Rule Extraction with Guaranteed Fidelity

Ulf Johansson, Rikard König, Henrik Linusson, Tuve Löfström, Henrik Boström

Rule Extraction with Guaranteed Fidelity

Ulf Johansson[1⋆], Rikard König[1], Henrik Linusson[1], Tuve Löfström[1], and Henrik Boström[2]

[1] School of Business and IT
University of Borås, Sweden
{ulf.johansson, rikard.konig, henrik.linusson, tuve.lofstrom}@hb.se
[2] Department of Systems and Computer Sciences
Stockholm University, Sweden
henrik.bostrom@dsv.su.se

Abstract. This paper extends the conformal prediction framework to rule extraction, making it possible to extract interpretable models from opaque models in a setting where either the infidelity or the error rate is bounded by a predefined significance level. Experimental results on 27 publicly available data sets show that all three setups evaluated produced valid and rather efficient conformal predictors. The implication is that augmenting rule extraction with conformal prediction allows extraction of models where test set errors or test sets infidelities are guaranteed to be lower than a chosen acceptable level. Clearly this is beneficial for both typical rule extraction scenarios, i.e., either when the purpose is to explain an existing opaque model, or when it is to build a predictive model that must be interpretable.

Keywords: Rule extraction, Conformal prediction, Decision trees

1 Introduction

When predictive models must be interpretable, most data miners will use decision trees like C4.5/C5.0 [1]. Unfortunately, decision trees are much weaker in terms of predictive performance than opaque models like support vector machines, neural networks and ensembles. Opaque predictive models, on the other hand, make it impossible to assess the model, or even to understand the reasoning behind individual predictions. This dilemma is often referred to as the *accuracy vs. comprehensibility trade-off*.

One way of reducing this trade-off is to apply *rule extraction*, which is the process of generating a transparent model based on a corresponding opaque predictive model. Naturally, extracted models must be as good approximations as possible of the opaque models. This criterion, called *fidelity*, is therefore a key part of the optimization function in most rule extracting algorithms. For classification, the *infidelity rate* is the proportion of test instances where the

⋆ This work was supported by the Swedish Foundation for Strategic Research through the project High-Performance Data Mining for Drug Effect Detection (IIS11-0053) and the Knowledge Foundation through the project Big Data Analytics by Online Ensemble Learning (20120192).

extracted model outputs a different label than the opaque model. Similarly, the fidelity is the proportion of test instances where the two models agree. Unfortunately, when *black-box* rule extraction is used, i.e., when the rule extractor utilizes input-output patterns consisting of the original input vector and the corresponding prediction from the opaque model to learn the relationship represented by the opaque model, the result is often a too specific or too general model resulting in low fidelity on the test set, that is, the extracted model is actually a poor approximation of the opaque. Consequently, decision makers would like to have some guarantee *before* applying the extracted model to the test instances that the predictions will actually mimic the opaque.

In conformal prediction [2], prediction sets with a bounded error are produced, i.e., for classification, the probability of excluding the correct class label is guaranteed to be less than the predetermined significance level. The prediction sets can contain one, multiple or even zero class labels, so the price paid for the guaranteed error rate is that not all predictions are informative. In, inductive conformal prediction (ICP) [2], just one model is induced from the training data, and then used for predicting all test instances, but a separate data set (called the *calibration set*) must be used for calculating *conformity scores*.

The conformal prediction framework has been applied to several popular learning schemes, such as ANNs [3], kNN [4] and SVMs [5]. Until now, however, the guarantee provided by conformal prediction has always been related to the error rate. In this paper, we extend the conformal prediction framework to rule extraction, specifically introducing the possibility to bound the infidelity rate by a preset significance level.

2 Background

Rule extraction has been heavily investigated for ANNs, and the techniques have been applied mainly to ANN models; for an introduction and a good survey of traditional methods, see [6]. For ANN rule extraction, there are two fundamentally different extraction strategies, *decompositional* (*open-box* or *white-box*) and *pedagogical* (*black-box*). Decompositional approaches focus on extracting rules at the level of individual units within a trained ANN. Typically, the output of each hidden and output unit is first modeled as a consequent of their inputs, before the rules extracted at the individual unit level are aggregated to form the composite rule set for the ANN. Two classic open-box algorithms are RX [7] and Subset [8].

The core pedagogical idea is to view rule extraction as a learning task, where the target concept is the function originally learned by the opaque model. Black-box rule extraction is therefore an instance of predictive modeling, where each input-output pattern consists of the original input vector $\boldsymbol{x}_i$ and the corresponding prediction $f(\boldsymbol{x}_i; \theta)$ from the opaque model. One typical and well-known black-box algorithm is TREPAN [9].

It must be noted that black-box rule extraction algorithms can be applied to any opaque model, including ensembles, and it can use any learning algorithm

producing interpretable models as the actual rule extractor. An inherent problem for open-box methods, regarding both running time and comprehensibility, is the scalability. The potential size of a rule for a unit with n inputs each having k possible values is k^n, meaning that a straightforward search for possible rules is normally impossible for larger networks. Consequently, most modern rule extraction algorithms are black-box, see the more recent survey [10].

There is, however, one very important problem associated with black-box rule extraction. Even if the algorithm aims for maximizing fidelity in the learning phase, there is no guarantee that the extracted model will actually be faithful to the opaque model when applied to test set instances. Instead, since black-box rule extraction is just a special case of predictive modeling, the extracted models may very well overfit or underfit the training data, leading to poor fidelity on test data. The potentially low test set fidelity for black-box techniques stands in sharp contrast to open-box methods where the rules, at least in theory, should have perfect fidelity, even on the test set. Consequently, in situations where a very high fidelity is needed, open-box methods may be necessary; see e.g., [11]. Ideally though, we would like to have the best of both worlds, i.e., providing the efficiency and the freedom to use any type of opaque model present in black-box rule extractors, while guaranteeing test set fidelity. Again, the purpose of this paper is to show how the conformal prediction framework can be employed for achieving this.

An interesting discussion about the purpose of rule extraction is found in [12], where Zhou argues that rule extraction really should be seen as two very different tasks; rule extraction *for* neural networks and rule extraction *using* neural networks[1]. While the first task is solely aimed at understanding the inner workings of an opaque model, the second task is explicitly aimed at extracting a comprehensible model with higher accuracy than a comprehensible model created directly from the data set. More specifically, in rule extraction *for* opaque models, the purpose is most often to explain the reasoning behind individual predictions from an opaque model, i.e., the actual predictions are still made by the opaque model. In that situation, test set fidelity must be regarded as the most important criterion, since we use the extracted model to understand the opaque. In rule extraction *using* opaque models, the predictions are made by the extracted model, so it is used both as the predictive model and as a tool for understanding and analysis of the underlying relationship. In that situation, predictive performance is what matters, so the data miner must have reasons to believe that the extracted model will be more accurate than other comprehensible models induced directly from the data. The motivation for that rule extraction *using* opaque models may work is that even a highly accurate opaque model is a smoothed representation of the underlying relationship. In fact, training instances misclassified by the opaque model are often atypical, i.e., learning such instances will reduce the generalization capability. Consequently, rule ex-

[1] Naturally this distinction is as relevant for rule extraction from any opaque model, not just from ANNs, so we use the terms rule extraction for or using *opaque models* instead.

traction is most often less prone to overfitting than standard induction, resulting in smaller and more general models.

2.1 Conformal prediction

A key component in ICP is the conformity function, which produces a score for each instance-label pair. When classifying a test instance, scores are calculated for all possible class labels, and these scores are compared to scores obtained from a calibration set consisting of instances with known labels. Each class is assigned a probability that it does conform to the calibration set based on the fraction of calibration instances with a higher conformity score. For each test instance, the conformal predictor outputs a set of predictions with all class labels having a probability higher than some predetermined *significance level*. This prediction set may contain one, several, or even no class labels. Under very general assumptions, it can be guaranteed that the probability of excluding the true class label is bounded by the chosen significance level, independently of the conformity function used, for more details see [2].

In ICP, the conformity function A is normally defined relative to a trained model M:

$$A(\langle \bar{x}, c \rangle) = F(c, M(\bar{x})) \tag{1}$$

where $\bar{x}$ is a vector of feature values (representing the example to be classified), c is a class label, $M(\bar{x})$ returns the class probability distribution predicted by the model, and the function F returns a score calculated from the chosen class label and predicted class distribution.

Using a conformity function, a *p-value* for an example $\bar{x}$ and a class label c is calculated in the following way:

$$p_{\langle \bar{x}, c \rangle} = \frac{|\{s : s \in S \wedge A(s) \leq A(\langle \bar{x}, c \rangle)\}|}{|S|} \tag{2}$$

where S is the calibration set. The prediction for an example $\bar{x}$, where $\{c_1, \ldots, c_n\}$ are the possible class labels, is:

$$P(\bar{x}, \sigma) = \{c : c \in \{c_1, \ldots, c_n\} \wedge p_{\langle \bar{x}, c \rangle} > \sigma\} \tag{3}$$

where σ is a chosen significance level, e.g., 0.05.

3 Method

The purpose of this study is to extend the conformal prediction framework to rule extraction, and show how it can be used for both rule extraction *for* opaque models and rule extraction *using* opaque models. Since standard ICP is used, the difference between the scenarios is just how the calibration set is used. For the final modeling, all setups use J48 trees from the Weka workbench [13]. Here J48, which is the Weka implementation of C4.5, uses default settings, but pruning was turned off and Laplace smoothing was used for calculating the probability estimates. The three different setups evaluated are described below:

- **J48**: J48 trees built directly from the data. When used as a conformal predictor, the calibration set uses the true targets, i.e., the guarantee is that the error rate is bounded by the significance level.
- **RE-a**: Rule extraction *using* opaque models. Here, an opaque model is first trained, and then a J48 tree is built using original training data inputs, but with the predictions from the opaque model as targets. For the conformal prediction, the calibration set uses the true targets, so the guarantee is again that the error rate is bounded by the significance level.
- **RE-f**: Rule extraction *for* opaque models. The J48 model is trained identically to RE-a, but now the conformal predictor uses predictions from the opaque model as targets for the calibration. Consequently, the guarantee is that the infidelity rate will be lower than the significance level.

In the experimentation, bagged ensembles of 15 RBF networks were used as opaque models. With guaranteed validity, the most important criterion for comparing conformal predictors is *efficiency*. Since high efficiency roughly corresponds to a large number of singleton predictions, *OneC*, i.e., the proportion of predictions that include just one single class, is a natural choice. Similarly, *MultiC* and *ZeroC* are the proportions of predictions consisting of more than one class, and empty predictions, respectively. One way of aggregating these number is *AvgC*, which is the average number of classes in the predictions.

In this study, the well-known concept of *margin* was used as the conformity function. For an instance i with the true class Y, the higher the probability estimate for class Y the more conforming the instance, and the higher the other estimates the less conforming the instance. For the evaluation, 4-fold cross-validation is used. The training data was split 2:1; i.e., 50% of the available instances were used for training and 25% were used for calibration. The 27 data sets used are all publicly available from either the UCI repository [14] or the PROMISE Software Engineering Repository [15].

4 Results

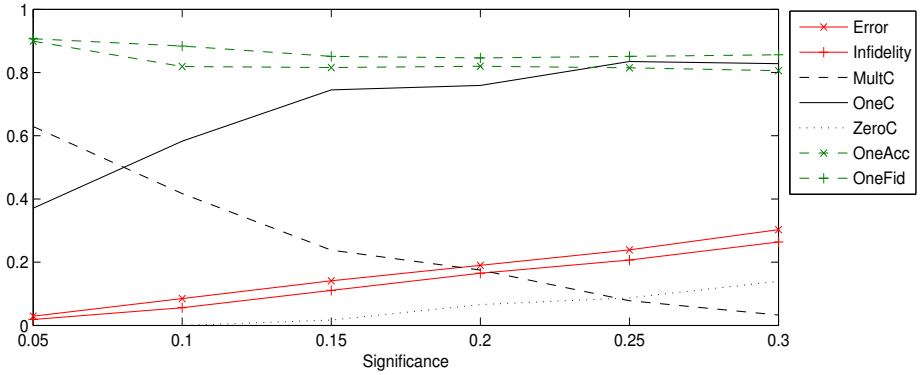
Table 1 below shows the accuracy, AUC and size (total number of nodes) for the J48 models produced using either standard induction or rule extraction. As described in the introduction, the rule extraction is supposed to increase model accuracy, produce smaller models, or both. Comparing mean values and wins/ties/losses, the results show that the use of rule extraction actually produced models with higher accuracy. A standard sign test requires 19 wins for significance when $\alpha = 0.05$, so the difference is statistically significant at that level. Looking at models sizes, the extracted models are also significantly less complex. When comparing the ranking ability, however, the larger induced tree models obtained higher AUCs, on a majority of the data sets.

Table 1. Accuracy, AUC and Size

	Accuracy		AUC		Size			Accuracy		AUC		Size	
Data set	Ind.	Ext.	Ind.	Ext.	Ind.	Ext.	Data set	Ind.	Ext.	Ind.	Ext.	Ind.	Ext.
ar1	.909	.913	.457	.608	8.0	6.3	kc1	.839	.848	.680	.595	100.0	10.8
ar4	.817	.808	.664	.660	8.5	5.8	kc2	.827	.828	.785	.674	26.5	8.0
breast-w	.921	.928	.953	.945	20.8	15.0	kc3	.889	.903	.688	.629	25.5	4.8
colic	.705	.717	.713	.731	34.8	21.8	letter	.824	.800	.838	.824	20.0	23.3
credit-a	.712	.751	.771	.800	57.5	32.5	liver	.578	.620	.561	.610	22.3	23.5
credit-g	.683	.712	.620	.643	108.0	51.3	mw1	.901	.917	.679	.616	15.5	4.5
cylinder	.644	.634	.630	.638	63.3	50.3	sonar	.618	.680	.657	.733	18.8	14.0
diabetes	.691	.711	.690	.684	33.3	29.5	spect	.771	.793	.699	.731	25.0	11.8
heart-c	.719	.723	.754	.773	31.0	21.8	spectf	.744	.756	.718	.691	20.3	13.0
heart-h	.760	.786	.769	.804	28.5	14.5	tic-tac-toe	.770	.694	.775	.631	52.8	45.5
heart-s	.767	.748	.810	.784	31.3	17.3	vote	.899	.905	.933	.928	19.8	13.5
hepatitis	.781	.781	.746	.701	18.5	14.0	vowel	.786	.725	.804	.782	13.5	15.3
iono	.769	.789	.732	.750	13.3	13.8	**Mean**	**.761**	**.767**	**.719**	**.712**	**31.9**	**19.0**
jEdit4042	.642	.639	.669	.671	21.3	14.8	**Wins**	**8**	**19**	**15**	**12**	**4**	**23**
jEdit4243	.583	.589	.606	.599	23.5	15.8							

Turning to the results for conformal prediction, Fig 1 shows the behavior of extracted J48 trees as conformal predictors on the Iono data set, when using true targets for calibration. Since the conformal predictor is calibrated using true targets, it is the error and not the infidelity that is bounded by the significance level.

Fig. 1. Rule extraction *using* opaque model. Iono.

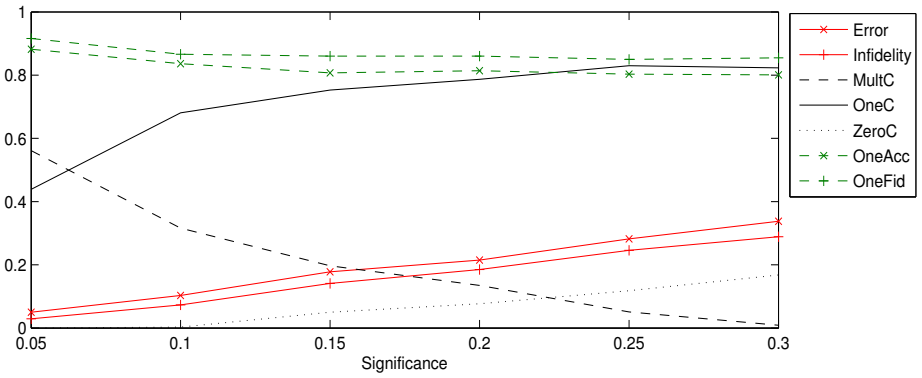


First of all, the conformal predictor is valid and well-calibrated, since the error rate is very close to the corresponding significance level. Analyzing the efficiency, the number of singleton predictions (OneC) starts at approximately

40% for $\epsilon = 0.05$, and then rises quickly to over 70% at $\epsilon = 0.15$. The number of multiple predictions (MultiC), i.e., predictions containing both classes, has the exact opposite behavior. The first empty predictions (ZeroC) appear at $\epsilon = 0.10$. Interestingly enough, OneAcc (the accuracy of the singleton predictions) is always higher than the accuracy of the underlying tree model (0.769), so singleton predictions from the conformal predictor could be trusted more than predictions from the original model. Finally, the fidelity of the singleton predictions (OneFid) is very high, always over 80%. In fact, the infidelity rate is always lower than the error, indicating that the extracted conformal predictor is very faithful to the opaque model, even if this is not enforced by the conformal prediction framework in this setup.

Fig 2 below shows the behavior of extracted J48 trees as conformal predictors on the Iono data set, when using the ensemble predictions as targets for calibration.

Fig. 2. Rule extraction *for* opaque model. Iono.



In this setup, it is the infidelity and not the error that is guaranteed, and indeed the actual infidelity rate is very close to the significance level. Here, singleton predictions are more common than for the other setup, i.e., it is easier to have high confidence in predictions about ensemble predictions than true targets. The error rate is slightly higher than the significance level, but interestingly enough both OneAcc and OneFid are comparable to the results for the previous setup.

Table 2 below shows detailed results for the three different conformal prediction setups, when the level of significance is $\epsilon = 0.10$.

Table 2. Conformal prediction with $\epsilon = 0.1$. Bold numbers indicate criteria that are guaranteed by the conformal prediction framework.

	Error			Infidelity		OneC			OneAcc			OneFid	
	J48	RE-a	RE-f	RE-a	RE-f	J48	RE-a	RE-f	J48	RE-a	RE-f	RE-a	RE-f
ar1	**.070**	**.070**	.124	.054	**.091**	.904	.821	.842	.919	.939	.930	.957	.950
ar4	**.043**	**.052**	.083	.042	**.056**	.369	.366	.649	.757	.870	.907	.881	.917
breast-w	**.090**	**.094**	.105	.095	**.102**	.936	.886	.852	.940	.944	.948	.946	.952
colic	**.098**	**.094**	.143	.063	**.086**	.538	.503	.652	.800	.797	.770	.870	.861
credit-a	**.094**	**.111**	.182	.055	**.089**	.556	.644	.818	.803	.818	.770	.908	.885
credit-g	**.104**	**.085**	.195	.024	**.090**	.440	.409	.745	.753	.780	.733	.935	.880
cylinder	**.097**	**.099**	.122	.073	**.093**	.317	.344	.427	.664	.695	.698	.764	.785
diabetes	**.083**	**.096**	.191	.044	**.098**	.415	.447	.723	.795	.774	.734	.888	.865
heart-c	**.073**	**.091**	.127	.045	**.084**	.407	.452	.586	.793	.787	.770	.879	.857
heart-h	**.083**	**.080**	.163	.034	**.075**	.509	.565	.858	.786	.845	.818	.931	.914
heart-s	**.070**	**.072**	.119	.037	**.059**	.461	.458	.625	.852	.834	.804	.915	.897
hepatitis	**.045**	**.055**	.081	.032	**.048**	.528	.445	.578	.925	.853	.851	.936	.914
iono	**.078**	**.089**	.108	.090	**.096**	.570	.608	.625	.845	.850	.805	.806	.813
jEdit4042	**.091**	**.089**	.197	.022	**.062**	.369	.291	.619	.717	.659	.662	.908	.880
jEdit4243	**.083**	**.080**	.265	.015	**.068**	.245	.221	.665	.647	.620	.627	.940	.888
kc1	**.093**	**.097**	.217	.002	**.095**	.784	.729	.900	.881	.866	.870	.997	.997
kc2	**.088**	**.100**	.216	.011	**.101**	.758	.691	.934	.883	.853	.840	.985	.969
kc3	**.075**	**.081**	.157	.011	**.088**	.878	.931	.916	.920	.913	.920	.989	.995
letter	**.098**	**.089**	.098	.082	**.090**	.657	.650	.657	.860	.853	.851	.871	.865
liver	**.094**	**.072**	.170	.042	**.104**	.253	.263	.516	.626	.724	.665	.851	.783
mw1	**.091**	**.091**	.129	.047	**.089**	.963	.990	.932	.911	.919	.935	.967	.983
sonar	**.070**	**.108**	.072	.091	**.089**	.233	.423	.303	.702	.737	.729	.788	.701
spect	**.070**	**.088**	.158	.026	**.056**	.549	.665	.844	.875	.866	.819	.960	.934
spectf	**.078**	**.085**	.114	.066	**.085**	.491	.444	.559	.826	.808	.805	.843	.807
tic-tac-toe	**.105**	**.087**	.235	.023	**.116**	.635	.370	.794	.792	.758	.707	.926	.857
vote	**.096**	**.079**	.091	.062	**.077**	.875	.845	.869	.923	.939	.934	.954	.946
vowel	**.083**	**.064**	.078	.100	**.097**	.581	.458	.378	.836	.854	.805	.777	.724
Mean	**.083**	**.085**	**.146**	**.048**	**.085**	**.564**	**.553**	**.699**	**.816**	**.821**	**.804**	**.903**	**.882**
Mean Rank	-	-	-	-	-	**2.19**	**2.41**	**1.41**	**1.85**	**1.81**	**2.33**	**1.26**	**1.74**

Investigating the errors and infidelities, it is obvious that the conformal prediction framework applies to both rule extraction scenarios, i.e., when the error rate or the infidelity rate must be lower than the significance level. On almost all data sets, the errors for J48 and RE-a are quite close to the significance level $\epsilon = 0.1$, indicating that the conformal predictors are valid and well-calibrated. Similarly, the infidelities for RE-f are also close to 0.1, on most data sets. Looking at the efficiency, measured using the OneC metric, RE-f is clearly the most efficient conformal predictor. An interesting observation is that the errors for RE-f often are much higher than the corresponding significance level, thus indicating that the extracted model quite often is certain about the prediction from the ensemble, even when the ensemble prediction turns out to be wrong. This phenomenon is also obvious from the lower OneAcc exhibited by RE-f.

Regarding infidelities and OneFid, it may be noted that RE-a turns out to be overly conservative. This actually results in a higher OneFid, compared to RE-f, but the explanation is the much fewer singleton predictions. Simply put, with a high demand on confidence in the selected singleton predictions, these tend to be predicted identically by the ensemble.

Table 3 below shows a summary, presenting averaged values and mean ranks over all data sets for three different significance levels. Included here is the metric AvgC, which is the average number of labels in the prediction sets. Since there are very few empty predictions at $\epsilon = 0.05$, OneC and AvgC will, for this significance level, produce the same ordering of the setups.

Table 3. Conformal prediction summary. Bold numbers indicate criteria that are guaranteed by the conformal prediction framework.

	$\epsilon = 0.05$			$\epsilon = 0.1$			$\epsilon = 0.2$		
	Ind	RE-a	RE-f	Ind	RE-a	RE-f	Ind	RE-a	RE-f
Error	**.034**	**.034**	.084	**.083**	**.085**	.146	**.184**	**.183**	.251
Infidelity	-	.018	**.035**	-	.046	**.084**	-	.124	**.190**
AvgC	1.66	1.70	1.46	1.43	1.44	1.26	1.15	1.15	1.01
Rank	2.11	2.70	1.19	2.30	2.48	1.22	2.48	2.33	1.19
OneC	.339	.297	.525	.564	.552	.701	.772	.778	.821
Rank	2.11	2.70	1.19	2.19	2.41	1.41	2.15	2.04	1.81
OneAcc	.772	.752	.778	.815	.819	.805	.794	.796	.794
Rank	1.78	1.89	2.33	1.85	1.89	2.26	2.07	1.93	2.00
OneFid	-	.824	.857	-	.906	.884	-	.878	.869
Rank	-	1.44	1.56	-	1.22	1.78	-	1.48	1.52

Even when analyzing all three significance levels, all conformal predictors seem to be valid and reasonably well-calibrated. Looking for instance at RE-a, the averaged errors over all data sets are 0.034 for $\epsilon = 0.05$, 0.084 for $\epsilon = 0.1$ and 0.183 for $\epsilon = 0.2$. Similarly, the averaged infidelities for RE-f are 0.035 for $\epsilon = 0.05$, 0.084 for $\epsilon = 0.1$ and 0.190 for $\epsilon = 0.2$.

Comparing efficiencies, RE-f is significantly more efficient, with regard to both OneC and AvgC, than the other two setups. J48 and RE-a have comparable efficiencies. Regarding OneAcc, J48 and RE-a are most often more accurate than RE-f. It must, however, be noted that RE-f has a fundamentally different purpose than RE-a and J48, so RE-a should only be compared directly to J48; they are both instances of, in Zhou's terminology, rule extraction *using* opaque models, while RE-f, is rule extraction *for* opaque models. Consequently the most important observation is that all setups have worked as intended, producing valid, well-calibrated and rather efficient conformal predictors for the two different rule extraction scenarios.

5 Concluding remarks

In this paper, which should be regarded as a proof-of-concept, conformal prediction has been extended to rule extraction *for* opaque models and rule extraction *using* opaque models. The results show that conformal prediction enables extraction of efficient and comprehensible models, where either the error rate or the infidelity rate is guaranteed. This represents an important addition to the rule extraction tool-box, specifically addressing the problem with a potentially poor test set fidelity present in most black-box rule extractors.

For some reason rule extraction has not been extensively used on regression models, so the next step is to apply conformal prediction to this. We believe that the prediction intervals produced by conformal prediction regression will be a natural part of making extracted regression models accurate and comprehensible.

References

1. Quinlan, J.R.: C4.5: programs for machine learning. Morgan Kaufmann (1993)
2. Vovk, V., Gammerman, A., Shafer, G.: Algorithmic Learning in a Random World. Springer-Verlag New York, Inc. (2005)
3. Papadopoulos, H.: Inductive conformal prediction: Theory and application to neural networks. Tools in Artificial Intelligence **18** (2008) 315–330
4. Nguyen, K., Luo, Z.: Conformal prediction for indoor localisation with fingerprinting method. Artificial Intelligence Applications and Innovations (2012) 214–223
5. Makili, L., Vega, J., Dormido-Canto, S., Pastor, I., Murari, A.: Computationally efficient svm multi-class image recognition with confidence measures. Fusion Engineering and Design **86**(6) (2011) 1213–1216
6. Andrews, R., Diederich, J., Tickle, A.B.: Survey and critique of techniques for extracting rules from trained artificial neural networks. Knowl.-Based Syst. **8**(6) (1995) 373–389
7. Rudy, H.L., Lu, H., Setiono, R., Liu, H.: Neurorule: A connectionist approach to data mining. (1995) 478–489
8. Fu, L.: Rule learning by searching on adapted nets. In: AAAI. (1991) 590–595
9. Craven, M.W., Shavlik, J.W.: Extracting tree-structured representations of trained networks. In: Advances in Neural Information Processing Systems, MIT Press (1996) 24–30
10. Huysmans, J., Baesens, B., Vanthienen, J.: Using rule extraction to improve the comprehensibility of predictive models. FETEW Research Report KBI 0612, K. U. Leuven (2006)
11. Martens, D., Huysmans, J., Setiono, R., Vanthienen, J., Baesens, B.: Rule extraction from support vector machines: An overview of issues and application in credit scoring. In: Rule Extraction from Support Vector Machines. (2008) 33–63
12. Zhou, Z.H.: Rule extraction: using neural networks or for neural networks? J. Comput. Sci. Technol. **19**(2) (2004) 249–253
13. Witten, I.H., Frank, E.: Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann (2005)
14. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007)
15. Sayyad Shirabad, J., Menzies, T.: PROMISE Repository of Software Engineering Databases. School of Information Technology and Engineering, University of Ottawa, Canada (2005)