IJIAET International Journal of Innovative Analyses and Emerging Technology
| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1

Parameterized Rough-Fuzzy Classification on a Decision Table using a Threshold-Algorithms and Implementations

D. Latha
Department of Computer Science and Engineering, AdikaviNannaya University, Andhra Pradesh, India

B. Venkataramana
Department of Computer Science & Engineering, Holy Mary Institute of Technology, Hyderabad, India

Abstract: Rough sets, the invention of Pawlak, find various applications in knowledge engineering, for improving the quality and accuracy, subsequently, Parameterized Rough Sets Model has come into existence. In this paper, three algorithms are developed to compute parameterized rough computing based indices of similar objects in decision table with fuzzy decision attributes using a threshold on fuzziness and these algorithms are implemented these algorithms using C.

Keywords: Rough Sets, Decision table, Rough Indices, Knowledge Engineering

1. Introduction

Rough Sets [6,7,8] find wider applications in various areas of technologies as well as bio sciences as similar as Fuzzy tools. Considering the importance of these tools various researchers made contributions by hybridizing these concepts. In particular, G.Ganesan et. al., [3,4,5] discussed the procedure of indexing any element of the Universe of discourse through fuzzy a fuzzy subset using rough tools. In this paper, we used parameterized RS Model instead of conventional RS Model and we worked on these contributions and developed the algorithms of three types of indices namely ***lower, upper*** and ***rough*** and implemented them using C Programming.

This paper is divided into six sections. In Second section, we provided the basic mathematical concepts related to the subsequent sections. In third section, we discussed the lower index algorithm on a decision table with fuzzy decision attribute with a single threshold and implemented the same using C Programming. In fourth section, upper index algorithm for a decision table with fuzzy decision attribute under one threshold is provided and the same has been implemented using C Programming. In fifth section, rough indexing algorithm using a threshold for a decision table with fuzzy decision attribute is provided and the same has been implemented using C Programming and the paper ends with concluding remarks as 6^{th} section.

2. Mathematical Concepts

In this section, we provide the basic concepts which are useful for reading other sections.

IJIAET International Journal of Innovative Analyses and Emerging Technology
| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1

2.1 Rough Sets

The theory of Rough Sets is defined as follows: For an equivalence relation R defined on a finite universe of discourse U, denote U/R ={X_1, X_2, …., X_n}be the collection of equivalence classes induced by R on U. For a given input A, the lower and upper approximations are algorithmically defined as

Lower Rough Approximations

\\X_1, X_2, …, X_n– Equivalence Classes
\\ A-Input
Let D=NULL
For i=1 to n do
    If X_i is subset of A, then D= D $\cup$ X_i
Return D

Upper Rough Approximations

\\X_1, X_2, …, X_n– Equivalence Classes
\\ A-Input
Let D=NULL
For i=1 to n do
    If A $\cap X_i \neq$NULL then D= D $\cup$ X_i
Return D

Here, the Positive, Negative and Boundary Regions of a set A are defined as the lower approximation of A, complement of the upper approximation of A and the difference between upper and lower approximations of A respectively.

2.2 Parameterization of RS Model

As the RS Model deals with mere inclusions and intersections in computing rough approximations, these approximations include only 100% inclusions in the computations. But, in several research problems, even it is necessary to address the issues which are *nearing* 100% which are being ignored by the conventional RS Model. Hence, the Parameterized RS Model came into existence and for a given input A and $0 \leq \mu \leq \tau \leq 1$, the positive (POS(A)), negative (NEG(A)) and boundary (BND(A)) regions of A are defined respectively as follows:

$$POS(A) = \left\{ x \in U / \left| \frac{A \cap [x]}{[x]} \right| \geq \tau \right\} \text{------------------------------------(2.2.1)}$$

$$NEG(A) = \left\{ x \in U / \left| \frac{A \cap [x]}{[x]} \right| \leq \mu \right\} \text{--------------------------------(2.2.2)}$$

$$BND(A) = \left\{ x \in U / \mu < \left| \frac{A \cap [x]}{[x]} \right| < \tau \right\} \text{---------------------------(2.2.3)}$$

IJIAET International Journal of Innovative Analyses and Emerging Technology
| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1

2.3 Hybridization of Fuzzy Sets and Parameterized Rough Sets

Consider a fuzzy subset A on a finite universe of discourse U. Let R be an equivalence relation defined on U and U/R={X_1, X_2, …., X_n} denote the partition on U induced by R. For a threshold α ranging between 0 and 1, let A[α] denote the strong α- cut [1, 2] on A. The positive, boundary and Negative regions of the fuzzy set A are respectively given by

$$POS(A) = \left\{ x \in U / \left| \frac{A[[\alpha] \cap [x]}{[x]} \right| \geq \tau \right\} \text{-----------------------------(2.3.1)}$$

$$BND(A) = \left\{ x \in U / \mu < \left| \frac{A[\alpha] \cap [x]}{[x]} \right| < \tau \right\} \text{----------------------(2.3.2)}$$

$$NEG(A) = \left\{ x \in U / \left| \frac{A[\alpha] \cap [x]}{[x]} \right| \leq \mu \right\} \text{----------------------------(2.3.3)}$$

3. Lower Indices in a Decision Table consisting a Fuzzy Decision Attribute

In this section, we propose an algorithm to compute the lower index using lower parameterized rough approximations. In the algorithm, single threshold is used on a fuzzy input A and using square and square root functions, the lower indices are obtained. Also, we illustrate the algorithm for a decision table with a fuzzy decision attribute.

3.1 Algorithm for Lower index of an element

```
Algorithm (alpha, A, x)
//for computing Lower index of x an element of universe of discourse
//Algorithm returns the Lower index

1.   Let x_ind be an integer initialized to M
2.   Let K be the equivalence class containing x.
3.   If  U(y)=0 for all y in K
Begin
     x_ind=-x_ind
     goto 7
End
4.   If  U(y)=1 for all y belongs to K
     goto 7
5.   Compute "POS of A[alpha]"
6.   If " x belongs to POS of A[alpha]"
While ("x belongs to POS of A[alpha]")
     Begin
  alpha= sqrt(alpha) //square root of alpha
x_ind=x_ind+1
  Compute "POS of A[alpha]"
End
else
While ("x NOT belongs to POS of A[alpha]")
```

IJIAET International Journal of Innovative Analyses and Emerging Technology
| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1

```
      Begin
   alpha= sqr(alpha) //square of alpha
x_ind=x_ind-1
   Compute "POS of A[alpha]"
End
7.    Return x_ind
```

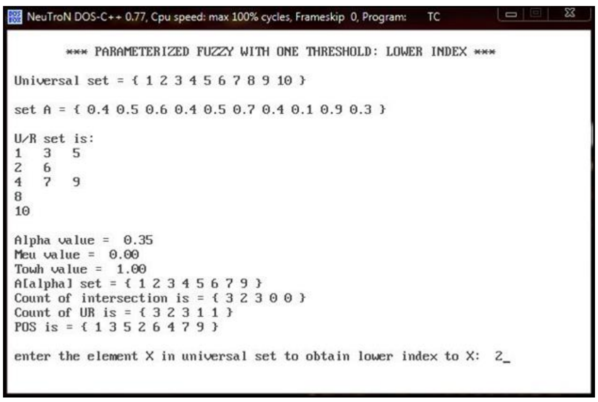
3.2 Experimental Results

Consider the following decision table with 10 records namely 1, 2,3,4,5,6,7,8,9 and 10 with three conditional attributes namely X_1, X_2, Xr_3 and a fuzzy decision attribute.

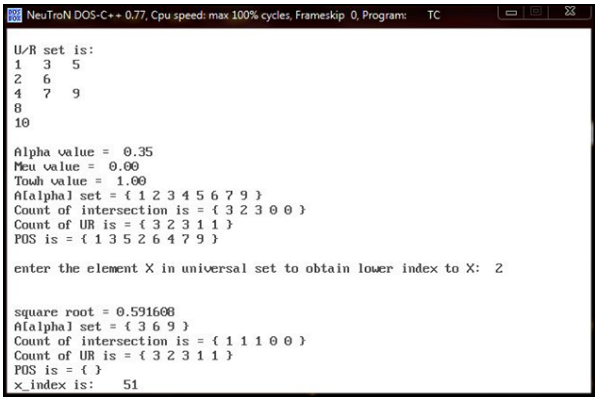
	X_1	X_2	X_3	Decision
1	Yellow	Red	Yellow	0.4
2	Red	Yellow	Green	0.5
3	Red	Red	Yellow	0.6
4	White	Green	Blue	0.4
5	Blue	Red	Blue	0.5
6	White	Yellow	Red	0.7
7	Yellow	Green	Blue	0.4
8	Yellow	Blue	White	0.1
9	Green	Green	Red	0.9
10	Blue	White	Blue	0.3

It may be noticed that the records are grouped according the similarity for each key or group of keys. i.e., the records are grouped as follows: For Xr_1, the grouping are {(Yellow,{1,7,8}),(Red, {2,3}), (White, {4,6}), (Blue, {5,10}), (Green,{9})}. For X_2, the grouping are {(Red,{1,3,5}), (Yellow,{2,6}), (Green,{4,7,9}), (Blue,{8}), (White,{10})} and for X_3, we obtain {(Yellow, {1,3}), (Green,{2}), (Blue,{4,5,7,10}), (Red,{6,9}), (White,{8})}.

The above example is implemented in C by using X_2 as the key and the threshold as 0.35 using the threshold as 0.35 and we obtain the lower index of 2 as 51



IJIAET
International Journal of Innovative Analyses and Emerging Technology
| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1



4. Upper Indices in a Decision Table consisting Fuzzy Decision Attribute

In this section, we propose an algorithm to compute an index using upper parameterized rough approximations. In this algorithm, a fuzzy input A is considered and using a single threshold. For each element, the upper index is obtained using square and square root functions. Also, we illustrate the algorithm for a decision table with a fuzzy decision attribute.

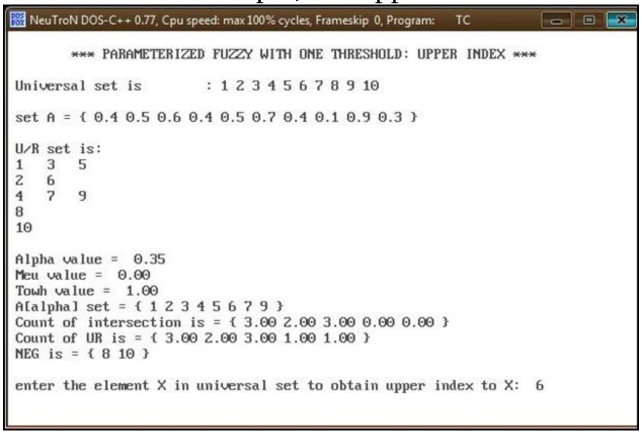
4.1 Algorithm for Upper index of an element

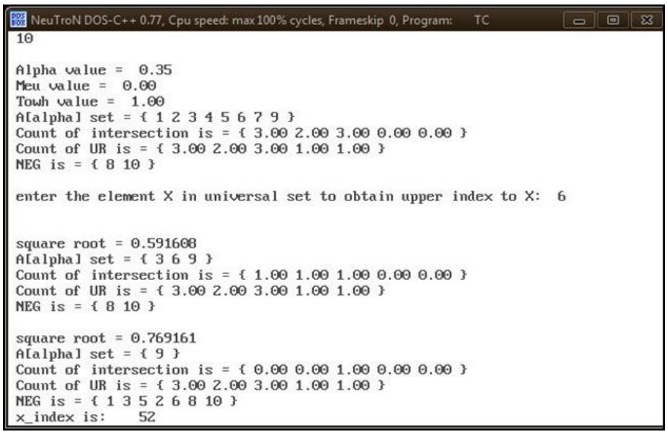
```
Algorithm (alpha, A, x)
//For obtaining Upper index of x an element of universe of discourse
//Algorithm returns the Upper index
1.   Let x_ind be an integer initialized to M
2.   Let K be the equivalence class containing x.
3.   If  U(y)=0 for all y belongs to K
Begin
       x_ind=-x_ind
       goto 7
End
4.   If  U(y)=1 for all y belongs to K
       goto 7
5.   Compute "NEG of A[alpha]"
6.   If " x belongs to NEG of A[alpha]"
While ("x belongs to NEG of A[alpha]")
       Begin
  alpha= sqr(alpha) //square of alpha
x_ind=x_ind-1
  Compute "NEG of A[alpha]"
End
else

While ("x NOT belongs to NEG of A[alpha]")
       Begin
  alpha= sqrt(alpha) //square root of alpha
x_ind=x_ind+1
```

Compute "NEG of A[alpha]"
End
7. Return x_ind

4.2 Experimental Results

For the above example, the upper indices of 6 is computed as 52.


```
NeuTroN DOS-C++ 0.77, Cpu speed: max 100% cycles, Frameskip 0, Program: TC

        *** PARAMETERIZED FUZZY WITH ONE THRESHOLD: UPPER INDEX ***

Universal set is        : 1 2 3 4 5 6 7 8 9 10

set A = { 0.4 0.5 0.6 0.4 0.5 0.7 0.4 0.1 0.9 0.3 }

U/R set is:
1   3   5
2   6
4   7   9
8
10

Alpha value =  0.35
Meu value =  0.00
Towh value =  1.00
A[alpha] set = { 1 2 3 4 5 6 7 9 }
Count of intersection is = { 3.00 2.00 3.00 0.00 0.00 }
Count of UR is = { 3.00 2.00 3.00 1.00 1.00 }
NEG is = { 8 10 }

enter the element X in universal set to obtain upper index to X:  6
```


```
NeuTroN DOS-C++ 0.77, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
10

Alpha value =  0.35
Meu value =  0.00
Towh value =  1.00
A[alpha] set = { 1 2 3 4 5 6 7 9 }
Count of intersection is = { 3.00 2.00 3.00 0.00 0.00 }
Count of UR is = { 3.00 2.00 3.00 1.00 1.00 }
NEG is = { 8 10 }

enter the element X in universal set to obtain upper index to X:  6


square root = 0.591608
A[alpha] set = { 3 6 9 }
Count of intersection is = { 1.00 1.00 1.00 0.00 0.00 }
Count of UR is = { 3.00 2.00 3.00 1.00 1.00 }
NEG is = { 8 10 }

square root = 0.769161
A[alpha] set = { 9 }
Count of intersection is = { 0.00 0.00 1.00 0.00 0.00 }
Count of UR is = { 3.00 2.00 3.00 1.00 1.00 }
NEG is = { 1 3 5 2 6 8 10 }
x_index is:    52
```

5. Parameterized Rough Indices in a Decision Table consisting Fuzzy Decision Attribute

In this section, by hybridizing the algorithms described in sections 3 and 4, parameterized rough indices are obtained for each element of the Universe of discourse. Similar to the above algorithms, by applying square and/ or square root functions on the threshold of the fuzzy input A, the parameterized rough indices are obtained accordingly. The algorithm is illustrated for a decision table with a fuzzy decision attribute.

5.1 Algorithm for Rough index of an element

Algorithm (alpha, A, x)
//for obtaining index of x an element of universe of discourse
//Algorithm returns the index
1. Let x_ind be an integer initialized to M
2. Let K be the equivalence class containing x.
3. If U(y)=0 for all y belongs to K
Begin

IJIAET International Journal of Innovative Analyses and Emerging Technology
| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1

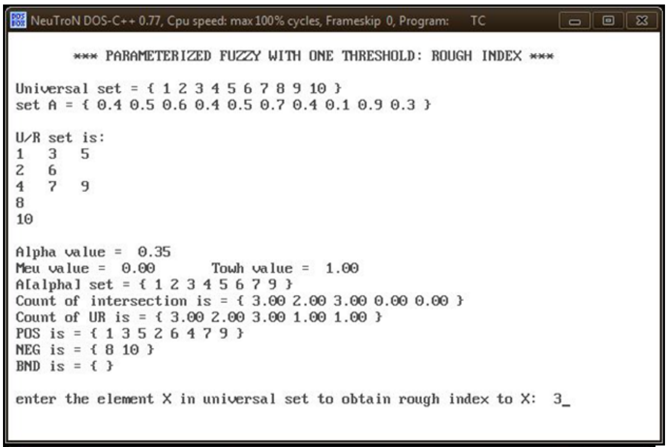
```
      x_ind = -x_ind
      goto 7
End
4.    If  U(y)=1 for all y belongs to K
      goto 7
5.    Compute POS of A[alpha], NEG of A[alpha], BND of A[alpha]
6.    If  " x belongs to POS of A[alpha]"
While (" x belongs to POS of A[alpha]")
      Begin
alpha= sqrt(alpha) //square root of alpha
x_ind=x_ind +1
  Compute POS of A[alpha]
End
else
7.    If  " x belongs to NEG of A[alpha]"

While ("x belongs to NEG of A[alpha]")
      Begin
alpha= sqr(alpha) //square of alpha
x_ind=x_ind-1
  Compute "NEG of A[alpha]"
End
else
      Begin
beta = alpha
Compute NEG of A[beta]
while("x NOT belongs to(POS of A[alpha] U NEG of A[beta]")
Begin
alpha = sqr(alpha) // square of alpha
beta = sqrt(beta) // square root of beta
compute POS of A[alpha] U NEG of A[beta]
x_ind=x_ind+1
End
If "x belongs to POS of A[alpha]"
x_ind = -x_ind
End
8.    Return x_ind
```

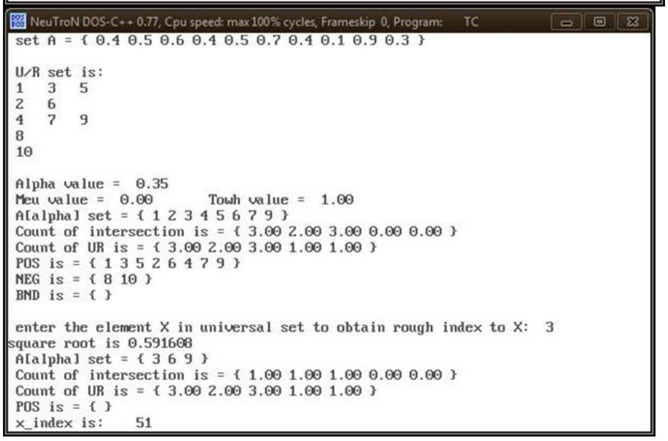
5.2 Experimental Results

For the above example, the parameterized rough indices of 3 is computed as 51.

IJIAET International Journal of Innovative Analyses and Emerging Technology

| e-ISSN: 2792-4025 | http://openaccessjournals.eu | Volume: 1 Issue: 1





6. Conclusion

In this paper, we implemented parameterized rough sets based algorithms to compute indices for the objects in fuzzy decision tables using a threshold.

References

1. B.Krishnaveni, V.Syamala, D.Latha, G.Ganesan; *Characterization of Information Systems with Fuzzy and Intuitionistic Fuzzy Decision Attributes*, 8th International Conference on Advanced Software Engineering & Its Applications, 2015, Korea, pp: 53-58, IEEE, 2015
2. D.Latha, D.Rekha, K.Thangadurai, G.Ganesan; *Probabilistic Rough Classification in Information Systems under Intuitionistic Fuzziness*, International Journal of Warehousing & Mining,, Vol:3, Issue: 2, pp: 82-86, 2013 (https://www.ijritcc.org)
3. G.Ganesan, et.al.; *Rough Index in Information System with Fuzziness in Decision Attributes*, International journal of Fuzzy Mathematics, Vol: 17, No: 1, pp: 183-190, 2008
4. Ganesan G, et.al., *Feature Selection using Fuzzy Decision Attributes*, Journal of INFORMATION, Vol 9, No3, pp:381-394, 2006
5. Ganesan G et.al., *Rough Set: Analysis of Fuzzy Sets using thresholds*, Computational Mathematics, Narosa, India,. pp: 81-87, 2005
6. Ganesan G, et.al., *An overview of rough sets*, proceedings of the National Conference on the emerging trends in Pure and Applied Mathematics, Palayamkottai, India, pp: 70-76, 2005
7. Zdzislaw Pawlak, *Rough Sets-Theoretical Aspects and Reasoning about Data*, Kluwer Academic Publications, 1991

8. Zdzislaw Pawlak, *Rough sets*, International Journal of Computer and Information Sciences, 11, 341-356, 1982.