Automated Empirical Selection of Rule Induction Methods based on Recursive Iteration of Resampling Methods

Shusaku Tsumoto, Shoji Hirano and Hidenao Abe

Department of Medical Informatics, Faculty of Medicine,
Shimane University
89-1 Enya-cho Izumo 693-8501 Japan
Email: {tsumoto,hirano, abe}@med.shimane-u.ac.jp

Abstract. One of the most important problems in rule induction methods is how to estimate which method is the best to use in an applied domain. While some methods are useful in some domains, they are not useful in other domains. Therefore it is very difficult to choose one of these methods. For this purpose, we introduce multiple testing based on recursive iteration of resampling methods for rule-induction (MULT-RECITE-R). We applied this MULT-RECITE-R method to monk datasets in UCI data repository. The results show that this method gives the best selection of estimation methods in almost the all cases.

1 Introduction

One of the most important problems in rule induction methods [1, 5, 6, 8] is how to estimate which method is the best to use in an applied domain. While some methods are useful in some domains, they are not useful in other domains. Therefore it is very difficult to choose one of these methods.

In order to solve this problem, we introduce multiple testing based on recursive iteration of resampling methods for rule induction methods (MULT-RECITE-R). MULT-RECITE-R consists of the following four procedures: First, it randomly splits training samples(S_0) into two parts, one for new training samples(S_1) and the other for new test samples(T_1) using a given resampling method(R). Second, S_1 are recursively split into training samples(S_2) and test samples(T_2) using the same resampling strategy(R). Then rule induction methods are applied to S_2, results are tested and given metrics(S_2 metrics) are calculated by T_2 for each rule induction methods. This second procedure, *as the inner loop*, is repeated for finite times estimated from precision set by users and the statistics of metrics are obtained. Third, in the same way, rules are induced from S_1 and metrics(S_1 metrics) are calculated by T_1 for each rule induction methods. Then S_1 metrics are compared with S_2 metrics. If the difference between both results are not statistically significant, then it is counted as a success. The second and the third procedure, as *the outer loop*, are iterated for certain times estimated from precision preset by users, which gives a total success rate which

shows how many times of total repetitions S_2 metrics predict S_1 metrics. Finally, fourth, the above results are interpreted in the following way. If a success rate is high, then this estimation method is expected to be well-performed, and the induction method which gives the best metric is selected as the most suitable induction method. If a success rate is low, then this estimation is expected not to be a good evaluation method. So a list of machine learning methods ordered by S_1 metrics is returned as an output.

We applied this MULT-RECITE-R method to monk datasets in UCI repository [7]. The results show that this method gives the best selection of methods in almost the all cases.

The paper is organized as follows: Section 2 and 3 present the strategy of MULT-RECITE-R and its algorithm. Section 4 gives experimental results. Finally, we conclude this paper in Section 5.

2 Strategy of MULT-RECITE-R

There are many reports on rule induction methods and their performance in the community of machine learning [11]. However, since each performance is different in each paper, it is very difficult to determine which method should be selected.

Each of these methods has interesting characteristics of induced rules. For example, CN2 induces a decision list subsection, while ID3 calculate a decision tree. Strangely, comparison of these features of induced rules are used as secondary, because of the difficulties in evaluation, although classification accuracy or error rate are as the primary comparison index. However, as to classification accuracy, it is pointed out that these performances may depend on applied domains [9, 10], although it is easy to apply statistical methods to testing significance. Actually, it is hard and controversial to determine what factor should be applied to evaluation of rule induction methods, which remains to be an open question in machine learning.

Since our objective is to develop a method which empirically selects rule induction methods, we use accuracy as a metric for statistical evaluation in this paper [1].

The next important thing is that one may want to evaluate these rule induction methods without domain knowledge in case when domain-specific knowledge may not be applicable.

Therefore, since one of the most characteristics of resampling methods is that they are domain-independent [3, 4, 10], one way for evaluation is to select one method from considerable resampling methods, that is to say, to select the best rule induction method by using subsets of training samples. For example, let us consider when we have training samples, say {1,2,3,4,5,6,7,8,9,10}. Then, first, they are split it into new training samples, say {1,3,5,7,9}, and new test samples, {2,4,6,8,10}. Using new training samples, rule induction methods are applied and the results are compared with the result by the new test samples.

[1] It is notable that our MULT-RECITE-R can be applied to any numeric metrics.

Then the method which gives the best metric, such as the best classification rate, will be selected. For example, let the accuracy of the induced decision tree be equal to 0.97, and the accuracy of the rule to be equal to 0.82. Then induction of decision tree is selected as the best method. It may depend on splitting, so these procedures should be repeated for certain times, say 100 times. several statistics of the given metrics are calculated over these 100 trials, such as average, variance, and t-statistics.

In this method, we implicitly assume that the "matryoshka" principle should be true. That is, the best method for total population can be selected from original training samples, and the best method for original training samples can be estimated from training samples generated by resampling plans. Therefore, in terms of Section 2 and 3, a domain of both R_1 and R_2 is the best select method ($R_1(F_0, F_1) \simeq R_2(F_1, F_2) = (\textit{the best method})$.)

3 An Algorithm for MULT-RECITE-R

An algorithm for MULT-RECITE-R can be described by embedding a rule induction method into the following algorithm based on a resampling scheme.

INPUTS: S_0: Training Samples
α: Precision for statistical test
α_{in}: Precision for the Inner Loop
α_{out}: Precision for the Outer Loop
L_r: a List and Subprocedures of Rule Induction Methods
L_m: a List of Metrics
R: Resampling Scheme

OUTPUTS: BI: the Best Induction method selected by success rate
M_1: a List of Induction Methods ordered by success rates
SR: Overall Success Rate
BI_p: the Best Induction method selected by adjusted-p Value
M_{1p}: a List of Induction Methods ordered by adjusted-p Values
SR_p: Overall (Adjusted-)p Value

1) Set Counter to 0 ($i := 0$, $succ := 0$, $p\_calc := 0$). And set B_{in} and B_{out} to $[10^{-\alpha_{in}}]$ and $[10^{-\alpha_{out}}]$, respectively [2].
2) Randomly split training samples(S_0) into two parts, one for new training samples(S_1) and the other for new test samples(T_1) using a given resampling plan(R).
3) Randomly split training samples(S_1) into two parts, one for new training samples(S_2) and the other for new test samples(T_2) using the same resampling plan(R). Then perform the following subprocedures.
3-a) Induce rules from S_2 for each member of L.

[2] $\lceil x \rceil$ denotes a maximum integer which do not exceed x. For example, $\lceil 4.6 \rceil$ is equal to 4.

3-b) Test induced results using T_2 and Calculate given metrics (S_2 metrics).

3-c) Repeat 3-b) and 3-c) for B_{in} times.

3-d) Calculate statistics of S_2 metrics.

4) Apply all the rule induction methods to S_1. Then execute the following procedures.

4-a) Test induced results by using T_1 and Calculate given metrics(S_1 metrics).

4-b) Compare S_1 metrics with S_2 metrics. If the best induction method j for S_1 metrics is the same as that of S_2 metrics, then Count this trial as a success on evaluation ($succ_j := succ_j + 1$). Otherwise, then Count it as a failure.

4-c) Test statistical significance between the best statistics of S_2 metrics and S_1 metrics using student t-test. If not significant, goto 5). Otherwise, Count this trial as a failure ($p\_calc_j := p\_calc_j + 1$).

5) Increment the counter ($i := i + 1$). If the counter is less than the upper bound($i < B_{out}$),
goto 2). If not, goto 6).

6) Calculate the overall success rate ($SR := \sum succ_j / B_{out}$). And calculate an ordered list of evaluation M_1 with the success rate $succ_j/B_{out}$ of each member in L.

7) Calculate the overall adjusted p-value ($p := \sum p\_calc_j / B_{out}$). And calculate an ordered list of evaluation M_1 with the success rate p_calc_j/B_{out} of each member in L.

8) Interpret the above results by the overall success rates. If a success rate is high, then this estimation method is expected to well-performed, and output the induction method j which gives the best metric is selected as the most suitable induction method ($BI := j$) and an ordered list M_1. If a success rate is low, then this estimation is expected to be not a good evaluation method. Thus, only a list of machine learning methods ordered by S_1 metrics is returned as an output ($BI := nil$).

9) Interpret the above results by the overall adjusted-p values. If $p < \alpha$, then this estimation method is expected to well-performed, and output the induction method j which gives the best metric is selected as the most suitable induction method ($BI_p := j$) and an ordered list M_1p. If $p \geq \alpha$, then this estimation is expected to be not a good evaluation method. Thus, only a list of machine learning methods ordered by S_1 metrics is returned as an output ($BI_p := nil$).

4 Experimental Results

We applied this MULT-RECITE-R method to monk datasets in UCI repository [7]. In these experiments, we set L_r, L_m α, α_{in} and α_{out} be equal to the same values as the above Monk's problems and set R to {2-fold cross-validation, the Bootstrap method}.

Unfortunately, in these databases, test samples are not given independently. So we first have to generate test samples from the original training samples. to evaluate our MULT-RECITE-R methods in the same way as evaluation shown in

Section 3 . First, given samples are randomly split into training samples(S_0) and test samples(T_0). This T_0 correspond to test samples of Monk's problems, and S_0 correspond to training samples of Monks problems. Then MULT-RECITE-R method is applied to new training samples. This splitting procedure is repeated for 100 times in order for the effect of random sampling to be small.

Table 1. Results of S_2 and S_1 Metrics(Accuracy)

		S_2 Metric		
Domain	Samples	C4.5	AQR	CN2
Monk-1	62	84.3±1.5	90.2±0.9	**92.0±1.8**
Monk-2	86	62.6±2.4	**74.8±1.9**	59.1±1.7
Monk-3	62	**87.7±1.4**	82.5±1.3	84.8±0.9
		S_1 Metric		
Domain	Samples	C4.5	AQR	CN2
Monk-1	124	85.3±0.9	91.2±0.5	**93.0±0.2**
Monk-2	169	66.7±1.3	**75.8±0.7**	60.1±0.8
Monk-3	122	**89.7±0.2**	83.5±0.4	83.8±0.5

Table 2. Success Rate (100 Trials)

	Overall Success	Success Rate		
Domain	Rate	C4.5	AQR	CN2
Monk-1	94	9	12	**73**
Monk-2	74	19	**31**	24
Monk-3	90	**79**	6	5

The above experimental results give us three interesting results, although all of the applied databases are of small size.

First, 2-fold repeated cross validation performs slightly better than the Bootstrap method, which corresponds to the characteristics derived by [2, 3]. Therefore, for predictive use, evaluation by cross-validation would be better, although the variance of estimation will be larger.

Second, the best selected method does not always perform better than other two methods. That is, in some generated samples, other methods will perform better. Finally, in the cases when MULT-RECITE-R does not go well, the differences of three rule induction methods in accuracy are not so significant. That is, we can select any of three methods, although the accuracy of each method is not so high.

Table 3. Adjusted-p Value (100 Trials)

	Overall p-	Adjusted-p Value		
Domain	Value	C4.5	AQR	CN2
Monk-1	0.02	0.01	0.01	**0.00**
Monk-2	0.10	0.04	**0.02**	0.04
Monk-3	0.05	**0.01**	0.02	0.02

5 Conclusion

One of the most important problems in rule induction methods is how to estimate which method is the best to used in an applied domain. For this purpose, we introduce multiple testing based on recursive iteration of resampling methods for rule-induction (MULT-RECITE-R). We apply this MULT-RECITE-R method to three original medical databases and seven UCI databases. The results show that this method gives the best selection of estimation methods in almost the all cases.

References

1. Clark, P., Niblett, T. The CN2 Induction Algorithm. *Machine Learning*, **3**,261-283, 1989.
2. Efron, B. How biased is the apparent error rate of a prediction rule ? *J. Amer. Statist. Assoc.* **82**, 171-200, 1986.
3. Efron, B. and Tibshirani, R. *An Introduction to the Bootstrap*. Chapman and Hall, London, 1994.
4. McLachlan, G.J., *Discriminant Analysis and Statistical Pattern Recognition,* John Wiley and Sons, New York, 1992.
5. Michalski, R.S., A Theory and Methodology of Machine Learning. Michalski, R.S., Carbonell, J.G. and Mitchell, T.M., *Machine Learning - An Artificial Intelligence Approach*, Morgan Kaufmann, Palo Alto, CA, 1983.
6. Michalski, R.S., et al. The Multi-Purpose Incremental Learning System AQ15 and its Testing Application to Three Medical Domains, in: *Proceedings of AAAI-86*, 1041-1045, AAAI Press, Palo Alto, CA, 1986.
7. Murphy, P.M. and Aha, D.W. *UCI Repository of machine learning databases* [Machine-readable data repository]. Irvine, CA: University of California, Department of Information and Computer Science.
8. Quinlan, J.R. *C4.5 - Programs for Machine Learning*, Morgan Kaufmann, CA, 1993.
9. Schaffer, C. Overfitting Avoidance as Bias. *Machine Learning*, **10**, 153-178, 1993.
10. Schaffer, C. Selecting a Classification Method by Cross-Validation. *Machine Learning*, **13**, 135-143, 1993.
11. Thrun, S.B. et al. The Monk's Problems- A performance Comparison of Different Learning algorithms. Technical Report CS-CMU-91-197, Carnegie Mellon University, 1991.