a categorical input variable into a continuous variable. ‘Dummy’, as the name suggests is a duplicate variable which represents one level of a categorical variable. Presence of a level is represented by 1 and absence is represented by 0. For every level present, one dummy variable will be created.

Supposing that there is a data-set D of N dimensions, there is a categorical feature x with n different symbolic features. The dummy coding of the categorical variable x with a binary string of length n will result in the expansion of the data-set horizontally. So, the new data-set would be on N +

n dimensions. Since there will be a valid value for only one position indicator for each record this results in the sparsity of the data-set and also the wastage of the storage space. Yeung and Chow (2002) used this scheme to convert 7 symbolic features from the KDD99 data-set into numeric features. Doing so, they converted data-sets of 41 features to one with 119 features.

4. Label encoder

It is used to transform non-numerical labels to numerical labels (or nominal categorical variables). Numerical labels are always between 0 and n classes-1. Horng *et al*. (2011) have made use of the label encoder to perform conversion. A common problem with the nominal categorical variable is that it may decrease the performance of a model. For example, we have two features "age" (range: 0-80) and "city" (81 different levels). Now, when we'll apply label encoder to 'city' variable, it will represent 'city' with numeric values range from 0 to 80. The 'city' variable is now similar to 'age' variable since both will have similar data points, which is certainly not a right approach. Using ASCII Categorical values actually are only a sequence of characters, and as we know, there is an encoding mechanism to represent the characters in the computer systems known as ASCII. ASCII is a character encoding standard for electronic communication. ASCII codes represent text in computers, telecommunications equipment, and other devices. Efforts have been made over the past to replace a categorical value by summing up its constituent characters. Liu *et al*. (2004) converted categorical value into numeric equivalent by taking the sum of difference of ASCII of each character with that of uppercase A.

5. Motivation

In mixed data analysis, we have the data records composed of different types of variables. Some of them may be quantitative and some may be qualitative. The qualitative variables use a symbolic representation for different values. Taking an example of Protocol variable in KDD data-set. The Protocol variable can have three symbolic values namely TCP, Internet Control Message Protocol (ICMP) and Internet Gateway Protocol (IGP). But most of the ML techniques are designed in such a way that they need data to be expressed in a numeric form. There are many ways of converting nominal values to numeric values and one that has been well accepted in ML fraternity is the replacement of different symbolic values by some integer constant. As for that case, we can replace TCP by 1, ICMP by 2 and Internet Group Message Protocol (IGMP) by 3. Even though performing the

mathematical operations on numeric values may be feasible, but is perfectly illogical e.g., subtracting TCP by ICMP, multiplying ICMP by IGMP etc. does not carry any meaning.

6. k-NN

Towards better understanding of the concepts, lets consider a case of ´ k-NN, a lazy classifier. The motive of k-NN is the distance measure used to calculate the distances between the objects and thereby finding out the nearest neighbors for a given point x. There are many distance measures available in the literature and one that has been pretty popular is the Euclidean Distance, as already mentioned. The Euclidean Distance between two instances X and Y of d dimensions is calculated as

$$Euclidean = \sum_{i=1}^{d} \sqrt[2]{|x_i - y_i|^2}$$

As it is very clear from the fact given above that the Euclidean Distance measure is feasible for the numeric data only, a mere replacement of TCP with 1 or IGMP with 3, doesn't warrant the subtraction of the two, i.e., (3-1), which in turn means the difference (IGMP-TCP), that doesn't signify anything. Consequently, we mentioned that to deal with such case we should have a metric that is well suited for the mixed data and one metric that we found in the literature was Gower metric (Gower, 1971).

As can be inferred from the Gower, that it is defined in such a way that it deals with the two attributes differently. In the case of the numeric attributes, it uses simple Euclidean and for the nominal attributes, it performs a simple comparison. Although Gower or for that matter any mixed dis (similarity) metric deal with the two types of attributes differently but have no way of predicting the type of variable and have to be entirely dependent on the programmer for assigning the types to each variable. There opens the room for error, as the incorrect assignment of the type of variable is going to affect the dis(similarity) computation and thereafter the retrieval of the incorrect neighbor set which can negatively impact the final classification or regression. In order to make a metric fully unsupervised and to eliminate the chances of incorrect assignment, there should be an automated technique for predicting the type of variable.

7. Working hypothesis

Let us assume that there is a random variable X taking on N different values. Let us assume that out of the N values there are in total n unique values a feature X can take.

One of the working hypothesis for this work is that if,

- N tends towards infinity if the feature is quantitative.
- N tends to a finite constant (the number of modalities) if the feature is qualitative. In practice, at all the times we have the cases where N ≥ n. Constraints on N.
- N cannot be infinite as there is always a limit on N.
- In an effort to differentiate between the two types of variables, N must not be too small because for the small values of N, the two types of variables will have the same behavior.
- N cannot be too large: I
- N cannot be too large: If N ≤ Measurement limit or if N ≤ Media Limit, the increasing behavior of n = f(N).

Entropy

Entropy is a measure of the capriciousness of the state, or proportionately, of its average information content. The measure of information Entropy associated with each possible data value is the negative logarithm of the probability mass function for the value. Thus, when the data source has a lower-probability value (i.e., when a low-probability event occurs), the event carries more "information" (surprisal) than when the source data has a higher-probability value. The amount of information conveyed by each event defined in this way becomes a random variable whose expected value is the information Entropy. Generally, Entropy refers to disorder or uncertainty, and the definition of Entropy used in information theory is directly analogous to the definition used in statistical thermodynamics. The seminal work of Shannon, based on papers by (Shannon, 2001) and (Bromiley *et al*., 2004), rationalized these early efforts into a coherent mathematical theory of communication and initiated the area of research, known as information theory.

Shannon entropy

Entropy is the measure of disorder in physical systems or an amount of information that may be gained by observations of disordered systems. Claude Shannon defined a formal measure of Entropy, called Shannon Entropy. Given a series of events p1, p2..., pn the amount of information H(p) contained in the series is bounded to satisfy three requirements:

- H should be continuous in p_i.
- With all pi equally probable then H should be a monotonic increasing function of N.

- H should be additive. He then proved that the only H satisfying these three requirements is:

$$H = -K \sum_{i=1}^{N} p_i \log p_i$$

Where K is a positive constant, pi is the probability of occurrence of an event (feature value), xi being an element of the event (feature), X that can take values x1, x2..., xn. The Shannon Entropy is a decreasing function of a scattering of a random variable and is maximal when all the outcomes are equally likely.

Renyi entropy

Extensions of Shannon's original work have resulted in many alternative measures of information or Entropy. As a way of illustration, by relaxing the third requirement of Shannon, that of additivity, Renyi was able to extend Shannon Entropy to a continuous family of Entropy measures. The Renyi Entropy is important in ecology and statistics as an index of diversity. The Renyi Entropy is also important in quantum information, where it can be used as a measure of entanglement. Renyi Entropy of order α, where $\alpha \geq 0$ and α not equal to 1, is defined as

$$H_\alpha(P) = \frac{1}{1-\alpha} \log \sum_{i=1}^{N} p_i^\alpha$$

As $\alpha \rightarrow 1$, Renyi Entropy tends to Shanon Entropy.

Entropy projection

Figure. 1 given below presents the preliminary results with the Shannon Entropy on the data. As can be seen from the Graph given in Figure. 1 the Entropy of the quantitative variables tends to be lower than that of qualitative variables. Of course, the two sets are not disjoint as we would like it to be. Same behavior is also shown in the Figure. 2. As it can be interpreted from Figure. 1 and Figure. 2 the Entropy of a variable is an important variable for predicting its type but not sufficient as there was a mix of Entropy.

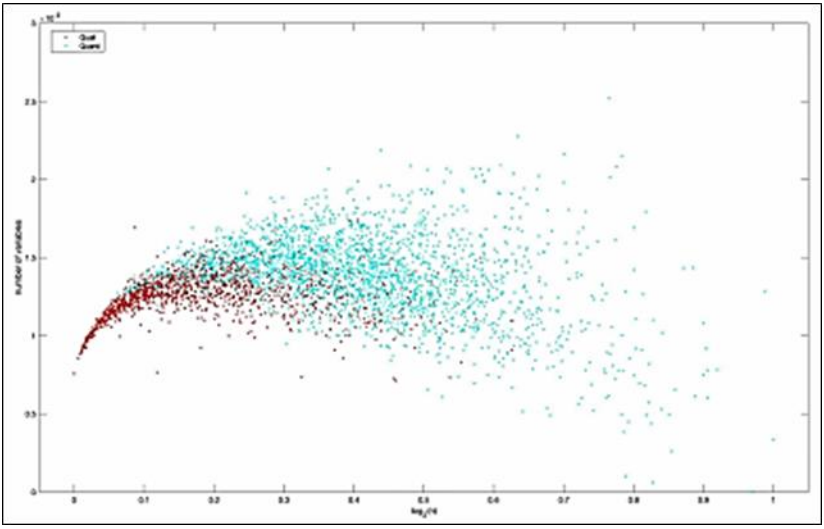

Fig 1: A Plot Shannon Entropy

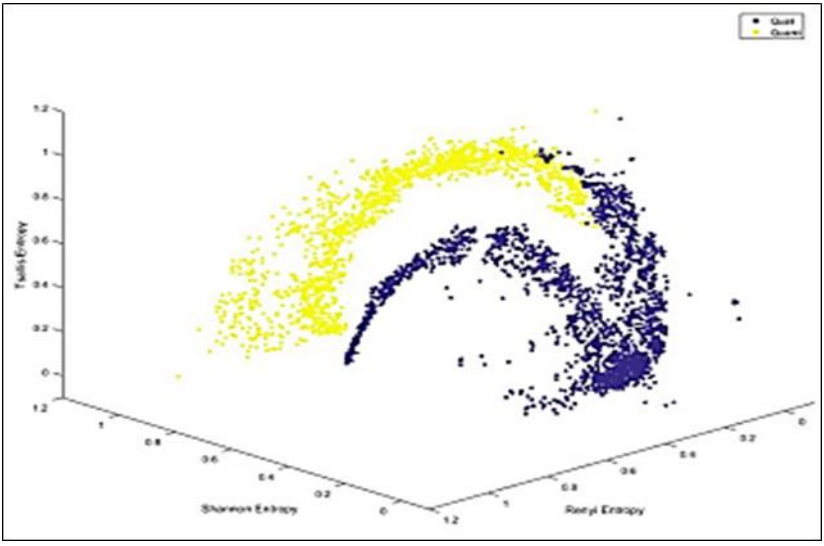

Fig 2: A Plot of three different Entropy

Data-set Formation A data-set was formed by extracting features from various public data-sets available on UCL library. As our goal was to form a data-set consisting of a mix of qualitative, i.e., data is represented using a symbolic scale if the feature is qualitative and quantitative features, i.e., data can be measured using a numerical or interval scale we selected the attributes across the data-set.

Shanon Entropy	Talis Entropy	Reyni Entropy	Type

Fig 3: Record Structure of Data-set

In order to obtain a representative data-set, we extract and concatenate the different features coming from various databank to obtain a large mixed-type features data-set. This new benchmark data-set is composed of mixed-type features. Some of them being numeric and quantitative (2477 features) and others being nominal and qualitative (1698 features). Finally it represented a data-set with 4175 mixed-type feature with dimensionality varying from 4 to 2450 dimensions. A qualitative feature uses symbols to identify categories while the quantitative features show numerical values on a well-defined interval scale that can be continuous or discrete. Symbolic features appear frequently in the network traffic data stream. Nevertheless, most ML methods are designed to work with numerical data. In order for these methods to use information from symbolic features in detection, some coding schemes are necessary. Afterwards, a coding scheme of arbitrary assignment that establishes a correspondence between each category of a symbolic feature and a sequence of integer was applied to replace the qualitative features with the numeric values. A mere replacement of the symbolic values with numbers doesn't ensure the validity of any mathematical operations except a simple comparison. Thereafter we had a data-set with all the features having numeric values but should not be confused between qualitative and quantitative. Subsequently, we calculated three different Entropies for each variable and stored in a separate file, in addition to this we stored the label against each row to signify if an attribute was numeric or nominal. Hence we have a data-set of size $4175 * 4$, where each record of the data-set is of the structure given in Figure 3. Thereafter we normalized the data-set to the range [0-1] so as to eliminate the chances of any attribute having greater values to dominate the attributes with lower values.

Classifier

Once a data-set has been formed, the next step is to subject this data-set to the classification. As we have already mentioned that each instance of the data-set can either be nominal or numeric, which means that there are only two labels in the data-set and hence is a binary classification problem. The next step is to select an appropriate classifier that is well suited for the binary classification. Through the extensive review of the literature, we found out that SVM is better suited for the binary classification. SVM belongs to the family of hyper-plane-based learning method where the objective is to

maximize the separation distance between two classes of objects. In addition to SVM we have taken other classifiers also, so as to show to provide a base of comparison among various classifiers. Table 6.1 lists out various classifiers that have been used in this work along with their configurations. For result evaluation we have used 10-fold cross-validation.

Table 1: Classifiers

S. No.	Classifier	Configuration
1.	*k-NN*	n = 5, Distance = Euclidean, No distance weighting, batch size = 100
2.	Artificial Neural Network (ANN)	Activation = Sigmoid, Layers = 3, Nodes = 3.5.2, learning rate = 0.3, momentum = 02
3.	SVM	Kernel = RBF, Gamma = 0.001 to 1000, eps = 0.001
4.	Decision Tree	Confidence Factor = 0.25, Punning = False
5.	Naïve Bayes	Batch Size=100, Use Supervised Discretization = False, use kernel Estimator = False

Methodology

This work commenced with the creation of data-set. Since this problem is novice, there is no benchmark data-set at present. We manually extracted the data from UCL library. In UCL library each data-set comprises a variety of features or attributes like qualitative, quantitative, binary etc. We took only nominal and numeric features. Considering the nominal features, the symbolic values were replaced by integer constants, as most of the ML algorithms deal with numeric data only. Numerical labels are always between 0 and n classes-1. Thereafter we calculated three different Entropies i.e., Shanon, Reyni and Tsallis for each variable. We stored the calculated Entropy values in separate excel, and in addition to this, we placed a binary label indicating if a variable was nominal or numeric. Likewise, we do for all the variables. Once we have prepared the dataset, we build a classification model using various classification algorithms. As far as the configuration of different classifiers is concerned we set it to the standard and well accepted one. Figure.4 presents the block diagram of this work. As can be seen from the Figure.4, the work is divided into two broad categories data-set preparation and classification, with feature extraction, nominal to numeric conversion and normalization belonging to the data-set preparation and the second phase consisting of classification.

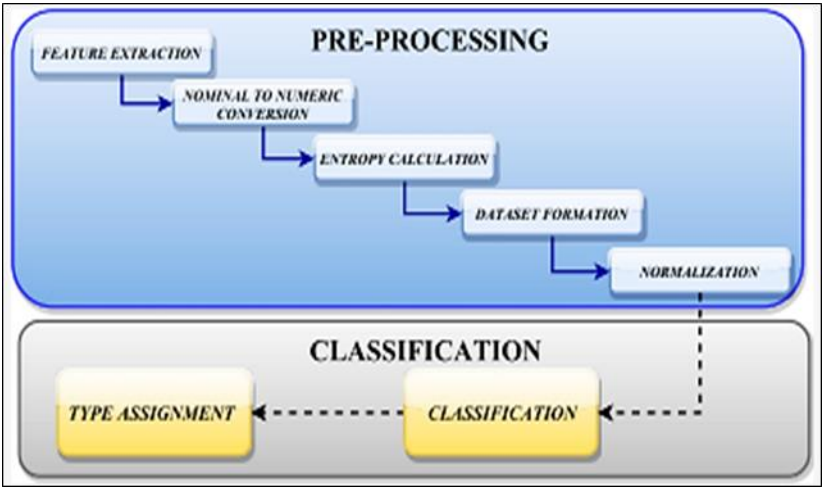

Fig 4: Feature Discrimination Block Diagram

In order to train and test the system, we made use of 10-fold cross-validation, wherein the data-set is partitioned into ten subsets, nine of which are used for training and one is spared for testing. The process is repeated a total of ten times. And finally, the average of the results over all the iterations is taken.

Performance measure

To check the effectiveness of any system, there should be some performance metrics that quantify the quality of the classification or clustering system. Since, our work is basically a classification process where-in the objective is to discern the features as nominal or numeric, we make use of the metrics that have been used extensively for the classification systems.

Table 2: Performance Metrics

Sno	Metric	Formula
1	Accuracy	$\frac{(AT\rightarrow AT+NR\rightarrow NR)}{(AT\rightarrow AT+NR\rightarrow NR++NR\rightarrow AT+AT\rightarrow NR)}$
2	Precision	$\frac{AT\rightarrow AT}{(AT\rightarrow AT+NR\rightarrow AT)}$
3	Recall	$\frac{AT\rightarrow AT}{(AT\rightarrow AT+AT\rightarrow NR)}$
4	f-Measure	$2\times\left[\frac{P\times R}{P+R}\right]$
5	ROC	$\frac{\alpha-\eta_\rho(\eta_\eta+1)/2}{\eta_\rho\cdot\eta_\eta}$

Table. 2 lists out the performance metrics that are used in this work along with their formulas.

8. Results and Discussion

A labeled data-set was constructed by manually taking the features from UCL library. Thereafter three different Entropies namely, Shannon, Reyn, and Cialis were calculated for each variable. Consequently, we applied many classifiers on the data-set. In total five different classifiers were applied to classify the data-set as presented in Table. 1. For the purpose of testing the model a 10-fold cross-validation approach is followed. The performance metrics mentioned in Table. 2 were used to evaluate the effectiveness of the classification model. Table. 3 presents the classification results reported by various algorithms. The accuracy of the model quantifies how precisely the model classifies the quantitative and qualitative data. As can be seen from the Table. 3 SVM has the highest accuracy of 99.951 followed by ANN with the accuracy of 97.859. We have applied SVM with RBF kernels on different gamma values and the results reported are the average over all the runs. As can be seen from the Table 6.3 that k-NN reports the lowest accuracy on all the data. The reason for such low results is the fact that kNN is the simple most classifier, and doesn't have a learning phase, rather it classifies the instance based on its neighboring instances, a proper neighborhood size, and an appropriate distance metric is a vital factor for the success of k-NN.

Table 3: Classification Results

Classifier	Accuracy	Precision	Recall	f-Measure	ROC	MAE
k-NN	56.837	0.323	0.568	0.412	0.61	0.0514
NB	73.537	0.677	0.785	0.713	0.937	0.0203
DT	92.740	0.989	0.927	0.8802	0.963	0.0063
ANN	97.859	0.962	0.979	0.971	0.991	0.0047
SVM	**99.951**	**0.999**	**0.999**	**0.999**	**1**	**0.0001**

SVM has better results than other classifier models not just in terms of accuracy but over all other metrics also. From the Table 6.3 it is clear that SVM reports precision 0.999 and recall 0.999 better than all the other classification models. Precision and Recall are two most popular metrics used to access the efficiency of any classification model, f-measure combines precision and recall to a single quantifiable inequality, which should be maximum. SVM returns a ROC of 1 which is much higher than all other classifiers. The Mean Absolute Error (MAE) of SVM is 0.0001 which is much lower than all the other classifiers with k-NN having the highest MAE of 0.0514. So, it is clear by now that SVM is able to classify the data-set accurately than other models. There were still few instances that SVM

classified incorrectly, but compared to all its competitors SVM has very good performance. The probable reason for SVM performing better than other algorithms for the problem is its effectiveness in dealing with voluminous data.

Summary

In this work, we provided a detailed discussion about various types of variables and mentioned that the two most frequent types of the data are nominal and numeric. Thereafter we discussed the ways of converting nominal to numeric. Consequently, we discussed the need of distinguishing the two types of variables and finally we proposed a methodology for the automatic attribute categorization. The results of the proposed model on the created data of 1000 instances, 630 tend to be nominal and 370 tend to be numeric shows that we were able to categorize the 93% of instances correctly.

References

1. Abbes T, Bouhoula A, Rusinowitch M. Efficient Decision Tree for Protocol Analysis in Intrusion Detection. International Journal of Security and Networks. 2010; 5(4):220-235.
2. Abduvaliyev A, Pathan ASK, Zhou J, Roman R, Wong WC. On the Vital areas of Intrusion Detection Systems in Wireless Sensor Networks. IEEE Communications Surveys & Tutorials. 2013; 15(3):1223-1237.
3. Abubakar AI, Chiroma H, Muaz SA, Ila LB. A Review of the Advances in Cyber Security Benchmark Datasets for Evaluating Data-driven based Intrusion Detection Systems. Procedia Computer Science. 2015; 62:221-227.
4. Aburomman AA, Reaz MBI. A novel SVM-kNN-PSO ensemble method for Intrusion Detection System. Applied Soft Computing. 2016a; 38:360-372.
5. Aburomman AA, Reaz MBI. Ensemble of Binary SVM classifiers based on PCA and LDA Feature Extraction for Intrusion Detection. In Proceedings of IEEE Advanced Information Management, Communicates, Electronic and Automation Control Conference, 2016b, 636-640.
6. Ahmed M, Mahmood AN, Hu J. A Survey of Network Anomaly Detection Techniques. Journal of Network and Computer Applications. 2016; 60:19-31.

7. Akhbardeh A, Jacobs MA. Comparative Analysis of Nonlinear Dimensionality Reduction Techniques for Breast MRI Segmentation. Medical Physics. 2012; 39(4):2275-2289.

8. Alarifi SS, Wolthusen SD. Detecting Anomalies in IaaS Environments through Virtual Machine Host System Call Analysis. In Proceedings of IEEE International Conference for Internet Technology and Secured Transactions, 2012, 211-218.

9. Almusallam NY, Tari Z, Bertok P, Zomaya AY. Dimensionality Reduction for Intrusion Detection Systems in Multi-data Streams-A Review and Proposal of Unsupervised Feature Selection Scheme. In Proceedings of Springer Emergent Computation, 2017, 467-487.

10. Amaral JP, Oliveira LM, Rodrigues JJ, Han G, Shu L. Policy and Network-based Intrusion Detection System for IPv6-enabled Wireless Sensor Networks. In Proceedings of IEEE International Conference on Communications, 2014, 1796-1801.