Subgroup Discovery in Defect Prediction

Rachel Harrison, Oxford Brookes University

Daniel Rodríguez, Univ of Alcala

José Riquelme, Univ of Seville

Roberto Ruiz, Pablo de Olavide University

Outline

Supervised Description

Subgroup Discovery

Preliminary Experimental Work

- Datasets
- Algorithms (SD and CN2-SD)
- Results

Conclusions and future work

Descriptive Models

Typically, ML algorithms have been divided into:

- Predictive (Classification, Regression, temporal series)
- Descriptive (Clustering, Association, summarisation)

Recently, ***supervised descriptive rule discovery*** is being introduced in the literature.

- The aim is to understand the underlying phenomena, not to classify new instances, i.e., to find information about a specific value in the class attribute.
- The information should be useful to the domain expert and easily interpretable.
- Types of supervised descriptive techniques include:
  - Contrast Set Mining (CSM)
  - Emerging Pattern Mining (EPM)
  - Subgroup discovery (SD)

SD – Definition

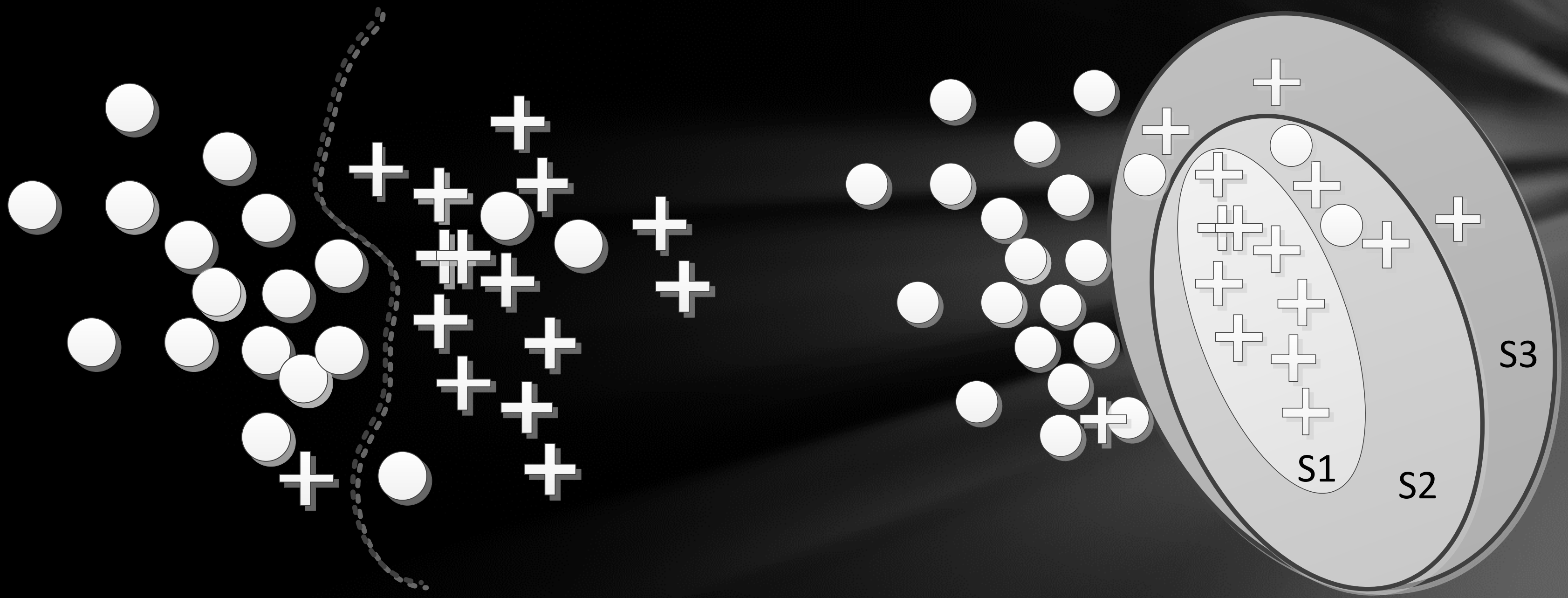
SD algorithms aims to find subgroups of data that are statistically different given a property of interest. [Klösgen, 96; Wrobel, 97]

- SD lies between predictive (finding rules given historical data and a property of interest) and descriptive tasks (discovering interesting patterns in data).
- SD algorithms generally extract rules subsets of the data of previously specified the concept, for example defective modules from a software metrics repository.
- Rules have also the "***`Condition → Class`***" where the condition is the conjunction of a set of selected variables (pairs attribute–value) among all variables.
  - Advantages of rules include that are well known representation easily understandable by the domain experts
- So far, SD has majoritarily been applied to the medical domain.

SD vs. Classification

	Classification	*Subgroup Discovery*
Induction	Predictive	Descriptive
Output	Set of classification rules (dependent rules)	Individual Rules to describe subgroups (independent rules)
Purpose	To learn a model for classification or prediction	To find interesting and interpretable patterns with respect to a specific attribute

SD vs. Classification



Following [Herrera et al, 2011]

SD Algorithms

SD algorithms could be classified as:

- Exhaustive (e.g.: SD-map, Apriori-SD)
- Heuristic (e.g.: SD, CN2-SD)
  - Fuzzy genetic algorithms (SDIGA, MESDIF, EDER-SD)

Or from their origin, evolved from different communities:

- Extension of classification algorithms (SD, CN2-SD, etc.)
- Extension of association algorithms (Apriori-SD, SD4TS, SD-Map, etc.)

Comprehensive survey by [Herrera et al. 2011]

Quality Measures in SD

Measures of Complexity

- Number of rules: It measures the number of induced rules.
- Number of conditions: It measures the number of conditions in the antecedent of the rule.

Measures of Generality

- Coverage: $Cov(R) = \frac{n(Cond)}{N}$

  where N is the number of samples and $n(Cond)$ is the no. of instances that satisfy the antecedent of the rule.

- Support: $Sup(R) = \frac{n(Cond \cdot Class)}{N}$

  where $n(Cond \cdot Class)$ is the no. of instances that satisfy both the condition and the class

Quality Measures in SD

Measures of precision

- Confidence: $Conf(R) = \frac{n(Cond \cdot Class)}{n(Cond)}$

- Precision Q_c : $Q_c = n(Class \cdot Cond) - c\, n(\neg Class \cdot Cond)$

- Precision Q_g : $Q_g = \frac{n(Class \cdot Cond)}{n(\neg Class \cdot Cond) + g}$

Measures of interest

- Significance: $Sig(R) = 2 \sum_{k=1}^{n} n(cond \cdot Class_k) \cdot log \frac{n(Cond \cdot Class_k)}{n(Class_k) \cdot p(Cond)}$

Other Measures

Sensitivity: $$Sens(R) = TPr = \frac{TP}{Pos} = \frac{n(Class \cdot Cond)}{n(Class)}$$

False alarm: $$FA(R) = FPr = \frac{FP}{Neg} = \frac{n(\neg \overline{Class} Class \cdot Cond)}{n(\neg Class)}$$

Specificity: $$Spec(R) = \frac{TN}{TN + FP} = \frac{TN}{Neg} = \frac{n(\neg Class \cdot \neg Cond)}{n(\neg Class)}$$

Unusualness: $$WRAcc\ (R) = \frac{n(Cond)}{N} = \left(\frac{n(Class \cdot Cond)}{n(Cond)} - \frac{n(Class)}{N}\right)$$

Experimental Work – Datasets

NASA Datasets

- Originally available from:
  - `http://mdp.ivv.nasa.gov/`
- From PROMISE, using the ARFF format (Weka – data mining toolkit):
  - `http://promisedata.org/`
    - Boetticher, T. Menzies, T. Ostrand, Promise Repository of Empirical Software Engineering Data, 2007.

Bug prediction dataset

- `http://bug.inf.usi.ch/`
  - D'Ambros, M., Lanza, M., Robbes, Romain, Empirical Software Engineering (EMSE), In press, 2011

Datasets Characteristics

Some of these datasets are highly unbalanced, with duplicates and contradictory instances, and irrelevant attributes for defect prediction.

	# inst	*Non-def*	*Def*	*% Def*	*Lang*
CM1	498	449	49	9.83	C
KC1	2,109	1,783	326	15.45	C++
KC2	522	415	107	20.49	C++
KC3	458	415	43	9.39	Java
MC2	161	109	52	32.29	C++
MW1	434	403	31	7.14	C++
PC1	1,109	1,032	77	6.94	C
Eclipse JDT Core	997	791	206	20.66	Java
Eclipse PDE-UI	1,497	1,288	209	13.96	Java
Equinox	324	195	129	39.81	Java
Lucene	691	627	64	9.26	Java
Mylyn	1,862	1,617	245	13.15	Java

Metrics Used from the Datasets

For the NASA datasets:

	Metric	*Definition*
McCabe	loc	McCabe's Lines of code
	v(g)	Cyclomatic complexity
	ev(g)	Essential complexity
	iv(g)	Design complexity
Halstead	uniqOp	Unique operators, n_1
	uniqOpnd	Unique operands, n_2
	totalOp	Total operators, N_1
	totalOpnd	Total operands N_2
Branch	branchCount	No. branches of the flow graph
Class	defective?	Reported defects? (true/false)

For the OO datasets:

	Metric	*Definition*
C&K	wmc	Weighted Method Count
	dit	Depth of Inheritance Tree
	cbo	Coupling Between Objects
	noc	No. of Children
	lcom	Lack of Cohesion in Methods
	rfc	Response For Class
Class	defective?	Reported defects?

Algorithms

The algorithms used:

- The **Subgroup Discovery** algorithm (SD) [Gamberger, 02] is a covering rule induction algorithm that using beam search aims to find rules that maximise:

$$q_g = \frac{TP}{FP+g}$$

  where *TP* and *FP* are the number of true and false positives respectively and *g* is a generalisation parameter that allow us to control the specificity of a rule, i.e., balance between the complexity of a rule and its accuracy.

- The **CN2-SD** [Lavrac, 04] algorithm is an adaptation of the CN2 classification rule algorithm [Clark, 89]. It induces subgroups in the form of rules using as a quality measure the relation between true positives and false positives. The original algorithm consists of a search procedure using beam search within a control procedure and the control procedure that iteratively performs the search.
  - The CN2-SD algorithm uses *Weighted Relative Accuracy* (explained next) as a covering measure of the quality of the induced rules.

Tool:

- Orange: `http://orange.biolab.si/`

Examples Rules – KC2 Dataset

	#	pd	pf	TP	FP	Rules
SD	0	.24	0	26	0	ev(g) > 4 ^ totalOpnd > 117
	1	.28	.01	30	5	iv(G) > 8 ^ uniqOpnd > 34 ^ ev(g) > 4
	2	.27	.01	29	5	loc > 100 ^ uniqOpnd > 34 ^ ev(g) > 4
	3	.27	.01	29	5	loc > 100 ^ iv(G) > 8 ^ ev(g) > 4
	4	.27	.01	29	5	loc > 100 ^ iv(G) > 8 ^ totalOpnd > 117
	5	.24	.01	26	5	iv(G) > 8 ^ uniqOp > 11 ^ totalOp > 80
	6	.24	.01	26	5	iv(G) > 8 ^ uniqOpnd > 34
	7	.23	.01	25	5	totalOpnd > 117
	8	.31	.01	34	5	loc > 100 ^ iv(G) > 8
	9	.29	.01	32	5	ev(g) > 4 ^ iv(G) > 8
	10	.29	.01	32	5	ev(g) > 4 ^ uniqOpnd > 34
	11	.28	.01	30	5	loc > 100 ^ ev(g) > 4
	12	.28	.01	30	5	iv(G) > 8 ^ uniqOp > 11
	13	.35	.01	38	5	ev(g) > 4 ^ totalOp > 80 ^ v(g) > 6 ^ uniqOp > 11
	14	.27	.01	29	5	iv(G) > 8 ^ totalOp > 80
	15	.27	.01	29	5	ev(g) > 4 ^ totalOp > 80 ^ uniqOp > 11
	16	.26	.01	28	5	ev(g) > 4 ^ totalOp > 80 ^ v(g) > 6
	17	.26	.01	28	5	loc > 100 ^ uniqOpnd > 34
	18	.31	.01	34	5	ev(g) > 4 ^ totalOp > 80
	19	.31	.01	34	5	iv(G) > 8
CN2-SD	0	.35	.01	38	5	uniqOpnd > 34 ^ ev(g) > 4
	1	.4	.02	43	9	totalOp > 80 ^ ev(g) > 4
	2	.78	.21	84	88	uniqOP>11

Example Rules – JDT Core Dataset

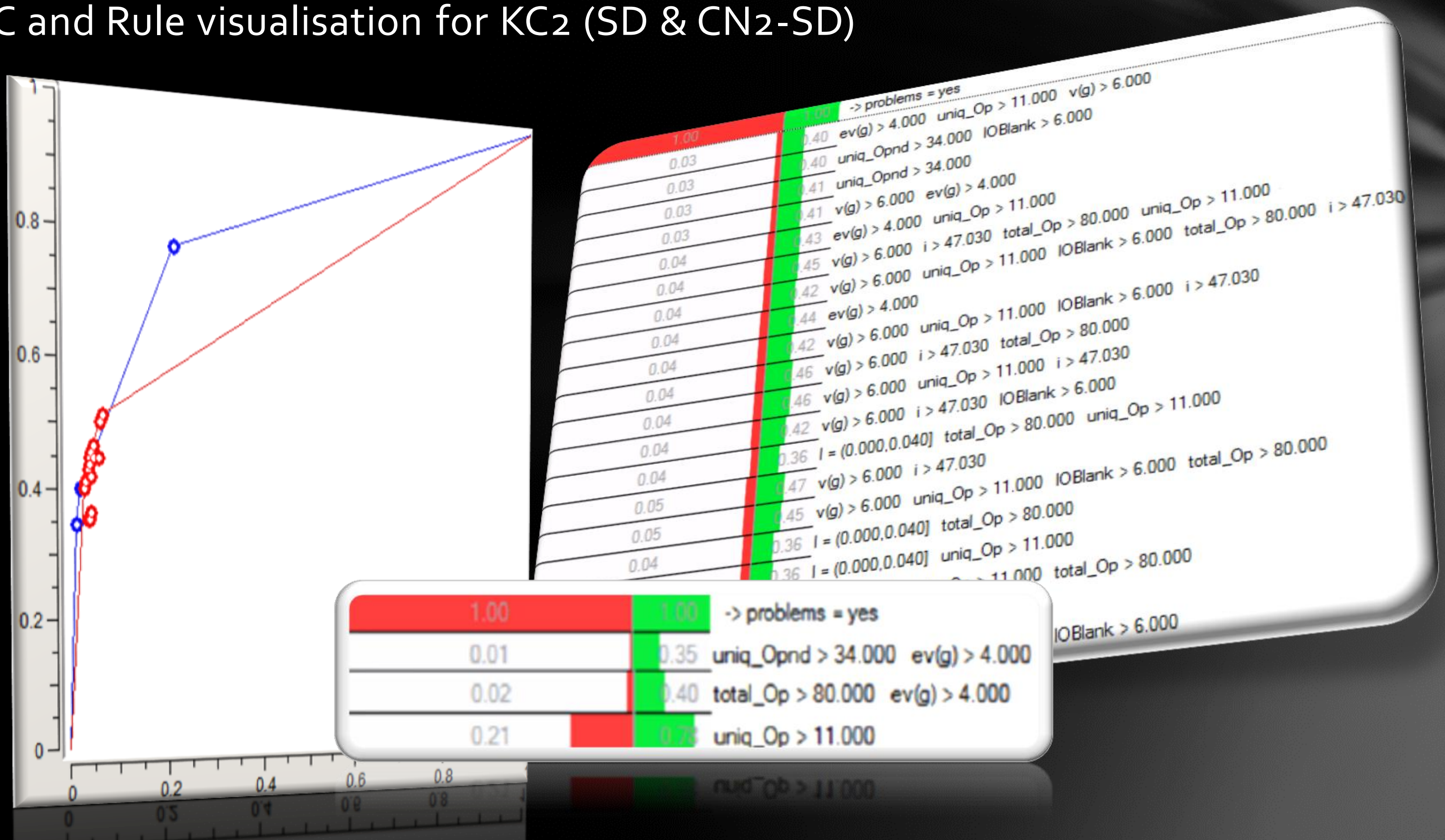
	#	pd	pf	TP	FP	Rules
SD	0	.27	.02	56	16	lcom > 171 ∧ rfc > 88 ∧ cbo > 16 ∧ wmc > 141
	1	.3	.02	62	16	rfc > 88 ∧ wmc > 141 cbo > 16
	2	.3	.02	62	16	cbo > 16 ∧ wmc > 141
	3	.29	.02	60	16	lcom > 171 ∧ rfc > 88 ∧ wmc > 141
	4	.29	.02	60	16	lcom > 171 ∧ wmc > 141
	5	.33	.03	68	24	rfc > 88 ∧ wmc > 141
	6	.32	.03	66	24	rfc > 88 ∧ wmc > 141 ∧ dit≤ 5
	7	.33	.03	68	24	wmc > 141
	8	.32	.03	66	24	dit <= 5 ∧ wmc > 141
	9	.18	.02	38	16	wmc > 141 ∧ noc = 0 ∧ dit≤ 5
	10	.19	.02	40	16	wmc > 141 ∧ noc = 0
	11	.18	.02	38	16	cbo > 16 ∧ rfc > 88 ∧ noc > 0 dit≤ 5
	12	.42	.04	87	32	cbo > 16 ∧ rfc > 88 ∧ dit≤ 5
	13	.3	.03	62	24	lcom > 171 ∧ rfc > 88 ∧ cbo > 16 ∧ dit≤ 5
	14	.2	.02	42	16	cbo > 16 ∧ rfc > 88 ∧ noc > 0
	15	.24	.03	50	24	cbo > 16 ∧ rfc > 88 ∧ noc = 0 ∧ dit≤ 5
	16	.45	.05	93	40	cbo > 16 ∧ rfc > 88
	17	.32	.03	66	24	lcom > 171 ∧ rfc > 88 ∧ cbo > 16
	18	.25	.03	52	24	cbo > 16 ∧ rfc > 88 ∧ noc = 0
	19	.33	.05	68	40	cbo > 16 ∧ lcom > 171
CN2-SD	0	.45	.05	93	40	rfc > 88 ∧ cbo > 16
	1	.55	.09	114	72	rfc > 88

Cross-validation Results (10 CV)

		Cov	Sup	Size	Cplx	Sig	RAcc	Acc	AUC
SD	CM1	.233	.72	20	3.045	4.548	.029	.602	.748
	KC1	.079	.426	20	2.61	16.266	.023	.61	.657
	KC2	.085	.533	20	2.185	9.581	.049	.703	.74
	KC3	.294	.91	20	2.435	5.651	.037	.608	.83
	MC2	.161	.647	20	2.055	2.204	.042	.643	.689
	MW1	.071	.5	20	2.515	3.767	.02	.736	.678
	PC1	.118	.37	20	3.515	3.697	.01	.66	.621
CN2-SD	CM1	.113	.64	5	1.3	2.972	.023	.628	.617
	KC1	.107	.607	5	1.1	2.912	.03	.634	.71
	KC2	.156	.795	5	1.6	11.787	.065	.733	.816
	KC3	.126	.885	4.9	1.295	3.146	.019	.68	.797
	MC2	.152	.427	5	2.32	2.186	.04	.593	.593
	MW1	.079	.558	5	2.02	3.517	.02	.661	.743
	PC1	.087	.661	5	1.86	2.814	.007	.632	.688
SD	JDT Core	.082	.539	20	2.485	13.774	.039	.662	.726
	PDE UI	.11	.407	20	3.94	1.936	.023	.603	.642
	Equinox	.269	.899	20	2.08	4.577	.054	.62	.759
	Lucene	.106	.579	20	2.295	4.368	.017	.741	.696
	Mylyn	.104	.425	20	2.9	12.631	.021	.675	.633
CN2-SD	JDT Core	.121	.543	5	1.58	18.961	.055	.613	.732
	PDE-UI	.144	.593	3.7	2.89	1.106	.023	.575	.684
	Equinox	.166	.797	5	1.020	3.772	.043	.636	.712
	Lucene	.070	.405	5	2.2	4.378	.016	.584	.653
	Mylyn	.081	.376	4.5	2.818	11.062	.018	.555	.632

Visualisation of SD

ROC and Rule visualisation for KC2 (SD & CN2-SD)



Conclusions

Rules obtained using SD are intuitive but needed to be analysed by an expert.

The metrics used for classifiers cannot be directly applied in SD and need to be adapted.

Current and future work

- Further validation and application in other software engineering domains, e.g., project management.
- SD is a search problem!
  - Development of new algorithms and metrics
    - EDER-SD (Evolutionary Decision Rules SD) in Weka
    - Unbalanced data (ROC, AUC metrics?), etc.
    - Feature Selection (as a pre-processing step, part of the algorithm?, which metrics really influence defects)
    - Discretisation
  - Different search strategies and fitness functions (and multi-objective!)
    - Use of global optimisation (set of metrics) vs. local metrics (individual metrics)

References

Kralj, P., Lavrac, N., Webb GI (2009) Supervised Descriptive Rule Discovery: A Unifying Survey of Constrast Set, Emerging Pateern and Subgroup Mining. Journal of Machine Learning Research 10: 377–403

Kloesgen, W. (1996), Explora: A Multipattern and Multistrategy Discovery Assistant. In: Advances in Knowledge Discovery and Data Mining, American Association for Artificial Intelligence, pp 249–271

Wrobel, S. (1997), An Algorithm for Multi-relational Discovery of Subgroups. Proceedings of the 1st European Symposium on Principles of Data Mining and Knowledge Discovery, Springer, LNAI, vol 1263, pp 78–87

Bay S., Pazzani, M. (2001) Detecting Group Differences: Mining Contrast Sets. Data Mining and Knowledge Discovery 5: 213–246

Dong, G., Li, J. (1999) Efficient Mining of Emerging Patterns: Discovering Trends and Differences. In: Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, pp 43–52

Herrera, F., Carmona del Jesus, C.J., Gonzalez, P., and del Jesus, M.J., An overview on subgroup discovery: Foundations and applications, Knowledge and Information Systems, 2011 – In Press.

Gamberger, D., Lavrac, N.: Expert-guided subgroup discovery: methodology and application. Journal of Artificial Intelligence Research 17 (2002) 501–527

Lavrac, N., Kavsek, B., Flach, P., Todorovski, L.: Subgroup discovery with CN2-SD. The Journal of Machine Learning Research 5 (2004) 153–188

Clark, P., Niblett, T. (1989) , The CN2 induction algorithm, Machine Learning 3 261–283