Evaluation of novel RandomRules methodology on synthetic data

M. Piškorec, D.Gamberger

The first stable version of RandomRules implementation (internal reference code rrD1) has been tested on a few synthetic datasets.

For this purpose a program for the construction of appropriate synthetic classification datasets has been developed. The main characteristic of the program is the possibility to construct sets of different size in respect of the number of attributes and the number of examples. Also, the program enables to select the number of classes and the percentage of examples in each class. By default all attributes values are random values in the range 0-100 (with two digits in the fractional part) and class values are set so that required statistics of the distribution among classes is satisfied. Finally, the user can select among a few different functions that connect attribute values and the class value. If no function is selected then the result is a random dataset with predefined number of examples, number of attributes, and number of classes but with no logical connection between attribute values and classes.

Evaluation has been performed on two types of functions (F1 and F2) and one random type (R1).

F1 - if example is in class 1 then A2 value is set exactly to value A1+10 (value of attribute 1 incremented by 10), if example is in class 2 then A4=A3+10.

F2 - if example is in class 1 then A1 is a random value in the range V to (V +50), where V= |A2-50| + |A3-50| - |A4-50| - |A5-50|. If example is in class 2 then A1 value is in the range (V+50) to (V+100).

It must be noted that F1 is a relative simple classification task that is partially non-deterministic because there can be examples that satisfy both conditions A2=A1+10 and A4=A3+10 and that are randomly classified either in class 1 or class 2. In contrast to that, F2 is a completely deterministic classification task that is difficult because it includes absolute value of even four different attributes. It is relative easy to recognize that larger A1 values are more characteristic for examples in class 2 but identification of the complete function is a real challenge.

For each of these three data types, we have constructed in total 16 groups of datasets each consisting of 11 datasets of the same size and type but constructed with different random seeds. Ten of these datasets are used to build predictive models while the eleventh is used for the evaluation of the achieved accuracy. Reported accuracy and execution times are mean values for ten experiments in each group.

The groups are different in respect of the number of examples and the number of attributes. The intention has been to test how the size of the problem influences the obtained predictive accuracy and the execution time. We have used following four different number of examples (350, 1000, 3500, 10000) and they have been combined with four different number of attributes (35, 100, 350, 1000). It means that in the first group we have datasets with 350 examples and 35 attributes while in the last one are datasets with 10000 of examples and 1000 attributes.

Evaluation has been done for RandomRules algorithm in its default setting with automatic stopping criteria when saturation of the estimated predictive accuracy has been achieved (referenced as “default”) and for the same algorithm when the option for the fixed number of constructed rules is used (referenced as “fixed”). In all experiments the number of generated rules is fixed at 50,000. No other options have been used or changed. The results are compared with those obtained by the Random Forest algorithm (PARF implementation) used in its default setting and with 1000 constructed decision trees.

In order to ensure reproducibility of the result and experiments with other versions of the RandomRules algorithm in the future, all datasets are made public at http://lis.irb.hr/DataSets/Synthetic_data/. Each dataset is available in the arrf form used by Weka algorithms and PARF implementation and the plain text form used by RandomRules. The names of files define type (F1, F2, R1), N (number of examples), A (number of attributes), and C (number of classes). Final s0-s9 part is the number in the group for datasets used for model induction (learning sets) while sT denotes the dataset on which predictive accuracy is measured (test set).

Experiments have been done on Intel 3.2 GMz processor.

Results

Tables 1-3 present achieved predictive accuracy by three different induction approaches (RandomRules default, RandomRules, fixed, and Random Forest) for functions F1, F2, and R1, respectively. Presented are percent's of the accuracy. Sign X denotes that execution time has been unacceptably long.

Tables 4-6 present execution time for the three algorithms for functions F1, F2, and R1. The presented time is in seconds.

Table 1 Predictive accuracy for F1 type of relation

	Number of examples	Number of attributes			
		35	**100**	**350**	**1000**
Random Rules default	**350**	92.48	89.03	91.97	81.66
	1000	94.80	94.81	95.19	95.58
	3500	97.82	97.85	97.42	97.73
	10000	98.85	98.58	98.62	98.64
Random Rules fixed	**350**	90.20	81.83	66.08	54.00
	1000	94.36	91.58	78.92	72.17
	3500	97.22	95.63	83.53	75.20
	10000	98.30	97.43	88.59	79.63
Random Forest PARF	**350**	68.80	62.32	56.94	51.66
	1000	74.04	65.90	62.49	58.55
	3500	79.86	69.27	63.74	60.28
	10000	87.46	72.84	66.02	X

Table 2 Predictive accuracy for F2 type of relation

	Number of examples	Number of attributes			
		35	**100**	**350**	**1000**
Random Rules default	**350**	80.11	77.80	78.49	74.2
	1000	80.07	77.59	76.52	78.29
	3500	83.61	79.67	77.80	77.15
	10000	86.57	82.49	77.99	77.58
Random Rules fixed	**350**	80.86	78.03	77.54	73.17
	1000	81.18	81.42	77.20	78.25
	3500	85.31	86.35	84.62	80.8
	10000	86.92	89.42	88.67	88.05
Random Forest PARF	**350**	78.83	76.51	77.06	72.46
	1000	78.24	77.15	76.23	78.01
	3500	79.87	77.86	77.73	77.19
	10000	81.8	79.45	77.81	X

Table 3 Predictive accuracy for the random type of the relation (R1)

	Number of examples	Number of attributes			
		35	**100**	**350**	**1000**
Random Rules default	**350**	70.00	70.00	X	X
	1000	70.00	70.00	X	X
	3500	70.00	70.00	X	X
	10000	70.00	70.00	X	X
Random Rules fixed	**350**	69.29	67.20	61.63	69.43
	1000	69.86	69.06	69.24	69.79
	3500	69.99	69.92	69.76	69.86
	10000	69.69	69.83	69.64	69.50
Random Forest PARF	**350**	70.00	70.00	70.00	70.00
	1000	70.00	70.00	70.00	70.00
	3500	69.99	70.00	70.00	X
	10000	70.00	70.00	70.00	X

Table 4 Execution time for F1 type of relation

	Number of examples	Number of attributes			
		35	**100**	**350**	**1000**
Random Rules default	**350**	4	4	5	36
	1000	7	8	13	15
	3500	63	71	176	200
	10000	436	436	1211	2585
Random Rules fixed	**350**	4	8	25	96
	1000	13	24	63	193
	3500	48	87	209	561
	10000	147	257	597	1490
Random Forest PARF	**350**	2	5	15	59
	1000	7	18	58	249
	3500	36	96	339	1470
	10000	138	382	1551	X

Table 5 Execution time for F2 type of relation

	Number of examples	Number of attributes			
		35	**100**	**350**	**1000**
Random Rules default	**350**	3	3	4	7
	1000	7	6	8	13
	3500	86	90	109	163
	10000	872	761	1022	1430
Random Rules fixed	**350**	4	9	29	102
	1000	12	24	71	216
	3500	46	84	227	589
	10000	141	256	625	1525
Random Forest PARF	**350**	2	4	15	59
	1000	7	17	58	250
	3500	32	90	337	1495
	10000	123	356	1568	X

Table 6 Execution time for R1 type of relation

	Number of examples	Number of attributes			
		35	**100**	**350**	**1000**
Random Rules default	**350**	4	178	X	X
	1000	13	476	X	X
	3500	53	1846	X	X
	10000	164	5076	X	X
Random Rules fixed	**350**	4	8	22	86
	1000	12	21	58	175
	3500	45	74	183	481
	10000	139	219	516	1276
Random Forest PARF	**350**	2	6	18	77
	1000	9	23	80	375
	3500	49	128	528	X
	10000	184	511	2500	X

Analysis of the results

The evaluation results presented in Tables 1-6 demonstrate that:

a) RandomRules (RR) performed well on synthetic data. All inductions finished successfully except that execution time for some large datasets has been so long that it has been intentionally interrupted.

b) Predictive accuracy achieved by RR (both in the default and the fixed mode) is satisfactory and it is typically better than the accuracy obtained by PPARF. It can be noticed how for all three tested approaches (RR_default, RR_fixed, and PARF) predictive accuracy *increases* when the number of examples *increases* (it is easier to identify correct function from larger datasets) and that predictive accuracy *decreases* when number of attributes *increases* (it is much more difficult to identify relevant relation when there are many random attributes). Such behavior is expected and the result demonstrates that RR behaves reasonable.

c) Execution time both for RR_default and RR_fixed are satisfactory and comparable with those obtained by PARF. Typically for small datasets RR_default is faster than RR_fixed while it is slower for larger datasets. Execution time for RR_fixed is very regular: if number examples or number of attributes increases by factor 10 then execution time increases 10 times. RR_default is unpredictable in respect to the execution time and typically it significantly increases with the number of examples (for increase of number of examples by factor 10, execution time increases 20-200 times) while when number of attributes increases 10 times then execution time increases only by factor 1.5 – 3 (except for the random function R1).

d) For the random function R1 all three approaches practically in all cases successfully identified majority class voting as the optimal strategy. RR_default had significant problems with the execution time; *execution time significantly increased both when number of examples and number of attributes has increased.*

e) A surprising effect has been detected in respect of the accuracy. For function F2 RR_fixed has better accuracy than RR_default although it is expected that RR_default will be better for all function types in the same way as it has demonstrated for function F1.

Discussion

The evaluation demonstrated that RandomRules is a stable algorithm applicable on diverse predictive tasks. The algorithm behaves reasonable both in respect of the achieved accuracy and the execution time. The complexity of the core of the algorithm (RR in the fixed mode) is linear both in respect of the number of examples and the number of attributes. Very good news is that time complexity is growing slower than for the PARF implementation resulting by better performance on larger datasets. Even more potentially relevant is that is that achieved predictive accuracy is better than for PARF

but this has to be further evaluated on other datasets and other Random Forest implementations.

The main result of the evaluation is detection of some serious problems related with RR_default. In this RR implementation we have tried to automatically identify the necessary number of generated rules so that rules are generated as long as there is increase in respect of the estimated predictive accuracy. The advantages of such an approach can be seen from the experiments with relation F1. The achieved predictive accuracy is systematically better than for RR_fixed and it is true especially if the number of attributes is large. Additionally, for small datasets execution time is shorter because induction can stop earlier if optimal predictive accuracy is achieved. But this approach has the problem to stop the rule generation process when there is no clear optimal point of the predictive accuracy. In case of a random dataset there is no at all improvement and the process had to be stopped some time-out criteria. The results demonstrate that the currently implemented criteria should be improved in order to enable that rule generation stops also for large random datasets in a reasonable time.

Even more serious is the problem detected for relation type F2. In contrast to F1 this is a very difficult, although well-defined classification task. In this situation it seems that the RR_defualt approach by insisting on predictive accuracy practically overfits the constructed model and achieves worse predictive accuracy. It can be noticed that the accuracy of RR_default is still comparable to the accuracy of PARF but the result of RR_fixed approach clearly demonstrate that there is space for the improvements. At this stage of the RR development it is not clear how to implement a more flexible RR_default which will optimally work also on very difficult classification concepts.