A Rule Evaluation Support Method with Learning Models Based on Objective Rule Evaluation Indexes

Hidenao Abe[1], S. Tsumoto[1], M. Ohsaki[2], T. Yamaguchi[3]

Dept. of Medical Informatics, Shimane University, School of Medicine, Japan[1]

Faculty of Engineering, Doshisha University, Japan[2]

Faculty of Science and Technology, Keio University, Japan[3]

Outline

- Background and Research Issues
- Rule Evaluation Support Method based on Objective Rule Evaluation indexes
- Comparisons of Leaning Algorithms for Rule Evaluation Model Construction
- Conclusion

Background

- It is difficult for human experts to evaluate many thousands of rules completely.

- Many efforts have done to select rules with single objective index such as recall, precision, and so forth.
- At least 40 objective interestingness measures are developed and investigated to express a human evaluation criterion.

- Ohsaki et al. investigated the relationship between each index and criterion of an expert. However, no single objective index can express the human criterion exactly. [Ohsaki04].
- Applicable domain of these interestingness measures have been never generalized.

Research Issues

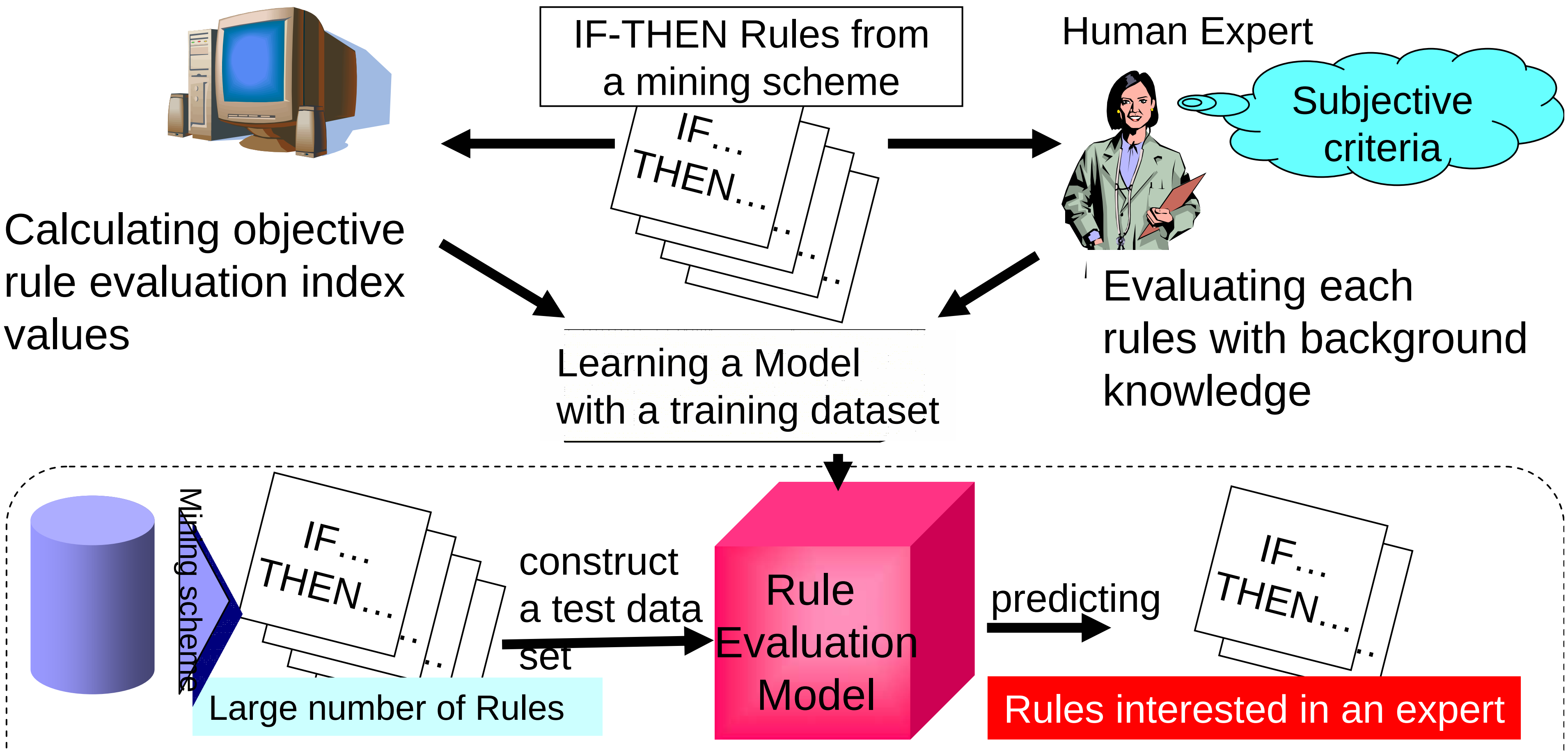
- A novel rule evaluation support method with rule evaluation models (REMs).
  - The system obtains a dataset to combine multiple objective indexes and evaluations from a human expert.

- Detailed issues of our rule evaluation support method
  - To construct more accurate REMs to support human experts more exactly
  - To construct a valid REM with smaller training dataset
  - To construct a reasonable REMs to given human evaluation

Outline

- Background
- Rule Evaluation Support Method based on Objective Rule Evaluation indexes
- Comparisons of Leaning Algorithms for Rule Evaluation Model Construction
- Conclusion

Overview of the rule evaluation support with REMs



Choosing valuable rules with predicting evaluation labels based on rule evaluation index values of the given rules

Outline

- Background
- Rule Evaluation Support Method based on Objective Rule Evaluation indexes
- Comparisons of Leaning Algorithms for Rule Evaluation Model Construction
- Conclusion

Comparisons of learning algorithms

- Comparison on an actual datamining result
  - To evaluate the availability on solid evaluations from a domain expert
- Comparison on rule sets of benchmark datasets with artificial class distributions
  - To evaluate the availability on evaluations without any particular human criterion

- Evaluation viewpoints for these comparisons:
  - Accuracies to the whole dataset and Leave-One-Out validation, and their recalls and precisions of each class label
  - Estimating minimum size of training subset to construct valid REMs with learning curves
  - Looking at elements of REMs from an actual data mining result

Objective Rule Evaluation indexes

calculated on a validation dataset for each classification rule

The 39 objective indexes [Ohsaki 04]

- Based on probability (26 indexes)
  - Coverage, Prevalence, Precision, Recall, Support, Accuracy, Specificity, Lift, Leverage, Added Value, Relative Risk, Jaccard, Certainty Factor, Odds ratio, Yule's Q, Yule's Y, Kappa, Koelesgen's Interestingness, Brin's Interestingness,Brin's Conviction, GOI, Credibility, KSI, Laplace Correction, Collective Strength
- Based on test statistics (3 indexes)
  - Chi-Square(with only True/Positive, with a whole confusion matrix) , Gini Gain
- Based on information theory (6 indexes)
  - Mutual Information, J-Measure, YLI1, YLI2, YZI, K-Measure
- Based on number of instances (3 indexes)
  - Φ coefficient, PSI , Cosine Similarity
- Based on similarity between rules on a validation dataset (2 indexes)
  - GBI, Peculiarity

Learning algorithms for comparisons

- **Decision Tree**: J4.8 (an Java implementation of C4.5)
- **Neural Network**: BPNN (with back-propagation)
  - Parameters of BP: learning rate=0.3, momentum= 0.2
  - Each unit corresponds to each class label in output layer
- **Classification Via Linear Regression**: CLR
  - Linear regression expressions: “1-the other” for each class label
  - explanatory variable selection: greedy search with AIC
- **SVM**: Sequential Minimal Optimization [Platt98]
  - SVM for multiple class: learning “1-the other” expressions for each class label
  - Kernel function setting: polynomial kernel
- **OneR**
  1. sorting with single objective index
  2. setting thresholds based on class labels
  3. constructs a rule set with the objective index

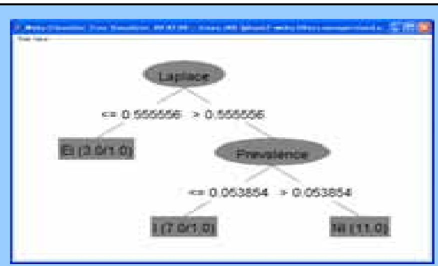
The Flow of the comparison with the meningitis datamining result [Hatazawa 00]

meningitis datasets having 140 inst.
6 (2×3) kinds of diagnostic problems

244 rules

The 39 objective indexes obtain attributes of the training dataset

Evaluations from a medical expert are added as class labels



- decision tree
- neural network
-support vector machine
-etc..

Model Construction

Evaluation on:
- Performance
- Estimating minimum training sub-set
- Contents of REMs

Sample of the data set

ruleID	Accuracy	Added_Value	・・・	YulesQ	YulesY	HumanExpert
Rule1	0.81	0.41	・・・	0.73	0.44	NI
Rule10	0.81	0.43	・・・	0.75	0.45	NI
Rule11	0.85	0.46	・・・	0.79	0.49	I
Rule12	0.84	0.56	・・・	0.87	0.58	I
Rule13	0.94	0.44	・・・	0.88	0.59	I
Rule14	0.81	0.43	・・・	0.75	0.45	NI

← 39 objective rule evaluation indexes →

Performance Comparison of the five algorithms
(All of rules =244, 'I'=48(19.7%), 'NI'=187(76.6%), 'NU'=9(3.7%))

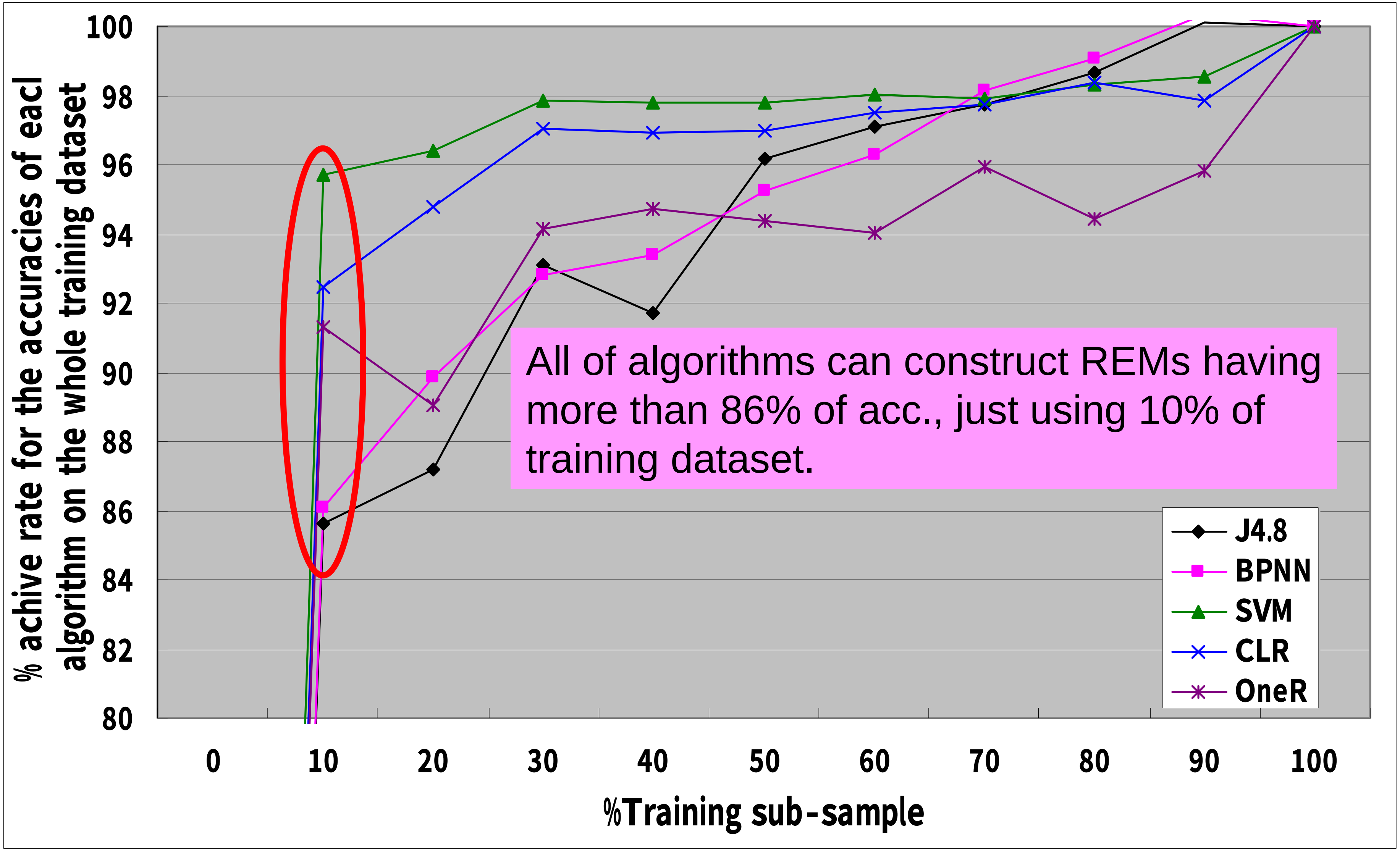
Learning Algorithms	Evaluation on the Whole Training Dataset						
	Acc.	Recall			Precision		
		I	NI	NU	I	NI	NU
J4.8	85.7	41.7	97.9	66.7	80.0	86.3	85.7
BPNN	86.9	81.3	89.8	55.6	65.0	94.9	71.4
SVM	81.6	35.4	97.3	0.0	68.0	83.5	0.0
CLR	82.8	41.7	97.3	0.0	71.4	84.3	0.0
OneR	82.0	56.3	92.5	0.0	57.4	87.8	0.0

Learning Algorithms	Evaluation with Leave－One-Out(LOO)						
	Acc.	Recall			Precision		
		I	NI	NU	I	NI	NU
J4.8	79.1	29.2	95.7	0.0	63.6	82.5	0.0
BPNN	77.5	39.6	90.9	0.0	50.0	85.9	0.0
SVM	81.6	35.4	97.3	0.0	68.0	83.5	0.0
CLR	80.3	35.4	95.7	0.0	60.7	82.9	0.0
OneR	75.8	27.1	92.0	0.0	37.1	82.3	0.0

1. J4.8 and BPNN achieve higher than 85.7% of acc. with more than 77.5% reliability.
   (BPNN tend to be over fitting, looking at it's LOO acc., recalls and precisions)
2. To predict very minor class 'NU', a proper learning algorithm will be needed.

Leaning curves on achieve rates

(achieve rate = (acc. of each sub-sample / acc. of whole sample) *100)

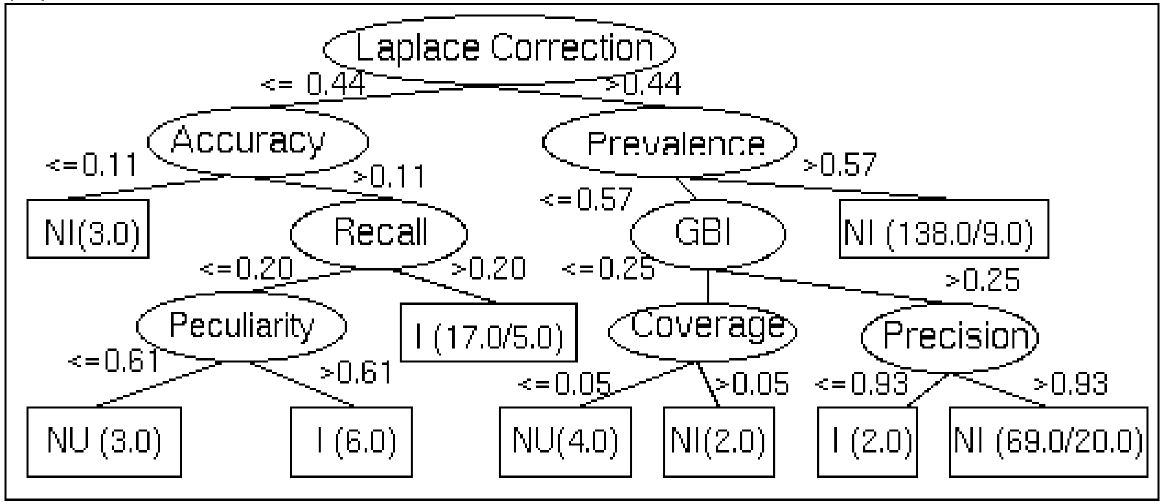


Rule Evaluation Models from the actual datamining result

(a) The rule set from OneR

```
IF                     YLI1 < 0.02 THEN "I"
IF YLI1 >= 0.02 and YLI1 < 0.29 THEN "NI"
IF YLI1 >= 0.29 and YLI1 < 0.43 THEN "I"
IF YLI1 >= 0.43 and YLI1 < 0.44 THEN "NI"
IF YLI1 >= 0.44 and YLI1 < 0.55 THEN "I"
IF YLI1 >= 0.55 and YLI1 < 0.63 THEN "NI"
IF YLI1 >= 0.63 and YLI1 < 0.83 THEN "I"
IF YLI1 >= 0.83                THEN "NI"
```

(b) The decision tree from J4.8



(c) The linear regression expressions from CLR

```
NU =                               NI =                              I =
  0.6202 * Specificity +             1.7173 * Precision +             -1.4417 * Precision +
  0.6224 * Accuracy +               -0.5063 * Accuracy +              -0.7286 * Specificity +
 -1.1384 * Leverage +                0.5673 * RelativeRisk +           0.4085 * Lift +
 -0.6895 * RelativeRisk +           -1.2718 * CertaintyFactor +        0.6297 * CertaintyFactor +
  0.3704 * CertaintyFactor +         0.5955 * YulesQ +                -1.4477 * CollectiveStrength +
  0.5722 * OddsRatio +              -0.4609 * K-Measure +              1.5449 * GiniGain +
  0.7656 * BI +                      0.4613 * PSI +                   -0.5318 * PSI +
 -0.222  * Credibility +            -0.4181 * Peculiarity +            0.4981 * Peculiarity +
 -0.3941 * LaplaceCorrection +       0.5302                            1.4872
  0.7986 * GiniGain +
 -0.0966 * GBI +
 -0.8895
```

Datasets from rule sets learned with four UCI benchmark data

(To make sure the availability of our method without any human criteria)

		#Mined Rules	#Class labels			%Def. class
			L1	L2	L3	
Distribution I			(0.30)	(0.35)	(0.35)	
	Mushroom	30	8	14	8	46.7
	InternetAd	107	26	39	42	39.3
	Heart	318	97	128	93	40.3
	Letter	6340	1908	2163	2269	35.8
Distribution II			(0.30)	(0.50)	(0.20)	
	Mushroom	30	11	16	3	53.3
	InternetAd	107	30	53	24	49.5
	Heart	318	99	140	79	44.0
	Letter	6340	1890	3198	1252	50.4
Distribution III			(0.30)	(0.65)	(0.05)	
	Mushroom	30	7	21	2	70.0
	InternetAd	107	24	79	9	73.8
	Heart	318	98	205	15	64.5
	Letter	6340	1947	4062	331	64.1

*All of rule sets are obtained by bagged PART with Weka [Witten 00]

Performances of REMs on the training datasets with three kinds of class distributions

	J48	BPNN	SVM	CLR	OneR
Distribution I					
Mushroom	80.0	93.3	56.7	66.7	53.3
InternetAd	84.1	82.2	29.9	53.3	60.7
Heart	78.0	75.8	40.3	42.5	54.7
Letter	36.8	36.4	30.1	36.6	52.1
Distribution II					
Mushroom	93.3	93.3	80.0	80.0	76.7
InternetAd	73.8	79.4	49.5	59.8	60.7
Heart	72.3	69.2	35.9	47.8	55.7
Letter	51.0	51.0	50.4	50.4	57.0
Distribution III					
Mushroom	93.3	96.7	70.0	70.0	76.7
InternetAd	86.0	90.7	70.1	69.2	72.0
Heart	78.0	77.7	64.5	65.7	71.4
Letter	64.1	64.3	64.1	64.1	68.3

- Performances of algorithms are suffered from probabilistic class distribution especially in larger datasets such as Heart(318 inst.) and Letter (6340 inst).
- Hyper-plain type learner (SVM and CLR) could not work well, because of the probabilistic class distributions.

Estimation of minimum training subset to construct valid REMs (from learning curve analysis)

		J48	BPNN	SVM	CLR	OneR
Distribution I						
	Mushroom	8	8	12	18	14
	InternetAd	14	14	-	30	14
	Heart	42	31	66	114	98
	Letter	189	217	-	955	305
Distribution II						
	Mushroom	6	4	4	6	12
	InternetAd	24	24	52	42	70
	Heart	52	40	-	104	92
	Letter	897	>1000	451	-	>1000
Distribution III						
	Mushroom	22	14	22	28	22
	InternetAd	80	66	-	-	-
	Heart	114	94	142	318	182
	Letter	>1000	>1000	998	>1000	>1000

- In Dist. I and II, almost learner succeeded in learning valid REMs with less than 20% of each data set.
- It is more difficult to construct valid REMs with smaller training subset on 'Distribution III', which has unbalanced class distribution.

-> If we construct REMs without particular human criterion, we should prepare small (<100) dataset with balanced class distribution.

Outline

- Background
- Rule Evaluation Support Method based on Objective Rule Evaluation indexes
- Comparison of Leaning Algorithms for Rule Evaluation Model Construction
- Conclusion

Conclusion

- Summary
  - Comparing learning algorithms to construct rule evaluation models for supporting a post-processing of data mining exactly
    - Our method can construct valid rule evaluation models with the 39 objective rule evaluation indexes at least the five learning algorithms.
    - The algorithms have been able to construct valid rule evaluation models with 10% of training subset with evaluations based on solid expert's criterion.
- Feature works
  - Introducing algorithm selection
    - attribute construction and selection algorithm selection
    - model learning algorithm selection
  - Applying this method to other data from other domains