Feature Selection with Adjustable Criteria

JingTao Yao and Ming Zhang

Department of Computer Science, University of Regina,
Regina, Saskatchewan, Canada S4S 0A2
jtyao@cs.uregina.ca

Abstract. We present a study on a rough set based approach for feature selection. Instead of using significance or support, Parameterized Average Support Heuristic (PASH) considers the overall quality of the potential set of rules. It will produce a set of rules with balanced support distribution over all decision classes. Adjustable parameters of PASH can help users with different levels of approximation needs to extract predictive rules that may be ignored by other methods. This paper finetunes the PASH heuristic and provides experimental results to PASH.

1 Introduction

One of the main research challenges of information analyzing from large databases is how to reduce the complexity of the data. One faces two characteristics of complexity, namely, the curse of dimensionality and the peaking phenomenon. The curse of dimensionality refers to the fact that the complexity grows exponentially with the dimension. Therefore, the time required to generate rules will increase dramatically with the number of features [2]. The peaking phenomenon says that if the number of training instances is relatively smaller than the number of features, it will degrade the accuracy of prediction [14]. Feature selection techniques aim at simplifying complexity of data by reducing the number of unnecessary, irrelevant, or unimportant features. The additional benefits of doing feature selection include improving the learning efficiency and increasing predicative accuracy.

The ability to process insufficient and incomplete information makes rough set theory a good candidate for classification and feature selection [3]. In fact, rough set theory has a very close tie with feature selection. Similar to the concept of keys in database, the reduct represents the minimal set of non-redundant features that are capable of discerning objects in a information table. Another concept, the core, which is the intersection of all reducts, represents the set of indispensable features. Many researchers have presented their study on using rough set theory for feature selection [4,7,11,16,17]. Normally, the measures of necessity of the features are calculated by the functions of lower and upper approximations. These measures are employed as heuristics to guide the feature selection processes. For example, Hu proposes a heuristic in favors of significant features, i.e., features causing the faster increase of the positive region [7]. The heuristic of Zhong *et al.* considers the positive region as well as the support of

D. Ślęzak et al. (Eds.): RSFDGrC 2005, LNAI 3641, pp. 204–213, 2005.

rules [17]. However, it may not be sufficient by considering only the significant or support factors. It may be useful to consider the overall quality of the set of potential rules. The new heuristic function called Average Support Heuristic is a study towards this direction [16]. To further develop this idea, 100% support may not be needed for all applications. Parameterized Average Support Heuristic (PASH) is the result of this improvement.

We will reformat and fine-tune the PASH heuristic in this paper. The experimental results will also be presented. The organization of this paper is as follows: Section 2 studies feature selection in brief term. Section 3 reviews rough set based feature selection methods. The PASH heuristic is presented in Section 4 and experimental results in Section 5. Finally, the paper ends with concluding remarks.

2 Brief of Feature Selection

Feature selection is considered as one of the important research topics of machine learning [6]. In many applications, especially in the age of an information explosion, one collects many features that are potentially useful. However, all of these features may not be useful or relevant to one's classification, forecasting, or clustering objects. Therefore, choosing a subset of the original features will often lead to better performance. Features may be classified as significant, relevant, dependent and useless according to their importance to the application. The goal of feature selection is to find the optimal subset of features that satisfy certain criteria. For instance, although there may be dozens of features (make, brand, year, weight, length, hight, engine size, transmission, colour, owner, price, etc.) available when one purchases a second hand vehicle, one may only read a handful of important features (e.g., make, year, engine, colour and price) that meet one's needs.

Studies show that there are at least four criteria to judge a feature selection method [5], such as,

- Find the minimal feature subset that is necessary and sufficient to decide the classes;
- Select a subset of M features from a set of N features, $M < N$, such that the value of a criterion function is optimized over all subsets of size M;
- Improve prediction accuracy or decrease the size of the feature subset without significantly decreasing prediction accuracy of the classifier built using only the selected features;
- Select a small subset such that the resulting class distribution, given only the values for the selected feature, is as close as possible to the original class distribution given all feature values.

It is observed that each of the criterion considers two parameters, namely, the size of the selected feature subset and the accuracy of the classifier induced using only the selected features. No matter what criterion is employed, one has to define an evaluation measure to express the chosen criterion. The evaluation measure must be able to reflect both of the parameters. From a machine learning point of view, the feature selection problem is in fact a search problem. The optimal feature subset is one that maximizes the value of an evaluation measure. Therefore, the general search principles apply to feature selection.

An exhaustive search of 2^n possible subsets for a feature set of size n is almost infeasible under most circumstances [6]. It could only be used in a domain where n is small. However, the needs for feature selection is limited in such cases. In random search, the candidate feature subset is generated randomly and each time the evaluation measure is applied to the generated feature subset to check whether it satisfies the criterion. This process repeats until one that satisfies the criterion is found. The process may stop when a predefined time period has elapsed or a predefined number of subsets have been tested. A random search algorithm worthwhile to mention is the LVF algorithm proposed by Liu and Setiono [12].

The third and most commonly used method is called the heuristic search, where a heuristic function is employed to guide the search [9,10]. The search is performed towards the direction that maximizes the value of a heuristic function. Heuristic search is an important search method used by the feature selection community. The rough set approaches for feature selection discussed in this article are heuristic search methods.

The exhaustive search is infeasible due to its high time complexity. The random and heuristic search reduce computational complexity by compromising performance. It is not guaranteed that an optical result can be achieved. They are not complete search techniques. However, if a heuristic function is monotonic, as the branch and bound method proposed by Narendra and Fukunaga, the optimal subset of features can be found muck quick than exhaustive search [13].

3 Evolution of Rough Sets Based Feature Selection

As we discussed above, reducts in a rough set represent sets with minimal number of features. These features are significant features. The most important features are those appearing in core, i.e., in every reduct. The measures of necessity of features are usually calculated based on the concept of lower and upper approximations. These measures are employed as heuristics to guide the feature selection process.

The concepts in the rough set theory can manifest the property of strong and weak relevance as defined in [8]. They can be used to define the necessity of features. There are at least three types of rough set based heuristics, namely the significance oriented method, the support oriented method, and average support heuristic appearing in literature. The heuristic in [7] favors significant features, i.e., features causing the faster increase of the positive region. Zhong's heuristic considers the positive region as well as the support of rules [17]. The Average Support Heuristic considers the overall quality of the potential set of rules rather than the support of the most significant rule [16].

3.1 Significance Oriented Methods

One of the simplest and earliest rough set based feature selection method is to use significance of features as the heuristic as studied by Hu [7]. The feature selection process selects the most significant feature at each step until the stop condition is satisfied. The most significant feature is the one that, by adding this feature, can cause the fastest increase of dependency between condition attributes and decision attribute, where the dependency reflects the relative size of positive region. In short, the significance oriented method always selects the feature that makes the positive region grow faster.

The significance oriented method is simple and the heuristic function can be computed with low time complexity. However, this method only considers one of the two factors in feature selection: the number of instances covered by the potential rules (the size of positive region). It ignores the second factor: the number of instances covered by each individual rule (the support of each rule.) Rules with very low support are usually of little use.

3.2 Support Oriented Methods

The support oriented method proposed Zhong *et al.* considers both factors [17]. This method selects features based on the composite metric: the size of consistent instance and the support of an individual rule. The heuristic function is defined as the product of the positive region and the support of the most significant rule, where the most significant rule is the one with the largest support. In the remaining part of the paper, we refer to Zhong's heuristic as the maximum support heuristic.

The maximum support heuristic is far from an ideal heuristic. It only considers the support of the most significant rule rather than the overall quality of the potential rules. Among the classes of the training instances, this method favors one of them. As a result, it will produce rules with a biased support distribution.

3.3 Average Support Heuristic

A newer heuristic function, called average support heuristic, was proposed recently [16]. The average support heuristic uses the average support of the rules to replace the highest support of the rule in the maximum support heuristic. The heuristic function is defined as the product of the positive region and the average support of the most significant rules over all decision classes, as follows:

$$F(R,a) = Card(POS_{R+\{a\}}(D))\times \tfrac{1}{n}\textstyle\sum_{i=1}^{n} S(R,a,d_i) \tag{1}$$

where

$$S(R,a,d_i) = MAXSize(POS_{R+\{a\}}(D=d_i)/IND(R+\{a\}))$$

is the support of the most significant rule for decision class $\{D=d_i\}$ and D is the decision attribute. The domain of D is $\{d_1,d_2,\ldots,d_n\}$. We call the second factor $\frac{1}{n}\sum_{i=1}^{n} S(R,a,d_i)$ the overall quality of potential rules. As the heuristic considers all the decision classes, the biased support distribution can be avoided.

4 Parameterized Average Support Heuristic

Completely ignoring the inconsistent instances of the information table, as the above heuristic functions do, is not a good strategy when the size of the boundary region increases [16]. Some useful predictive rules obtained from the boundary region might be lost in the result. The predictive rules hold true with high probability but are not necessarily 100%.

All the above heuristics are defined on the basis of the traditional lower approximation, the union of which includes only the consistent instances. In order to include the predictive rules, we give a broader concept of lower approximation, upon which a parameterized average support heuristic is defined.

The decision-theoretic rough set model and variable precision rough set model are two examples of non-traditional lower approximation [15,18]. They consider the information in the boundary region. However, the a priori probability of each decision class required by these models is usually unknown in the real world application. Furthermore, the pair of lower and upper limit certainty threshold parameters confines these models to information tables with only a binary decision attribute.

Our new lower approximation does not require known a priori probabilities of the decision classes and it is applicable to multi-valued decision attribute. Suppose we have an information table T, in which the domain of decision attribute D, denoted by V_D, contains n values, such that $V_D = \{d_1, d_2, \ldots, d_n\}$. Here we consider two different situations: (1) the a priori probabilities are unknown; and (2) the a priori probabilities are known.

4.1 Lower Approximation with Unknown a Priori Probability

When the a priori probabilities are unknown, we assume they are equal, i.e. $P(D = d_1) = P(D = d_2) = \cdots P(D = d_n)$. In this case, we define the lower approximation of class $\{D = d_i\}$ as follows:

$$R_*(D = d_i) = \bigcup\{E_j \in U/IND(R) : P(D = d_i|E_j) > P(D \neq d_i|E_j)\}, \quad (2)$$

where $P(D \neq d_i|E_j) = \sum_{k=1,k\neq i}^{n} P(D = d_k|E_j)$. The lower approximation of class $\{D = d_i\}$ is the set of such objects E_j in U that, given E_j, the probability of $D = d_i$ is greater than the probability of $D \neq d_i$. In other words, E_j is predictive of concept $D = d_i$ from $D \neq d_i$.

Since $P(D \neq d_i|E_j) = 1 - P(D = d_i|E_j)$, we can rewrite Equation 2 to Equation 3:

$$R_*(D = d_i) = \bigcup\{E_j \in U/IND(R) : P(D = d_i|E_j) > 0.5\}, \quad (3)$$

where $P(D = d_i|E_j)$ could be estimated by taking the ratio of
$Card(D = d_i \bigcap E_j)/Card(E_j)$.

When the decision attribute has fewer number of values, in the extreme case, the decision attribute is binary, that is, $|V_D| = 2$, Equation 2 may be too broad and degrade the performance. We can introduce a parameter $k(k \geq 1)$ to Equation 2 as follows:

$$R_*(D = d_i) = \bigcup\{E_j \in U/IND(R) : P(D = d_i|E_j) > k \times P(D \neq d_i|E_j)\}. \quad (4)$$

Equation 4 reflects that, given E_j, the concept $D = d_i$ is k times more probable than the concept $D \neq d_i$.

By replacing $P(D \neq d_i|E_j)$ with $1 - P(D = d_i|E_j)$, Equation 4 becomes

$$R_*(D = d_i) = \bigcup\{E_j \in U/IND(R) : P(D = d_i|E_j) > \tfrac{k}{k+1}\}. \quad (5)$$

As $k \geq 1 \Longrightarrow \frac{k}{k+1} \geq 0.5$, we can simplify Equation 5 as:

$$R_*(D = d_i) = \bigcup\{E_j \in U/IND(R) : P(D = d_i|E_j) > t(t \geq 0.5)\}. \quad (6)$$

Clearly, Equation 3 is a special case of Equation 6. Equation 6 guarantees that each object $E \in U$ is contained in at most one lower approximation, that is,

$$R_*(D = d_i) \bigcap R_*(D = d_j) = \phi, (i \neq j).$$

4.2 Lower Approximation with Known a Priori Probability

In the case that the a priori probabilities of decision classes are known, Equation 6 is too simple to be effective. Assume that the information table obtained from the training data can reflect the distribution of decision classes. The a priori probability of class $(D = d_i)$ could be estimated by

$$(D = d_i) = \frac{Card(D = d_1)}{Card(U)}.$$

We can modify Equation 6 to Equation 7:

$$\begin{aligned} R_*(D = d_i) = \bigcup \{E_j \in U/IND(R) : & \\ & \tfrac{P(D=d_i|E_j)}{P(D=d_i)} = MAX\{\tfrac{P(D=d_k|E_j)}{P(D=d_k)}, 1 \leq k \leq n\} \quad (7) \\ & \text{and } P(D = d_i|E_j) > t(t \geq 0.5)\}. \end{aligned}$$

Equation 7 ensures that the lower approximation of class $\{D = d_i\}$ contains such objects $E_j \in U$ that, given E_j, the probability of class $\{D = d_i\}$ increases faster than any other classes. Equation 7 also guarantees

$$R_*(D = d_i) \bigcap R_*(D = d_j) = \phi, (i \neq j).$$

Equation 6 is a special case of Equation 7.

4.3 PASH

Parameterized average support heuristic or PASH is defined the same as the average support heuristic in appearance. It is also a product of two factors: $Card(POS_{R+\{a\}}(D)) \times Q(R,a)$, where $Card(POS_{R+\{a\}}(D))$ is the cardinality of the positive region and $Q(R,a)$ is the overall quality of potential rules. The difference is that, in PASH, the positive region is the union of the new lower approximations and $Q(R,a)$ is also defined on the new lower approximations.

In summary, there are two cases to be considered when using PASH:

- When the a priori probabilities of decision classes are unknown, we assume they have equal a priori probability and use Equation 6.
- When the a priori probabilities of decision classes are known, we use Equation 7.

Average support heuristic and parameterized average support heuristic can be viewed as extensions to maximum support heuristic.

5 Experiments

We will give brief experiments and analysis of results in this section. We conducted a series of experiments with PASH using the mushroom data set obtained from the UC Irvine's machine learning repository [1]. Comparisons with results achieved with other methods running on the same data set were also performed. The mushroom data set has 8,124 instances with 22 condition attributes and 1 decision attribute. These algorithms are implemented in C language and executed on a PC with CPU 1.7GHz and 128MB RAM. There were three groups of experiments conducted.

5.1 Comparison of PASH with the Other Three Methods

We first tested PASH with the parameter value 1 under the stop condition $POS_R(D) = POS_C(D)$, that is, the program stops when one reduct is found. The execution time was around 15 minutes under this stop condition.

Table 1. Result of feature selection with stop condition $POS_R(D) = POS_C(D)$

Method	Selected features
Significance-oriented	5,20,8,12,3
Maximum support	5,10,17,6,8,16,18,13,12,11,22,4
Average support	5,10,17,6,8,16,18,13,12,11,4,7,19,20
PASH (parameter=1)	5,16,17,6,18,8,10,12,13,11,4,7,19,20

The comparison of the PASH result with results of significance-oriented method, maximum support method and average support method is presented

Table 2. Result of PASH with stop condition $POS_R(D) = POS_C(D)$

Parameter	Selected features
5	5,16,17,6,18,8,10,12,13,11,4,7,19,20
15	5,16,17,6,18,7,4,12,13,11,8,10,19,20
30	5,18,16,17,6,7,4,12,13,11,8,10,19,20
60	5,18,16,17,6,8,10,13,12,11,4,7,19,20
100	5,10,17,6,8,16,18,13,12,11,4,7,19,20

in Table 1. The left column indicates the method used and the right column lists the selected features in the order of selection. For example, the last row indicates that PASH selects the 5th feature as the most important feature, followed by the 16th feature, and then the 17th. The significance-oriented method obtained the smallest reduct which contains only five features. It may be concluded as the most time-efficient method. However, the features obtained from the significance-oriented method are not so important if they are evaluated by the criteria used in other methods. In other words, although a smaller and concise reduct is obtained, it may lose some important features. In fact, the significance-oriented method selected the 20th feature as the second most important feature whereas the maximum support method did not select it at all. The other two methods consider the 20th feature as the least important feature in the reducts. Another finding is that all three methods except the first one selected the 17th feature as the third important one but the significance-oriented method ignored it.

5.2 PASH with a Standard Stop Condition

The second set of experiments aimed to find out how the parameters value affect the feature selection results.

We tested PASH with random parameters under the same stop condition as the first set of experiments. The experimental results are shown in Table 2. The left column is the value of the parameter and the right column lists the selected features in the order of selection. It is suggested that the values of the parameter do not affect the size of reducts. However, the value of the parameter does influence the order of features in the reduct, i.e., the importance of the features. It is interesting that no matter what parameter value is used, the most important features (e.g. the 5th, the 17th) would be ordered in the first few steps and the least important ones would appear in the later parts of the reduct (e.g. the 19th, the 20th). In other words, PASH is not very sensitive to the parameter value and quite stable in feature selection.

5.3 Approximate Reducts with Different Parameter Levels

Finally, we tested PASH with different parameters under the stop condition $POS_R(D)/POS_C(D)$ >85%. This allows the program to stop when an approximate reduct is obtained. 85% is an accuracy threshold.

Table 3. Results of the PASH with stop condition $POS_R(D)/POS_C(D)$ >85%

Parameter	Selected features
5	5,16,17,6,18,8
15	5,16,17,6,18,7,4
30	5,18,16,17,6,7,4
60	5,18,16,17,6,8
100	5,10,17

In real world applications where the size of data set is large, we may not need to complete the computation of a reduct with PASH. If some of the most important features can be obtained in the first few steps, it may not need to compute the remaining less important features. The remaining part may cost a large part of the execution time. An approximate reduct which includes the most important features can be obtained with an accuracy threshold. In the test, we set the threshold as 85% and the program stops when the condition $POS_R(D)/POS_C(D)$ >85% is satisfied. Table 3 shows the result using PASH with different parameter values under this stop condition. It is shown that PASH stopped after selecting 3 to 7 features. Comparing with Table 2, PASH obtained an approximate reduct in much fewer steps. It is more efficient to use an approximate reduct with fewer features. It is suggested that when an appropriate parameter (e.g. parameter = 100) is given, PASH can produce satisfactory results efficiently. In fact, reducts with 3 features were obtain with parameter size over 100.

6 Concluding Remarks

We present a recently proposed rough set based feature selection method, parameterized average support heuristic, and report a set of experiments results based on PASH in this paper. PASH considers the overall quality of the potential rules and thus may produce a set of rules with balanced support distribution over all decision classes. PASH includes a parameter to adjust the level of approximation and keeps the predictive rules that are ignored by the existing methods. The experiment results suggest that the an approximate reduct can be obtained with adjustable criteria. Further experiments with different data sets and parameter values need to be conducted.

Acknowledgement

Financial support through a grant of NSERC, Canada is gratefully acknowledged. Gratitude is given to anonymous reviewers of RSFDGrC'05 and participants of NAFIPS'04 for their generous and constructive comments.

References

1. C.L. Blake and C.J. Merz, UCI Repository of machine learning databases. Available at http://www.ics.uci.edu/~mlearn/MLRepository.html, University of California, 1998.
2. R. Bellman, *Adaptive Control Processes: A Guided Tour*, Princeton University Press, 1961.
3. J. S. Deogun, V. V. Raghavan, and H. Sever, "Rough set based classification methods and extended decision tables", *Proc. of The Int. Workshop on Rough Sets and Soft Computing*, pp302-309, 1994.
4. J. S. Deogun, S. K. Choubey, V. V. Raghavan, and H. Sever, "On feature selection and effective classifiers", *Journal of American Society for Information Science*, 49(5), 423-434, 1998.
5. M. Dash, H. Liu, "Feature selection for classification," *Intelligence Data Analysis*, 1, 131-156, 1997.
6. J.G. Dy and C. E. Brodley, "Feature selection for unsupervised learning", *The Journal of Machine Learning Research archive*, 5, 845 - 889, 2004.
7. X. Hu, *Knowledge discovery in databases: an attribute-oriented rough set approach*, PhD thesis, University of Regina, Canada, 1995.
8. G. H. John, R. Kohavi and K. Pfleger, "Irrelevant features and the subset selection problem," *Proceedings of the 11th International Conference on Machine Learning*, pp121-129, 1994.
9. K. Kira, L. Rendell, "A practical approach to feature selection," *Proceedings of the 9th International Conference on Machine Learning,* pp249–256, 1992.
10. I. Kononenko, "Estimating attributes: analysis and extension of relief," *Proceedings of European Conference on Machine Learning,* pp171-182, 1994.
11. T. Y. Lin, "Attribute (Feature) completion- the theory of attributes from data mining prospect," *Proceedings of International Conference on Data Mining*, Maebashi, Japan, pp.282-289, 2002.
12. H. Liu and R. Setiono, "A probabilistic approach to feature selection - a filter solution," *Proceedings of the 13th International Conference on Machine Learning*, pp319-327, 1996.
13. P.M. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection", *IEEE Transactions on Computers*, C-26(9), 917-922, 1977.
14. G.V. Trunk, "A problem of dimensionality: a simple example," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1(3), 306-307, 1979.
15. Y.Y. Yao and S.K.M. Wong, "A decision theoretic framework for approximating concepts," *International Journal of Man-machine Studies*", 37(6), 793-809, 1992.
16. M. Zhang and J.T. Yao, "A rough set approach to feature selection", *Proceedings of the 23rd International Conference of NAFIPS*, Canada, pp434-439, 2004.
17. N. Zhong, J.Z. Dong and S. Ohsuga, "Using rough sets with heuristics for feature selection," *Journal of Intelligent Information Systems*, 16, 199-214, 2001.
18. W. Ziarko, "Variable precision rough set model," *Journal of Computer and System Sciences*, 46, 39-59, 1993.