The Local Outlier Data Mining Algorithm Based on MapReduce and the Associated Subspace

Hong Chen[1, a], **Wenjiao Wang**[2,b*]

[1]University of California San Diego, Probability& Statistics, United States

[2]Guangzhou University, Guangzhou Developmental Academy, China

[a,b]venuswenjiao@126.com

Keywords: Mapreduce; mass data; classification increment mining

Abstract. With the continuous appearance of incremental data in mass data, it is necessary to update mining results. The existing incremental data mining methods may be not suitable for incremental mining of mass data. By aiming this problem and combining MapReduce with the associated classification mining.

1.Introduction

The study of mass data mining is concentrated on the MapReduce treatment of single-node mass data, but there are few research achievements about MapReduce mining by aiming at the distributed mass data. Moreover, it has the low efficiency. Lots of mass data are distributed. How to combine MapReduce with the associated classification mining by aiming at mass data features is the key study to realize mass data mining.

2. The Associated Substance and Outlier in Data

Set up the dataset: DS is the d-dimensional dataset of a data object obj; the data object obj_i stands for the ith data object. The attribute dimension stands for ith attribute dimension, $x_{ij}(i=1,2,..n;j=1,2,3..d)$ represents the value of ith data object obj_i on jth attribute. The dataset LDS is the K neighbor set of any data object obj_i in the dataset DS, shown as follows: the dataset $DS=\{obj_1,obj_2,obj_i,...obj_n\}$, K neighbor set $LDS=\{obj_1,obj_2,obj_3,...obj_k\}$, and relevant concept description of the attribute set $FS=\{A_1,A_2,...A_d\}$ is shown as follows: set up ε as the local sparse difference threshold, dij is the local sparse difference factor of ith data object obj on the jth attribute. If $d_{ij}<\varepsilon$, thus let $v_{ij}=0$, vice versa, let $v_i=\{v_{i1},v_{i2},v_{ij},...v_{id}\}$, which is the subspace definition vector of obj. In the vector v, it is the subspace constituted by the attribute dimensional set with the value of $v_{ij}=1$, called as the associated subspace of obj_i. The value of $v_{ij}=0$ forms the subspace constituted by the attribute dimension. The outlier of the outlier data obj on the associated subspace is defined as follows:

Factor(obj)=max

$$\begin{cases} 0, & erf\left(\dfrac{POLF_{RS}(obj)}{\sqrt{2}.\sqrt{Eo \in LDS(obj) \cup obj[(PLOF_{RS}(O))^2]}}\right)\} \end{cases} \quad (1)$$

In the Formula(1), $PLOF_{RS}$ (obj) stands for the local anomaly factor of probability in the associated subspace RS(erf:error function or Gauss error function).

A. The Outlier Data of the Context

The associated subspace is composed of the attribute dimension set with non-uniform distribution. The outlier factor value of data objects is measured in the associated subspace, thus the associated subspace provides some valuable information.

Set up the vector: $v_i =\{ v_{ij} \}$ as the subspace definition vector of the data object, where i=1,2…|DS|, j=1,2,... ,d, called as the outlier factor Factor(obj) and vector: the context information of obj with the attribute dimension set SS with v_i=1. The outlier data with the context information is called as the context outlier data. In the Formula(1), $PLOF_{RS}$(obj) and $PLOF_{RS}$(O)(O∈LDS(obj)) stand for the local outlier degree of obj and O on the associated subspace(LDS: Linear Data Set). After PLOF value is translated into the probability value, the outlier factor(obj) and attribute dimension set SS constitute the context information to describe the outlier degree of outlier data object and attribute dimension information of non-uniform distribution.

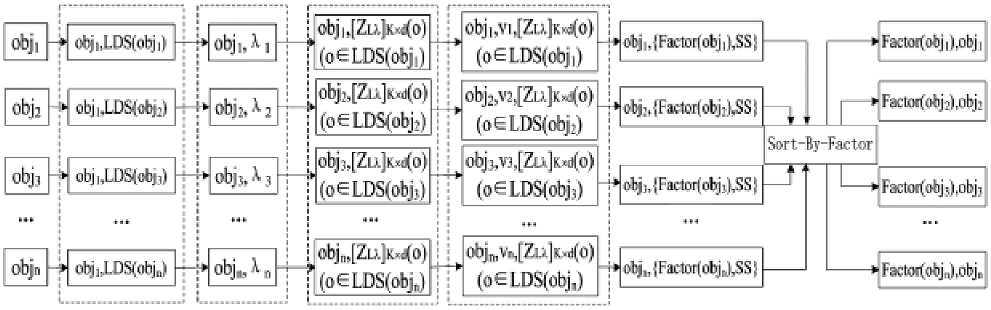
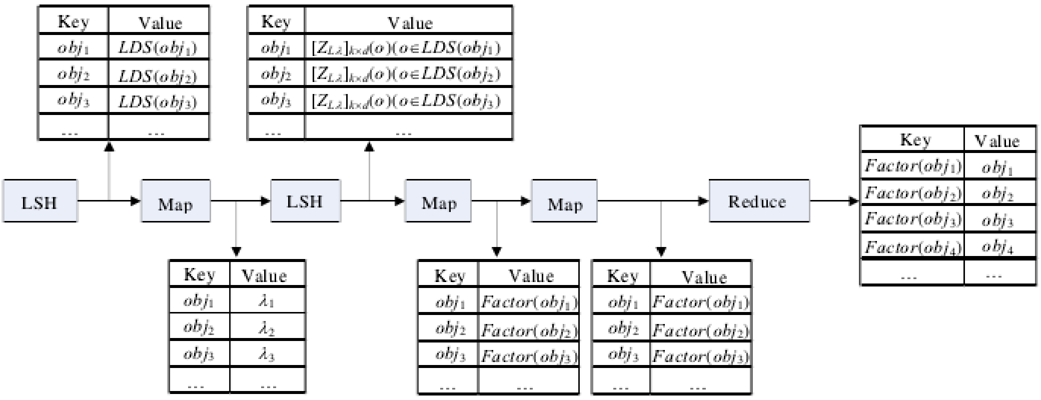


Fig.1 Context Information of the Calculation Data Object obj_i (SS stands for the attribute dimension set n obj_i as v_{ij}=1)

According to the Formula(1), the context outlier data has the following parallel mining process, shown as follows: firstly, the neighbor set LDS of each data object obj(obj_i) in the data set DS is called. According to the K neighbor set LDS(obj_i) of bj, the sparse factor of obj_i is calculated to generate the sparse factor matrix cluster $[Z_{L\lambda}]_{L\times M}$ of the data set DS. Secondly, according to the K neighbor set LDS(obj_i) of $[Z_{L\lambda}]_{L\times M}$ and obj_i, the local sparse difference di of obj_i is generated. Based on the established local sparse difference ε, the attribute dimension set SS of v_{ij}=1 is obtained. In the end, the context information of the outlier factor Factor(obj_i) and attribute dimension set SS containing the data object can be confirmed. The calculation process of the outlier data subject factor value with context is shown in Figure 1.

B. The Parallel Mining

The MapReduce is a programming model. The main operation is divided into two stages, including Map and Reduce. Input and output of each stage are based on the key value pairs. In Map stage, Map function will change each line in input document in the form of the key value pair(K_1, V_1). After map function treatment, multiple new key pairs List(K_2, V_2) are output. In the Reduce stage, the output keys of Map stage are grouped(K_2, List(V_2)). The process is called as shuffle. Each group(K2, List(V_2) is used as the input of the Reduce function. After the treatment of the Reduce function, the final result is output(K_3,V_3). As calculating the local sparse factor matrix$[Z_{L\lambda}]_{L\times d}$(O) (O∈LDS($obj_i$)), the distributed strategy LSH can be applied, but the steps should be changed, namely, LDS(O) (O∈LDS(obj_i) should be inquired in the associated data set of obj_i index value. As calculating the sparse factor and outlier factor, it can be realized by Map. As conducting the full-ranking in line with Factor, a Map is needed to sample Factor, so as to determine the Partition function of the node for each (K_2, V_2). Reduce is needed to sort each node(K_2, V_2). The realization process of MapReduce programming model is shown in Figure 2.



Key	Value
obj_1	$LDS(obj_1)$
obj_2	$LDS(obj_2)$
obj_3	$LDS(obj_3)$
…	…

Key	Value
obj_1	$[Z_{L\lambda}]_{k\times d}(o)(o\in LDS(obj_1)$
obj_2	$[Z_{L\lambda}]_{k\times d}(o)(o\in LDS(obj_2)$
obj_3	$[Z_{L\lambda}]_{k\times d}(o)(o\in LDS(obj_3)$
…	…

Key	Value
$Factor(obj_1)$	obj_1
$Factor(obj_2)$	obj_2
$Factor(obj_3)$	obj_3
$Factor(obj_4)$	obj_4
…	…

Key	Value
obj_1	λ_1
obj_2	λ_2
obj_3	λ_3
…	…

Key	Value
obj_1	$Factor(obj_1)$
obj_2	$Factor(obj_2)$
obj_3	$Factor(obj_3)$
…	…

Key	Value
obj_1	$Factor(obj_1)$
obj_2	$Factor(obj_2)$
obj_3	$Factor(obj_3)$
…	…

Fig.2. The Realization Process of MapReduce Procedure

In figure 2, LSH (Locality Sensitive Hashing)strategy should be applied to calculate the appropriate K-NN for the data object obj and confirm the local data set for each data object obj_i in DS. This can be realized through two MapReduce tasks. The local sparse factor λ_{ij} of each attribute value for obj_i uses (obj_i,i) as the output and save it. This can be realized through the Map stage in the MapReduce tasks. The corresponding local sparse factor matrix $[Z_{L\lambda}]_{L\times d}(O)$ ($O\in LDS(obj_i)$) of LDS(O) ($O\in LDS(obj_i)$) generating obj_i also can apply the LSH strategy. Moreover, it can be realized through two MapReduce. According to the corresponding local sparse factor matrix $[Z_{L\lambda}]_{L\times d}(O)(obj_i)$ for obj_i, the local sparse difference factor corresponding to each attribute dimension for obj_i is calculated. The definition 2 is used to confirm the corresponding subspace definition vector for obj_i and calculate the outlier factor of obj_i. This can be realized through the Map stage in MapReduce task. According to Factor, data objects are sorted in parallel. A MapReduce task is used to be realized. n data objects with the larger outlier degree can be used as the outlier data. The parallel algorithm description is shown as follows: algorithm. RSLODA; input: dataset DS(attribute number is d), neighbor number K, and sparse difference factor threshold ε.

Output: n outlier data

(1) Execute Map Reduce task in LSH and generate {(obj,LDS(obj))};//* and calculate LDS(obj) of data object obj

(2) Use {(obj,LDS(obj))} as the output, execute MapReduce task, generate {(obj,λ)};//* and confirm the sparse factor λ corresponding to each data object obj

(3) Use {(obj,λ)} as input, execute MapReduce task in LSH, generate {(obj,($[Z_{L\lambda}]k\times d(O)(O\in LDS(obj_i)$))))};//* and confirm the local sparse factor matrix corresponding to eac data object obj in DS

(4) Use {(obj,($[ZL_{L\lambda}]k\times x\ d(O)(O\in LDS(obj_i)$))))} as the input, execute MapReduce task, generate {(obj,Factor(obj))};//* and confirm the corresponding outlier factor(obj) for each data object in DS

(5) Use {(obj,Factor(obj))} as the input to execute the MapReduce task, conduct the full ranking for {(obj,Factor(obj))} as the Factor //* and confirm to conduct full ranking for each data object obj as the corresponding outlier factor(obj) in DS

(6) Output the maximal n data objects in the outlier degree, and //* and select Top(N) as the outlier data

Algorithm statement:

1. In the above-mentioned RSLODA parallel algorithm, MapReduce task in step(1) and step(3) is to realize the LSH strategy, namely the approximate KNN of each data object is inquired to give the detailed analysis in the treatment process. Sample the result for MapReduce in step(5) and conduct the parallel full ranking.

2. In RSLODA, the MapReduce task execution process in step(2) is shown as follows:

```
receive{(obj,LDS(obj))}
Foreach(obj,LDS(obj))in{(obj,LDS(obj))}
Map:
For(m=1;m<=d;m++){
For(j=1;i<=K+1;j++){
Set[j]=(LDS[j][m]);//* the each attribution of mth column for the corresponding data in L[i] is added in the array.
}
$\lambda_{im=}$=computer $\lambda_{im}$ ($\lambda$Set);//* calculate the sparse factor of the corresponding attribute value of L[i] on the mth dimension
$\lambda_i$[m]= $\lambda_{im}$;//* Add the corresponding sparse factor of the corresponding attribute value of L[i] on the mth dimension in the array.
}
emit<obj, $\lambda$>
```

3. In RSLODA, MapReduce executive process in the step(4) is shown as follows:

```
receive{(obj,($[Z_{L\lambda}]\ k\times d\ (O)\ (O\in LDS(obj_i)$))))}
```

```
Foreach(obj,($[Z_{L\lambda}]$ k×d
(O) (O∈LDS($obj_i$))))in{(obj,($[Z_{L\lambda}]$ k×d(O) (O∈LDS($obj_i$))))}
Map:
1)For(j=1;j<=d;j++){
2)x=0;
3)For(m=1;m<=K+1;m++){
x+=$[Z_{L\lambda}]$k×d(obj)[m][j];
4)}
5)$d_{ij}$($ob_j$);//* Formula(9), calculate the corresponding local sparse difference factor of ith data object in jth attribute value
6)If($d_{ij}$($ob_j$)<ε)
7){$v_i$[j]=0;
8)}else{$v_i$[j]=1;}
9)}
10)If(||$v_i$||1=0){
11)Factor(obj)=0;
12)}else{
13) Factor(obj);//* Calculate the outlier degree of ith data object
14) }
15) emit<obj,Factor(obj)>.
```

3. Instance Analyssi

A 2D data example constituted by 3 clusters and outlier data is given, including 100 normal data objects and 10 lines of outlier data. The RSLODA algorithm has the operation result for the example under the pseudo distribution, shown in Figure 3. Parameter k=1 and w=50, the number of has table is 4. The number of KNN is 20. In the Figure 3(a), the data set contains more relatively obvious 3 gathering clusters. It can be observed from Figure 3(a)-Figure 3(d) that with the increase of ε, the data objects without the associated space are increasing.

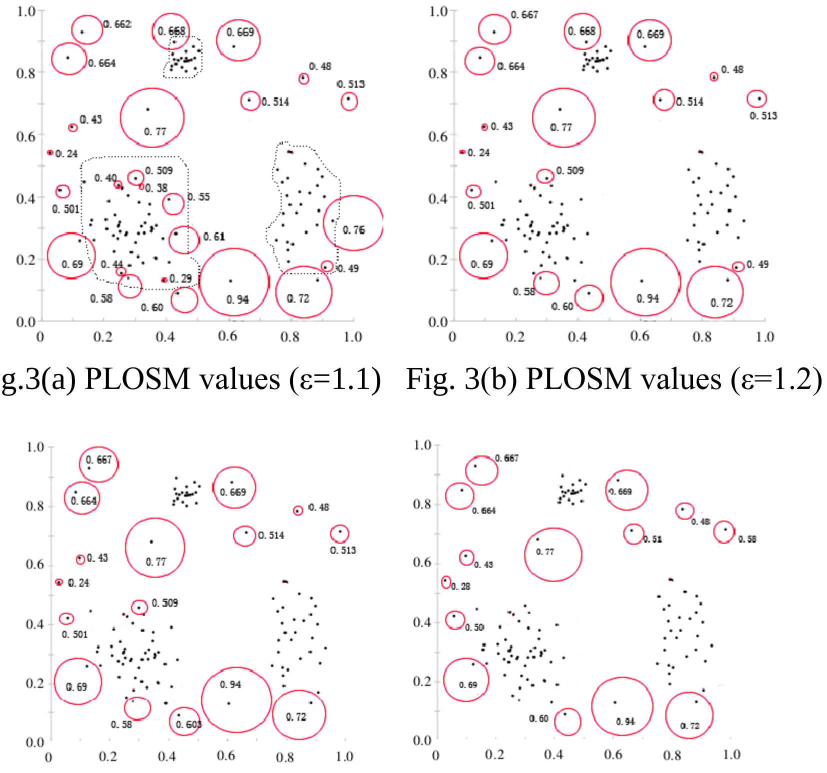


Fig.3(a) PLOSM values (ε=1.1)   Fig. 3(b) PLOSM values (ε=1.2)

Fig.3 (c) PLOSM values (ε=1.3)   Fig. 3(d) PLOSM values (ε=1.4)

4. Conclusions

In outlier data mining, the context outlier data mining algorithm of MapReduce programming model effectively improves the interpretability and intelligibility of outlier data.

References

[1] Li Yonghong, Zhang Jifu and Gou Yaling, the Study on the Local Outlier Data Mining Algorithm in the Associated Space[J], the Mini-Micro System, 2015, 36(03): 460-465

[2] Xu Lin and Zhao Maoxian, the Study on the Local Outlier Data Mining Algorithm Based on the Density[J], Journal of Shandong University of Technology(natural Science), 2016, 30(06): 7-11

[3] Feng Tingting and Zhang Jifu, the Outlier Data Mining Method Based on the Grid Unit[J], Journal of Taiyuan University of Science and Technology, 2016, 37(05): 359-364;

[4] Lou Shengjin, Zhang Jifu and Liu Aiqin, the Outlier Data Mining Algorithm Based on P Weight[J], the Mini-Micro System, 2014, 35(01): 55-59;

[5] Zhang Guang, the Study on the Recommended E-commerce System Based on the Outlier Data Mining[J], Automation and Instrument, 2017, (08): 21-22+ 25.