Variance-based Learning Classifier System without Convergence of Reward Estimation

Takato Tatsumi
The University of
Electro-Communications
1-5-1, Chofugaoka, Chofu-shi
Tokyo, Japan
tatsumi@uec.ac.jp

Takahiro Komine
The University of
Electro-Communications
1-5-1, Chofugaoka, Chofu-shi
Tokyo, Japan
tkomine@cas.hc.uec.ac.jp

Masaya Nakata
Yokohama National University
79-5 Tokiwadai, Hodogaya-ku,
Yokohama, Japan
nakata-masaya-tb@ynu.ac.jp

Hiroyuki Sato
The University of
Electro-Communications
1-5-1, Chofugaoka, Chofu-shi
Tokyo, Japan
sato@hc.uec.ac.jp

Tim Kovacs
University of Bristol, Bristol,
BS8 1TH, UK
kovacs@cs.bris.ac.uk

Keiki Takadama
The University of
Electro-Communications
1-5-1, Chofugaoka, Chofu-shi
Tokyo, Japan
keiki@inf.uec.ac.jp

CCS Concepts

•**Computing methodologies** → *Rule learning;*

Keywords

learning classifier system; XCS; accuracy; generalization

1. INTRODUCTION

Learning Classifier System (LCS) [2] is an evolutionary machine learning method that is constituted by reinforcement learning and genetic algorithm. As an important feature of LCS, LCS can acquire generalized rules that match multiple states using # symbol. Among LCSs, Accuracy-based LCS (XCS) [4] can acquire "accurate" generalized rules by reducing the difference between the predicted reward and the acquired reward, but XCS is hard to correctly estimate such difference in noisy environments. To address this issue, our previous research proposed XCS-SAC (XCS with Self-adaptive Accuracy Criterion) [3] for noisy environments. Since the estimated standard deviation of the rewards of the inaccurate rules is larger than that of the accurate ones, the fitness of rules in XCS-SAC is calculated according to the estimated standard deviation of the rewards.

However, XCS-SAC needs to wait until convergence of the estimated standard deviation of all state-action pairs. This paper pays attention that the average value of rewards is distributed around a true value. To overcome this problem, this paper proposes XCS without Convergence of Reward Estimation (XCS-CRE) that can determine the accuracy of rules according to the distribution range of the average value of rewards of the matched state-action pair.

GECCO '16 July 20-24, 2016, Denver, CO, USA

ACM ISBN 978-1-4503-4323-7/16/07.
DOI: http://dx.doi.org/10.1145/2908961.2909008

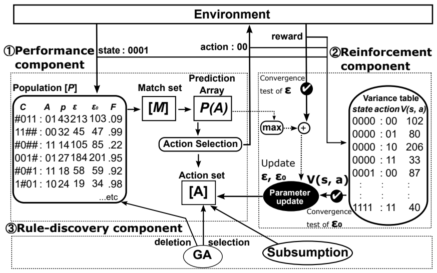
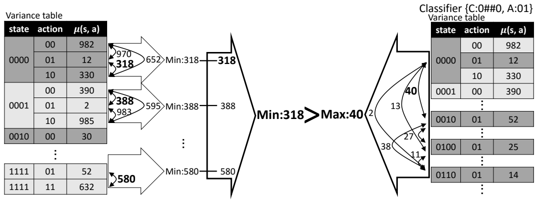


Figure 1: Learning mechanism of XCS-SAC

2. XCS-SAC

The classifiers of XCS-SAC are composed of if-then rule and some evaluation values such as error (ϵ) and the accuracy criterion ϵ_0. These values are updated until the sample standard deviation $S(s, a)$ for all possible state-action pairs are converged. The $S(s, a)$ is called the variance table which is updated while learning. Since the different values of the rewards are received in noisy environments even when the same action is executed in a state, the estimated standard deviation is also different from the convergent value in some cases especially when the number of data is small. After the convergence of values in the variance table, the estimated standard deviation values are fixed and the ϵ_0 of the classifier are determined as the weighted average of all matched state-action pairs. If the ϵ_0 is less than the estimated standard deviation of acquired rewards (ϵ), XCS-SAC regards that the classifier is accurate.

The algorithm of XCS-SAC is summarized as follows: (1) the match set $[M]$ is created from the classifiers which condition part matches the input in population $[P]$; (2) the prediction array is calculated by the prediction of the matched classifiers and their actions; (3) the action set $[A]$ is created from the classifiers that have the same action in $[M]$; (4) the action is executed and the reward P is received from the environment; (5) after receiving the reward, the reinforcement component is executed and the evolution component is executed at the certain time.



state	action	μ(s, a)
0000	00	982
	01	12
	10	330
0001	00	390
	01	2
	10	985
0010	00	30
⋮		
1111	01	52
1111	11	632

state	action	μ(s, a)
0000	00	982
	01	12
	10	330
0001	00	390
	⋮	
0010	01	52
	⋮	
0100	01	25
	⋮	
0110	01	14
	⋮	

Figure 2: Subsumption condition of XCS-CRE

3. XCS-CRE

Although the update mechanism of the accuracy criterion ϵ_0 and the error ϵ of XCS-CRE are the same as those of XCS-SAC, XCS-CRE does not have the convergence condition on the standard deviation of rewards like XCS-SAC because it can learn accurate generalized classifiers without waiting their convergence.For this issue, the new accuracy determination condition is added in XCS-SAC as eq. (1) > eq. (2), where $\mu(s, a)$ indicates the averaged reward of executing the action a in the state s. This inequality means that the reward of the accurate classifier fluctuates in a narrow range. In static noisy environment, the averaged reward is close to the true value of the reward.

$$\min\{\min\{|\mu(s,a) - \mu(s,a')|, \{a, a'\} \in action\}, s \in state\} \tag{1}$$

$$\max\{|\mu(s,a) - \mu(s',a)|, \{(s,a),(s',a)\}\ matched\ cl\} \tag{2}$$

In order to understand the above new condition, let's focus on the classifier {C:0##0, A:01}, where the condition is represented 0##0 while the action is represented by 01. As shown in the left side of Figure 2, XCS-CRE starts to calculate the minimum difference among the averaged rewards ($\mu(s, a)$) of the different actions in all states, (*e.g.,* 318 in state 0000, 388 in state 0001, and 580 in state 1111) and selects the minimum one (*i.e.,* 318) among the above values (*i.e.,* 318, 388, ..., 580). As shown in the right side of Figure 2, on the other hand, XCS-CRE calculates the maximum difference among the averaged rewards ($\mu(s, a)$) of the different states (*i.e.,* 0000, 0010, 0100, 0110), matched to the condition of the classifier (*i.e.,* 0##0), in the same action (*i.e.,* 01), and selects the maximum one (*i.e.,* 40) among the above values (*i.e.,* 1, 11, 13, 27, 38, 40). Since 318 > 40, the classifier {C:0##0, A:01} can be regarded as the accurate one.

4. EXPERIMENT

4.1 Noisy multiplexer problem

To investigate the effectiveness of XCS-CRE, this paper compares the results of XCS, XCS-SAC, and XCS-CRE in 6-Multiplexer problem.In this problem, the first 2 bit of the input data converts to the decimal number d, the $2 + d$-th bit is the correct answer. XCSs are expected to acquire the accurate generalized classifiers replacing the bits (which do not affect the answer) with #. The reward is set 1000 for the correct answer and 0 for the wrong answer. The noise that follows a normal distribution of mean 0 and variance 300^2 is added to all rewards.

In the experiment, 100000 iterations is conducted in one trial and the 50 trials are conducted with the different random seeds. The following evaluation criteria are employed: 1) **Correct rate**, where the higher correct rate is better than lower one; and 2) **Population size**, which evaluates the number of the classifiers in the [P], and the smaller population size is better than larger one because the necessary memory size becomes small. The parameters of XCSs are the mostly standard ones [1].

4.2 Results

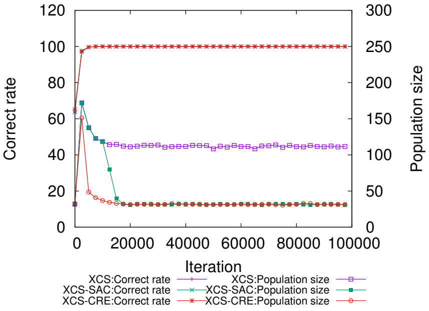


Figure 3: Correct rate and population size

Figure 3 shows the result, the horizon axis represents the iteration, the left vertical axis represents correct rate, and the right vertical axis represents population size. As shown in this figure, the correct rate of XCS, XCS-SAC and XCS-CRE reached 100%. The population size of XCS becomes approximately 110, while that of XCS-SAC and XCS-CRE become approximately 30 which is smaller than 110.This result also suggests that the convergence time to XCS-CRE is faster than that of XCS-SAC.

5. CONCLUSION

This paper proposed XCS-CRE which can acquire accurate generalized classifiers in noise environments without waiting for the convergence of the averaged rewards of all state-action pairs. The experimental result suggested that the correct rate of XCS-CRE converges 100%, while its population size of XCS-CRE quickly recuses in comparison with other LCSs.

The following research much be done in the near future: (1) the environment changes in the middle of learning; (2) environment where the average of rewards depend on the intensity of noise.

6. REFERENCES

[1] M. V. Butz, T. Kovacs, P. L. Lanzi, and S. W. Wilson. Toward a Theory of Generalization and Learning in XCS. *Evolutionary Computation, IEEE Transactions on*, 8(1):28–46, 2004.

[2] J. H. Holland. Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems. *Machine learning*, pages 593–623, 1986.

[3] T. Tatsumi, T. Komine, M. Nakata, H. Sato, and K. Takadama. A Learning Classifier System that Adapts Accuracy Criterion. *Transaction of the Japanese Society for Evolutionary Computation*, 6(2):90–103, 2015.

[4] S. W. Wilson. Classifier Fitness Based on Accuracy. *Evol. Comput.*, 3(2):149–175, June 1995.