International Journal of Grid and High Performance Computing
Volume 16 • Issue 1

An Approach for Detecting Local Outliers in Grid Queries

Shuang Li, Hunan International Economics University, Changsha, China
Xiaoguo Yao, Hunan International Economics University, Changsha, China*

ABSTRACT

The density local outlier factor algorithm (LOF) needs to calculate the distance matrix for k-nearest neighbor search. The algorithm has high time complexity and is not suitable for the detection of large-scale data sets. A local outlier detection algorithm is proposed based on grid query (LOGD). In the algorithm, the k other data points closest to the data point in the target grid must be in the target grid or in the nearest neighboring grid of the target grid, it is used to improve the neighborhood query operation of the LOF algorithm, the calculation amount of the LOF algorithm is reduced in the neighborhood query. Experimental results show that the proposed LODG algorithm can effectively reduce the time of outlier detection under the condition, the detection accuracy of the original LOF algorithm is basically the same.

KEYWORDS

Distance matrix, Grid, k-nearest neighbors, Local outlier factor (LOF), Memory, Neighborhood query

1. INTRODUCTION

Outlier detection is one of the basic tasks of data mining. Its purpose is to eliminate noise or discover potential and meaningful knowledge*(Li X, Lv J, Yi Z., 2018)*. After long-term development, outlier detection has been widely used in fraud detection, intrusion detection, and abnormal natural climate discovery.

Outlier detection can be roughly divided into five categories based on distribution, bias, clustering, distance, and density*(Peng T. & Yang N. Y.,2018; Zhu L. & Qiu Y. Y.,2017; Xu H. L. & Tang S., 2017; Marateb H R & Rojas-Martínez M,2012)*. The representative density-based algorithm is the LOF algorithm *(Breunig M M & Kriegel H P, 2000)*, local outlier factor (*lof*) is used to represent the degree of local outlier of an object. The larger the *lof* value, the higher the degree of outlier of the data. However, the LOF algorithm also has huge defects, that is, it needs to calculate the distance matrix to determine the k nearest neighbor distance, but a large amount of memory consumption makes outlier detection of large-scale data sets unacceptable. In response to this problem, Lee J et al. used a grid to divide the data *(Lee J & Cho N W, 2016)*, and then calculated the local outliers between the centroid points of each grid. Zhang J. et al. proposed a dense grid method *(Zhang J.*

DOI: 10.4018/IJGHPC.336474 *Corresponding Author

& Sun Z. H., 2011), the data in the dense cell grid was eliminated first, the *lof* value of each data is calculated in the remaining data. Although the above two algorithms speed up the calculation, the spatial distribution characteristics of the data is not considerd, and the detection accuracy rate needs to be improved. Rundensteiner E A proposed a fast branch reduction strategy TOLF that does not need to calculate *knn (Yan Y Z & Cao L, 2017)*. This method performs well on multiple data sets. Kim D et al. proposed the DILOF algorithm *(Na G S & Kim D, 2018)*. Inthis method, a new sampling method is used to avoid data distribution assumptions, while new distance approximation techniques are use to speed up detection. In addition, some scholars have proposed clustering to exclude non-outliers before performing outlier detection *(Yin N. & Zhang L.., 2017; Cao K. Y. & Luan F. J.,2017; Wang J. H. & Jin P., 2015)*, adding information entropy to determine attribute weights, and reducing the impact of attribute differences on outlier detection*(Wang J. H. & Zhao X. X.,2013; Xin L. L. & He W., 2015; Hu C. P. & Qin X. L., 2010)*, using parallel computing *(Bai M & Wang X,2016; Wang X. T. & Shen D. R., 2016)*, introduction of square neighborhoods *(Jie C. M. & Liu H. J.,2012)*, introduction of rough sets*(Yuan Z. & Feng S., 2018)*, and use of average density *(Zhou P. & Cheng Y. Y., 2017)*.

Although many published algorithms have been proposed in recent years to improve the LOF, these algorithms improve the detection efficiency by reducing the number of data points or attributes, and the influence of the filtered data points is ignored on outliers. Aiming at this problem, in this paper, an outlier detection algorithm LOGD is proposed based on grid query. The data is mapped to the grid, and the grid can remember the relative position information between the data points. By querying the adjacent grid of the target grid, the k nearest neighbors of the data points are queried in the target grid, the distance calculations between data are reduced in the LOF algorithm when performing k-nearest neighbor query, and the local outlier value of each data is finally calculated. This algorithm can reduce the amount of calculation and improve the detection speed, while still having the same accuracy rate as the original LOF algorithm.

2. BASIC CONCEPTS OF THE LOF ALGORITHM

Local outlier factor (LOF) algorithm is a popular method for outlier detection in data mining. It is based on the concept of local density and compares the density of an instance to its neighbors to identify outliers *(Breunig M. M.,2000; Tang J., 2015; Wen J.,2013; Saeed H.,2017)*.

The basic idea behind LOF is that outliers are instances that have significantly lower density compared to their neighbors. LOF measures the degree of outlierness for each instance by examining the local density ratio. It calculates the LOF score for an instance by comparing its local reachability density (LRD) to the LRDs of its k-nearest neighbors. If an instance has a much lower LRD than its neighbors, it is considered an outlier.

Definition 1: (*k*-th distance of object *p: k-distance (p)*). For any natural number *k*, dataset *D*, define the *k*-th distance of *p* as the distance between *p* and some object *o*, and record it as *k- distance (p)*, where the object *o* meets the following conditions:

(1) At least *k* objects $o' \in D\backslash\{p\}$ satisfy: $d(p, o') \leq d(p, o)$
(2) At most *k-1* objects $o' \in D\backslash\{p\}$ satisfy: $d(p, o') < d(p, o)$

Where the distance between object *p* and object *o* is written as *d (p, o)*.

Definition 2: (*k*-th distance neighborhood of object *p:* $N_k(p)$) . Given the *k-distance(p)* of data object *p*, the *k*-th distance neighborhood of object *p* is the set of data points that its distance from *p* does not exceed *k-distance(p)*:

$$N_k(p) = \{q \mid d(p, q) \leq k\text{-distance}(p)\} \quad (1)$$

Computational Grids: An Introduction to Potential Biomedical Uses and Future Prospects in Oncology; Neuro-Oncology Applications as a Model for Cancer Sub-Specialties

Ribhi Hazin, Ibrahim Qaddoumiand Francisco Pedrosa (2012). *Grid and Cloud Computing: Concepts, Methodologies, Tools and Applications (pp. 141-152).*
www.irma-international.org/chapter/computational-grids-introduction-potential-biomedical/64482