Density-Based Evolutionary Outlier Detection

Amit Banerjee
School of Science, Engineering and Technology
Penn State University, Harrisburg
Middletown, PA
1-717-948-6661
aub25@psu.edu

ABSTRACT
A novel density-based distance measure and an outlier detection method using evolutionary search are presented in this paper. A fitness function based on nearest neighbor distances is proposed and the genetic recombination operators are designed to achieve a balance of exploration and exploitation in the nearest neighborhood space. The methodology is tested on datasets of varying sizes (small to moderate) and dimensionalities and performance is compared to existing evolutionary methods for outlier detection.

Categories and Subject Descriptors
H.2.8 [**Database Applications**]: Data Mining

General Terms
Algorithms

Keywords
Outlier Detection, data density, genetic algorithm, nearest neighbor distance

1. INTRODUCTION
Identifying outliers is central to data mining activities such as credit card fraud detection, voting irregularity analysis, and network intrusion among others where the outlier is an object of interest. This is in contrast to supervised and unsupervised classification tasks where outliers impede system performance and should therefore be removed or ignored. In either case, detection of outliers is of paramount importance. The task of outlier detection in data can be categorized into three groups – (1) statistical methods which assume *a priori* knowledge of the distribution of the data and outliers are defined as data objects that do not belong to the distribution, (2) deviation-based methods (also called spatial methods) which identify outliers as those objects whose spatial attributes deviate significantly from the main characteristics of the data, and (3) the most popular distance-based methods which identify outliers simply according to distances to nearest neighbors. In this approach, an object in a dense region of data will be close to its neighbors while a potential outlier will have relatively fewer points in its immediate neighborhood. The advantages of defining outliers using distances are that no explicit statistical distribution needs to be defined and that they can be applied to any feature space for which a distance measure can be defined [2]. An obvious disadvantage of using nearest-neighbor based techniques is the computational burden – a simple nested loop will be of quadratic complexity. Researchers have proposed improved data structures, data preprocessing and indexing techniques to deal with this issue [3]. In this paper, an improved density-based distance measure is proposed to be used with an evolutionary search algorithm for outlier detection. The density-based distance measure will be shown to reduce the number of pairwise computations needed to identify an outlier. A novel recombination operator that exploits the density-related information is also proposed to be used with the evolutionary algorithm.

GECCO'12 Companion, July 7–11, 2012, Philadelphia, PA, USA.
ACM 978-1-4503-1178-6/12/07.

2. METHODOLOGY
Consider three objects in a two-dimensional data: A (0, 0), B (1.5, 2.6) and C (3, 0) such that C is in a sparse region of the data. The pairwise Euclidean distances between the three objects is the same (= 3 units), and unless distance computations with all other objects in the dataset are performed, C will never be identified as a potential outlier. However, if data density around the objects is considered the distances AC and BC can be weighted so as to reflect the fact that C's attribute values (3, 0) are in a sparse region of the data. The distances between objects is defined such that for each attribute, the distance is made up of two parts – a simple Euclidean distance that measures how far in the feature space the objects are, and a density-based component based on Lancaster's modified mean valued χ^2 transformation [7] inversely related to the immediate density of the attribute, which relates directly to the observed frequency of the feature (in case of categorical or nominal features) or the frequency of the bin (in case of continuous numerical features). Another advantage of this modified distance measure is attribute distances for categorical attributes can be combined with those for continuous attributes.

2.1 Representation
An n-integer representation is used where n is the cardinality of the dataset (every object is labeled from 1 to n). The value of the i^{th} gene represents the nearest neighbor of the i^{th} object. The nearest neighbor is defined in terms of the modified distance measure that combines a Euclidean measure with a density-dependent measure of distance. As an example for n = 5, an individual [21235] decodes to a solution where object labeled 2 is the nearest neighbor of objects 1 and 3, and object 1 is the nearest neighbor of object 1, object 3 is the nearest neighbor of object 4 and object 5 is its own nearest neighbor. As will be described later, genes which self-refer themselves (such as the object 5 being its own nearest neighbor) will be prohibited by a mandatory mutation. The idea is to generate an integer string that captures nearest neighbor information as effectively as possible. As can be seen this representation is unwieldy for very large datasets. For such datasets a representation similar to the one in [5] can be used where the size of the integer chromosome is variable. A chromosome encodes for k outliers where k varies from individual

to individual in the population and the i^{th} gene in the chromosome to the label of the i^{th} outlier.

2.2 Fitness Function

A chromosome is assigned a fitness value inversely proportional to the sum of nearest neighbor distances encoded in it. If a chromosome encodes for the optimal (or a near-optimal) solution, the sum of nearest neighbor distances will be less than one that does encodes for a sub-optimal solution. However, to be valid, an object should not be allowed to refer to itself as its nearest neighbor. If k outliers are sought, the n nearest neighbor distances in a fit individual can be sorted and the genes with the largest k distances can be identified as the outliers.

2.3 Recombination

A modified uniform crossover operator with a predefined probability is used as the primary source of variation in the evolving population. For two parents selected for crossover, the crossover operator compares nearest neighbor distances for each position, and the offspring inherits the allele which represents the smaller distance. For example if [21234] and [34151] are selected for crossover, the crossover operator compares if the object labeled 1 is closer to object 2 or object 3. If object 1 is closer to object 2 than it is to object 3, then the offspring chromosome will inherit 2 as the allele for the 1st position, and so on. A mutation operator is also used to mutate positions with a predefined probability as a secondary source of variation. In addition the mutation operator is also used in a corrective mode – to modify an allele if a self-reference is found. In order to amplify the exploitation component of the search, a few experiments were conducted with the mutation operator switched off after a certain number of generations.

2.4 Initial Population

An initial population is created randomly by assigning allele values to genes (while explicitly avoid self-referencing). Another method initialization method investigated for moderately-sized datasets (higher dimensional datasets) is clustering using k-means to identify clusters and then assigning allele values to individuals in the initial population from its own k-means cluster. A simple Euclidean measure is used as the distance function with the initial k-means step. Although, this adds complexity to the algorithm, it is shown to speed up convergence and improve performance.

2.5 Selection and Evolution

A mating pool is created by using a fitness-proportional selection operator. Parents are selected at random for crossover from the mating pool and an offspring population of half the size of the parent is created. The new population is created by retaining the top half (fitness) of the existing parent population and replacing the bottom half by the newly created offspring population. The evolution is continued till a fixed number of generations or until the average fitness of the population stabilizes (which happens quickly in the absence of mutation later on in the evolution process).

3. RESULTS

In the first series of experiments, five benchmark datasets from statistical outlier detection literature are tested to tune algorithm parameters. The performance is compared to a Least-Squares (LS) error-based genetic algorithm for outlier detection as reported in [5]. It was found that when the initial population was created using k-means, the average number of iterations to convergence reduced by a factor of 1.5-2.0. In almost all the runs, the mutation operator was switched off completely after a fixed number of generations. Two synthetic datasets (n = 1600) of different dimensionalities are also tested to validate parameters identified in the first series of experiments. Both datasets include objects generated from three Gaussian distributions with uniformly distributed outliers. The outliers occupy sparse regions in a two dimensional plane (first dataset) and in 12-dimensions (second dataset). In the third series of experiments, two actual datasets – the yeast dataset from UCI Machine Learning Repository [6], and 2010 traffic volume data for major California highways [4]. The yeast dataset has 1484 instances in 6-dimensions while the highway dataset has 7089 instances (after checking for missing values) in 4-dimensions. In the yeast data, the algorithm was able to identify the most unrepresentative objects of the 10 yeast groups for small values of k. In the highway data, the algorithm consistently identified 6-7 outliers, which are believed to be erroneous measurements in the data. Complete results on all datasets will be reported in the detailed version of this paper [1].

4. DISCUSSION AND FUTURE WORK

A novel density-based distance measure for use with an evolutionary search algorithm for outlier detection in moderately sized data is investigated. The algorithm is substantiated with experiments on datasets of varying degree of complexity and initial results are encouraging. Future work includes developing a methodology to automatically identify the right number of outliers (k) from the components of the fitness formulation. For k outliers, the gradient of the k^{th} nearest neighbor distance measured with respect to the $(k–1)^{th}$ distance will be the largest for uniformly distributed outliers in normally distributed data. This aspect will be investigated in subsequent work. The potential of a varying k-integer representation will also be investigated.

5. REFERENCES

[1] Banerjee, A. Distance measures for outlier detection in mixed feature data, *unpublished*.

[2] Bay, S. D., and Schwabacher, M. Mining distance-based outliers in near linear time with randomization and a linear pruning rule. In *Proceedings of the SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington D.C., Aug 24-27, 2003). ACM Press, New York, NY, 2003, 29-38.

[3] Berchtold, S., Keim, D., and Kreigel, H. –P. The X-tree: an index structure for high-dimensional data. In *Proceedings of the 22nd International Conference on Very Large Databases* (Bombay, India, Sep 3-6, 1996). Morgan Kauffmann, San Francisco, CA, 1996, 28-39.

[4] California 2010AADT Data [http://traffic-counts.dot.ca.gov]. California Department of Transportation (2012).

[5] Crawford, K. D., and Wainwright, R. L. Applying genetic algorithms to outlier detection. In *Proceedings of the 6th International Conference on Genetic Algorithms* (Pittsburgh, PA, July 15-19). Morgan Kaufmann, San Francisco, CA, 1995, 546-550.

[6] Frank, A., and Asuncion, A. UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Univ of California, Irvine, CA: Computer Science (2010).

[7] Lancaster, H. O. The combining of probabilities arising from data in discrete distributions, *Biometrika 36*, (1949) 370-382.