IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 9, No. 1, March 2020, pp. 1~10
ISSN: 2252-8938, DOI: 10.11591/ijai.v9.i1.pp1-10

A fuzzy neighborhood rough set method for anomaly detection in large scale data

El Meziati Marouane[1], Ziyati Elhoussaine[2]
[1]Associated member LRIT University Mohammed V, Rabat, Morocco
[2]Associated member LRIT University Hassan 2 ESTC, RITM LAB, Rabat, Morocco

Article Info

Article history:

Received Oct 20, 2019
Revised Dec 25, 2019
Accepted Jan 6, 2020

Keywords:

Big data
Clustering
Fuzzy neighborhood
Map Reduce
Outlier detection
Rough set
Time complexity

ABSTRACT

Recent research studies on outlier detection have focused on examining the nearest neighbor structure of a data object to measure its outlierness degree. Moreover, popular outlier detection methods require the pairwise comparison of objects to compute the nearest neighbors. This quadratic problem is not scalable to large data sets, making multidimensional outlier detection for big data still an open challenge. In this article, we present a new approach for outlier detection, based on highly scalable approach to compute the nearest neighbors of objects using fuzzy rough set theory. At the same time, the outlier ranking process is accelerated by using a high-performance and a parallel computating using mapreduce framework.

Corresponding Author:

El Meziati Marouane,
Associated member LRIT,
University Mohammed V, Rabat, Morocco.
Email: elmeziati.marouane@gmail.com

1. INTRODUCTION

Outliers are the unusual, unexpected patterns in the observed world. Outliers exist extensively in real world, and they are generated from different sources: a heavily tailed distribution or errors in inputting the data. While there is no single, generally accepted, formal definition of an outlier, Hawkins' definition captures the spirit: "an outlier is an observation that deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism" [1]. Anomaly detection is an important problem that has been researched within diverse research areas and application domains such as fraud detection [2], intrusion discovery [3], video surveillance, pharmaceutical test and weather prediction. There are different surveys about classical outliers and abnormal detectio. They vary between density based approaches [3], statistical [4], distance-based [5], neural networks and machine learning techniques.

Recent research studies on outlier detection have focused on examining the nearest neighbor structure of a data object to measure its outlierness degree [6-7]. Such techniques are based on the key assumption that instances of normal data occur in dense neighborhoods, while outliers occur far away from their closest neighbors [8]. Popular outlier detection methods require the pairwise comparison of objects to compute the nearest neighbors. This quadratic problem is not scalable to large data sets, making outlier detection for large scale data still an open challenge. This paper proposes a fast outlier detection method for large scale datasets, which consists of two steps: a granulation of the universe into parts with the same properties then the computing of the degree of outlierness called Fuzzy neighborhood rough set outlier factor (FNROF) for each granule formed. Granulation of the obesevable universe involves grouping of similar elements into granules. With granulated views, we deal with approximations of concepts, represented by subsets of the universe, in terms of

granules [9]. The remainder of this paper is organized as follows. In the next section, we present some preliminaries of rough set theory that are relevant to this paper and discussion of the granularity of knowledge in connection with rough and fuzzy sets. In Section 3, we propose an efficient parallel computing system based on Map Reduce in order to improve the speed of computation and the algorithm proposed that deal with more complex outlier detection problems for large scale data.

2. ROUGH SETS (RST)

Rough set theory RST [10-11] is a new mathematical approach to imperfect knowledge. The theory has attracted attention of many researchers and practitioners all over the world, who contributed essentially to its development and applications. The main advantage of rough set theory in data analysis is that it does not need any preliminary or additional information about data. Rough set theory is a popular and powerful machine learning tool. It is especially suitable for dealing with information systems that exhibit inconsistencies. In rough set theory, an information table is defined as a tuple T = (U, A) where U and A are two finite, non-empty sets with U the universe of primitive objects and A the set of attributes. Each attribute or feature a ∈ A is associated with a set V_a of its value, called the domain of a. We may partition the attribute set A into two subsets C and D, called condition and decision attributes, respectively. Let P ⊂ A be a subset of attributes. The indiscernibility relation, denoted by:

$$\text{IND}(P) = \{(x, y) \in U^2 / \forall a \in P, a(x) = a(y)\} \quad (1)$$

Where a(x) denotes the value of feature of object x.
If (x, y) ∈ IND (P), x and y are said to be indiscernible with respect to P. The family of all equivalence classes of IND (P), referring to a partition of U determined by P, is denoted by U/IND(P). Each element in U/IND (P) is a set of indiscernible objects with respect to P. The family of all equivalence classes of IND (P), referring to a partition of U determined by P, is denoted by U/IND (P).

$$\text{Where } A \otimes B = \{X \cap \text{Y/X} \in \text{A,Y} \in \text{B,X} \cap Y \neq \emptyset\} \quad (2)$$

For any concept X ⊆ U, X could be approximated by the P-lower approximation and P-upper approximation using the knowledge of P. The lower approximation of X is the set of objects of U that are surely in X:

$$\underline{P}(X) = \cup\{E \in \text{U/IND}(P) : E \subseteq X\} \quad (3)$$

The upper approximation of X is the set of objects of U that are possibly in X, defined as:

$$\overline{P}(X) = \cup\{E \in \text{U/IND}(P) : E \cap X \neq \emptyset\} \quad (4)$$

The concept defining the set of objects that can possibly, but not certainly, be classified in a specific way is called the boundary region, which is defined as: $\text{BN}(P) = \overline{P}(X) - \underline{P}(X)$ as shown in Figure 1.

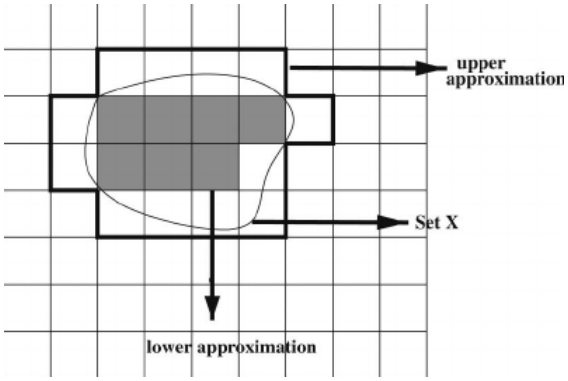

Figure 1. Representation of the data partitioning for a subset X

2.1. Rough set and fuzzy discretization

The extraction of knowledge from a huge volume of data using rough set methods requires the transformation ofcontinuous value attributes to discrete intervals, in order to form a grid structure and then form clusters from the cells in the grid structure. Clusters correspond to regions that are denser in data points than their surroundings. The great advantage of grid-based clustering is a significant reduction in time complexity, especially for very large data sets. The concepts of real rough space, it is well known that one of the research premises in the classical rough sets theory is the information or the data to be discrete. Discretization can be viewed as a data reduction technique which reduces the range of values of a continuous values attribute into a minimum number of discrete intervals. The numbers of cut-points can determine the level of data reduction. The fewerthe number of cut-points the more the data will be reduced and hence a generalized classifierwill be possible. The term "cut-point" refers to a real value within the range of continuous values that divides the range into intervals. Cut-point is also known as split-point. The great advantage of grid-based clustering is a significant reduction in time complexity, especially for very large data sets. But during the discretization process, if the discretization is too rough, much useful information may be lost. And if the discretization is too exact, it will take a lot of time complexity. So, it can be said that the disadvantages of classical rough sets are too much depending on good or bad of the discretization methods and the limited application domain.

Let $X = (x_1, x_2, .., x_n)$ be a provided dataset having n objects and Aattributes, v_{minj}=min(x_i), v_{maxj}=max(x_i) be the minimum and maximum values of attributes i. Each attribute $[V_{mini}, V_{maxi}]$ is equally divided into M intervals $w_i = (v_{maxi} - v_{mini})/M$. The set of all initial interval of an attribute i is shown as: $Interv_i = \{v_{min\,i}, (v_{min\,i} + w_j), (v_{min\,i} + 2*w_j), ..., v_{max\,i}\}$

2.2. Fuzzy rough sets

Fuzzy rough set theory extends rough set theory to data with continuous attributes, and detects degrees of inconsistency in the data. Key to this is turning the indiscernibility relation into a gradual relation. The fuzzy set is actually a fundamentally broader set compared with the classical or crisp set. The classical set only considers a limited number of degrees of membership such as '0' or '1', or a range of data with limited degrees of membership as shown in Figure 2.

Definition 1: (Fuzzy Sets) A fuzzy set, F, defined over universe X is a function defined as:

$$F = \{(x, \mu(x)) | \mu(x) \in [0,1], \forall x \in X\} \quad (5)$$

Function $\mu(x)$ is called the membership function, which maps object x to the membership space. The rough membership function expresses conditional probability that x belongs to X given P and can be interpreted as a degree that x belongs to X. One of the most important concepts in fuzzy set theory and applications is the α-cut decomposition theorem developed by Zadeh in 1971 under the name resolution identity. These cuts are crisp sets associated with certain levels α that represent distinct grades of membership.

Definition 2: (FS α-cut) given a number $\alpha \in [0,1]$, a α-cut or α-level set, ofa fuzzy set F is defined by:

$$F_\alpha = \{(x, \mu_x) | \mu_x \geq \alpha, \forall \alpha \in [0,1]\} if\, \alpha_0 < \alpha_1, F_{\alpha_0} \supseteq F_{\alpha_1} \quad (6)$$

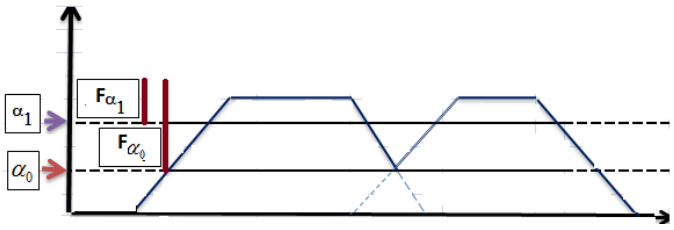

Figure 2. (Alpha, Beta)-cuts of fuzzy set F

We define the membership function of the Intersection of two fuzzy sets A(x,$\mu_{A_\alpha}(x)$) and B(x,$\mu_{B_\alpha}(x)$) as: $(A \cap B)_\alpha = (x,\ \mu_{A_\alpha \cap B_\alpha}(x) = \frac{1}{2} * (\mu_{A_\alpha}(x) + \mu_{B_\alpha}(x))\ )$ x $\in X$

2.3. Rough sets: neighborhood systems

The concept of information granulation was first introduced by Zadeh in the context of fuzzy sets in 1979 [12]. The basic ideas of information granulation have appeared in fields, such as interval analysis, quantization, rough set theory and many others. There is a fast growing and renewed interest in the study of information granulation and computations under the term of Granular Computing (GrC). [13] Granulation of a universe involves grouping of similar elements parts, or the grouping of individual elements or objects into a family of disjoint subsets, based on available information and knowledge. The combination of topological spaces and rough sets and the properties of topological rough spaces are discussed [14] used neighborhood systems and topological concept in the study of approximations. Neighborhood system is a mathematical structure of granular computing to model granules, and can be used to compute structure of granules and/or between granules. A neighborhood system at a point is a framework to capture the concept of "near" objects, and any subset of objects can be approximated by a set of neighborhoods. A neighborhood system defines a set of binary relations, and a set of binary relationships can be used to define a neighborhood system.

Definition 3 (neighborhood of object x_i): Given an arbitrary$x_i \in U$ and P $\subseteq C$, the nearest neighborhood $\delta_\partial^P(x_i)$ of x_i in feature space P is defined as:

$$\delta_s^P(x_i) = \{x_j | \Delta^P(x_i, x_j) \leq \varepsilon, s \in \Re+\} \quad (7)$$

Where $\Delta: U \times U \rightarrow R+$, a distance (similarity) function and R+ is the set of non-negative real number.δ_s^P: The neighborhood information granule included objects x_i and the size of the neighborhood depends on thresholdε.

For each value of$s \in R+$, we propose the following neighborhood system as the collection of all neighborhoods of x ∈ U as:

$$N_s^P(x) = \{\delta_s^P(x) | s \in \Re+, P \subseteq C\} \quad (8)$$

Where s is a sliding windows for overlapping computation: s<M.

Theorem 1: For each$P_1 \subseteq A, P_2 \subseteq A$, $N_s^P(x)$ is a neighborhood relation induced in feature subspace P.

We have: $N_s^{P_1 \cup P_2}(x) = N_s^{P_1}(x) \cap N_s^{P_2}(x)$

$$if\ A = \cup_i P_i \quad \text{so} \quad N_s(x) = \cap_i N_s^{P_i}(x) \quad (9)$$

Given a set of objects U and a neighborhood system N_s over U, we call <U, N_s> a neighborhood approximation space. The lower and upper approximations ($\underline{NX}$,$\overline{NX}$) of X in <U, Ns>, are defined as:

$$\underline{NX} = \bigcup_{N_s(x) \subseteq X} N_S(x) \quad \overline{NX} = \cup_{N_s(x) \cap X \neq \emptyset} N_s(x)$$

Obviously, $\underline{NX} \subseteq X \subseteq \overline{NX}$. The boundary region of X in the approximation space is defined as:

$$BNX = \overline{NX} - \underline{NX}.$$

The size of boundary region reflects the degree of roughness of set X in the approximation space <U,N_s>. Assuming X is the sample subset with a decision label; generally speaking, we hope the boundary region of the decision should be as small as possible for decreasing uncertainty in decision. The size of boundary region depends on X, attributes to describe U.

For a fixed pair of numbers $(\alpha_0, \alpha_1) \in$ [0, 1]×[0, 1], we obtain a submodel in which a crisp set $F_\alpha$is approximated in a crisp approximation space $apr_{\Re\alpha_0} = (U, \mu_{\Re\alpha_0})$ The result is a rough set $((\underline{apr}_{\Re\alpha_0}(F), \overline{apr}_{\Re\alpha_0}(F))$ with the reference set F.Each granule in fuzzy sets $F$is a neighborhood of an element of the universe. The approximation is defined by show in Figure 3:

$$\underline{NA} = \cup_{N_s(x) \subseteq F_{\alpha_1}} N_s(x) \quad , \text{ for } F_{\alpha_1} \subseteq F_{\alpha_0} \subseteq U \quad (10)$$

$$\overline{NA} = \cup_{N_s(x) \subseteq F_{\alpha_0}} N_s(x) \quad (11)$$

In this case, the subset F_{α_1} (lower approximation) contains two clusters C1 (grid 2) and C2 (grid 3) $F_{\alpha_1} = C_1 \cup C_2$.

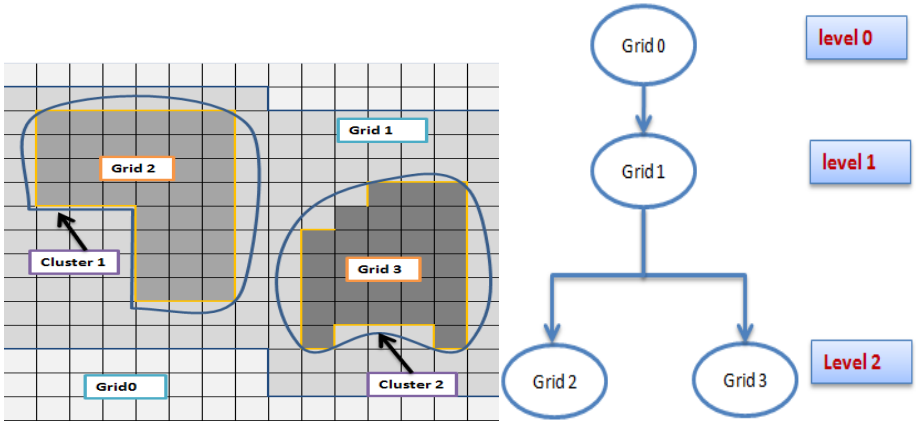

Figure 3. Fuzzy rough set approximation

The root grid Grid0 ($universe\ U$) with the coarsest granularity covers the entire datasets, which contains one sub grids: grids 1 ($upper\ approximation: F_{\alpha_0}$) at level 1 also contains two sub grids at level 2 ($lower\ approximation.: F_{\alpha_1}$)

2.4. Fuzzy neighborhood rough set outlier factor (FNROF)

In this paper, a new method for ranking outlier which is proposed based on fuzzy rough set denoted "Fuzzy neighborhood rough set outlier factor" FNROF. After dividing each dimension into intervals of equal length M, the density distribution of each cell (information granularity) can be defined as the ratio of its density and the average density of its k neighboring cells.

$$\Im_i^P = \sum_{j=1}^{n} \left(\frac{d_i}{d_j}\right) * \log\left(\frac{d_i}{d_j}\right) = \sum_{j=1}^{n} \left(\frac{n_i}{n_j}\right) * \log\left(\frac{n_i}{n_j}\right) \quad (12)$$

Proof: $\frac{d_i}{d_j} = \frac{n_i}{\boxed{M^n}} * \frac{\boxed{M^n}}{n_j} = \frac{n_i}{n_i}$

A normalized score of ζ_i^P is given as follow:

$$\zeta_i^P = 1 - \frac{\Im_i^P - \Im_{min}^P}{\Im_{max}^{P_{min}^P}} \quad 0 \leq \zeta_i^P \leq 1$$

It's viewed as the relative density measure of cl_i (di) with respect to the density of n surrounding neighbor's cell. When the probability is uniformly distributed, we are most uncertain about the outcome, the entropy (score) is the highest in this case. On the other hand, when the data points have a highly probability mass function, we know that the variable is likely to fall within a small set of outcomes so the uncertainty and the entropy (score) are low. The size of interval must be carefully selected. If the interval size is too small, there will be many cells so that the average number of points in each cell can be too small. On the other hand, if the interval size is too large, we may not be able to capture the differences in density in different regions of the space. Unfortunately, without knowing the distribution of the data sets, it is difficult to estimate the minimal average number of points required in each cell to have the correct result.

Definition 4: Directly density-reachable: A cell cl_i is directly density-reachable from a cell cl_j if only if, ${\zeta_i}^P \geq \beta$ and $cl_j \in N(cl_i) where\ \Delta^P(cl_i, cl_j) = {\zeta_j}^P - {\zeta_i}^P$

That is, cl_i is a core cell and cl_j is in its neighborhood.

Definition 5: Density-connected. A cell cl_i is density-connected to a cell cl_j if there is a cell cl_k such that both cl_i and cl_j are density-reachable from cl_k as shown in Figure 4.

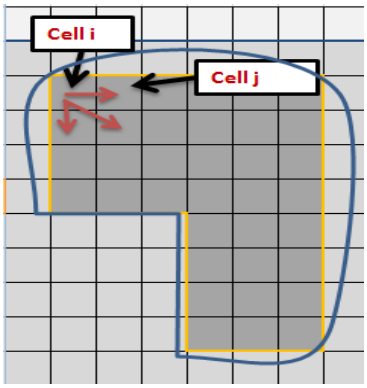

Figure 4. The concept of density-reachability and density-connectivity to form clusters as contiguous dense regions in lower approximation

2.5. A novel approach: A high-performance parallel and distributed computation using mapreduce

In order to compute an optimal set of cut-points, most of discretization algorithms perform an iterative search in the space of candidate discretizations, using different types of scoring functions for evaluating a discretization, that take a lot of time. In this paper, we propose a parallel process of discretization based on MapReduce using sliding grid. A sliding grid is specified by defining its range M and slide S. The range M is an interval of discretization while the slide S specifies the portion of the grid that is moved forward. A sliding window is specified as a tuple (M,s). A smooth sliding specification is highly desired where the slide S issmall relative to the range M. where$S < M$. The proposed algorithm based on MapReduce computed for each node i($P_i \subseteq A$) is a parallel process that consists of three steps: map, shuffle, and reduce as shown in Figure 5.

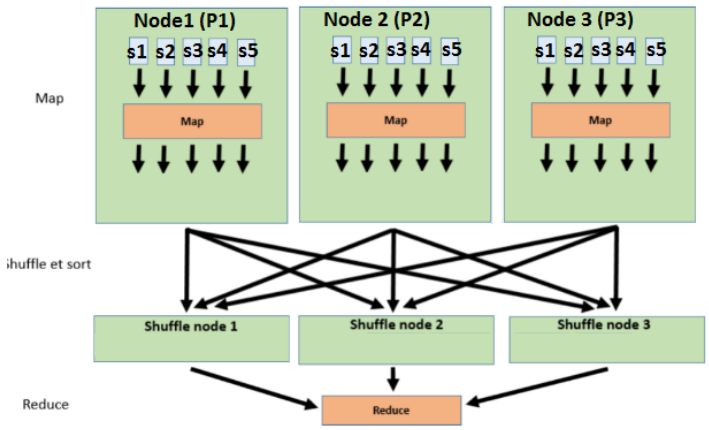

Figure 5. Framework MapReduce proposed

Example: S=<U,A={C1,C2,C3,C4,C5}>
P1= {C1,C2 }
P2= {C2, C3}
P3= {C3, C4, C5}
($P_i \subseteq A$ and $A = P_1 \cup P_2 \cup P_3$)

At node 1 (P1):
Each worker node that applies the map function related to each grid defined by tuple {(M,s1),(M,s2),(M,s3),(M,s4),(M,s5)}

Inmap phase, for each grid given tuple (M,s), we generates a list (key=cl_i,value=$\zeta_i{}^P$) where $\zeta_i{}^P$ is a score of cl_i. In shuffle phase, the output pairs are partitioned and then transferred to reducers. In reduce phase, pairs with the same key are grouped together as (cl_j, list($\zeta_j{}^P$)) as shown in Figure 6.

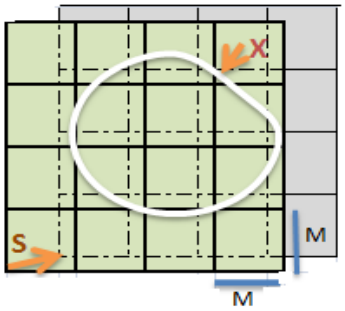

Figure 6. Illustrates how the cell overlaps when the grid move

Then the reduce function generates the final output pairs list (cl_k,$\zeta_k{}^P$) for each fuzzy approximation. The whole process can be summarized as follows:
Map: (M,s) ➔ (cl_i,$\zeta_i{}^P$)
Reduce: (cl_i, list($\zeta_j{}^P$)) ➔(cl_k,$\zeta_k{}^P$)
A parallel computing of FNROF and its template implementation

Master:

Get $\{cl_i,\zeta_i{}^P, ListCl_j\}$ from the result queue
If ($\zeta_i{}^P \geq \beta$)
for each cell cl_cin $ListCl_j$
if ($\zeta_c{}^P \geq \beta$ and cl_c is not labeled)
{
$\mathbb{C}_{clustID} = \mathbb{C}_{clustID} \cup cl_c$
Label (cl_c) = clustID
putcl_c in the candidate queue

}

Slave:
Get Cellcl_i from the candidate queue

$$ListCl_j = neighborhood(cl_i)$$

put $\{cl_i,\zeta_i{}^P, ListCl_j\}$ in the result queue

Algorithm MR- FNROF: Fast outlier detection algorithm based on fuzzy neighborhood rough and a pipeline parallelism between master and slave module.

clustID =0
for each cell cl_i in grid database
{
if($cl_i$is not labeled)
{
If($\zeta_i{}^P \leq \beta$)
{

$NEG = NEG \cup cl_i$
Label (cl_i) =noise
} else if ($\zeta_i{}^P \leq \alpha$)
{
$NEG = NEG \cup cl_i$
Label (cl_i) =boundary
} else {

While (there are pending results)
{
Master in (neighborcl_i) out candidatecl_c
Parallel: Slave in (candidatecl_c) out (neighborcl_i)
}
clustID =clustID + 1
}

Example of computation (At a single node P):
Map phase:
Cell i: cl_i

20	40	10
70	90	100
80	99	102

$\Im_i^P$=(90/20)*log(90/20)+(90/40)*log(90/40)+(90/10)*log(90/10)+(90/70)*log(90/70)+(90/100)*log(90/100)+(90/80)*log(90/80)+(90/99)*log(90/99)+(90/102)*log(90/102)
$\Im_i^P$=28.53
Cell k: cl_k after moving the grid

24	44	14
73	94	105
84	103	107

$\Im_k^P$=(94/24)*log(94/24)+(94/44)*log(94/44)
+(94/14)*log(94/14)+(94/73)*log(94/73)+(94/105)*log(94/105)+(94/84)*log(94/84)+(94/103)*log(94/103)+(94/107)*log(94/107)
$\Im_k^P$==19.90
$\Im_{max}^P$80
$\Im_{min}^P$
ζ_i^P = 0.36 $\zeta_k^P = 0.25$

Shuffle and Reduce phase:
Given a cut point $\alpha_0 = 0.3$
$\zeta_{cl_i - cl_k}^P = \zeta_{cl_i}^P = 0.36 > \alpha_0$
$\zeta_{cl_k - cl_i}^P = \zeta_{cl_k}^P = 0.25 < \alpha_0$
$\zeta_{cl_i \cap cl_k}^P = \frac{1}{2} * (\zeta_{cl_i}^P + \zeta_{cl_k}^P)$
$\zeta_{cl_i \cap cl_k}^P = \frac{1}{2} * (0.36 + 0.25) = 0.305 > \alpha_0$

Lower approximation:
$(cl_i \cap cl_k) \subseteq \underline{X}$
$(cl_i - cl_k) \subseteq \underline{X}$
$\underline{X} = \underline{X} + (cl_i - cl_k) + (cl_i \cap cl_k)$

Upper approximation:
$(cl_k \cup cl_i) \subseteq \overline{X}$
$\overline{X} = \overline{X} + (cl_k \cup cl_i)$

3. EXPERIMENTS AND RESULTS

The algorithm proposed is tested with synthetic and real data collected from NOAA center. The implementation of this work was realized in R using RStudio. Datasets NOAA: [15] The National Climatic Data Center – NOAA: collects a wide range of data; including sensor streams with temporal information, sensor spatial information, temperature, etc.

3.1. Improvement in search time efficiency

The purpose of the experiment was to compare the performance between the algorithm proposed MR-FNROF and the original LOF algorithm in terms of matching detected outliers and execution time. Comparing the performance of the tow methods, it shows that our method have a very fast processing time with acceptable trade-off errors as show in Table 1.

Table 1. Time taken and matching detected outliers according to the number of objects in the dataset for both MR-FNROF and LOF method

Number of objects	Time taken (seconds)		Number of outliers detected	
	MR-FNROF Method (9 nodes)	LOF method	MR-FNROF Method (9 nodes)	LOF method
2023	0.29	5.37	203	123
4845	0.34	11.3	302	284
19768	1.9	50.2	713	688
938419	8.49	523.4	2023	1987

3.2. Performance of MR-FNROF according to number of workers nodes

The second experiment shows that reduction of the risk of a Type I & II error is performed by increasing the number of workers nodes as shown in Figure 7. With high number of workers nodes, we are getting more outlier detected in upper approximation rough set (less of type II errors).

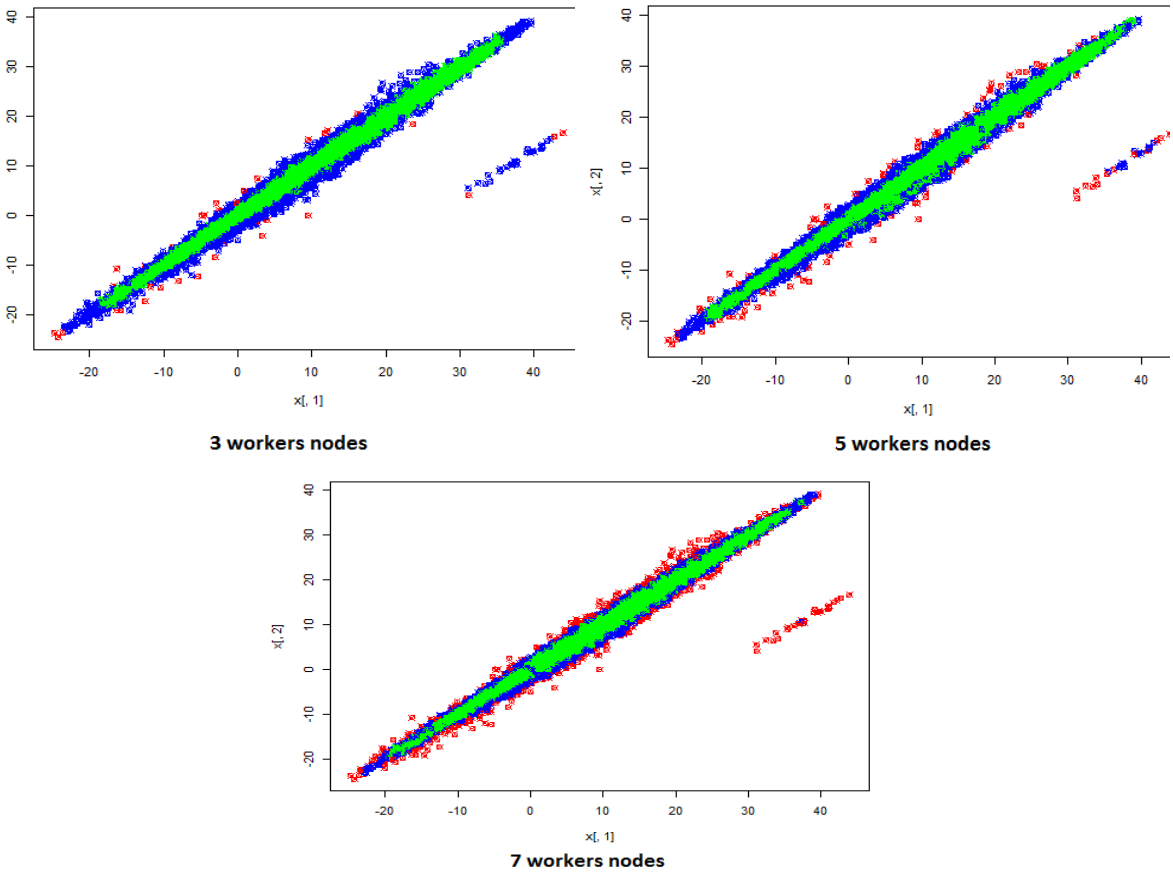

Figure 7. Anomaly detection using successively 3, 5 and 7 workers nodes given (alpha, beta)-cuts = (20%, 50%)

4. CONCLUSION

The aim of this paper is to propose a new algorithm of outlier detection that reduces the computation time required by using granular computing method and fuzzy rough set thoery. The algorithm MR- FNROF divides the universes into a smaller number of granules, and calculates the factor of outlierness for each granule. To examine the effectiveness of the proposed method, several experiments incorporating different parameters were conducted. The proposed method MR- FNROF, demonstrated a significant computation time reduction. Moreover, it can also be effectively used for real-time outlier detection.

ACKNOWLEDGEMENTS

The authors are very much thankful to the unanimous reviewers of the paper and editors of the journal for their constructive and helpful comments that improved the quality of the paper

REFERENCES

[1] Hawkins, D.: Identifications of Outliers, (Chapman and Hall, London, 1980).
[2] Andrea Dal Pozzolo, Giacomo Boracchi, "Credit Card Fraud Detection: A Realistic Modeling and a Novel Learning Strategy"- September 2017.
[3] Jabez J, B.Muthukumar, "Intrusion Detection System (IDS): Anomaly Detection using Outlier Detection Approach" in International Conference on Intelligent Computing, Communication & Convergence (ICCC-2015) Elsevier.
[4] S. Hido, Y. Tsuboi, H. Kashima, M. Sugiyama, and T. Kanamori, "Statistical outlier detection using direct density ratio estimation," *Knowl. Inform. Syst*., vol. 26, no. 2, pp. 309-336, 2011.
[5] K. Bhaduri, B. L. Matthews, and C. Giannella, "Algorithms for speeding up distance-based outlier detection," in Proc. ACM SIGKDD Int. Conf. KDD, New York, NY, USA, 2011, pp. 859-867.
[6] P.Rajashekar, "Ranking outlier detection for high dimensional data using symmetric neighborhood relationship", Vol.8, Issue2-2016.
[7] Jayshree S.Gosavi, Vinod S.Wadne, "Unsupervised Distance-Based Outlier Detection Using Nearest Neighbours Algorithm on Distributed Approach: Survey" - Vol. 2, Issue 12, December 2014.
[8] Shuchita Upadhyaya, Karanjit Singh, "Nearest Neighbour Based Outlier Detection Techniques", volume3 Issue2-2012.
[9] Guoyin Wang, Jie Yang, ji Xu, "Granular computing: from granularity optimization to multi-granularity joint problem solving"- Volume 2, issue3-2017.
[10] Pawlak Za (1982) Rough sets. *Int J Parallel Program*. 11(5):341-356.
[11] Kalaivani.R, M.V.Suresh, N.Srinivasan "A Study of Rough Sets Theory and its Application Over Various Fields" Volume 3, No.2, (2017).
[12] Yao, Y.Y., Information granulation and rough set approximation, *International Journal of Intelligent Systems*, Vol. 16, No. 1, 87-104, 2001.
[13] Yao, Jingtao & Vasilakos, Athanasios & Pedrycz, Witold. (2013). Granular Computing: Perspectives and Challenges. *IEEE transactions on cybernetics*. 43. 10.1109/TSMCC.2012.2236648.
[14] T. Lin, Neighborhood systems and relational datebase, in: Proceedings of CSCÆ88, 1988.
[15] ftp://ftp.ncdc.noaa.gov/pub/data/noaa/