"Science Stays True Here"
Journal of Mathematics and Statistical Science, 1-13 | Science Signpost Publishing

Selecting Adaptive Number of Nearest Neighbors in *k*-Nearest Neighbor Classifier Apply Diabetes Data

Natalia Labuda[1], Julia Seeliger[1], Tomasz Gedrande[1], Karol Kozak[1,2]

1. Medical Faculty, Dresden University of Technology, Carl Gustav Carus University Hospital Dresden.

2. Faculty of Management, Finances and Informatics, Wroclaw University of Economy.

Abstract

The k-nearest neighbours (knn) is a simple but effective method of classification. *K* is the most important parameter in medical data classification based on *k*-nearest neighbor algorithm (knn). The major drawback with respect to knn is dependency on the selection of a "good value" for *k*. The value of *k* is usually determined by the cross-validation method but if *k* is too large, big classes will overwhelm small ones. On the other hand, if *k* is too small, the advantage of the knn algorithm will not be exhibited. Therefore, it is very likely that a fixed *k* value will result in a bias on large classes. In this paper we propose a modified *k*-nearest neighbor method, which uses different *k* values for different regions in an entire data set, rather than a fixed *k* value for a complete data set. The number of nearest neighbors is selected locally based on P-value Rate criteria. We apply the modified knn method to diagnose type II diabetes dataset which includes 768 samples from diabetic patients taken from Pima Indians Dataset.

1. Introduction

The *k*-nearest neighbours (knn) is a non-parametric classification method, which is simple but effective in many cases [2]. Many researchers have found that the knn algorithm achieves very good performance in their experiments on different data sets [3][4][5]. Also in categorization, diabetic data is one of the most popular algorithms [6]. For a data record *t* to be classified, its *k* nearest neighbors are retrieved, and this forms a neighborhood of *t*. Majority voting among the data records in the neighborhood

Corresponding author: Karol Kozak, Medical Faculty, Dresden University of Technology, Carl Gustav Carus University Hospital Dresden.

is usually used to decide the classification for t with or without consideration of distance-based weighting. However, to apply knn we need to choose an appropriate value for k, and the success of classification is very much dependent on this value. In a sense, the knn method is biased by k. There are many ways of choosing the k value, but a simple one is to run the algorithm many times with different k values and choose the one with the best performance.

In general, we hope to average as many nearest points as possible to obtain a small variance of the estimate, but the assumption above might no longer hold. A large bias may be introduced with the increasing value of k and, thus, with the enlarged neighborhood of a test patient. The k chosen by means of cross-validation aims to balance decreasing variance with increasing bias. Choosing a single value of k for all data points may not be optimal. For example, if the density of points varies across the predictor space, a constant value of k implies neighborhoods of differing sizes. Choosing k adaptively by examining local areas can be helpful. The question is: what criterion should be used to determine the optimal value of k? The next points of this paper will focus on methods which allow us to determine k for every new tested point x depending on the density region it is located. We test the modified knn method to diagnose type II diabetes dataset which includes 768 samples from diabetic patients taken from Pima Indians Dataset [15].

This paper is organized as follows. In section 2 we define the probability for decisions made by the majority rule based on a finite number of observations. Then we define the P-value Rate as the complement of the probability and use it as a criterion for determining the neighborhood size in the k-nearest neighbor rule. The details of the proposed method are presented in section 3. In section 4 we test our method on diabetic dataset. The conclusion will be given in section 5.

2. Classification

The final decision in a recognition task is affected by two types of "a priori" knowledge: the number of samples previously seen of each of the objects to be recognized, and the discriminant power provided by the features extracted. The prior knowledge is reflected in the *a priori probabilities* that measure how likely we are to find each type in the data set. The proportions in which each type (class) is present in the sample area may provide such a measure. If we let ω_i (i = 1, ..., M) denote the *state of nature*, i.e. the variable indicating the M possible classes, $P(\omega_i)$ denote the *a priori probabilities*. Generally speaking ω_i are called *classes* and the prior knowledge available is used to estimate the a *priori probabilities*.

If we let x denote the vector containing a set of measurements (parameters), the *state-conditional probability density* $p(x|\omega_i)$ express the probability density function for x given that the state of nature is ω_i.

The state conditional probability densities can also be estimated from the samples for each class. The two probabilistic measures derived by the samples, the *a priori* probability for each class $p(\omega_i)$ and the *state-conditional probability density* (*scpd*) $p(x|\omega_i)$, can be used to estimate the *a posteriori* probability $P(\omega_i|x)$ by means of Bayes rule:

$$P(\omega_i \mid x) = \frac{p(x \mid \omega_i)p(\omega_i)}{\sum_{j=1}^{M} p(x \mid \omega_j)P(\omega_j)} \tag{1}$$

Given the feature vector x of an unclassified pattern, classification is carried out estimating the posterior probabilities for each class and, deciding for the class with the higher posterior probability value. The effect of such a decision rule is to divide the feature space into M decision regions.

According to Bayes rule, the class with the largest posterior probability is selected as the label of x. Ties are broken randome. Bayes rule guarantees the minimum misclassification rate. Sometimes the misclassifications rate differently for different classes. Then we can use a *loss matrix* $\Lambda=[\lambda_{ij}]$, where λ_{ij} is a measure of the loss incurred if we assign class label ω_i when the true label is ω_j. The *minimum risk classifier* assigns x to the class with the minimum expected risk:

$$R_x(\omega_i) = \sum_{j=1}^{M} \lambda_{ij} P(\omega_j \mid x). \tag{2}$$

In general, the classifier output can be interpreted as a set of M degrees of support, one for each class (discriminant scores obtained through discriminant functions). We label x in the class with the largest support. In practice, a priori probabilities and the scpd are not known. The scpd can be estimated from the data using either a parametric or nonparametric approach.

When we look at the classification problem in a supervised classifier, we observe a labeled (training) set of n observations $O_n = \{(x_1, \omega_1), \ldots, (x_i, \omega_i)\}$, where x_i are the feature vectors, ω_i are scalar labels of the d-dimensional real vectors X_i and (x_i, ω_i) are assumed to be from some unknown distribution Q of (x, ω) on n dimensional space R with M number of classes ω, $R^d \times \{\omega_1, \ldots, \omega_M\}$. Here we assume simply that the training vectors are random vectors in d-dimensional Euclidean space with well-behaved distributions and a well-defined density function f.

The goal in supervised learning is to design a function $\Phi_n : R^d \rightarrow \{\omega_1, \ldots, \omega_M\}$ that maps a new feature vector x drawn from Q to its desired class from $\{\omega_1, \ldots, \omega_M\}$. K-nearest neighbor (knn) belongs

to one nonparametric approach in supervised learning strategies.

2.1 Probability in the *k*-Nearest Neighbors Method

Given n samples we can easily estimate the joint probability $p(x, \omega_i)$, by placing a cell of volume V around observation x. Let k_i be the number of samples labeled ω_i captured by the cell, then the joint probability can be estimated as:

$$P_n(x, \omega_i) = \frac{k_i / n}{V} \tag{3}$$

The above measure can be use to provide a reasonable estimate of the posterior probability:

$$P_n(\omega_i \mid x) = \frac{p_n(x, \omega_i)}{\sum_{i=1}^{M} p_n(x, \omega_i)} = \frac{k_i}{k} \tag{4}$$

where k is the total number of samples captured by the cell. Calculation of probability in this case is based on density estimation via Bayes theorem. Generally the knn rule suggests classifying x by assigning it to the class that appears most frequently among its k nearest neighbors. As we know, selecting ideal k neighbors in practical data is quite difficult, because only a finite amount of training data is available.

2.2 Aim of modification

Our goal for modification of the knn algorithm is to classify the data from a diabetes clinical study where we consider a two-class decision problem (positive or negative diagnosed). It is common to take $p(\omega_i|x)$ as a score to rank the k nearest neighbors region and prioritize elements in the test set. However, these scores do not always provide a reliable ranking. If we consider two regions *R1* and *R2*, one *R1* having k=100 neighbors where 99 of class ω_1 (in our case positive) probability is equal to p = 0.99 and *R2* having only k=1 neighbors, which belongs also to the same class ω_1 then probability will be p = 1. The score for first region *R1* is a much more reliable estimate of the true belonging to class ω_1, although, according to a p, the region *R2* should be ranked first. In this situation we see, that p is uncertain and we assume a P-value Rate (PR) based on hypergeometric distribution as probability criteria. Lam in his research [9] proposed the p-value as selection criteria in the context of prediction of diabetes type II with patients. In his study the goal was to classify entire points as positive or negative diagnosed patients. In our study we also tried to find positive (ω_1) diagnosed patients among negative. Suppose that patients are

distributed in the region *R*, and that then have a probability of *p* to be positive and *k* neighbors are also in this region. With the number of points which have to be positive, ω_1, we can define *p*-value as criteria for selecting the best *k*.

2.3 P-value Rate

Many problems in data mining require that we decide whether to accept or reject a statement about some parameter. The statement is called a hypothesis, and the decision-making procedure about the hypothesis is called hypothesis testing. Statistical hypothesis testing and confidence interval estimation of parameters are the fundamental methods used at the data analysis stage of a comparative experiment, in which someone is interested, for example, in comparing the mean of a population to a specified value. A statistical hypothesis is a statement about the parameters of one or more populations. Hypothesis can be defined as null hypothesis, when statement H_0:o=u or an alternative hypothesis, when statement H_1:o≠u where *o* is a single population and *u* is a specific constant. A procedure leading to a decision about a particular hypothesis is called a test of a hypothesis. Hypothesis-testing procedures rely on using the information in a random sample from the population of interest. If this information is consistent with the hypothesis, we will conclude that the hypothesis is true; however, if this information is inconsistent with the hypothesis, we will conclude that the hypothesis is false. One way to report the results of a hypothesis test is to state that the null hypothesis was or was not rejected at a specified α-value or level of significance. This statement of conclusions is often inadequate because it gives the decision maker no idea about whether the computed value of the test statistic was just barely in the rejection region or whether it was very far into this region. Furthermore, stating the results this way imposes the predefined level of significance on other users of the information. This approach may be unsatisfactory because some decision makers might be uncomfortable with the risks implied by α. To avoid these difficulties the *P*-value approach has been adopted widely in practice. The P-value is the probability that the test statistic will take on a value that is at least as extreme as the observed value of the statistic when the null hypothesis H_0 is true. Thus, a *P*-value conveys a lot of information about the weight of evidence against H_0, and so a decision maker can draw a conclusion at any specified level of significance. The P-value is the smallest level of significance that would lead to rejection of the null hypothesis H_0 with the given data. It is customary to call the test statistic (and the data) significant when the null hypothesis H_0 is rejected; therefore, we may think of the P-value as the smallest level at which the data are significant. Once the P-value is known, the decision maker can determine how significant the data are without the data analyst

formally imposing a preselected level of significance.

In probability theory and statistics, the hypergeometric distribution is a discrete probability distribution that describes the number of successes in a sequence of *n* draws from a finite population without replacement. Our example, using knn, is typical for a hypergeometric distribution theory: we have a collection of *N* objects in which *Na* are positive. The hypergeometric distribution describes the probability that, in a sample of *k* distinctive objects drawn from the collection, exactly *a* objects are positive.

Again for definition, let *N* be the total number of patients in a data set, and let *Na* be the number of training elements of class ω_1 (positive) in this data set. Consider a given neighbors region *R* in a given subspace, which has *k* training points, in which *a* belong to class ω_1 (positive). Suppose the *Na* is distributed such that they fall in or outside the given region at random. Under this statistical null hypothesis, the probability of observing *a* out of *k* training elements is given by the hypergeometric distribution:

$$P(a;k,Na,N)=\frac{\binom{N_a}{a}\binom{N-N_a}{k-a}}{\binom{N}{k}} \tag{5}$$

The PR (P-value rate) is the probability of having at least *a* positive elements out of *k*: PR = P($A \geq a \mid k$).

$$PR=P(A\geq a\mid k)=\sum_{i=a}^{\min(N_a,k)}\frac{\binom{N_a}{i}\binom{N-N_a}{k-i}}{\binom{N}{k}}=1-\sum_{i=0}^{a-1}\frac{\binom{N_a}{i}\binom{N-N_a}{k-i}}{\binom{N}{k}} \tag{6}$$

where *A* is a hypergeometric random variable. If the PR is small, there is little chance of seeing *a* out of *n* and vice versa. Therefore, small PR provide the most evidence against the null hypothesis of random allocation of *a* (positive) in the region (and hence most evidence that the number of elements labeled as ω_1 in the *k* is better than chance).

The PR method tends to pick *k* nearest neighbors with large numbers of patients even if they have a fairly low number of elements of class ω_1. Suppose we take 80 patients of class ω_1 in a data set of 782 objects of class ω_1 and ω_2 (number of all elements). Then 7 elements of class ω_1 out of 70 gives

PR=8.23x10^{-7} but 4 out of 4 (p = positive/negative=1.00) gives PR = 9.4x10^{-7} (Table 1). The statistical evidence is stronger in the first case because of the larger sample size, even though the *p* rate is much lower. This illustrates the major drawback of the *p* rate criterion.

Table 1. Probability and P-value Rate results for different numbers of neighbors in a region of the data set having 782 elements where 78 belongs to class ω1 (positive)

Region	k neighbors	Number of class ω_1 (positive) in k	$P(\omega_1\|x)$	RP	Ranking
1	78	7	7/78	8,42x10^{-7}	2
2	4	4	4/4	9,27x10^{-7}	3
3	98	98	98/100	8,53x10^{-7}	1
4	1	1	1/1	9,7x10^{-7}	4

3. Algorithm

In the traditional knn algorithm, the value of *k* is fixed beforehand. In practice, the value of *k* is usually optimalized by many trials on the training and validation sets using typically leave-one-out cross-validation. In cross validation we randomly split the set of labeled training samples into two parts: one is used as the traditional training set for adjusting model parameters in the classifier. The other set — the *validation set* — is used to estimate the generalization *validation set* error. Since our ultimate goal is low generalization error, we train our knn classifier until we reach a minimum of this validation error. In leave-one-out cross-validation we estimate the accuracy of the knn algorithm by training the classifier *n* separate times, each time using the training set from which a different single training point has been deleted. Each resulting classifier is tested on the single deleted point and the cross validation estimate of the accuracy is then simply the mean of these leave-one-out accuracies.

Choosing a single value of *k* for all data points may not be optimal as we claimed in previously. For example, if the density of points varies across the predictor space, a constant value of *k* implies neighborhoods of differing sizes. To deal with described problems of using static *k*, we propose a modified *k*-nearest neighbor algorithm based on PR as criteria, which uses different *k* values by examining local regions, rather than a fixed *k* value for all observed points. Our proposed method for calculation of the number of neighbors in region R_t (localization of *k* training elements around test point in feature space) for each test point n=1; 2; ; ; ; n_t is defined as:

$$k' = \arg\min_{k=1}^{m} PR(k) \tag{7}$$

where n is a test point, n_t is total number of test points, k is number of neighbors and m is the maximum number of neighbors to the test point calculated by leave-one-out cross validation. Generally we can describe the proposed algorithm very simply:

- Iteration for all test points from n = 1; 2; ; ; ; ; n_t
  a. Estimate maximum number of neighbors m for the test point using leave-one-out cross-validation
  b. Iteration for different k neighbors values from k = 1; 2; ; ; ; m
     i. Find k neighbors to test point n in the training set
     ii. Calculate the probability of observing a_k positive in region R_t out of all (positive/negative) k training elements using hypergeometric distribution P (a_k, k, Na, N) (5), where a is a number training elements of class ω_1 (positive) elements in region R_t, k is a number training elements of class ω_1 (positive) and ω_2 (negative) in region R_t, N be the total number of patients in a data set, Na be the number of training elements of class ω_1 (positive) in this data set
     iii. Based on the formula (6) calculate for region R_t p-value ratio PR(k) from P(a, k, Na, N)
     iv. Note best k' value for test point which give minimum PR(k) (7)

Summarizing, for each test point n, the method looks from 1 to m nearest neighbors at the same time, and finds the value k' that reaches the smallest PR(k). Next we will compare the performance of the proposed algorithm together with traditional knn.

3.1 Algorithm Testing

A probability distribution is similar to the frequency distribution of a quantitative population because both provide a long-run frequency for outcomes. In other words, a probability distribution is listing all the possible values that a random variable can take along with their probabilities. A distribution is called discrete if its cumulative distribution function consists of a sequence of finite jumps, which means that it belongs to a discrete random variable Z: a variable which can only attain values from a certain finite or countable set. One of the most widely known of all discrete probability distributions is the hypergeometrical distribution. In the proposed algorithm, to avoid the problem of selecting the number of

neighbors we used the distribution that describes exactly a situation as in the knn method where atotal number of N elements exist in which N_a are of class ω_1. The hypergeometric distribution describes the probability that, in a sample of n selected objects (region with k elements) drawn from the training data, exactly a objects are positive. Probability of success in this distribution is calculated not only localy arround the test point but also looking at the entire collection of data N. Looking at a complete set of balanced data, where in features space in one region we have more elements then in other and also a different number of elements labeled as ω_1, proposed criteria give us better results.

Is our new criteria PR(k) significantly better then $p(\omega_i|x)$? To check it we provided two sided hypothesis testing. For our test we use a k' number of training elements calculated from (7) and k_P calculated from the traditional knn method:

$$k_p = \arg\max_{k=1}^{m} \mathrm{p}(\omega_1 \mid \mathrm{x_k}) \tag{8}$$

We use H_0 to represent the null and H_A to represent the alternative hypothesis. We are interested in whether there has been a different number of optimal locale selected nearest neighbors elements using traditional criteria in knn and using proposed criteria $PR(k)$ in a region around the test element. Alternative hypothesis and the null hypothesis are defined by:

$$H_o : k_p = k' \quad vs \quad H_A : k_p \neq k' \tag{9}$$

For testing hypotheses about a single proportion we applied a test statistic:

$$z = \frac{k' - k_p}{\sqrt{k'(1-k')/k}} \tag{10}$$

where n is number of training elements.

When carrying out a hypothesis test we usually use the P-value to decide whether or not the null hypothesis should be rejected which is given by:

$$P - value \quad P(Z \geq |z|) \tag{11}$$

where Z is a random test statistic calculated from a data set produced by the null distribution. The smaller the P-value the more sure we feel about rejecting the null hypothesis and assuming that our criteria is statistically better. We can view the P-value as a measure of the strength of the evidence against the null hypothesis (and for the alternative). For example, a P-Value of 0.1 would be considered (very) weak

evidence, 0.05 would be evidence, 0.01 strong evidence and .001 very strong evidence.

3.2 Performance Measuring

There are many possible ways to assess the prediction error of a knn classifier with accuracy, precision, and recall probably being the most widely used measures. As we built our model to predict positive diagnosed we can also use Average Rate (AR). The average rate (proportion of positive diagnosed amongst those selected) is other popular measure (e.g., Tatsuoka, Gu, Sacks, and Young 1998) [10] for evaluating the predictive performance of classification models. The definition of AR is equivalent to average "precision" [11], which is a common measurement in data mining. The AR of the data set having n elements is given by:

$$AR = \frac{\sum_{i=1}^{n} \left(\frac{\sum_{j=1}^{i} y(j)}{i} \right)}{A}, \tag{12}$$

where:

$$\begin{aligned} A &= \sum_{i=1}^{n} y(i) \quad \text{(the number of actives in the list)} \\ y_{(i)} &= \begin{cases} 1 & \text{if the (i) -th compund is active} \\ 0 & \text{if the (i)-th compound is inactive} \end{cases} \end{aligned} \tag{13}$$

The next point is to compare the knn algorithm with our modified version on real patient data using AR and common measurement criteria for classifier evaluation. Both of the processes of sequentially selecting *k* and AR as a performance measurement make the algorithm hard to translate from a theoretical point of view, while implementation is relatively simply.

4. Experiments

For our purpose of illustrating the usage of modified knn based on PR criteria (kPR) and comparing to the traditional knn approach we used Pima Indians Diabetes Dataset [15] which consists of nine features for each patient (sample). Applying these features, type II diabetes is possible to diagnose via classifier. The first eight features are inputs, and feature IX is the output. Feature IX values are zero and one; zero indicating the Normal, and one, the Sick. For data classification, feature IX is classified into two classes of Normal and Sick.

Table 2. Information about the Pima Indians Diabetes Data Set

	Total	Training data	Test data
Patients	782	350	432
Positive	62 (8%)	28	34
Negative	720	322	398

For algorithm estimations we have divided both stage 0 and stage 1 datasets into training and test datasets. Both knn with leave-one-out cross-validation for *k* and modified knn algorithm are applied to the training set and predictions on the test set. From the test set we have generated four subsets and calculated an average hit rate (Table 3). Based on AR we could observe that in both datasets kPR performed better than normal knn. To determine whether the improvement of knn is significant or not, we conducted a statistical test. The p-value calculated from the statistical test suggests the AR of the kPR algorithm is statistically significantly larger than knn, on average.

Table 3. AR for kPR and traditional knn in stage 0

i-th split	AR(knn)	AR(kPR)	Difference
1	0.2382	0.2459	0.0003
2	0.1943	0.2411	0.028
3	0.1910	0.2323	0.0393
4	0.2238	0.2557	0.0359
Mean d = 0.0259			
Standard deviation sd = 0.018			
$t = \frac{\bar{d} - 0}{s_d / 4} = 2.71$ (p-value=0.031)			

5. Conclusion

The main difference between the modified knn rule and the original knn lies in the fact that the actual value of *k* at each query point varies, depending on the region where the test point is lying, while in the knn rule, the value of *k*, once set, is the same for all query points in the feature (Burden descriptors) space. For different regions, according to test point and neighbors in the training set, we used a suitable number

of nearest neighbors to predict the class of a test document. Preliminary experiments on classification diabetic datasets show that our method is less sensitive to parameter *k* than the traditional one, and it can properly classify elements as positive by selecting *k* for each test point individual. The method is promising for some special cases where estimating the parameter *k* via cross-validation is not effective.

It should be noted that the modified knn rule differs significantly from previous methods that have been developed for adapting neighborhoods in the knn rule, such as the flexible metric method by Friedman [7], the discriminant adaptive method by Hastie and Tibshirani [8], and the adaptive metric method by Domeniconi et. al [14]. Despite the difference in their approaches, the common idea of the method is that they estimate feature relevance locally at each query point and compute a weighted metric for measuring the distance between a query point and the training data.

We plan to experiment with our improved method on more different medical data sets in the future and also same datasets with different patient parameters. Even though we have tested the method only on diabetes data, the method should be universally applicable to classification problems for clinical study data with administrative and clinical patient parameters.

References

[1]. Klopman, G. 1984., Artificial Intelligence Approach to Structure-Activity Studies. Com-puter Automated Structure Evaluation of Biological Activity of Organic Molecules, American Chemical Society, Vol. 106, No. 24, 7315-7321.

[2]. D. Hand, H. Mannila, P. Smyth.: Principles of Data Mining. The MIT Press. (2001)

[3]. Yang Y. and Liu X., 1999. A Re-examination of Text Categorization Methods [A]. In: Proceedings of 22nd Annual International ACM SIGIR Conference on Research and De-velopment in Information Retrieval [C]. 42-49.

[4]. Joachims T., 1998. Text Categorization with Support Vector Machines: Learning with Many Relevant Features [A]. In: Proceedings of the European Conference on Machine Learning [C].

[5]. Li Baoli, Chen Yuzhong, and Yu Shiwen, 2002. A Comparative Study on Automatic Categorization Methods for Chinese Search Engine [A]. In: Proceedings of the Eighth Joint International Computer Conference [C]. Hangzhou: Zhejiang University Press, 117-120.

[6]. Gregory A Landrum, Julie E Penzotti and Santosh Putta,Machine-learning models for combinatorial catalyst discovery. Rational Discovery LLC, 555 Bryant St 467, Palo Alto, CA 94301, USA

[7]. Friedman, J., Flexible metric nearest neighbor classification, Technical Report 113, Stanford University

Statistics Department (1994)

[8]. Hastie, T., Tibshirani, R., Discriminant adaptive nearest neighbor classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 18 (1996) 607-615

[9]. Raymond L.H. Lam., Cell-Based Analysis of High Throughput Screening Data for Drug Discovery. Biomedical Data Sciences. 2002

[10]. Tatsuoka, K., Gu, C., Sacks, J., and Young, S. S. 1998, Predicting extreme values in large datasets, Technical Report, National Institute of Statistical Sciences.

[11]. Harman, D. K. (ed.) 2000. Overview of the Eighth Text Retrieval Conference (TREC- 8), Appendix, NIST Special Publication, A1. http://trec.nist.gov/pubs/trec8/appendices/A/appendixa.cover.pdf

[12]. National Cancer Institute URL: http://dtp.nci.nih.gov/yacds/

[13]. Pearlman, R. S. and Smith, K. M. 1998, Novel software tools for chemical diversity, Perspectives in Drug Discovery and Design, 9/10/11, 339-353.

[14]. Domeniconi, C., Peng, J., Gunopulos, D., Locally adaptive metric nearest-neighbor classi-fication. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (2002) 1281-1285

[15]. Pima Indians Diabetes Data Set, http://mlr.cs. umass.edu/ml/datasets/Pima+Indians+Diabetes [Last Available: April 2015].

Published: Volume 2017, Issue 1 / January 25, 2017