Progressive re-sampling for better rule set discovery

Hyontai Sug
Division of Computer Eng., Dongseo University, Busan 47011, Korea
sht@gdsu.dongseo.ac.kr

Abstract. Rough set theory-based knowledge discovery systems have the good property of discovering rule sets solely based on data so that objectivity in the discovered rule sets is very good. But, sometimes this property may hinder to find better knowledge models, depending on the composition of target data sets, especially when the data sets have key-like attributes, or data sets have some attributes having poor dependency to class attributes. Key-like attributes have minutely detailed values and the correspondence between conditional attributes and class attributes is precise. This paper shows how we can surmount the problem thru experiment. Two data sets are adapted for the experiment. A data set that has attributes of key-like and poor dependency was selected. And the other data set that does not have such properties is used to present a counter case. The experiment showed very good results and we found precautions when we apply rough set-based knowledge discovery systems. In addition, the experiment confirmed the fact that adding more data for target data set may increase the accuracy of knowledge models using rough set.

1. Introduction
Rough set theory is invented to characterize imperfect knowledge, and is considered an alternative or supplementary for fuzzy set theory [1]. A good point of rough set theory is that it does not need any pre-assumption about knowledge domain, while fuzzy set theory needs to set the value of possibility that human expert should assign. If we have good knowledge about target domain, assigning such possibility or the grade of membership may not be difficult. But if we have very little knowledge about the target domain, such task can be very difficult, and many target domains for knowledge discovery are known to have such property [2]. In this sense rough set theory is a very good tool for such task, because knowledge discovery based on rough set theory is solely based on data. As an effort to compensate the subjectivity of assigning membership grade by human, there have been some efforts that try to use rough set theory-based technique. For example, Chakraborty applied rough set to refine membership functions [3], and Meng et al. tried to combine fuzzy set, soft set, and rough set together to generate so called soft fuzzy rough set [4]. On the other hand, because rough sets may give objective ability of judgment on dependency between attributes, some researchers have used rough set-based method as a pre-processing technique to determine a good subset of attributes from the raw data. For example, Coaquira and Acuña applied rough set-based dependency analysis between attributes, and selected some part of attributes [5]. Huijian and Feipeng applied principle component analysis that is combined with rough set-based method to select a subset of attributes for high dimensional business data before they supply the data to their SVM [6]. Some others use other
technique as a pre-processing technique before they try to find knowledge models using rough set-based method. For example, Mahapatra et al. applied correlation analysis between attributes before they try to find rough set-based rules to find out whether dependency between attributes exist [7]. Therefore, these works show that rough set theory can be a good tool to determine dependency between attributes or to find knowledge models.

But, even though trying to find knowledge model from data themselves only is one of good points of rough set based knowledge discovery systems, sometimes this property may hinder to find better knowledge models depending on the composition of given data set, especially data have key-like attributes, or data have some conditional attributes that have poor dependency to class attributes. Key-like attributes have minutely detailed values and the correspondence between conditional attributes and class attributes is precise. In section 2 we will discuss the method to surmount the problem, and result of experiments will be discussed, and in section 3 conclusions will be presented.

2. The method and experiment

Rough set theory was suggested by Pawlak [8] in early ‘80s, and the theory has attracted much attention, because the theory is based on very solid mathematical background, while most other artificial intelligence technologies rely on imperfect or satisfactory methods like heuristics. For details about rough set theory and related technologies, please refer to [9]. Because rough set-based rule discovery systems try to find rules solely based on data, it is highly possible that data instances of higher occurrence as well as their attributes will be treated more importantly in the process of rule discovery to find more accurate rules.

So, in order to see progressively how often each attribute is referred in the found rule set, re-sampling will be performed, and rule set discovery process will be repeated until there is no further improvement in accuracy in our experiment. Note that supplying more training instances may increase the accuracy of trained knowledge models in general. For the experiment two data sets, called Hayes-Roth and Iris, are used that can be found in the UCI machine learning repository [10]. Hayes-Roth data set is selected because it has a key-like attribute, while Iris data set doesn’t have. All the experiments are based on 10-fold cross validation. The used rough set-based rule discovery system is MODLEM which is one of successful implementation of rough set-based rule discovery system [11, 12].

2.1. Hayes-Roth data set

Hayes-Roth data set contains data of exemplars that were used to build property set models of classification for concept learning [13]. The data set contains 5 conditional attributes, name, hobby, age, educational level, and marital status. All the conditional attributes except name attribute have a few nominal values. The name attribute in the original data set is transformed from numerical to nominal values because name attribute usually doesn’t have numerical meaning. There are 3 classes, and total number of instances is 132.

The following 3 rules with accuracy of 0% in 10-fold cross validation were found from the original data set. Note that experiment is performed in 10-fold cross validation, so even the training stage has found 100% secure rules as we can see in the rule description, testing generates bad result.

Rule 1: (name in {1, 5, 7, 9, 15, 16, 17, 19, 20, 28, 30, 35, 36, 40, 42, 45, 50, 51, 54, 57, 59, 66, 67, 68, 69, 74, 81, 89, 91, 92, 93, 94, 95, 96, 97, 103, 105, 106, 108, 112, 114, 118, 119, 120, 121, 123, 126, 127, 130, 131, 132}) => (class = 1)  (51/51, 100%)

Rule 2: (name in {2, 6, 10, 14, 21, 23, 24, 25, 26, 29, 31, 32, 33, 34, 37, 39, 41, 49, 52, 53, 56, 58, 60, 62, 63, 65, 70, 71, 73, 76, 77, 78, 79, 80, 82, 84, 85, 86, 88, 90, 98, 99, 104, 109, 111, 113, 115, 116, 117, 124, 129}) => (class = 2)  (51/51, 100%)
Rule 3: (name in \{3, 4, 8, 11, 12, 13, 18, 22, 27, 38, 43, 44, 46, 47, 48, 55, 61, 64, 72, 75, 83, 87, 100, 101, 102, 107, 110, 122, 125, 128\}) \Rightarrow (\text{class} = 3) \quad (30/30, 100\%)

The rule set above reveals the property of rough set-based rule set generation method very well—rule set generation is frank, and solely based on data, but the rules are over-fitted as we see that name is unique for each data instance so that very poor generalization.

The following 3 rules with accuracy of 83.333\% were found with re-sampling rate of 200%.

Rule 1: (name in \{1, 15, 16, 20, 89, 120, 36, 42, 51, 68, 81, 108, 112, 118, 121, 126, 30, 40, 57, 66, 74, 92, 94, 96, 103, 127, 17, 19, 28, 50, 54, 59, 69, 91, 93, 95, 97, 106, 114, 119, 123, 130, 131, 132\}) \Rightarrow (\text{class} = 1) \quad (94/94, 100\%)

Rule 2: (name in \{58, 62, 76, 90, 111, 14, 21, 23, 52, 63, 85, 124, 6, 34, 39, 60, 2, 25, 31, 41, 70, 71, 79, 82, 98, 99, 109, 113, 129, 10, 24, 26, 32, 33, 73, 78, 84, 88, 104, 116, 117\}) \Rightarrow (\text{class} = 2) \quad (105/105, 100\%)

Rule 3: (name in \{100, 46, 102, 110, 13, 22, 27, 38, 64, 72, 107, 122, 3, 4, 12, 43, 44, 55, 87, 101, 8, 18, 47, 48, 75, 83, 125, 128\}) \Rightarrow (\text{class} = 3) \quad (65/65, 100\%)

Seven names are dropped from rule 1 above compared to the rule 1 from the original data – 5, 7, 9, 35, 45, 67, and 105. Ten names are dropped from rule 2 above compared to the rule 2 from the original data – 29, 37, 49, 53, 56, 65, 77, 80, 86, and 115. Two names are dropped from rule 3 above compared to the rule 3 from the original data – 11, and 61. Even though the accuracy of the rule set is improved at lot, 44 instances are unclassified among 264 instances with the above rule set. Table 1 shows corresponding confusion matrix.

| Actual class | Predicted class |
|--------------|----------------|
| 1            | 76 0 0         |
| 2            | 0 89 0         |
| 3            | 0 0 55         |

As re-sampling rate was increased, the accuracy of found rule set increased with slight change in the condition part of the rule set. That is, a few more names were added as the re-sampling rate was increased. When re-sampling rate reached 800\%, the accuracy became 100\%.

From the experiments performed with various re-sampling rates we have found the fact that enough number of instances may improve the accuracy of the rough set based knowledge models for the data set. But, because attribute ‘name’ has key-like property, this fact hinders the generalization of rules, so we run the algorithm again to see the effect of eliminating ‘name’ attribute.

Thirteen rules were found with the accuracy of 78.7879\% for the original data set after dropping name attribute. The following is the result.

Rule 1: (age in \{1\}) & (edu in \{1\}) & (marital in \{2, 3\}) \Rightarrow (\text{class} = 1) \quad (11/11, 30.56\%)

Rule 2: (marital in \{1\}) & (age in \{1\}) & (edu in \{2, 3\}) \Rightarrow (\text{class} = 1) \quad (11/11, 30.56\%)

Rule 3: (edu in \{1\}) & (marital in \{1\}) & (age in \{2, 3\}) \Rightarrow (\text{class} = 1) \quad (11/11, 30.56\%)
Rule 4: (age in \{3\}) & (edu in \{1, 3\}) & (marital in \{1, 3\}) \Rightarrow (class = 1) \quad (3/3, 8.33\%)

Rule 5: (edu in \{3\}) & (marital in \{3\}) & (age in \{1\}) \Rightarrow (class = 1) \quad (1/1, 2.78\%)

Rule 6: (age in \{2\}) & (edu in \{2\}) & (marital in \{1, 3\}) \Rightarrow (class = 2) \quad (11/11, 30.56\%)

Rule 7: (marital in \{2\}) & (age in \{2\}) & (edu in \{1, 3\}) \Rightarrow (class = 2) \quad (11/11, 30.56\%)

Rule 8: (edu in \{2\}) & (marital in \{2\}) & (age in \{1, 3\}) \Rightarrow (class = 2) \quad (3/3, 8.33\%)

Rule 9: (age in \{3\}) & (marital in \{2, 3\}) & (edu in \{2, 3\}) \Rightarrow (class = 2) \quad (11/11, 30.56\%)

Rule 10: (edu in \{3\}) & (age in \{2\}) & (hobby in \{1\}) \Rightarrow (class = 2) \quad (1/1, 2.78\%)

Rule 11: (age in \{4\}) \Rightarrow (class = 3) \quad (12/12, 40\%)

Rule 12: (edu in \{4\}) \Rightarrow (class = 3) \quad (12/12, 40\%)

Rule 13: (marital in \{4\}) \Rightarrow (class = 3) \quad (12/12, 40\%)

Table 2 shows corresponding confusion matrix.

| Actual class | Predicted class | 1 | 2 | 3 |
|--------------|----------------|---|---|---|
| 1            |                | 39| 12| 0 |
| 2            |                | 12| 39| 0 |
| 3            |                | 2 | 2 | 26|

Re-sampling rate 200% generated rule set of 16 rules with accuracy of 87.5%, and re-sampling rate 300% generated rule set of 18 rules with accuracy of 87.3737%. Table 3 shows the total number of covering instances for each attribute that constitutes each rule. For example, there are 11 instances each for attribute ‘age’, ‘edu’, and ‘marital’ covered by rule 1 in the above.

| Re-sampling rate | Total instances covered for each attribute |
|------------------|-------------------------------------------|
|                  | age | edu | marital | hobby |
| 100%(original)   | 86  | 86  | 86      | 1     |
| 200%             | 236 | 280 | 260     | 36    |
| 300%             | 397 | 387 | 399     | 59    |

Based on the summary in table 3 attribute hobby may be dropped because of the least contribution in rule set formation. After dropping name and hobby attributes, we found the following 11 rules with accuracy of 84.0909%. Note that this accuracy is comparable to the best accuracy yet found that is
based on lazy algorithm achieving accuracy of 84% [14]. But, lazy algorithms have the limitation of not being able to generate rules;

Rule 1: (marital in \{1, 3\}) & (edu in \{1\}) & (age in \{2, 3, 1\}) \Rightarrow (\text{class} = 1) \quad (23/23, 35.94\%)

Rule 2: (age in \{1, 3\}) & (marital in \{1\}) & (edu in \{2, 3\}) \Rightarrow (\text{class} = 1) \quad (22/22, 34.38\%)

Rule 3: (age in \{1\}) & (edu in \{1\}) & (marital in \{2\}) \Rightarrow (\text{class} = 1) \quad (10/10, 15.63\%)

Rule 4: (edu in \{3\}) & (age in \{1\}) & (marital in \{2, 3\}) \Rightarrow (\text{class} = 1) \quad (9/11, 14.06\%)

Rule 5: (marital in \{2\}) & (edu in \{3, 2\}) & (age in \{1, 3, 2\}) \Rightarrow (\text{class} = 2) \quad (23/23, 37.1\%)

Rule 6: (age in \{2\}) & (marital in \{2, 3\}) & (edu in \{1, 2, 3\}) \Rightarrow (\text{class} = 2) \quad (22/23, 35.48\%)

Rule 7: (edu in \{2\}) & (age in \{2\}) & (marital in \{1\}) \Rightarrow (\text{class} = 2) \quad (10/10, 16.13\%)

Rule 8: (age in \{3\}) & (edu in \{2\}) & (marital in \{1, 3\}) \Rightarrow (\text{class} = 2) \quad (8/11, 12.9\%)

Rule 9: (age in \{4\}) \Rightarrow (\text{class} = 3) \quad (12/12, 40\%)

Rule 10: (edu in \{4\}) \Rightarrow (\text{class} = 3) \quad (12/12, 40\%)

Rule 11: (marital in \{4\}) \Rightarrow (\text{class} = 3) \quad (12/12, 40\%)

Table 4 shows corresponding confusion matrix, and there is no unclassified instances.

| Actual class | Predicted class |
|--------------|----------------|
|              | 1   | 2   | 3   |
| 1            | 43  | 8   | 0   |
| 2            | 11  | 40  | 0   |
| 3            | 1   | 1   | 28  |

From the experiment above we see that dropping attribute hobby effects that other attributes may have more data instance, and the rough set rule discovery system does not care about minute details due to attribute hobby, resulting in better accuracy. Similar reason can be explained for attribute name also.

2.2. Iris data set

Iris data set contains data of 3 class of iris plant, iris Setosa, iris Versicolor, iris Virginica for 150 instances. There are 4 attributes, attribute sL contains sepal length in cm, attribute sW contains sepal width in cm, attribute pL contains petal length in cm, attribute pW contains petal width in cm. The problem domain of the data set is known as pattern recognition problem. Many works have been done to find better classification models for the data set since the data set became public in 1988 [15].

The following 12 rules with accuracy of 96% were found for the original data set. The accuracy is comparable to other researchers’ one of best data mining models, for example, ensemble of Baysian
networks called LogitBoostNB achieved the accuracy of 94.87% in 10 fold cross validation [16], but it
is not in easy to read form like ours.

Rule 1: \((pL < 2.45) \Rightarrow \text{(class = Iris-setosa)}\) \((50/50, 100\%)

Rule 2: \((sW < 2.95) \& \& (pW < 1.35) \& \& (sL >= 4.85) \Rightarrow \text{(class = Iris-versicolor)}\) \((26/26, 52\%)

Rule 3: \((pL \in [2.45, 4.75]) \& \& (sL >= 4.95) \Rightarrow \text{(class = Iris-versicolor)}\) \((43/43, 86\%)

Rule 4: \((sW < 2.55) \& \& (sL >= 6.05) \& \& (pL < 4.95) \Rightarrow \text{(class = Iris-versicolor)}\) \((3/3, 6\%)

Rule 5: \((sL >= 6.55) \& \& (pL < 5.05) \Rightarrow \text{(class = Iris-versicolor)}\) \((8/8, 16\%)

Rule 6: \((sW \in [2.65, 2.75]) \& \& (pW < 1.65) \Rightarrow \text{(class = Iris-versicolor)}\) \((5/5, 10\%)

Rule 7: \((sL \in [5.85, 5.95]) \& \& (sW >= 3.05) \Rightarrow \text{(class = Iris-versicolor)}\) \((1/1, 2\%)

Rule 8: \((pL >= 5.15) \Rightarrow \text{(class = Iris-virginica)}\) \((34/34, 68\%)

Rule 9: \((pW >= 1.75) \& \& (sW < 3.15) \Rightarrow \text{(class = Iris-virginica)}\) \((32/32, 64\%)

Rule 10: \((pL >= 4.95) \& \& (pW < 1.55) \Rightarrow \text{(class = Iris-virginica)}\) \((3/3, 6\%)

Rule 11: \((pW >= 1.85) \Rightarrow \text{(class = Iris-virginica)}\) \((34/34, 68\%)

Rule 12: \((pW >= 1.65) \& \& (sL < 4.95) \Rightarrow \text{(class = Iris-virginica)}\) \((1/1, 2\%)

Table 5 shows corresponding confusion matrix.

| Actual class | Predicted class |
|--------------|----------------|
| Iris-sotosa  | 50 0 0         |
| Iris-versicolor | 0 47 3 |
| Iris-versinica | 0 3 47 |

As re-sampling rate was increased, the accuracy of found rule set was increased with slight change in the composition of found rules. Table 6 shows the summary.

| Re-sampling rate | Accuracy  | Number of rules |
|-----------------|-----------|-----------------|
| 100% (original) | 96%       | 12              |
| 300%            | 99.6667%  | 12              |
| 300%            | 99.7788%  | 12              |
| 400%            | 100%      | 13              |
| 500%            | 100%      | 12              |
The followings are the 12 rules with accuracy of 100% for the data set of re-sampling rate of 500%.

Rule 1: \((pL < 2.45) \Rightarrow \) (class = Iris-setosa)  \((236/236, 100\%)\)

Rule 2*: \((pL \in [2.45, 4.45]) \Rightarrow \) (class = Iris-versicolor)  \((154/154, 58.11\%)\)

Rule 3: \((pL \in [2.45, 4.75]) \& (sL \geq 4.95) \Rightarrow \) (class = Iris-versicolor)  \((226/226, 85.28\%)\)

Rule 4*: \((sW < 2.55) \& (pL < 4.95) \& (sL \geq 4.95) \Rightarrow \) (class = Iris-versicolor)  \((62/62, 23.4\%)\)

Rule 5: \((sL \geq 6.55) \& (pL < 5.05) \Rightarrow \) (class = Iris-versicolor)  \((38/38, 14.34\%)\)

Rule 6: \((sW \in [2.65, 2.75]) \& (pW < 1.65) \Rightarrow \) (class = Iris-versicolor)  \((35/35, 13.21\%)\)

Rule 7: \((sL \in [5.85, 5.95]) \& (sW \geq 3.05) \Rightarrow \) (class = Iris-versicolor)  \((7/7, 2.64\%)\)

Rule 8: \((pL \geq 5.15) \Rightarrow \) (class = Iris-virginica)  \((171/171, 68.67\%)\)

Rule 9*: \((pW \geq 1.75) \& (sW < 3.05) \Rightarrow \) (class = Iris-virginica)  \((130/130, 52.21\%)\)

Rule 10: \((pL \geq 4.95) \& (pW < 1.55) \Rightarrow \) (class = Iris-virginica)  \((17/17, 6.83\%)\)

Rule 11: \((pW \geq 1.85) \Rightarrow \) (class = Iris-virginica)  \((160/160, 64.26\%)\)

Rule 12: \((pW \geq 1.65) \& (sL < 4.95) \Rightarrow \) (class = Iris-virginica)  \((5/5, 2.01\%)\)

Different rules from the rules from the original data set are indicated by * on the rule number in the above rule set. Table 7 is the corresponding confusion matrix for the rule set of 12 rules.

### Table 7 Confusion matrix

| Actual class | Iris-setosa | Iris-versicolor | Iris-virginica |
|--------------|-------------|----------------|---------------|
| Iris-setosa  | 236         | 0              | 0             |
| Iris-versicolor | 0         | 265            | 0             |
| Iris-virginica | 0         | 0              | 249           |

In order to see contribution of each attribute in rule generation, the occurrence of each attribute in the rules are counted, table 8 shows the result.

### Table 8 Total number of covering instances

| Re-sampling rate | Total instances covered for each attribute |
|------------------|-------------------------------------------|
| pL               | sW | pW | sL |
| 100% (original)  | 141 | 67 | 101 | 82 |
| 200%             | 288 | 53 | 114 | 84 |
The rank of attributes is like pL, pW, sL, and sW. But, there is no much difference in the importance of attributes with respect to data instance coverage for each attribute in the found rules. In order to see dropping some attributes from the original data set, we may drop sW attribute. After dropping sW attributes, we found 11 rules with slightly less accuracy of 94.6667% than the result from the original data set. Anyway this result is comparable to other researcher’s previous best accuracy [16]. All in all, because the rough set rules from the original data is yet the best and the re-sampled data showed better results, from these two results-dropping an attribute and re-sampling for over-sampling, we can conclude that we may improve the accuracy of the knowledge model of Iris data set, if we can have more data, not by dropping attributes.

3. Conclusions
Rough set theory has a very good property that related knowledge discovery tools can analyse data solely based on data themselves. So, a lot of research have been done and reported success of related works. But, even though trying to find knowledge models from data themselves only is one of good points of rough set based knowledge discovery systems, sometimes this property may hinder to find better knowledge models, depending on the composition of given data sets, especially when data have key-like attributes, or data have attributes that have a very small portion of dependency to class attributes. Key-like attributes have minutely detailed values and the correspondence between conditional attributes and class attributes is precise. In order to see how we can surmount the problem a data set that has such properties was selected and experimented. And the other data set that does not have such properties is used to show a counter case. The experiment with the two data sets showed very good results and also revealed precautions when we apply rough set-based knowledge discovery system. Moreover, the experiments confirm the fact that supplying more target data may increase the accuracy of knowledge models of rough set.

References
[1] Zhang Q, Xie Q and Wang G 2016 CAAI Tr. On Intelligence Technology 1 pp 323-333
[2] Cao L 2010 IEEE Tr. On Knowledge and Data Eng. 22 pp 755-769
[3] Chakraborty M K 2014 Int. J. of Approximate Reasoning 55 pp 402-411
[4] Meng D, Zhang X, and Qin K 2011 Computer and Mathematics with Applications 62 pp 4635-4645
[5] Coaquira F and Acuña E 2007 Proc. World Congress on Engineering and Computer Science (San Francisco) pp 707-712
[6] Huijian X and Feipeng G 2012 J. of Convergence Information Technology 7 pp 132-140
[7] Mahapatra S, Sreekumar, and Mahapatra SS 2010 IIMB Management Review 22 pp 16-24
[8] Pawlak Z 1982 Int. J. of Parallel Programming 11 pp 341-356
[9] Rissinos and Lambert-Torres G 2009 Data Mining and Knowledge Discovery in real Life Applications ed J Ponce and A Karahoca p 438
[10] Dua D and Karra Taniskidou E 2017 UCI Machine Learning Repository [http://archive.ics.uci.edu/ml] Irvine, CA, University of California, School of Information and Computer Science
[11] Stefanowski J 1998 Rough Sets in Data Mining and Knowledge Discovery ed L Polkowski and A Skowron pp 500-529
[12] Grzymala-Busse J W and Stefanowski J 2001 Int. J. of Intelligent Systems 16 pp 29-38
[13] Hayes-Roth B and Hayes-Roth F 1977 J. of Verbal Learning and Verbal Behavior 16 pp 321-
[14] Gaber M 1998 *A Lazy Model-Based Approach to On-Line Classification*, Master Thesis, Simon Fraser University p 114
[15] Duda R O and Hart P E 1973 *Pattern Analysis and Scene Analysis* (John Wiley & Sons) p 218
[16] Kotsiantis S B and Pintelas P E 2005 *Informatica* 29 53-59