XCS and GALE: a Comparative Study of Two Learning Classifier Systems with Six Other Learning Algorithms on Classification Tasks

Ester Bernadó, Xavier Llorà, and Josep M. Garrell
Research Group in Intelligent Systems. Computer Science Dept.
Enginyeria i Arquitectura La Salle. Ramon Llull University.
Psg. Bonanova 8, 08022, Barcelona, Catalonia, Spain.
{esterb,xevil,josepmg}@salleURL.edu

Abstract

This paper compares the learning performance, in terms of prediction accuracy, of two genetic-based machine learning systems (GBML), XCS and GALE, with six well-known learning algorithms, coming from instance based learning, decision tree induction, rule-learning, statistical modeling and support vector machines. The experiments, performed on several datasets, show the suitability of the genetic-based learning classifier systems for classification tasks. Both XCS and GALE significantly achieved better results than IB1 and Naive Bayes. Besides, any method could not outperform XCS and GALE significantly.

1 INTRODUCTION

The purpose of this paper is twofold: to evaluate the performance of two different GBML algorithms in their application to data mining, focusing on classification tasks, and their comparison with other non-evolutionary learning schemes. We chose one representative GBML algorithm for each of the two main approaches: the Michigan approach (or Classifier Systems, CS) and the Pittsburgh approach (or Learning Systems, LS). XCS (Wilson, 1995) is a CS that represents the current state-of-the-art in the *Michigan* approach, whereas GALE (Llorà & Garrell, 2000a) is a knowledge-independent parallel GA that summarizes the characteristics of LS, exploiting fine-grained parallelism. We compare these GBML systems with six non-evolutionary classifier schemes, coming from different disciplines like instance-based learning, rule and decision-tree induction, statistical modeling and support vector machines. The algorithms are compared on several datasets, from both artificial and real-world domains. The real-world domains used in this paper come from our own repository (Martínez & Santamaría, 1996; Martí et al., 1998) and from the UCI repository (Blake & Merz, 1998).

This paper is organized as follows. Section 2 and 3 gives a short description of XCS and GALE. Next, we present our experimental framework and the results we achieved. Finally, section 5 summarizes the conclusions and further work.

2 XCS

XCS (Wilson, 1995) is a classifier system which differs from traditional classifier systems in two main aspects: a) the definition of fitness, which is based on the accuracy of the payoff prediction rather than on the prediction itself and b) the application of the GA to the environmental niches defined by the action sets. For more details, please see (Wilson, 1995; Wilson, 1998; Butz & Wilson, 2000).

XCS has shown a strong tendency to evolve consistent and complete knowledge representations that, moreover, tend to be minimal because of the generalization bias (see the generalization hypothesis (Wilson, 1995)). This makes XCS particularly interesting for data mining tasks (Wilson, 2000; Saxon & Barry, 2000).

Several types of rule representations are reported for XCS: binary (Wilson, 1995), enumeration encoding (Saxon & Barry, 2000), real ranges (Wilson, 1999) and integer ranges (Wilson, 2000). We have introduced the ability of handling with mixed attributes. Thus, the condition part of a rule is a conjunction of tests, where each test t_i is codified depending on the type of the corresponding attribute. The genetic operators are adapted accordingly.

3 GALE

GALE is a classification scheme based on fine-grained parallel genetic algorithms. It was introduced in (Llorà & Garrell, 2000b), being designed for solving classification tasks. Its learning scheme is based on the Pittsburgh approach, where individuals represent complete solutions to the classification problem. Its main contribution is its knowledge-independent model; it can evolve indistinctly rules, instances, partially-defined instances, and decision trees (orthogonal, oblique, and multivariate based on nearest neighbor).

GALE uses a fine-grained parallel algorithm based on spreading the population over a 2D grid. Each cell on the grid, which contains up to one individual, is connected to the surrounding cells defining demes, where evolution occurs locally. This architecture can be parallelized massively, reducing thus the computation learning time. Theoretical expressions show that the time complexity is independent of the population size. For further details, see (Llorà & Garrell, 2000a; Llorà & Garrell, 2001a; Llorà & Garrell, 2001b).

4 EXPERIMENTS

We compare the performance, in terms of classification accuracy, of XCS and GALE with six traditional non-evolutionary classifier schemes. The comparison is performed with fifteen datasets, from both real-world and artificial domains.

4.1 CLASSIFIER SCHEMES

For these experiments, we use our own implementation of XCS in *C++*. Prior to the generation of these results, we tested our software with the results reported in previous papers and with those obtained using the XCSC code, provided by Barry in `http://www.csm.uwe.ac.uk/~ambarry/LCSWEB/computer.htm`. The implementation of GALE is a serial version coded in *Java*. Parallel implementations of GALE, and their analysis, are beyond the scope of this paper.

The non-evolutionary algorithms chosen for the comparison come from different learning schemes. The chosen algorithms are: a, b) instance-base learning, **IB1** and **IBk** with $k = 3$ (Aha & Kibler, 1991), c) statistical modeling, **Naive Bayes** (John & Langley, 1995), d) tree induction, **C4.5 revision 8** (Quinlan, 1993), e) rule learning, **PART** (Frank & Witten, 1998), and f) support vector machines, **SMO** (Platt, 1998). We also include **0-R**, a simple classifier scheme that predicts the majority class in the training data, introduced to establish a lower bound for the other learning schemes (Witten & Eibe, 2000). All these algorithms are obtained from the *Weka* package developed at the University of Waikato in New Zealand, available from the http address: *http://www.cs.waikato.ac.nz/ml/weka*. These algorithms are run with the default configuration.

4.2 DATASETS

Our experiments are performed on fifteen datasets, belonging to a variety of domains, having numeric and nominal attributes, with different number of classes and ranging over different dataset sizes.

Three of the datasets are generated artificially: `led`, mux11 (`mux`) and `tao`. `Led` (Blake & Merz, 1998) is the classification of seven binary attributes, representing seven light-emitting diodes, into a decimal digit. The examples are obtained with the addition of 10% noise. `Mux` is the multiplexer problem with eleven binary inputs. `Tao` is a dataset obtained by sampling the TAO figure with 1888 instances equally spaced along the horizontal and vertical axis. The resulting instances have two real valued attributes corresponding to the $\langle x, y \rangle$ coordinates, and the associated class which is the color of the figure on that point.

Twelve datasets belong to real-world domains. Ten of them come from the the UCI repository (Blake & Merz, 1998): breast-w (`bre`), bupa (`bpa`), `cmc`, glass (`gls`), heart-c-14 (`h-c`), heart-h-14 (`h-h`), iris (`irs`), pima-indians (`pmi`), vehicle (`veh`), and wine (`wne`). The biopsies (`bps`) (Martínez & Santamaría, 1996) and mammograms (`mmg`) (Martí et al., 1998) datasets belong to our own repository and consist of the prediction of breast cancer.

4.3 EXPERIMENTAL SETUP

In order to compare the performance of the different algorithms in terms of classification accuracy, we use the following methodology. Classification accuracy is estimated in two different ways, depending on the dataset size. For large datasets, we use the holdout estimate. In fact, this is only applied to the *led* problem, where 2000 instances were available in the training set and 4000 instances in the test set. The remaining datasets are run on a stratified ten-fold cross-validation test. To estimate the difference in performance between the algorithms, we use a test for the difference of two proportions, based on the approximation of the binomial distribution by a normal distribution, if the runs are based on holdout, and a paired t-test if the experiments are based on cross-validation (Dietterich, 1998).

Table 1: Prediction accuracy on all the datasets (average and standard deviation). Each GALE result also marks the used knowledge representation. A $\star$ stands for rule sets, a † for instance sets, $\oplus$ for orthogonal decision trees, $\otimes$ for oblique decision trees, and $\odot$ for multivariant decision trees.

DS	0-R	IB1	IBK	NaiveBayes	C4.5r8	PART	SMO	XCS	GALE
bps	51.6±0.6	83.2±3.2	82.8±4.3	78.6±5.5	80.1±4.8	79.0±3.3	86.4±3.0	83.2±3.1	$83.7\pm3.8^{\otimes}$
bre	65.5±1.1	96.0±1.5	96.7±1.4	96.0±2.3	95.4±1.6	95.3±2.2	96.7±1.7	96.4±2.5	$95.7\pm2.2^{\odot}$
bpa	58.0±1.4	63.5±6.6	60.6±6.6	54.3±2.8	65.8±6.9	65.8±10.0	58.0±1.4	65.4±6.9	$68.4\pm6.7^{\dagger}$
cmc	42.7±0.4	44.4±2.7	46.8±3.3	50.6±2.8	52.1±2.3	49.8±3.6	-	55.5±2.5	$50.3\pm5.1^{\dagger}$
gls	34.6±2.5	66.3±10.9	66.4±10.9	47.6±8.9	65.8±10.4	69.0±10.0	-	70.8±8.5	$65.6\pm11.9^{\oplus}$
h-c	54.5±2.2	77.4±7.6	83.2±5.2	83.6±6.0	73.6±8.8	77.9±6.4	-	80.3±7.8	$79.9\pm5.2^{\dagger}$
h-h	63.9±2.1	78.3±6.4	82.4±8.4	83.7±7.8	80.3±9.0	79.6±10.6	-	79.9±6.3	$78.2\pm7.2^{\dagger}$
irs	33.3±0.0	95.3±3.2	95.3±3.2	94.7±2.8	95.3±3.2	95.3±3.2	-	94.7±5.3	$98.7\pm2.8^{\otimes}$
led	10.5±0.0	62.4±0.0	75.0±0.0	74.9±0.0	74.9±0.0	75.1±0.0	-	74.5±0.0	$75.0\pm0.0^{\star}$
mmg	56.0±2.9	63.0±12.4	65.3±6.3	64.7±7.7	64.8±6.4	61.9±4.2	67.0±7.4	64.3±6.4	$71.3\pm5.9^{\oplus}$
mux	49.9±0.1	78.6±4.0	99.8±0.3	61.9±2.7	99.9±0.2	100.0±0.0	61.6±3.0	100.0±0.0	$100.0\pm0.0^{\star}$
pmi	65.1±1.0	70.3±3.4	73.9±5.3	75.4±6.8	73.1±5.2	72.6±5.0	76.7±4.6	75.4±4.7	$75.8\pm4.0^{\oplus}$
tao	49.8±0.2	96.1±1.2	96.0±1.4	80.8±1.8	95.1±2.0	93.6±2.8	83.6±2.3	89.9±1.3	$95.5\pm1.0^{\dagger}$
veh	25.1±0.5	69.4±5.3	69.7±5.9	46.2±5.7	73.6±5.3	72.6±4.6	-	73.0±4.4	$68.8\pm3.8^{\dagger}$
wne	39.8±4.5	95.6±5.0	96.8±4.5	97.8±2.9	94.6±6.6	92.9±6.1	-	95.1±6.8	$97.2\pm2.9^{\dagger}$
Avg	46.7	76.0	79.4	72.7	79.0	78.7	75.7	79.9	80.3

Additionally, a Wilcoxon signed rank test (Conover, 1971) is used on the average accuracies of each method.

As mentioned before, GALE can evolve either of five different knowledge representations (based on rule sets, instance sets and induction trees). This gives us the possibility of testing the best suited knowledge representation for each dataset, thus minimizing the error due to the language limitations (language-intrinsic error (Martin & Hirschberg, 1996)). Therefore, we conducted five different experiments on every dataset, each one with a different knowledge representation. From there, we can can analyze the performance of GALE as well as the most appropriate knowledge representation.

4.4 RESULTS

The results of each algorithm on all the datasets are listed in table 1. They give the percentage of correct classifications, averaged over the ten-fold cross-validation runs, along with the standard deviation. The results on the *led* problem show the percentage of correct classifications, measured on the test set of a holdout experiment. The average of each method over all the datasets is also given in the last row of the table. Statistical differences of XCS and GALE compared to the other non-evolutionary algorithms are shown in table 2. For each dataset, we show the result of a one-tailed t-test, except for *led* where a test for the difference of two proportions based on the normal approximation is used.

The results in table 1 show, under GALE learning scheme, that knowledge representations based on instances or decision trees were best suited for the majority of the datasets (13 of 15). Rule sets were only used in two datasets, `led` and `mux`, both with binary attributes. This is possibly due to the binary encoding used by GALE in its rules, which performs worse in real-valued attributes than other representations, based on instances or trees. A rule encoding based on hyper-rectangles, as XCS uses, could counterbalance this effect.

Some observations can be made from the results in table 2. In `cmc` and `mux` datasets, XCS outperforms significantly nearly all classifier schemes. XCS also shows a good performance on `bpa` and `veh` datasets, whereas the worst performance of XCS is obtained on `tao`. In this dataset, XCS is overcome by five methods, outperforming only three of them. On the contrary, GALE seems to be well suited for this particular dataset. This dataset represents the TAO figure, where boundaries between classes are non-linear. It was generated artificially, as explained in section 4.2, and no noise was added. The results of XCS may be due to the knowledge representation: XCS evolves rules, while GALE was run using instances. This fact, together with the results achieved by IB1 and IBK in `tao`, suggest that a knowledge representation based on instances is more appropriate for the `tao` dataset. Further efforts on the analysis of the final rule set obtained by XCS, as well as on the rule dynamics during learning should be done. On the other hand, GALE shows a significant outperformance in several datasets (e.g., `bpa`, `irs`, `mux` and `tao`), whereas there is not any dataset where it performs poorly.

From the summary rows of table 2, we can observe that both XCS and GALE have significantly higher prediction accuracy than IB1, Naive Bayes and SMO in several datasets, according to a t-test at 99% level.

Table 2: Significant tests of XCS and GALE compared to non-evolutionary learning schemes. Differences in accuracies are significant using a one-tailed test at $p = .05^{\bullet}, .01^{\bullet\bullet}, .005^{\bullet\bullet\bullet}$. A $\bullet$ means that XCS or GALE outperform significantly the compared algorithm, while $\circ$ means a significant degradation. Rows labeled as `b-w` list the number of improvements and degradations of XCS and GALE to the column being compared, at a certain significant level. The last row shows the Wilcoxon test confidence level. Positive values indicate that XCS or GALE are "better" than the scheme being compared, while negative values indicate that XCS or GALE are "worse."

	Comparison of XCS								Comparison of GALE							
DS	**0-R**	**IB1**	**IBk**	**NBa**	**C4.5**	**PART**	**SMO**	**GALE**	**0-R**	**IB1**	**IBk**	**NBa**	**C4.5**	**PART**	**SMO**	**XCS**
bps	•••			•••	••	•	○○		•••			•••	•	•••	○○○	
bre	•••								•••							
bpa	•••			•••			•••		•••		••	•••			•••	
cmc	•••	•••	•••	•••	•••	•••		••	•••	••	•					○○
gls	•••			•••					•••			•••				
h-c	•••		○		•				•••							
h-h	•••			○					•••			○				
irs	•••							○○○	•••	•	•	•••	••	••		•••
led	•••	•••							•••	•••						
mmg	•••							○	•••	•	•	•		•••		•
mux	•••	•••	•	•••			•••		•••	•••	•	•••			•••	
pmi	•••	•••							•••	•••			•	•		
tao	•••	○○○	○○○	•••	○○○	○○○	•••	○○○	•••			•••		•	•••	•••
veh	•••	••	•	•••				•	•••			•••	○	○		○
wne	•••								•••					•		
Average	46.7	76.0	79.4	72.7	79.0	78.7	75.7	80.3	46.7	76.0	79.4	72.7	79.0	78.7	75.7	79.9
b-w .05	15-0	5-1	3-2	7-1	3-1	2-1	3-1	2-3	15-0	6-0	5-0	8-1	3-1	6-1	3-1	3-2
b-w .01	15-0	5-1	1-1	7-0	2-1	1-1	3-1	1-2	15-0	4-0	1-0	7-0	1-0	3-0	3-1	2-1
b-w .005	15-0	4-1	1-1	7-0	1-1	1-1	3-0	0-2	15-0	3-0	0-0	7-0	0-0	2-0	3-1	2-0
Wilcoxon	99.5	99.0	54.0	96.5	81.2	96.9	75.0	-62.5	99.5	98.2	76.7	97.7	96.0	95.0	90.0	62.5

Besides, the results of the Wilcoxon test indicate that XCS and GALE algorithms have a significant improvement over IB1, Naive Bayes and PART, at a confidence level greater than 95%. These two statistical tests agree that the genetic-based approaches have higher accuracy than IB1 and Naive Bayes, for this kind of datasets. They also indicate that these GBML approaches are not outperformed significantly by the other methods, in terms of classification accuracy. Other interesting performance criteria as training time, size of solution set, explanatory capabilities, etc, are left for future work. In comparing the prediction accuracy of XCS with GALE, no significant differences were found (regarding both the *t*-test and the Wilcoxon signed rank test). Further efforts should be done on this direction, including other performance measures as well.

5 CONCLUSIONS

This paper has compared two different GBML approaches, XCS and GALE, with six well-known learning algorithms, in terms of prediction accuracy, on fifteen datasets. The results obtained by both XCS and GALE reach or even improve the performance of the non-evolutionary learning schemes. In particular, statistical tests indicate a significant improvement of XCS and GALE over IB1 and Naive Bayes, whereas we did not obtain a significant degradation of XCS and GALE compared to the other methods. An advantage of using GAs for learning might be its robustness across different domains and its independence on the evolved knowledge representation. In these experiments, GALE was run with rule sets, instance sets and decision trees, while XCS evolved rule sets, although other representations can be considered in further work. Further research will also be focused on extending this comparison to other representative datasets, including other performance measures such as training time, complexity and comprehensibility of the obtained solution.

Although XCS and GALE belong to different genetic-based machine learning approaches, they did not show significant differences in their prediction accuracy. In XCS, the individual members represent a partial solution (a single rule), whereas in GALE an individual is a complete solution (e.g., a set of rules). This seems to have no influence on the prediction accuracy, but may produce different rule sets, for example in terms of the number of produced rules. This could be another interesting area of research.

Appendix

To allow replication of our results we include the configuration parameters of XCS and GALE. XCS parameters are set as follows: $R = 1000 - 0$, β=0.2, α=0.1, ϵ_0=1, $\nu = 5$, χ=0.8, μ=.04, m=0.1, s_0=0.7, $P_{\#}$=0.33, θ_{GA}=50, θ_{del}=50, δ=0.1, θ_{mna}=# available actions, 100000 explore iterations, $doGASubsumption = true$, $doActionSetSubsumption = false$. The population size is N=5400 for `bre`, `bpa`, `cmc`, `h-c`, `h-h`, `irs`, `pmi`, `tao`, `wne`, N=13000 for `bps`, `gls`, `led` and `mmg` datasets, and N=800 for `mux`. Please, see (Butz & Wilson, 2000; Wilson, 1999) for notation. GALE parameters are: $grid\_size$=64×64, r=1, p_m=.4, p_s=.01, k_{sr}=-.25, and $iterations = 150$. See (Llorà & Garrell, 2001b) for notation.

Acknowledgements

The authors would like to thank Alwyn Barry and Erick Cantú-Paz for valuable comments. The authors acknowledge the support provided under grant numbers DOGC 30/12/1997, 1999FI-00719, Epson-RRA:1999, and FIS-00/0033-2 as well as the support of *Enginyeria i Arquitectura La Salle*. We are also grateful to Witten & Frank and Barry for providing their code online, and to all the people who donated the datasets.

References

Aha, D., & Kibler, D. (1991). Instance-based learning algorithms. *Machine Learning, Vol. 6*, 37–66.

Blake, C., & Merz, C. (1998). UCI Repository of machine learning databases, [http://www.ics.uci.edu/~mlearn/MLRepository.html]. University of California, Irvine, Dept. of Information and Computer Sciences.

Butz, M., & Wilson, S. (2000). *An algorithmic description of XCS.* IlliGAL Report (No. 2000017)). University of Illinois at Urbana-Champaign.

Conover, W. (1971). *Practical Nonparametric Statistics.* New York: John Wiley, pp.206-209, 383.

Dietterich, T. G. (1998). Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Computation, 10*, 1895–1924.

Frank, E., & Witten, I. H. (1998). Generating Accurate Rule Sets Without Global Optimization. *Machine Learning: Proceedings of the Fifteenth International Conference* (pp. 144–151). Morgan Kaufmann.

John, G. H., & Langley, P. (1995). Estimating Continuous Distributions in Bayesian Classifiers. *11th. Conference on Uncertainty in Artificial Intelligence* (pp. 338–345).

Llorà, X., & Garrell, J. M. (2000a). Automatic Classification and Artificial Life Models. *Proceedings of Learning00 Workshop.*

Llorà, X., & Garrell, J. M. (2000b). Evolving Hierarchical Agents using Cellular Genetic Algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference, (GECCO 2000)* (p. 868). Morgan Kaufmann.

Llorà, X., & Garrell, J. M. (2001a). Inducing Partially-Defined Instances with Evolutionary Algorithms. *Proceedings of the 18th International Conference on Machine Learning (ICML'2001).* To appear.

Llorà, X., & Garrell, J. M. (2001b). Knowledge-Independent Data Mining with Fine-Grained Parallel Evolutionary Algorithms. *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'2001).* To appear.

Martí, J., Cufí, X., & Regincós, J. (1998). Shape-based feature selection for microcalcification evaluation. *Proceedings of the SPIE Medical Imaging Conference on Image Processing* (p. 1215:1224).

Martin, J., & Hirschberg, D. (1996). *Small Sample Statistics for Classification Error Rates I : Error Rate Measurements.* Technical Report No. 96-21). Department of Information and Computer Science, University of California, Irvine.

Martínez, E., & Santamaría, E. (1996). Morphological Analysis of Mammary Biopsy Images. *8th Mediterranean Electrotechnical Conference on Industrial Applications in Power Systems, Computer Science and Telecommunications* (pp. 1067–1070).

Platt, J. (1998). Fast training of support vector machines using sequential minimal optimization. *Advances in Kernel Methods-Support Vector Learning.* Cambridge, MA:MIT Press.

Quinlan, R. (1993). *C4.5: Programs for Machine Learning.* Morgan Kaufmann Publishers.

Saxon, S., & Barry, A. (2000). XCS and the Monk's Problems. *Learning Classifier Systems: From Foundations to Applications* (pp. 223–242).

Wilson, S. W. (1995). Classifier Fitness Based on Accuracy. *Evolutionary Computation, 3*, 149–175.

Wilson, S. W. (1998). Generalization in the XCS Classifier System. *Genetic Programming: Proceedings of the Third Annual Conference.* San Francisco, CA: Morgan Kaufmann.

Wilson, S. W. (1999). Get Real! XCS with Continuous-Valued Inputs. *Festschrift in Honor of John H. Holland.* Center for the Study of Complex Systems, University of Michigan.

Wilson, S. W. (2000). Mining Oblique Data with XCS. *Third International Workshop on Learning Classifier Systems (IWLCS-2000).*

Witten, I. H., & Eibe, F. (2000). *Data Mining. Practical Machine Learning Tools and Techniques with Java Implementations.* Morgan Kaufmann.