Scalable Non-Parametric Methods for Large Data Sets

V. Suresh Babu
Indian Institute of Technology-Guwahati, India

P. Viswanath
Indian Institute of Technology-Guwahati, India

M. Narasimha Murty
Indian Institute of Science, India

INTRODUCTION

Non-parametric methods like the nearest neighbor classifier (NNC) and the Parzen-Window based density estimation (Duda, Hart & Stork, 2000) are more general than parametric methods because they do not make any assumptions regarding the probability distribution form. Further, they show good performance in practice with large data sets. These methods, either explicitly or implicitly estimates the probability density at a given point in a feature space by counting the number of points that fall in a small region around the given point. Popular classifiers which use this approach are the NNC and its variants like the k-nearest neighbor classifier (k-NNC) (Duda, Hart & Stock, 2000). Whereas the DBSCAN is a popular density based clustering method (Han & Kamber, 2001) which uses this approach. These methods show good performance, especially with larger data sets. Asymptotic error rate of NNC is less than twice the Bayes error (Cover & Hart, 1967) and DBSCAN can find arbitrary shaped clusters along with noisy outlier detection (Ester, Kriegel & Xu, 1996).

The most prominent difficulty in applying the non-parametric methods for large data sets is its computational burden. The space and classification time complexities of NNC and k-NNC are $O(n)$ where n is the training set size. The time complexity of DBSCAN is $O(n^2)$. So, these methods are not scalable for large data sets. Some of the remedies to reduce this burden are as follows. (1) Reduce the training set size by some editing techniques in order to eliminate some of the training patterns which are redundant in some sense (Dasarathy, 1991)**.** For example, the condensed NNC (Hart, 1968) is of this type. (2) Use only a few selected prototypes from the data set. For example, *Leaders-subleaders* method and *l*-DBSCAN method are of this type (Vijaya, Murthy & Subramanian, 2004 and Viswanath & Rajwala, 2006). These two remedies can reduce the computational burden, but this can also result in a poor performance of the method. Using enriched prototypes can improve the performance as done in (Asharaf & Murthy, 2003) where the prototypes are derived using adaptive rough fuzzy set theory and as in (Suresh Babu & Viswanath, 2007) where the prototypes are used along with their relative weights.

Using a few selected prototypes can reduce the computational burden. Prototypes can be derived by employing a clustering method like the leaders method (Spath, 1980)**,** the *k*-means method (Jain, Dubes, & Chen, 1987), *etc.,* which can find a partition of the data set where each block (cluster) of the partition is represented by a prototype called leader, centroid, *etc*. But these prototypes can not be used to estimate the probability density, since the density information present in the data set is lost while deriving the prototypes. The chapter proposes to use a modified leader clustering method called the *counted-leader* method which along with deriving the leaders preserves the crucial density information in the form of a *count* which can be used in estimating the densities. The chapter presents a fast and efficient nearest prototype based classifier called the *counted k-nearest leader classifier (ck-NLC)* which is on-par with the conventional k-NNC, but is considerably faster than the k-NNC. The chapter also presents a density based clustering method called *l*-DBSCAN which is shown to be a faster and scalable version of DBSCAN (Viswanath & Rajwala, 2006). Formally, under some assumptions, it is shown that the number of leaders is upper-bounded by a constant which is independent of the data set size and the distribution from which the data set is drawn.

BACKGROUND

Supervised learning and *unsupervised learning* are two main paradigms of learning. Supervised learning refers to learning with a teacher, typically in situations where one has a set of training patterns whose class labels are known. The objective is to assign a label to the given test pattern. This is called pattern classification. On the other hand, unsupervised learning or learning without a teacher refers to situations where training patterns are not labeled and the typical objective is to find the natural grouping (or categories) among the given patterns. This is called pattern clustering (Jain, Murty & Flynn, 1999). Among various classification and clustering methods non-parametric methods are those which either explicitly or implicitly estimates the arbitrary density function from the data sets based on which classification or clustering tasks can be performed. Prominent non-parametric classifiers are NNC and k-NNC. Whereas DBSCAN is a popular non-parametric density based clustering method.

NNC works as follows. Let $\{(X^1, y^1),\ldots,(X^n, y^n)\}$ be the training set where y^i is the class label for the pattern X^i, for $1 \leq i \leq n$. For a given test pattern T, Let X^l be the nearest neighbor in the training set based on the given distance measure, then NNC assigns the class label of X^l *(i.e.,*y^l*)* to T. An extension of the above method is to find k nearest neighbors of the test pattern and assigning the class label to the test pattern based on a majority vote among the k neighbors. Assuming that k is a small constant when compared with n, the time required to classify a pattern, either by NNC or by k-NNC, is $O(n)$.

Density based clustering methods like DBSCAN groups the data points which are dense and connected into a single cluster. Density at a point is found non-parametrically. It is assumed that probability density over a small region is uniformly distributed and the density is given by m/nV, where m is the number of points out of n input data points that are falling in a small region around the point and V is the volume of the region. The region is assumed to be a hyper sphere of radius ε and hence threshold density can be specified by a parameter *MinPts,* the minimum number of points required to be present in the region to make it dense. Given an input dataset D, and the parameters ε and *MinPts,* DBSCAN finds a dense point in D and expands it by merging neighboring dense points. Patterns in the data set which do not belong to any of the clusters are called noisy patterns. A non dense point can be a part of a cluster if it is at distance less than or equal to ε from a dense pattern, otherwise it is a noisy outlier (Viswanath & Rajwala, 2006). The time complexity of DBSCAN is $O(n^2)$.

Many non-parametric methods suffer from the huge computational requirements and are not scalable to work with large data sets like those in data mining applications.

MAIN FOCUS

Using only a few selected prototypes from the data set can reduce the computational burden of the non-parametric methods. The prototypes need to be rich enough to compute the probability density at an arbitrary point in the feature space by using them. First, *counted-leader* method is described followed by *counted k-nearest leader classifier* and a hybrid density based clustering method which uses the counted prototypes in place of the large training data set. Finally, some experimental results are given in support of the methods in this chapter.

Counted-Leader Method

Counted-leader method which is a modified leader clustering method scans the database only once and is an incremental method. It has running time that is linear in the size of the input data set. More precisely, it can find a partition of the data set in $O(n)$ time where n is the data set size. It derives prototypes and also a count for each prototype which indicates the prototypes relative importance. The count of a prototype represents the number of patterns falling under that prototype. Conventional leader method (Spath, 1980) derives the prototypes called the leaders set. These leaders represent the semi-spherical clusters as shown in the Figure 1, and are more or less uniformly spread over some regions of the feature space. Hence by using the leaders alone it is not possible to find the probability density at a point whereas the counted leaders can be used to estimate the density at a point.

For a given threshold distance t, the counted-leader method works as follows. It maintains a set of leaders L, which is initially empty and is incrementally built. For each pattern x in D, if there is a leader l in L such that distance between x and l is less than t, then x