Globalization of Local Models with SVMs

Stefan Rüping[1]

Fraunhofer IAIS, Schloss Birlinghoven, 53754 St. Augustin, Germany
stefan.rueping@iais.fraunhofer.de,
WWW home page: http://www.stefan-rueping.de

Abstract. Local models are high-quality models of small regions of the input space of a learning problem. The advantage of local models is that they are often much more interesting and understandable to the domain expert, as they can concisely describe single aspects of the data instead of describing everything at once as global models do. This results in better plausibility and reliability of models.
While it is relatively easy to find an accurate global and interesting local model independently, the combination of both goals is a challenging task, which is hard to solve by current methods. This paper presents an SVM-based approach that integrates multiple local classification models into one global model, such that the resulting global model is both accurate and adequately reflects the local patterns on which it is based.

1 Introduction

Local models are high-quality models of small regions of the input space of a learning problem. The advantage of local models is that they are often much more interesting and understandable to the domain expert, as they can concisely describe single aspects of the data instead of describing everything at once as global models do. Concentrating on single, local aspects facilitates the inspection of models by human experts, which very soon gets too complex when multiple correlations and real-world effects are mixed [1].

Interpretability is a critical goal in a knowledge discovery process [2], as an understandable model allows to assess information encoded in the model, in particular the reliability of the learner's predictions, far better than by simple statistical error measure and also enables the user to discover modeling errors like biased sampling that are otherwise hard to detect. In addition, in real-world applications often a black-box prediction of a class label alone is not of very much interest, for example in the analysis of customer churn. Predicting whether or not a single customer will leave is nice, but even more important is understanding the reasons why customers are leaving, such that adequate reactions can be performed, e.g. targeting a marketing campaign.

The goal of this paper is to integrate local models into the framework of Support Vector Machines [3], where a set of local models is used to describe interesting patterns in the data, and an approach inspired by the SVM is used to integrate all local models into a single prediction. We make no assumption

on the nature of the local models, such that any learner that returns a set of partially defined classification functions can be used in this scheme.

The approach presented here allows local models to build an approximative understandable classifier, while the more adaptive SVM classifier improves accuracy on those parts of the input space that are hard to model. This idea is depicted in Figure 1. In addition, this approach allows to identify regions of the input space, where relevant information is missed by the local patterns, but can be identified by a more complex numerical classifier.

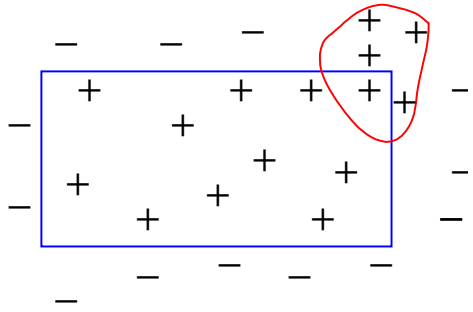

Fig. 1. Simple pattern (rectangle) and complex correction of the pattern (nonlinear area)

Obviously, this approach is only meaningful if the local models and the combined (SVM plus local models) learner are closely correlated; we want to avoid the pathological situation where the local models are completely wrong and are overruled by the SVM on any single example. Hence, it is necessary to integrate the following constraint into the learning process

τ-constraint: $P(f_{LM} \neq f_{combined}) \leq \tau$ for a given constant $\tau \in [0, 1]$

where $f_{LM}(x)$ is the function obtained by simply averaging all applicable local models on x (it may be advisable to generate local models that are pairwise disjoint, such that f_{LM} simply selects the applicable local model, if such exists, and returns a default value otherwise). It will be shown later that the proposed approach can indeed find a model which can be used for all possible values of τ simultaneously.

Additionally, the new method will identify points which are not adequately modeled by the existing local models, such that it can be used to target a follow-up iteration of the local model learner to interesting regions of the input space.

The paper is organized as follows: the next section gives a short introduction to the paradigm of local patterns on which the work in this paper is built. Section 3 introduces the new learner, which is empirically evaluated in Section 4. Section 5 concludes.

2 Detecting Local Patterns

Experience shows that large, global patterns in a model are often not interesting to a user because they are easy to detect and well known to domain experts. Patterns are more informative if they contradict what is already known [4]. Indeed, one can often with very different methods find a simple model which provides not an optimal solution, but a reasonably good approximation. The hard work usually lies in improving an already good model. An interesting alternative is to let a domain expert explicitly define what he already knows and then detect patterns that map out where his believes are wrong.

To deal with this situation, the field of local pattern detection [5, 6] has come into attention. Local pattern detection is defined as the unsupervised search for local accumulations of data points with unexpected high density with respect to some background model [7]. In short, the idea can be summarized in the intuitive equation

$$Data = Global\ Model + Local\ Model + Noise.$$

Please note that the cases described by the local model are not to be confused with indescribable outliers or over-fitted patterns. The local models are always assumed to be carefully validated and to describe useful patterns in the data.

Given a set of local models, the question in a classification setting is now how to combine the local models back into a single prediction, i.e. how to globalize them. It is possible to approach this problem by using ensemble classifiers. However, in these approaches usually a more or less complicated rule to combine the predictions of the individual classifiers into a single prediction. Multiple classifiers are a most successful approach in terms of accuracy, but the interpretability of theirs models is very limited. The problem is that one has to understand each base model plus the combination strategy in order to understand the complete model. Further, one cannot interpret one of the base models on its own to gain insight into the model, but always has to check all interactions between the base models.

One could also think of approaching globalization by the seemingly more simple idea of separating the descriptive and predictive tasks by using the local models only to present the dependencies in the data to the user, while the class prediction of unseen examples is performed by an independent, high-accuracy model. The problem here is that one cannot guarantee to the user that what he understood about the data from the local patterns actually matches the way in which the predictions are generated by the predictive model.

The goal of this paper is to globalize the local patterns while assuring that there is a strict, well-defined correspondence between the set of independent local models and the globalized model. The philosophy behind this approach is that accuracy and interpretability are two aspects of the same problem and that their solutions should be as independent as necessary, but as close as possible.

Several approaches to solve the problem of discovering and selecting local models have been investigated, e.g. in [8] and [9]. The particular question of globalization of local models has been addressed in [10], where attribute weights

are used to give a global insight into the classification process. Perhaps the closest work to the one presented in this paper is [11], where trees of SVMs and clustering is used to define a tree of local models.

3 Local Model SVMs

The critical part in globalization of local models is to guarantee that the combined model adequately reflects the single local models. In particular, the problem is to integrate the τ-constraint with the goal of minimum error, which a standard learner is not designed to do. This section will tackle this problem by defining a new SVM-type classifier that integrates the information of a global model, such that it combines the excellent performance of Support Vector Machines with a minimal disagreement from the set of local models.

Several approaches have been discussed in the literature in order to combine the excellent understandability of a logical representation with a better generalization performance of numerical models. For example, [12] proposes to optimize first order rules on numerical domains by using a gradient descend method to optimize the discretization that is necessary to transform numerical data into first-order representation. First-order rules are used as features for a numerical learner in [13] by constructing a bit vector of the outputs of each rule. This approach does not increase understandability, as the combination of the features is completely unintelligible, but is well suited to tackle noise in the input data. Similar approaches can also be found in [14] and [15].

In this paper, the local models will be expressed in propositional logic form and will be learned by the JRip algorithm [16], which is a variant of RIPPER [17], a well-known propositional logic rule set learner. However, the new algorithm will not be based on any specific features of this learner, such that any learner that returns a set of rules can be used. In particular, the rules can also be given by a human expert. In fact, we will only need to assume that we are given the local models in terms of prediction functions $p_i : X \to \{-1, 1\}$. Note that in this paper we are not interested in optimizing the set of local models, e.g. by finding a minimally redundant set; instead we are only interested in finding a globalization of these models, and in particular mapping out the regions that are not adequately modeled. Hence, in the following we will assume that there exists a single local model function $p : X \to \{-1, 1\}$ that aggregates all local models in a trivial way, e.g. by averaging, and seek to optimize this function.

Technically, the new method is based on Basis Pursuit [18], which is a method for the sparse approximation of a real function f in terms of a set of basis functions g_i. There is no restriction on the form of the basis functions. The idea is to generate a set of test points x_i and find a linear combination $\sum_i \alpha_i g_i$ of the basis functions that approximates the target function on this test points as closely as possible. To achieve sparseness, the 1-norm $||\alpha||_1 = \sum_i |\alpha_i|$ of the parameter vector is added as a complexity term similar as in Support Vector Machines. This norm is chosen because it will give much sparser result than the 2-norm, but is computationally more tractable than the otherwise preferable

0-norm $||\alpha||_0 = \#\{i|\alpha_i \neq 0\}$. The criterion of sparseness will guarantee that only the most informative basis functions will be chosen in the final function approximation.

The new idea is to apply the same approach to a classification framework with Support Vector Machines. Standard Support Vector Machines already achieve good generalization performance by combining basis function $g_i(x) = K(x_i, x)$ given by the kernel centered at the training points x_i. When the prediction $p(x)$ of a local model (or a set thereof) is added as another basis function, it should already give a good prediction of the labels and hence be a very informative feature, such that most of the other features can be ignored.

Remember that the standard SVM optimization task is given by

$$\begin{aligned} ||w||^2 + C\sum_{i=1}^{n}\xi_i \;\; &\rightarrow \;\; \min \\ &\text{w.r.t.} \\ &\forall_{i=1}^{n}\; y_i f(x_i) \geq 1 - \xi_i \\ &\forall_{i=1}^{n}\; \xi_i \geq 0 \end{aligned}$$

where the decision function f has the form

$$f(x) = \sum_i \alpha_i K(x_i, x) + b.$$

To integrate the local model predictions $p(x)$ into the SVM and to achieve focus on the local errors, the form of the decision function is now changed to

$$\begin{aligned} f(x) \;&=\; p(x) + \sum_i \alpha_i K(x_i, x) \\ &=: p(x) + f_{num}(x) \end{aligned}$$

where it is assumed that $p(x) \in \{-1, 1\}$. That is, we try to learn a kernel-based function f_{num} that explains the difference between the local models captured in $p(x)$ and the optimal decision function $f(x)$. This SVM formulation will be called the Local Model SVM. The change in the decision function has two effects. First, for any example (x_i, y_i) that is correctly classified by p it holds that

$$\begin{aligned} & y_i f(x_i) \geq 1 - \xi_i \\ \Leftrightarrow\;\; & y_i(p(x_i) + f_{num}(x_i)) \geq 1 - \xi_i \\ \Leftrightarrow\;\; & y_i p(x_i) + y_i f_{num}(x_i) \geq 1 - \xi_i \\ \Leftrightarrow\;\; & y_i f_{num}(x_i) \geq -\xi_i \end{aligned}$$

which is automatically fulfilled when $f_{num}(x_i) = 0$ and hence for the examples correctly predicted by p the optimal solution is $w = 0$. This relaxation of the constraints on f from $|yf(x)| \geq 1$ to $|yf(x)| \geq 0$ automatically leads to a drastic reduction in the number of support vectors on the already correctly classified examples, which will be the majority of the examples for any reasonable p.

Additionally, the constant b has been removed from the decision function. This has a special effect for locally concentrated basis functions such as the radial basis kernel function

$$f_{\gamma,x_i}(x) = e^{-\gamma||x_i - x||^2}.$$

Any nonzero term b would require that in order to meet $f_{num}(x_i) \geq 0$ there need to be basis functions with nonzero α if there are examples such that $by_i < 0$. Hence, setting $b = 0$ leads to a sparse decision function on the region correctly predicted by p. It follows further that only errors of p lead to a value of $f_{num}(x)$ that is significantly different from 0 and hence the absolute value of f_{num} may give an indication of the error probability of p.

Using the standard technique of Lagrangian multipliers it can easily be seen that the dual formulation of the new SVM problem is given by

$$W(\alpha) = \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \sum_{i=1}^{n} (1 - y_i p(x_i))\alpha_i \rightarrow \min$$
$$\text{subject to: } \forall_{i=1}^{n} 0 \leq \alpha_i \leq C$$

In particular, the removal of b leads to the removal of the linear constraint $\sum y_i \alpha_i = 0$ and hence to a simpler optimization problem.

Figure 2 shows an example data set for the local model SVM. The one-dimensional data set is given by the points at $y = \pm 1$, while the prediction of the logical model p – in this case a simple threshold rule – is plotted in between. Figure 3 additionally shows the prediction of the local model SVM, more precisely the function $f_{num}(x)$. The training points where the prediction of the logical rule $p(x)$ is false are marked as crosses. Notice that f_{num} drops to zero outside the error region of p and that for all points $y_i(p(x_i) + f_{num}(x_i)) \geq 1$ holds, which implies that the local model SVM fulfills the margin property of the standard SVM, which is imperative for good generalization performance. However, it must be noticed that the generalization property of the local model SVM, in particular the susceptibility to overfitting, is directly dependent on the quality of the set of local models as expressed by p. A set of completely overfit local model will lead to a zero modeling error and hence to a overfit combined model. This risk can for example be reduced by training p and the local model SVM on different data sets. The main idea of this paper, however, is that the best guard against overfitting is allowing the user to check the local models in p to get some additional evidence on the plausibility of the results from outside the actual data set.

In order to meet the τ-constraint the local decision function

$$f(x) = p(x) + f_{num}(x)$$

can easily be replaced by

$$f_\lambda(x) = p(x) + \lambda f_{num}(x)$$

where $0 \leq \lambda \leq 1$ and λ is reduced until the disagreement rate between f_λ and p drop below τ. As f_{num} reaches its maximum for points mispredicted by p, this new function will still be maximally correct.

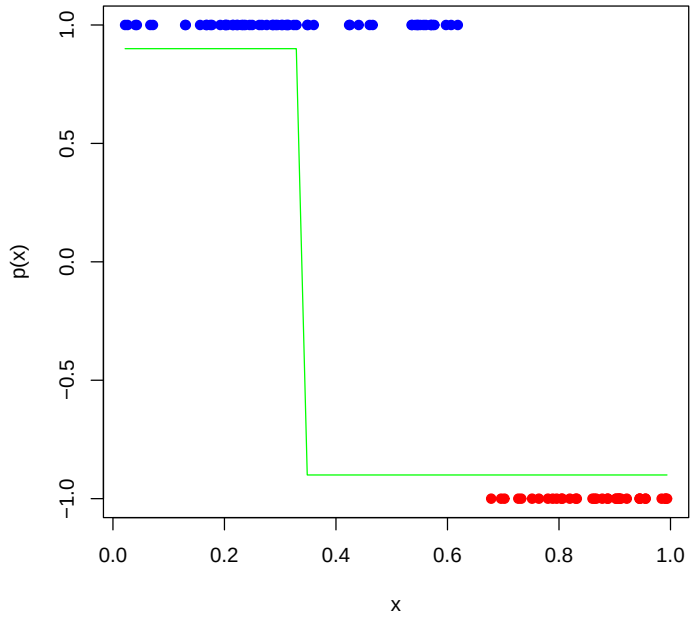

Fig. 2. Data for Local Model SVM

Assuming that higher absolute values of f_{num} are indicative of a higher probability of an error of p, one can use probabilistic calibration to transform $|f_{num}|$ into an estimator of the error probability of p. In this paper Isotonic Regression [19] is used because it does not need any assumptions about the distribution of the errors except the assumption of a monotonic dependency.

4 Experiments

The following experiments compare the local model SVM against two baseline techniques. As the local model SVM is principally a linear combination of the local models in p and a SVM-type classifier, the most direct solution would be to learn a linear model on the outputs of p and a standard SVM. This approach is an instance of Stacking [20]. It is straight-forward to modify the stacked classifier

$$s(x) = Ap(x) + Bf_{svm}(x)$$

into a classifier

$$s_\lambda(x) = Ap(x) + \lambda Bf_{svm}(x)$$

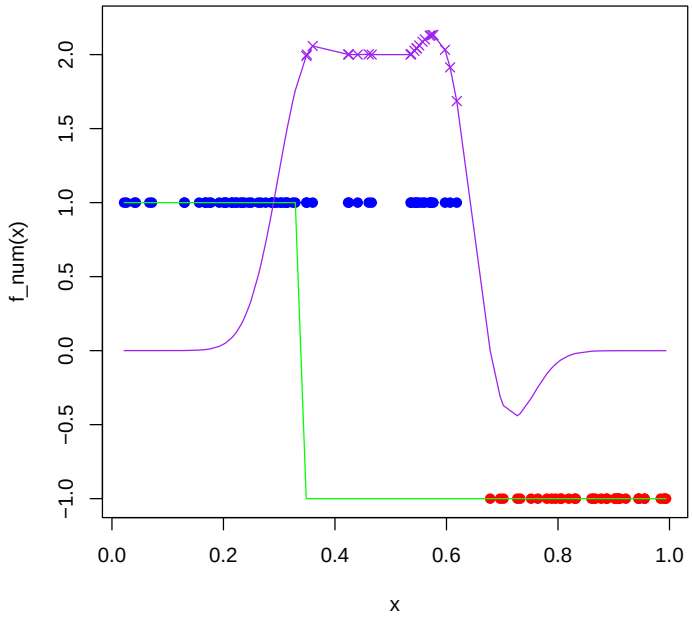

Fig. 3. Prediction of Local Model SVM

that adheres to the τ-constraint.

On the other hand, the stacked classifier does not incorporate any local information present in the classification errors of p. This can be changed by removing all examples that are correctly classified by p from the SVMs training set. With this modified SVM classifier $f_{svm,rm}$ one defines the stacked classifier

$$r_\lambda(x) = Ap(x) + \lambda B f_{svm,rm}(x)$$

accordingly to s_λ.

The following table compare the errors of the local model SVM, stacking and reduced stacking on a set of UCI data sets [21] plus some additional real-world data sets (business, medicine, insurance and garageband). All results reported in this paper have been obtained using 10-fold cross-validation.

The JRip classifier was used to a set of local models p. In order to simulate incomplete information in the local models, the most specific rules in each rule set were removed, such that on the average only half of the rules found by JRip were used (as the more general rules cover more examples, still most of the examples were classified by their original JRip rules). The local threshold was

set to $\tau = 0.1$. The significance values for the error differences at levels $\alpha = 0.05$ and $\alpha = 0.01$ are also given.

Name	LMSVM	Stacking	$L \leq S$	Red. Stacking	$L \leq R$
business	0.216	0.255	o	0.293	$++$
covtype	0.241	0.264	$+$	0.274	$++$
diabetes	0.245	0.253	o	0.27	$++$
digits	0.027	0.003	$--$	0.027	o
physics	0.409	0.417	$+$	0.417	$+$
ionosphere	0.137	0.105	$-$	0.174	$++$
liver	0.327	0.292	$-$	0.33	o
medicine	0.248	0.247	o	0.254	o
mushroom	0.482	0.482	o	0.482	o
promoters	0.372	0.391	o	0.391	o
insurance	0.03	0.068	$++$	0.07	$++$
balance	0.247	0.219	o	0.257	o
dermatology	0.317	0.273	o	0.317	o
iris	0.013	0.02	o	0.013	o
voting	0.233	0.207	$-$	0.233	o
wine	0.14	0.175	o	0.14	o
breast	0.078	0.086	o	0.099	o
garageband	0.298	0.316	$+$	0.337	$++$

Comparing the local model SVM with stacking we can see that both perform significantly better than the other on 4 data sets, so it is safe to say that both perform equally well. Indeed, a Wilcoxon signed rank test finds no significant difference between both approaches. In comparison to reduced stacking, the local model SVM performs signifcantly better on 7 data sets and is never worse. In this case, a Wilcoxon signed rank test confirms that LMSVM performs significantly better than reduced stacking with $p = 0.030$.

We have seen that LMSVM and Stacking perform equally well in terms of classification accuracy, but what about the goal of identifying local models? To investigate whether the local classifier f_{num} holds information about the errors of p, we calibrate it into a probabilistic classifier. In the following table, the mean squared error (Brier score) between the predicted probability and the actual occurrence of an error is given.

Name	LMSVM	Stacking	$L \leq S$	Red. Stacking	$L \leq R$
business	0.187	0.228	+	0.236	++
covtype	0.196	0.202	*o*	0.226	++
diabetes	0.193	0.196	*o*	0.202	*o*
digits	0.027	0.027	*o*	0.027	*o*
physics	0.227	0.263	++	0.257	++
ionosphere	0.123	0.143	+	0.155	++
liver	0.226	0.223	*o*	0.233	*o*
medicine	0.187	0.186	*o*	0.187	*o*
mushroom	0.086	0.336	++	0.383	++
promoters	0.34	0.275	−−	0.303	*o*
insurance	0.014	0.034	++	0.033	++
balance	0.186	0.214	+	0.218	*o*
dermatology	0.021	0.194	++	0.195	++
iris	0.016	0.014	−	0.014	−
voting	0.049	0.162	+	0.159	++
wine	0.129	0.124	*o*	0.137	*o*
breast	0.078	0.111	++	0.08	*o*
garageband	0.236	0.241	*o*	0.284	++

We can see that the local model SVM has a significantly lower mean squared error than the other two methods on 9 of the data sets and is significantly worse on only 2 data sets for stacking and only 1 for reduced stacking. A Wilcoxon signed rank test confirms that LMSVM has significantly lower error than Stacking with a p-level of 0.010 and has significantly lower error than Reduced Stacking with a p-level of 0.002. This shows that for the local model SVM $|f_{num}|$ carries much information about the error probability of p. This means in particular that the local model SVM will perform well for any value of τ, because selecting a value of λ is equivalent to cutting of the influence of the points with lowest values of $|f_{num}|$, which are least likely to be mispredicted by p.

To validate whether the local model SVM also leads to a sparser solution, the following table compares the number of Support Vectors for the standard SVM (as used in stacking) and the local model SVM.

Name	SV Local	SV All	$L \leq A$
business	65.5	92.2	++
covtype	618.8	746.5	++
diabetes	349.9	388.1	++
digits	42.9	146.9	++
physics	717.4	801.6	++
ionosphere	110.8	119.3	o
liver	194.3	252.6	++
medicine	592.6	577.8	o
mushroom	999.9	999.4	−
promoters	78.7	95.4	+
insurance	174.8	178.3	o
balance	117.5	125.9	+
dermatology	100.1	116.9	o
iris	83.5	12.1	−
voting	134.4	181.9	+
wine	35.7	128.9	++
breast	68.9	86.8	++
garageband	734.8	807.2	++

We can see that on 12 of the data sets, the local model SVM returns significantly less Support Vectors than the standard SVM. It should also be noted that the kernel parameter γ of both SVMs was selected to optimize the classification performance of the standard SVM. An optimization of γ for the local model SVM is likely to achieve even better results.

5 Conclusions

Globalization of local models is an important tasks that bridges the gap between the understandability and interestingness of local models, and the construction of a highly performant global classifier, which necessarily needs to combine and integrate multiple, possibly redundant or contradicting local models. In this paper, such an approach was presented which both effectively combines local models into a single prediction, follows the local models as closely as necessary using the $\tau-$constraint, and can be used to identify regions of the input space in need of additional local models.

In conclusion, the Local Model SVM perform equally well as the standard approach of Stacking, but gives far superior information about the local models errors compared to the competing solutions and is hence very well suited for extracting sparse local models and local patterns at the same time without any loss of accuracy.

Future work will concentrate on identifying the individual contributions of each local model more closely, instead of aggregating all local models into a single function p as in this approach. This will hopefully provide some valuable feedback to the local model learning algorithm.

Acknowledgments The financial support of the European Commission under the project RACWeB is gratefully acknowledged.

References

1. Miller, G.: The magical number seven, plus or minus two: Some limits to our capacity for processing information. Psychol Rev **63** (1956) 81 – 97
2. Giraud-Carrier, C.: Beyond predictive accuracy: What? In: ECML'98 Workshop Notes - Upgrading Learning to the Meta-Level: Model Selection and Data Transformation, Technical University of Chemnitz (1998) 78–85
3. Vapnik, V.: Statistical Learning Theory. Wiley, Chichester, GB (1998)
4. Guyon, I., Matic, N., Vapnik, V.: Discovering informative patterns and data cleaning. In Fayyad, U.M., Piatetsky-Shapiro, G., Smyth, P., Uthurusamy, R., eds.: Advances in Knowledge Discovery and Data Mining. AAAI Press/The MIT Press, Menlo Park, California (1996) 181–204
5. Hand, D., Adams, N., Bolton, R., eds.: Pattern Detection and Discovery. Springer (2002)
6. Morik, K., Boulicaut, J.F., Siebes, A., eds.: Local Pattern Discovery. Lecture Notes in Computer Science. Springer (2005)
7. Hand, D.: Pattern detection and discovery. In Hand, D., Adams, N., Bolton, R., eds.: Pattern Detection and Discovery. Springer (2002)
8. Knobbe, A., Ho, E.: Pattern teams. In: Proc. PKDD 2006. (2006) 577–584
9. van Leeuwen, M., Vreeken, J., Siebes, A.: Compression picks item sets that matter. In: Proc. PKDD 2006. (2006) 585–592
10. Subianto, M., Siebes, A.: Understanding discrete classifiers with a case study in gene prediction. In: Proc. 7th IEEE International Conference on Data Mining (ICDM 2007),. (2007) 661–666
11. Pang, S., Kasabov, N.: Inductive vs transductive inference, global vs local models: Svm, tsvm, and svmt for gene expression classification problems. In: Proc. 2004 IEEE International Joint Conference on Neural Networks. Volume 2. (2004) 1197–1202
12. Botta, M., Piola, R.: Refining numerical constants in first order logic theories. Machine Learning Journal **38** (2000) 109–131
13. Kijsirikil, B., Sinthupinyo, S.: Approximate ilp rules by backpropagation neural network: A result on thai character recognition. In Dzeroski, S., Flach, P., eds.: ILP-99. Volume 1634 of LNAI., Springer (1999) 162–173
14. Basilio, R., Zaverucha, G., Barbosa, V.C.: Learning logic programs with neural networks. In Rouveirol, C., Sebag, M., eds.: ILP 2001. Volume 2157 of LNAI., Springer (2001) 15–26
15. Popescul, A., Ungar, L.H., Lawrence, S., Pennock, D.M.: Towards structural logistic regression: Combining relational and statistical learning. In: Proceedings of the Workshop on Multi-Relational Data Mining (MRDM-2002) at KDD-2002, Edmonton, Canada (2002) 130–141
16. Witten, I., Frank, E.: Data Mining – Practical Machine Learning Tools and Techniques with Java Implementations. Morgan Kaufmann (2000)
17. Cohen, W.: Fast effective rule induction. In: Proccedings of the Twelfth International Conference on Machine Le arning, Tahoe City, CA, Morgan Kaufmann (1995) 115–123

18. Chen, S.S., Donoho, D.L., Saunders, M.L.: Atomic decomposition by basis pursuit. SIAM Journal of Scientific Computing **20**(1) (1998) 33–61
19. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining. (2002) 694–699
20. Wolpert, D.: Stacked generalizations. Neural Networks **5** (1992) 241–259
21. Murphy, P.M., Aha, D.W.: UCI repository of machine learning databases (1994)