Pareto inspired multi-objective rule fitness for noise-adaptive rule-based machine learning

Ryan J. Urbanowicz, Randal S. Olson, and Jason H. Moore

University of Pennsylvania, Institute for Biomedical Informatics
Perelman School of Medicine, Philadelphia, PA, USA
{ryanurb,olsonran,jhmoore}@upenn.edu
http://www.epistasis.org/

Abstract. Learning classifier systems (LCSs) are rule-based evolutionary algorithms uniquely suited to classification and data mining in complex, multi-factorial, and heterogeneous problems. The fitness of individual LCS rules is commonly based on accuracy, but this metric alone is not ideal for assessing global rule 'value' in noisy problem domains and thus impedes effective knowledge extraction. Multi-objective fitness functions are promising but rely on prior knowledge of how to weigh objective importance (typically unavailable in real world problems). The Pareto-front concept offers a multi-objective strategy that is agnostic to objective importance. We propose a Pareto-inspired multi-objective rule fitness (PIMORF) for LCS, and combine it with a complimentary rule-compaction approach (SRC). We implemented these strategies in ExSTraCS, a successful supervised LCS and evaluated performance over an array of complex simulated noisy and clean problems (i.e. genetic and multiplexer) that each concurrently model pure interaction effects and heterogeneity. While evaluation over multiple performance metrics yielded mixed results, this work represents an important first step towards efficiently learning complex problem spaces without the advantage of prior problem knowledge. Overall the results suggest that PIMORF paired with SRC improved rule set interpretability, particularly with regard to heterogeneous patterns.

Keywords: data mining, classifier systems, fitness evaluation, multi-objective optimization, machine learning

1 Introduction

Rule-based machine learning (RBML) algorithms learn a set of 'IF:THEN' association rules capturing piece-wise local patterns to map the problem. Learning classifier systems (LCS) are a well-studied type of RBML predominantly applied to supervised and reinforcement learning tasks [1]. LCSs evolve a set of rules that collectively comprise a solution/prediction model. This distributed solution varies from the standard machine learning paradigm of a single model solution, which has made LCS particularly well suited to complex, multifactorial, and heterogeneous problems such as the n-bit multiplexer machine learning benchmarks [2]. While most early LCS research has focused on reinforcement learning, supervised learning has become a major focus in recent years, particularly with regards to real-world applications [2–5]. One major area includes biomedical data mining and prediction. These types of problems are typically characterized as 'noisy', can include a large number of variables, and can involve complex underlying patterns of association such as epistatic interactions and heterogeneity. In 2015, [2] introduced ExSTraCS 2.0, a more scalable Michigan-style supervised LCS. This approach was able to detect and characterize epistatic and heterogeneous patterns in noisy simulated genetic data, and was the first algorithm to report solving the 135-bit multiplexer directly. However additional emphasis on accuracy in the fitness function was necessary to efficiently solve the set of multiplexer problems (i.e the ν parameter, which controls the influence of accuracy on fitness, was set to 10 rather than the default of 1). Having prior knowledge that these problems were 'clean' (i.e. the problem could be optimally solved with 100% prediction accuracy) was an important part of choosing an appropriate objective weight. In that case, accuracy was overemphasized as the only explicit objective. The same logic is true for being able to solve noisy problems. In [6, 2] it was found that having ν set above 1 reduced performance in noisy domains. This is because noisy problems can not be solved with 100% prediction accuracy, and 'optimal' rules for these problems will have an accuracy below 1. Overemphasizing accuracy in a noisy problem leads to dramatic over-fitting, and a loss of generalization, prediction accuracy, and interpretability.

Only a handful of studies have explored a multi-objective fitness functions in LCS. Implicit and explicit multi-objective learning approaches for Michigan and Pittsburgh-style LCS algorithms were reviewed in [7]. Multi-objective research in Pittsburgh-style LCSs has focused on balancing rule-set accuracy with parsimony [8, 9]. The MOLeCS algorithm was introduced as the first explicitly multi-objective Pittsburgh LCS [10], applying competing objectives of rule-accuracy and coverage, where coverage refers to the number of training instances that were matched, and thus 'covered' by the rule. MOLeCS was the first LCS to consider a Pareto-front based rule fitness. Two different Pareto-front approaches were proposed in [10] to determine rule fitness ranking each generation of the genetic algorithm. Each involved the formation of a non-dominated rule-fitness front from rules in the current population. The first strategy gave all rules on the front the same 'best' fitness score, and all beneath, the same lower fitness score. The second strategy gave all rules on the front the best set of overall scores but rules on the non-dominated front with the highest accuracy also had the highest fitness. These approaches are not applicable to Michigan-style LCSs, which perform online rather than batch learning. Seeking to improve performance in noisy problems, a weighted-sum approach to multi-objective fitness function for Michigan-style LCS rules was recently proposed in ExSTraCS 2.1 [11] to avoid the overfitting issues that persist even when ν was set to 1 as seen in ExSTraCS 2.0 [2]. This new fitness function improved the interpretability and power to automatically characterize underlying complex patterns in the evolved rule set without sacrificing accuracy [11]. However this approach relies on the assumption that the data is noisy. Case in point, ExSTraCS 2.1 was no longer able to solve clean multiplexer problems beyond the 20-bit version since accuracy was now being undervalued.

In this study we present preliminary results for a Pareto-inspired multi-objective rule fitness (PIMORF). Our goal was to see if we could implement a Pareto-based Michigan-style LCS and determine whether we could identify Pareto-front properties that could be used to switch the objective weighting in favor of accuracy (in clean problems), and coverage (in noisy ones), without the advantage of prior knowledge. Also, building off work in [12], we introduce a fast rule compaction strategy that takes advantage of the multi-objective fitness function to globally rank rules for efficient rule set reduction that preserves performance. This proposed PIMORF was implemented and tested within the ExSTraCS 2.1 algorithm and evaluated over the 6-bit to 135-bit multiplexer problems, as well as a spectrum of complex, noisy simulated genetic datasets concurrently modeling epistatic and heterogeneous patterns of association. We expect that this work will (1) demonstrate the feasibility of adapting the Pareto-front concept to the Michigan-style LCS architecture, (2) improve knowledge extraction, and (3) pave the way for other data-driven fitness function adaptations to encourage assumption-free automated machine learning and data mining.

2 Methods

In this section we briefly (1) introduce the ExSTraCS algorithm, (2) describe how the PIMORF is updated and applied, (3) describe our proposed rule compaction strategy, and (4) outline the evaluation strategy.

2.1 Algorithm

The ExSTraCS algorithm [2] is a Michigan-style LCS algorithm, that has been expanded and adapted to better suit the needs of real-world supervised learning problems wherein classification, prediction, data mining, and/or knowledge discovery is the goal. Most recently in version 2.1, it was expanded to include a multi-objective fitness function that utilized a balanced weighting for the accuracy and coverage objectives. The accuracy and coverage metrics used in the present study were calculated as described in [11]. In short, the accuracy objective is the accuracy above what would be expected by random chance (based on the ratio cases to controls), transformed with an exponential function so that accuracy improvement beyond random chance were highly valued, but less emphasis was being placed on achieving 100% accuracy. The coverage metric is a state-frequency adjusted measure of the proportion of instances correctly (i.e. accurately) covered by the given rule. For rules that have not yet seen all of the training instances (i.e. so called 'Not Epoch Complete' (NEC) rules), we extrapolate this proportion up to the expected correct coverage once all data has been observed. For a detailed description of the ExSTraCS algorithm see [2] and [11]. For comparison we also evaluate ExSTraCS 2.0.1.2, which employs the typical accuracy-based LCS fitness [2]. All implementations are available on *sourceforge.com* or by request.

2.2 PIMORF for LCS

The Pareto-front is part of the Pareto-optimization approach popularized for multi-objective learning in genetic algorithms [13]. Figure 1A illustrates compo-

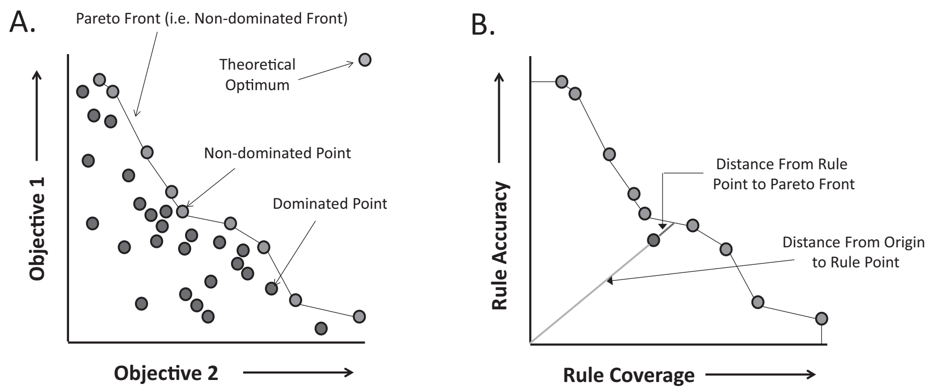

Fig. 1. Pareto front illustrations: (A) General representation of a 2-objective Pareto-front. (B) Application of the Pareto front concept to the calculation of rule-fitness in PIMORF.

nents of a general Pareto-front as it might be applied to any evolutionary modeling approach. Typically, a population of models are generated and objective performance is evaluated (often accuracy and parsimony). Each model appears as a point in the objective space (see Figure 1A). The 'front' (i.e. non-dominated front) is the set of all non-dominated points. A point is non-dominated if at least one of its objective values is the maximum observed given the value of the second objective. Next, the set of non-dominated points/models are chosen as the parents for the next generation of models, while dominated models can be discarded. Over multiple generations, the goal is to evolve the front closer to the theoretical optimum. The benefit of this approach is that evolution takes both objectives into account without making any assumptions about objective weighting (i.e. all points on the non-dominated front are treated with equal priority). Our Pareto adaptation to Michigan-style supervised LCS algorithms, (PIMORF) is differently designed to calculate rule-front-relative multi-objective rule fitness values. Instead of points representing models in the multi-objective front space, they represent LCS rules, that are each only part of the overall LCS 'model'. In PIMORF, the rule-fitness front is updated during the course of learning i.e. every time a new rule is generated and added to the rule-population, we check to see if the rule is non-dominated. If it is, the rule-fitness front is updated accordingly. The PIMORF rule-front constitutes the current standard for optimal multi-objective rule fitness, and rules that do not fall on the front (i.e. dominated rules) can be preserved since they might be important contributors to the overall solution despite not possessing a non-dominated combination of objective values. Implementation of PIMORF involves the following: (1) Scaling the 'correct coverage' objective using the maximum observed rule coverage [11]. (2) Learning and updating two separate rule-fitness fronts: one for 'epoch complete' (EC) rules that have been around long enough to have trained on the entire dataset, and another for NEC rules which have seen at least 1000 instances in the training set. To allow for a fair coverage comparison, NEC rule coverage values are extrapolated as described in [11] up to the total training set size. (3) In the first 1000 learning iterations, prior to either front being established, accuracy alone is applied as a surrogate for multi-objective fitness. (4) Rule-fitness is calculated as the relative distance between the origin (where accuracy and coverage objectives are both 0) and the rule point vs. the origin and the intercept point on the rule-fitness front (see Figure 1B). This is an agnostic approach to multi-objective fitness weighting since any rule on the front has the maximum fitness value. We also explored averaging this agnostic fitness value with a linear accuracy or coverage bias, to be applied in the case that we wanted to apply prior knowledge assuming a clean or noisy problem, or utilize characteristics of the rule-fitness front to detect this automatically. This PIMORF implementation, combining the relative parato distance with a coverage gradient bias will be referred to as ExSTraCS 2.1.1.

2.3 Rule Compaction

Rule compaction is a form of post-processing applied to the evolved LCS population following training. Its goal is to remove poor or redundant rules from the population and yield a more compact rule-set that is easier to interpret (i.e. extract knowledge), and ideally that preserves or improves power and predictive accuracy. In previous work, a variety of LCS rule compaction strategies were implemented and compared [12]. These strategies relied on an accuracy-based fitness function, and therefore has the drawback of being poor for globally ranking rules in the context of noisy problems. This is because highly accurate rules in the population consistently over-fit the training data. In this study, we introduce a simple rule compaction (SRC) scheme which we contrast with QRC, a rapid scheme from [12], that preserves or improves performance, but minimally reduces the overall rule-set size by removing clearly poor or inexperienced rules. SRC complements PIMORF which yields a more globally reliable rule-ranking metric than accuracy or rule-numerosity (i.e. the number of copies of a rule in the population). Numerosity had previously been applied as a rough estimator of global rule-value with mixed success[12]. SRC is implemented as follows: (1) Rank all rules in the population by PIMORF. (2) Progress through the rule set by descending PIMORF. (3) For each rule, identify and remove any instances in the training data that the rule correctly covers. If no remaining instances can be correctly covered, or the rule has an accuracy below the probability of randomly selecting the class specified by the rule, or the rule has not yet had the opportunity to train on the whole dataset (i.e. a NEC rule), this rule is excluded from the final rule-set. SRC stops once the training set is empty (i.e. it has been completely covered), or it has gone through the entire rule set.

2.4 Evaluation

In the present study we compare and evaluate ExSTraCS with and without the proposed PIMORF as well as compare QRF to SRC in the case where PIMORF is applied. Both implementations were run over the same set of 960 noisy (i.e. heritabilities of 0.1, 0.2, or 0.4), complex simulated genetic datasets with 20 discrete-valued attributes that were described and applied in [11] and generated

Table 1. Average performance over all 960 datasets.

Rule Population Performance After 200,000 Iterations									
Performance	ExSTraCS								
Statistics	v2.1	v2.0.2.1	p	v2.1.1	p	+QRF	p	+SRC	p
Training Accuracy	.7472	.7975	↑ **	.7519	↑ **	.7485	↓ *	.7648	↑ **
Test Accuracy	.6215	.6177	↓ **	.6123	↓ **	.6130	-	.6192	↑ *
Both Power	.4104	.4031	-	.3895	↓ **	.3901	-	0.3875	↓ *
Single Power	.7802	.7542	↓ **	.7635	↓ **	.7710	↑ *	.7740	↑ **
Both Co-Occur. Power	.2292	.0333	↓ **	.2656	↑ **	.2675	-	.3542	↑ **
Single Co-Occur. Power	.8271	.7688	↓ **	.8260	-	.8266	-	.8375	↑ **
Macro Population	1248.5	1351.5	↑ **	875.6	↓ **	810.2	↓ **	192.7	↓ **
Run Time (min)	52.57	50.56	↓ **	35.56	↓ **	35.61	-	35.58	-

− No significant change

* $p < 0.05$ (Direction of change given by arrows)

** $p < 6.94x10^{-4}$ (Cutoff assumes Bonferroni multiple test correction based on 72 comparisons)

using GAMETES [14]. Each dataset concurrently modeled patterns of epistasis and heterogeneity concurrently where four of the attributes were predictive and 16 were non-predictive. 20 replicates of each dataset were analyzed and 10-fold cross validation (CV) was employed to measure average testing accuracy and account for over-fitting. ExSTraCS was run up to 200,000 learning iterations. Pairwise statistical comparisons were made using the Wilcoxon signed-rank tests. All statistical evaluations were completed using R. Comparisons were considered to be significant at $p \leq 0.05$. All analyses were performed using 'Discovery', a 2400 core Linux cluster available to the Dartmouth College research community. These comparisons are performed over a set of key performance metrics [2]. Both accuracy metrics were calculated as a respective 'balanced accuracy' to account for imbalanced datasets as the default output of ExSTraCS. 'Both Power' is the ability to correctly identify both two-locus heterogeneous models. 'Single Power' is the ability to have found at least one. 'Both Co-occur. Power' indicates the ability to detect both correct heterogeneous patterns, while 'Single Co-occur. Power' is to detect at least one. Macro Population refers to the number of unique classifiers in the classifier population. Additionally we generated 18 toy simulated genetic datsets each with 20 attributes and 1600 training instances. These included datasets with either (1) a single locus linear model, (2) a two-locus XOR interaction model, or (3) a three-locus XOR interaction model each with varying degrees of noise (0-100%). Another 6 clean datasets with increasing sample sizes were generated for respective multiplexer benchmarks of (6-bit through 135-bit) [2]. This secondary analysis was designed to explore rule front properties that may serve as a 'switch' to automatically direct ExSTraCS to adopt an accuracy or coverage objective bias in a problem dependent manner.

3 Results and Discussion

Table 1 summarizes the statistical results comparing ExSTraCS with a multi-objective fitness function (v2.1) to ExSTraCS with a simpler accuracy based fitness (v2.0.2.1), as well as to our proposed implementation of PIMORF in

ExSTraCS (v2.1.1). This table further presents statistical comparisons between v2.1.1 following the application of QRF rule compaction, and differently with the application of the proposed SRC approach. As expected, preliminary testing applying SRC to ExSTraCS with accuracy-based fitness yielded a much smaller rule-set but with large performance losses (not shown). As can be reiterated from this table, a multi-objective fitness function (in v2.1) globally improved or maintained average performance measures when compared to accuracy based fitness (in v2.0.2.1) over a spectrum of noisy datasets. Closer inspection of these results, replicating findings in [11], suggest some data set specific trade offs for accuracy and power metrics, enforcing the suboptimality of a multi-objective fitness function with constant equal objective weights. With the substitution of PIMORF as the fitness metric in ExSTraCS (in v2.1.1), we do observe significant performance losses in testing accuracy, Both Power and Single Power, but on the other hand observe a significant increase in Both Co-Occurence Power, which reflects the ability of the algorithm to accurately detect and interpret both underlying heterogeneous models, a critical advantage of LCS algorithms in comparison to other machine learning approaches. Closer inspection of the v2.1.1 results yielded similar dataset specific trade offs in performance, suggesting that when averaged over all datasets this new implementation was not ideal in terms of some key performance metrics, but universal performance metric improvements could be expected if the dataset could be paired to the proper objective weights. Furthermore, v2.1.1 significantly and dramatically reduced the macro-population size (i.e. number of unique rules in the final population), and significantly reduced algorithm run time. While PIMORF performance is not yet optimal without proper objective weighting, the results are promising and support the importance of a multi-objective fitness in noisy rule-based machine learning. Next we examine the effect of our proposed rule compaction strategy (SRC) in comparison with no rule compaction and QRF. The results for v2.1.1 in Table 1 represent no rule compaction, and the following two columns present the results of QRF and SRC being independently applied to the same rule populations summarized in the column for v2.1.1. This comparison reveals that while QRF does indeed further reduce the rule population size while preserving if not slighly improving some performance metrics, SRC, benefiting from multi-objective fitness that better captures a global sense of rule value, significantly improves testing accuracy as well as all other power metrics with the exception of Both Power which yields a relatively small loss. Using SRC, we observe the largest significant increase in Both Co-Occurrence Power observed for any implementation of LCS or ExSTraCS on this array of simulated genetic benchmarks [6, 5, 2, 11]. This performance metric has been by far the most difficult to improve. Given that SRC dramatically reduces the population size, while simultaneously improving performance relative the population without compaction, this strategy is an improvement over QRF and other strategies evaluated in [12].

In a related analysis, we sought to characterize evolved rule-fitness fronts learned under different conditions of problem complexity and noise. The goal was to see if properties of the front could be applied to appropriately adapt

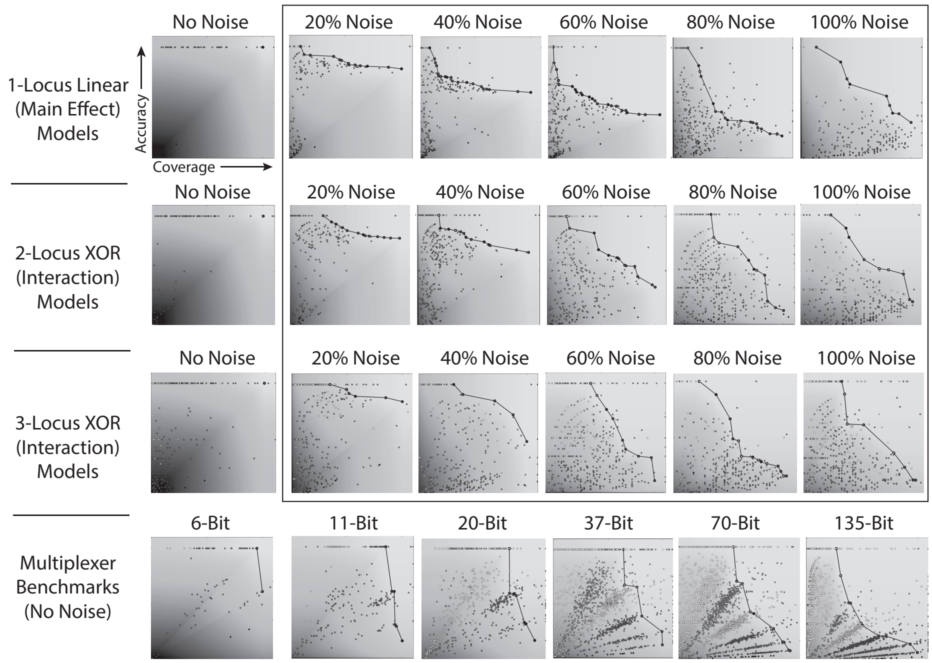

Fig. 2. Pareto inspired rule front comparisons. Each box gives the respective rule-fitness front, with accuracy and coverage axes each scaled between 0 and 1. Points represent rules in the final rule population. The background shading under the curve represents a basic illustration of underlying relative rule-fitness. Note that points found above the front are NEC rules with likely overestimates of objectives. The large black box groups all analyses involving noisy data.

the fitness function to include a more appropriate objective bias without prior problem knowledge. Figure 2 organizes a series of PIMORF rule-fronts learned on an array of benchmark datasets modeling main effects, pure 2-way, or 3-way interactions, or clean multiplexer benchmark problems of increasing complexity. We summarize some interesting observations, but concede that preliminary analyses seeking to apply these characteristics to predict whether the underlying problem was clean or noisy during learning, suggest that none of these trends can be universally applied as a reliable discriminator of clean vs. noisy problems. Relatively 'simple' patterns in the data such as main effects or relatively complex clean data patterns tend to yield a single point rule front (1, 2, and 3 locus models without noise). In such problems, objective weighting likely makes little to no difference, since optimal rules will be perfectly accurate and correctly cover the largest number of training instances. As clean problems become more complex (e.g. 4, 5, or 6 locus interactions), or include heterogeneity, we would not expect optimal rules to also cover the most instances. This is because over-general rules, with sub-optimal accuracy, can correctly cover a larger number of instances than an optimally accurate rule in complex problem spaces.

For each front with multiple points, consider the points at the ends of the front. Let's call the far right point the '*CoverMax*' or the accuracy observed at the largest coverage. The point on the far left we will call the '*Accuracy-Max*', or the largest coverage observed at the maximum rule accuracy. One interesting trend is that in partially noisy problems, *CoverMax* tends to be not only large, but larger than *AccuracyMax*. A more general way to view this trend is to notice that partially noisy problems tend to have a shallow overall slope. Alternatively, in clean, complex problems, such as the set of increasingly complex multiplexer problems, *AccuracyMax* tends to be both large and larger than *CoverMax*, or more generally, the slope of these fronts are steep. Unfortunately, these trends become unreliable indicators when (A) there is insufficient signal, or (B) problem complexity increases but the noise level fixed, or (C) the complexity/dimensionality of a problem become so great that the magnitude of *AccuracyMax* maxes it difficult to distinguish a complex clean rule-front from a completely noisy signal. This makes the implementation of an automated 'switch', shifting from accuracy to coverage bias problematic. In a clean but complex problem, until at least one optimal rule is found, the characteristics of the front might suggest that the problem is noisy and add a coverage bias. The addition of the wrong bias makes it even more unlikely that optimal rules will be identified, and that the rule front will be correctly updated to an accurately characteristic shape. One final observation for the multiplexer problems, is that we can see clusters of rules forming linear patterns. These groups turned out to correspond with the number of attributes specified in respective rules. Here we can effectively observe the different linear relationships between the accuracy and coverage within candidate rules that have not specified all of the necessary attributes to correctly cover the underlying multiplexer problem (e.g. in the 135-bit problem the 5 clearly identifiable groups correspond to 1-5 attributes having been specified in those rules.

4 Conclusions

The initial results presented in this paper demonstrate the potential benefits of a Pareto-front inspired LCS rule-fitness and support taking an agnostic approach to objective weighting in the likely absence signal to noise ratio prior knowledge in real-world problems. Therefore to promote effective modeling (i.e. accurate prediction and interpretable solutions) in problem domains that are not known to be 100% signal, a key goal should be to identify or properly estimate the signal to noise ratio, and apply this information to correctly weight accuracy and coverage objectives in the rule fitness function. Despite observing some interesting trends comparing simulated datasets with clean to noisy signals, we have not identified a reliable 'switch' that could be employed to automatically adapt the algorithm to employ the proper objective bias. Future work will explore a purely agnostic Pareto-based rule-fitness to evolve rules and rely on a rule compaction scheme to test different objective weight ratios, and select the best one as the final rule-set. While this work focuses on the adaptation of rule-based machine learning to problems with unknown noise properties, multi-objective fitness could still benefit performance on clean problems, where a small explicit generalization pressure, has the potential to speed up learning beyond the underlying implicit generalization pressures and the use of subsumption.

Acknowledgments. The computations in this work were performed on the Discovery cluster supported by the Research Computing group, ITS at Dartmouth College. This work was supported by NIH grants AI116794, LM009012, LM010098, EY022300, LM011360, CA134286, and GM103534.

References

1. Urbanowicz, R.J., Moore, J.H.: Learning classifier systems: a complete introduction, review, and roadmap. Journal of Artificial Evolution and Applications (2009) 1
2. Urbanowicz, R.J., Moore, J.H.: ExSTraCS 2.0: description and evaluation of a scalable learning classifier system. Evolutionary intelligence **8**(2-3) (2015) 89–116
3. Bernadó-Mansilla, E., Garrell-Guiu, J.M.: Accuracy-based learning classifier systems: models, analysis and applications to classification tasks. Evolutionary Computation **11**(3) (2003) 209–238
4. Orriols, A., Bernadó-Mansilla, E.: Class imbalance problem in ucs classifier system: fitness adaptation. In: Evolutionary Computation, 2005. The 2005 IEEE Congress on. Volume 1., IEEE (2005) 604–611
5. Urbanowicz, R.J., Bertasius, G., Moore, J.H.: An extended Michigan-style learning classifier system for flexible supervised learning,classification, and data mining. In: Parallel Problem Solving from Nature–PPSN XIII. Springer (2014) 211–221
6. Urbanowicz, R., Moore, J.: The application of Michigan-style learning classifier systems to address genetic heterogeneity and epistasis in association studies. In: Proceedings of the 12th annual conference on Genetic and evolutionary computation, ACM (2010) 195–202
7. Bernadó-mansilla, E., Llorà, X., Traus, I.: Multiobjective learning classifier systems: an overview. Tech Rep 2005020 (2005)
8. Llorà, X., Goldberg, D.E.: Bounding the effect of noise in multiobjective learning classifier systems. Evolutionary Computation **11**(3) (2003) 279–298
9. Bacardit, J., Garrell, J.M.: Bloat control and generalization pressure using the minimum description length principle for a pittsburgh approach learning classifier system. In: Learning Classifier Systems. Springer (2007) 59–79
10. Mansilla, E.B., Guiu, J.M.G.: MOLeCS: Using multiobjective evolutionary algorithms for learning. In: Evolutionary Multi-Criterion Optimization, Springer (2001) 696–710
11. Urbanowicz, R., Moore, J.: Retooling fitness for noisy problems in a supervised Michigan-style learning classifier system. In: Proceedings of the 2015 on Genetic and Evolutionary Computation Conference, ACM (2015) 591–598
12. Tan, J., Moore, J., Urbanowicz, R.: Rapid rule compaction strategies for global knowledge discovery in a supervised learning classifier system. In: Advances in Artificial Life, ECAL. Volume 12. (2013) 110–117
13. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast and elitist multiobjective genetic algorithm: NSGA-II. Evolutionary Computation, IEEE Transactions on **6**(2) (2002) 182–197
14. Urbanowicz, R.J., Kiralis, J., Sinnott-Armstrong, N.A., Heberling, T., Fisher, J.M., Moore, J.H.: GAMETES: a fast, direct algorithm for generating pure, strict, epistatic models with random architectures. BioData Mining **5**(1) (2012) 1