Class Association Rule Mining Using Multi-Dimensional Numbered Information Spaces

Iliya Mitov

universiteit hasselt

imob

INSTITUUT
VOOR MOBILITEIT

Institute
of
Mathematics
and
Informatics

IMI

BULGARIAN ACADEMY
OF SCIENCES

15.11.2011, Hasselt University, Belgium

The goals

The goals of this thesis are two-fold:

- to introduce a parameter-free class association rule algorithm, which focuses primarily on the confidence of the association rules and only in a later stage on the support of the rules;
- to show the advantages of using multi-dimensional numbered information spaces for developing memory structuring in data mining processes on the example of implementation of the proposed class association rule algorithms.

To achieve these goals I:

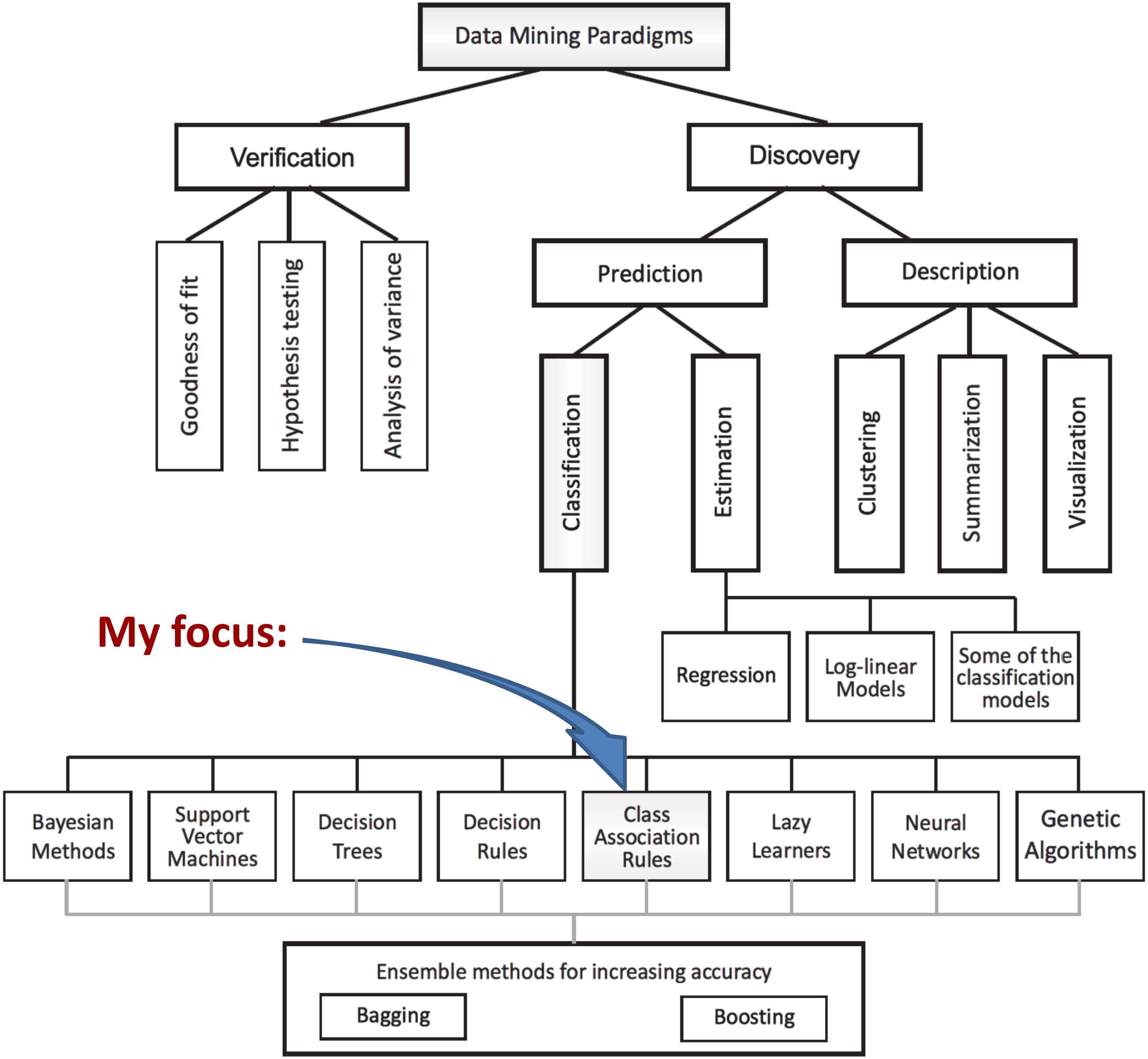
- developed a pyramidal multi-dimensional model for memory organization in classification systems;
- implemented the corresponding experimental classification system;
- conducted experiments and evaluation of the results for testing the hypothesis.

The Structure

1. Introduction
2. Data Mining and Knowledge Discovery
3. CAR Algorithms
4. Multi-Dimensional Numbered Information Spaces
5. PGN and MPGN Algorithms
6. Program Realization
7. Example of Lenses Dataset
8. Sensitivity Analysis
9. Conclusions and Future Work

Appendix

Data Mining


Data Mining Paradigms
Verification
Discovery
Goodness of fit
Hypothesis testing
Analysis of variance
Prediction
Description
Classification
Estimation
Clustering
Summarization
Visualization
Regression
Log-linear Models
Some of the classification models
My focus:
Bayesian Methods
Support Vector Machines
Decision Trees
Decision Rules
Class Association Rules
Lazy Learners
Neural Networks
Genetic Algorithms
Ensemble methods for increasing accuracy
Bagging
Boosting

CAR Classifiers

CAR Classifiers

They appeared initially within the field of market basket analysis for discovering interesting rules from large data collections.

Some advantages of CAR-classifiers:

- The training is very efficient regardless of the size of the training set;
- Training sets with high dimensionality can be handled with ease and no assumptions are made on dependence or independence of attributes;
- The classification is very fast;
- Classification based on association methods presents higher accuracy than others classification methods;
- The classification model is a set of rules easily interpreted by human beings and can be edited.

CAR Classifiers

The structure of CAR-classifiers consists of:

1. Association rule mining
2. Pruning (optional)
3. Recognition

Examples of CAR Classifiers:

- CBA [Liu et al, 1998]
- CMAR [Li et al, 2001]
- ARC-AC and ARC-BC [Zaïane and Antonie, 2002]
- CPAR [Yin and Han, 2003]
- CorClass [Zimmermann and De Raedt, 2004]
- ACRI [Rak et al, 2005]
- TFPC [Coenen and Leng, 2005]
- HARMONY [Wang and Karypis, 2005]
- MCAR [Thabtah et al, 2005]
- CACA [Tang and Liao, 2007]
- ARUBAS [Depaire et al, 2008]

Association Rule Mining

Several techniques for creating association rules are used:

- Apriori algorithm (CBA, ARC-AC, ARC-BC, ACRI, ARUBAS);
- FP-tree algorithm (CMAR);
- FOIL algorithm (CPAR);
- Morishita & Sese Framework (CorClass).

Generating association rules can be made:

- from all training transactions together (CBA,CMAR, ARC-AC)
- for transactions grouped by class label (ARC-BC)

Pruning

Types of pruning:

- pre-pruning - in parallel with creating association rules
- post-pruning - after that

Different heuristics for pre-pruning are used, based on:

- minimum support
- minimum confidence
- different kinds of error pruning

Different criteria in post-pruning phase:

- support, confidence, cardinality
- data coverage (ACRI)
- correlation between consequent and antecedent (CMAR)

Recognition

Different approaches can be discerned in the recognition stage:

- using a single rule (CBA)
- using a subset of rules (CPAR)
- using all rules (CMAR)

Order-based combined measures for a subset or all rules:

- Select all matching rules
- Group rules per class value
- Order rules per class value according to criterion
- Calculate combined measure for best Z rules
- Laplace Accuracy (CPAR)

PGN and MPGN Classifiers

PGN Classifier

- Typical for CAR classifiers:
  firstly take in account the support of the association rule, after that the confidence.

- We study a new associative classifier algorithm, which turns the priorities around and focuses on confidence first by retaining only 100% confidence rules.
- The main goal is to verify the quality of the confidence-first concept.

PGN Classifier

PGN Classifier is based on:

- The association rule mining goes from the longest rules (instances) to the shorter ones until no intersections between patterns in the classes are possible.
- At the first step of the pruning phase the contradictions of more general rules between classes are cleared.
- After that the pattern set is compacted excluding all more concrete rules within the classes.

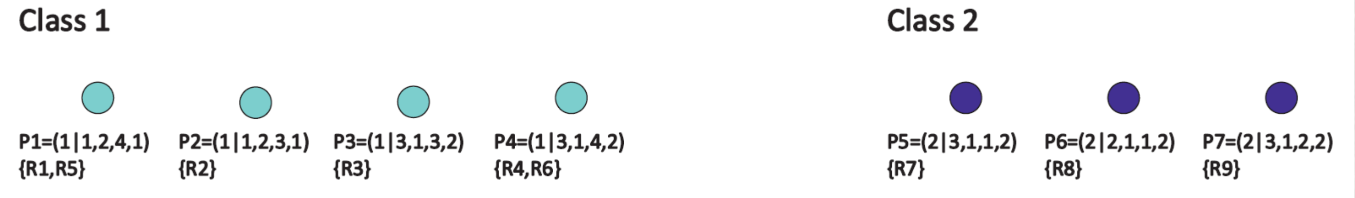
PGN Classifier

The steps will be visualized on the example of this dataset:

```
R1:  (1| 1, 2, 4, 1)
R2:  (1| 1, 2, 3, 1)
R3:  (1| 3, 1, 3, 2)
R4:  (1| 3, 1, 4, 2)
R5:  (1| 1, 2, 4, 1)       Equal to R1
R6:  (1| 3, 1, 4, 2)       Equal to R4
R7:  (2| 3, 1, 1, 2)
R8:  (2| 2, 1, 1, 2)
R9:  (2| 3, 1, 2, 2)
```

PGN Training Process: Generalization

1. Adding instances to the pattern set.



Class 1

P1=(1|1,2,4,1) {R1,R5}

P2=(1|1,2,3,1) {R2}

P3=(1|3,1,3,2) {R3}

P4=(1|3,1,4,2) {R4,R6}

Class 2

P5=(2|3,1,1,2) {R7}

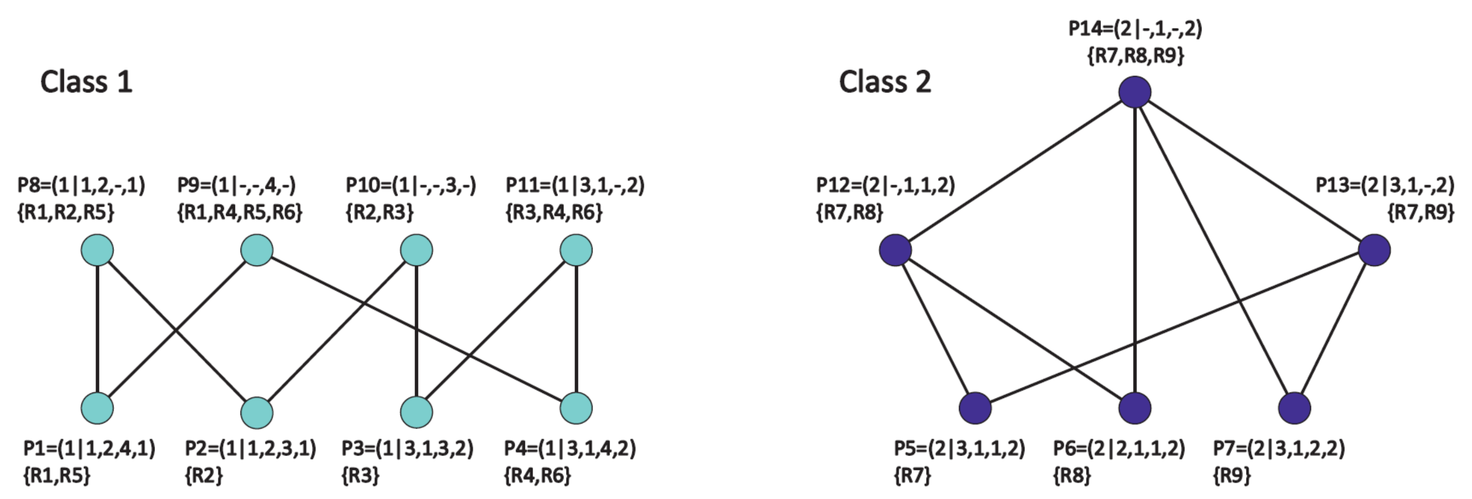
P6=(2|2,1,1,2) {R8}

P7=(2|3,1,2,2) {R9}

$$LS = \left\{ R^i \right\} \quad i = 1, \ldots, t$$

PGN Training Process: Generalization

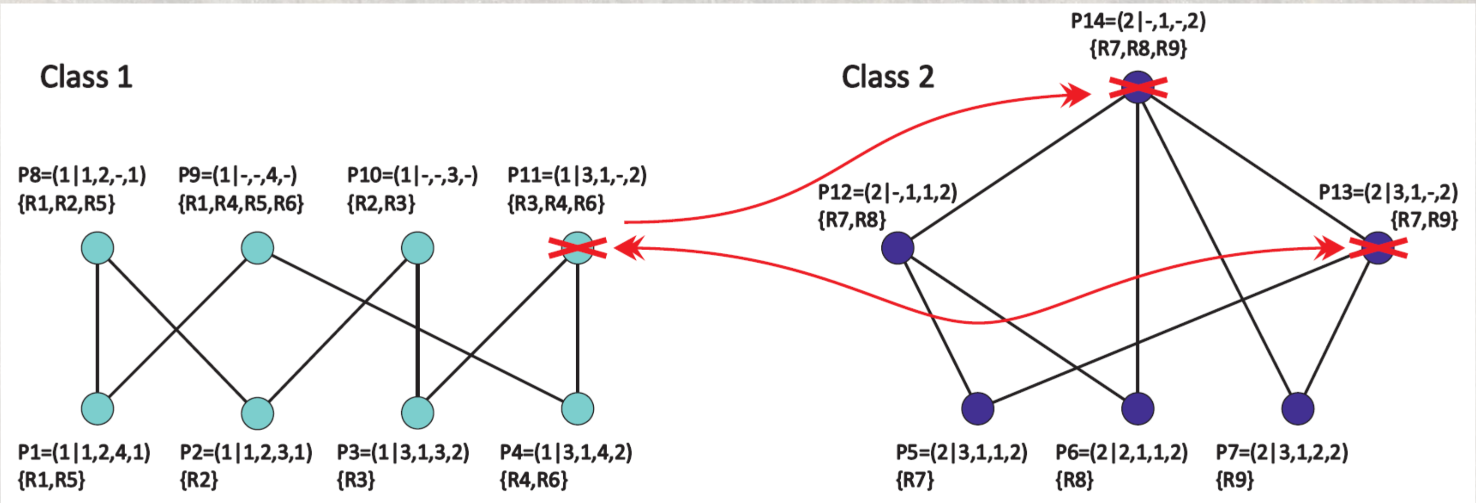
2. Creating all possible intersection patterns between patterns within the class



$$PS = \left\{ P^l \right\} \quad P^l : \begin{cases} R^l \in LS \\ P^l = P^i \cap P^j ;\; P^i \in PS, P^j \in PS, c^i = c^j ;\; \left| P^l \right| > 0 \end{cases}$$

PGN Training Process: Pruning

Deleting contradictory patterns as well as general patterns that have exception patterns in some other class.

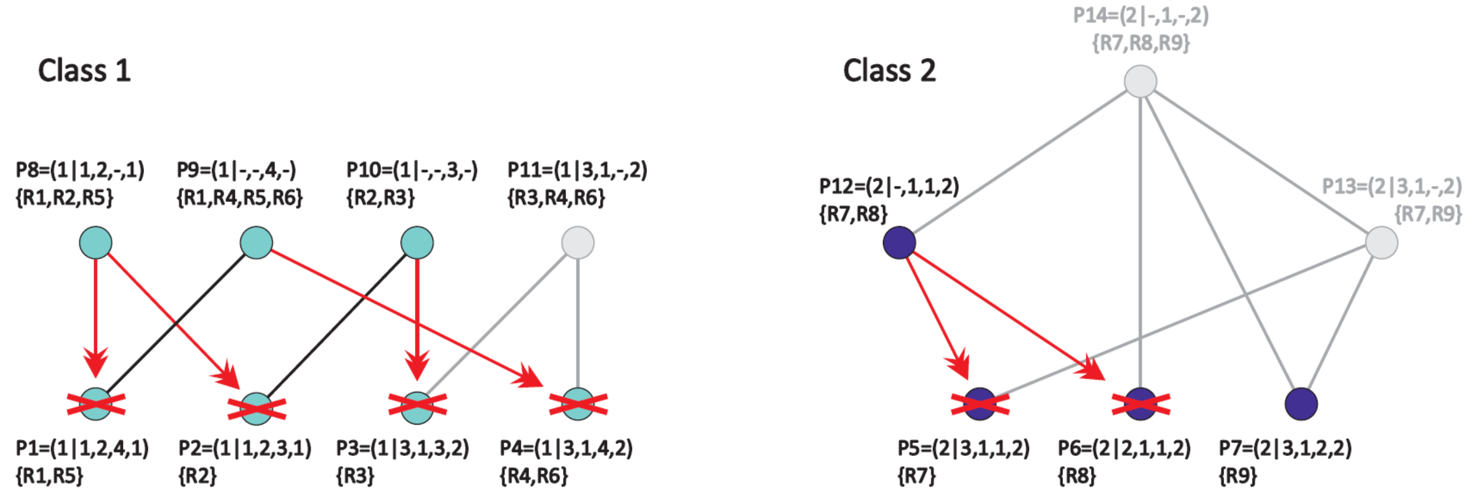


$$P^i, P^j \in PS,\ c^i \neq c^j : \begin{cases} \left|P^i \cap P^j\right| = \left|P^i\right| < \left|P^j\right| : \text{remove } P^i \\ \left|P^i \cap P^j\right| = \left|P^j\right| < \left|P^i\right| : \text{remove } P^j \\ \left|P^i \cap P^j\right| = \left|P^i\right| = \left|P^j\right| : \text{remove } P^i, P^j \end{cases}$$

PGN Training Process: Pruning

Removing more concrete patterns within the classes.

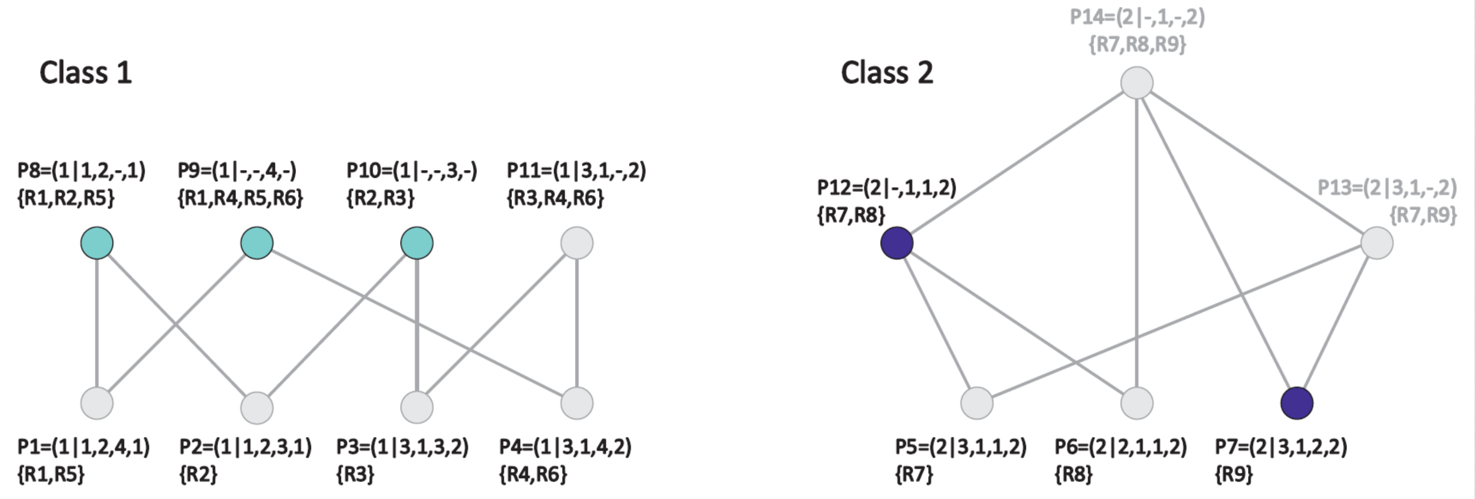
This step ensures compactness of the pattern set
that can be used in the recognition stage.



$$P^i, P^j \in PS,\ c^i = c^j : \begin{cases} \left|P^i \cap P^j\right| = \left|P^i\right| < \left|P^j\right| : \text{remove } P^j \\ \left|P^i \cap P^j\right| = \left|P^j\right| < \left|P^i\right| : \text{remove } P^i \end{cases}$$

PGN Training Process: Result

That are remaining patterns in the recognition model for the example dataset.



PGN: Recognition

Query: $Q = (? \mid a_1, a_2, \ldots, a_n)$

The association rule size corresponds to the number of input attributes which have a non-missing value:

$$\left|P\right| = \left|\{a_i \mid 1 \le i \le n-1, a_i \ne "-"\}\right|$$

The intersection percentage between pattern *P* and query *Q:*

$$IP(P,Q) = \frac{\left|P \cap Q\right|}{\left|P\right|}$$

PGN: Recognition

$$Q = (?\,|\,1,2,1,2)$$

	Pattern set	**P intersect. Q**	**IP(P,Q)**	**Support**	**Support set**
	Class 1				
P8	(1| 1, 2, -, 1)	(?| 1, 2, -, -)	**0.667**	3	**{R1,R2,R5}**
P9	(1| -, -, 4, -)	(?| -, -, -, -)	0	4	{R1,R4,R5,R6}
P10	(1| -, -, 3, -)	(?| -, -, -, -)	0	2	{R2,R3}
	Class 2				
P7	(2| 3, 1, 2, 2)	(?| -, -, -, 2)	0.250	1	{R9}
P12	(2| -, 1, 1, 2)	(?| -, -, 1, 2)	**0.667**	2	{R7,R8}

Equal IP: IP=0.667 for P8 ("class1") and for P12 ("class 2")
Support for "class 1" = 3 > Support for "class 2" = 2

=> The new instance is predicted to belong to "class 1"

PGN Advantages/Disadvantages

"+" **advantages** which are specific for all CAR classifiers:

- creating compact pattern set, used in the recognition stage
- easy interpretation of the results

"+" **advantages** which are specific for PGN:

- parameter free algorithm
- very good accuracy for clear datasets

"-" disadvantages

- exponential growth of operations during the process of creating the pattern set

MPGN Classifier

- MPGN is abbreviation from "Multi-layer Pyramidal Growing Networks" of information spaces.
- MPGN uses advantages of numbered information spaces.
- The main goal is to extend the possibilities of network structures by using a special kind of multi-layer memory structures called "pyramids", which permits defining and realizing of new opportunities.
- MPGN deals with instances and patterns separately for each class.
- This allows the MPGN algorithm to be implemented for use on parallel computers.

MPGN - Example Data Set

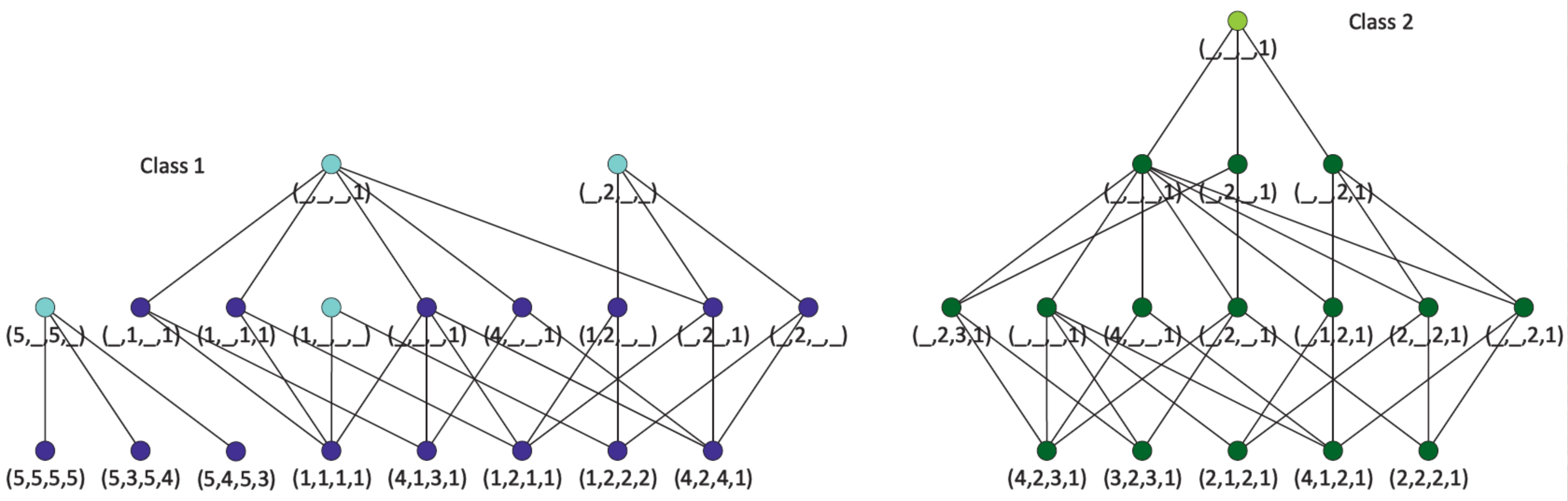
```
Class 1
    R1: (1|5,5,5,5)
    R2: (1|5,3,5,4)
    R3: (1|5,4,5,3)
    R4: (1|1,1,1,1)
    R5: (1|4,1,3,1)
    R6: (1|1,2,1,1)
    R7: (1|1,2,2,2)
    R8: (1|4,2,4,1)
Class 2
    R9: (2|4,2,3,1)
   R10: (2|3,2,3,1)
   R11: (2|2,1,2,1)
   R12: (2|4,1,2,1)
   R13: (2|2,2,2,1)
```

MPGN Training Process

The training process in MPGN consists of:

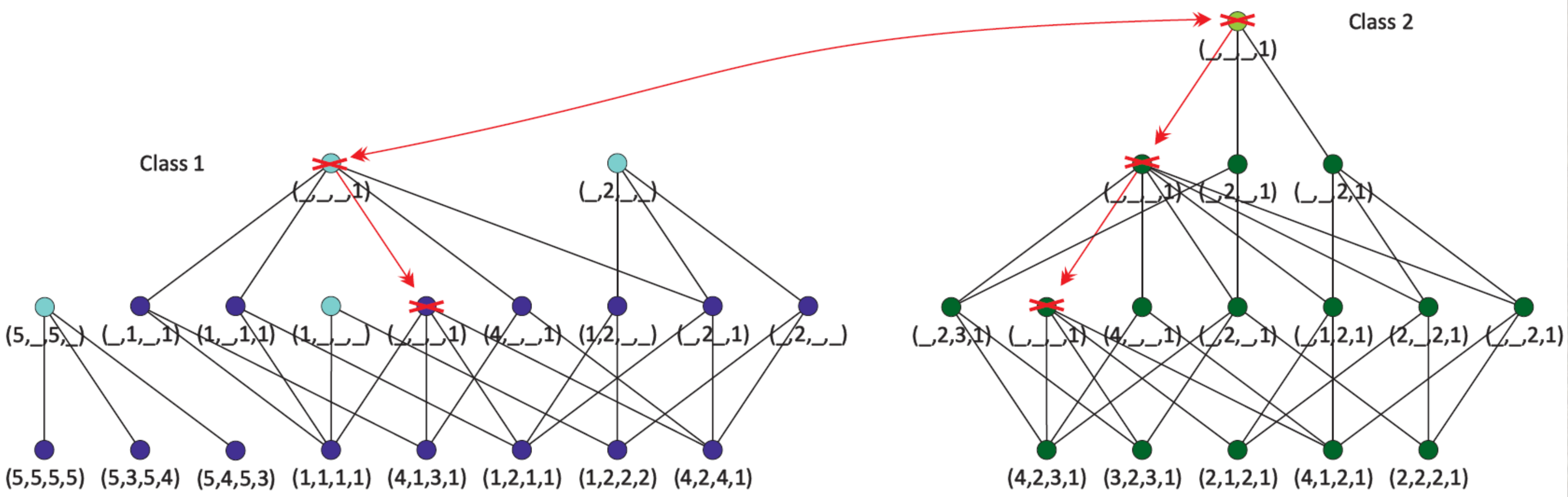
- preprocessing step
  - discretization of numerical attributes;
  - numbering the attributes' values.
- generalization step
  - chain of creating the patterns of upper layer as intersection between patterns from lower layer until new patterns are generated.
- pruning step
  - iterative analysis of vertex patterns of all pyramids from different classes and removing all contradictory vertex patterns.

MPGN Training Process: Generalization



Result of the generalization step of MPGN

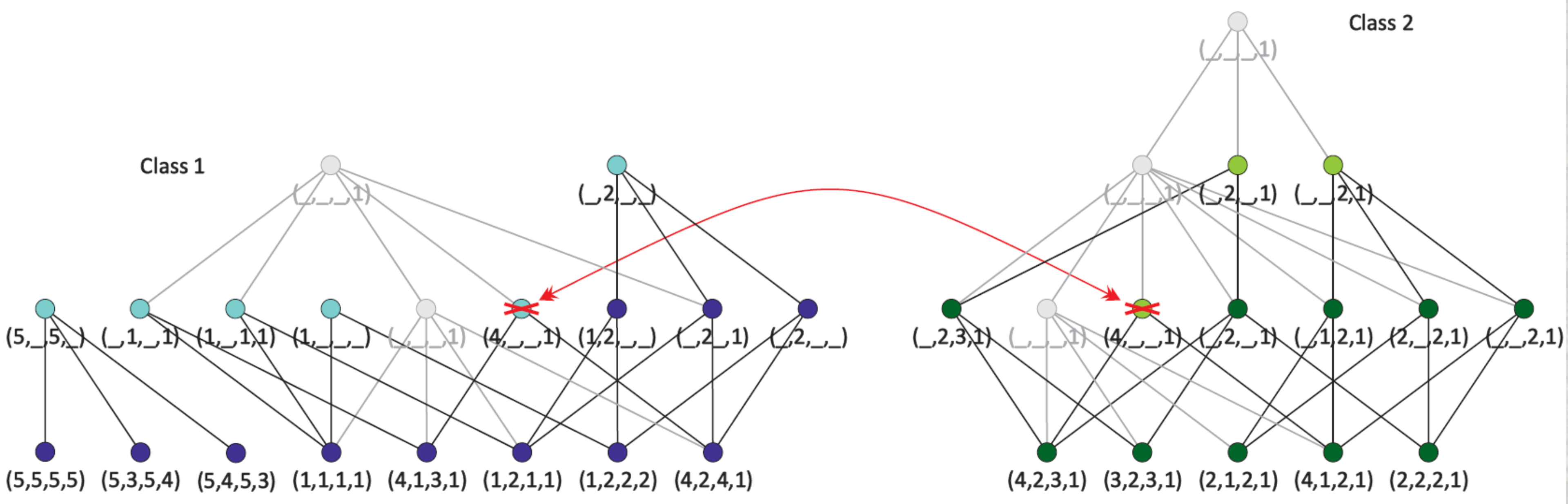
MPGN: Pruning



The start of the pruning process:

- vertexes of pyramids of class 1 and class 2 are compared
- contradictory vertexes are destroyed
- all equal patterns of them are destroyed too

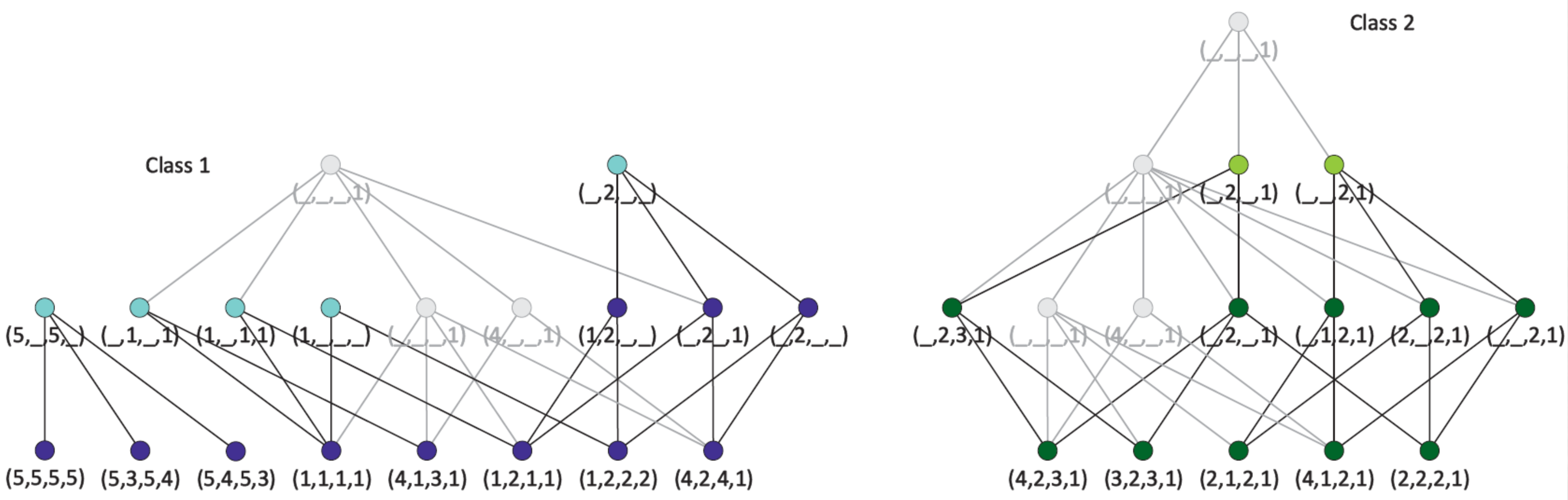
MPGN: Pruning



Next iteration:

- For new vertexes the searching and destroying of contradictory patterns are applied again.

MPGN: Pruning



The final result of the pruning process

MPGN: Recognition

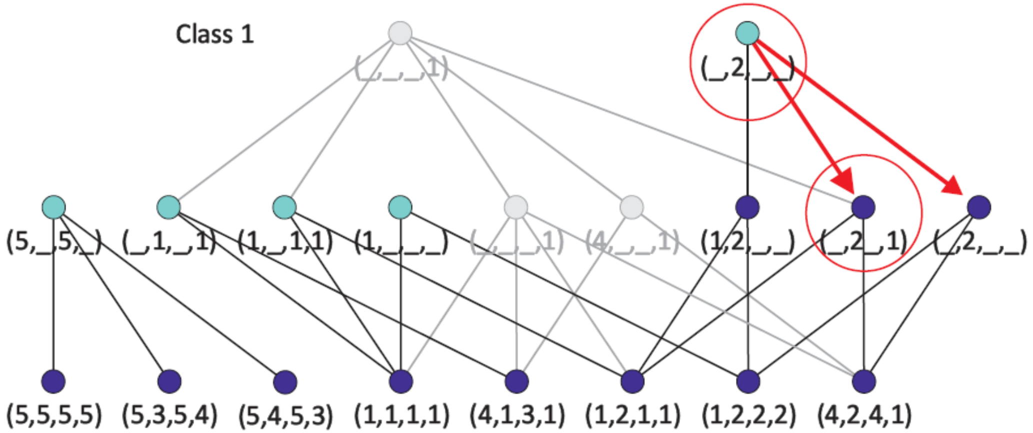
The recognition process consists of two main steps:

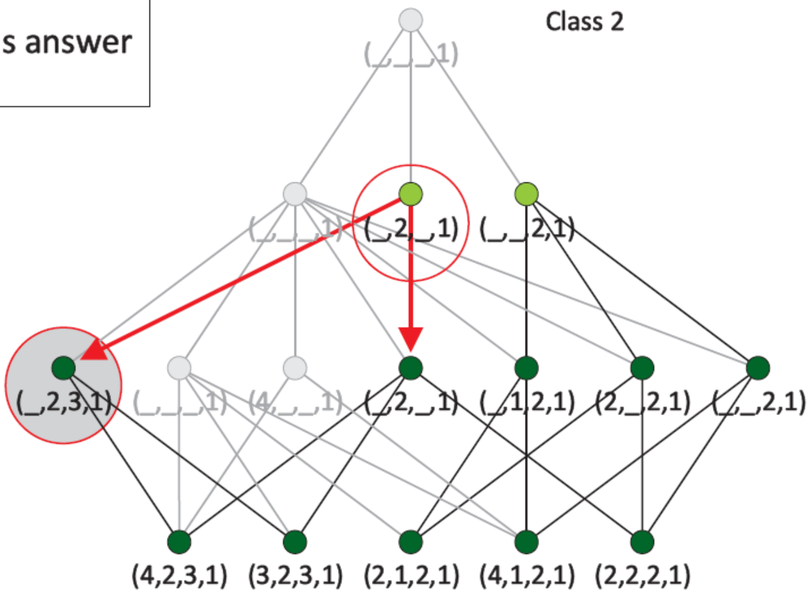
- creating recognition set for every class separately;
- analyzing resulting recognition sets from all classes and making decision which class to be given as answer.

MPGN: Recognition - Exit Point 1

Q=(5,2,3,1)

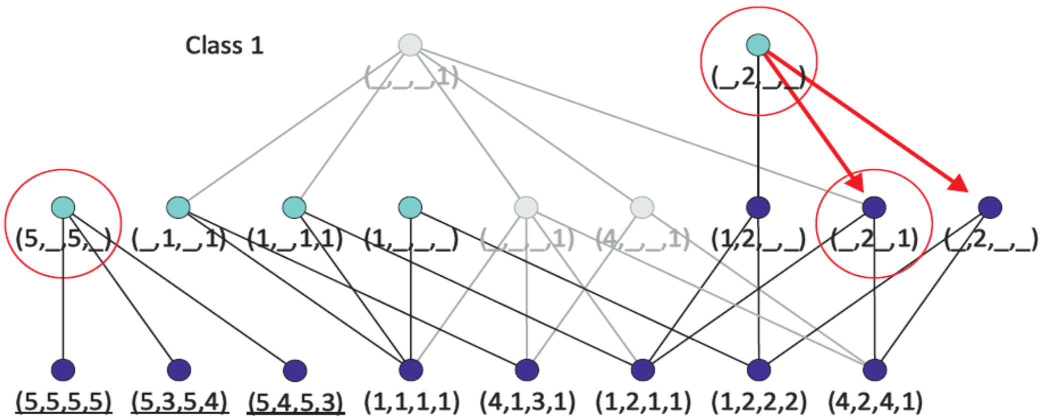
MaxCard(RS cl.1)=2 < MaxCard(RS cl.2)=3 -> Exit Point 1 -> Class 2 is answer

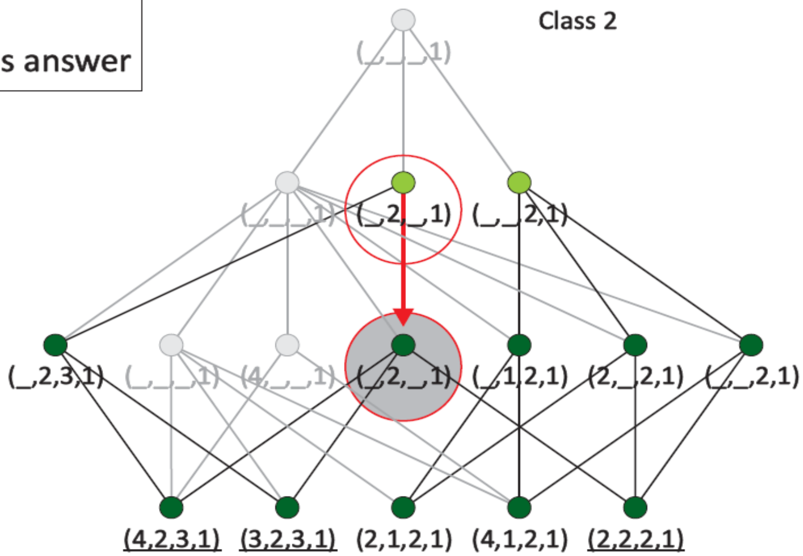




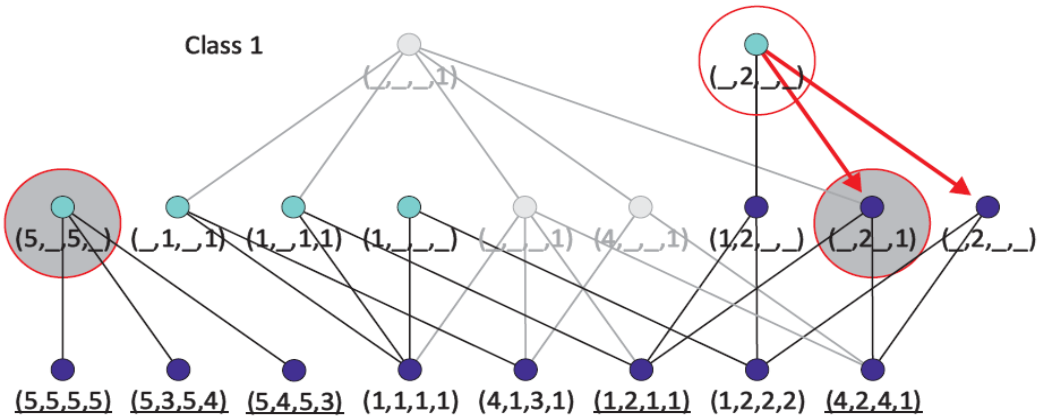
MPGN: Recognition - Exit Point 2/3

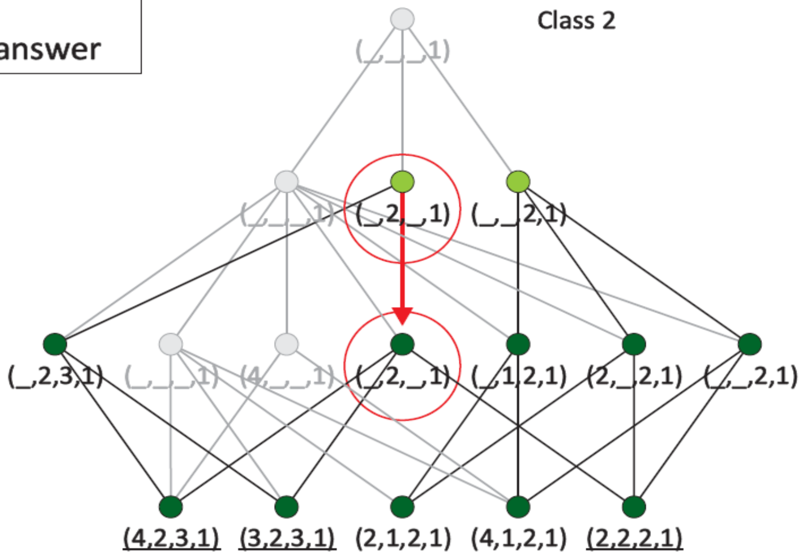
Q=(5,2,5,1)
MaxCard(RS cl.1)=MaxCard(Rs cl.2)=2 -> Exit point 2/3
Strategy S1: MaxConf(RS cl.1)=3/8 < MaxConf(RS cl.2)=3/5 -> Class 2 is answer





Q=(5,2,5,1)
MaxCard(RS cl.1)=MaxCard(Rs cl.2)=2 -> Exit point 2/3
Strategy S2: Conf(RS cl.1)=3/8+2/8 > Conf(RS cl.2)=3/5 -> Class 1 is answer





Program Realization - PaGaNe

PGN and MPGN are implemented in the frame of data mining environment PaGaNe. It uses the advantages of multi-dimensional numbered information spaces, provided by the access method ArM 32, such as the possibilities:

- to change searching with direct addressing in well-structured tasks;
- to build growing space hierarchies of information elements;
- to build interconnections between information elements stored in the information base.

- An important feature of the approaches used in PaGaNe, is the replacement of the symbolic values of the objects' features with integer numbers of the elements of corresponding ordered sets.
- All instances or patterns can be represented by a vector of integer values, which may be used as co-ordinate address in the corresponding multi-dimensional information space.

Sensitivity Analysis

The Experimental Datasets

from UCI Machine Learning Repository

Data set	Number of attributes	Number of classes	Number of instances	Type of attributes
audiology	69	24	200	Categorical
balance_scale	4	3	624	Categorical
blood_transfusion	3	2	748	Real
breast_cancer_wo	9	2	699	Categorical
car	6	4	1728	Categorical
cmc	9	3	1473	Categorical, Integer
credit	15	2	690	Categorical, Integer, Real
ecoli	7	8	336	Real
forestfires	12	2	517	Real
glass	9	6	214	Real
haberman	3	2	306	Integer
hayes-roth	4	3	132	Categorical
hepatitis	19	2	155	Categorical, Integer, Real
iris	4	3	150	Real
lenses	4	3	24	Categorical
monks1	6	2	432	Categorical
monks2	6	2	601	Categorical
monks3	6	2	554	Categorical
post-operative	8	3	90	Categorical, Integer
soybean	35	19	307	Categorical
tae	5	3	151	Categorical, Integer
tic_tac_toe	9	2	958	Categorical
wine	13	3	178	Integer, Real
winequality-red	11	6	1599	Real
zoo	16	7	101	Categorical, Integer

Global Frame of the Experiments

- We applied k-fold cross-validation (CV) technique.
- For achieving equality of used data for different classifiers we exported learning sets and examining sets from PaGaNe as input files in other programs (Weka, LUCS-KDD).
- For analysis of preprocessing discretizing step we used 3-fold CV with datasets with real attributes: Blood Transfusion, Ecoli, Forest Fires, Glass, and Iris.
- For studying the appropriate size of the learning set: 2,3,4,5-fold CV.
- For other experiments: 5-fold CV; all datasets are preprocessed with Chi-merge (95% significance level).

Comparison with Other Classifiers

CARs:

- CMAR: uses FP-growth method for rule generation and taking into account class label distribution in pruning phase;

Rules:

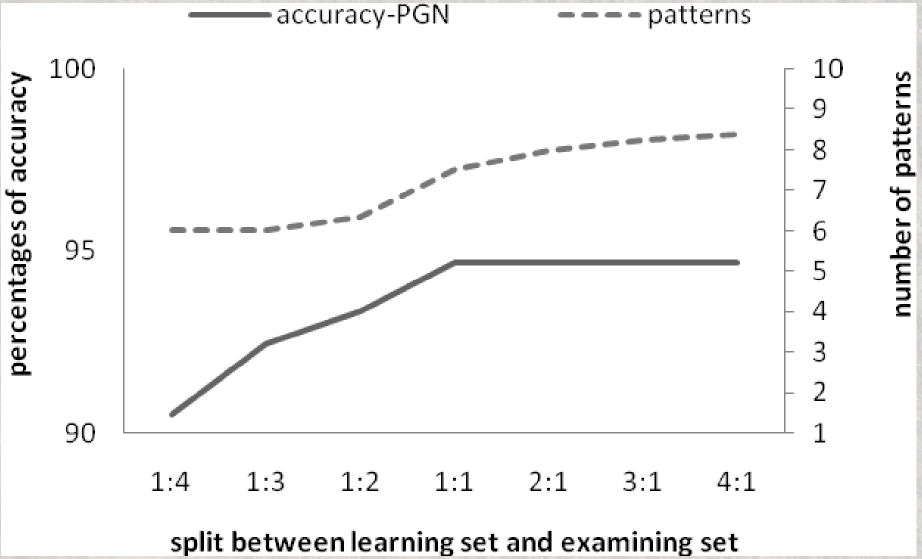
- OneR: one-level decision tree expressed in the form of a set of rules that all test one particular attribute;
- JRip: implementation a propositional rule learner, Repeated Incremental Pruning to Produce Error Reduction (RIPPER);

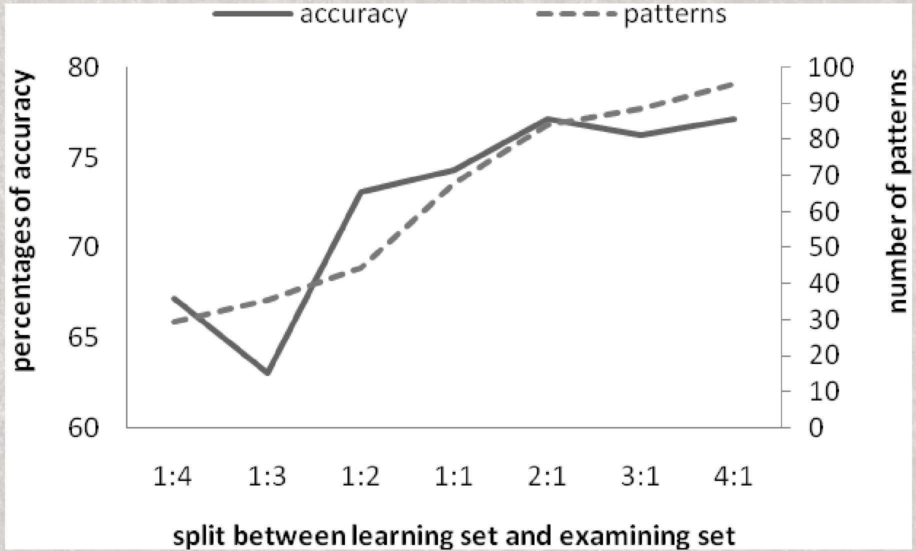
Trees:

- J48: a Weka implementation of C4.5 that produces a decision tree;
- REPTree: an extension of C4.5, which builds a decision tree using information gain reduction and prunes it using reduced-error pruning.

Studying the Size of the Learning Set (PGN)

The aim is to study the dependence of recognition accuracy from the size of the learning set.





"Iris" dataset -
half of the instances are enough to receive stable good recognition for Iris dataset.
The created model consists of about 8 patterns.

"Glass" dataset -
good recognition with relatively small number of patterns is in the case of about 140 instances for learning set (2:1).
Increasing the number of learning instances did not receive better accuracy, but expanded the pattern set.

Examining the Exit Points of MPGN

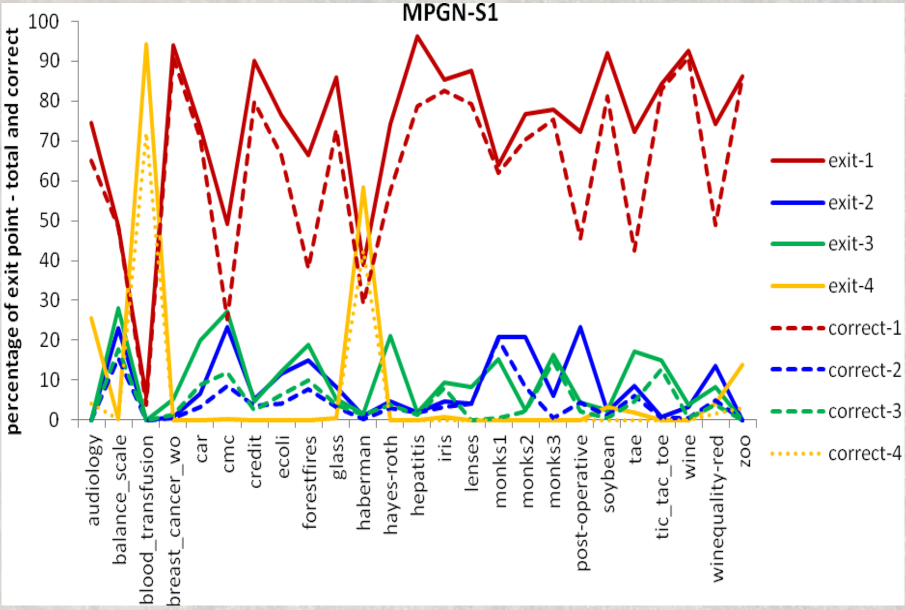
- Only one class is class-candidate - Exit point 1;
- Several classes are class-candidates:
  - Exit point 2 (maximal confidence in one class)*;
  - Exit point 3 (equal maximal confidence in more classes - maximal support in one class)

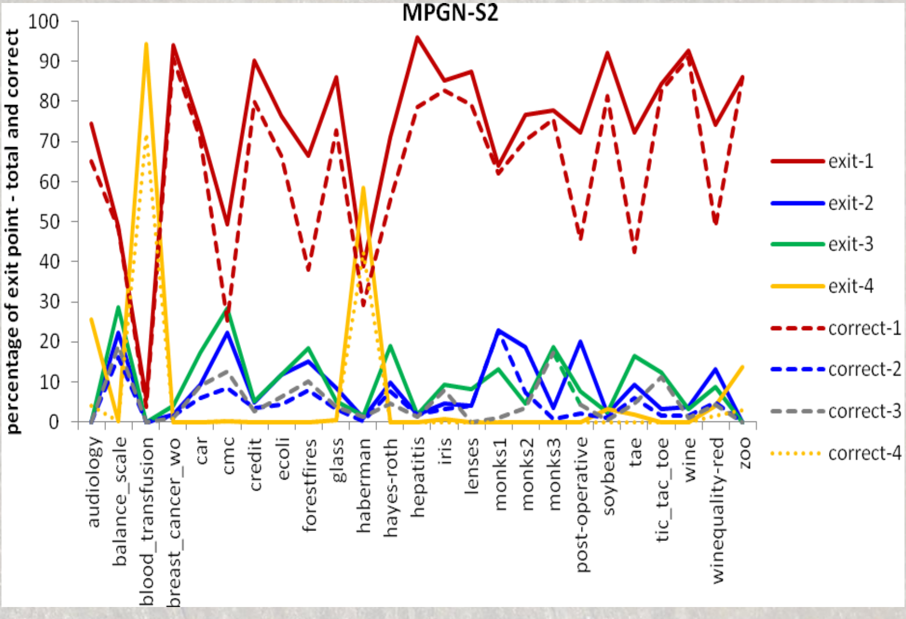
*) "maximal confidence" - different strategies:

- S1: from each class choose single rule with maximal confidence within the class;
- S2: find "confidence of recognition set", i.e. the number of instances that are covered of patterns from recognition set of this class over the number of all instances of this class.

- Empty recognition sets (another algorithm) – Exit point 4.

Examining the Exit Points of MPGN

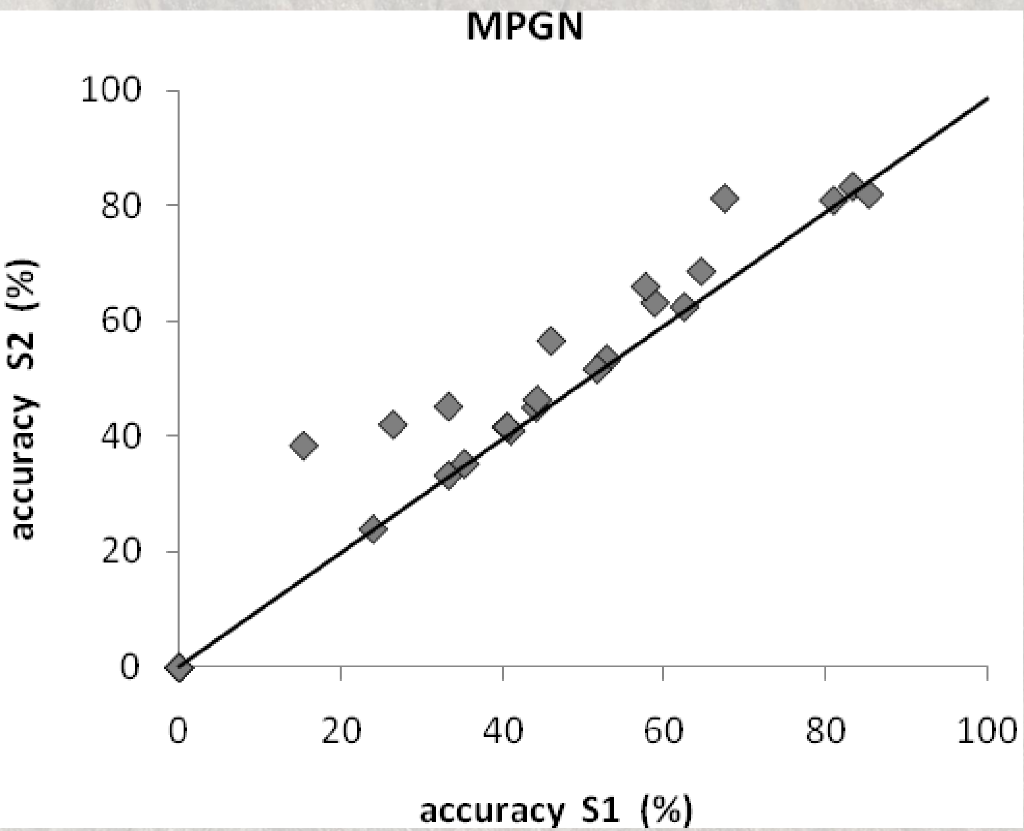




*) The unbroken line signs percentages of different kinds of exits, the dashed line signs percentage of correct ones (the number of correct exits divided by total number of queries).

- In most cases the recognition leads to Exit point 1, which means that applying of the MPGN is worthwhile.

Examining the Exit Points of MPGN



The Friedman test shows that MPGN-S2 statistically outperforms MPGN-S1.

$$\chi_F^2 = 2.56 \qquad \alpha_{0.05} = 1.960$$

Noise in the Datasets

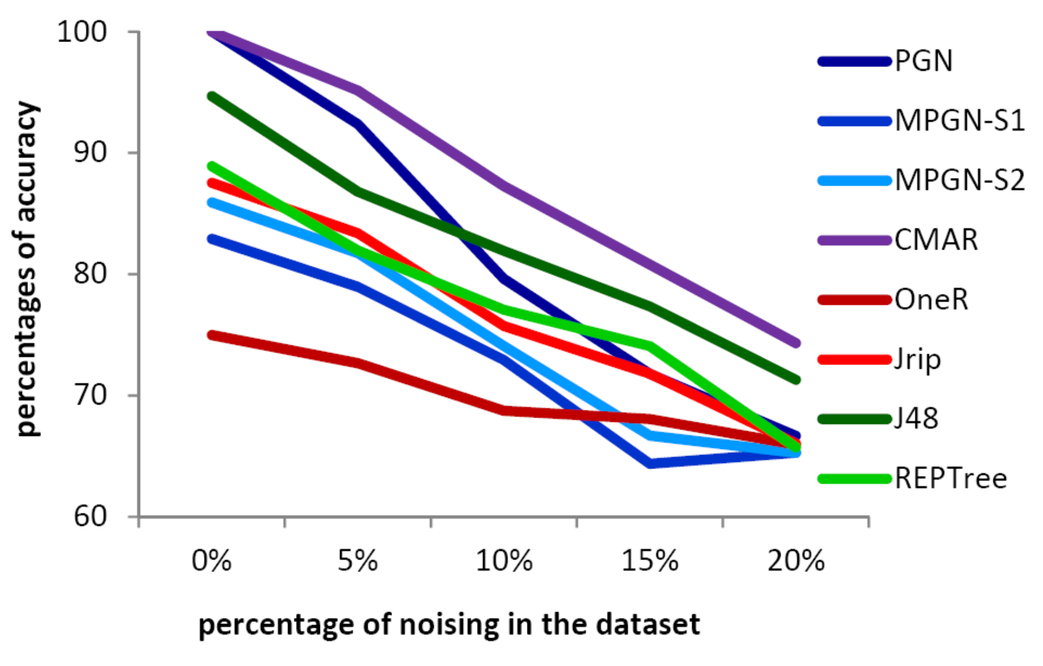
"Monks1" dataset (clear dataset with uniform class distribution)

- The noising of the datasets is made by choosing random instance and attribute and replacing the value with arbitrary chosen possible for this attribute values. The system keeps the information for the instances and position when such changes are already made and does not make repetitive changing of the same positions. Such replacing is made until a desired percentage of noising is achieved.
- Noising within attributes reflects to noising of class labels because of the appearance of contradictory instances. The table shows the resulting noise in class labels.

Percentage of noising in attributes	Resulting noise between class labels
0%	0.00 %
5%	6.00 %
10%	12.50 %
15%	17.25 %
20%	22.45 %

Noise in the Datasets

"Monks1" dataset



- The best performing method in this experiment is CMAR
- Also J48 is very stable
- The PGN and MPGN are most sensitive to noise

(confirms our hypothesis that confidence-prioritising approach has its disadvantages in noising datasets)

Comparison with Other Classifiers

Datasets	PGN	MPGN-S1	MPGN-S2	CMAR	OneR	JRip	J48	REPTree
audiology	75.50	69.00	69.00	59.18	47.00	69.50	72.00	62.50
balance_scale	77.89	81.41	83.49	86.70	60.10	71.95	66.18	67.15
breast_cancer_wo	96.43	92.85	93.56	93.85	91.85	93.28	94.28	93.99
car	92.59	82.87	85.71	81.77	70.03	86.75	90.80	88.20
cmc	49.90	46.03	46.64	53.16	47.25	50.38	51.60	50.17
credit	87.54	85.65	86.09	87.10	85.51	85.07	85.36	85.07
haberman	55.27	73.21	73.21	71.90	72.88	73.21	73.21	74.20
hayes-roth	81.94	65.22	67.49	83.42	50.77	78.12	68.23	73.53
hepatitis	80.65	81.94	81.94	84.52	81.94	77.42	79.36	79.36
lenses	74.00	83.00	83.00	88.00	62.00	83.00	83.00	80.00
monks1	100.00	82.9	85.92	100.00	74.98	87.53	94.68	88.91
monks2	73.06	80.52	81.02	59.74	65.73	58.73	59.90	63.90
monks3	98.56	90.43	93.50	98.92	79.97	98.92	98.92	98.92
post-operative	66.67	52.22	52.22	51.11	68.89	70.00	71.11	71.11
soybean	93.15	84.00	84.00	78.48	37.44	85.35	87.64	78.18
tae	52.94	52.88	52.88	35.74	45.76	34.43	46.97	40.43
tic_tac_toe	88.93	96.13	95.62	98.75	69.93	98.02	84.23	80.37
wine	96.09	92.19	93.87	91.70	78.63	90.45	87.03	88.16
winequality-red	64.98	59.35	59.60	56.29	55.54	53.72	58.22	57.03
zoo	98.10	89.24	89.24	94.19	73.29	88.19	94.14	82.19

Comparison with Other Classifiers

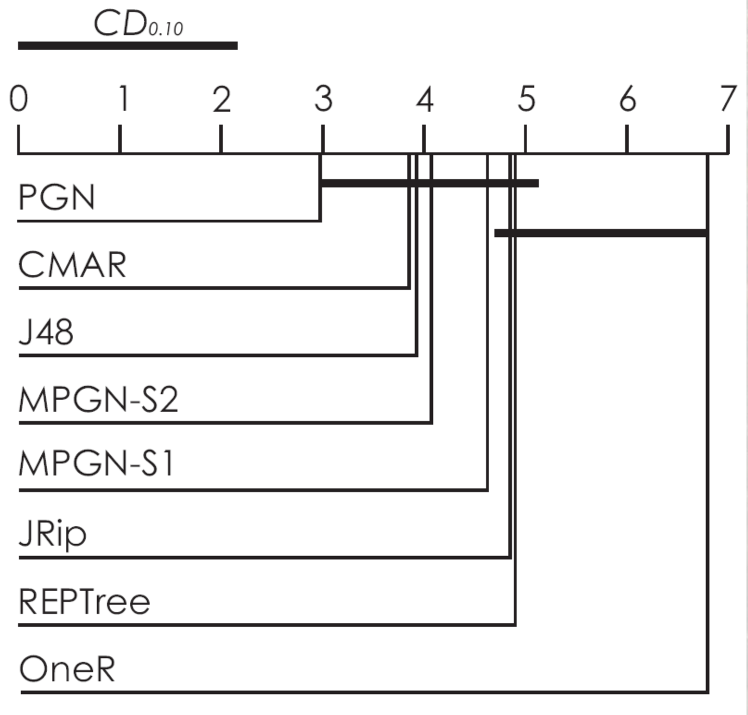
Friedman test:

- the number of datasets: n=20
- the number of classifiers: k=8
- the null hypothesis critical values is $\alpha_{0.10}$=12.017
- in our case: χ^2=29.492

-> the classifiers are statistically different

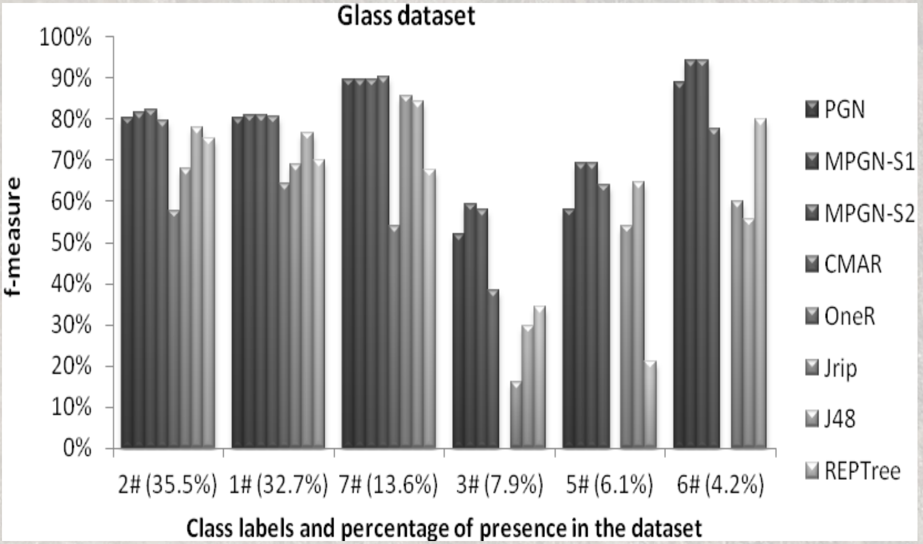
Nemenyi test:

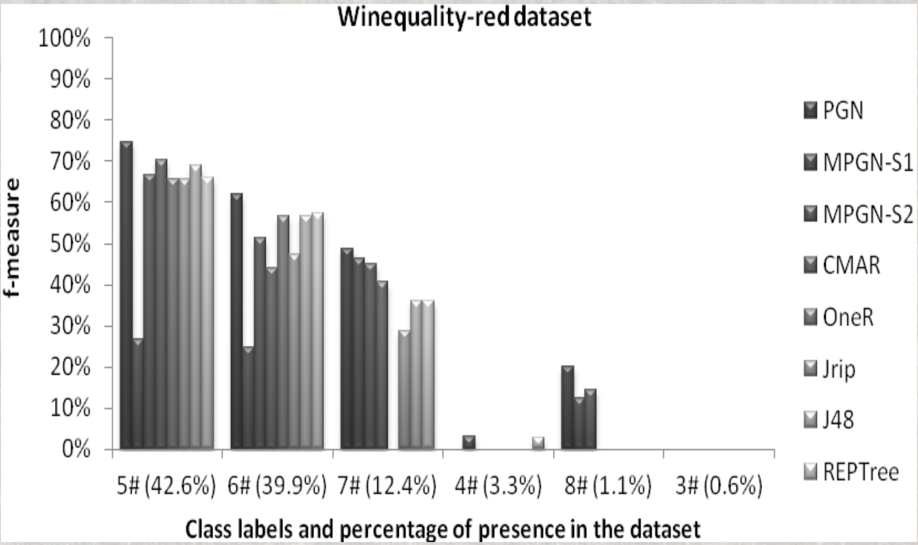
Critical distance $CD_{0.10}$=2.153



- PGN has best overall performance between examined classifiers
- MPGN-S2 is very close to CMAR and J48
- The first five classifiers (PGN, CMAR, J48, MPGN-S2 and MPGN-S1) significantly outperform OneR

Analyzing F-measures on Multi-class Datasets





- The analysis of F-measures for different datasets with multiple classes and non-uniform distribution showed that PGN and MPGN have not only good recognition accuracy for the chosen datasets, but also they recognize small classes controversy to the other classifiers.

Conclusions …

Main contributions can be summarized as:

- a new CAR-classifier PGN that questions the common approach to prioritize the support over the confidence and focuses on confidence first by retaining only 100% confidence rules has been elaborated;
- a method for effective building and storing of pattern set in multi-layer structure MPGN during the process of associative rule mining using the possibilities of multi-dimensional numbered information spaces has been developed;
- software of proposed algorithms and structures has been implemented in the frame of data mining environment system PaGaNe;
- the conducted experiments prove the vividness of proposed approaches showing the good performance of PGN and MPGN in comparison with other classifiers from CAR, rules and trees, and especially in the case of multi-class datasets with uneven distribution of the class labels.

… and Plans for Future Work

Possible directions for further research are:

- implementing PGN pruning and recognition ideas over pyramidal structures of MPGN;
- proposing different techniques for rule quality measure taking into account confidence of the rule in order to overcome the process of rejecting one rule preferring other one, rarely observed in the dataset;
- testing the possibilities of MPGN using Exit point 1 recognition in the field of campaign management;
- applying the established algorithms PGN and MPGN in different application areas such as business intelligence or global monitoring.

Thank you for your attention !