

学習分類子システムにおける不確実性を伴う
データからの知識獲得メカニズムの設計

辰巳 嵩豊

電気通信大学 大学院情報理工学研究科
博士(工学)の学位申請論文

2020年3月

学習分類子システムにおける不確実性を伴う
データからの知識獲得メカニズムの設計

論文審査委員会

主査	高玉	圭樹	教授
委員	西野	哲朗	教授
委員	田野	俊一	教授
委員	庄野	逸	教授
委員	佐藤	寛之	准教授

Copyright (C) 2020 Takato Tatsumi All Rights Reserved.

Abstract

This thesis focuses on a Learning Classifier System (LCS) as one of data mining methods, proposes the four LCSs that can acquire knowledge from uncertain data, and aims at investigating the effectiveness of the proposed LCSs through the general benchmark problem. For this purpose, this thesis starts to classify the uncertain data as (i) the uncertain input and/or output in data and (ii) the uncertain evaluation of data, and explores the mechanisms that can cope with the two types of the uncertain data to incorporate them with Accuracy-based Learning Classifier System (XCS) as the major conventional LCS. Concretely, this thesis proposes XCS based on Estimated Reward (XCS-ER) for the uncertainty (i), XCS based on Range of Reward (XCS-RR) for the uncertainty (ii), XCS based on Estimated Correctness (XCS-EC) and XCS based on Reward Group (XCS-RG) for the both uncertainties (i) and (ii). Note that XCS-EC is extended from XCS-ER while XCS-RG is extended from XCS-RR. Through the intensive comparison of the proposed LCSs with XCS on the Multiplexer problem which is modified to add the noise in data to simulate the uncertain data, the following implications have been revealed: (1) XCS-ER and XCS-RR can acquire the optimal generalized classifiers which result in a high correct rate of the data classification in the uncertainties (i) and (ii), respectively; (2) XCS-EC and XCS-RG can acquire the optimal generalized classifiers in all uncertainties; (3) In the uncertainty (ii), XCS-RR, XCS-EC, and XCS-RG can acquire the optimal generalized classifiers even if the shape of the reward distribution is normal, uniform, exponential, etc. ; (4) XCS-ER, XCS-EC, and XCS-RG can acquire the optimal generalized classifiers in the uncertainties that can be combined with the uncertainties that can be handled; (5) The analytical applicable range of each proposed method is acquired by the conditions that must be met by an accurate classifier.

概要

本論文はデータマイニング手法の学習分類子システムにおいて、知識獲得が困難な不確実性を伴うデータから知識獲得する手法を提案し、その有効性の検証を目的とする。その実現に向け、データが伴う不確実性を、(i) 学習データである入力や出力が変化する入力における不確実性と、(ii) 入力に対する出力結果の信憑性（評価値）が変化する判定の信憑性における不確実性に分け、それらに対処可能なメカニズムを考案し、従来手法の正確性に基づく学習分類子システム (Accuracy-based Learning Classifier System: XCS) に組み込む。さらに、評価実験を通し不確実性の種類に依らないメカニズムの設計を目指す。本論文では (i) と (ii) それぞれあるいは (i) と (ii) の両方の不確実性に対処する手法を考案する。具体的には、(i) の不確実性に関して、今までの判定結果の正解・不正解の回数の大小関係から、一時的な誤判定を判断し、判定ルールの維持/削除を決める。このアプローチを組み入れた XCS based on Estimated Reward (XCS-ER) を提案する。次に (ii) の不確実性に関して、他の判定ルールより揺らぎ幅の広い判定ルールを削除する。このアプローチを組み入れた XCS based on Range of Reward (XCS-RR) を提案する。さらに、(i) と (ii) の両方の不確実性に関し、(i) の不確実性に対するアプローチを「正解率・不正解率の大小関係」に拡張して判定する XCS based on Estimated Correctness (XCS-EC) と、(ii) の不確実性に対するアプローチを「正解と不正解の両方にまたがる揺らぎの識別」に拡張して判定する XCS based on Reward Group (XCS-RG) を提案する。提案手法の有効性の検証のため、ベンチマーク問題の Multiplexer 問題に (i) と (ii) の不確実性を模擬したノイズを付加し、分類精度（正解率）と、多くのデータを分類する汎用的なルール（最適な判定ルール）の獲得を評価する。XCS と 4 つの提案手法に適用し次の知見を得た。(1) XCS-ER および XCS-RR は不確実性の種類が対応可能なものであれば、不確実性の強度が知らされていなくても正解率が高く最適な判定ルールの獲得に成功した。(2) XCS-EC および XCS-RG は不確実性の種類およびその強度が知らされていなくても正解率が高く最適な判定ルールの獲得に成功した。(3) XCS-RR, XCS-EC および XCS-RG は判定の信憑性における不確実性による報酬の分布が正規分布以外の分布であっても正解率が高く最適な判定ルールの獲得に成功した。(4) XCS-ER, XCS-EC および XCS-RG は単体で対応可能な不確実性の種類であれば、それらの不確実性が複合した

場合においても正解率が高く最適な判定ルールの獲得に成功した。(5)4つの提案手法について正確とみなすべき分類子が満たす条件から解析的な適用範囲を求めた。

目次

第 1 章	序論	1
1.1	背景	1
1.2	目的	3
1.3	章立て	4
第 2 章	対象とする不確実性	6
2.1	不確実性の分類	6
2.1.1	入力と出力における不確実性	6
2.1.2	判定の信憑性における不確実性	7
2.2	不確実性への従来対処法	8
2.2.1	前処理による対処	8
2.2.2	不確実性に対応を図った学習手法	8
2.3	本研究の不確実性へのアプローチ	13
第 3 章	学習分類子システム	15
3.1	フレームワーク	15
3.2	学習分類子システムの分類	16
3.2.1	個体表現による分類	16
3.2.2	学習戦略とルール評価戦略による分類	16
3.2.3	対象とする問題による分類	17
3.3	正確性に基づく学習分類子システム	17
3.3.1	構成	18
3.3.2	学習メカニズム	19
3.3.3	アルゴリズム	27
第 4 章	問題設計	29
4.1	Multiplexer 問題	29
4.2	対象とする不確実性の模擬	30

4.2.1	入力における不確実性の模擬	31
4.2.2	出力における不確実性の模擬	31
4.2.3	判定の信憑性における不確実性の模擬	31
4.3	評価基準	32
第 5 章	入出力における不確実性に対応可能な学習分類子システム	35
5.1	入出力における不確実性に対するアプローチ	36
5.2	構成	37
5.2.1	アーキテクチャ	37
5.2.2	分類子	37
5.3	学習メカニズム	38
5.3.1	強化部	38
5.3.2	発見部	41
5.4	アルゴリズム	42
5.5	実験	42
5.5.1	実験内容	42
5.5.2	実験設定	42
5.5.3	実験結果	43
5.6	考察	52
5.6.1	XCS の入出力における不確実性への不応	52
5.6.2	XCS-ER の報酬推測の有効性とその限界	53
5.6.3	XCS-ER の新設されたパラメータ	55
第 6 章	判定の信憑性における不確実性に対応可能な学習分類子システム	58
6.1	判定の信憑性における不確実性に対するアプローチ	58
6.2	構成	59
6.2.1	アーキテクチャ	59
6.2.2	分類子	60
6.3	学習メカニズム	60
6.3.1	強化部	60
6.3.2	発見部	62
6.4	アルゴリズム	62
6.5	実験	65
6.5.1	実験内容	65
6.5.2	実験設定	65
6.5.3	実験結果	65

6.6	考察	69
6.6.1	XCS-RR の正確な分類子の識別	69
6.6.2	XCS-RR の新設されたパラメータ	71
第 7 章	入出力および判定の信憑性における不確実性に対応可能な学習分類子システム	73
7.1	入出力における不確実性および判定の信憑性における不確実性に対するアプローチ	73
7.2	獲得報酬記録に基づく学習分類子システム	75
7.2.1	構成	75
7.2.2	学習メカニズム	76
7.2.3	アルゴリズム	80
7.3	正誤推測に基づく学習分類子システム	80
7.3.1	構成	80
7.3.2	学習メカニズム	82
7.3.3	アルゴリズム	85
7.4	実験	87
7.4.1	実験内容	87
7.4.2	実験設定	88
7.4.3	実験結果	88
7.5	考察	102
7.5.1	XCS-RG の収束速度	102
7.5.2	XCS-RG の包摂条件	103
7.5.3	XCS-EC の正誤推測の有効性とその限界	105
7.5.4	XCS-EC の新設されたパラメータ	108
第 8 章	様々な不確実性を伴う場合に対する適応性	114
8.1	実験	114
8.1.1	実験内容	114
8.1.2	実験設定	116
8.1.3	実験結果	117
8.2	考察	119
第 9 章	不確実性に対応可能な学習分類子システムの設計論	127
9.1	不確実性を伴うデータから知識を獲得するメカニズムの設計	127

9.1.1	入出力における不確実性を伴うデータに対応するメカニズム の設計	127
9.1.2	判定の信憑性における不確実性を伴うデータに対応するメカ ニズムの設計	128
9.1.3	アプローチによる不確実性への対応の差異	128
9.2	提案手法の解析的適用範囲	130
9.2.1	XCS-ER	131
9.2.2	XCS-RR	133
9.2.3	XCS-EC	137
9.2.4	XCS-RG	143
第 10 章	結言	151
10.1	本研究の成果	151
10.2	今後の課題	153
10.2.1	複数の手法に共通する問題	153
10.2.2	手法固有の問題	155
	謝辞	157
	参考文献	158

目次

1.1	分類問題に用いられる手法	2
1.2	分類子の一般化の例	2
2.1	入力における不確実性の例	7
2.2	出力における不確実性の例	7
2.3	判定の信憑性における不確実性の例	7
2.4	非マルコフ過程の迷路問題	9
2.5	内部状態による入力の識別	10
2.6	スリップする迷路問題	10
2.7	判定の信憑性における不確実性を伴うデータにおける分類子の一般化	12
2.8	本研究の不確実性へのアプローチ	14
3.1	学習フレームワーク	15
3.2	LCS の分類子の一般化の概要	17
3.3	分類子の正確性	18
3.4	XCS のアーキテクチャ	18
3.5	XCS の学習メカニズム	20
3.6	入力との照合と出力の決定	21
3.7	正確性の評価	22
3.8	分類子の包摂	23
3.9	GA による分類子の生成と削除	26
4.1	11-Multiplexer 問題	29
4.2	入力における不確実性の模擬	31
4.3	出力における不確実性の模擬	31
4.4	判定の信憑性における不確実性の模擬	32
4.5	最大適合度 F の分類子	33
5.1	入出力における不確実性を伴う場合	35

5.2	XCS-ER の概要	36
5.3	XCS-ER のアーキテクチャ	37
5.4	報酬推測機構	39
5.5	$P_I = P_O = 0$ における正答率の平均および分類子数の平均	45
5.6	入力における不確実性 (ケース U: $P_I = 0.05$) を伴う場合における正答率の平均および分類子数の平均	46
5.7	入力における不確実性 (ケース U: $P_I = 0.1$) を伴う場合における正答率の平均および分類子数の平均	46
5.8	入力における不確実性 (ケース U: $P_I = 0.2$) を伴う場合における正答率の平均および分類子数の平均	46
5.9	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.05$) を伴う場合における正答率の平均および分類子数の平均	47
5.10	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.1$) を伴う場合における正答率の平均および分類子数の平均	47
5.11	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.2$) を伴う場合における正答率の平均および分類子数の平均	47
5.12	出力における不確実性 (ケース U: $P_O = 0.2$) を伴う場合における正答率の平均および分類子数の平均	49
5.13	出力における不確実性 (ケース U: $P_O = 0.3$) を伴う場合における正答率の平均および分類子数の平均	49
5.14	出力における不確実性 (ケース U: $P_O = 0.4$) を伴う場合における正答率の平均および分類子数の平均	49
5.15	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.2$) を伴う場合における正答率の平均および分類子数の平均	50
5.16	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.3$) を伴う場合における正答率の平均および分類子数の平均	50
5.17	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.4$) を伴う場合における正答率の平均および分類子数の平均	50
5.18	入出力における不確実性の複合 ($(P_I, P_O) = (0.05, 0.2)$) を伴う場合における正答率の平均および分類子数の平均	52
5.19	XCS における正確な分類子の ϵ	53
5.20	XCS-ER の報酬推測機構	54
5.21	報酬推測の正答率	55
5.22	$P_O = 0.2$ における $[O]$ を獲得できない場合の報酬推測の正答率	55
5.23	$\theta_{RE} = 0$ における正答率の平均および分類子数の平均	56

5.24	$\theta_{RE} = 2$ における正答率の平均および分類子数の平均	56
5.25	$\theta_{RE} = 5$ における正答率の平均および分類子数の平均	56
5.26	$\theta_{RE} = 10$ における正答率の平均および分類子数の平均	57
6.1	判定の信憑性における不確実性を伴う場合	58
6.2	XCS-RR の概要	59
6.3	XCS-RR のアーキテクチャ	60
6.4	正確な分類子の識別	62
6.5	判定の信憑性における不確実性 ($\sigma_R = 0$) を伴う場合における正答率の平均および分類子数の平均	66
6.6	判定の信憑性における不確実性 (ケース U: $\sigma_R = 200$) を伴う場合における正答率の平均および分類子数の平均	67
6.7	判定の信憑性における不確実性 (ケース U: $\sigma_R = 700$) を伴う場合における正答率の平均および分類子数の平均	67
6.8	判定の信憑性における不確実性 (ケース U: $\sigma_R = 1500$) を伴う場合における正答率の平均および分類子数の平均	67
6.9	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 200$) を伴う場合における正答率の平均および分類子数の平均	68
6.10	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 700$) を伴う場合における正答率の平均および分類子数の平均	68
6.11	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 1500$) を伴う場合における正答率の平均および分類子数の平均	68
6.12	判定の信憑性における不確実性を伴う場合の報酬の分布	70
6.13	$\theta_{RR} = 0\%$ における正答率の平均および分類子数の平均	71
6.14	$\theta_{RR} = 10\%$ における正答率の平均および分類子数の平均	71
6.15	$\theta_{RR} = 70\%$ における正答率の平均および分類子数の平均	72
6.16	$\theta_{RR} = 90\%$ における正答率の平均および分類子数の平均	72
7.1	入出力における不確実性および判定の信憑性における不確実性を伴う場合	74
7.2	XCS-RG の概要	74
7.3	XCS-EC の概要	75
7.4	XCS-RG のアーキテクチャ	76
7.5	報酬間の距離 d_r と分類子の獲得報酬範囲 r_{cl}	78
7.6	XCS-EC のアーキテクチャ	82
7.7	正誤推測	83

7.8	$P_I = P_O = \sigma_R = 0$ における正答率の平均および分類子数の平均 . . .	89
7.9	入力における不確実性 (ケース U: $P_I = 0.05$) を伴う場合における正答率の平均および分類子数の平均	90
7.10	入力における不確実性 (ケース U: $P_I = 0.1$) を伴う場合における正答率の平均および分類子数の平均	90
7.11	入力における不確実性 (ケース U: $P_I = 0.2$) を伴う場合における正答率の平均および分類子数の平均	90
7.12	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.05$) を伴う場合における正答率の平均および分類子数の平均	91
7.13	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.1$) を伴う場合における正答率の平均および分類子数の平均	91
7.14	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.2$) を伴う場合における正答率の平均および分類子数の平均	91
7.15	出力における不確実性 (ケース U: $P_O = 0.2$) を伴う場合における正答率の平均および分類子数の平均	93
7.16	出力における不確実性 (ケース U: $P_O = 0.3$) を伴う場合における正答率の平均および分類子数の平均	93
7.17	出力における不確実性 (ケース U: $P_O = 0.4$) を伴う場合における正答率の平均および分類子数の平均	93
7.18	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.2$) を伴う場合における正答率の平均および分類子数の平均	94
7.19	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.3$) を伴う場合における正答率の平均および分類子数の平均	94
7.20	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.4$) を伴う場合における正答率の平均および分類子数の平均	94
7.21	判定の信憑性における不確実性 (ケース U: $\sigma_R = 200$) を伴う場合における正答率の平均および分類子数の平均	96
7.22	判定の信憑性における不確実性 (ケース U: $\sigma_R = 700$) を伴う場合における正答率の平均および分類子数の平均	96
7.23	判定の信憑性における不確実性 (ケース U: $\sigma_R = 1500$) を伴う場合における正答率の平均および分類子数の平均	96
7.24	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 200$) を伴う場合における正答率の平均および分類子数の平均	97
7.25	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 700$) を伴う場合における正答率の平均および分類子数の平均	97

7.26	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 1500$) を伴う場合における正答率の平均および分類子数の平均	97
7.27	入出力における不確実性の複合 ($(P_I, P_O) = (0.05, 0.15)$) を伴う場合における正答率の平均および分類子数の平均	102
7.28	入出力における不確実性の複合 ($(P_I, P_O) = (0.05, 0.2)$) を伴う場合における正答率の平均および分類子数の平均	102
7.29	入力における不確実性および判定の信憑性における不確実性の複合 ($(P_I, \sigma_R) = (0.03, 500)$) を伴う場合における正答率の平均および分類子数の平均	103
7.30	入力における不確実性および判定の信憑性における不確実性の複合 ($(P_I, \sigma_R) = (0.1, 200)$) を伴う場合における正答率の平均および分類子数の平均	103
7.31	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.1, 500)$) を伴う場合における正答率の平均および分類子数の平均	105
7.32	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.2, 100)$) を伴う場合における正答率の平均および分類子数の平均	105
7.33	入出力における不確実性および判定の信憑性における不確実性の複合 ($(P_I, P_O, \sigma_R) = (0.03, 0.5, 200)$) を伴う場合における正答率の平均および分類子数の平均	106
7.34	入出力における不確実性および判定の信憑性における不確実性の複合 ($(P_I, P_O, \sigma_R) = (0.03, 0.15, 500)$) を伴う場合における正答率の平均および分類子数の平均	106
7.35	入力における不確実性下での d_r および r_{cl}	107
7.36	出力における不確実性下での d_r および r_{cl}	107
7.37	判定の信憑性における不確実性下での d_r および r_{cl}	107
7.38	XCS-EC の判定における不確実性を伴う場合の正誤推測機構	108
7.39	正誤推測の正答率	109
7.40	$\sigma_R = 200$ における正誤推測の正答率	109
7.41	$\theta_{EC} = 0$ ($P_O = 0.2$) における正答率の平均および分類子数の平均	110
7.42	$\theta_{EC} = 2$ ($P_O = 0.2$) における正答率の平均および分類子数の平均	110
7.43	$\theta_{EC} = 5$ ($P_O = 0.2$) における正答率の平均および分類子数の平均	111
7.44	$\theta_{EC} = 10$ ($P_O = 0.2$) における正答率の平均および分類子数の平均	111
7.45	$\theta_{EC} = 0$ ($\sigma_R = 200$) における正答率の平均および分類子数の平均	112

7.46	$\theta_{EC} = 2$ ($\sigma_R = 200$) における正答率の平均および分類子数の平均 . . .	112
7.47	$\theta_{EC} = 5$ ($\sigma_R = 200$) における正答率の平均および分類子数の平均 . . .	113
7.48	$\theta_{EC} = 10$ ($\sigma_R = 200$) における正答率の平均および分類子数の平均 . . .	113
8.1	指数分布	115
8.2	一様分布	116
8.3	判定の信憑性における不確実性 (指数分布: $(\lambda, x) = (2, 100)$) を伴う 場合における正答率の平均および分類子数の平均	117
8.4	判定の信憑性における不確実性 (指数分布: $(\lambda, x) = (1, 300)$) を伴う 場合における正答率の平均および分類子数の平均	118
8.5	判定の信憑性における不確実性 (指数分布: $(\lambda, x) = (1, 400)$) を伴う 場合における正答率の平均および分類子数の平均	118
8.6	判定の信憑性における不確実性 (指数分布: $(\lambda, x) = (1, -100)$) を伴う 場合における正答率の平均および分類子数の平均	119
8.7	判定の信憑性における不確実性 (指数分布: $(\lambda, x) = (1, -500)$) を伴う 場合における正答率の平均および分類子数の平均	119
8.8	判定の信憑性における不確実性 (一様分布: $w = \pm 700$) を伴う場合 における正答率の平均および分類子数の平均	120
8.9	判定の信憑性における不確実性 (一様分布: $w = \pm 1500$) を伴う場合 における正答率の平均および分類子数の平均	120
8.10	判定の信憑性における不確実性 (一様分布: $w = \pm 2000$) を伴う場合 における正答率の平均および分類子数の平均	120
8.11	判定の信憑性における不確実性 (一様分布: $w = -2000$) を伴う場合 における正答率の平均および分類子数の平均	121
8.12	判定の信憑性における不確実性 (一様分布: $w = 500$) を伴う場合に おける正答率の平均および分類子数の平均	121
8.13	判定の信憑性における不確実性 (一様分布: $w = 700$) を伴う場合に おける正答率の平均および分類子数の平均	121
8.14	判定の信憑性における不確実性 (一様分布: $\sup\{w\} = 700$ (ケース V)) を伴う場合における正答率の平均および分類子数の平均	124
8.15	正規分布, 指数分布および一様分布が Iteration ごとに変わる場合に おける正答率の平均および分類子数の平均	125
8.16	分布形状と提案手法における正確性判定への影響	126
9.1	報酬分布のモデル化	131
9.2	出力における不確実性 ($P_O = 0.3$) を伴う場合における最適解の獲得	133

9.3	出力における不確実性 ($P_O = 0.35$) を伴う場合における最適解の獲得	133
9.4	XCS-RR の正確な分類子が満たす条件	134
9.5	判定の信憑性における不確実性 (一様分布: $w = \pm 900$) を伴う場合における正答率の平均および分類子数の平均	136
9.6	判定の信憑性における不確実性 (一様分布: $w = \pm 1000$) を伴う場合における正答率の平均および分類子数の平均	136
9.7	判定の信憑性における不確実性 ($\sigma_R = 200$) を伴う場合における最適解の獲得	138
9.8	判定の信憑性における不確実性 ($\sigma_R = 300$) を伴う場合における最適解の獲得	138
9.9	判定の信憑性における不確実性 ($\sigma_R = 900$) を伴う場合における正答率の平均および分類子数の平均	140
9.10	判定の信憑性における不確実性 ($\sigma_R = 1000$) を伴う場合における正答率の平均および分類子数の平均	140
9.11	出力における不確実性 ($P_O = 0.3$) を伴う場合における最適解の獲得	142
9.12	出力における不確実性 ($P_O = 0.35$) を伴う場合における最適解の獲得	142
9.13	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.25, 200)$) を伴う場合における正答率の平均および分類子数の平均	144
9.14	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.3, 200)$) を伴う場合における正答率の平均および分類子数の平均	144
9.15	XCS-RG の正確な分類子が満たす条件	144
9.16	判定の信憑性における不確実性 ($\sigma_R = 1000$) を伴う場合における最適解の獲得	146
9.17	判定の信憑性における不確実性 ($\sigma_R = 1100$) を伴う場合における最適解の獲得	146
9.18	出力における不確実性 ($P_O = 0.25$) を伴う場合における最適解の獲得	148
9.19	出力における不確実性 ($P_O = 0.3$) を伴う場合における最適解の獲得	148
9.20	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.2, 300)$) を伴う場合における正答率の平均および分類子数の平均	150
9.21	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.25, 300)$) を伴う場合における正答率の平均および分類子数の平均	150

10.1	$\nu = 5$ としたときの判定の信憑性における不確実性 ($\sigma_R = 200$) を伴う場合における正答率の平均および分類子数の平均	154
------	---	-----

表目次

2.1	不確実性に対応を図った学習手法	9
2.2	アプローチと対応する不確実性	14
4.1	11-Multiplexer 問題における最適解	30
5.1	$P_I = P_O = 0$ における最適解の獲得	44
5.2	入力における不確実性 (ケース U: $P_I = 0.05$) を伴う場合における最適解の獲得	45
5.3	入力における不確実性 (ケース U: $P_I = 0.1$) を伴う場合における最適解の獲得	45
5.4	入力における不確実性 (ケース U: $P_I = 0.2$) を伴う場合における最適解の獲得	45
5.5	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.05$) を伴う場合における最適解の獲得	48
5.6	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.1$) を伴う場合における最適解の獲得	48
5.7	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.2$) を伴う場合における最適解の獲得	48
5.8	出力における不確実性 (ケース U: $P_O = 0.2$) を伴う場合における最適解の獲得	51
5.9	出力における不確実性 (ケース U: $P_O = 0.3$) を伴う場合における最適解の獲得	51
5.10	出力における不確実性 (ケース U: $P_O = 0.4$) を伴う場合における最適解の獲得	51
5.11	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.2$) を伴う場合における最適解の獲得	51
5.12	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.3$) を伴う場合における最適解の獲得	51

5.13	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.4$) を伴う場合における最適解の獲得	51
5.14	入出力における不確実性の複合 ($(P_I, P_O) = (0.05, 0.2)$) を伴う場合における最適解の獲得	52
6.1	判定の信憑性における不確実性 ($\sigma_R = 0$) を伴う場合における最適解の獲得	69
6.2	判定の信憑性における不確実性 (ケース U: $\sigma_R = 200$) を伴う場合における最適解の獲得	69
6.3	判定の信憑性における不確実性 (ケース U: $\sigma_R = 700$) を伴う場合における最適解の獲得	69
6.4	判定の信憑性における不確実性 (ケース U: $\sigma_R = 1500$) を伴う場合における最適解の獲得	69
6.5	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 200$) を伴う場合における最適解の獲得	70
6.6	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 700$) を伴う場合における最適解の獲得	70
6.7	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 1500$) を伴う場合における最適解の獲得	70
7.1	$P_I = P_O = \sigma_R = 0$ における最適解の獲得	89
7.2	入力における不確実性 (ケース U: $P_I = 0.05$) を伴う場合における最適解の獲得	92
7.3	入力における不確実性 (ケース U: $P_I = 0.1$) を伴う場合における最適解の獲得	92
7.4	入力における不確実性 (ケース U: $P_I = 0.2$) を伴う場合における最適解の獲得	92
7.5	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.05$) を伴う場合における最適解の獲得	95
7.6	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.1$) を伴う場合における最適解の獲得	95
7.7	入力における不確実性 (ケース V: $\sup\{P_I\} = 0.2$) を伴う場合における最適解の獲得	95
7.8	出力における不確実性 (ケース U: $P_O = 0.2$) を伴う場合における最適解の獲得	98

7.9	出力における不確実性 (ケース U: $P_O = 0.3$) を伴う場合における最適解の獲得	98
7.10	出力における不確実性 (ケース U: $P_O = 0.4$) を伴う場合における最適解の獲得	98
7.11	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.2$) を伴う場合における最適解の獲得	99
7.12	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.3$) を伴う場合における最適解の獲得	99
7.13	出力における不確実性 (ケース V: $\sup\{P_O\} = 0.4$) を伴う場合における最適解の獲得	99
7.14	判定の信憑性における不確実性 (ケース U: $\sigma_R = 200$) を伴う場合における最適解の獲得	100
7.15	判定の信憑性における不確実性 (ケース U: $\sigma_R = 700$) を伴う場合における最適解の獲得	100
7.16	判定の信憑性における不確実性 (ケース U: $\sigma_R = 1500$) を伴う場合における最適解の獲得	100
7.17	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 200$) を伴う場合における最適解の獲得	101
7.18	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 700$) を伴う場合における最適解の獲得	101
7.19	判定の信憑性における不確実性 (ケース V: $\sup\{\sigma_R\} = 1500$) を伴う場合における最適解の獲得	101
7.20	入出力における不確実性の複合 ($(P_I, P_O) = (0.05, 0.15)$) を伴う場合における最適解の獲得	101
7.21	入出力における不確実性の複合 ($(P_I, P_O) = (0.05, 0.2)$) を伴う場合における最適解の獲得	101
7.22	入力における不確実性および判定の信憑性における不確実性の複合 ($(P_I, \sigma_R) = (0.03, 500)$) を伴う場合における最適解の獲得	104
7.23	入力における不確実性および判定の信憑性における不確実性の複合 ($(P_I, \sigma_R) = (0.1, 200)$) を伴う場合における最適解の獲得	104
7.24	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.1, 500)$) を伴う場合における最適解の獲得	104
7.25	出力における不確実性および判定の信憑性における不確実性の複合 ($(P_O, \sigma_R) = (0.2, 100)$) を伴う場合における最適解の獲得	104

7.26	入出力における不確実性および判定の信憑性における不確実性の複合 ((P_I, P_O, σ_R) = (0.03, 0.5, 200)) を伴う場合における最適解の獲得	108
7.27	入出力における不確実性および判定の信憑性における不確実性の複合 ((P_I, P_O, σ_R) = (0.03, 0.15, 500)) を伴う場合における最適解の獲得	108
8.1	判定の信憑性における不確実性 (指数分布: (λ, x) = (2, 100)) を伴う 場合における最適解の獲得	122
8.2	判定の信憑性における不確実性 (指数分布: (λ, x) = (1, 300)) を伴う 場合における最適解の獲得	122
8.3	判定の信憑性における不確実性 (指数分布: (λ, x) = (1, 400)) を伴う 場合における最適解の獲得	122
8.4	判定の信憑性における不確実性 (指数分布: (λ, x) = (1, -100)) を伴う 場合における最適解の獲得	122
8.5	判定の信憑性における不確実性 (指数分布: (λ, x) = (1, -500)) を伴う 場合における最適解の獲得	123
8.6	判定の信憑性における不確実性 (一様分布: $w = \pm 700$) を伴う場合 における正答率の平均および分類子数の平均	123
8.7	判定の信憑性における不確実性 (一様分布: $w = \pm 1500$) を伴う場合 における正答率の平均および分類子数の平均	123
8.8	判定の信憑性における不確実性 (一様分布: $w = \pm 2000$) を伴う場合 における正答率の平均および分類子数の平均	123
8.9	判定の信憑性における不確実性 (一様分布: $w = -2000$) を伴う場合 における正答率の平均および分類子数の平均	124
8.10	判定の信憑性における不確実性 (一様分布: $w = 500$) を伴う場合に おける正答率の平均および分類子数の平均	124
8.11	判定の信憑性における不確実性 (一様分布: $w = 700$) を伴う場合に おける正答率の平均および分類子数の平均	125
8.12	判定の信憑性における不確実性 (一様分布: $\sup\{w\} = 700$ (ケース V)) を伴う場合における正答率の平均および分類子数の平均	125
8.13	正規分布, 指数分布および一様分布が Iteration ごとに変わる場合に おける最適解の獲得	126
9.1	メカニズムと対応する不確実性	129
9.2	入出力における不確実性に対する XCS-ER の適用範囲	132
9.3	入出力における不確実性を伴う場合の XCS-ER の最適解の獲得試行数	132

9.4	判定の信憑性における不確実性（一様分布）に対する XCS-RR の適用範囲	135
9.5	判定の信憑性における不確実性（一様分布）を伴う場合の XCS-RR の最適解の獲得試行数	135
9.6	判定の信憑性における不確実性（正規分布）に対する XCS-RR の適用範囲	137
9.7	判定の信憑性における不確実性（正規分布）を伴う場合の XCS-RR の最適解の獲得試行数	137
9.8	$(7/8)^n$ の値	137
9.9	判定の信憑性における不確実性（正規分布）に対する XCS-EC の適用範囲	139
9.10	判定の信憑性における不確実性（正規分布）を伴う場合の XCS-EC の最適解の獲得試行数	139
9.11	入出力における不確実性に対する XCS-EC の適用範囲	141
9.12	入出力における不確実性を伴う場合の XCS-EC の最適解の獲得試行数	141
9.13	入出力における不確実性および判定の信憑性における不確実性を伴う場合に対する XCS-EC の適用範囲	143
9.14	入出力における不確実性および判定の信憑性を伴う場合の XCS-EC の最適解の獲得試行数	143
9.15	判定の信憑性における不確実性（正規分布）に対する XCS-RG の適用範囲	145
9.16	判定の信憑性における不確実性（正規分布）を伴う場合の XCS-RG の最適解の獲得試行数	145
9.17	入出力における不確実性に対する XCS-RG の適用範囲	147
9.18	入出力における不確実性を伴う場合の XCS-RG の最適解の獲得試行数	147
9.19	入出力における不確実性および判定の信憑性における不確実性を伴う場合に対する XCS-RG の適用範囲	149
9.20	入出力における不確実性および判定の信憑性を伴う場合の XCS-RG の最適解の獲得試行数	149

第 1 章

序論

1.1 背景

産業に限らず様々な分野において、現在の状況を把握しそれに対して適した選択を取ることが求められる。特に人間にとって困難なデータのインスタンス数や属性の種類が多い問題において機械学習手法を用いて対応することが期待されている。その対象として医療、金融や交通など人の命や財産に関わる分野も含まれ、それらの分野において正答率などの精度が求められることも然る事ながら、なぜそのような判断がされたのかを人間が理解できる様に明らかにする説明可能性および学習データに含まれるどのデータに対しても適応可能な網羅性が求められる。特にどの属性が判断に寄与したかを明らかにすることは重要である [1]。このようなデータベースから人間が理解可能なモデルである知識の導出を Knowledge Discovery from Databases (KDD)[2, 3] と呼ぶ。KDD のプロセスはデータ集合からの目的データの抽出、ノイズや外れ値の除去などの前処理、属性の作成や選択の変換、パターンの抽出、解釈および評価からなる。

KDD で扱う分類問題に対してよく用いられる手法として、深層学習 [4]、線形・非線形分類器（例: サポートベクターマシン [5]）、決定木の組み合わせ（例: ランダムフォレスト [6] や eXtreme Gradient Boosting[7]）および IF-THEN ルールの組み合わせ（相関ルール分析 [8]）がある。それぞれの手法を、分類の誤りの少なさを表す正しさ・適応性と分類器あるいは知識の簡潔さを表す解釈性の観点においてプロットすると図 1.1 が得られる。解釈性は人間が理解できるかを指標とするのが理想的であるが、人によって経験や理解度が異なり適切に判断することが難しいため、一般的に簡潔なルールであるほど理解しやすいので分類器あるいは知識の簡潔さを指標とした。深層学習および線形・非線形分類器はネットワークあるいは入力 \mathbf{x} を生成した 1 つの関数 $f(\mathbf{x})$ からなる分類器に入力して出力を得る。分類器 1 つで全ての入力に対応するため分類器の構造が複雑になり、判断根拠を求める際には係数の大きさの比較やニューロンの重みから入力の情報がどのようにニューロンをたどるかを追跡する必要があるため、一般の人が深層学習の判断根拠を理解

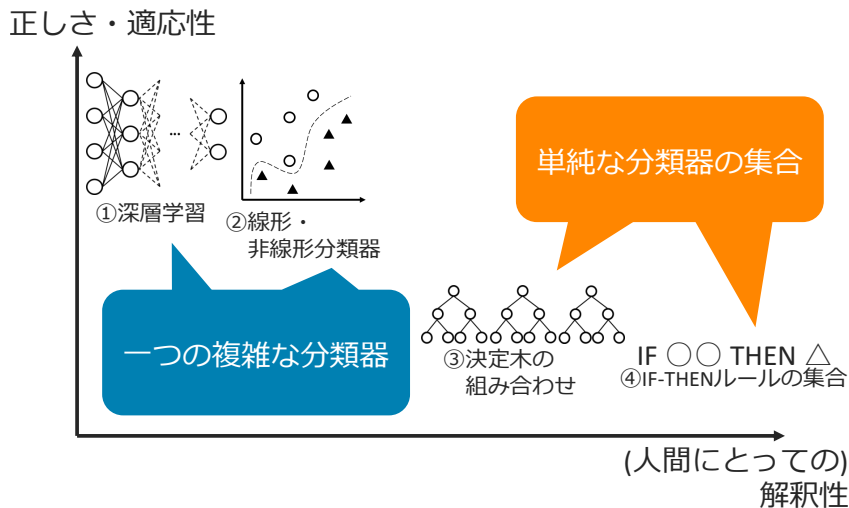


図 1.1 分類問題に用いられる手法



図 1.2 分類子の一般化の例

することは難しい。決定木あるいは IF-THEN ルールを組み合わせた手法において構成する分類器ひとつひとつは比較的簡潔であるが、分類器の数が膨大でありそれぞれの分類器の出力した結果の組み合わせ、あるいは尤も良く照合するルールによって出力が決定されるため、判断根拠を求めるためには構成する分類器全てを分析する必要がある。また、決定木では最初の数個の分岐条件のみを取り出すことにより簡潔なルールを得ることができるが、網羅性が低くなる問題がある。いずれの手法も入力 x においてどの属性が出力に寄与するかという知識を獲得する事は難しい。

これらのことから人間が容易に解釈するためには獲得される知識は少数で単純な構造の分類器の集合である必要がある。本研究では一般化と呼ばれる機構によって複数の入力に照合する IF-THEN ルールを生成可能な学習分類子システム (Learning Classifier System: LCS)[9] に着目する。LCS において IF-THEN ルールは分類子と呼ばれる。図 1.2 に示す様に、一般化された分類子は入力の特定の属性に照合する部分を “don't-care” を表す記号 # に置き換えられており、その属性が如何なる値であっても照合するため、複数の入力に照合可能である。この図の例ではそれぞれの分類子の真ん中の 3 つの属性の値が共通しているが、両端の属性の値は共通していない。そのため、両端の属性は記号 # に置き換えられる。また、それぞれの分類子は重複がない様に一般化が進められ、複数の分

類子の集合によって全ての学習データに対応を図るため、網羅性が高い特徴がある。LCSの中でも一般化能力が高いため主流である正確性に基づく LCS(Accuracy-based LCS: XCS)[10] は出力の評価が常に同一となる分類子を正確として獲得する。XCS で獲得される分類子集合は一つ分類子で複数の入力に照合可能なことに加え、# 記号に置き換えられていない属性が出力の評価に寄与する属性であると分かるため、これまでに紹介した機械学習手法の中でも解釈性が高い手法である。

しかしながら、実問題においては様々な不確実性を伴うことが考えられるため、XCS の適用が難しいという問題がある。それは不確実性を伴うことで入力に対する出力の評価が安定しないためである。評価が安定しなくとも有用な知識は存在し、それを獲得することは有用である。例えば、天気データで雨が降ったときの知識だけでなく、(雨は降らない場合もあるが) 降水確率が高いときの知識も有用である例が挙げられる。しかしながら、たった一つでも異なる評価が混在すると分類子は不正確と判定され削除対象となる。出力の評価が安定しない場合には評価が同じ分類子の共通する属性を発見することができないため、分類子の評価を正確性に基づいて行う XCS は正しく学習ができず、一般化された分類子を生成できない。生成された分類子は獲得する評価が安定せず様々な値を取るため不正確と判断され保持されない。さらに一般化された分類子が生成されたとしても、一般化度合が高いつまり # 記号が多いほど多くの入力に照合するため、本来とは異なる評価を受ける可能性が高くなる。一般化されるほど分類子が保持されない対象になる可能性が高くなるため分類子の一般化が進まず、XCS が獲得する集合の分類子数は増大し、解釈性が低下する。

学習データにおける不確実性に対処するために前処理を行うことが考えられるが、同一のデータにおける評価の違いやその分布を調べる必要があり、正確性の基準を満たす様に同一の評価となる様に調整する必要がある。不確実性の種類は分布が少数の値のみで構成されているかそうで無いかによって知ることができるが、その強度については正確に把握することは難しい。これには膨大なコストがかかり、データ数や属性の数が増えるほど増大するため、XCS の活用に対する障害となる。また、不確実性の影響を受けたとみなす対象を決めることが難しく、前処理によって不確実性を取り除こうとしても完全に取り除けない場合がある。そのため前処理に頼らずかつ不確実性に関する情報(不確実性の種類とその強度)無しに、XCS 自体に不確実性に対処する機構を導入する必要がある。

1.2 目的

前節の背景より、本研究では不確実性を伴う環境においても一般化された分類子を獲得する学習分類子システムを提案し、その有効性を不確実性を模擬したノイズを付加した Multiplexer 問題に適用することにより検証することを目的とする。Multiplexer 問題は一般化性能を測ることができるため LCS の研究においてよく用いられている。具体的に

は、(1) 不確実性を考慮した正確性を測定する指標の考案とその基準における正確性を判定する基準の動的な設定、(2) 正しく一般化された分類子の生成および保持が可能なシステムを探求する。

1.3 章立て

本論文の構成は以下の通りである。第 1 章で本研究における背景として分類問題に用いられる手法を紹介し、その中で解釈性の高い知識を獲得する LCS を取り上げた。さらに、データに不確実性を伴った場合の知識獲得における影響の概要を述べ、本研究における目的を記した。第 2 章では本研究が扱う不確実性について説明する。不確実性の種類を (i) 入出力における不確実性と (ii) 判定の信憑性における不確実性に分け、それぞれの学習に対する影響を述べるとともに、不確実性を伴うデータに対する従来の対処法について述べる。第 3 章では学習分類子システム (Learning Classifier System: LCS) のフレームワークと LCS の分類について述べ、LCS の中でも分類子の一般化性能が高い正確性に基づく学習分類子システム (Accuracy based LCS: XCS) の学習メカニズムについて説明する。第 4 章では本研究で扱う LCS のベンチマーク問題である Multiplexer 問題について説明し、本研究で扱う (i) 入出力における不確実性および (ii) 判定の信憑性における不確実性を Multiplexer 問題上で模擬する方法および学習の成否を判断する評価基準について述べる。

第 5 章では (i) 入出力における不確実性に対応可能な学習分類子システムである推測報酬に基づく学習分類子システム (XCS based on Estimated Reward: XCS-ER) を提案し、XCS からの変更点である報酬推測機構とそれに関連するパラメータ更新および包摂判定について説明する。さらに、XCS と XCS-ER を (i) 入出力における不確実性を伴う Multiplexer 問題に適用した際の実験を行いとその結果から XCS-ER の有効性について述べる。第 6 章では判定の信憑性における不確実性に対応可能な学習分類子システムである報酬の範囲に基づく学習分類子システム (XCS based on Range of Reward: XCS-RR) を提案し、XCS からの変更点である正確とみなす分類子の判定基準とそれに関連するパラメータ更新および包摂判定について説明する。さらに、XCS と XCS-RR を (ii) 判定の信憑性における不確実性を伴う Multiplexer 問題に適用した際の実験を行いとその結果から XCS-RR の有効性について述べる。第 7 章では本研究で扱う (i) 入出力および (ii) 判定の信憑性における不確実性に対応可能な学習分類子システムである報酬の組分けに基づく学習分類子システム (XCS based on Reward Group: XCS-RG) および正誤推測に基づく学習分類子システム (XCS based on Estimated Correctness: XCS-EC) を提案し、XCS-RG の XCS からの変更点である Recording table とそれに関連するパラメータ更新および包摂判定と XCS-EC の XCS からの変更点である報酬推測機構とそれに関連するパラメータ更新および包摂判定について説明する。さらに、XCS と XCS-RG および

XCS-EC を (i) 入出力および (ii) 判定の信憑性における不確実性を伴う Multiplexer 問題に適用した際の実験を行いとその結果から XCS-RG および XCS-EC の有効性について述べる。

第 8 章では実問題を想定して複数種類の不確実性を同時に伴った場合および判定の信憑性における不確実性が正規分布以外の分布である場合において、第 5 章から第 7 章までで扱ってきた提案メカニズムの適応性について述べる。第 9 章では 4 種類のメカニズムを基にした不確実性に対応可能な学習分類子システムの設計論および提案メカニズムの解析的適用範囲について述べる。第 10 章では本論文のまとめを述べ、今後の課題と展望を示す。

第 2 章

対象とする不確実性

実問題においては様々な不確実性を伴うことが考えられる。分類問題においては入力と出力およびその評価があるため、それぞれにおける不確実性を入力における不確実性、出力における不確実性および判定の信憑性における不確実性とする。以降は天気図からの天気の予報を例に取りそれぞれの不確実性を説明する。

2.1 不確実性の分類

2.1.1 入力と出力における不確実性

入力における不確実性の例として画像におけるノイズが挙げられる。図 2.1 の例ではある地点においてデータが欠測し得られない場合が該当する。ノイズがのると判定の根拠となる部分が隠れ、判定が異なるデータにおいて異なるべき部分が同じと認識されたり、同じと認識されるべき部分が異なると認識される場合がある。そのため、入力における不確実性が伴うと判定が同一なデータにおける共通部分を見つけることが困難になる。さらにノイズが多い場合には、異なる評価のデータと同一の入力と認識され、複数の評価（正解/不正解）が混在するため、この入力に照合する分類子は正解と不正解が混在する。正解と不正解が混在する分類子は判定の結果を予測するための知識として価値が低いいため削除対象となる。維持すべき分類子が削除されるため、入力における不確実性は学習に影響を及ぼす。

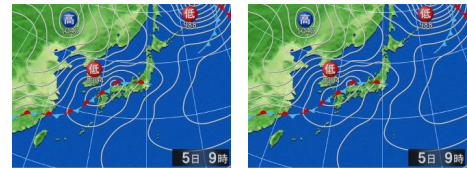
出力における不確実性の例として判定の誤りが挙げられる。図 2.2 の例では同じ画像にもかかわらず、左側では曇りと判定され、右側では雨と判定されている。これは同一の入力に対して異なる人が判定する場合や判定に重要な要素が入力データに含まれていない場合に起こる。同一の入力に対して異なる判定が混在するため、このデータに照合する分類子は正解と不正解が混在する。入力における不確実性を伴う場合と同様に、正解と不正解が混在する分類子は判定の結果を予測するための知識として価値が低いいため削除対象とな



×日
雨

△日
雨

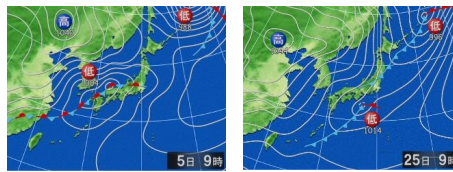
図 2.1 入力における不確実性の例



×日
曇り

×日
雨

図 2.2 出力における不確実性の例



□日 ×日

湿度: 80% 湿度: 70%

図 2.3 判定の信憑性における不確実性の例

る。維持すべき分類子が削除されるため、出力における不確実性は学習に影響を及ぼす。

2.1.2 判定の信憑性における不確実性

判定の信憑性における不確実性の例として天気の種類を断言するのではなく湿度や降水確率を示す場合を挙げる。分類やその評価に多大なコストがかかる場合に、学習に使われるデータが実際に判定されるのではなく、簡易的なものとして判定の確信度（正解の確率）を代わりに付与することが考えられる。図 2.3 の様に同様な入力であったとしても、入力データに含まれない情報や個人差などの影響や評価者の違いにより、その値が異なることがある。評価の値が微小に異なっても評価が高い傾向のデータあるいは評価が低い傾向のデータに共通する属性の値が分かれば、知識としては有用である。しかし、一般的には不確実性を伴う原因やその頻度（確率）が分からず、事前にばらつきに対する許容値を設定することは困難である。値が異なると具体的な値が予測できないため、XCS では不正確な分類子と判定され、評価が安定しない分類子として削除される。維持すべき分類子が削除されるため、判定の信憑性における不確実性は学習に影響を及ぼす。

入力における不確実性および出力における不確実性を伴う場合においては、維持すべき分類子であっても誤判定により正解と不正解が混在すること、判定の信憑性における不確実性を伴う場合においては、評価が数値で表されその値が変動し安定しないという違いが

ある。これらの特徴から (i) 入出力における不確実性および (ii) 判定の信憑性における不確実性の正答率への影響の観点から分類する。データの入力と取得におけるエラーが不確実性の要因として割合を大きく占めており、このエラーに対する特別な対策が取られていない場合 5% 程度のエラーが生じる [11, 12, 13] と言われているので、不確実性を伴うデータから安定的に知識を獲得する手法は必要である。

2.2 不確実性への従来対処法

不確実性への対処は前処理による学習データを整備する方法と学習手法を不確実性に対処可能にする方法に大別される。以下にそれぞれ詳しく述べる。

2.2.1 前処理による対処

不確実性を伴うデータに対処するためには一般的に前処理による不確実性を伴うデータの除去が行われる [14]。しかし、不確実性を伴うデータを除去するためにはデータにどのような種類と強度の不確実性が伴うのかを正確に把握する必要があり、多大なコストがかかる。データに伴う不確実性全てを正確に把握できず、学習データの中に不確実性を伴うデータが残ったままの場合がある。うまく不確実性を伴うデータを除去できたとしても、学習データが減るため、分類器の性能が低下する場合は考えられる。

2.2.2 不確実性に対応を図った学習手法

不確実性に対応を学習手法を表 2.1 に示す様に、(i) 入出力における不確実性のみに対応を図った学習手法、(ii) 判定の信憑性における不確実性のみに対応を図った学習手法、(i) および (ii) の両方の不確実性に対応を図った学習手法に分けて紹介する。それぞれにおいて LCS での対処法を述べた後、他の手法での対処法を述べる。表は以下で取り上げる手法の中から代表的なものを取り上げ、列に入出力における不確実性への対応の可否を、行に判定の信憑性における不確実性への対処の可否を取り、配置したものである。

(i) 入出力における不確実性のみに対応を図った学習手法

Lanzi らは入力における不確実性への対処を図る LCS として XCSMH を提案した。[15] 図 2.4 の下側の様にスタート S からゴール G を目指す迷路問題を周囲 8 近傍を観測可能なエージェントで解くことを考える。ここで T は障害物の木を表す。エージェントは矢印で示された 2 地点において図 2.4 の上側の様に同一の入力として認識する。この 2 地点においてゴール G に向かう向きが異なるため、同一の入力として認識すると最適と考えられる向きに進んだとしてもゴールする場合とできない場合が混在する。2 地点を識

表 2.1 不確実性に対応を図った学習手法

		入出力における不確実性	
		無し	有り
判定の 信憑性 における 不確実性	無し	/	<ul style="list-style-type: none"> ・ LCS 前状態の情報付加により識別[Lanzi, 99a] 評価のばらつきを把握し許容[Lanzi, 99b] ・ NN, SVM, RF 一度モデルを作り誤ったデータを除く [Lee, 18], [Zhang, 18], [Khoshgoftaar, 04], [Gil, 19]
	有り		<ul style="list-style-type: none"> ・ LCS 複数の評価を数値化しばらつきを閾値で許容 [Nazmi, 18] ・ 線形回帰 サンプリングにより評価のばらつきの影響軽減 [Granichin, 02], [Digailova, 00]

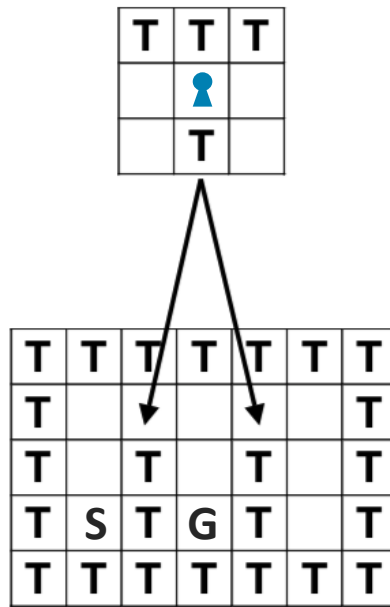


図 2.4 非マルコフ過程の迷路問題

別するために図 2.5 で示す様に直前の入力（認識）と直前の出力（移動方向）を内部状態として保持し、今回の入力と組み合わせることで全ての地点の識別を可能にする手法を提案した。ただし、この手法は迷路問題を始めとする時系列問題にのみ有効であり、本研究で想定する分類問題で用いることはできない。

Lanzi らは出力における不確実性への対処を図る LCS として XCS μ を提案した。[16] スタート S からゴール G を目指す迷路問題を考える。図 2.6 で示す様に下向きに進めば

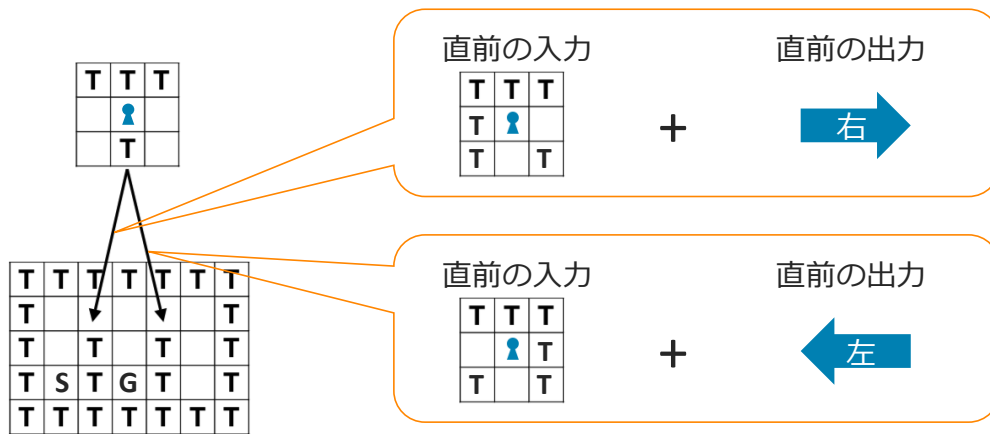


図 2.5 内部状態による入力の識別

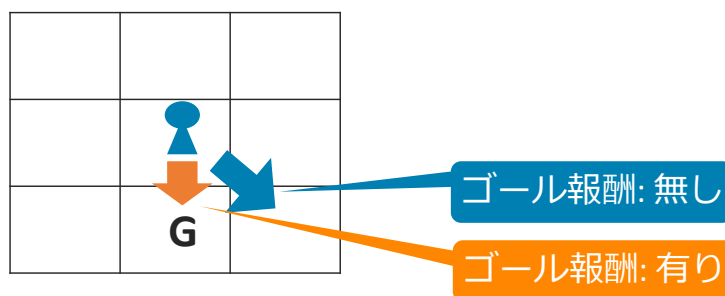


図 2.6 スリップする迷路問題

ゴールできる場合に下に進もうとしているにもかかわらず、確率的に異なる方向に進む環境ではゴールする場合とできない場合が混在する。予測した評価と実際の評価の差分の最小値の分だけその差分の許容値を大きくして、出力（移動方向）が異なる場合がある環境で獲得した分類子の正確性を正しく判定する手法を提案した。ただし、この手法では予測した評価と実際の評価の差分が直近の評価による影響が強く、不確実性の頻度の偏りによって差分の値が変動しやすく、安定して分類子の安定性を評価できない問題がある。

不確実性を伴う分類問題において正答率のみを高めることにより対応する方法は (a) フィルターによって不確実性の影響を受けたデータを取り除き学習データの質を向上する方法、(b) 不確実性をモデル化して学習モデルと分離する方法の 2 つに大別される [14]。それぞれの方法を用いた LCS 以外の手法について説明する。

- (a) フィルターによって不確実性の影響を受けたデータを取り除き学習データの質を向上する方法
 - (a) の方法は、(a-1) 閾値によって不確実性の影響を受けたデータを識別する方法、(a-2) モデルベースの不確実性の影響を受けたデータを識別する方法および (a-3) データの分布による不確実性の影響を受けたデータを識別する方法に分か

れる。まずは (a-1) 閾値によって不確実性の影響を受けたデータを識別する方法である。誤って分類されたデータは条件付きエントロピーが高いため、閾値を上回った際に条件付きエントロピーが低いクラスにラベルを付け替える方法 [17] や誤って分類されたデータを用いて学習すると決定木ではサイズが大きくなるなど [18, 19, 20, 21] 学習モデルは複雑になるため、一部の学習データを抜いて再学習した際に学習モデルの複雑さが低下した場合に抜いたデータを取り除く方法がある [22, 23, 24, 25]。次に (a-2) モデルベースの不確実性の影響を受けたデータを識別する方法である。この方法では一度学習データを全て用いて学習モデルを作り、そのモデルにおいて誤って分類されたデータを学習データから取り除く。SVM を学習モデルとして用いたもの [26]、ニューラルネットワークを学習モデルとして用いたもの [27]、ランダムフォレストや k 近傍法など複数種類の手法を組み合わせたものを学習モデルとして用いたもの [28]、適応サンプリングを学習モデルとして用いたもの [29] がある。さらに決定木の C4.5 手法では誤ったデータが分類されたノードを枝刈りして、データの削減とモデルの複雑さの低減を同時に図っている [30]。また、ローカルモデルという k 近傍のデータのみを用いて作成した SVM の学習モデルを用いて同様に誤って分類されたデータを取り除く手法 [31, 32, 33] や学習データを k 個に分けてそれぞれのデータで学習したモデルの投票によって除くデータを決定する方法 [34, 35, 36] および誤って分類されたデータを識別するためのニューラルネットワークを構築する手法 [37, 38] もある。そして、(a-3) データの分布による不確実性の影響を受けたデータを識別する方法である。k 近傍法によって多数決を行い少数派のデータを削除する方法 [39] や分類の境界上のデータは誤る確率が高いため取り除く方法 [40] がこれに当たる。

- (b) 不確実性をモデル化して学習モデルと分離する方法

(b) の方法は (b-1) 確率的手法を用いた不確実性のモデル化と (b-2) モデルベースの不確実性のモデル化に分かれる。まずは (b-1) 確率的手法を用いた不確実性のモデル化である。誤って分類される確率をベイジアンアプローチを用いてモデル化する方法 [41, 42, 43] や分類クラスを用いずにクラスタリングを行い割り当てられたクラスタ内のクラスを用いる方法 [44] がある。次に (b-2) モデルベースの不確実性のモデル化である。ランダムフォレストにノイズ耐性関数によってノイズ情報を組み込む方法 [45] や SVM において誤って分類されたデータにヒンジ損失でペナルティを与える方法 [46] やニューラルネットワークにおいて入トリックによって誤って分類されたデータの影響を減らす方法 [47, 48] がある。



図 2.7 判定の信憑性における不確実性を伴うデータにおける分類子の一般化

(ii) 判定の信憑性における不確実性のみに対応を図った学習手法

Nazmi らは判定の信憑性における不確実性への対処を図る LCS として PRBF を提案した。[49] 分類問題において答えとなるクラスが与えられるのではなく、クラスへの分類確率が示される問題を考える。分類問題において正解となるクラスを推測するのではなく、出力毎に正解となる確率を推測することで判定の信憑性における不確実性に対応を図る手法を提案した。図 2.7 に示す様に、正解の確率が 90% と 100% に近い分類子で一般化し、正解の確率が大きく異なる値である 70% の分類子とは一般化を行わない。ただし、この手法を用いて分類子を一般化するためには、近いとみなす確率の許容範囲を事前に定める必要がある。この許容範囲を小さく設定すると分類子の一般化が進まず、大きく設定すると過剰に一般化されるため、慎重に設定する必要がある。

LCS 以外の手法として線形回帰においてノイズのデータをフィルタリングによって除く手法もある [50, 51]。ただし、フィルタリングによる手法は学習に用いても性能の悪化を引き起こさないデータを過剰に削除し適用性が低下する場合がある [14]。

(i) および (ii) の両方の不確実性に対応を図った学習手法

両不確実性に対応を図った LCS は、正答率を高めることを目的とした手法である。正解率を高めるためには必ずしも正確な分類子を獲得する必要は無い。不正確であっても照合する入力と出力の組における正解と不正解の割合が偏っていれば、獲得する報酬の平均値や分類子が持つ期待報酬値 p の大小関係が大まかには合うため正解となる出力を選ぶことができる。そのため、高い正解率を実現しても正確かつ最大限に一般化された分類子が得られることは少なく、知識獲得の観点では向かない手法である。

正解率を高めるために学習した LCS を複数組み合わせるアンサンブル手法を取り入れたものがある。複数の LCS を学習させる際に、それぞれの LCS に学習データとして渡す情報を分割する。情報の分割法として属性を分割する垂直分割とデータを分割する水平分割がある。垂直分割によるアンサンブル LCS は Gershoff らの [52], Abedini らの [53], Zhu らの [54], Debie らの [55] があり、水平分割によるアンサンブル LCS は Liu らの [56, 57], Gao らの [58, 59, 60], Rojanavasulu らの [61], Lanzi らの [62, 63] がある。これらの手法は学習データの一部の情報を用いて LCS を学習させるため、学習データ全体に

対する最適に一般化された分類子を獲得することができない場合がある。垂直分割も水平分割も情報が少なく、与えられた学習データにおいて最適に一般化され正答率が高い分類子が獲得されるため、学習データ全体が与えられた場合に対して過剰に一般化された分類子が獲得される傾向が強い。

LCS 以外の手法として線形回帰において信号を正規化することによりノイズの影響を少なくし、最小平均自乗法、再帰最小自乗法やカルマンフィルターベースの適応アルゴリズムの指数安定性を保証し性能を安定させる手法もある [64, 65, 66, 67]。

2.3 本研究の不確実性へのアプローチ

従来の不確実性への対応を図った手法は、データに伴う不確実性の種類だけでなくその強度を学習前に把握する必要がある、不確実性を除くためのモデルを学習用のモデルとは別に生成する、分類器が学習データ全てを網羅できない場合がある問題がある。そこで本研究では不確実性の強度を把握せず、不確実性を除くためのモデルの生成が不要で、学習データ全てを網羅したうえで、不確実性に対応を図る以下のアプローチをとる。

(i) 入出力における不確実性を伴うと正確な分類子であっても正解の評価と不正解の評価が混在するため、その時点の入出力の組み合わせ（入出力対）における分類子が獲得した最も回数の多い評価を採用するアプローチにより (i) 入出力における不確実性に対応を図る XCS-ER を提案する。(ii) 判定の信憑性における不確実性を伴うと獲得する報酬の値がばらつくため、分類子が獲得した報酬値をクラスタリングによって正解と不正解の 2 クラスに分けるアプローチにより (ii) 判定の信憑性における不確実性に対応を測る XCS-RR を提案する。それぞれのアプローチは他方の不確実性には対応ができないため、それぞれの要素を組み合わせ (i) および (ii) の不確実性の両方に対応を図る。一つ目は不確実性を伴った場合でも入出力対における分類子が獲得した報酬から求める期待値（平均値）のクラスタリングから正しい正誤評価を推測するアプローチによって不確実性に対応を図る XCS-EC を、二つ目は一般化の影響がない入出力対のクラスタリングによって求めた正誤評価の組み合わせによって分類子の正確性を判定するアプローチによって不確実性に対応を図る XCS-RG を提案する。図 2.8 に本研究の不確実性へのアプローチを示す。左上は XCS-ER を右上は XCS-RR のアプローチを示しており、それぞれのアプローチを組み合わせ両方の不確実性に対応可能なアプローチをとる XCS-EC および XCS-RG をそれぞれ左下と右下に示す。

提案手法の対応可能な不確実性をまとめると表 2.2 となる。○ は考慮・対応を示し、× は未考慮・未対応を示す。XCS-ER および XCS-RR は (i) 入出力における不確実性あるいは (ii) 判定の信憑性における不確実性のみに対応を図るが、XCS-EC および XCS-RG は両方の不確実性に対応を図るため、両方の不確実性が複合した場合にも対応を図る。

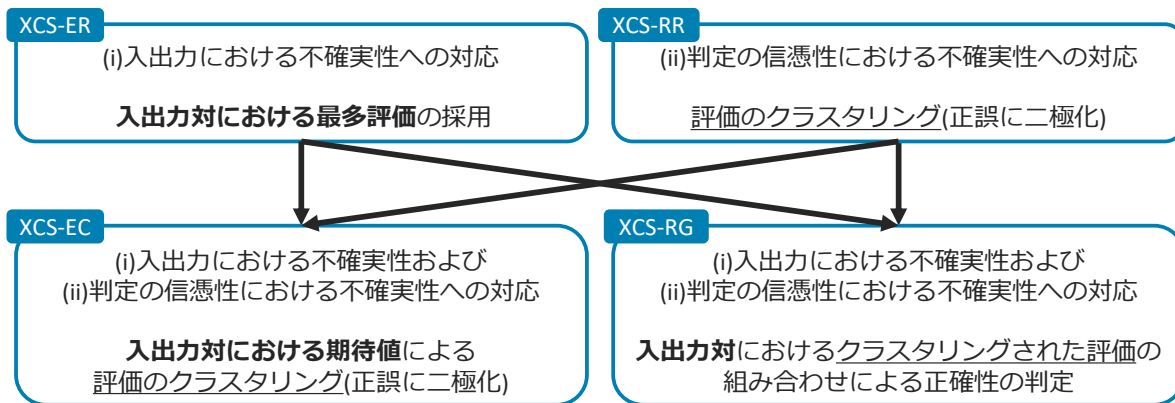


図 2.8 本研究の不確実性へのアプローチ

表 2.2 アプローチと対応する不確実性

アプローチ	(i) 入出力における不確実性	(ii) 判定の信憑性における不確実性	(i)+(ii) 複合した不確実性
入出力対における 最多評価の採用 (XCS-ER)	○	×	×
評価のクラスタリング (XCS-RR)	×	○	×
入出力対における 期待値による 評価のクラスタリング (XCS-EC)	○	○	○
入出力対における クラスタリングされた 評価の組み合わせによる 正確性の判定 (XCS-RG)	○	○	○

第 3 章

学習分類子システム

LCS は入力に対して正解となる出力を予測する分類問題に用いられる手法であるとともに、分類子を一般化することにより正解の出力を予測する分類子間あるいは不正解の出力を予測する分類子間で共通の属性を発見する。

3.1 フレームワーク

学習分類子システム (LCS) は強化学習 (Reinforcement Learning : RL)[68] と遺伝的アルゴリズム (Genetic Algorithm : GA)[69] を組み合わせた環境適応システムである。図 3.1 に一般的な学習器と LCS のフレームワークを示す。一般的な学習器ではデータセットから入力とその入力において正解となる出力の組が渡される。一方で LCS はデータセットと出力の正誤判定を行う機構をセットにして環境と呼び、環境からの入力に対し出力を返す。その出力が環境に評価され報酬として LCS に渡される。正解あるいは不正解の行動を出力した際にそれぞれ報酬 $R, 0$ を獲得する。一般的には R は 1000 と設定されることが多い。

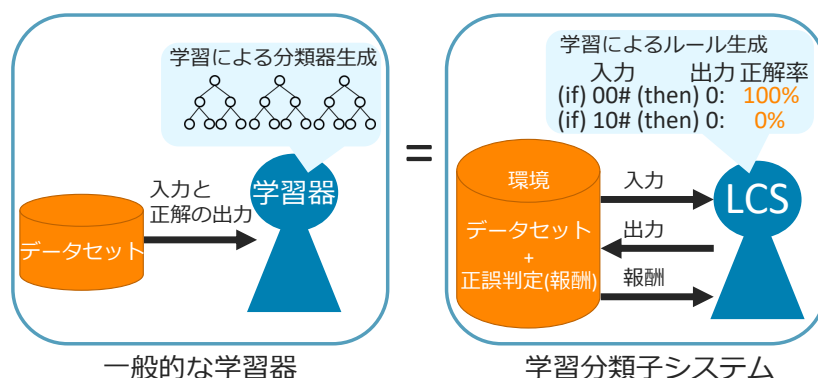


図 3.1 学習フレームワーク

3.2 学習分類子システムの分類

3.2.1 個体表現による分類

学習分類子システムは Holland の提案した Cognitive System[70] を端に発し、個体表現の違いからミシガンスタイル [71] とピッツバーグスタイル [72] に大別される。ミシガンスタイルは一つの分類子が一つの個体（解の一つ）として扱われるのに対し、ピッツバーグスタイルは複数の分類子からなる分類子集合が一つの個体として扱われる。ミシガンスタイルの特徴は一つの解である分類子が照合する範囲を柔軟に変更でき表現力が高いため、適用可能な問題の範囲が広いことである。ミシガンスタイルの例として XCS や sUpervised Classifier System (UCS)[73] や Zeroth-level Classifier System (ZCS)[74] などがある。一方でピッツバーグスタイルの特徴はそれぞれの個体において多数のインスタンスに照合することで得られる全体的なパフォーマンスによって評価されるオフラインバッチ処理のため、最適なルール数が分かればコンパクトな解集合を求めることができデータマイニングや知識発見に向いていると言われている [75]。ピッツバーグスタイルの例として GABL(GA Batch concept Learner)[76] や GAssist[77] や BioHEL[78] などがある。

本研究では未知の不確実性を伴う問題における知識発見を目的としており、得られる結果の最適性よりも不確実性への適応性の高さを重視するため、以降ではミシガンスタイルについて扱う。

3.2.2 学習戦略とルール評価戦略による分類

学習分類子システムは学習戦略とルール評価戦略によっても分類できる。学習戦略ではベストアクションマップ [74] とコンプリートアクションマップ [10] の 2 種類がある。ベストアクションマップは与えられた入力に対して正解となる行動つまり獲得報酬の期待値が一番大きくなる行動のみを選択し学習する戦略である。ベストアクションマップの例として ZCS および UCS がある。一方でコンプリートアクションマップは獲得報酬の期待値によらず行動を（ランダムで）選択し、全ての入力と出力の組み合わせを網羅する戦略である。コンプリートアクションマップの例として XCS がある。コンプリートアクションマップは全ての入出力の組み合わせを網羅する様に学習するためベストアクションマップに比べて学習に要するデータ数が多くなるが、ベストアクションマップでは学習に含めない情報も保持するため学習の安定性が高い。

ルール評価戦略には強度値ベース [74] と正確性ベース [10] の 2 種類がある。強度値ベースでは獲得報酬が高い分類子ほど良いと判断され保持される。強度値ベースの例とし

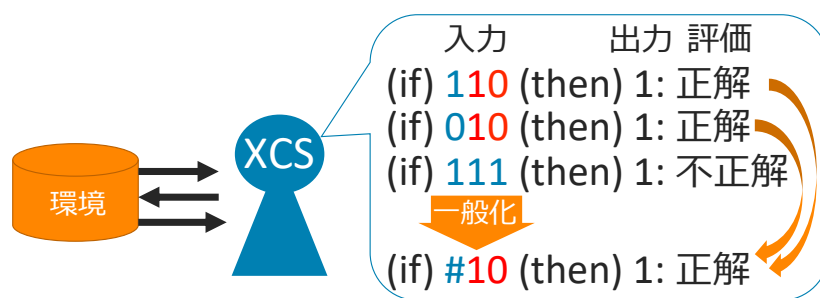


図 3.2 LCS の分類子の一般化の概要

て ZCS がある。正確性ベースでは獲得報酬の高低ではなく同値性を分類子の評価とする。獲得報酬の平均値が高くてそれらの値にばらつきがある分類子は、獲得報酬の平均は低いにばらつきが全くない分類子に比べて悪い分類子と評価され保持されにくくなる。正確性ベースの例として XCS や UCS がある。正確性ベースの評価は報酬の同値性を重視するため、獲得した分類子から導出される知識は特徴量の報酬への寄与の有無を表すため、理解がしやすいものとなる。また、正確性ベースの方が強度値ベースのものよりも分類子の一般化性能が高いと言われている [79]。

3.2.3 対象とする問題による分類

上記で取り上げた手法は分類問題に適用されるが、他にも回帰問題への適用を目的とした XCSF [80, 81] や多目的強化学習問題への適用 [82, 83, 84] がある。一般に、分類問題は離散的で入力に対する正解となる出力が定まるため一般化された分類子を獲得しやすい。一方で回帰問題は連続値の入力に対し連続値の出力を求める必要があること、多目的強化学習問題は目的が多数なため問題の構造が複雑で、一般化された分類子つまり解釈性の高い知識を獲得しにくく多くの分類子が獲得される。

3.3 正確性に基づく学習分類子システム

本研究では実問題からの知識獲得に向けて、適用可能な問題の範囲が広いミシガンスタイルであり、学習の安定性が高いコンプリートアクションマップであり、一般化性能が高い正確性ベースである XCS に着目する。

図 3.2 に示す様に、入力に対する出力の評価である正解あるいは不正解を把握し、同一の評価となる入力と出力を組み合わせることで一般化された分類子の獲得を目指す。図 3.3 において、上の例では正解の分類子同士を一般化するので一般化された分類子は正確と判定され、真ん中の例では不正解の分類子同士を一般化するので一般化された分類子は正確と判定されるが、下の例では正解と不正解の分類子で一般化するため一般化された分類子は

入力	出力	正解率		入力	出力	正解率	正確性
(if) 000 (then) 0:		100%	→ 一般化	(if) 00# (then) 0:		100%	○ 正確 (常に正解)
(if) 001 (then) 0:		100%		(if) 1#1 (then) 0:		0%	○ 正確 (常に不正解)
(if) 101 (then) 0:		0%	→ 一般化	(if) #0# (then) 0:		50%	✕ (正解/不正解混在)
(if) 111 (then) 0:		0%		(if) 000 (then) 0:		100%	
(if) 101 (then) 0:		0%	→ 一般化				

図 3.3 分類子の正確性

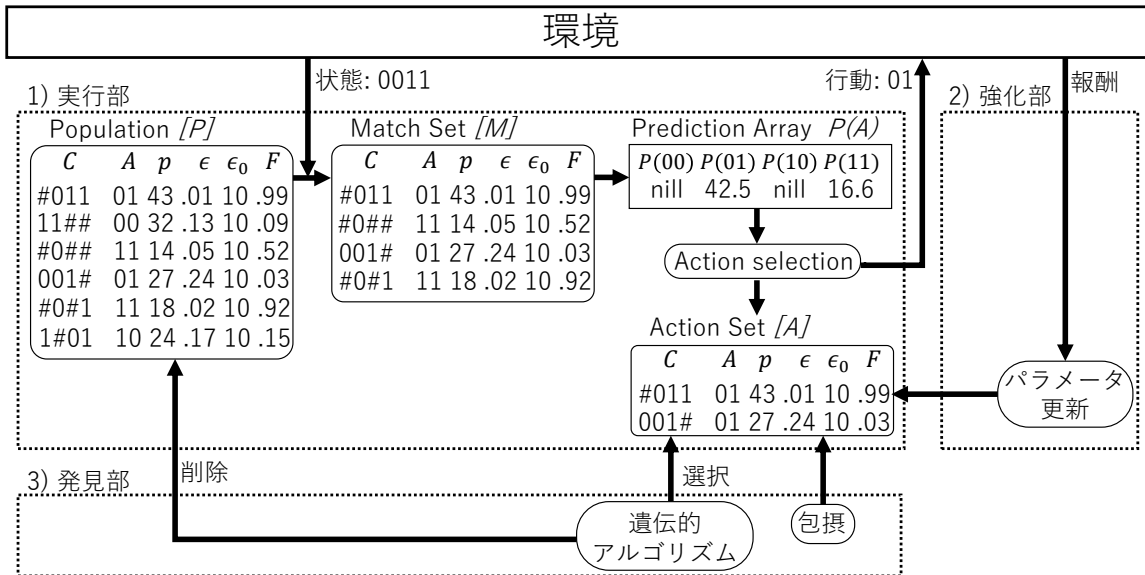


図 3.4 XCS のアーキテクチャ

不正確と判定される。

3.3.1 構成

アーキテクチャ

XCS の以下に示す学習の 1 エピソードを繰り返し一般化された正確な分類子を獲得する。1 エピソードは図 3.4 に示す様に、1) 実行部、2) 強化部、3) 発見部の 3 機構が順に実行され、分類子集合 (Population: [P]) を更新していく。図の例では状態 “0011” が入力され [P] から照合する分類子を抽出し、照合集合 (Match Set: [M]) が生成される。[M] の分類子を行動ごとに分け予測報酬値 P を計算し、出力する行動を決める。この例では出力が “01” となったため、[M] から出力が “01” の分類子を抽出し行動集合 (action set: [A]) を作る。[A] の分類子を対象に獲得報酬に基づくパラメータ更新が行われ、その結果が [P] に反映される。

分類子

XCS の分類子 cl は IF-THEN ルールとそのルールを評価するために必要となるパラメータ群から構成される。入力および出力は一般に 0 または 1 からなるビット列で構成されるため、IF-THEN ルールの条件部 (condition: C) は入力ビット長 L に合わせた $C \in \{0, 1, \#\}^L$ の形式を取る。ただし、 $\#$ は任意 (don't care) の値を表す記号であり、これを組み込むことにより複数の入力状態に照合する条件部を生成することが可能である。例えば、“10###” なる条件部は “10000”, “10001”, “10010”, “10011”, “10100”, “10101”, “10110”, “10111” の 8 通りのビット列を表現する。さらに IF-THEN ルールの行動部 (action: A) は取りうる行動の値の中から一つのみを取ることができる。また、パラメータ群は IF-THEN ルールを実行した際に獲得される報酬の予測値 (prediction: p), その p からの実際に獲得した報酬 ρ からの誤差 (error: ϵ), 適合度 (fitness: F), 学習回数 (experience: exp), 最終学習エピソード (timestamp: ts), 行動集合に同時に含まれた分類子数 (actionset size: as) および重合度 (numerosity: n) からなる。分類子 cl のそれぞれのパラメータ $*$ は式中で $cl.*$ と表される。

3.3.2 学習メカニズム

図 3.5 に XCS の学習メカニズムの概要を示す。分類子を初期生成した後に、入力との照合、出力決定、正確性評価、分類子の一般化、新規分類子生成、分類子削除の 6 ステップを繰り返すことにより正確な分類子の獲得を目指す。

1) 実行部

実行部は入力された状態に対し行動を選択し出力する。状態が入力されると $[P]$ の中から入力された状態と照合する分類子のみを取り出し、 $[M]$ を生成する。ただし、入力と分類子の各要素において 0 あるいは 1 が一致した場合あるいは分類子の要素が $\#$ である時に照合したと判定される。このアルゴリズムを Algorithm 1 に示す。2 行目でそれぞれの属性において照合を行い、一つでも異なる場合には 3 行目で非照合と判定し、全ての属性において一致したならば、6 行目で照合と判定される。この時、照合した分類子集合を持つ行動の数がハイパーパラメータ θ_{mna} を下回った場合、被覆 (covering) と呼ばれる分類子を生成する操作が入る。被覆で生成される分類子の条件部 C は入力の各要素を独立に確率 $P_{\#}$ で $\#$ に変更したものを設定される。条件部 A は $[M]$ を構成する分類子が持たない行動の中からランダムに選択され設定される。パラメータ $p, \epsilon, F, exp, as, n$ はそれぞれ初期値 $p_I, \epsilon_I, F_I, 0, 1, 1$ に設定され、 ts は現在の Iteration (現在までに用いた学習データの個数を意味する) が設定される。

次に出力する行動を決定する。XCS はコンプリートアクションマップなため学習時は

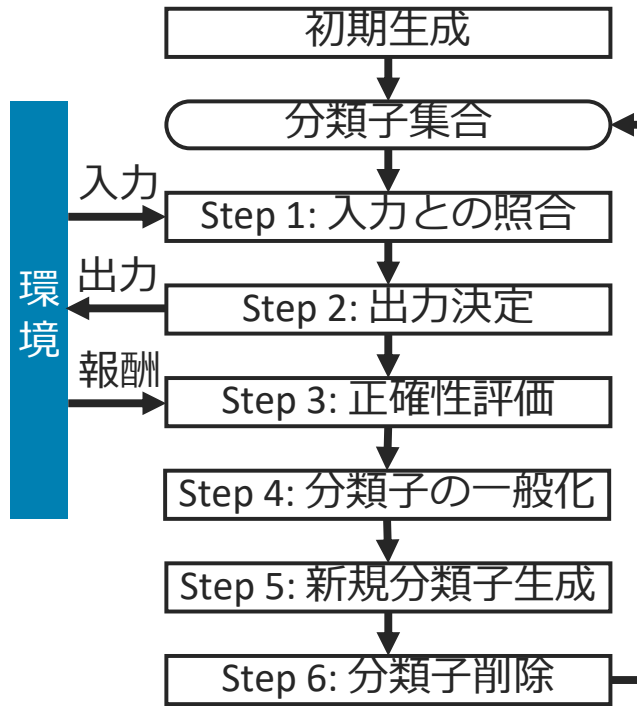


図 3.5 XCS の学習メカニズム

Algorithm 1 分類子照合 (Does Match)

```

1: for each attribute  $x$  in  $cl.C$  do
2:   if ( $x \neq \#$  and  $x \neq$  the corresponding attribute in input) then
3:     return false: 非照合
4:   end if
5: end for
6: return true: 照合

```

行動はランダムに決定される。しかし検証時には正しいと推測される行動を選択する必要がある。行動 a_i ごとに予測配列 (prediction array: $P(a_i)$) を下式によって算出し、その値を用いて ϵ -greedy[85] で行動を選択する。

$$P(a_i) = \frac{\sum_{cl_k \in [M] | a_i} cl_k \cdot p \times cl_k \cdot F}{\sum_{cl_l \in [M] | a_i} cl_l \cdot F} \quad (3.1)$$

ϵ (XCS においてはハイパーパラメータ p_{explr}) の確率で $P(a_i)$ の値に関係なくランダムに選択し、 $1 - \epsilon$ の確率で $P(a_i)$ の中で一番大きな値を持つ行動を選択する。ただし、 $[M] | a_i$ は、 $[M]$ 内で行動部に行動 a_i をもつ分類子の集合であり、 $[M]$ 内に存在しない行動の予測報酬は nil とされ行動選択の対象とならない。このアルゴリズムを Algorithm 2 に示す。1 行目で行動の選択方法を確率的に決め、2 行目のランダム選択あるいは 4 行目の期待値

Algorithm 2 行動選択 (Select action)

- 1: **if** RandomNumber[0,1) < p_{explr} **then**
 - 2: return a ronedomly chosen action from those notnil in $P(a)$: ランダムに行動
 選択
 - 3: **end if**
 - 4: return the best action in $P(a)$: 期待値の高い行動の選択
-

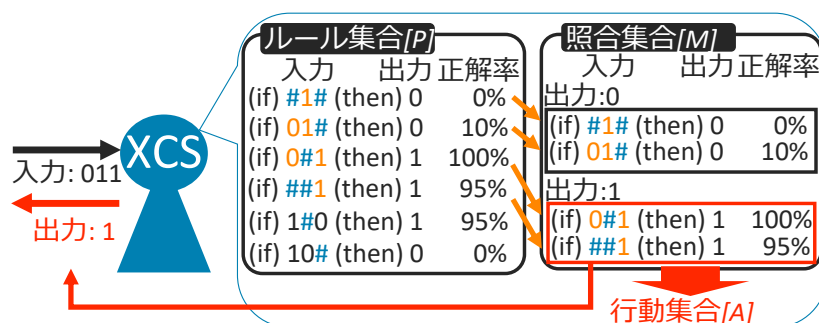


図 3.6 入力との照合と出力の決定

の高い行動選択を行う。[M] を構成する分類子の中から選択された行動を持つ分類子のみを抜き出し [A] を生成するとともに、選択した行動 a_i を環境に出力し実行部は終了する。

図 3.6 に入力との照合から [A] の生成までを図示したものを示す。この図の [P] において、入力 “011” に照合する分類子は $\{C : \#1\#, A : 0\}$, $\{C : 01\#, A : 0\}$, $\{C : 0\#1, A : 1\}$, $\{C : \##1, A : 1\}$ の 4 つである。この 4 つの分類子で [M] を生成し、出力毎に分ける。出力が 0 の分類子よりも出力が 1 の分類子の方が正解率が高いため、出力 1 を持つ分類子で [A] を生成する。

2) 強化部

行動を出力した結果得られた報酬 ρ に基づいて [A] を構成する分類子のパラメータを更新する。更新対象の分類子 cl の更新回数 exp を 1 だけ増加させた後、 p, ϵ, as の順で更新する。ここで、 β ($\beta \in [0, 1]$) は学習率と呼ばれ、システムの学習速度に寄与する。

$$cl.p \leftarrow cl.p + \beta(\rho - cl.p) \quad (3.2)$$

$$cl.\epsilon \leftarrow cl.\epsilon + \beta(|\rho - cl.p| - cl.\epsilon) \quad (3.3)$$

$$cl.as \leftarrow cl.as + \beta\left(\sum_{c \in [A]} c.n - cl.as\right) \quad (3.4)$$

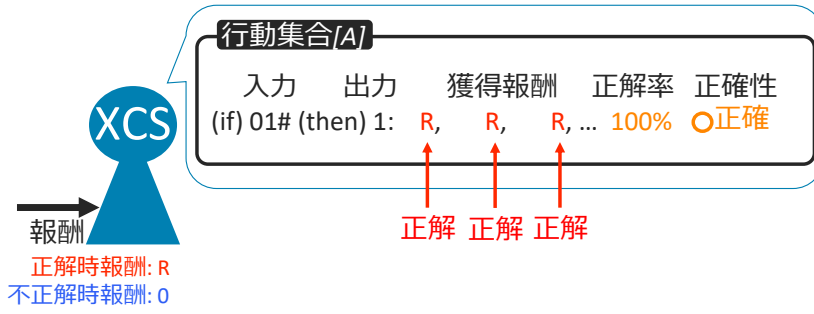


図 3.7 正確性の評価

ただし, $cl.exp$ が $1/\beta$ よりも小さい時上記の Widrow-Hoff の更新式を下記のように学習率を増やし学習速度を速める Moyeme adaptive modifee (MAM) が行われる [86].

$$cl.p \leftarrow cl.p + (\rho - cl.p)/cl.exp \quad (3.5)$$

$$cl.\epsilon \leftarrow cl.\epsilon + (|\rho - cl.p| - cl.\epsilon)/cl.exp \quad (3.6)$$

$$cl.as \leftarrow cl.as + \left(\sum_{c \in [A]} c.n - cl.as \right) / cl.exp \quad (3.7)$$

次に, 分類子の正確性 (accuracy : κ) を求め F を更新する. ただし, ϵ_0 は正確性の基準を定めるハイパーパラメータであり $\alpha (\alpha \in [0, 1])$ および $\nu (\nu \in (0, +\infty))$ は不正確な分類子における正確性の低下率を制御するハイパーパラメータである.

$$\kappa(cl) = \begin{cases} 1 & \text{if } \epsilon < \epsilon_0 \\ \alpha \left(\frac{\epsilon}{\epsilon_0} \right)^{-\nu} & \text{otherwise} \end{cases} \quad (3.8)$$

図 3.7 に分類子の正確性の評価を示す. 分類子 $\{C : 0\#, A : 1\}$ は常に報酬 R を獲得する (正解と評価される) 分類子である. 常に報酬 R が獲得さればらつきが無く ϵ が ϵ_0 を下回るため, 正確 (accurate) と判定され $\kappa = 1$ となる. F は先に求めた絶対的な正確性ではなく下記で求める相対的な正確性 κ' によって求められる.

$$\kappa'(cl) = \frac{\kappa(cl) \times cl.n}{\sum_{cl_A \in [A]} \kappa(cl_A) \times cl_A.n} \quad (3.9)$$

ここで, κ および κ' は再帰的な更新式ではなく, 分類子 cl のその時の状態 (パラメータの値) に基づいて計算されるため, $\kappa(cl)$, $\kappa'(cl)$ と表される. F は下式の様に更新される.

$$cl.F \leftarrow cl.F + \beta(\kappa'(cl) - cl.F) \quad (3.10)$$

F は 1 が最大値であり大きい値を取るほど $[A]$ 内に, 正確性が高い競合する分類子がなく照合範囲が重複していないことを示す. 照合範囲に重複がなければ少ないルール数で全ての入力に対応が可能である. 以上で強化部の操作は終了する.

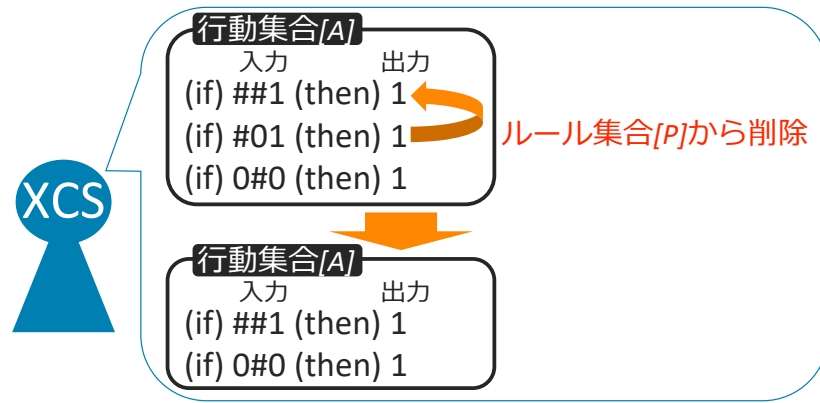


図 3.8 分類子の包摂

3) 発見部

発見部では包摂による分類子の縮約と遺伝的アルゴリズムによる新たな分類子の生成と F の低い分類子の削除による $[P]$ の最適化が行われる。

包摂は $[A]$ 内の分類子を $[A]$ 内でより一般化された C を持つ分類子に縮約する機構である。 $[A]$ を構成する分類子の中で評価回数が閾値 θ_{sub} よりも大きく、正確であり、 $[A]$ 内の分類子で C の # 記号の数が一番多い条件を満たす分類子を包摂元の分類子として選択する。

- $cl.exp > \theta_{sub}$
- $\kappa = 1$
- $[A]$ 内の分類子で C の # 記号の数が一番多いこと。

ただし 3 番目の条件については、同数の # 記号を持つ分類子が複数ある場合にはランダムで選択する。 2 つの分類子 cl_{gen} と cl_{spec} において、各要素 $C[i]$ で $cl_{gen}.C[i]$ が # 記号あるいは $cl_{gen}.C[i]$ と $cl_{spec}.C[i]$ が同値であれば cl_{gen} は cl_{spec} を一般化した形式であると定義される。 $[A]$ の分類子の中で他の分類子よりも一般化された形式の分類子が包摂元として選ばれた場合に、選ばれなかった分類子の中で選ばれた分類子が一般化された形となる分類子は包摂元の分類子に包摂される。包摂される分類子の n は包摂元の分類子の n に合算された後、包摂される分類子は削除される。図 3.8 に分類子の包摂を示す。この図に示されている 3 つの分類子は全て正確と判定されているとする。分類子 $\{C : ##1, A : 1\}$ は一番 # の数が多いので包摂元の分類子となる。この分類子は 3 番目のビットが 1 であるので分類子 $\{C : #01, A : 1\}$ は包摂対象となり、 $\{C : 0#0, A : 1\}$ は包摂対象とならない。包摂対象となった分類子 $\{C : #01, A : 1\}$ は $[P]$ から削除される。

GA は $[A]$ 内の分類子の前回の GA からの経過した学習エピソード数の平均がパラメー

Algorithm 3 GAにおけるルーレット選択 (Select offspring)

```
1:  $fitnessSum \leftarrow 0$ : 合計計算用
2: for each classifier  $cl$  in  $[A]$  do
3:    $fitnessSum \leftarrow fitnessSum + cl.F$ 
4: end for
5:  $choicePoint \leftarrow RandomNumber[0,1) * fitnessSum$ : 選択点の決定
6:  $fitnessSum \leftarrow 0$ 
7: for each classifier  $cl$  in  $[A]$  do
8:    $fitnessSum \leftarrow fitnessSum + cl.F$ 
9:   if  $fitnessSum > choicePoint$  then
10:    return  $cl$ : 分類子の選択
11:   end if
12: end for
```

タ θ_{GA} を超えた場合に行われる。GA の処理は以下の通りである。分類子の適合度 F の $[A]$ 内における比率を選択確率として、ルーレット選択を用いて2つの親個体を選択し、それらを複製し2つの子個体を生成する。 $[A]$ 内の分類子の前回のGAから経過した学習エピソード数の平均がハイパーパラメータ θ_{GA} を超えた時に新たな分類子の生成と分類子の削除 ($[P]$ の整理) が行われる。 $[A]$ から2つの分類子を $[A]$ 内での F の比率を選択確率としたルーレット選択を用いて選択する。このアルゴリズムを Algorithm 3 に示す。2行目から3行目で全体の $fitnessSum$ を計算し、5行目で選択点を決める。7行目から8行目は2行目から3行目と同様に計算し、9行目で選択点と比較し選択する分類子を決定する。選択した2つの分類子それぞれを複製し新たな分類子を生成する。ここでハイパーパラメータ χ で表される確率で新たに生成した分類同士で2点交叉する。2点交叉はそれぞれの分類子の C を先頭から x 番目から y 番目 ($0 \leq x, y < L$) を入れ替える。ここで、各子個体のパラメータ p, ϵ の値は2つの親個体の平均値に、 F は親個体の平均に対して小さな値 (一般的には $0.1[86]$) を掛けた値を設定する。これにより、子個体は親個体の値を引き継ぎつつも初期値に近い値を持ち、行動決定などに強い影響を持たない。また、 n, exp は初期値である1および0に設定される。この時、子個体の条件部および行動部を遺伝子としてとらえ、確率 χ で子個体の条件部を2点交叉させる。交叉位置はランダムに選ばれる。このアルゴリズムを Algorithm 4 に示す。1, 2行目において親個体を選択し3, 4行目においてコピーによって子個体の候補を生成する。5, 6行目において n, exp を初期値に設定する。8から19行目において2点交叉を行う。20から25行目において p, ϵ, F の値を決定する。また、27, 28行目において F を低下させる。

また突然変異をそれぞれの子個体の条件部の各ビットと行動部に対しそれぞれ確率 μ

Algorithm 4 子個体の生成

```
1:  $parent_1 \leftarrow$  Select offspring in  $[A]$ : 親個体の選択
2:  $parent_2 \leftarrow$  Select offspring in  $[A]$ : 親個体の選択
3:  $child_1 \leftarrow$  copy  $parent_1$ : 子個体の生成
4:  $child_2 \leftarrow$  copy  $parent_2$ : 子個体の生成
5:  $child_1.n = child_2.n \leftarrow 1$ : 初期値設定
6:  $child_1.exp = child_2.exp \leftarrow 0$ : 初期値設定
7: if RandomNumber $[0, 1) < \chi$  then
8:    $x \leftarrow$  RandomNumber $[0, 1) \times$  length of  $child_1.C + 1$ 
9:    $y \leftarrow$  RandomNumber $[0, 1) \times$  length of  $child_1.C + 1$ 
10:  if  $x > y$  then
11:    switch  $x$  and  $y$ 
12:  end if
13:   $i \leftarrow 0$ 
14:  while  $i < y$  do
15:    if  $x \leq i$  and  $i < y$  then
16:      switch  $child_1.C[i]$  and  $child_2.C[i]$ 
17:    end if
18:     $i ++$ 
19:  end while
20:   $child_1.p \leftarrow (parent_1.p + parent_2.p)/2$ 
21:   $child_1.\epsilon \leftarrow (parent_1.\epsilon + parent_2.\epsilon)/2$ 
22:   $child_1.F \leftarrow (parent_1.F + parent_2.F)/2$ 
23:   $child_2.p \leftarrow child_1.p$ 
24:   $child_2.\epsilon \leftarrow child_1.\epsilon$ 
25:   $child_2.F \leftarrow child_1.F$ 
26: end if
27:  $child_1.F \leftarrow child_1.F \times 0.1$ 
28:  $child_2.F \leftarrow child_2.F \times 0.1$ 
```

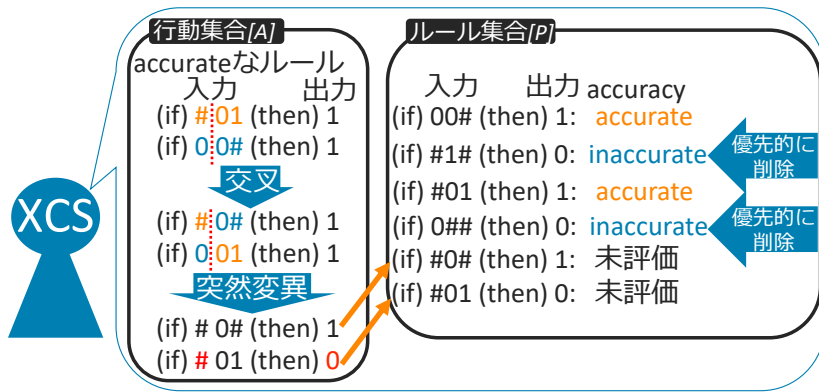


図 3.9 GA による分類子の生成と削除

で起こし子個体を再構成する。突然変異が起こった場合、条件部においてはそのビットが # であれば入力値に、# でなければ # に変更する。行動部においては、他の取りうる行動の中からランダムに選ばれる。ただし、生成された 2 つの子個体の分類子が 2 つの親個体のどちらかにでも包摂される関係にある時、その子個体は親個体に包摂され親個体の n が増加する。そうして生成された子個体を $[P]$ に追加し、マイクロ分類子の数 n の合計がパラメータ N を上回った場合は適合度 F が低いものから優先的に削除する。分類子 cl の削除の選択確率 (*deletion vote*) は以下の様に算出される。

$cl.exp > \theta_{del}$ および $cl.F/cl.n < \delta \times avFitnessInPopulation$ を満たす時

$$deletion\ vote = cl.as \times cl.n \times avFitnessInPopulation / (cl.F/cl.n) \quad (3.11)$$

どちらか一方でも満たさない時

$$deletion\ vote = cl.as \times cl.n \quad (3.12)$$

ただし、 $avFitnessInPopulation$ は $[P]$ 中の分類子の適合度の合計を重合度の合計で割ったもの、 $\delta (\delta \in [0, 1])$ は分類子の削除を制御するパラメータである。 $cl.F/cl.n$ が小さい分類子すなわち過剰に一般化された分類子を優先的に削除する。ここでいう分類子の数とはマイクロ分類子の数である重合度 n を表し、削除対象になると重合度を 1 減らし、重合度が 0 になるとその分類子は $[P]$ から削除される。

図 3.9 に GA による分類子の生成と削除を示す。左側が分類子の生成を、右側が分類子の削除を示す。左側の図より、交叉と突然変異を経て分類子 $\{C : \#01, A : 1\}$ と $\{C : 00\#, A : 1\}$ に似た分類子 $\{C : \#0\#, A : 1\}$ と $\{C : \#01, A : 0\}$ が生成される。生成された分類子は $[P]$ に挿入される。右側の図より、ルーレット選択に基づいて不正確な分類子が優先的に削除される。

3.3.3 アルゴリズム

XCS のアルゴリズムを Algorithm 5 に示す。XCS は環境から入力された状態に照合する分類子を $[P]$ の中から抜き出し (2 行目), $[M]$ を生成する (3 行目)。4 行目から 6 行目で $P(action)$ を基に決定した行動を有する分類子を $[M]$ から抜き出し, $[A]$ を生成する。7 行目から 19 行目にかけて, 選択した行動を環境に対して出力し得られた報酬を用いて $[A]$ の分類子のパラメータを更新し学習を行う。その後, 包摂条件や GA の条件を満たした場合に $[A]$ の分類子を対象として, 包摂や GA を行う (20 行目から 24 行目)。上記の流れを 1 学習エピソードとして, 学習および GA による生成と淘汰を繰り返すことによって, 環境に適した分類子集団を獲得する。

Algorithm 5 XCS のアルゴリズム

```
1: while (! end of iterations) do
2:   state ← environment: 環境から状態 state を受け取る
3:   generate [M]: [P] から state に照合する分類子を選択
4:    $P(a_i) = \sum_{cl_k \in [M] | a_i} cl_k.p \times cl_k.F / \sum_{cl_l \in [M] | a_i} cl_l.F$ : [M] から各行動の予測値を算出
5:   action ←  $P(action)$ :  $P(action)$  から行動を選択
6:   generate [A]: [M] から action に照合する分類子を選択
7:   reward ← execute action: 行動を実行し, 環境より報酬を獲得
8:   for all classifier cl in [A] do
9:      $cl.p \leftarrow cl.p + \beta(\rho - cl.p)$ 
10:     $cl.\epsilon \leftarrow cl.\epsilon + \beta(|\rho - cl.p| - cl.\epsilon)$ 
11:     $cl.as \leftarrow cl.as + \beta(\sum_{c \in [A]} c.n - cl.as)$ 
12:     $cl.exp \leftarrow cl.exp + 1$ 
13:    if  $\epsilon < \epsilon_0$  then
14:       $\kappa(cl) = 1$ 
15:    else
16:       $\kappa(cl) = \alpha(\epsilon/\epsilon_0)^{-\nu}$ 
17:    end if
18:     $\kappa'(cl) = \kappa(cl) \times cl.n / (\sum_{x \in [A]} \kappa(x) \times x.n)$ 
19:     $cl.F \leftarrow cl.F + \beta(\kappa'(cl) - cl.F)$ 
20:    if  $cl.exp > \theta_{sub}$  and  $\kappa = 1$  then
21:      subsumptionFlag = true
22:    end if
23:  end for
24:  if subsumptionFlag = true then
25:    run subsumption: [A] 内の分類子を対象に包摂を実行
26:  end if
27:  if run GA then
28:    generate new classifiers: 子個体の生成
29:    delete classifier: 削除
30:  end if
31: end while
```

第 4 章

問題設計

4.1 Multiplexer 問題

第 2 章で挙げた不確実性を模擬する問題として学習分類子システムにおいてベンチマーク問題として扱われる Multiplexer 問題 [87] を本研究で対象とする。この問題では、入力長 $l = k + 2^k$ (ただし k は任意の自然数定数) ビット $(b_0b_1\dots b_{l-1})$ のうち、先頭 k ビット $(b_0b_1\dots b_{k-1})$ を 10 進数に変換した値を d として、先頭から $k + d$ 番目のビット b_{k+d} の値を正答とする問題である。例えば図 4.1 に示す様に、 $l = 11(k = 3)$ である 11-Multiplexer 問題において、“01000110010” なる入力を与えられた場合、 $b_0b_1b_2 \rightarrow$ “010” を 10 進数に変換して $d = 2$ となり、よって $b_{2+3} = b_5 = 0$ を出力した場合に正答となる。また、先頭 k ビットをアドレスビット、後半 2^k ビットをリファレンスビットと呼ぶ。この問題においてアドレスビットである先頭 k ビット $(b_0b_1\dots b_{k-1})$ と、 $k + d$ ビット目 (b_{k+d}) のみが出力に寄与し、他のビットは出力に寄与しないため値を問わない。11-Multiplexer 問題の場合、アドレスビットである先頭 3 ビットと、それによって決定されたリファレンスビットの内の 1 ビットの合計 4 つのビットのみが出力に寄与する。よって、XCS は表 4.1 に示す様に、出力に寄与しないビットを # (“don’t-care”) 記号で置き換えた一般化された条件-行動則を獲得することが期待される。このようなルールは最適解と呼ばれる。なお、

アドレスビット リファレンスビット

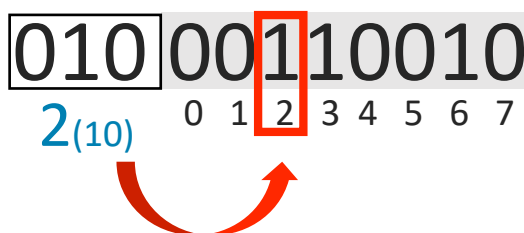


図 4.1 11-Multiplexer 問題

11-Multiplexer 問題における最適解は，出力に寄与する 4 ビットの組み合わせと，0,1 の行動による正答の場合，誤答の場合を考慮した，計 $2^4 \times 2 = 32$ 個存在する。

表 4.1 11-Multiplexer 問題における最適解

条件部	行動部	条件部	行動部
0000#####	0	100####0###	0
0000#####	1	100####0###	1
0001#####	0	100####1###	0
0001#####	1	100####1###	1
001#0#####	0	101#####0##	0
001#0#####	1	101#####0##	1
001#1#####	0	101#####1##	0
001#1#####	1	101#####1##	1
010##0#####	0	110#####0#	0
010##0#####	1	110#####0#	1
010##1#####	0	110#####1#	0
010##1#####	1	110#####1#	1
011###0####	0	111#####0	0
011###0####	1	111#####0	1
011###1####	0	111#####1	0
011###1####	1	111#####1	1

Multiplexer 問題は回路の動作から作られた問題であるため，（不確実性を伴う）実問題とは直接的な関係はないが，上記で示した様に簡潔なルールの獲得によって一般化能力を測ることができるため，知識獲得のベンチマーク問題として適している。さらに，獲得すべき簡潔なルールが分かっているため，下記に示す様に不確実性を模擬しても，適切に簡潔なルールを獲得できるかにより，その手法の獲得した知識の正当性の裏付けになる。

4.2 対象とする不確実性の模擬

XCS は図 3.1 に示す様に観測対象である環境と XCS は入力，出力および報酬で情報をやりとりするフレームワークのため，入力，出力および報酬において不確実性を模擬する。入力あるいは出力に対して値を変更するノイズを付加することで (i) 入出力における不確実性を，報酬に対して乱数を付加することで (ii) 判定の信憑性における不確実性を模擬する。以下，Multiplexer 問題上に模擬したそれぞれの不確実性について説明する。

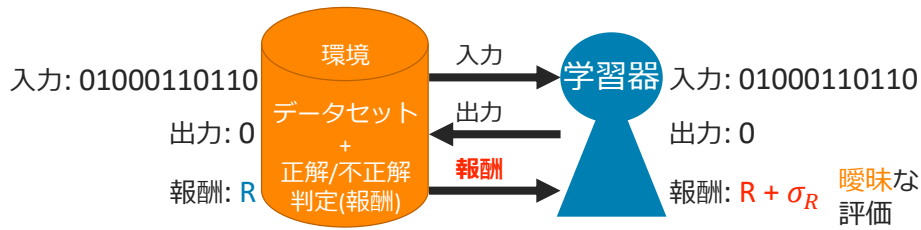


図 4.4 判定の信憑性における不確実性の模擬

て模擬する。本論文において特に言及が無いときには乱数として平均 0 で標準偏差が σ_R のガウス分布 $N(0, \sigma_R^2)$ を想定する。

4.3 評価基準

正しさ・適応性を測るために正答率 (Correct rate) の平均を，解釈性を測るために分類子数 (Population size) の平均，最適解集合 $[O]$ を獲得した試行数の平均，最大適合度 F の分類子数の平均を評価基準とする。それぞれの指標を採用した意図を以下に説明する。ただし正答率の平均および分類子数の平均について，それらの値は 100 学習ステップで移動平均を取ったものに対し，学習データの生成や実行部における学習時の行動選択および発見部における GA に関する乱数を生成するシードを変えた試行の平均を用いる。

- 正答率の平均

分類問題に対する学習度合を評価するために，1 学習エピソードが終了するたびに学習データとは別の評価用データを与え，正解の行動を選択可能かを測定する。出力が正解であれば 1 を不正解であれば 0 と評価する。平均を取った正答率が 1 に近いほど環境からの入力に対して正しい行動を選択できていることを示す。また，正答率が常に高ければ様々な入力に対して正しい行動を選択できるため適応性（網羅性）が高いことが分かる。ただし，評価においては被覆無しの実行部のみが実行され，行動選択は prediction array で一番値が高い行動が選択される。被覆，強化部および発見部が作動しないため分類子のパラメータは更新されず $[P]$ に影響はない。

- 分類子数の平均

分類子の一般化度合を $[P]$ を構成する分類子の数で評価する。1 学習エピソードが終了するたびに $[P]$ を構成する（マクロ）分類子の数を計測する。一般的に，正答率が同じであれば，分類子数が小さいほど一般化が進んでいるが，大きいほど一般化が進んでいないことを示す。ただし，この分類子数の平均の指標は GA で生成されて間もない評価途中の分類子も含まれるため，表 4.1 に示すような最適な分類子

例) 入力: 011, 出力: 1の場合
 行動集合[A]

	評価値 F
(if) 01# (then) 1:	0.9
(if) 0#0 (then) 1:	0.05
(if) #1# (then) 1:	0.01
(if) ### (then) 1:	0.04

例) 入力: 010, 出力: 1の場合
 行動集合[A]

	評価値 F
(if) 01# (then) 1:	0.9
(if) 0## (then) 1:	0.04
(if) #10 (then) 1:	0.03
(if) ##0 (then) 1:	0.02

図 4.5 最大適合度 F の分類子

数よりも大きくなる。

- 最適解集合 $[O]$ を獲得した試行数

本研究は知識発見を主眼にしており，その一つとして学習分類子システムによる一般化された分類子の獲得を目的にしている．その目的が達成されたかをそれぞれの試行の学習終了時点で表 4.1 に示される様な Multiplexer 問題における全ての最適解 (最適解集合 $[O]$) を表す分類子の獲得について評価する．さらに，学習終了時に $[P]$ を構成する分類子に対し F で降順にソートし上位 n 個 (11-Multiplexer 問題であれば上位 32 個) 全てが $[O]$ の分類子であるかを評価する．この指標により $[O]$ を正確かつ保持すべき分類子としてシステムが認識していることを評価する．

- 最大適合度 F の分類子数の平均

データ全体を少数で簡潔なルールで表現することを目的としていることと XCS が式 (3.9) から一つの入出力対において一つの (一般化された) 分類子が照合する様に学習することから，[88] を参考にして $[A]$ の中で F が最大となる分類子を抜き出す．適切に分類子が一般化され $[O]$ が安定的に獲得できていれば $[O]$ を構成する分類子と同じ数 (11-Multiplexer 問題であれば 32 個) となる．この操作によって $[P]$ から手法に不正確と認識されている分類子，正確だが $[O]$ と異なった (効率の悪い) 一般化をされた分類子，評価回数の少ない分類子を除くことができる．この指標により XCS が $[O]$ を正確かつ保持すべき分類子としてシステムが認識していることおよびその分類子を適切に抜き出せるかを評価する．図 4.5 に最大適合度 F の分類子の例を示す．分類子 $\{C : 01\#, A : 1\}$ は入力が “011” であっても “010” であっても F が最大の分類子である． F が大きいほど XCS が適切な分類子であると認識しているため，獲得された分類子の中で入力 “011” および “010” に対応する分類子は分類子 $\{C : 01\#, A : 1\}$ であると言える．分類子は一般化によって複数の入力に照合するため，この例の様に最大適合度 F の分類子が共通する場合がある．

実験に用いるデータは手法間で違いは無く，正答率の平均と分類子数の平均それぞれに

ついて手法が異なっても同一の分散を持つ正規分布に従うとは限らないため多群検定として Kruskal-Wallis 検定と 2 群検定として Brunner-Munzel 検定を用いて有意性を判定する。ただし，有意水準は 1% とする。

第 5 章

入出力における 不確実性に対応可能な 学習分類子システム

入出力における不確実性において図 4.2 および図 4.3 からシステムに与えられる報酬の種類は正解と不正解を表す R(1000 と設定) と 0 の 2 種類のみであることに着目し, 入出力における不確実性を伴う環境においても認識した入力と出力の組み合わせにおける報酬 (0 または 1000) を予測して学習する, 推測報酬に基づく学習分類子システム (XCS based on Estimated Reward: XCS-ER) を提案する.

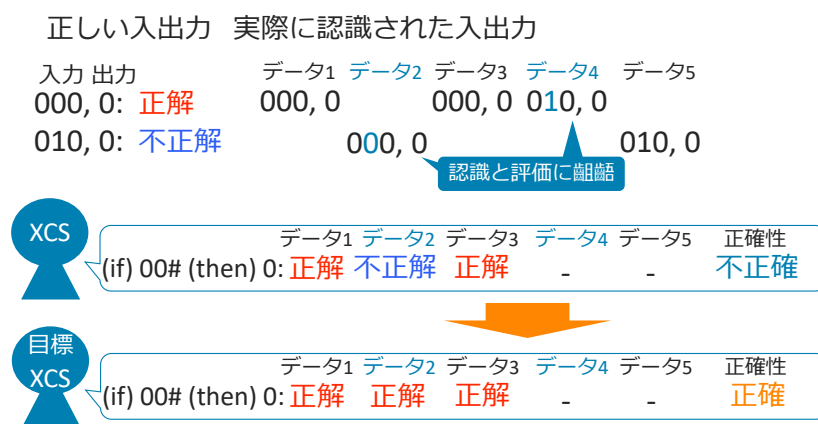


図 5.1 入出力における不確実性を伴う場合

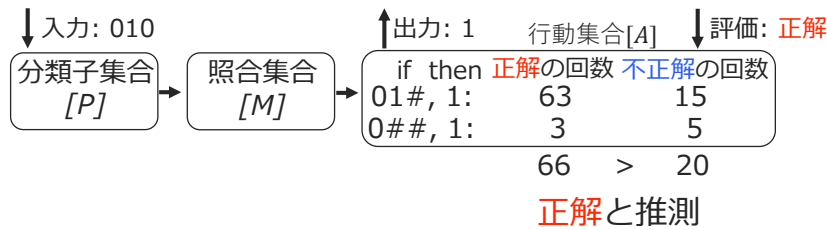


図 5.2 XCS-ER の概要

5.1 入出力における不確実性に対するアプローチ

図 5.1 は入出力における不確実性を伴った場合の入出力に対する判定とそれに基づく XCS の正確性判定および、目標とする不確実性に対処可能な XCS の正確性判定を示す。上部では入出力における不確実性を伴う場合のデータを例示しており、データ 2 およびデータ 4 において不確実性により認識された入出力に対する正誤評価が異なる。このデータにおいて XCS は正確な分類子 $\{C : 00\#, A : 0\}$ が認識された入力が “000” の場合に照合し、正解（報酬 R ）と不正解（報酬 0）が混在し不正確と評価される。この様に入出力における不確実性を伴うと、分類子が獲得した報酬（全体）の分布（形状・最大値や最小値）は正確な分類子と不正確な分類子で比率は異なるものの同一なので、分類子の正確性の判定に用いることが困難である。

この問題を解決するため、入出力における不確実性に対し、分類子が獲得した報酬全体の分布ではなく、ひとつひとつの報酬の “正しさ” に着目し、分類子が本来獲得する報酬を推測するアプローチを取る。入出力における不確実性を伴うため、正確な分類子であっても報酬 0 と R の両方を獲得するが、報酬を獲得した際に分類子が本来獲得すると推測される報酬に上書きできれば、獲得する報酬は正確な分類子であれば 1 種類のみ、不正確な分類子であれば 0 と R の両方となる。獲得された報酬ではなく、推測報酬を正確性の判定に用いることで正しく分類子の正確性を判定できる。

図 5.2 に XCS-ER の概要を示す。入力と出力が定まった時点つまり $[A]$ が生成された時点において評価回数が多い評価を採用する。図の例では正解の回数が不正解の回数を上回っているため、 $\{\text{入力} : 010, \text{出力} : 1\}$ は正解と推測する。以下に、XCS-ER の詳細を記す。

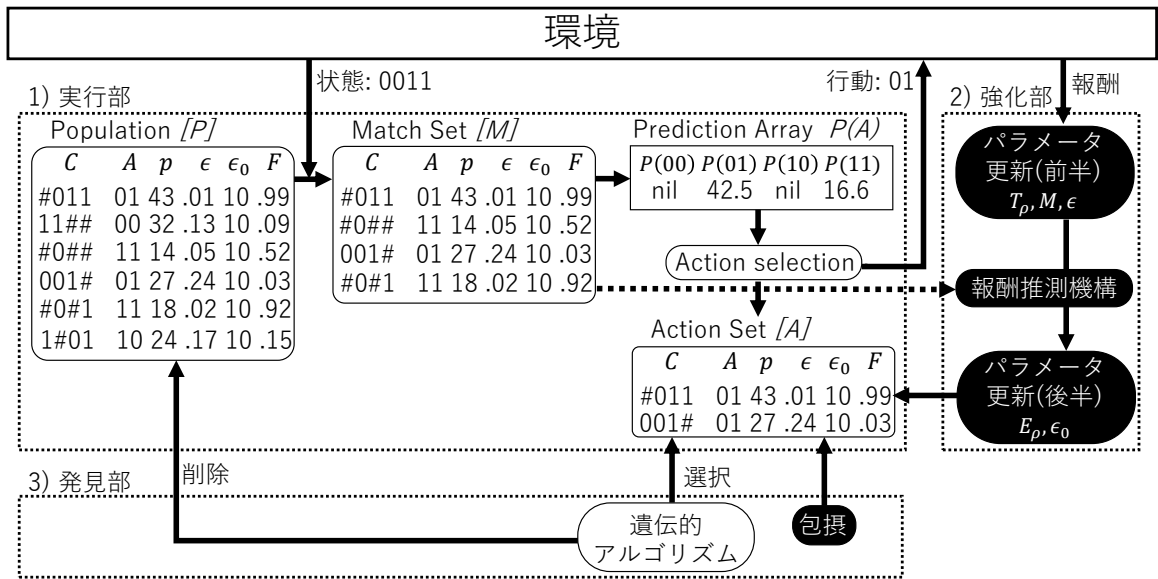


図 5.3 XCS-ER のアーキテクチャ

5.2 構成

5.2.1 アーキテクチャ

XCS-ER の XCS との差分は強化部の報酬推測機構およびパラメータ更新と発見部の包摂である。図 5.3 の黒塗りの部分はその差分を表す。また、新設されたパラメータおよび異なる方法で更新されるパラメータを明記している。XCS-ER は報酬推測により報酬に伴う不確実性を低減し、分類子の正しい正確性の評価を図る。報酬を獲得すると直ちに報酬推測に用いるパラメータを更新し、報酬推測を行う。推測結果を基に更新対象の分類子の正確性に関わるパラメータを更新する。

5.2.2 分類子

XCS-ER の分類子は XCS の分類子が持つ $C, A, p, \epsilon, F, exp, ts, as, n$ に加えて、獲得した報酬 $reward$ の平均 M 、正確性基準 ϵ_0 、環境から報酬 $reward$ を獲得した回数 $T_{\rho=reward}$ および推測報酬が推測した値が $reward$ であった回数 $E_{\rho=reward}$ を持つ。ただし、 $T_{\rho=reward}, E_{\rho=reward}$ はそれぞれ $reward$ の種類だけ存在する。つまり、2 クラス分類問題の Multiplexer 問題では $T_{\rho=0}, T_{\rho=1000}, E_{\rho=0}, E_{\rho=1000}$ となる。

5.3 学習メカニズム

5.3.1 強化部

パラメータ更新は $p, exp, ts, as, n, T_{\rho=reward}, M, \epsilon$ を更新する前半と $E_{\rho=reward}, \epsilon_0, \kappa, \kappa', F$ を更新する後半に分かれ、その間に報酬推測機構が動作する。パラメータ更新を分け、報酬推測に用いるパラメータを報酬の獲得直後に更新することで、報酬推測に獲得したばかりの報酬の情報を含めて用いることができるため、より安定した推測が行われることが見込まれる。

パラメータ更新（前半）

パラメータ更新（前半）では報酬推測に必要な分類子が獲得する報酬全体についての情報を集める。パラメータ p, exp, ts, as, n については XCS と同様の更新を行うが ϵ については XCS の更新と異なる。パラメータの更新対象となる分類子は XCS と同様に $[A]$ を構成する分類子のみである。新設されたパラメータおよび ϵ の更新について述べる。まず報酬 $reward$ が与えられるとすべての分類子 $cl \in [A]$ において $cl.T_{\rho=reward}$ を 1 だけ増加させ、報酬 $reward$ を獲得した回数を記録する。

$$cl.T_{\rho=reward} \leftarrow cl.T_{\rho=reward} + 1 \quad (5.1)$$

次に獲得した報酬の平均値 M を逐次的に求める。

$$cl.M \leftarrow (reward - cl.M)/cl.exp \quad (5.2)$$

さらに ϵ を求めるために、 $cl.T_{\rho=reward}$ を用いて分類子 cl が獲得した回数が一番多い報酬を求める。分類子ごとに報酬を獲得した回数が異なるため、便宜上獲得した回数が一番多い報酬を $cl.mfr$ と表す。 ϵ は $cl.mfr$ と $cl.M$ を用いて

$$cl.\epsilon \leftarrow cl.\epsilon + \beta(|cl.mfr - cl.M| - cl.\epsilon) \quad (5.3)$$

と求められる。 ϵ が小さいほど特定の値の報酬を多く獲得し、大きいほど獲得報酬の偏りが少ないことを表す。パラメータ更新（前半）のアルゴリズムを Algorithm 6 に示す。3 行目でパラメータ p, exp, ts, as, n を XCS と同様に更新し、その後 $T_{\rho=reward}, M, \epsilon$ を更新する。

報酬推測機構

入力に対する出力の評価はそれぞれ異なると想定し、XCS-ER は $[M]$ を構成する分類子を分類子が持つ行動ごとに分けた集合（collective と呼ぶ）の評価回数を重みとした獲

Algorithm 6 パラメータ更新 (前半)

- 1: receive *reward*: 環境より報酬を獲得
 - 2: **for all** classifier *cl* in [*A*] **do**
 - 3: update *p, exp, ts, as, n*: XCS と同様の更新
 - 4: $cl.T_{\rho=reward} \leftarrow cl.T_{\rho=reward} + 1$
 - 5: $cl.M \leftarrow (reward - cl.M)/cl.exp$
 - 6: $cl.\epsilon \leftarrow cl.\epsilon + \beta(|cl.mfr - cl.M| - cl.\epsilon)$
 - 7: **end for**
-

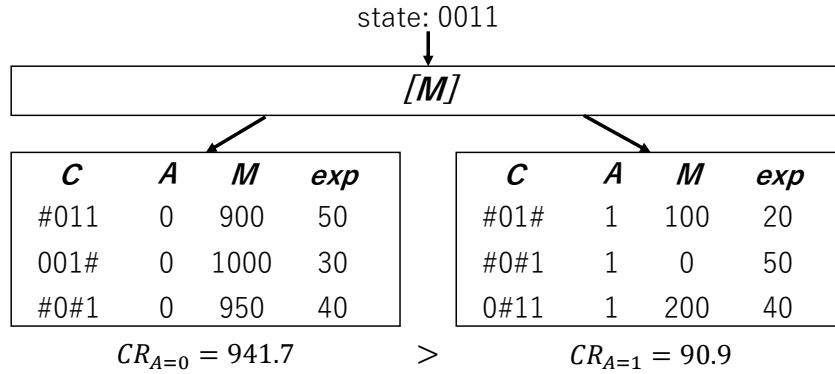


図 5.4 報酬推測機構

得報酬の平均の加重平均を比較することにより、[*A*] が不確実性によらず認識した入力と出力の組において獲得すべき報酬を推測する。加重平均 $CR_{A=a}$ は以下の様に求められる。ここで、XCS-ER は入力に対する正しい出力の正誤を認識できないことおよび学習に用いられる [*A*] とは別に生成されることから、[*M*] を構成する分類子を分類子が持つ行動ごとに分けた集合を *collective* と呼ぶ。

$$CR_{A=a} = \frac{\sum_{cl \in [M] | A=a} cl.M \times cl.exp}{\sum_{C \in [M] | A=a} C.exp} \quad (5.4)$$

ここで、 $cl \in [M] | A = a$ は [*M*] を構成する分類子の中で行動が *a* である分類子 *cl* を表す。評価回数を重みとすることで正解および不正解の回数的大小関係を把握することができる。行動ごとに求められた $CR_{A=a}$ をソートし、その順番で *collective* が認識した入力と出力の組において獲得すると推測される報酬を XCS-ER がこれまでに獲得した報酬をソートした順番に割り当てる。つまり、 $CR_{A=a}$ が一番大きい行動が一番多く正解の評価を獲得するため、正解の出力と推測される。Multiplexer 問題における図 5.4 の例では、 $CR_{A=0}$ の方が $CR_{A=1}$ よりも大きく、XCS-ER は 0 と 1000 の 2 種類の報酬を獲得しているため、行動が 0 の *collective* は報酬 1000 を行動が 1 の *collective* は報酬 0 を獲得すると推測される。報酬推測機構のアルゴリズムを Algorithm 7 に示す。1 行目で [*M*] を

Algorithm 7 報酬推測機構

- 1: divide $[M]$: 行動ごとに $[M]$ の分類子を collective に分割
 - 2: **for all** collective $[C]$ **do**
 - 3: $CR_{A=a} = (\sum_{cl \in [M]|A=a} cl.M \times cl.exp) / \sum_{C \in [M]|A=a} C.exp$
 - 4: **end for**
 - 5: sort CR : 加重平均 CR を並び替え
 - 6: estimate ρ' : 報酬 ρ' の推測
-

collective に分割し、3行目で CR を求める。5行目および6行目で CR の大小関係から現在の行動の正誤を推測し、これまでに獲得した報酬を順に割り当てる。

パラメータ更新 (後半)

報酬推測機構によって $[A]$ が獲得すると推測した報酬値 ρ' を用いて推測報酬が1種類の分類子のみを正確と判定する。報酬推測機構によって推測された回数 $E_{\rho=reward}$ を

$$cl.E_{\rho=\rho'} \leftarrow cl.E_{\rho=\rho'} + 1 \quad (5.5)$$

と更新する。図 5.4 の例において、行動 0 が選ばれた場合に $E_{\rho=1000}$ が行動 1 が選ばれた場合には $E_{\rho=0}$ が更新される。

さらに、 ϵ_0 を次の手順で求める。

$$\begin{cases} cl.exp > 2^{numberof\#} \times \theta_{RE} \\ \exists cl.E_{\rho=reward}, cl.exp = cl.E_{\rho=reward} \end{cases} \quad (5.6)$$

を満たす分類子の ϵ から最大値 $Max\epsilon$ を求める。ただし、 θ_{RE} は評価回数の下限を定めるハイパーパラメータである。この条件は C にある $\#$ 記号の数 ($numberof\#$) に比例する回数の条件を設けることで、分類子が照合する入力ほぼ全てにおいて評価されることを期待する。式 (5.6) の下側の条件によって推測報酬値 ρ' が1種類の分類子のみが対象となる。求めた $Max\epsilon$ を用いて ϵ_0 は

$$cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(Max\epsilon - cl.\epsilon_0) \quad (5.7)$$

と更新される。 $Max\epsilon$ となる ϵ を持つ分類子は式 (5.6) を満たし正確であると判断されるため、それよりも ϵ が小さい分類子も正確であるとみなす。その後、 κ, κ', F が更新される。パラメータ更新 (後半) のアルゴリズムを Algorithm 8 に示す。4行目において推測報酬の獲得回数を記録し、5行目で正確とみなすべき分類子を調べ、6, 7行目においてその分類子を正確と判定できる様に ϵ_0 を設定する $Max\epsilon$ を求める。11行目から14行目において $\epsilon_0, \kappa, \kappa', F$ を更新する。

Algorithm 8 パラメータ更新 (後半)

```
1: receive  $\rho'$ : 推測報酬を獲得
2: initialize  $Max\epsilon$ 
3: for all classifier  $cl$  in  $[A]$  do
4:    $cl.E_{\rho=\rho'} \leftarrow cl.E_{\rho=\rho'} + 1$ 
5:   if  $cl.exp > 2^{numberof\#} \times \theta_{RE}$  and  $\exists cl.E_{\rho=reward}, cl.exp = cl.E_{\rho=reward}$  then
6:     if  $Max\epsilon < cl.\epsilon$  then
7:        $Max\epsilon = cl.\epsilon$ 
8:     end if
9:   end if
10: end for
11: for all classifier  $cl$  in  $[A]$  do
12:    $cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(Max\epsilon - cl.\epsilon_0)$ 
13:   update  $\kappa(cl), \kappa'(cl), F$ : パラメータ更新 (XCS と同様)
14: end for
```

5.3.2 発見部

包摂

分類子は包摂されるとその分類子に関する情報は全て失われるため、包摂は慎重になされるべきである。一般化度合いが高い（# 記号の数が多い）分類子は多くの分類子を包摂する可能性があるため、一般化度合いに応じた制限が必要となる。正確性はすべての照合する入力において同一の評価である必要があるため、 C にある # 記号の数 ($numberof\#$) に比例する回数の条件を設けることで、分類子が照合する入力のほぼ全てにおいて評価されることを期待し、XCS の包摂条件に対し $cl.exp > 2^{numberof\#} \times \theta_{RE}$ を加え、

- $cl.exp > \theta_{sub}$
- $cl.exp > 2^{numberof\#} \times \theta_{RE}$
- $\kappa = 1$
- $[A]$ 内の分類子で C の # 記号の数が一番多いこと。

とする。これにより、XCS よりも慎重に包摂元の分類子を選択できる。

5.4 アルゴリズム

XCS-ER のアルゴリズムを Algorithm 9 に示す。下線が引かれている部分が XCS からの変更点である。XCS-ER は環境から入力された状態に照合する分類子を $[P]$ の中から抜き出し、 $[M]$ を生成する (2, 3 行目)。P(action) を基に決定した行動を有する分類子を $[M]$ から抜き出し、 $[A]$ を生成する (4 から 6 行目)。選択した行動を環境に対して出力し得られた報酬を用いて $[A]$ の分類子のパラメータを更新し学習を行う。8, 9, 10 行目はそれぞれ Algorithm 6 のパラメータ更新 (前半)、Algorithm 7 の報酬推測機構、Algorithm 8 のパラメータ更新 (後半) に対応する。その後、包摂条件や GA の条件を満たした場合に $[A]$ の分類子を対象として、包摂 (17 行目) や GA (20 行目) を行う。包摂条件では、包摂を行う分類子が包摂をする分類子のいずれよりも評価回数が多いかを確認する (12 行目)。上記の流れを 1 学習エピソードとして、学習及び GA による生成と淘汰を繰り返すことによって、環境に適した分類子集団を獲得する。

5.5 実験

5.5.1 実験内容

XCS-ER は入出力における不確実性に対応可能なため第 4.2.1 項および第 4.2.2 項の不確実性を扱う。それぞれの不確実性における不確実性の強度を表すパラメータが $P_I = 0, 0.05, 0.1, 0.2$ と $P_O = 0, 0.1, 0.2, 0.25$ の時および両不確実性を複合した $(P_I, P_O) = (0.05, 0.2)$ を扱う。ただし、 $P_O = 0$ は $P_I = 0$ の不確実性を伴わない場合と同様であるため省略する。また、不確実性の強度を以下の 2 種類のケースとした実験を行う。

- ケース U: すべての入力と出力の組において均一の不確実性強度
- ケース V: 入力と出力の組ごとに P_I あるいは P_O を上限とした不確実性強度

ケース U (Uniform) では不確実性強度の強弱による手法の適応性を評価し、ケース V (Various) では不確実性強度が均一でない場合における適応性を評価する。

5.5.2 実験設定

XCS で用いるパラメータは標準的に用いられている [89] のものを参考に以下のものを用いた。具体的な値は $\theta_{mna} = 2$, $P_{\#} = 0.35$, $p_I = 0.01$, $\epsilon_I = 0.01$, $F_I = 0.01$, $p_{explr} = 1.0$, $\beta = 0.2$, $\epsilon_0 = 10$, $\alpha = 0.1$, $\nu = 10$, $\theta_{sub} = 20$, $\theta_{GA} = 25$, $\chi = 0.8$, $\mu = 0.04$, $N = 800$, $\theta_{del} = 20$, $\delta = 0.1$ となる。XCS-ER で新設されたパラメータ θ_{RE}

Algorithm 9 XCS-ER のアルゴリズム

```
1: while (! end of iterations) do
2:   state  $\leftarrow$  environment: 環境から状態 state を受け取る
3:   generate [M]: [P] から state に照合する分類子を選択
4:    $P(a_i) = \sum_{cl_k \in [M]|a_i} cl_k \cdot p \times cl_k \cdot F / \sum_{cl_l \in [M]|a_i} cl_l \cdot F$ : [M] から各行動の予測値を算出
5:   action  $\leftarrow$   $P(action)$ :  $P(action)$  から行動を選択
6:   generate [A]: [M] から action に照合する分類子を選択
7:   reward  $\leftarrow$  execute action: 行動を実行し, 環境より報酬を獲得
8:   update parameters (first half): パラメータ更新 (前半) (Algorithm 6)
9:   estimate reward  $\rho'$ : 報酬  $\rho'$  の推測 (Algorithm 7)
10:  update parameters (second half): パラメータ更新 (後半) (Algorithm 8)
11:  for all classifier cl in [A] do
12:    if  $cl.exp > \theta_{sub}$  and  $\kappa = 1$  and  $cl.exp > 2^{numberof\#} \times \theta_{RE}$  then
13:      subsumptionFlag = true
14:    end if
15:  end for
16:  if subsumptionFlag = true then
17:    run subsumption: [A] 内の分類子を対象に包摂を実行
18:  end if
19:  if run GA then
20:    generate new classifiers: 子個体の生成
21:    delete classifier: 削除
22:  end if
23: end while
```

は入力長 L に基づいて $\theta_{RE} = 2$ と定めた。Iteration は 1,000,000 回, 試行数は不確実性が単体の時は 50, 不確実性が複合した場合は 5 と設定する。また, 評価基準は第 4.3 節で説明した正答率の平均, 分類子数の平均および最適解集合 [O] を獲得した試行数と最大 F の分類子数の平均である。

5.5.3 実験結果

不確実性が伴わない場合 $P_I = P_O = 0$ の時の正答率の平均と分類子数の平均を図 5.5 に, 入力における不確実性が $P_I = 0.05, 0.1, 0.2$ (ケース U) の時の正答率の平均と分類子数の平均をそれぞれ図 5.6, 5.7, 5.8 に, $\sup\{P_I\} = 0.05, 0.1, 0.2$ (ケース V) の時の正

表 5.1 $P_I = P_O = 0$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	47 (94%)	47 (94%)	32.02
XCS-ER	50 (100%)	50 (100%)	32

答率の平均と分類子数の平均をそれぞれ図 5.9, 5.10, 5.11 に示す. それぞれの図の横軸は Iteration を表し, 縦軸は正答率の平均あるいは分類子数の平均を表す. また, それぞれにおける最適解集合 [O] を獲得した試行数と最大 F の分類子数の平均をそれぞれ表 5.1 - 5.7 に示す.

不確実性を伴わない場合 ($P_I = P_O = 0$), XCS および XCS-ER ともに正答率の平均は 100% に達した. 分類子数の平均については XCS は 99.1, XCS-ER は 97.6 と XCS-ER の方が小さいが ($p < 0.01$) ほぼ同程度である. [O] の獲得については上位 32 個の分類子に限定しても, 全ての試行において獲得した. また, 最大 F の分類子数の平均は両手法とも 32 となっている. このことから, 不確実性を伴わない環境において XCS および XCS-ER ともに分類子の正確性を正しく認識し分類子の一般化が可能であることが分かる.

入力における不確実性 (P_I) を伴う環境における結果について述べる. ケース U の場合 $P_I = 0.05$ であれば XCS-ER は正答率の平均 100%, 分類子数の平均 96.6, [O] の獲得試行数 50 および最大 F の分類子数の平均が 32 と不確実性を伴わない場合と同等の結果を得られている. しかし, XCS は正答率の平均が 82.4% と 100% まで達しておらず分類子数の平均が 574.6 にまで増加し [O] の獲得試行数は 0 であることおよび最大 F の分類子数の平均が 32 よりも大きいことから, 不確実性の影響から正確な分類子を認識することができず一般化度合が低い分類子が多い. 正答率の平均および分類子数の平均について有意差が見られた. ($p < 0.01$) P_I を上昇させると XCS-ER は $P_I = 0.2$ の場合において正答率の平均が 69.9% と正しく行動を選択することができていなかった. 入力に対して正解の出力が分からないため, 分類子を適切に一般化できていない. 正答率の平均も分類子数の平均も XCS-ER は XCS よりも良い結果であった. ($p < 0.01$) ケース V の場合 XCS は $\sup\{P_I\} = 0.05, 0.1, 0.2$ のどの場合においてもケース U よりも悪化度合は少ないものの, 正答率の平均は 100% まで達さず分類子数の平均は増加した. また, [O] の獲得試行数はどの場合においても 0 であり最大 F の分類子数の平均の数が 32 よりも大きな値であった. XCS-ER は $\sup\{P_I\} = 0.05, 0.1, 0.2$ のどの場合においても正答率の平均が 100% にまで達し, 分類子数の平均も 100 程度を維持し, ほとんどの試行において [O] を獲得し最大 F の分類子数の平均も 32 に近い値である. ケース V では全ての場合において正答率の平均と分類子数の平均に有意差が見られた. ($p < 0.01$)

出力における不確実性 (P_O) を伴う環境における結果について述べる. 出力における

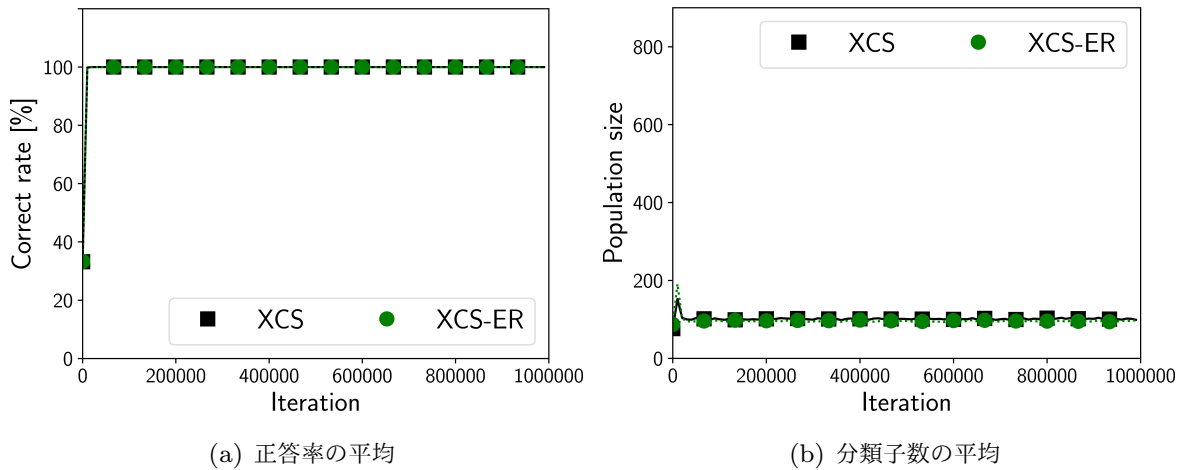


図 5.5 $P_I = P_O = 0$

表 5.2 ケース U: $P_I = 0.05$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	56.08
XCS-ER	50 (100%)	50 (100%)	32

表 5.3 ケース U: $P_I = 0.1$ における最適解の獲得

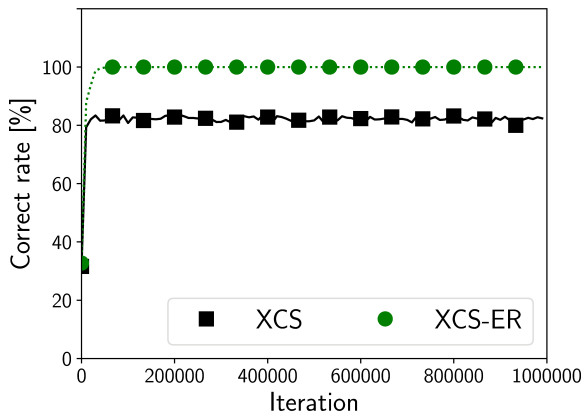
手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	50.16
XCS-ER	49 (98%)	49 (98%)	31.66

表 5.4 ケース U: $P_I = 0.2$ における最適解の獲得

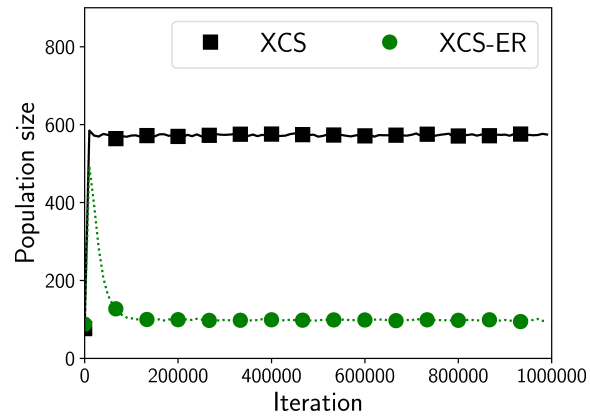
手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.24
XCS-ER	0 (0%)	0 (0%)	14.88

不確実性が $P_O = 0.2, 0.3, 0.4$ (ケース U) の時の正答率の平均と分類子数の平均をそれぞれ図 5.12, 5.13, 5.14 に, $\sup\{P_O\} = 0.2, 0.3, 0.4$ (ケース V) の時の正答率の平均と分類子数の平均をそれぞれ図 5.15, 5.16, 5.17 に示す. それぞれの図の横軸は Iteration を表し, 縦軸は正答率の平均あるいは分類子数の平均を表す. また, それぞれにおける最適解集合 [O] を獲得した試行数と最大 F の分類子数の平均をそれぞれ表 5.8 - 5.13 に示す.

ケース U およびケース V のどちらの場合においても入力における不確実性を伴う場合と同様の結果を得た. XCS はケース U およびケース V においても, 正答率の平均は 100% まで達しておらず, 分類子数の平均は不確実性を伴わない環境における値である

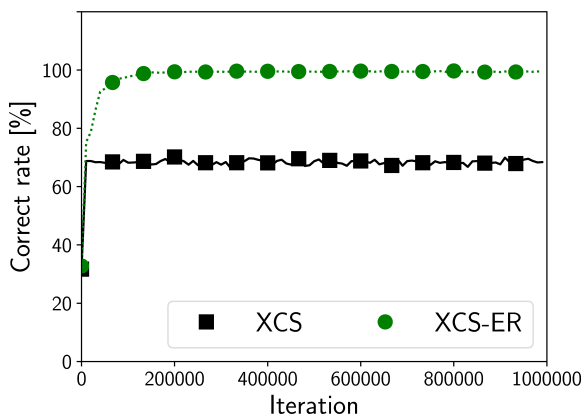


(a) 正答率の平均

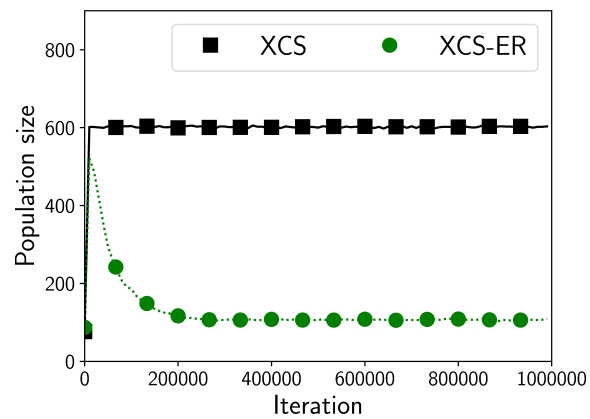


(b) 分類子数の平均

図 5.6 ケース U: $P_I = 0.05$

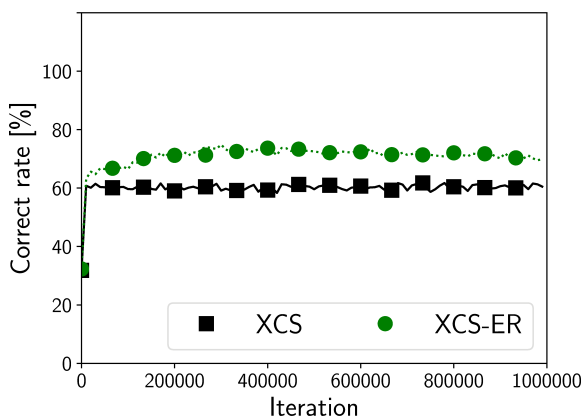


(a) 正答率の平均

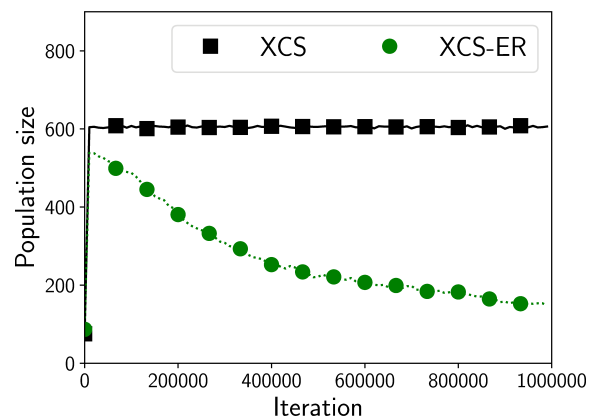


(b) 分類子数の平均

図 5.7 ケース U: $P_I = 0.1$

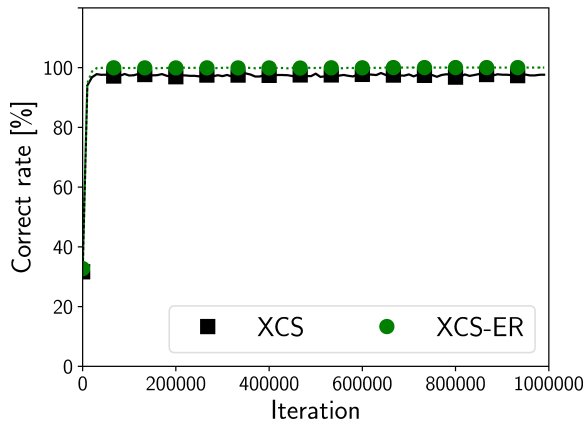


(a) 正答率の平均

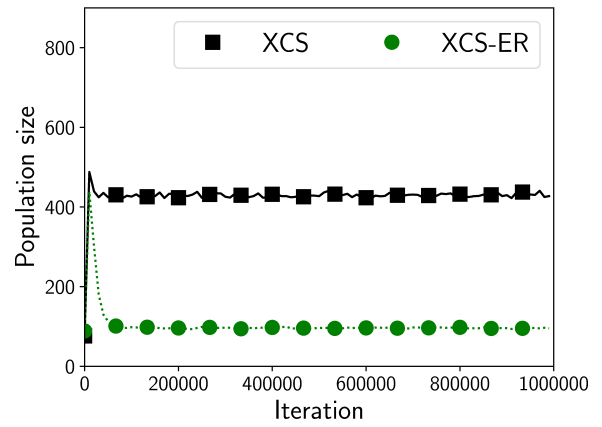


(b) 分類子数の平均

図 5.8 ケース U: $P_I = 0.2$

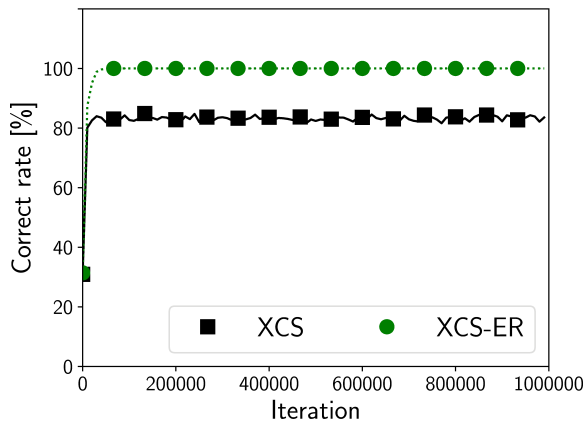


(a) 正答率の平均

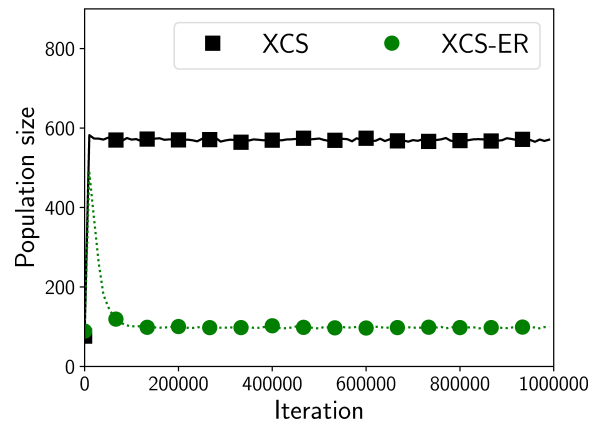


(b) 分類子数の平均

図 5.9 ケース V: $\sup\{P_I\} = 0.05$

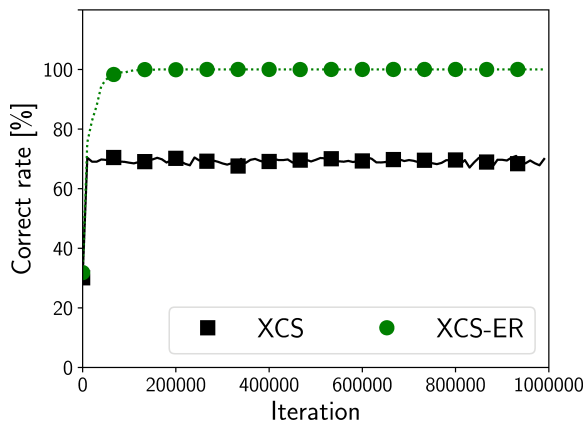


(a) 正答率の平均

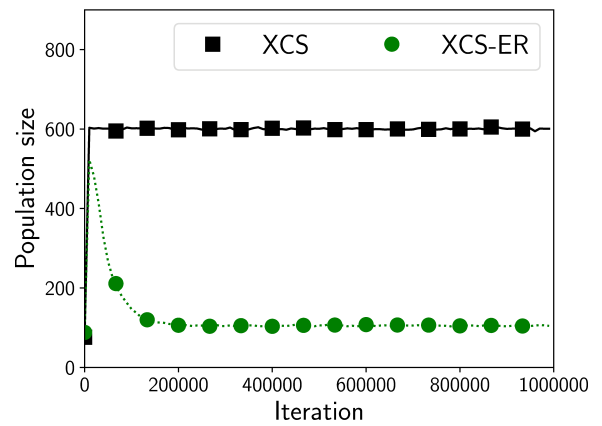


(b) 分類子数の平均

図 5.10 ケース V: $\sup\{P_I\} = 0.1$



(a) 正答率の平均



(b) 分類子数の平均

図 5.11 ケース V: $\sup\{P_I\} = 0.2$

表 5.5 ケース V: $\sup\{P_I\} = 0.05$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	52.34
XCS-ER	50 (100%)	49 (98%)	32.00

表 5.6 ケース V: $\sup\{P_I\} = 0.1$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	58.08
XCS-ER	50 (100%)	50 (100%)	32.00

表 5.7 ケース V: $\sup\{P_I\} = 0.2$ における最適解の獲得

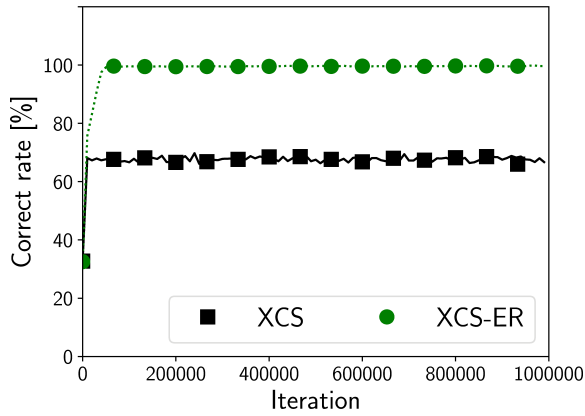
手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	52.40
XCS-ER	50 (100%)	49 (98%)	32.00

100 から明らかに増加した値を取り，どの試行においても [O] を獲得できず最大 F の分類子数の平均も 32 よりも大きい．一方で XCS-ER は $P_O = 0.4$ (ケース U) を除き正答率の平均は 100% 近くを維持し，分類子は 100 程度であり，ほとんどの試行で [O] を獲得し最大 F の分類子数の平均が 32 に近い値を取っていた．全ての場合において XCS-ER は XCS よりも正答率の平均は高く，分類子数の平均は少なかった．($p < 0.01$)

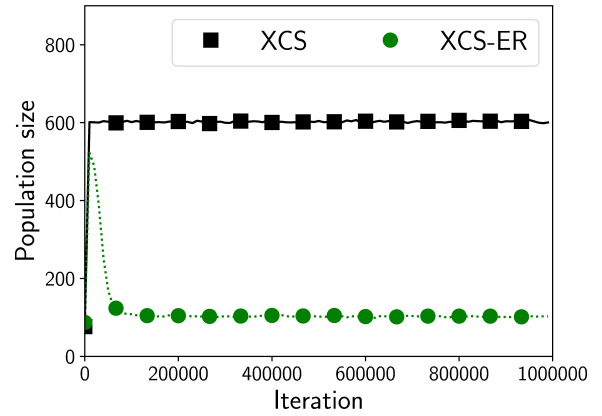
両不確実性においてケース U とケース V の比較から XCS-ER は不確実性 P_I, P_O の最大値ではなく平均的な不確実性に適応範囲が依存することが分かる．平均的な不確実性が $P_I = 0.1$ あるいは $P_O = 0.3$ までであれば XCS-ER は [O] を獲得する事が可能である．

これまでの結果から XCS-ER は入力における不確実性および出力における不確実性を伴う環境に対応可能なため，入力における不確実性と出力における不確実性の組み合わせ ($(P_I, P_O) = (0.05, 0.2)$) における結果について述べる．正答率の平均と分類子数の平均を図 5.18 に示す．図の横軸は Iteration を表し，縦軸は正答率の平均あるいは分類子数の平均を表す．最適解集合 [O] を獲得した試行数と最大 F の分類子数の平均を表 5.14 に示す．

入力における不確実性あるいは出力における不確実性が単体で伴う場合の結果と同様に，XCS は正答率の平均は 100% まで達しておらず，分類子数の平均は不確実性を伴わない環境における値である 100 から明らかに増加した値を取り，どの試行においても [O] を獲得できず最大 F の分類子数の平均も 32 よりも大きい．さらに，XCS-ER は正答率の平均は 100% に達し，分類子は 100 程度であり，全ての試行で [O] を獲得し最大 F の分類子数の平均が 32 に近い値を取っていた．XCS-ER は XCS よりも正答率の平均は高

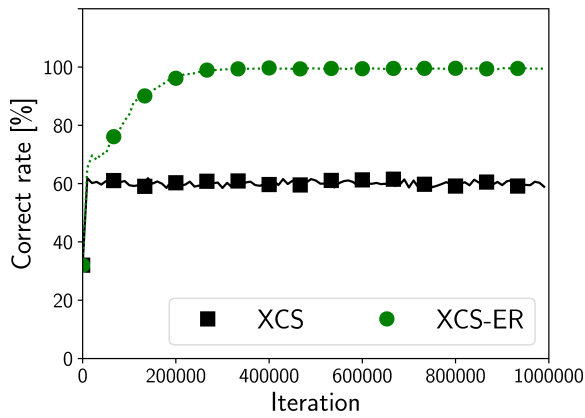


(a) 正答率の平均

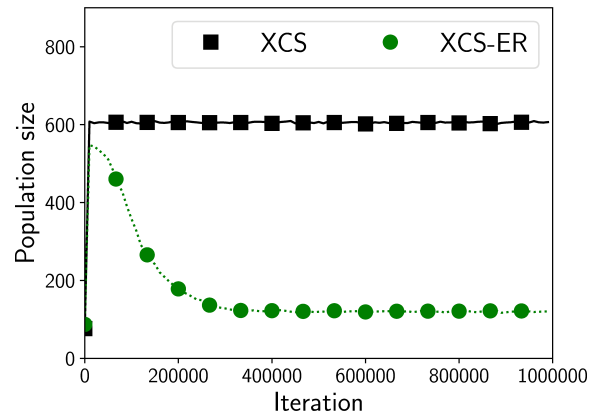


(b) 分類子数の平均

図 5.12 ケース U: $P_O = 0.2$

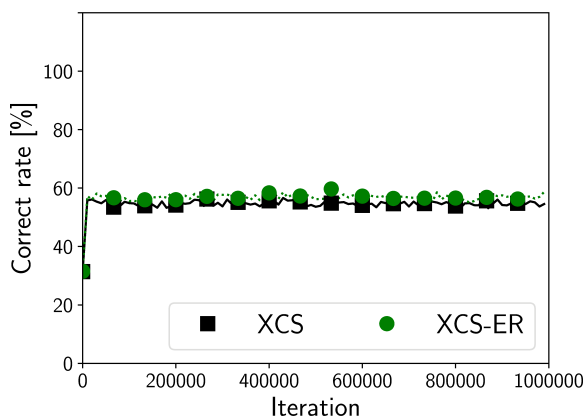


(a) 正答率の平均

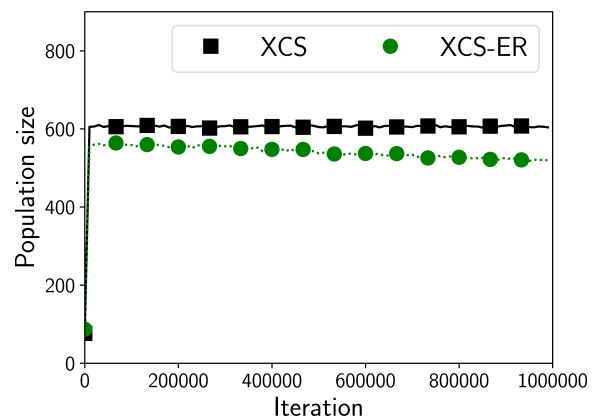


(b) 分類子数の平均

図 5.13 ケース U: $P_O = 0.3$

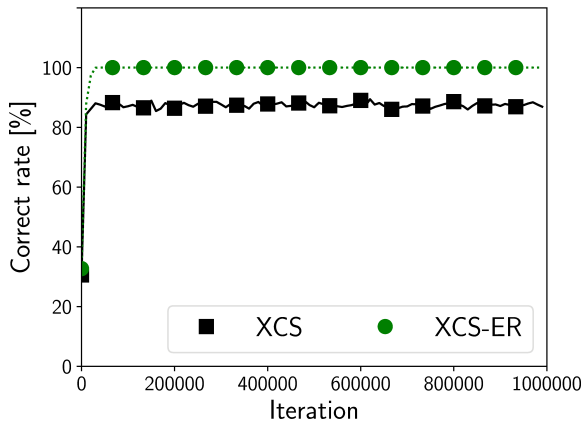


(a) 正答率の平均

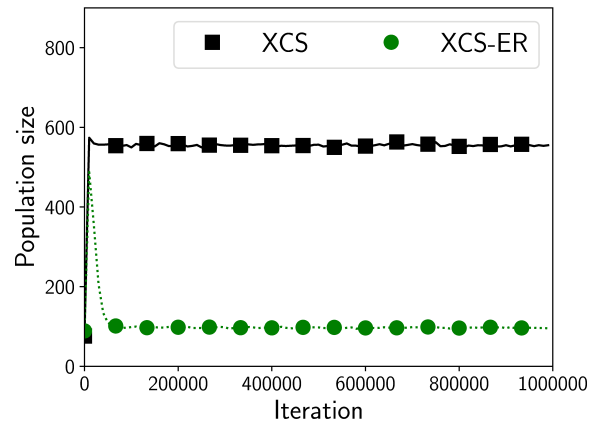


(b) 分類子数の平均

図 5.14 ケース U: $P_O = 0.4$

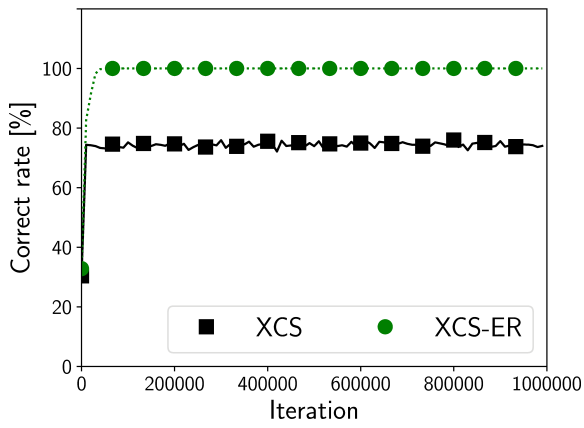


(a) 正答率の平均

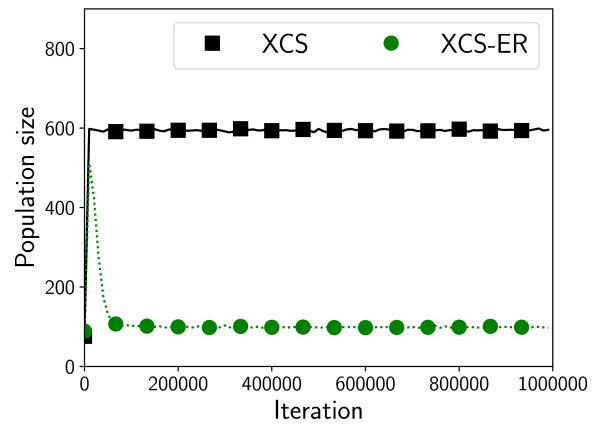


(b) 分類子数の平均

図 5.15 ケース V: $\sup\{P_O\} = 0.2$

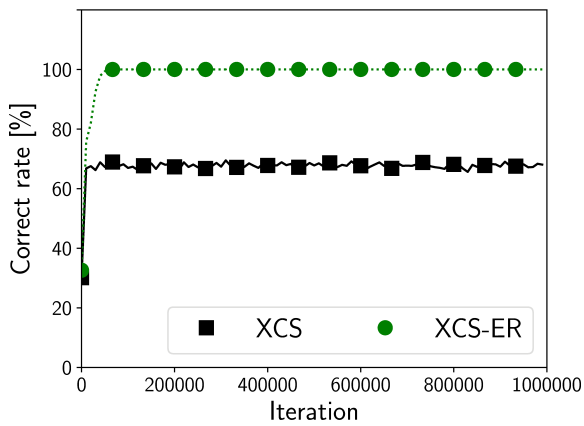


(a) 正答率の平均

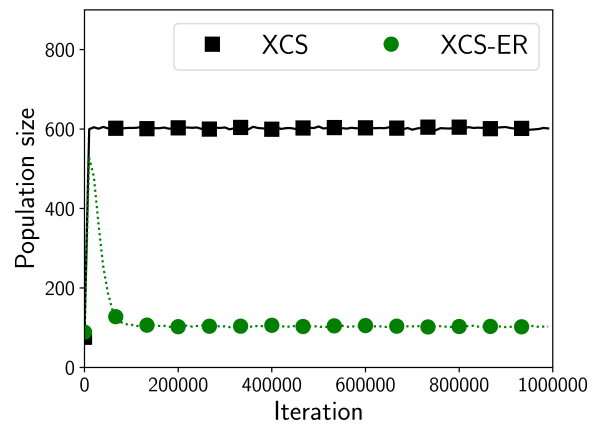


(b) 分類子数の平均

図 5.16 ケース V: $\sup\{P_O\} = 0.3$



(a) 正答率の平均



(b) 分類子数の平均

図 5.17 ケース V: $\sup\{P_O\} = 0.4$

表 5.8 ケース U: $P_O = 0.2$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	52.62
XCS-ER	49 (98%)	47 (94%)	31.76

表 5.9 ケース U: $P_O = 0.3$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.78
XCS-ER	46 (92%)	32 (64%)	31.54

表 5.10 ケース U: $P_O = 0.4$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	47.06
XCS-ER	0 (0%)	0 (0%)	33.54

表 5.11 ケース V: $\sup\{P_O\} = 0.2$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	59.14
XCS-ER	50 (100%)	50 (100%)	32

表 5.12 ケース V: $\sup\{P_O\} = 0.3$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	53.52
XCS-ER	50 (100%)	50 (100%)	32

表 5.13 ケース V: $\sup\{P_O\} = 0.4$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	50.96
XCS-ER	50 (100%)	50 (100%)	32

く, 分類子数の平均は少なかった. ($p < 0.01$)

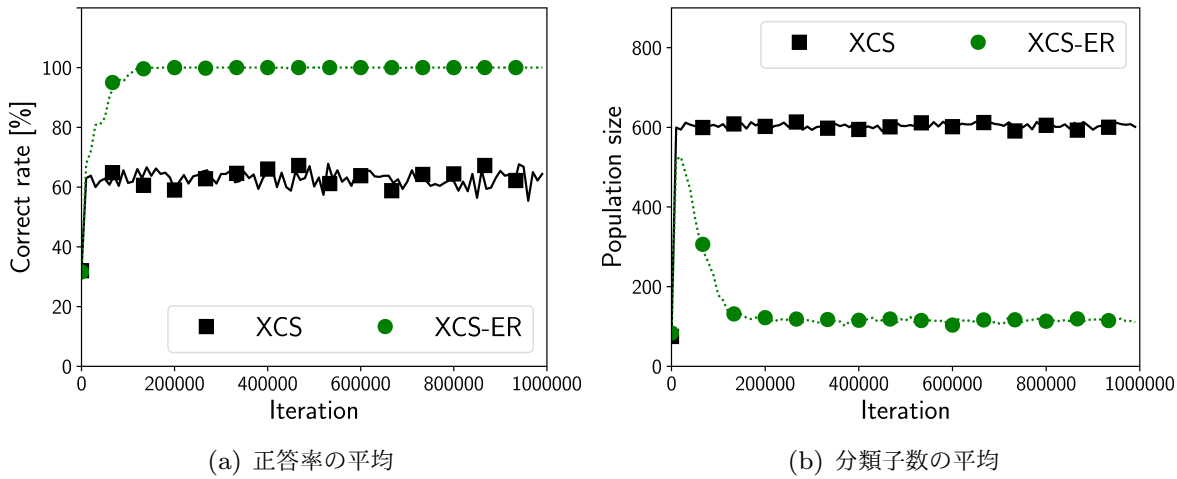


図 5.18 $(P_I, P_O) = (0.05, 0.2)$

表 5.14 $(P_I, P_O) = (0.05, 0.2)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	49.8
XCS-ER	5 (100%)	4 (80%)	32.2

5.6 考察

5.6.1 XCS の入出力における不確実性への不適合

XCS が入出力における不確実性に適応できない原因は式 (3.3), (3.6) による ϵ の算出と分類子の正確性の判定が式 (3.8) で表される様に予め定められた ϵ_0 によることである。図 5.19 に $P_I = 0.05$ における正確で正しい分類子 $\{C : 0001####, A : 1\}$ の ϵ の値の遷移を示す。縦軸は ϵ , 横軸は更新回数 exp を表す。式 (3.3) および (3.6) から ϵ はそれまでの ϵ を獲得した報酬 ρ に近づく様に更新される。入出力における不確実性を伴う問題において、正確な分類子であっても報酬 ρ は 0 および 1000 を取る。そのため変動が大きくなり、収束しない。さらに ϵ_0 は予め定められる定数（本実験において $\epsilon_0 = 10$ と設定）であるため、不確実性の強度（本実験における P_I および P_O ）についての情報が無い場合では適切に設定することができない。不正確な分類子の ϵ も同様に変動が大きいため、 ϵ_0 を大きな値に設定すると誤って正確と判定する場合が生じる。安定して正確と判定される分類子がないため、XCS は不確実性を伴う問題に適応し一般化した分類子を獲得できない。式 (3.8) から ϵ が小さいほど $\kappa(d)$ は大きくなるつまり、 F も大きくなるため、不確実性を伴わない場合に不正確と判定されるべき分類子よりも正確と判定されるべ

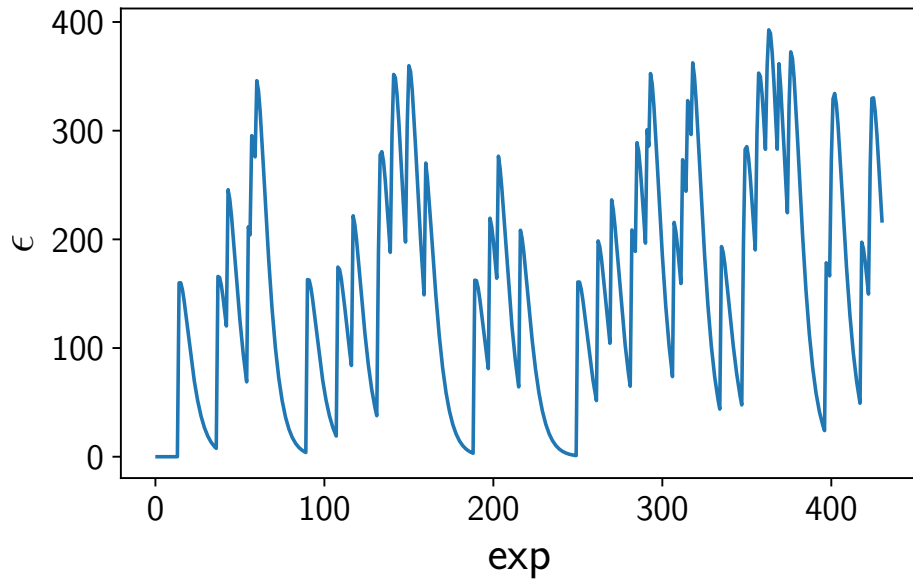


図 5.19 XCS における正確な分類子の ϵ

き分類子の方が $[P]$ に残りやすいため、正答率の平均の悪化（低下）は分類子数の平均の悪化（増加）よりも不確実性の影響を受けにくい。

5.6.2 XCS-ER の報酬推測の有効性とその限界

XCS-ER が入出力における不確実性を伴う問題において一般化した分類子を獲得しそれを保持し続けられるのは、強化部に導入した報酬推測機構によるものである。 $[M]$ に含まれる分類子はその時の入力に照合する分類子である。そのため、各分類子が照合した入力における評価される回数を合計すると、入力によってその出現頻度が大きく異なる限り、その入力において評価された回数が多くなる。図 5.20 の例では、分類子 $\{C : \#011, A : 0\}$ は入力 0011 および 1011 において評価され、分類子 $\{C : 001\#, A : 0\}$ は入力 0010 および 0011 において評価され、分類子 $\{C : \#0\#1, A : 0\}$ は入力 0001, 0011, 1001 および 1011 において評価される。これら 3 つの分類子に共通する入力は 0011 のみである。加重平均から計算される $CR_{A=0}$ は入力 0011 における評価の影響が強くなる。また、XCS-ER の ϵ_0 の初期値は小さく設定されるため、すべての分類子の ϵ が ϵ_0 を超えて不正確と判断されたとしても、不正確と判定されるべき分類子の ϵ は正確と判定されるべき分類子の ϵ よりも大きいため、不正確な分類子の F は正確な分類子の F よりも小さい。 F が小さい分類子から優先的に削除されるため $[P]$ に不正確と判定されるべき分類子は残りにくく正確と判定されるべき分類子は残りやすい。つまり正確な分類子ほど exp が大きい傾向がある。そのため評価回数が少なく正確性を判定できない分類子や不正確な分類子の影響が小さくなり、その時の入力における獲得報酬を推測可能である。

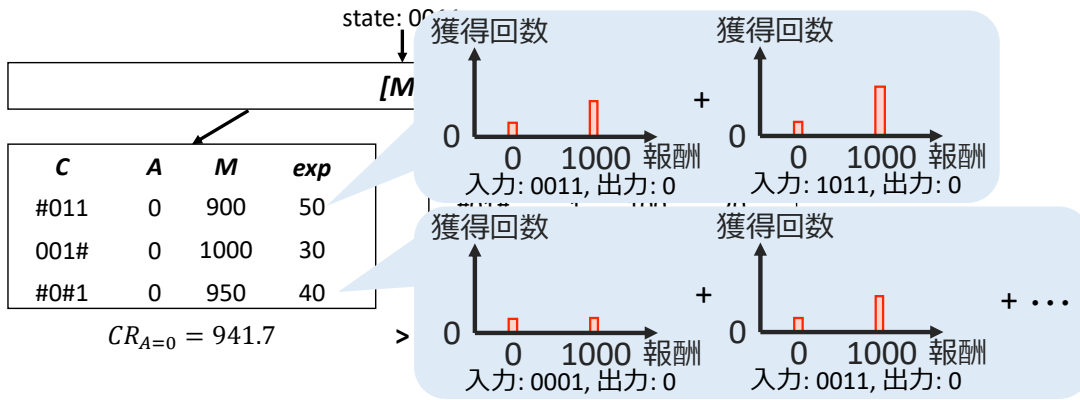
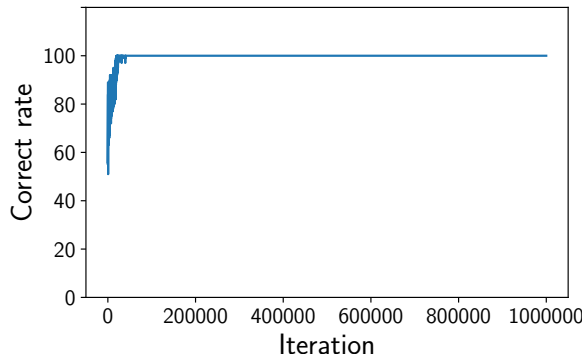


図 5.20 XCS-ER の報酬推測機構

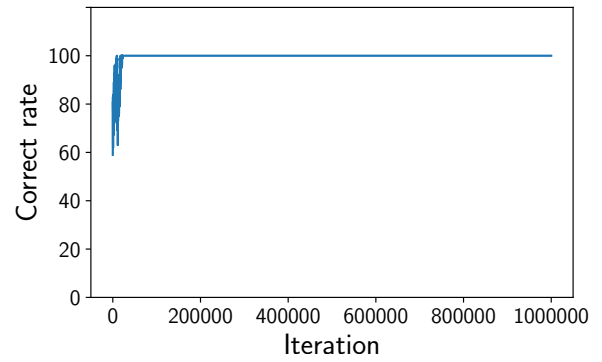
図 5.21(a) および 5.21(b) に $P_I = 0.05$ および $P_O = 0.2$ における $[O]$ を獲得した試行の XCS-ER に入力された状態に対する報酬推測機構の報酬推測の正答率を示す。この推測率は評価指標の正答率の平均と同様に 100Iteration で移動平均を取ったものである。縦軸は報酬推測の正答率、横軸は Iteration を表す。 $P_I = 0.05$ においても $P_O = 0.2$ においても正答率は 100% に達している。このことから報酬推測機構の報酬推測によって XCS-ER は入力された状態に対して正解となる出力を選択することができることが分かる。

また、表 5.8 から XCS-ER が $[O]$ を獲得できていない試行が 1 つある。この場合における報酬推測の正答率を図 5.22 に示す。この試行において $[P]$ は過剰に一般化された分類子に占められており、報酬推測機構が想定する $[P]$ の大半が正確な分類子で占められているという仮定を満たしていなかった。これは、学習の初期において不確実性の影響が連続して不正確な分類子であっても正確な分類子と判定されたため、報酬推測機構がその分類子の影響を強く受けたことが理由である。表 5.2 および 5.8 から、連続的な不確実性の影響への耐性を高めることが好ましいがほとんどの試行において $[O]$ を獲得しているため、XCS-ER が導入した報酬推測機構は $[O]$ を獲得するには十分な正答率を有していると言える。

これらの結果から XCS-ER は入力における不確実性を伴う場合には $P_I = 0.05$ までであれば全ての試行で、 $P_I = 0.1$ までであれば大半の試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し $[O]$ を獲得する。出力における不確実性を伴う場合には $P_O = 0.3$ までであれば大半の試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し $[O]$ を獲得する。



(a) $P_I = 0.05$ における報酬推測の正答率



(b) $P_O = 0.2$ における報酬推測の正答率

図 5.21 報酬推測の正答率

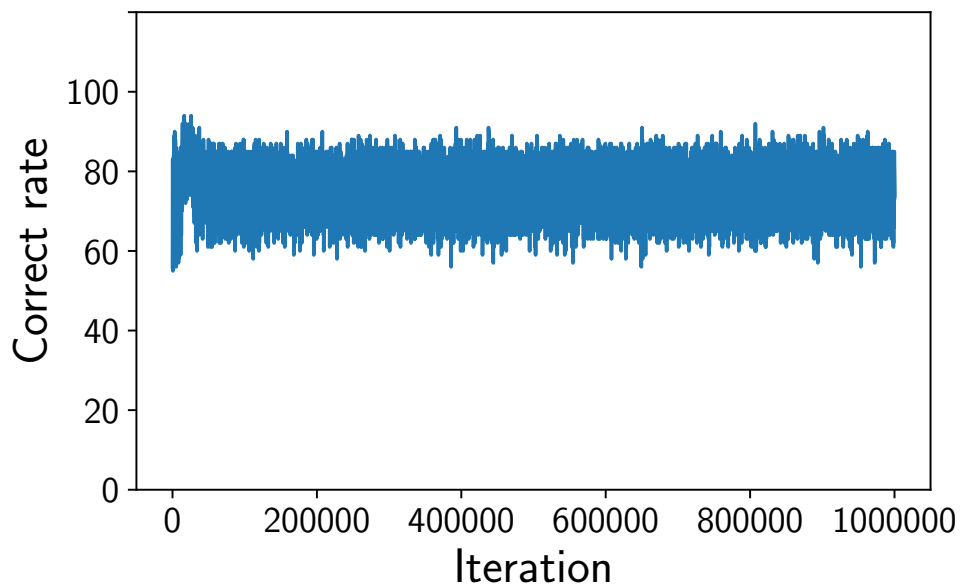
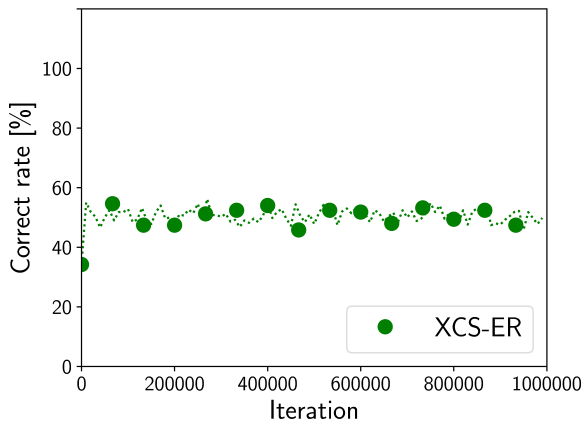


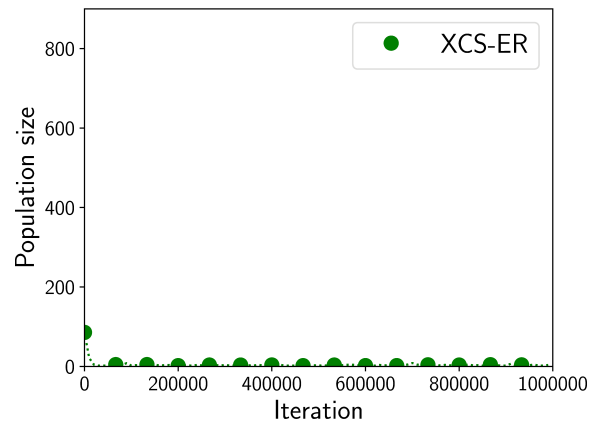
図 5.22 $P_O = 0.2$ における $[O]$ を獲得できない場合の報酬推測の正答率

5.6.3 XCS-ER の新設されたパラメータ

XCS-ER はハイパーパラメータ θ_{RE} を持つ。この節では θ_{RE} の設定について述べる。図 5.23 から 5.26 に出力における不確実性 ($P_O = 0.2$) を伴う場合に θ_{RE} を 0, 2, 5, 10 に設定した際の正答率の平均と分類子数の平均を示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。 $\theta_{RE} = 0$ の場合において正答率の平均が約 50% で分類子数の平均が 2 と分類子が過剰に一般化されている。 $\theta_{RE} = 0$ では評価回数が少なくても ϵ_0 のための $Max\epsilon$ の計算対象に入る。そのため、不正確な分類子が評価回数が少ない内に $Max\epsilon$ の対象となり ϵ_0 が大きく設定され、不正確な分類子が正確と

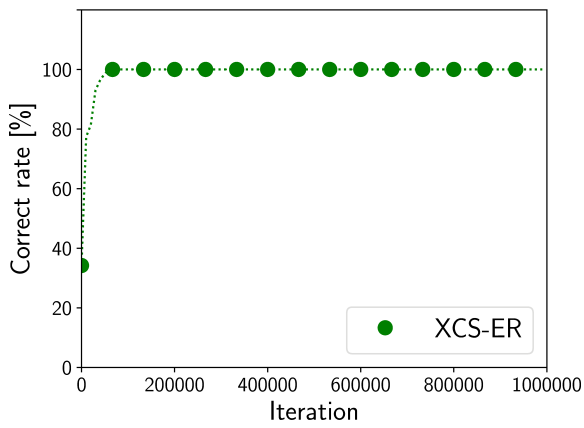


(a) 正答率の平均

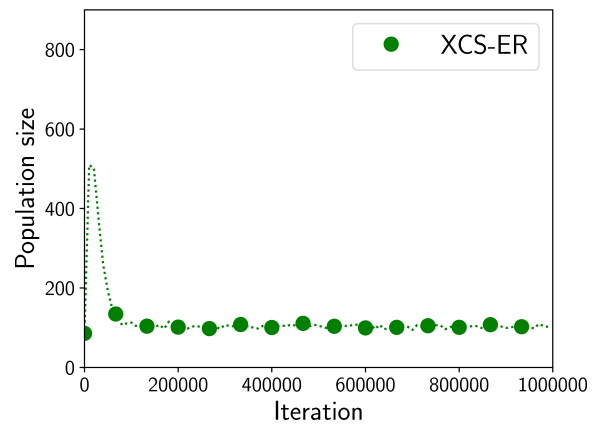


(b) 分類子数の平均

図 5.23 $\theta_{RE} = 0$

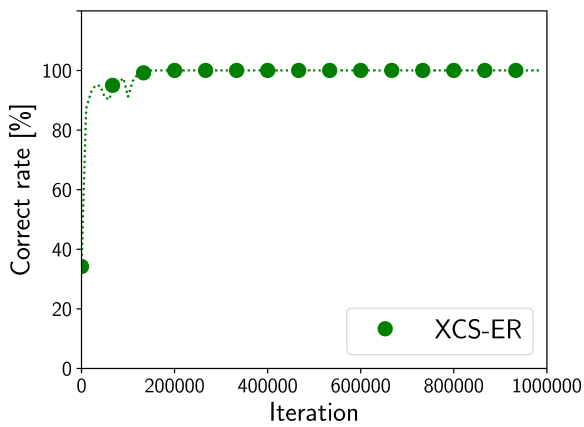


(a) 正答率の平均

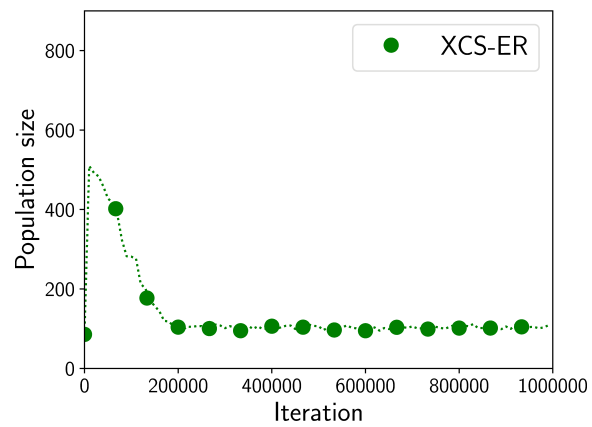


(b) 分類子数の平均

図 5.24 $\theta_{RE} = 2$

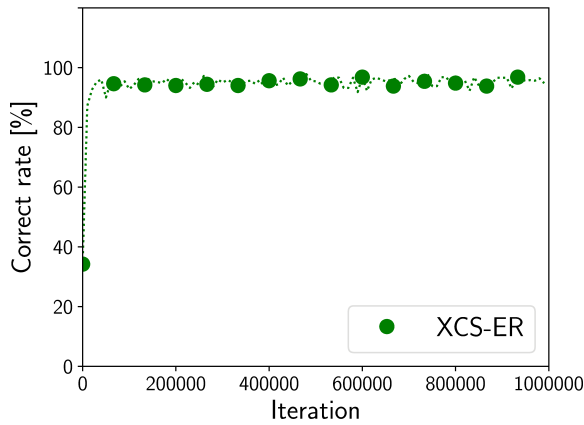


(a) 正答率の平均

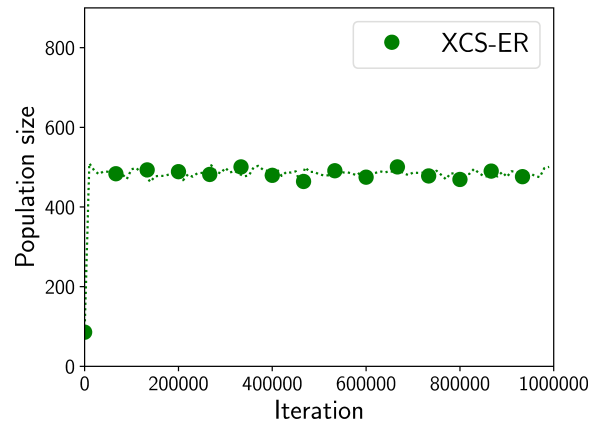


(b) 分類子数の平均

図 5.25 $\theta_{RE} = 5$



(a) 正答率の平均



(b) 分類子数の平均

図 5.26 $\theta_{RE} = 10$

判定される。

$\theta_{RE} = 2, 5$ の場合において、収束速度に若干の違いが見られるもののどちらも正答率の平均が 100% に達し、分類子数の平均は 100 程度と適切に学習ができています。

$\theta_{RE} = 10$ の場合において正答率の平均は 100% には届かず、分類子数の平均が 500 程度と分類子の一般化が行われていない。 θ_{RE} が大きいと分類子が包摂元になる条件を満たすまでに多くの評価回数を要することになり、正確な分類子による包摂が行われず n が増えにくく、包摂が行われるときに比べ削除されやすくなる。また、 θ_{RE} が大きいと過剰な一般化が行われず θ_{RE} が小さい場合と比べて正答率の平均の低下度合は大きくない。これらの結果から θ_{RE} を 2 から 5 程度の値に設定するのが好ましい。

第 6 章

判定の信憑性における不確実性に対応可能な学習分類子システム

判定の信憑性における不確実性において図 4.4 からシステムに与えられる報酬の種類は 0 を中心としてばらつく分布と R を中心としてばらつく分布であり、正確な分類子であればどちらか一方の分布のみに従う報酬を獲得するが、不正確な分類子であれば獲得する報酬は両方の分布に従うものになることに着目し、報酬の範囲に基づく学習分類子システム (XCS based on Range of Reward: XCS-RR) を提案する。

6.1 判定の信憑性における不確実性に対するアプローチ

図 6.1 は判定の信憑性における不確実性を伴った場合の入出力に対する判定とそれに基づく XCS の正確性判定および、目標とする不確実性に対処可能な XCS の正確性判定を示す。上部では判定の信憑性における不確実性を伴う場合のデータを例示しており、全てのデータにおいて不確実性により判定の値に差異がある。このデータにおいて XCS は正

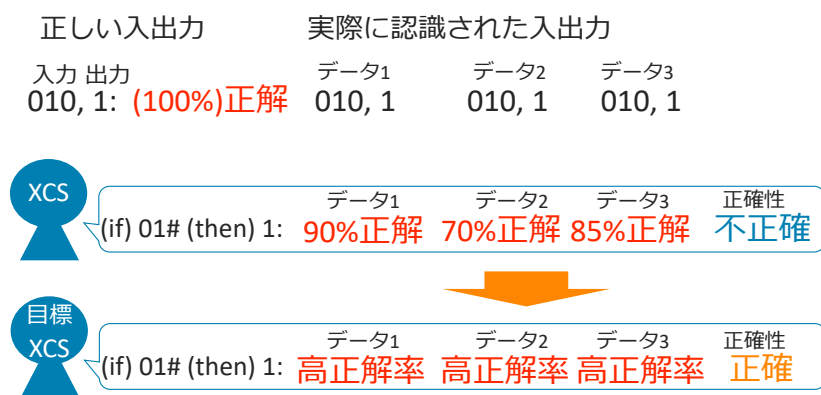


図 6.1 判定の信憑性における不確実性を伴う場合

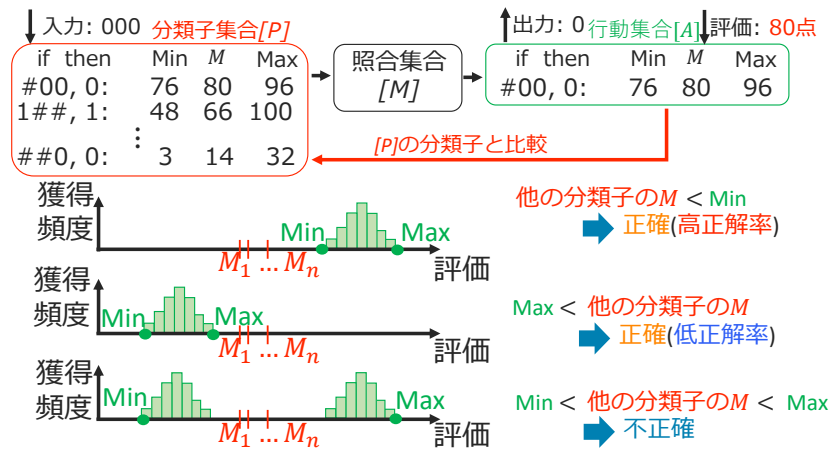


図 6.2 XCS-RR の概要

確な分類子 $\{C : 01\#, A : 0\}$ が入力 “010” のみに照合するにもかかわらず，判定の値がばらつき不正確と評価される．特に 0 と R の間の値を取る報酬では正解なのか不正解なのかを判断することが難しい．

この問題を解決するため，判定の信憑性における不確実性に対し，ひとつひとつの報酬で分類子の正確性を判断するのではなく，分類子が獲得した報酬全体の分布（形状・最大値や最小値）つまり，分類子が獲得した報酬の範囲に基づいて正確性を判定するアプローチを取る．正確な分類子であれば，報酬は 0 付近のみあるいは R 付近のみの一方を獲得する．また，不正確な分類子が獲得する報酬は 0 付近および R 付近の両方を獲得する．そのため，正確な分類子が獲得する報酬の範囲は不正確な分類子が獲得する報酬の範囲よりも狭い．XCS は獲得報酬が 1 種類である分類子のみを正確と判定するが，獲得報酬の範囲が狭い分類子を正確と判断できれば判定の信憑性における不確実性を伴う場合においても，正しく分類子の正確性を判定できる．

図 6.2 に XCS-RR の概要を示す．評価対象の $[A]$ のルールをその時点での $[P]$ にある分類子と比較し，評価対象の分類子が獲得した報酬全てが大きめなのか小さめなのかにより正確性を判定する．以下に，XCS-RR の詳細を記す．

6.2 構成

6.2.1 アーキテクチャ

XCS-RR の XCS との差分は強化部のパラメータ更新と発見部の包摂である．図 6.3 の黒塗りの部分はその差分を表す．また，新設されたパラメータおよび異なる方法で更新されるパラメータを明記している．分類子が獲得した報酬全体に関する情報（最大値および最小値）は報酬を獲得するたびに更新する必要があるため，パラメータ更新に組み

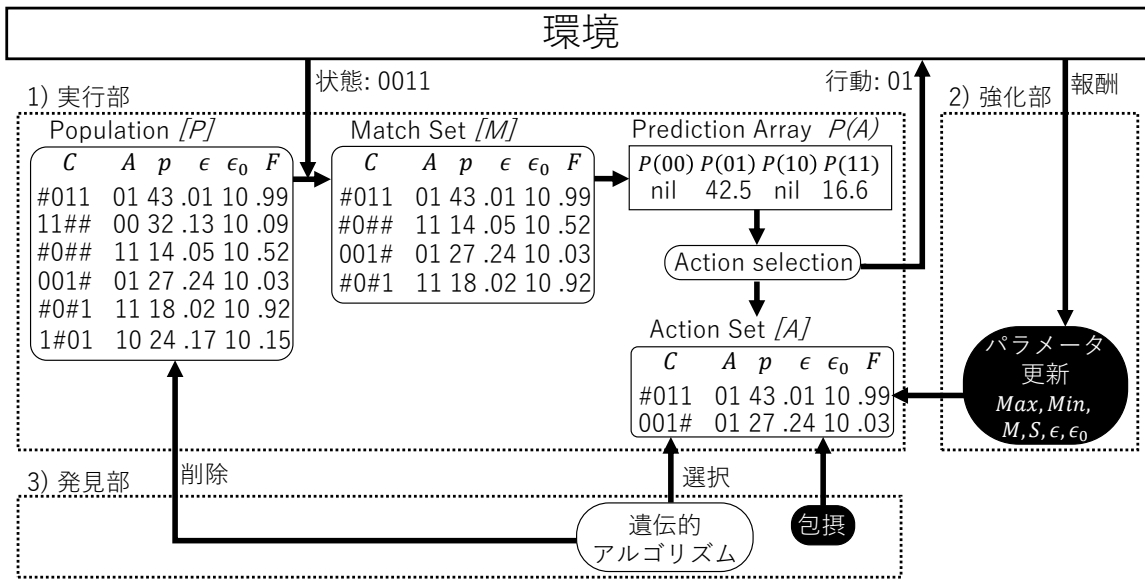


図 6.3 XCS-RR のアーキテクチャ

込む。

6.2.2 分類子

XCS-RR の分類子は XCS の分類子が持つ $C, A, p, \epsilon, F, exp, ts, as, n$ に加えて、獲得した報酬の最大値、最小値、平均値、標本標準偏差 Max, Min, M, S を記録するパラメータを持つ。さらに第 5 章の XCS-RE と同様に分類子の正確性の基準となる許容誤差 ϵ_0 を持つ。

6.3 学習メカニズム

6.3.1 強化部

パラメータ更新

分類子が獲得した報酬全体の分布に関する情報を更新した後に、それを用いて分類子の正確性を判定する。XCS-RR は報酬を獲得すると [A] を構成するすべての分類子において報酬の範囲およびそのばらつきを把握するために、それぞれの分類子 cl がこれまでに獲得した報酬から求まる最大値 $cl.Max$ 、最小値 $cl.Min$ 、平均 $cl.M$ 、標本標準偏差 $cl.S$ を更新する。 $cl.M$ を

$$cl.M \leftarrow (reward - cl.M) / cl.exp \quad (6.1)$$

と $cl.S$ を

$$cl.S \leftarrow \sqrt{(cl.exp - 1) \times (\rho - cl.M)^2 / cl.exp} \quad (6.2)$$

と逐次的に求める．この $cl.S$ の値を $cl.\epsilon$ とする．

上記で求めた， $cl.Max, cl.Min, cl.M$ を用いて獲得した報酬の範囲が狭い分類子を探し，その分類子を正確な分類子であると見なし， ϵ_0 を定める．まず $[P]$ の中で， exp が閾値 θ_{sub} よりも多い分類子を抽出し $[P']$ を生成する．評価回数が少ないと適切にその分類子が獲得する報酬の分布（特に最大値および最小値）を正しく求めることができないため，閾値による制限を設けた． $[P']$ を構成する分類子数を $|[P']|$ と表す． $[A]$ を構成する分類子の内 exp が θ_{sub} 以上の分類子 cl が式 (6.3) と式 (6.4) のどちらか一方を $|[P']|$ の閾値 $\theta_{RR}\%$ 以上の回数満たした場合にその分類子 cl は正確であるとみなす． $[P']$ には正確で正解を導く分類子，正確で不正確を導く分類子および不正確な分類子の 3 種類の分類子が混在している．評価対象の分類子 cl が正確で正解を導く分類子であれば正確で不正確を導く分類子と，正確で不正確を導く分類子であれば正確で正解を導く分類子と下記の条件を満たすことを期待する．

$$(cl.Max < cl'.M) \wedge \left(cl.Max < \frac{cl'.Max + cl'.Min}{2} \right) \quad (6.3)$$

$$(cl.Min > cl'.M) \wedge \left(cl.Min > \frac{cl'.Max + cl'.Min}{2} \right) \quad (6.4)$$

ただし， $cl' \in [P']$ である．式 (6.3) は分類子 cl の最大値が他の分類子 cl' の平均よりも小さく範囲が狭いときに満たし，式 (6.4) は分類子 cl の最小値が他の分類子 cl' の平均よりも大きく範囲が狭いときに満たす．図 6.4 に正確な分類子の識別法を図示する．一番上の分類子は R 付近の値の報酬のみを獲得し，真ん中の分類子は 0 付近の値報酬のみを獲得し，一番下の分類子は 0 付近および R 付近の値の報酬を獲得している．一番上の分類子は獲得した報酬の Min および Max は共に他のルールを獲得した報酬の平均 μ_B, μ_C よりも大きい．真ん中の分類子は獲得した報酬の Min および Max は共に他のルールを獲得した報酬の平均 μ_A, μ_C よりも小さい．このことから 2 つの分類子の獲得した報酬の範囲は狭いため，両分類子は正確と判定される．一方で一番下の分類子は Min は他のルールを獲得した報酬の平均 μ_A, μ_B よりも小さく， Max は他のルールを獲得した報酬の平均 μ_A, μ_B よりも大きいため，獲得した報酬の範囲が広いと分かる．このことからこの分類子は不正確な分類子と判定される．正確な分類子の識別法のアルゴリズムを Algorithm 10 に示す．2 行目から 6 行目で $[P']$ となる分類子を抜き出す．10 行目で式 (6.3) の条件，11 行目で式 (6.4) の条件で分類子を比較する．17 行目で分類子 cl が式 (6.3) の条件あるいは式 (6.4) の条件を満たした回数により cl の正確性を決める．

上記の条件により正確とみなされた分類子 cl の中で最大の S を $MaxS$ とし， $[A]$ を構成する分類子の ϵ_0 を

$$cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(MaxS + E_0 - cl.\epsilon_0) \quad (6.5)$$

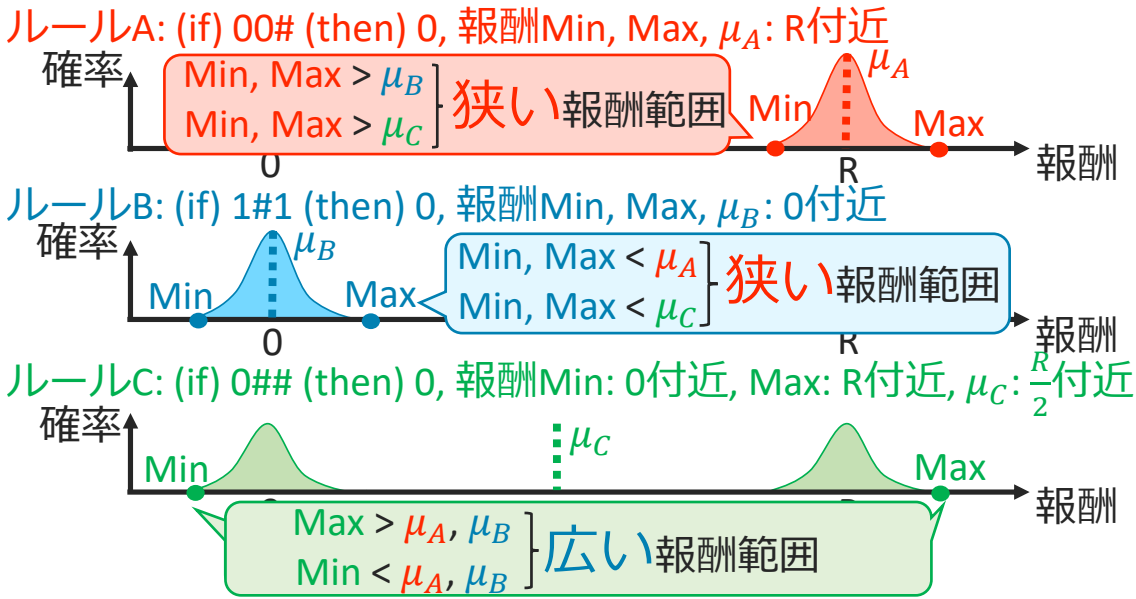


図 6.4 正確な分類子の識別

と更新し、条件を満たした分類子全てを正確と扱う。また、 ϵ_0 を $MaxS$ から定数 E_0 だけ大きくすることにより、標本標準偏差のばらつきを考慮した設定にする。

6.3.2 発見部

包摂

包摂する分類子は包摂される分類子よりも一般化度合（#の数）が高いため、包摂される分類子が照合する入力に加えて他の入力にも照合する。[A]にある包摂される分類子の評価回数よりも大きな評価回数を包摂する分類子に求めることで多くの入力において評価されることを担保する。よって、包摂元として分類子が選択される条件に包摂対象となるいずれの分類子の評価回数よりも大きいという条件を加え、

- $cl.exp > \theta_{sub}$
- $\kappa = 1$
- $cl.exp > \sup\{c.exp, c \text{ could subsumed } cl\}$

とする。ただし、 $c \text{ could subsumed } cl$ は分類子 cl が分類子 c を包摂できる事を表す。

6.4 アルゴリズム

XCS-RR のアルゴリズムを Algorithm 11 に示す。下線が引かれている部分が XCS からの変更点である。XCS-RR は環境から入力された状態に照合する分類子を [P] の中か

Algorithm 10 正確な分類子の識別法

```
1: initialize  $[P']$ :  $[P']$  を空に初期化
2: for all classifier  $cl$  in  $[P]$  do
3:   if  $cl.exp > \theta_{sub}$  then
4:     add to  $[P']$ :  $[P']$  に追加
5:   end if
6: end for
7:  $countCorrect = countIncorrect \leftarrow 0$ 
8: for all classifier  $cl$  in  $[A]$  do
9:   for all classifier  $cl'$  in  $[P']$  do
10:    if  $(cl.Max < cl'.M) \wedge (cl.Max < (cl'.Max + cl'.Min)/2)$  then
11:       $countIncorrect \leftarrow countIncorrect + 1$ 
12:    end if
13:    if  $(cl.Min > cl'.M) \wedge (cl.Min > (cl'.Max + cl'.Min)/2)$  then
14:       $countCorrect \leftarrow countCorrect + 1$ 
15:    end if
16:  end for
17:  if  $countCorrect > |[P']| \times \theta_{RR}$  or  $countIncorrect > |[P']| \times \theta_{RR}$  then
18:     $cl$  is accurate
19:  else
20:     $cl$  is inaccurate
21:  end if
22: end for
```

ら抜き出し, $[M]$ を生成する (2, 3 行目). $P(\text{action})$ を基に決定した行動を有する分類子を $[M]$ から抜き出し, $[A]$ を生成する (4 から 6 行目). 選択した行動を環境に対して出力し得られた報酬を用いて $[A]$ の分類子のパラメータを更新し学習を行う. 9, 10, 11 行目は分類子 cl が獲得した報酬の最大値や最小値, 平均値および標本標準偏差を求め, 12 行目で標本標準偏差を ϵ としている. 15 行目は Algorithm 10 の正確な分類子の識別法である. 16 行目から 18 行目で Algorithm 10 によって正確とみなされた分類子の中で最大の ϵ を求めている. その値を用いて 20 行目で ϵ_0 を更新する. その後, 包摂条件や GA の条件を満たした場合に $[A]$ の分類子を対象として, 包摂 (29 行目) や GA (32 行目) を行う. 包摂条件では, 包摂を行う分類子が包摂をする分類子のいずれよりも評価回数が多いかを確認する (24 行目). 上記の流れを 1 学習エピソードとして, 学習及び GA による生成と淘汰を繰り返すことによって, 環境に適した分類子集団を獲得する.

Algorithm 11 XCS-RR のアルゴリズム

```
1: while (! end of iterations) do
2:    $state \leftarrow$  environment: 環境から状態  $state$  を受け取る
3:   generate  $[M]$ :  $[P]$  から  $state$  に照合する分類子を選択
4:    $P(a_i) = \sum_{cl_k \in [M]|a_i} cl_k \cdot p \times cl_k \cdot F / \sum_{cl_l \in [M]|a_i} cl_l \cdot F$ :  $[M]$  から各行動の予測値を算出
5:    $action \leftarrow P(action)$ :  $P(action)$  から行動を選択
6:   generate  $[A]$ :  $[M]$  から  $action$  に照合する分類子を選択
7:    $reward \leftarrow$  execute  $action$ : 行動を実行し, 環境より報酬を獲得
8:   for all classifier  $cl$  in  $[A]$  do
9:     update MinValue and MaxValue: 最大値および最小値の更新
10:     $cl.M \leftarrow (reward - cl.M) / cl.exp$ 
11:     $cl.S \leftarrow \sqrt{(c.exp - 1) \times (\rho - cl.M)^2 / cl.exp}$ 
12:     $cl.\epsilon = cl.S$ 
13:    update  $p, as, exp$ : パラメータ更新 (XCS と同様)
14:   end for
15:   identificate classifier accuracy: 正確な分類子の識別 (Algorithm 10)
16:   for all classifier  $cl$  in  $[A]$  do
17:     calculate  $MaxS$ : 条件を満たした分類子の中で最大の  $\epsilon$  を求める
18:   end for
19:   for all classifier  $cl$  in  $[A]$  do
20:      $cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(MaxS + E_0 - cl.\epsilon_0)$ 
21:     update  $\kappa(cl), \kappa'(cl), F$ : パラメータ更新 (XCS と同様)
22:   end for
23:   for all classifier  $cl$  in  $[A]$  do
24:     if  $cl.exp > \theta_{sub}$  and  $\kappa = 1$  and  $cl.exp > \sup\{c.exp, c \text{ could subsumed } cl\}$ 
25:       then
26:          $subsumptionFlag = true$ 
27:       end if
28:   end for
29:   if  $subsumptionFlag = true$  then
30:     run subsumption:  $[A]$  内の分類子を対象に包摂を実行
31:   end if
32:   if run GA then
33:     generate new classifiers: 子個体の生成
34:     delete classifier: 削除
35:   end if
36: end while
```

6.5 実験

6.5.1 実験内容

XCS-RR は判定の信憑性における不確実性に対応可能なため第 4.2.3 項の不確実性を扱う。不確実性の強度を表すパラメータが $\sigma_R = 0, 200, 700, 1500$ の時を扱う。また第 5 章における実験と同様に、不確実性の強度を以下の 2 種類のケースとした実験を行う。

- ケース U: すべての入力と出力の組において均一の不確実性強度
- ケース V: 入力と出力の組ごとに σ_R を上限とした不確実性強度

ケース U (Uniform) では不確実性強度の強弱による手法の適応性を評価し、ケース V (Various) では不確実性強度が均一でない場合における適応性を評価する。

6.5.2 実験設定

XCS で用いるパラメータは第 5.5.2 項で記した値と同様に $\theta_{mna} = 2$, $P_{\#} = 0.35$, $p_I = 0.01$, $\epsilon_I = 0.01$, $F_I = 0.01$, $p_{explr} = 1.0$, $\beta = 0.2$, $\epsilon_0 = 10$, $\alpha = 0.1$, $\nu = 10$, $\theta_{sub} = 20$, $\theta_{GA} = 25$, $\chi = 0.8$, $\mu = 0.04$, $N = 800$, $\theta_{del} = 20$, $\delta = 0.1$ である。XCS-RR で新設されたパラメータ θ_{RR} は入力長 L に基づいて $\theta_{RR} = 10$ と定めた。Iteration は 1,000,000 回と設定する。また、評価基準は第 4.3 節で説明した正答率の平均、分類子数の平均および最適解集合 $[O]$ を獲得した試行数と最大 F の分類子数の平均である。

6.5.3 実験結果

不確実性が伴わない場合 $\sigma_R = 0$ の時の正答率の平均と分類子数の平均を図 6.5 に、判定の信憑性における不確実性が $\sigma_R = 200, 700, 1500$ (ケース U) の時の正答率の平均と分類子数の平均をそれぞれ図 6.6, 6.7, 6.8 に、 $\sup\{P_I\} = 200, 700, 1500$ (ケース V) の時の正答率の平均と分類子数の平均をそれぞれ図 6.9, 6.10, 6.11 に示す。それぞれの図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。また、それぞれにおける最適解集合 $[O]$ を獲得した試行数と最大 F の分類子数の平均をそれぞれ表 6.1 - 6.7 に示す。

不確実性を伴わない $\sigma_R = 0$ 場合、XCS は入出力における不確実性を伴わない場合と同等なため省略する。XCS-RR は XCS と同様に正答率の平均 100%, 分類子数の平均は 98.1 であった。さらに、 $[O]$ を全ての試行において獲得し最大 F の分類子数の平均は 32 である。このことから、不確実性が伴わない環境において XCS-RR は分類子の正確性を正しく認識し分類子の一般化が可能である。

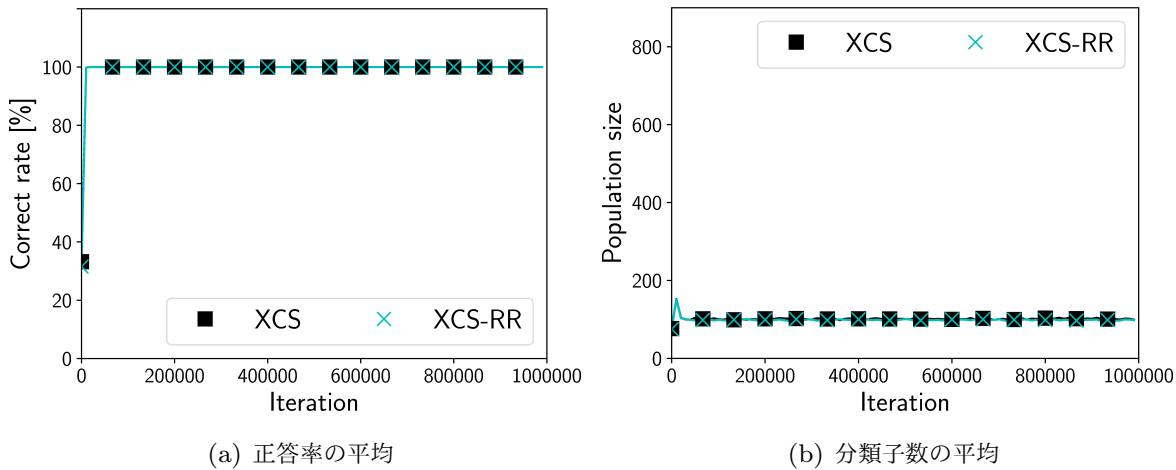
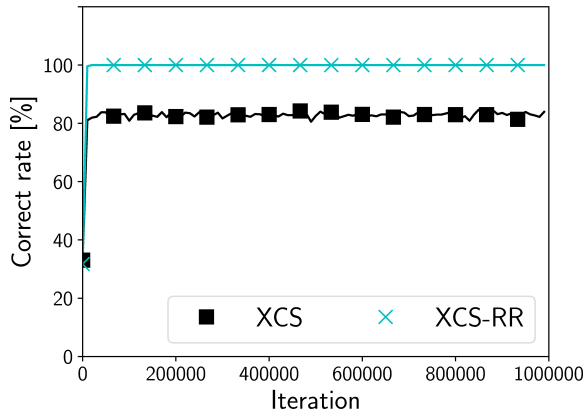


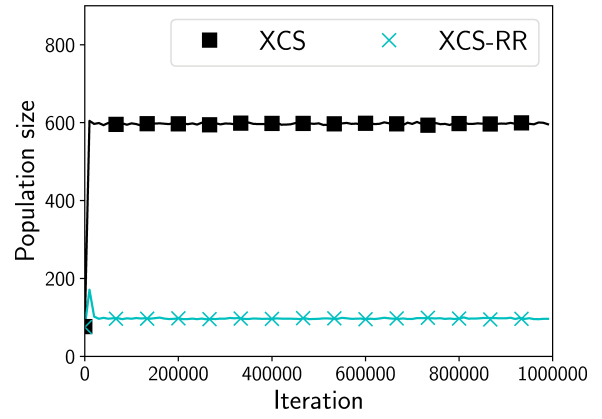
図 6.5 $\sigma_R = 0$

判定の信憑性における不確実性を伴う環境においての結果について述べる。XCS はケース U, B 問わず全ての場合において正答率の平均は 100% に達せず、分類子数の平均は不確実性を伴わない場合よりも明らかに増加したことに加え、[O] は全ての試行で獲得されておらず最大 F の分類子数の平均は 32 よりも大きい値を取っていた。XCS-RR はケース U において $\sigma_R = 200$ の場合には正答率の平均 100%, 分類子数の平均 96.5, 全ての試行において [O] を獲得し最大 F の分類子数の平均 32 であるので不確実性を伴わない場合と同等の結果を得た。この時、XCS よりも正答率の平均は高く、分類子数の平均は低い。($p < 0.01$) しかし、 $\sigma_R = 700, 1500$ と伴う不確実性の強度が高くなると正答率の平均が 6 割程度、分類子数の平均が 200 を超え結果が悪化した。全ての試行において [O] が獲得できず最大 F の分類子数の平均が 32 よりも小さい値を取り、獲得された分類子は過剰に一般化された分類子が多かった。ケース V においては $\sup\{\sigma_R\} = 200, 700$ において正答率の平均は 100% 近くであるが、分類子数の平均は $\sup\{\sigma_R\} = 700, 1500$ では増加した。[O] の獲得は $\sup\{\sigma_R\} = 200$ では全ての試行数でできており最大 F の分類子数の平均は 32 であるが、 $\sup\{\sigma_R\} = 700, 1500$ の場合ではほとんどの試行において獲得できていなかった。

ケース U とケース V の比較から XCS-RR の適用範囲は、正答率の平均の観点では平均的な不確実性に依存するが、一般化された分類子の獲得の観点から不確実性の強度の最大値 ($\sigma_R = 200$ 程度) に依存することが分かる。

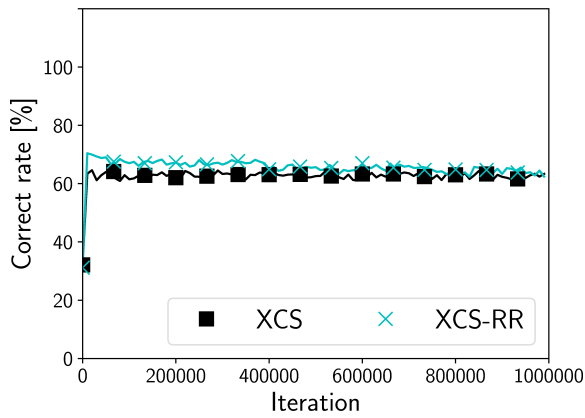


(a) 正答率の平均

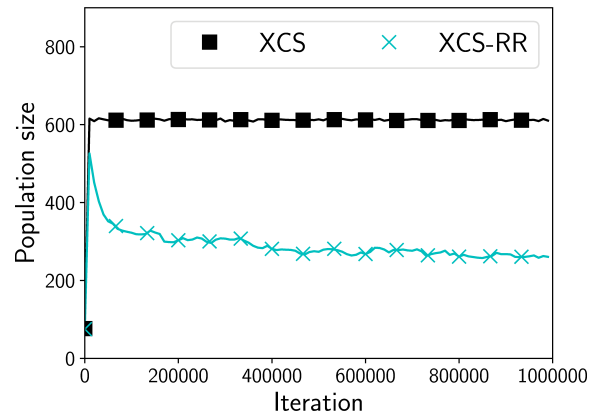


(b) 分類子数の平均

図 6.6 ケース U: $\sigma_R = 200$

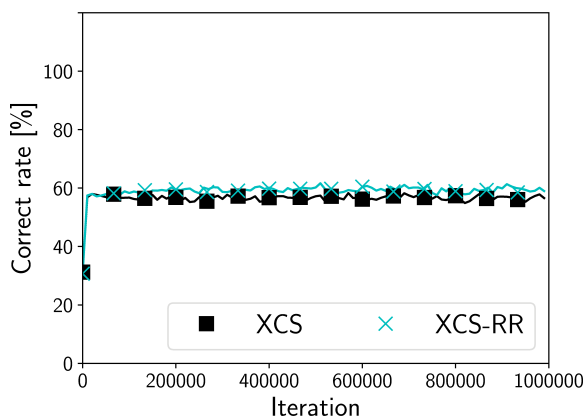


(a) 正答率の平均

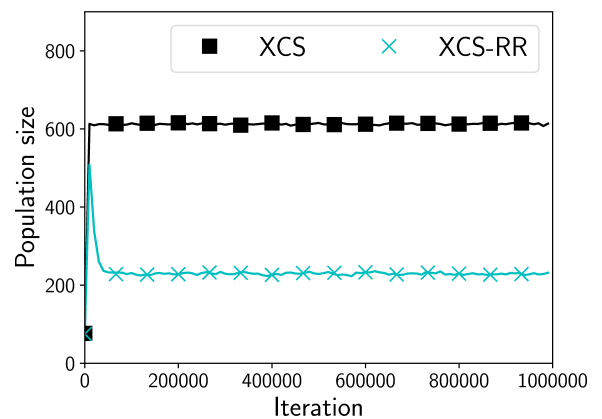


(b) 分類子数の平均

図 6.7 ケース U: $\sigma_R = 700$

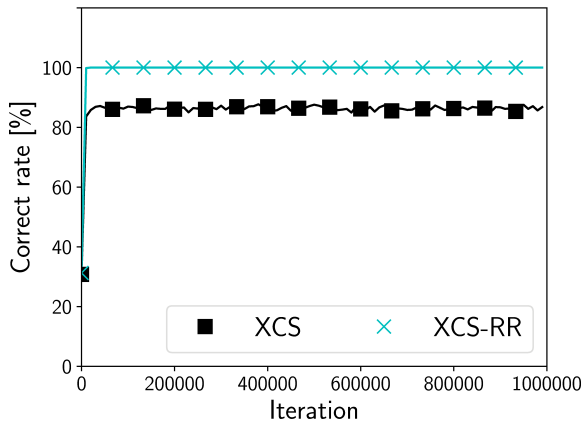


(a) 正答率の平均

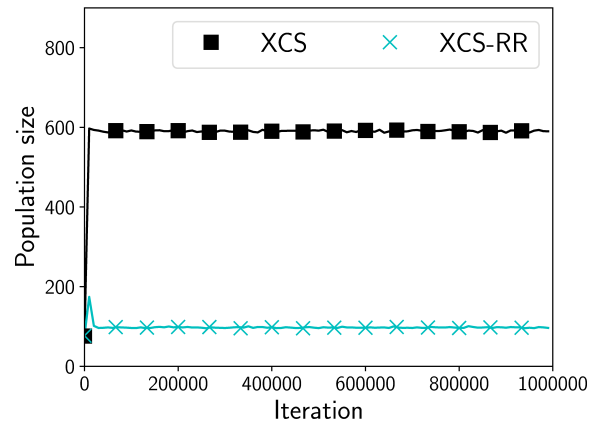


(b) 分類子数の平均

図 6.8 ケース U: $\sigma_R = 1500$

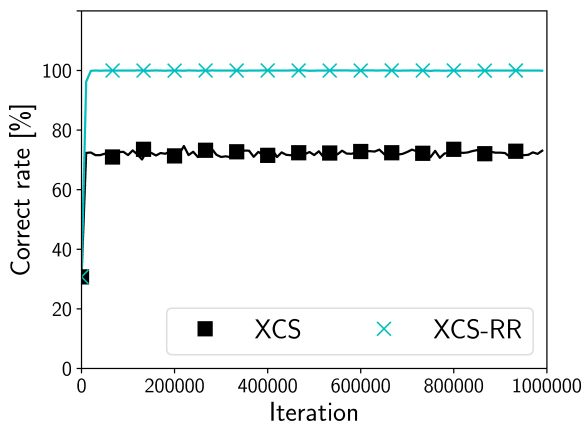


(a) 正答率の平均

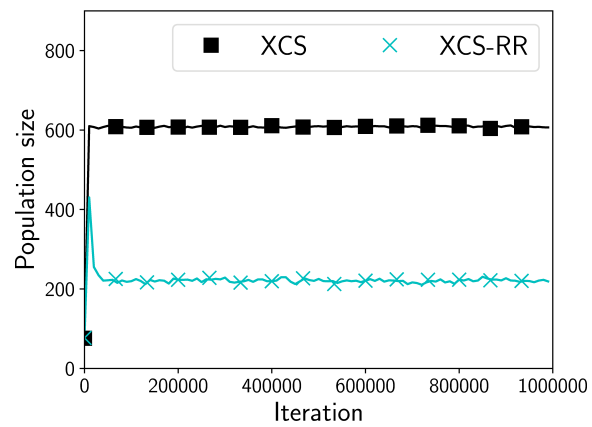


(b) 分類子数の平均

図 6.9 ケース V: $\sup\{\sigma_R\} = 200$

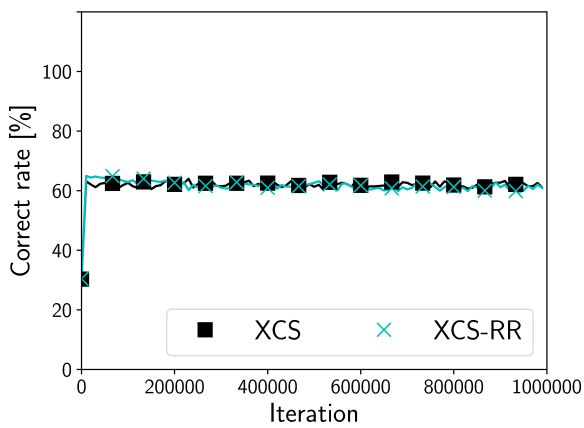


(a) 正答率の平均

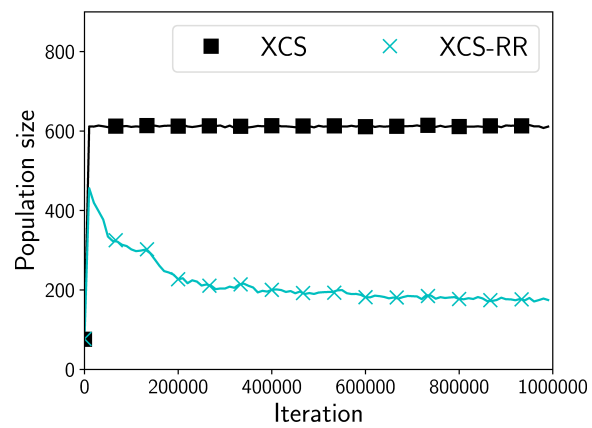


(b) 分類子数の平均

図 6.10 ケース V: $\sup\{\sigma_R\} = 700$



(a) 正答率の平均



(b) 分類子数の平均

図 6.11 ケース V: $\sup\{\sigma_R\} = 1500$

表 6.1 $\sigma_R = 0$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	47 (94%)	47 (94%)	32.02
XCS-RR	50 (100%)	50 (100%)	32

表 6.2 ケース U: $\sigma_R = 200$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	59.98
XCS-RR	50 (100%)	50 (100%)	32

表 6.3 ケース U: $\sigma_R = 700$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.22
XCS-RR	0(0%)	0 (0%)	4.60

表 6.4 ケース U: $\sigma_R = 1500$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	47.16
XCS-RR	0 (0%)	0 (0%)	2.30

6.6 考察

6.6.1 XCS-RR の正確な分類子の識別

$\sigma_R = 200$ の判定の信憑性における不確実性を伴うデータにおいて XCS-RR は全ての試行で [O] を獲得したが, $\sigma_R = 700, 1500$ の場合においては全ての試行で [O] を獲得できなかった. 式 (6.3) および (6.4) にある様に XCS-RR は分類子の正確性を分類子が獲得した報酬の最大値および最小値に基づいて決定する. 図 6.12(a) および図 6.12(b) に σ_R が 200, 700 の時の報酬の分布を示す. 横軸は報酬の値を, 縦軸はその報酬が獲得される確率を表す. 赤色が正答時の報酬の分布, 青色が誤答時の報酬の分布を表す. ガウス分布は $(-\infty, \infty)$ の範囲で値を取るが, 図 6.12(a) および図 6.12(b) より報酬の範囲は実質 $\sigma_R = 200$ の場合は ± 700 程度の範囲, $\sigma_R = 700$ の場合は ± 2000 程度の範囲となる. さらにガウス分布の平均は 0 なため, $cl.M$ および $\frac{cl.Max+cl.Min}{2}$ は 0 あるいは 1000 に近い値を取る. つまり, $\sigma_R = 200$ の場合に正解時に獲得する報酬の大半は 300 から 1700 の範囲であり不正解時に獲得する報酬の大半は -700 から 700 の範囲である. 一方で

表 6.5 ケース V: $\sup\{\sigma_R\} = 200$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	61.66
XCS-RR	50 (100%)	50 (100%)	32

表 6.6 ケース V: $\sup\{\sigma_R\} = 700$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	54.96
XCS-RR	14(28%)	0 (0%)	37.92

表 6.7 ケース V: $\sup\{\sigma_R\} = 1500$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.58
XCS-RR	0 (0%)	0 (0%)	6.34

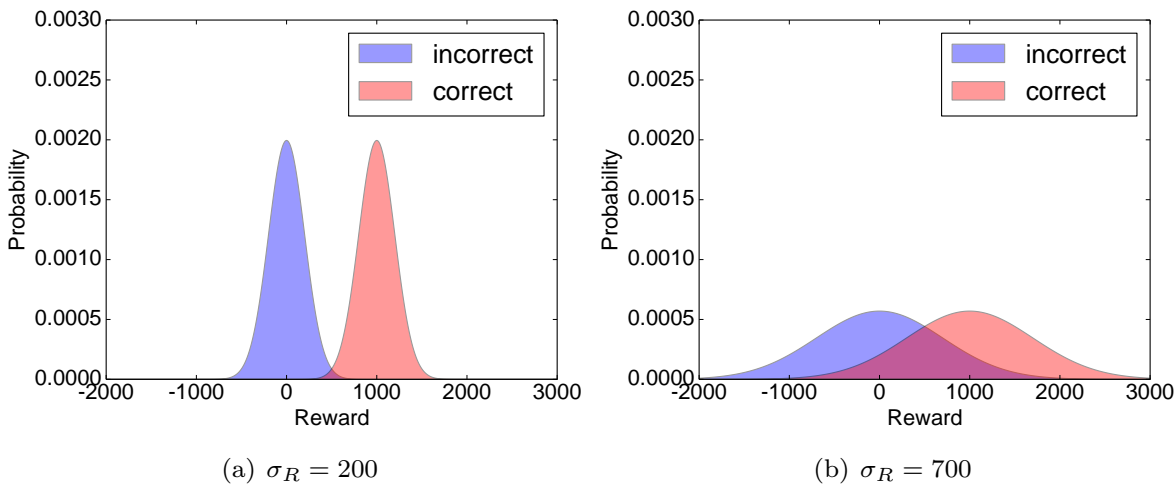


図 6.12 判定の信憑性における不確実性を伴う場合の報酬の分布

$\sigma_R = 700$ の場合に正解時に獲得する報酬の大半は-1000 から 3000 の範囲であり不正解時に獲得する報酬の大半は-200 から 2000 の範囲である。そのため、 $\sigma_R = 200$ の場合に正確な分類子は式 (6.3) あるいは (6.4) を満たし、 $\sigma_R = 700$ の場合に式 (6.3) および (6.4) を満たさない。さらに、正確性の判定が最大値あるいは最小値に基づいているため学習の途中で不確実性の影響を連続して受けても、最大値および最小値に与える影響は XCS の ϵ の計算や平均値および標本標準偏差に比べて小さいため、分類子の正確性を安定して評価可能である。

これらの結果から XCS-RR は判定の信憑性における不確実性を伴う場合には $\sigma_R = 200$

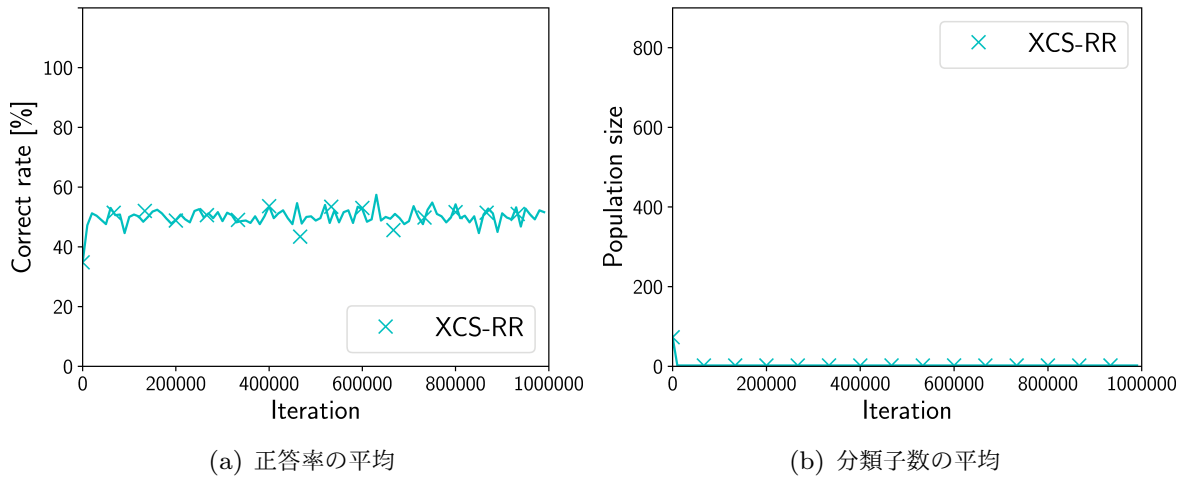


図 6.13 $\theta_{RR} = 0\%$

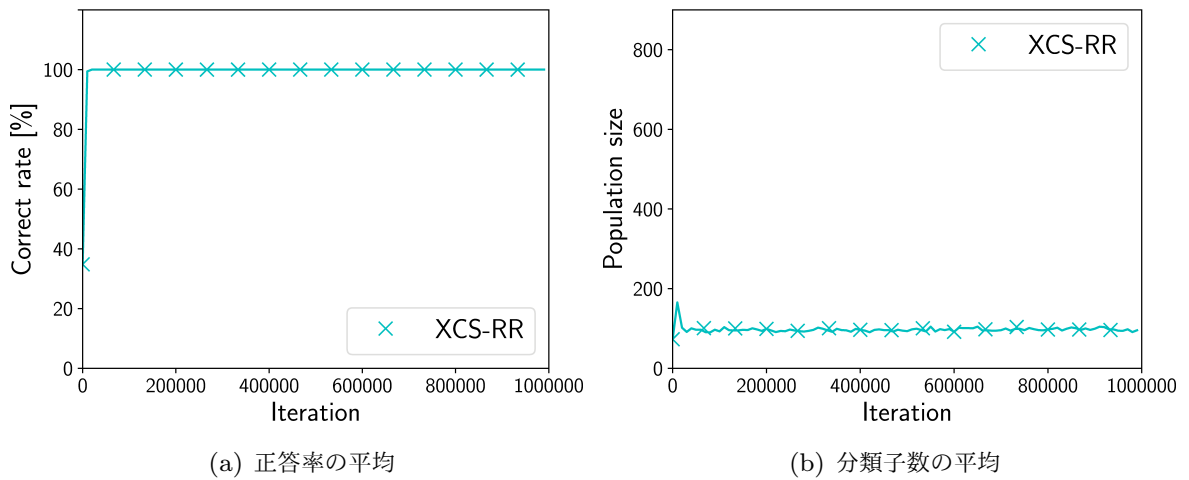


図 6.14 $\theta_{RR} = 10\%$

までであれば全ての試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し [O] を獲得する。

6.6.2 XCS-RR の新設されたパラメータ

XCS-RR はハイパーパラメータ θ_{RR} を持つ。この節では θ_{RR} の設定について述べる。図 6.13 から 6.16 に判定の信憑性における不確実性 ($\sigma_R = 200$) を伴う場合に θ_{RR} を 0%, 10%, 70%, 90% に設定した際の正答率の平均と分類子数の平均を示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。 $\theta_{RR} = 0\%$ の場合において正答率の平均が約 50% で分類子数の平均が 2 と分類子が過剰に一般化されている。 $\theta_{RR} = 0$ では他の分類子と式 (6.3) あるいは式 (6.4) の条件を満たさなくても正確

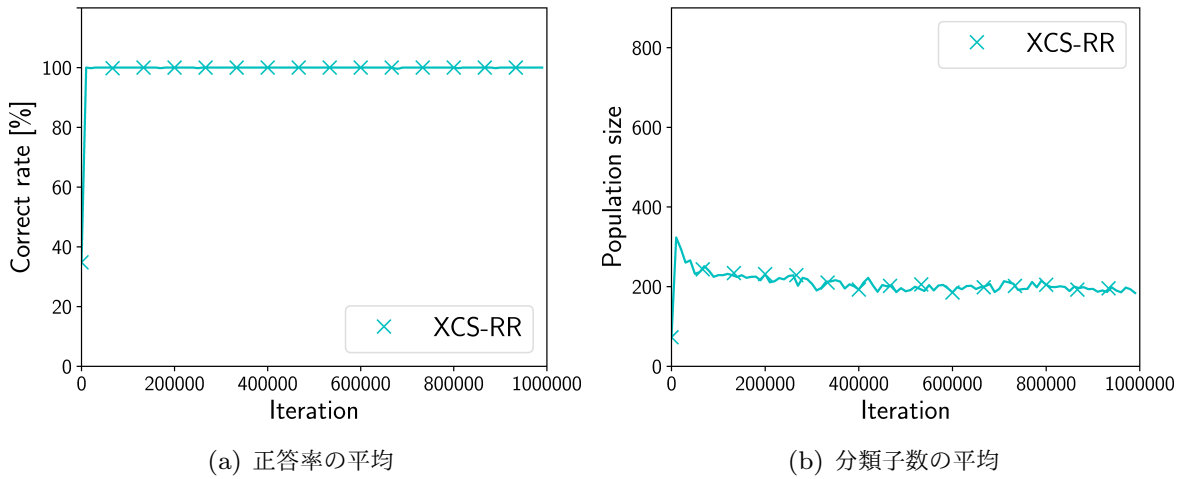


図 6.15 $\theta_{RR} = 70\%$

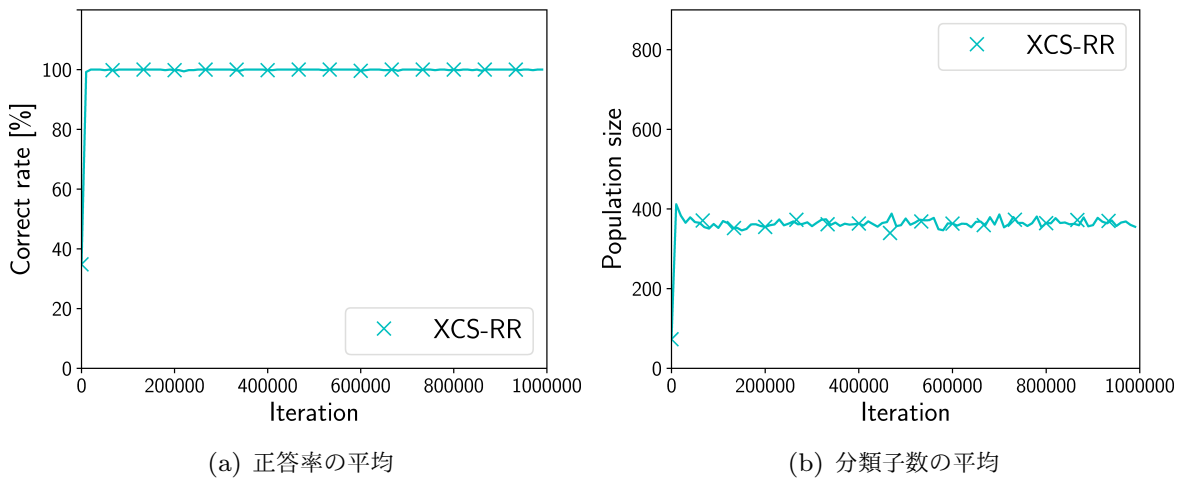


図 6.16 $\theta_{RR} = 90\%$

とみなされるため、不正確な分類子も正確と判定される。

$\theta_{RR} = 10\%$ の場合において正答率の平均が 100% に達し、分類子数の平均は 100 程度と適切に学習ができています。

$\theta_{RR} = 70\%, 90\%$ の場合においてどちらも正答率の平均は 100% 付近であるが、分類子数の平均が 100 よりも明らかに大きく分類子の一般化が適切に行われていない。 θ_{RR} が大きいと式 (6.3) あるいは式 (6.4) の条件を $[P']$ の分類子の多くと満たす必要がある。 $[P']$ には正解を導く正確な分類子、不正解を導く正確な分類子および不正確な分類子が混在しているため、 θ_{RR} が大きいほど厳しくなる。また、 θ_{RR} が大きいと過剰な一般化が行われず正答率の平均はほぼ 100% を維持している。これらの結果から θ_{RE} を 10% から 70% よりも小さな値に設定するのが好ましい。

第 7 章

入出力および判定の信憑性における不確実性に対応可能な学習分類子システム

実問題では入出力および判定の信憑性における不確実性のいずれに対しても適応可能な、報酬の組分けに基づく学習分類子システム (XCS based on Reward Group: XCS-RG) および正誤推測に基づく学習分類子システム (XCS based on Estimated Correctness: XCS-EC) を提案する。XCS-RG は XCS-RR の報酬の大小による正誤判定および XCS-ER の分類子を構成する入出力の組の報酬分布の考慮を基に、報酬の大小によって正誤を推測した入出力の組み合わせで分類子の正確性を判定する。また、XCS-EC は XCS-RR の報酬平均を用いた正誤の推測および XCS-ER の獲得報酬の推測を基に、報酬の平均による入出力の組み合わせの正誤推測によって分類子の正確性を判定する。

7.1 入出力における不確実性および判定の信憑性における不確実性に対するアプローチ

図 7.1 は入出力における不確実性および判定の信憑性における不確実性を伴った場合の入出力に対する判定とそれに基づく XCS の正確性判定および、目標とする不確実性に対処可能な XCS の正確性判定を示す。上部では入出力における不確実性を伴う場合のデータを例示しており、データ 2 において入出力における不確実性により認識された入出力に対する正誤評価が異なる。さらに判定の信憑性における不確実性によりそれぞれの正誤評価の値がばらつきを持った報酬として XCS に与えられる。このデータにおいて XCS は正確な分類子 $\{C : 00\#, A : 0\}$ が認識された入力が “000” の場合に照合し、獲得報酬がばらつくため正解 (報酬 R) と不正解 (報酬 0) が混在すると認識され不正確と評価される。

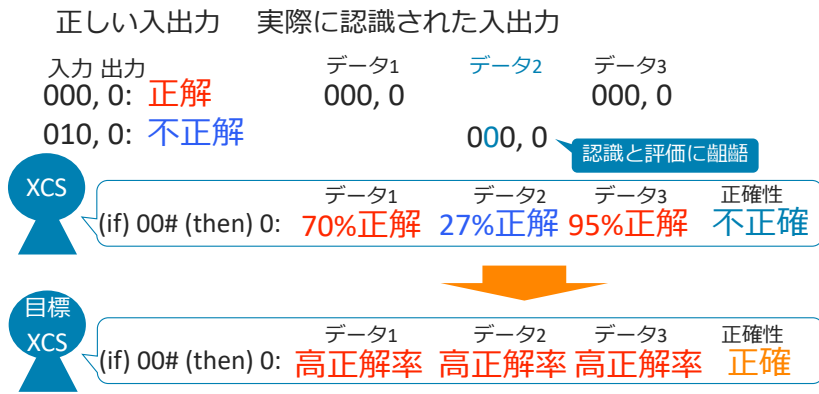


図 7.1 入出力における不確実性および判定の信憑性における不確実性を伴う場合

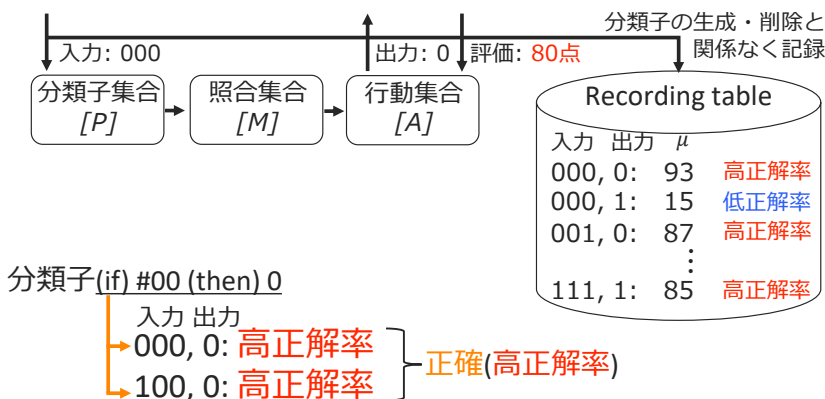


図 7.2 XCS-RG の概要

これまでに提案した XCS-ER は入出力における不確実性を伴う場合に対して報酬の“正しさ”を推測することで対応を図り、XCS-RR は判定の信憑性における不確実性を伴う場合に対して分類子が獲得した報酬全体の分布（最大値および最小値）によって正確性を判定する事で対応を図った。両手法のアプローチを組み合わせることにより入出力における不確実性および判定の信憑性における不確実性の両方に対応を図る。つまり、1) 獲得した報酬を分布上での位置を基に“正しさ”を推測し正誤評価に変換するアプローチと、2) 入力と出力の組における報酬の“正しさ”を推測した後にその分布により分類子の正確性を判定するアプローチである。これらのアプローチにより、認識された入出力における正しい正誤評価を推測し、それに基づいて分類子の正確性を判断することができる。

図 7.2 に XCS-RG の概要を示す。入力と出力が定まった時点で分類子の更新や生成および削除など関係なく Recording table にその入出力対における獲得報酬の平均を記録し、その値が高い値なのか低い値なのかを判断する。その判断を用いて分類子の正確性を判定する。

図 7.3 に XCS-EC の概要を示す。入力が与えられ [M] が作られた段階で、[M] を行動

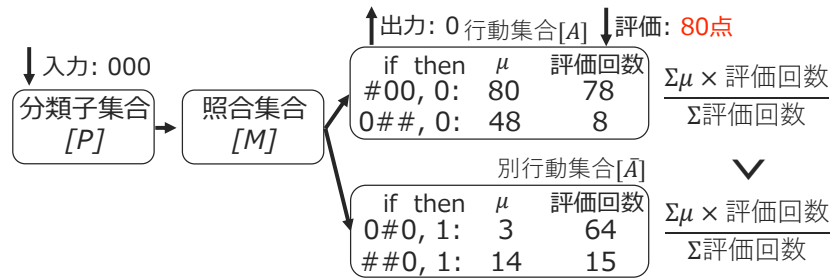


図 7.3 XCS-EC の概要

毎に分けそれぞれにおいて期待値として評価回数を重みとした獲得報酬の平均の加重平均を求め、どの行動が入力に対して正解か不正解かを推測する。その推測された評価に基づいて分類子の正確性を判定する。以下に、XCS-RG および XCS-EC の詳細を記す。

7.2 獲得報酬記録に基づく学習分類子システム

7.2.1 構成

アーキテクチャ

XCS-RG の XCS との差分は強化部の Recording table とパラメータ更新および発見部の包摂である。図 7.4 の黒塗りの部分はその差分を表す。また、新設されたパラメータおよび異なる方法で更新されるパラメータを明記している。XCS は獲得した報酬の情報は分類子が更新対象となった場合にその分類子に記録されるのみなので、学習データ全体における情報は持たない。そのため、XCS-RG は Recording table を分類子と別に持つことで学習データ全体における情報を記録する。

分類子

XCS-RG の分類子は XCS の分類子が持つ $C, A, p, \epsilon, F, exp, ts, as, n$ に加えて、正確性の基準となる許容誤差 ϵ_0 および獲得した報酬の平均値、標本標準偏差 M, S を記録するパラメータを持つ。

Recording table

XCS-RG は分類子が一般化されていない状態つまり状態行動対は正確であると仮定し、入力に対する出力ごとの獲得する報酬を把握するために、分類子とは別に、入力と出力の組み合わせである状態行動対ごとに獲得した報酬の平均と分散を記録する。

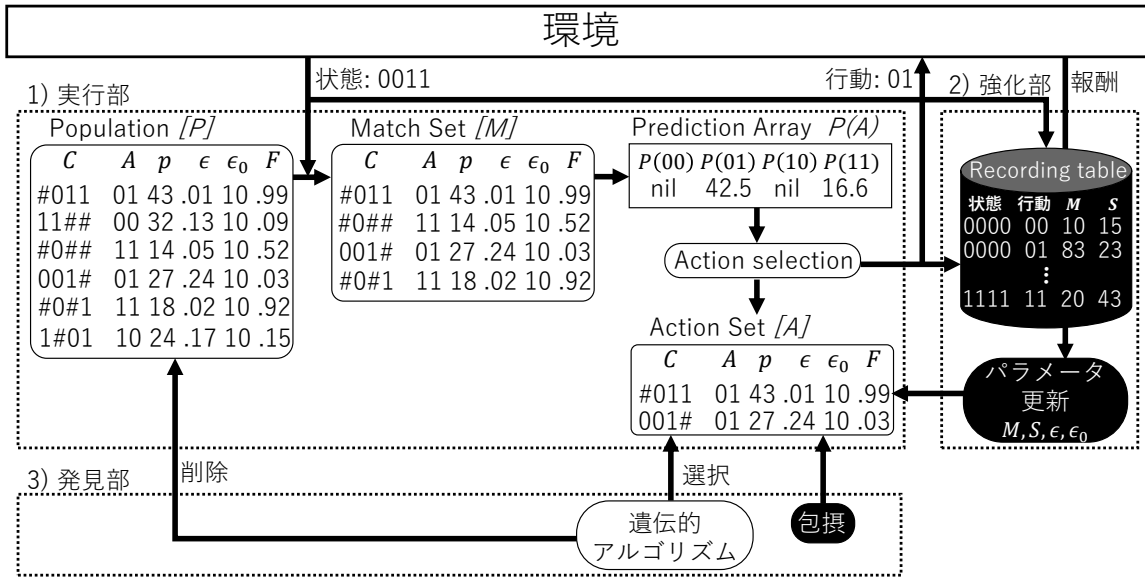


図 7.4 XCS-RG のアーキテクチャ

7.2.2 学習メカニズム

Recording table に全ての獲得した報酬の情報を記録し、それを用いて分類子の正確性を判定する。

Recording table の更新

XCS-RG は報酬が与えられると Recording table 内の入力された状態 s と出力された行動 a の組として当てはまる箇所 (s, a) の平均 $M(s, a)$ と不偏分散 $V(s, a)$ を更新する。 $M(s, a)$ を

$$M(s, a) \leftarrow (reward - M(s, a)) / cl.exp \quad (7.1)$$

と $V(s, a)$ を

$$V(s, a) \leftarrow ((s, a).exp - 1) \times (\rho - M(s, a))^2 / (s, a).exp \quad (7.2)$$

と逐次的に求める。ただし、 $(s, a).exp$ は状態行動対 (s, a) の評価回数である。入力された状態 s と出力された行動 a の組ごとに平均 $M(s, a)$ と不偏分散 $V(s, a)$ を記録することで分類子の生成や淘汰による記録の断絶が起こらず、正しく報酬の分布を把握可能である。Recording table の更新のアルゴリズムを Algorithm 12 に示す。4 行目から 5 行目において現在の入力 s と出力 a に照合する状態行動対 (s, a) が獲得した報酬の平均と分散を記録する。

Algorithm 12 Recording table の更新

```
1: receive  $\rho$ : 報酬を獲得
2: for all state-action pair  $(s, a)$  in Recording table do
3:   if  $(s, a)$  is current input and output then
4:      $M(s, a) \leftarrow (reward - M(s, a))/cl.exp$ 
5:      $V(s, a) \leftarrow ((s, a).exp - 1) \times (\rho - M(s, a))^2 / (s, a).exp$ 
6:   end if
7: end for
```

正確性判定

XCS-RG は Recording table に記録された値を用いて分類子の正確性を判定する。2 クラス分類問題で2種類の出力は一方が正解でありもう一方が不正解であることに加え、正解時に獲得する報酬および不正解時に獲得する報酬（の平均）それぞれは入力あるいは出力が変わったとしてもほぼ同一とみなせると仮定する。その仮定から、正解時の獲得報酬の平均と不正解時の獲得報酬の平均の差分を求め、それに基づいて分類子の正確性を判定する。まず基準距離 d_r を見積もる。基準距離は任意の状態において行動が異なった際に獲得報酬の平均値が最低どれだけ異なるかを示す指標である。 d_r は全ての状態において、それぞれの行動間での差分を算出しその最小値を求める。状態ごとに求めた最小値の中から最小のものを d_r とする。正解時と不正解時に獲得する報酬の平均は少なくとも d_r は異なることが分かる。式で表すと以下の様になる。ただし \inf は下限 (infimum) を表す。

$$d_r = \inf \{ \inf \{ |M(s, a) - M(s, a')|, \{a, a'\} \in action \}, s \in state \} \quad (7.3)$$

次に分類子が獲得する報酬の範囲 r_{cl} を求める。報酬の範囲 r_{cl} は分類子 cl に照合する状態行動対の Recording table に記録された獲得報酬の平均値が分布する範囲を示す指標である。分類子 cl に照合する状態行動対の Recording table に記録された獲得報酬の平均値の最大値と最小値の差分が r_{cl} となる。分類子 cl が獲得する報酬の平均は r_{cl} だけばらつきを持つことが分かる。式で表すと以下の様になる。ただし \sup は上限 (supremum) を表す。

$$r_{cl} = \sup \{ M(s, a), (s, a) \text{ matched } cl \} - \inf \{ M(s, a), (s, a) \text{ matched } cl \} \quad (7.4)$$

図 7.5 に d_r および r_{cl} の算出例を示す。左側は d_r の算出を表し、状態 0000 に対する行動 01 と 10 における獲得報酬の平均の差 318 が一番小さいため、 d_r は 318 となる。右側は分類子 $\{C : 0\#\#0, A : 01\}$ における r_{cl} の算出を表す。入力 0010 および 0000 における獲得報酬の平均が分類子 $\{C : 0\#\#0, A : 01\}$ に照合する状態行動対が獲得する報酬の平均の最大値と最小値に当たるため、 r_{cl} は 40 となる。 r_{cl} が d_r よりも小さければ、分

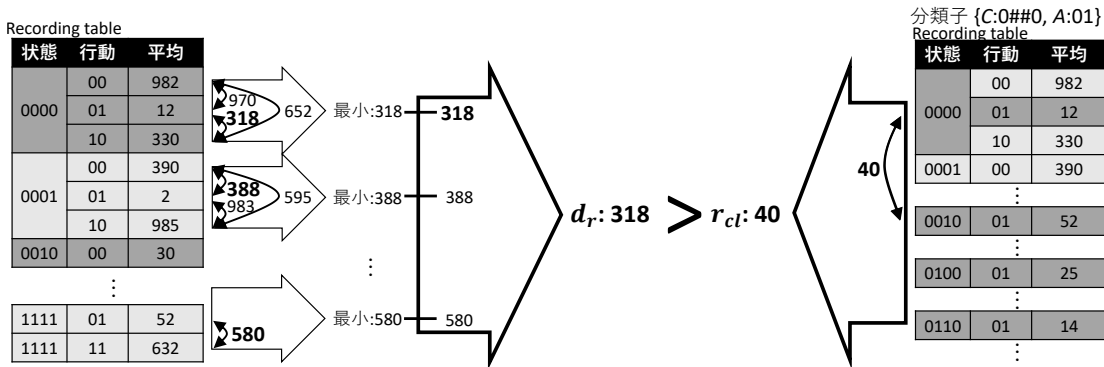


図 7.5 報酬間の距離 d_r と分類子の獲得報酬範囲 r_{cl}

類子 cl は獲得報酬のばらつきが小さいため、正誤の混在（不確実性）が無いと判断する。Recording table に記録された値を用いた正確性判定のアルゴリズムを Algorithm 13 に示す。1 行目から 8 行目において全ての状態行動対を対象として $M(s, a) - M(s, a')$ を求め、9 行目において d_r に代入している。11 行目から 13 行目において分類子 cl に照合する状態行動対を対象として $M(s, a)$ の最大値および最小値を求め、14 行目において r_{cl} を求める。15 行目の条件を満たす分類子を正確とみなし、条件を満たさなければ不正確とみなす。

パラメータ更新

XCS-RG は報酬を獲得すると $[A]$ を構成するすべての分類子においてそれぞれの分類子がこれまでに獲得した報酬から求まる平均 M 、標本標準偏差 S を更新する。 $cl.M$ を

$$cl.M \leftarrow (reward - cl.M)/cl.exp \quad (7.5)$$

と $cl.S$ を

$$cl.S \leftarrow \sqrt{(cl.exp - 1) \times (\rho - cl.M)^2 / cl.exp} \quad (7.6)$$

と逐次的に求める。XCS-RG は分散によって報酬のばらつきを把握し、その値に基づいて正確性を判定する。この $cl.S$ の値を $cl.\epsilon$ とする。また ϵ_0 は下記の包摂条件を満たした分類子の中で最大の S を $MaxS$ とし、 $[A]$ を構成する分類子の ϵ_0 を

$$cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(MaxS + E_0 - cl.\epsilon_0) \quad (7.7)$$

と更新し、条件を満たした分類子全てを正確と扱う。また、 ϵ_0 を $MaxS$ から定数 E_0 だけ大きくすることにより、標本標準偏差のばらつきを考慮した設定にする。パラメータの更新のアルゴリズムを Algorithm 14 に示す。3 行目から 5 行目において Algorithm 13 の正確性の条件を満たした分類子の中から最大の ϵ を求める。6 行目から 12 行目において $[A]$ の分類子のパラメータを更新する。

Algorithm 13 Recording table に記録された値を用いた正確性判定

```
1: for all state  $s$  in all state  $\mathbf{S}$  do
2:   for all action  $a$  in all action  $\mathbf{A}$  do
3:     for all action  $a'$  in all action  $\mathbf{A}$  without  $a$  do
4:       calc  $M(s, a) - M(s, a')$ : 行動が異なるときの獲得報酬の平均の差
5:       calculate  $Min$ : 最小の  $M(s, a) - M(s, a')$  を求める
6:     end for
7:   end for
8: end for
9:  $d_r = Min$ 
10: for all classifier  $cl$  in  $[A]$  do
11:   for all state-action pair  $(s, a)$  in matched  $cl$  do
12:     calculate  $Max, Min$ :  $M(s, a)$  の最大値および最小値を求める.
13:   end for
14:    $r_{cl} = Max - Min$ 
15:   if  $d_r > r_{cl}$  then
16:      $cl$  is accurate
17:   else
18:      $cl$  is inaccurate
19:   end if
20: end for
```

Algorithm 14 パラメータの更新

```
1: recieve  $\rho$ : 報酬を獲得
2: initialize  $MaxS$ 
3: for all classifier  $cl$  in  $[A]$  do
4:   calculate  $MaxS$ : 条件を満たした分類子の中で最大の  $\epsilon$  を求める
5: end for
6: for all classifier  $cl$  in  $[A]$  do
7:    $cl.M \leftarrow (reward - cl.M)/cl.exp$ 
8:    $cl.S \leftarrow \sqrt{(cl.exp - 1) \times (\rho - cl.M)^2 / cl.exp}$ 
9:    $cl.\epsilon = cl.S$ 
10:   $cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(MaxS + E_0 - cl.\epsilon_0)$ 
11:  update  $p, as, exp, \kappa(cl), \kappa'(cl), F$ : パラメータ更新 (XCS と同様)
12: end for
```

包摂条件

Recording table は分類子の生成と削除と関係なく更新されるので、分類子よりも多くの獲得報酬の情報を有しており、分類子の情報のみを用いるよりも正しく報酬分布を把握し分類子の正確性を判定できる。Recording table の値を用いて判定する正確な分類子が満たす条件 $d_r > r_{cl}$ を加えて XCS-RG の包摂条件は以下の通りになる。

- $cl.exp > \theta_{sub}$
- $\kappa = 1$
- $d_r > r_{cl}$

7.2.3 アルゴリズム

XCS-RG のアルゴリズムを Algorithm 15 に示す。下線が引かれている部分が XCS からの変更点である。XCS-RG は環境から入力された状態に照合する分類子を $[P]$ の中から抜き出し、 $[M]$ を生成する (2, 3 行目)。P(action) を基に決定した行動を有する分類子を $[M]$ から抜き出し、 $[A]$ を生成する (4 から 6 行目)。選択した行動を環境に対して出力し得られた報酬を用いて Variance table における入力と出力が一致した箇所の平均及び標本標準偏差の値を更新 (8 行目) した後、正確とみなすべき分類子を調べ (9 行目)、 $[A]$ の分類子のパラメータを更新し学習を行う (10 行目)。その後、包摂条件や GA の条件を満たした場合に $[A]$ の分類子を対象として、包摂 (17 行目) や GA (20 行目) を行う。包摂条件を判定する際には、Variance table を用いてその分類子がただ一つの基準報酬から報酬を獲得しているのかを確認する (12 行目)。上記の流れを 1 学習エピソードとして、学習及び GA による生成と淘汰を繰り返すことによって、環境に適した分類子集団を獲得する。

7.3 正誤推測に基づく学習分類子システム

7.3.1 構成

アーキテクチャ

XCS-EC の XCS との差分は強化部の正誤推測機構およびパラメータ更新と発見部の包摂である。図 7.6 の黒塗りの部分はその差分を表す。また、新設されたパラメータおよび異なる方法で更新されるパラメータを明記している。XCS-EC は正誤推測により報酬に伴う不確実性を低減し、分類子の正しい正確性の評価を図る。報酬を獲得すると直ちに正誤推測に用いるパラメータを更新し、正誤推測を行う。推測結果を基に更新対象の分類子

Algorithm 15 XCS-RG のアルゴリズム

```
1: while (! end of iterations) do
2:    $state \leftarrow$  environment: 環境から状態  $state$  を受け取る
3:   generate  $[M]$ :  $[P]$  から  $state$  に照合する分類子を選択
4:    $P(a_i) = \sum_{cl_k \in [M]|a_i} cl_k.p \times cl_k.F / \sum_{cl_l \in [M]|a_i} cl_l.F$ :  $[M]$  から各行動の予測値を算出
5:    $action \leftarrow P(action)$ :  $P(action)$  から行動を選択
6:   generate  $[A]$ :  $[M]$  から  $action$  に照合する分類子を選択
7:    $reward \leftarrow$  execute  $action$ : 行動を実行し, 環境より報酬を獲得
8:   update Recording table: Recording table の更新 (Algorithm 12)
9:   accuracy judgment: 正確性判定 (Algorithm 13)
10:  update parameters: パラメータ更新 (Algorithm 14)
11:  for all classifier  $cl$  in  $[A]$  do
12:    if  $cl.exp > \theta_{sub}$  and  $\kappa = 1$  and  $d_r > r_{cl}$  then
13:       $subsumptionFlag = true$ 
14:    end if
15:  end for
16:  if  $subsumptionFlag = true$  then
17:    run subsumption:  $[A]$  内の分類子を対象に包摂を実行
18:  end if
19:  if run GA then
20:    generate new classifiers: 子個体の生成
21:    delete classifier: 削除
22:  end if
23: end while
```

の正確性に関わるパラメータを更新する.

分類子

XCS-EC の分類子は XCS-ER の分類子と同じく $C, A, p, \epsilon, F, exp, ts, as, n$ を持つが、報酬の獲得回数 $T_{\rho=reward}$ を持たず、推測報酬の獲得回数 $E_{\rho=reward}$ の代わりに正誤推測機構によってその分類子が照合する入出力対における推測正誤の回数 $E_{c=correctness}$ を持つ。ただし 2 クラス分類で正解と不正解に分かれるため、 $E_{c=correctness}$ には $E_{c=correct}$ および $E_{c=incorrect}$ がある。

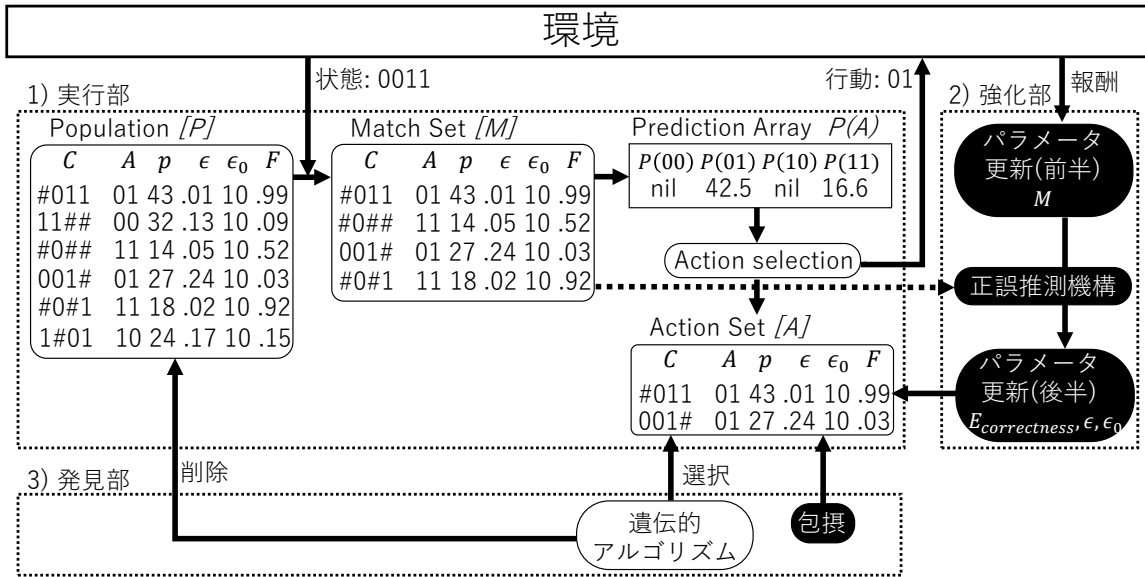


図 7.6 XCS-EC のアーキテクチャ

7.3.2 学習メカニズム

パラメータ更新は p, exp, ts, as, n, M を更新する前半と $E_{c=correctness}, \epsilon, \epsilon_0, \kappa, \kappa', F$ を更新する後半に分かれ、その間に報酬推測機構が動作する。パラメータ更新を分け、正誤推測に用いるパラメータを報酬の獲得直後に更新することで、正誤推測に獲得したばかりの報酬の情報を含めて用いることができるため、より安定した推測が行われることが見込まれる。

パラメータ更新 (前半)

XCS-EC の分類子は $cl.T_{\rho=\rho}$ を持たないため式 (5.1) の処理は行われない。 p, exp, ts, as, n の他に獲得した報酬の平均値 M を逐次的に求める。

$$cl.M \leftarrow (reward - cl.M) / cl.exp \quad (7.8)$$

パラメータ更新 (前半) のアルゴリズムを Algorithm 16 に示す。3 行目では XCS と同様に p, exp, ts, as, n を更新する。4 行目で獲得報酬の平均を更新する。

正誤推測機構

報酬の同一性ではなく (推測) 正誤の同一性によって分類子の正確性を判定するため、 $[M]$ を構成する分類子を分類子が持つ行動ごとに分けた集合 (collective と呼ぶ) の評価回数を重みとした獲得報酬の平均の加重平均 $CR_{A=a}$ を比較する。ここで、XCS-EC は入力に対する正しい出力の正誤を認識できないことおよび学習に用いられる $[A]$ とは別

Algorithm 16 パラメータ更新 (前半)

- 1: receive *reward*: 環境より報酬を獲得
 - 2: **for all** classifier *cl* in [*A*] **do**
 - 3: update *p, exp, ts, as, n*: パラメータ更新 (XCS と同様)
 - 4: $cl.M \leftarrow (reward - cl.M)/cl.exp$
 - 5: **end for**
-

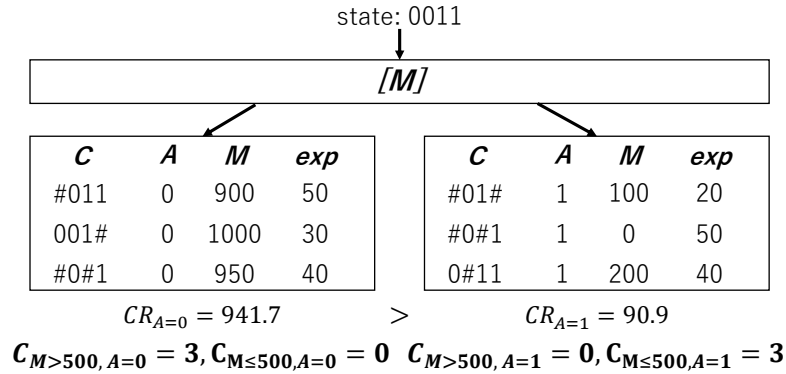


図 7.7 正誤推測

に生成されることから, [*M*] を構成する分類子を分類子を持つ行動ごとに分けた集合を collective と呼ぶ.

$$CR_{A=a} = \frac{\sum_{cl \in [M] | A=a} cl.M \times cl.exp}{\sum_{C \in [M] | A=a} C.exp} \quad (7.9)$$

ここで, $cl \in [M] | A = a$ は [*M*] を構成する分類子の中で行動が *a* である分類子 *cl* を表す. それに加えそれぞれの collective ごとにその時点までに XCS-EC が獲得した報酬の平均値 μ よりも獲得した報酬の平均 *M* が大きいあるいは小さい分類子の数 $C_{cl.M > \mu, A=a}$, $C_{cl.M \leq \mu, A=a}$ を数える. 正解時に *R*, 不正解時に 0 の報酬が与えられる XCS で一般的な設定の 2 クラス分類問題において $C_{cl.M > \mu, A=a_0}$, $C_{cl.M > \mu, A=a_1}$, $C_{cl.M \leq \mu, A=a_0}$ および $C_{cl.M \leq \mu, A=a_1}$ の 4 つを数える. ただし, *cl.M* は分類子 *cl* が獲得した報酬の平均値を a_0, a_1 は 2 つのクラスを表す. 図 7.7 は正解時の報酬が 1000 であり不正解時の報酬が 0 で μ は 500 となる場合の例である. 行動が 0 の collective において *cl.M* が μ を超えている分類子が 3 個と *cl.M* が μ 以下の分類子が 0 個存在し, 行動が 1 の collective において *cl.M* が μ を超えている分類子が 0 個と *cl.M* が μ 以下の分類子が 3 個存在するため, $C_{cl.M > \mu, A=1} = 0, C_{cl.M \leq \mu, A=1} = 3$ となる. これらの値を用いて正誤を推測する. 獲得報酬の平均の加重平均 $CR_{A=a}$ の大小関係と獲得した報酬の平均値 μ よりも *M* が大きいあるいは小さい分類子の数の大小関係にの両方から推測することにより, 評価回数がかかなり大きい分類子および大量の評価回数の少ない分類子の影響を減らしつつ安定的に正誤を

推測する.

$$\begin{cases} CR_{A=a_0} > CR_{A=a_1} \\ C_{cl.M>\mu,A=a_0} > C_{cl.M\leq\mu,A=a_0} \\ C_{cl.M>\mu,A=a_1} < C_{cl.M\leq\mu,A=a_1} \end{cases} \quad (7.10)$$

の条件を満たし、行動 a_0 が選択された時 $E_{c=correct}$ を、行動 a_1 が選択された時 $E_{c=incorrect}$ を 1 だけ増加させる。上記の条件を満たさず

$$\begin{cases} CR_{A=a_0} \leq CR_{A=a_1} \\ C_{cl.M>\mu,A=a_0} < C_{cl.M\leq\mu,A=a_0} \\ C_{cl.M>\mu,A=a_1} > C_{cl.M\leq\mu,A=a_1} \end{cases} \quad (7.11)$$

の条件を満たし、行動 a_0 が選択された時 $E_{c=incorrect}$ を、行動 a_1 が選択された時 $E_{c=correct}$ を 1 だけ増加させる。正誤推測のアルゴリズムを Algorithm 17 に示す。3 行目において獲得報酬の平均の加重平均 $CR_{A=a}$ を求め、6 行目および 8 行目において $cl.M$ が μ が大きくなるあるいは小さくなる分類子の個数を数える。12 行目から 20 行目において求めた値を用いて正誤推測を行う。

パラメータ更新（後半）

XCS-ER と同様の処理を $E_{\rho=reward}$ の代わりに $E_{c=correctness}$ を用いて行う。また、判定の信憑性における不確実性により獲得される報酬が様々な値を取ることが考えられるため、 ϵ は XCS-ER と異なる方法で求める。分類子の生成や淘汰にかかわらず XCS-EC が獲得した報酬全体の中で最大値 (Max) および最小値 (Min) を記録する。分類子が獲得した報酬の平均値 $cl.M$ と Max および Min の差分を計算し、その小さい方を採用する。報酬の平均値 $cl.M$ と Max が近いときには cl が正解の評価を（多く）獲得する分類子とみなし、 $cl.M$ と Min が近いときには cl が不正解の評価を（多く）獲得する分類子とみなす。正確で正解を導く分類子の獲得報酬の平均 $cl.M$ は大きく、正確で不正解を導く分類子の獲得報酬の平均 $cl.M$ が小さく、不正確な分類子の獲得報酬の平均 $cl.M$ はその間の値を取ると想定する。

$$cl.\epsilon = \begin{cases} Max - cl.M & \text{if } Max - cl.M < cl.M - Min \\ cl.M - Min & \text{otherwise} \end{cases} \quad (7.12)$$

と求められる。また、 ϵ_0 は XCS-ER と同様に正確とみなされる分類子の ϵ の最大値を設定するが、 $\#$ が少なく一般化度合が低い分類子においても評価回数を担保するために exp が θ_{GA} 以上の条件を加え、

$$\begin{cases} cl.exp > 2^{\text{numberof}\#} \times \theta_{EC} \\ cl.exp > \theta_{GA} \\ \exists cl.E_{c=correctness}, cl.exp = cl.E_{c=correctness} \end{cases} \quad (7.13)$$

とする。 θ_{GA} は一般化度合が低い分類子に対して評価回数を確保するためのハイパーパラメータである。パラメータ更新（後半）のアルゴリズムを Algorithm 18 に示す。3 行

Algorithm 17 正誤推測

```
1: divide  $[M]$ : 行動ごとに  $[M]$  の分類子を collective に分割
2: for all collective  $[C]$  do
3:    $CR_{A=a} = (\sum_{cl \in [M]|A=a} cl.M \times cl.exp) / \sum_{C \in [M]|A=a} C.exp$ 
4:   for all action  $a$  in all action  $A$  do
5:     if  $cl.M > \mu$  then
6:        $C_{cl.M > \mu, A=a} \leftarrow C_{cl.M > \mu, A=a} + 1$ 
7:     else
8:        $C_{cl.M \leq \mu, A=a} \leftarrow C_{cl.M \leq \mu, A=a} + 1$ 
9:     end if
10:  end for
11: end for
12: if  $CR_{A=a_0} > CR_{A=a_1}$  and  $C_{cl.M > \mu, A=a_0} > C_{cl.M \leq \mu, A=a_0}$  and  $C_{cl.M > \mu, A=a_1} <$   

    $C_{cl.M \leq \mu, A=a_1}$  then
13:    $a_0$  is judged as correct ( $a_1$  is judged as incorrect)
14: else
15:   if  $CR_{A=a_0} \leq CR_{A=a_1}$  and  $C_{cl.M > \mu, A=a_0} < C_{cl.M \leq \mu, A=a_0}$  and  $C_{cl.M > \mu, A=a_1} >$   

      $C_{cl.M \leq \mu, A=a_1}$  then
16:      $a_1$  is judged as correct ( $a_0$  is judged as incorrect)
17:   else
18:     correct action is none: 正解の出力なし
19:   end if
20: end if
```

目から 8 行目において推測された正誤の個数を数える。9 行目から 13 行目において ϵ を求める。15 行目において正確とみなす分類子を識別し、20 行目においてその分類子を正確とみなせる様に ϵ_0 を更新する。

7.3.3 アルゴリズム

XCS-EC のアルゴリズムを Algorithm 19 に示す。下線が引かれている部分が XCS からの変更点である。XCS-EC は環境から入力された状態に照合する分類子を $[P]$ の中から抜き出し、 $[M]$ を生成する (2, 3 行目)。P(action) を基に決定した行動を有する分類子を $[M]$ から抜き出し、 $[A]$ を生成する (4 から 6 行目)。選択した行動を環境に対して出力し得られた報酬を用いて $[A]$ の分類子のパラメータを更新し学習を行う。8, 9, 10 行目はそれぞれ Algorithm 16 のパラメータ更新 (前半), Algorithm 17 の正誤推測機構,

Algorithm 18 パラメータ更新 (後半)

```
1: receive  $\rho'$ : 推測正誤を変換した報酬を獲得
2: for all classifier  $cl$  in  $[A]$  do
3:   if  $a$  is judged as correct then
4:      $E_{c=correct} \leftarrow E_{c=correct} + 1$ 
5:   end if
6:   if  $a$  is judged as incorrect then
7:      $E_{c=incorrect} \leftarrow E_{c=incorrect} + 1$ 
8:   end if
9:   if  $Max - cl.M > cl.M - Min$  then
10:     $cl.\epsilon = cl.M - Min$ 
11:   else
12:     $cl.\epsilon = Max - cl.M$ 
13:   end if
14:   initialize  $Max\epsilon$ 
15:   if  $cl.exp > 2^{numberof\#} \times \theta_{EC}$  and  $cl.exp > \theta_{GA}$  and  $\exists cl.E_{c=correctness}, cl.exp =$   

    $cl.E_{c=correctness}$  then
16:     if  $Max\epsilon < cl.\epsilon$  then
17:        $Max\epsilon = cl.\epsilon$ 
18:     end if
19:   end if
20:    $cl.\epsilon_0 \leftarrow cl.\epsilon_0 + \beta(Max\epsilon + E_0 - cl.\epsilon_0)$ 
21:   update  $\kappa(cl), \kappa'(cl), F$ : パラメータ更新 (XCS と同様)
22: end for
```

Algorithm 18 のパラメータ更新 (後半) に対応する。その後、包摂条件や GA の条件を満たした場合に $[A]$ の分類子を対象として、包摂 (17 行目) や GA (20 行目) を行う。包摂条件では、包摂を行う分類子が包摂をする分類子のいずれよりも評価回数が多いかを確認する (12 行目)。上記の流れを 1 学習エピソードとして、学習及び GA による生成と淘汰を繰り返すことによって、環境に適した分類子集団を獲得する。

Algorithm 19 XCS-EC のアルゴリズム

```
1: while (! end of iterations) do
2:    $state \leftarrow$  environment: 環境から状態  $state$  を受け取る
3:   generate  $[M]$ :  $[P]$  から  $state$  に照合する分類子を選択
4:    $P(a_i) = \sum_{cl_k \in [M]|a_i} cl_k.p \times cl_k.F / \sum_{cl_l \in [M]|a_i} cl_l.F$ :  $[M]$  から各行動の予測値を算出
5:    $action \leftarrow P(action)$ :  $P(action)$  から行動を選択
6:   generate  $[A]$ :  $[M]$  から  $action$  に照合する分類子を選択
7:    $reward \leftarrow$  execute  $action$ : 行動を実行し, 環境より報酬を獲得
8:   update parameters (first half): パラメータ更新 (前半) (Algorithm 16)
9:   estimate correctness  $\rho'$ : 正誤の推測 (Algorithm 17)
10:  update parameters (second half): パラメータ更新 (後半) (Algorithm 18)
11:  for all classifier  $cl$  in  $[A]$  do
12:    if  $cl.exp > \theta_{sub}$  and  $\kappa = 1$  and  $cl.exp > 2^{numberof\#} \times \theta_{EC}$  then
13:       $subsumptionFlag = true$ 
14:    end if
15:  end for
16:  if  $subsumptionFlag = true$  then
17:    run subsumption:  $[A]$  内の分類子を対象に包摂を実行
18:  end if
19:  if run GA then
20:    generate new classifiers: 子個体の生成
21:    delete classifier: 削除
22:  end if
23: end while
```

7.4 実験

7.4.1 実験内容

XCS-RG および XCS-EC は入出力および判定の信憑性における不確実性に対応可能なため,

- 1) 不確実性を伴わない ($P_I = P_O = \sigma_R = 0$) 場合
- 2) 入力における不確実性 ($P_I = 0.05, 0.1, 0.2$) を伴う場合

- 3) 出力における不確実性 ($P_O = 0.2, 0.3, 0.4$) を伴う場合
- 4) 判定の信憑性における不確実性 ($\sigma_R = 200, 700, 1500$) を伴う場合
- 5) 入力における不確実性および出力における不確実性 ($(P_I, P_O) = (0.05, 0.15), (0.05, 0.2), (0.1, 0.2)$) を伴う場合
- 6) 入力における不確実性および判定の信憑性における不確実性 ($(P_I, \sigma_R) = (0.03, 500), (0.1, 200)$) を伴う場合
- 7) 出力における不確実性および判定の信憑性における不確実性 ($(P_O, \sigma_R) = (0.1, 500), (0.2, 100)$) を伴う場合
- 8) 入力における不確実性, 出力における不確実性および判定の信憑性における不確実性 ($(P_I, P_O, \sigma_R) = (0.03, 0.15, 500)$) を伴う場合

を扱う。また、入力における不確実性, 出力における不確実性あるいは判定の信憑性における不確実性が単体で伴う場合は第 5 章および第 6 章における実験と同様に、不確実性の強度を以下の 2 種類のケースとした実験を行う。

- ケース U: すべての入力と出力の組において均一の不確実性強度
- ケース V: 入力と出力の組ごとに P_I, P_O あるいは σ_R を上限とした不確実性強度

ケース U (Uniform) では不確実性強度の強弱による手法の適応性を評価し、ケース V (Various) では不確実性強度が均一でない場合における適応性を評価する。

7.4.2 実験設定

XCS で用いるパラメータは第 5.5.2 項で記した値と同様の $\theta_{mna} = 2, P_{\#} = 0.35, p_I = 0.01, \epsilon_I = 0.01, F_I = 0.01, p_{explr} = 1.0, \beta = 0.2, \epsilon_0 = 10, \alpha = 0.1, \nu = 10, \theta_{sub} = 20, \theta_{GA} = 25, \chi = 0.8, \mu = 0.04, N = 800, \theta_{del} = 20, \delta = 0.1$ である。XCS-EC で新設されたパラメータ θ_{EC} はそれぞれ入力長 L に基づいて $\theta_{EC} = 2$ と定めた。Iteration は 1,000,000 回、試行数は不確実性が単体の時は 50、不確実性が複合した場合は 5 と設定する。また、評価基準は第 4.3 節で説明した正答率の平均、分類子数の平均および最適解集合 $[O]$ を獲得した試行数および最大 F の分類子数の平均である。

7.4.3 実験結果

1) 不確実性を伴わない場合

不確実性が伴わない場合 $P_I = P_O = \sigma_R = 0$ の時の正答率の平均と分類子数の平均を図 7.8 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。最適解集合 $[O]$ を獲得した試行数および最大 F の分類子数の平均を表 7.1 に示す。

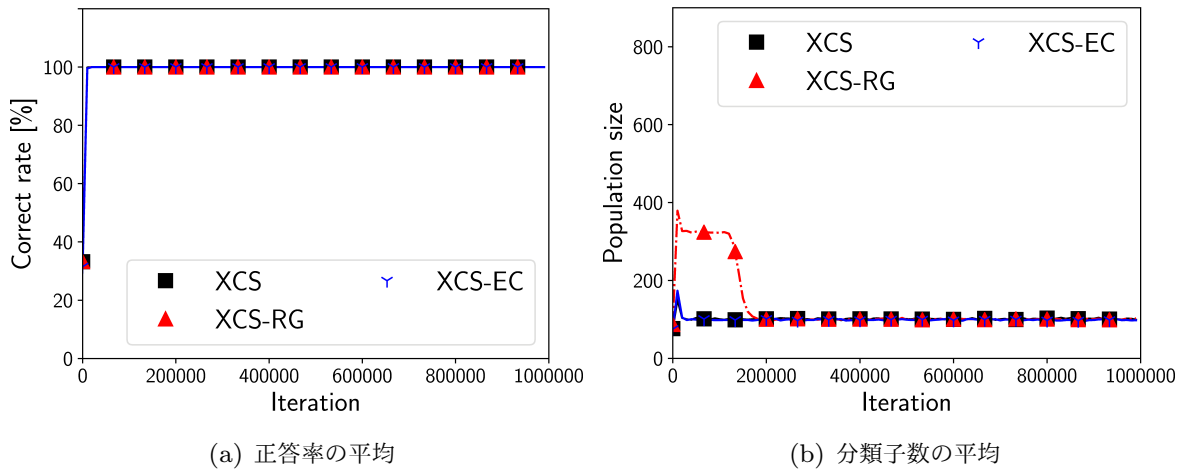


図 7.8 $P_I = P_O = \sigma_R = 0$

表 7.1 $P_I = P_O = \sigma_R = 0$ における最適解の獲得

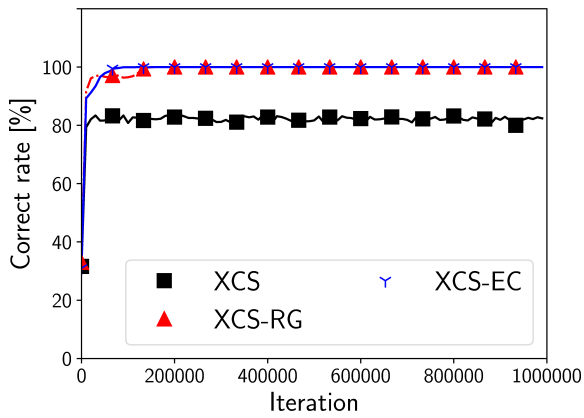
手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	47 (94%)	47 (94%)	32.02
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

XCS, XCS-RG および XCS-EC の正答率の平均は 100% に達し、分類子数の平均は 99.1, 102.6, 97.8 と 100 程度でほぼ同等である。XCS-RG および XCS-EC 共に [O] の獲得については上位 32 個の分類子に限定しても、全ての試行で獲得した。また、最大 F の分類子数の平均は 32 であった。このことから、不確実性を伴わない環境において XCS-RG, XCS-EC とともに分類子の正確性を正しく認識し分類子の一般化が可能であることが分かる。

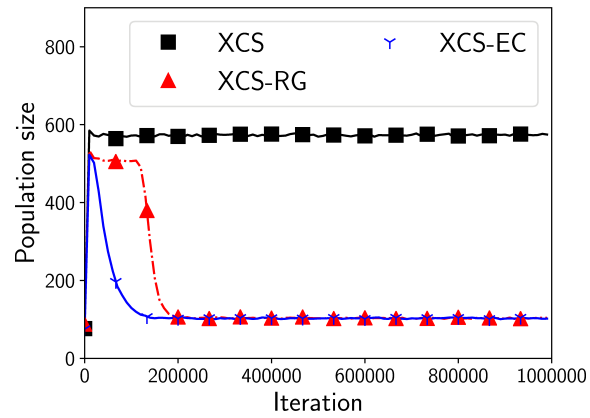
2) 入力における不確実性を伴う場合

入力における不確実性が $P_I = 0.05, 0.1, 0.2$ (ケース U) の時の正答率の平均と分類子数の平均をそれぞれ図 7.9, 7.10, 7.11 に示し、最適解集合 [O] を獲得した試行数および最大 F の分類子数の平均を表 7.2, 7.3, 7.4 に示す。 $\sup\{P_I\} = 0.05, 0.1, 0.2$ (ケース V) の時の正答率の平均と分類子数の平均をそれぞれ図 7.12, 7.13, 7.14 に示し、最適解集合 [O] を獲得した試行数および最大 F の分類子数の平均を表 7.5, 7.6, 7.7 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。

ケース U において $P_I = 0.05$ の場合は XCS-RG および XCS-EC は正答率の平均 100% に達し、分類子数の平均は 100 程度であり、[O] は全ての試行において獲得されており最大 F の分類子数の平均は 32 であった。XCS は XCS-RG および XCS-EC それぞれと有

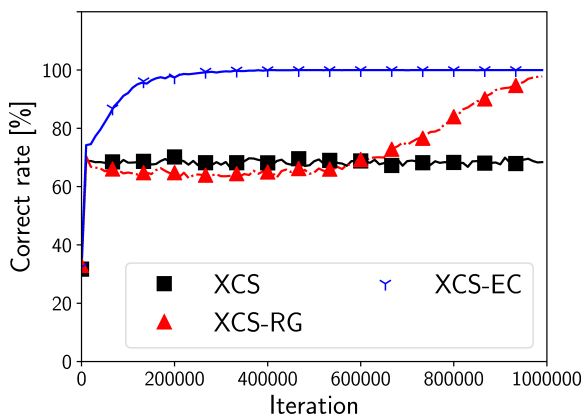


(a) 正答率の平均

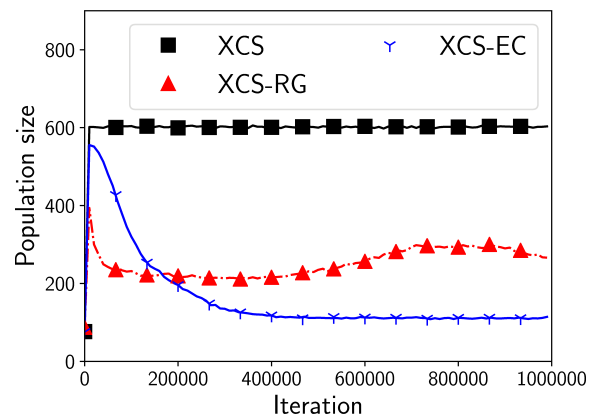


(b) 分類子数の平均

図 7.9 ケース U: $P_I = 0.05$

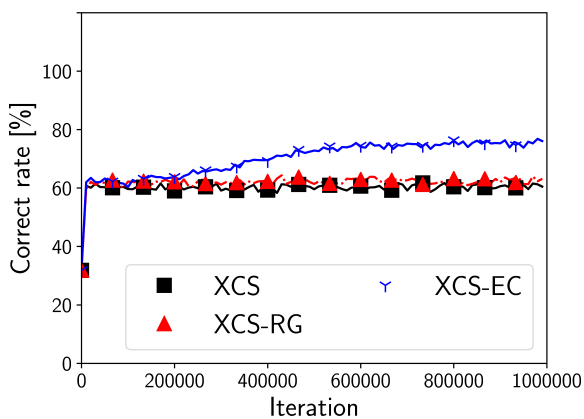


(a) 正答率の平均

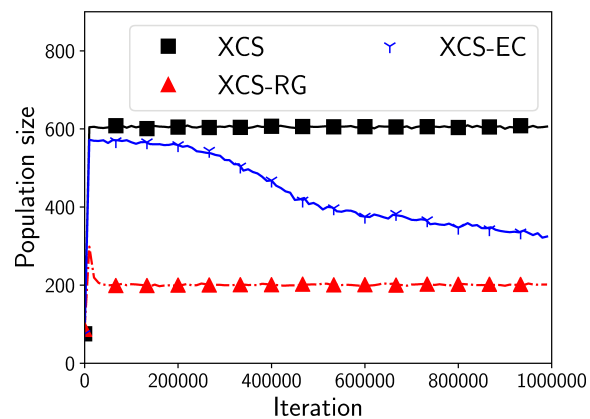


(b) 分類子数の平均

図 7.10 ケース U: $P_I = 0.1$

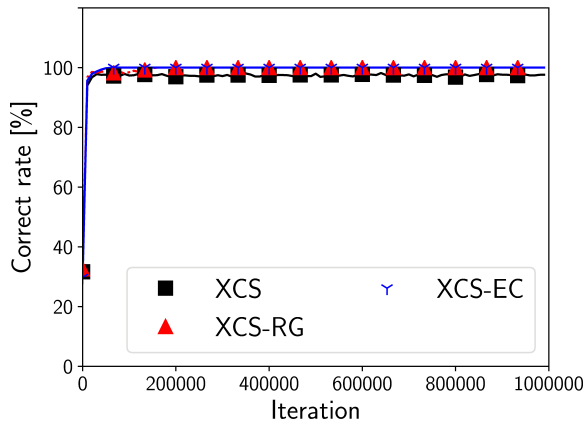


(a) 正答率の平均

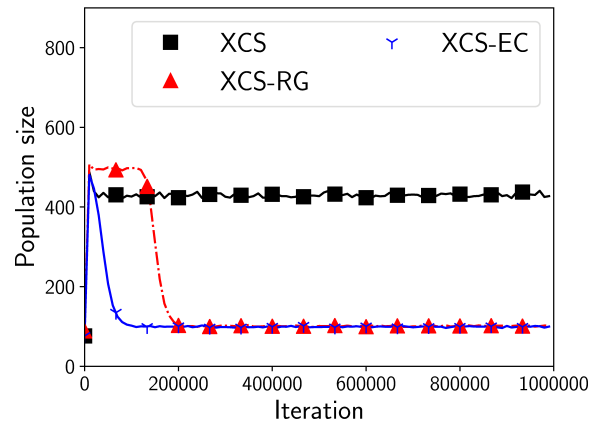


(b) 分類子数の平均

図 7.11 ケース U: $P_I = 0.2$

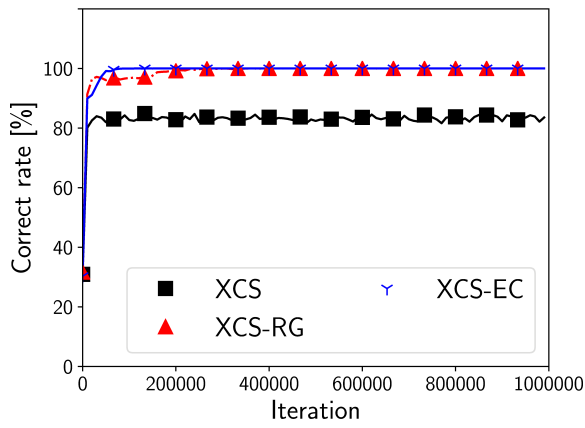


(a) 正答率の平均

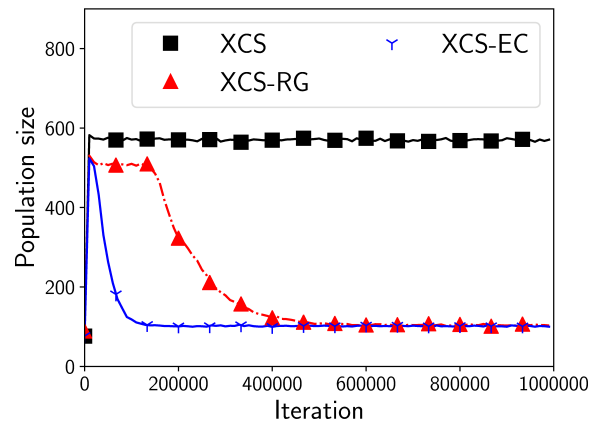


(b) 分類子数の平均

図 7.12 ケース V: $\sup\{P_I\} = 0.05$

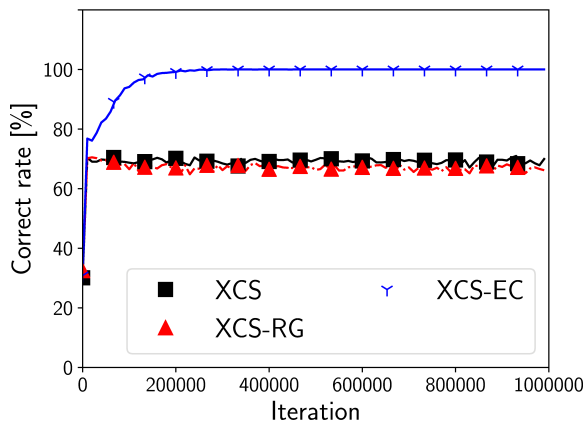


(a) 正答率の平均

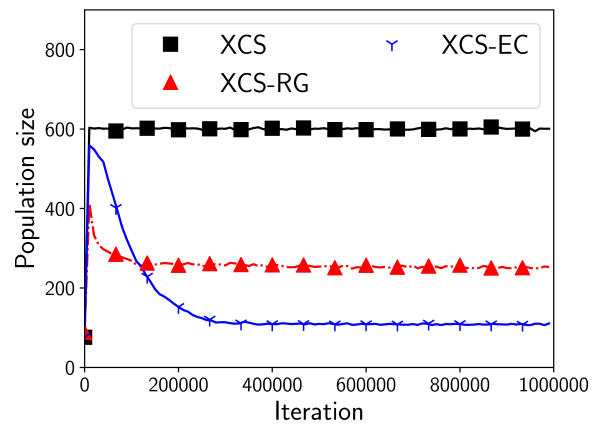


(b) 分類子数の平均

図 7.13 ケース V: $\sup\{P_I\} = 0.1$



(a) 正答率の平均



(b) 分類子数の平均

図 7.14 ケース V: $\sup\{P_I\} = 0.2$

表 7.2 ケース U: $P_I = 0.05$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	56.08
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

表 7.3 ケース U: $P_I = 0.1$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	50.16
XCS-RG	2 (4%)	0 (0%)	38.06
XCS-EC	50 (100%)	49 (98%)	32

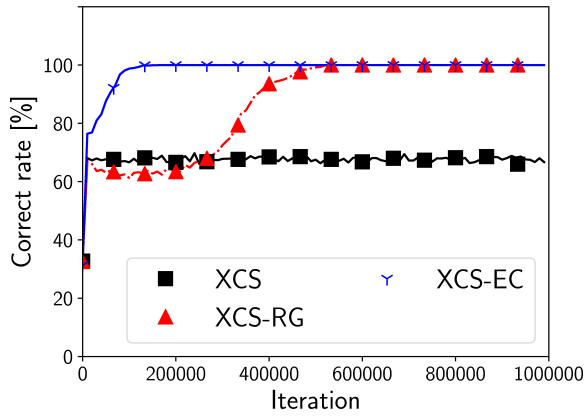
表 7.4 ケース U: $P_I = 0.2$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.24
XCS-RG	0 (0%)	0 (0%)	2.10
XCS-EC	0 (0%)	0 (0%)	25.12

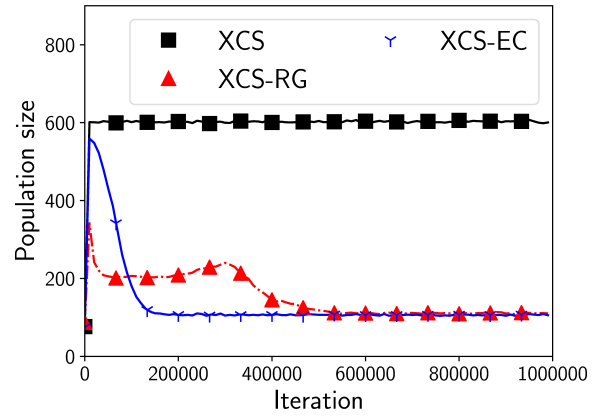
意差が見られた ($p < 0.01$) が, XCS-RG と XCS-EC の間では有意差は見られなかった. しかし, $P_I = 0.1, 0.2$ と不確実性の強度が増すと正答率の平均が低下し, XCS-RG の分類子数の平均は増加した. XCS-EC は $P_I = 0.2$ では Iteration の終盤においても正答率の平均は上昇し分類子は減少していることから学習が収束していないことが分かる. [O] を獲得した試行数は共に減少した. ケース V においてケース U と類似した結果が得られた. $\sup\{P_I\} = 0.05, 0.1$ の場合に正答率の平均 100% に達し, 分類子数の平均は 100 程度であり, [O] は全ての試行において獲得され最大 F の分類子数の平均は 32 であり, XCS は XCS-RG および XCS-EC それぞれと有意差が見られた ($p < 0.01$) が, XCS-RG と XCS-EC の間では有意差は見られなかった. $\sup\{P_I\} = 0.2$ の場合に XCS-RG は正答率の平均が低下し, 分類子数の平均は増加した. [O] を獲得した試行数も減少し最大 F の分類子数の平均は 32 よりも小さい値を取った. XCS-EC は 1 試行で上位 32 個で [O] を獲得できなかったが, 不確実性を伴わない場合とほぼ同等の結果を得られた.

3) 出力における不確実性を伴う場合

出力における不確実性が $P_O = 0.2, 0.3, 0.4$ (ケース U) の時の正答率の平均と分類子数の平均をそれぞれ図 7.15, 7.16, 7.17 に示し, 最適解集合 [O] を獲得した試行数および最大 F の分類子数の平均を表 7.8, 7.9, 7.10 に示す. $\sup\{P_O\} = 0.2, 0.3, 0.4$ (ケース V) の時の正答率の平均と分類子数の平均をそれぞれ図 7.18, 7.19, 7.20 に示し, 最適解集合

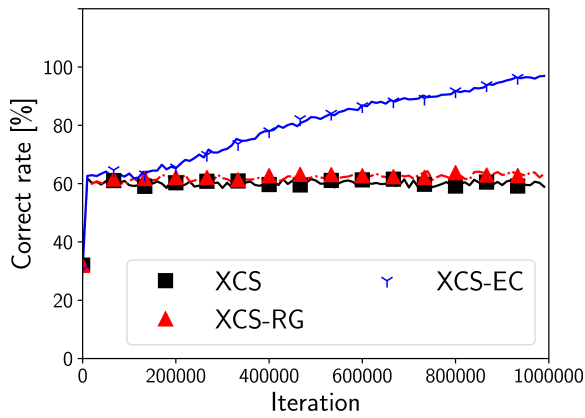


(a) 正答率の平均

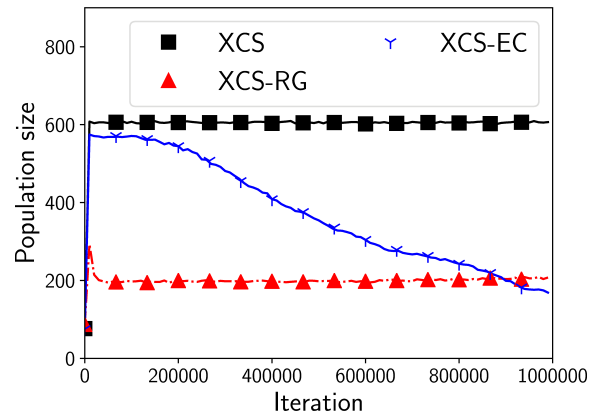


(b) 分類子数の平均

図 7.15 ケース U: $P_O = 0.2$

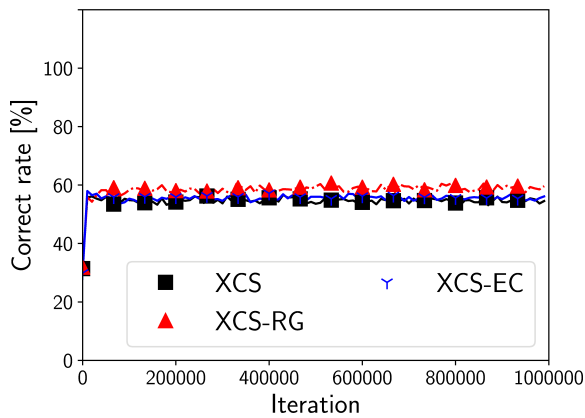


(a) 正答率の平均

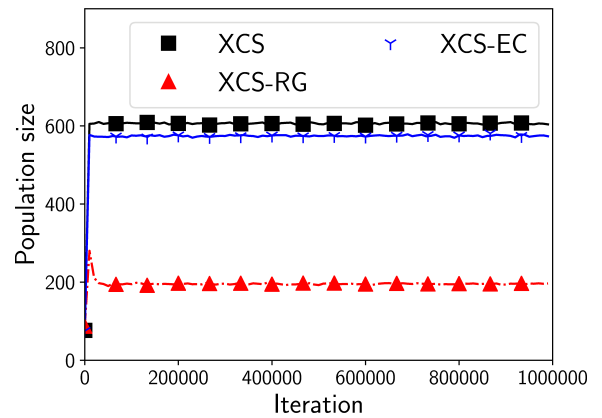


(b) 分類子数の平均

図 7.16 ケース U: $P_O = 0.3$

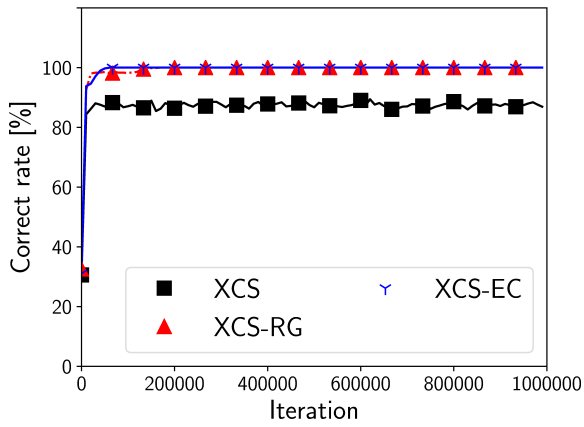


(a) 正答率の平均

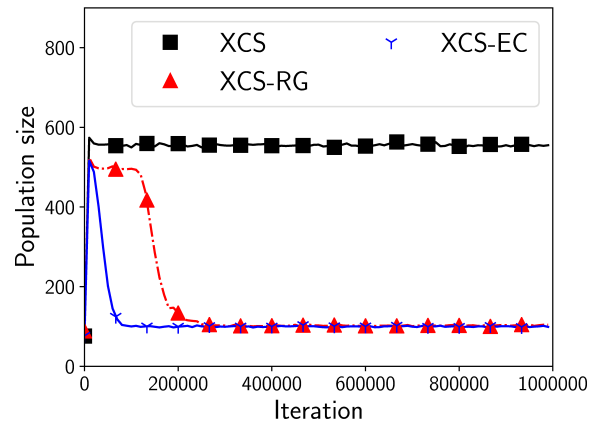


(b) 分類子数の平均

図 7.17 ケース U: $P_O = 0.4$

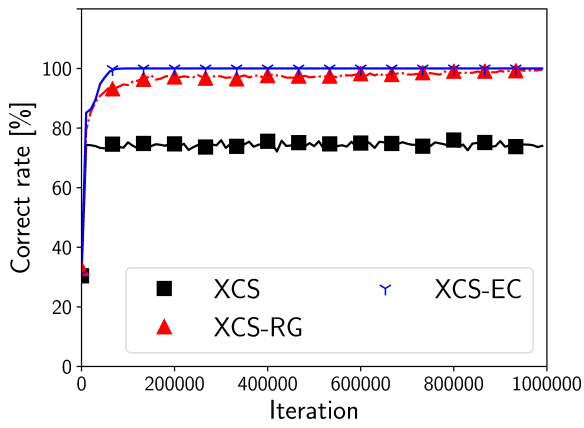


(a) 正答率の平均

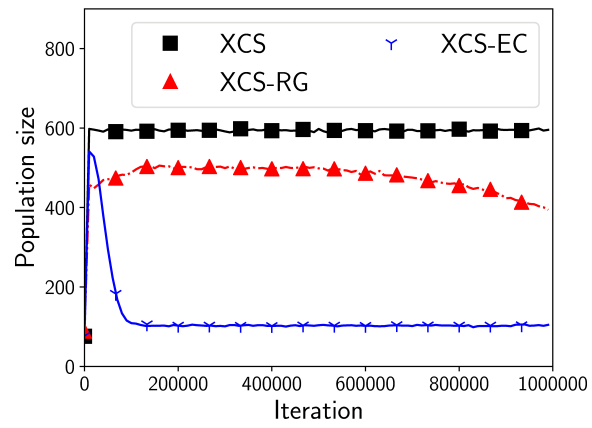


(b) 分類子数の平均

図 7.18 ケース V: $\sup\{P_O\} = 0.2$

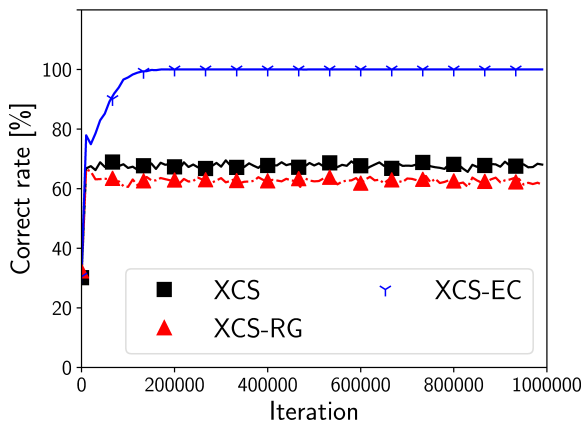


(a) 正答率の平均

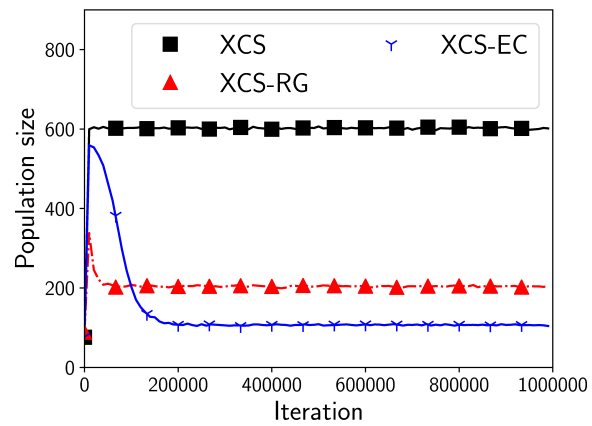


(b) 分類子数の平均

図 7.19 ケース V: $\sup\{P_O\} = 0.3$



(a) 正答率の平均



(b) 分類子数の平均

図 7.20 ケース V: $\sup\{P_O\} = 0.4$

表 7.5 ケース V: $\sup\{P_I\} = 0.05$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	52.34
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

表 7.6 ケース V: $\sup\{P_I\} = 0.1$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	58.08
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

表 7.7 ケース V: $\sup\{P_I\} = 0.2$ における最適解の獲得

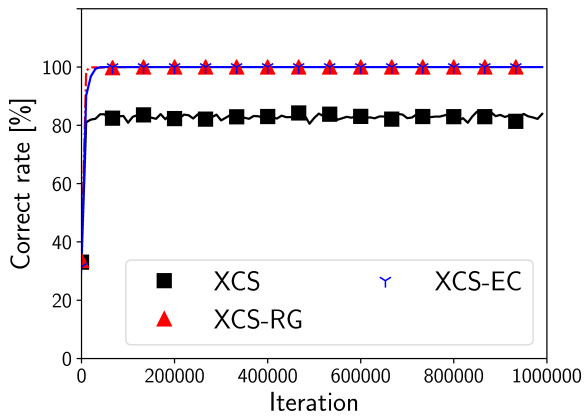
手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	49.86
XCS-RG	0 (0%)	0 (0%)	7.16
XCS-EC	50 (100%)	49 (98%)	32

[O] を獲得した試行数および最大 F の分類子数の平均を表 7.11, 7.12, 7.13 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。

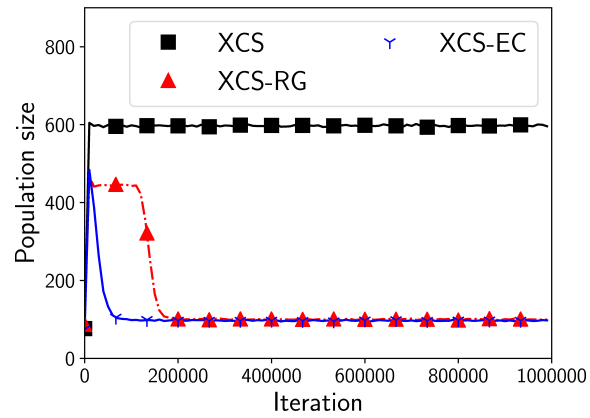
ケース U およびケース V のどちらの場合においても入力における不確実性を伴う場合と同様の結果を得た。XCS-RG は $P_O = 0.2$ (ケース U), $\sup\{P_O\} = 0.2$ (ケース V) の場合に、XCS-EC は $P_O = 0.2$ (ケース U), $\sup\{P_O\} = 0.2, 0.3, 0.4$ (ケース V) の場合に正答率の平均, 分類子数の平均, [O] の獲得試行数が高い水準を維持しており (XCS に対して $p < 0.01$), 最大 F の分類子数の平均は 32 を取っていた。また, XCS-RG は $P_O = 0.3$ (ケース U) の場合に Iteration の終盤においても正答率の平均は上昇し分類子は減少していることから学習が収束していないことが分かる。

4) 判定の信憑性における不確実性を伴う場合

判定の信憑性における不確実性が $\sigma_R = 200, 700, 1500$ (ケース U) の時の正答率の平均と分類子数の平均をそれぞれ図 7.21, 7.22, 7.23 に示し、最適解集合 [O] を獲得した試行数および最大 F の分類子数の平均を表 7.14, 7.15, 7.16 に示す。 $\sup\{P_I\} = 200, 700, 1500$ (ケース V) の時の正答率の平均と分類子数の平均をそれぞれ図 7.24, 7.25, 7.26 に示し、最適解集合 [O] を獲得した試行数および最大 F の分類子数の平均を表 7.17, 7.18, 7.19 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。

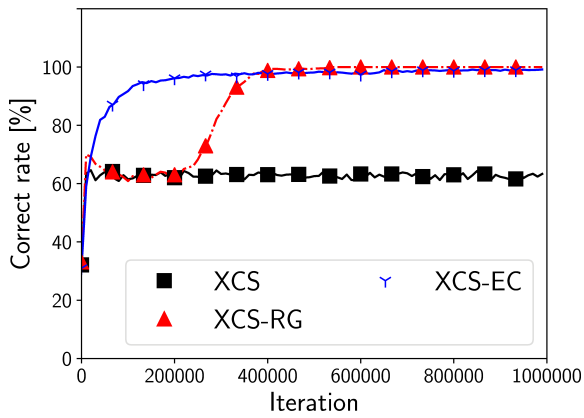


(a) 正答率の平均

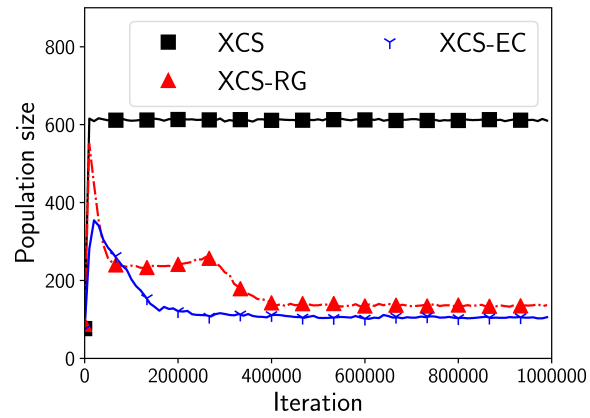


(b) 分類子数の平均

図 7.21 ケース U: $\sigma_R = 200$

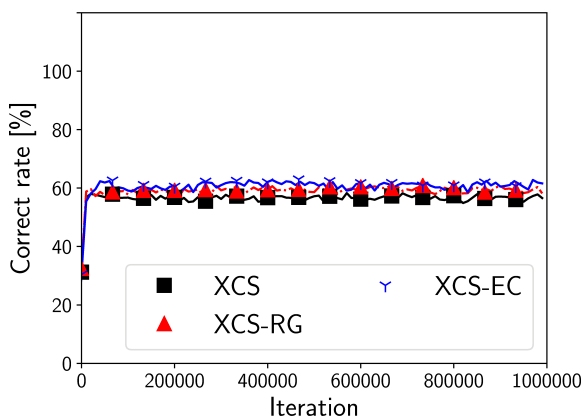


(a) 正答率の平均

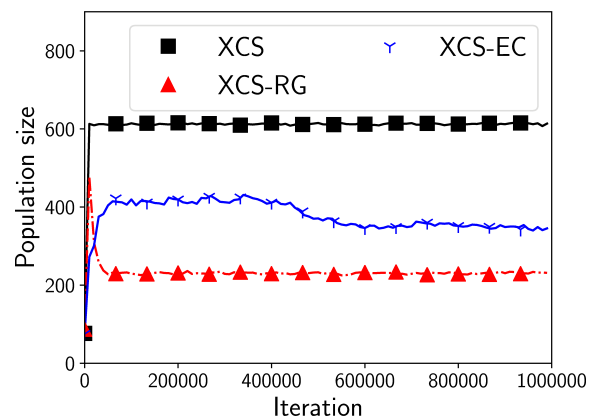


(b) 分類子数の平均

図 7.22 ケース U: $\sigma_R = 700$

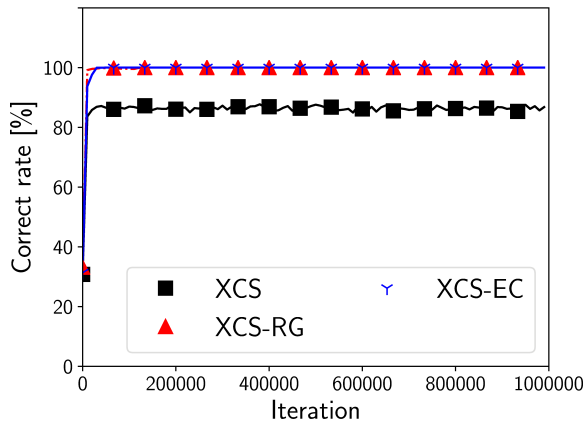


(a) 正答率の平均

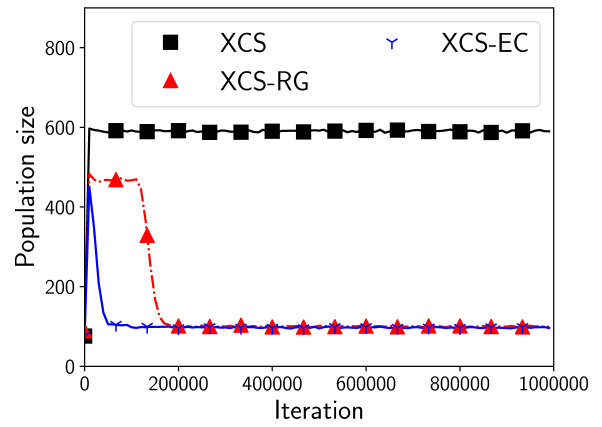


(b) 分類子数の平均

図 7.23 ケース U: $\sigma_R = 1500$

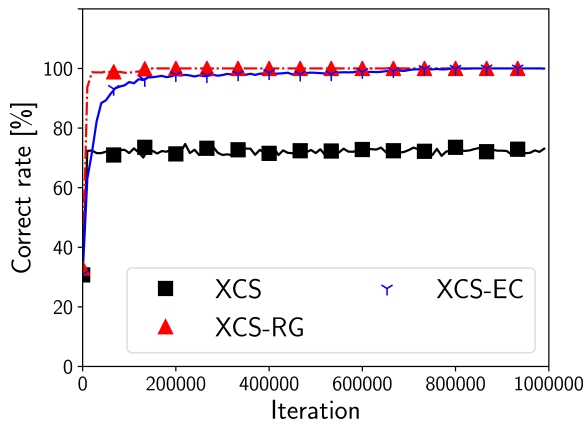


(a) 正答率の平均

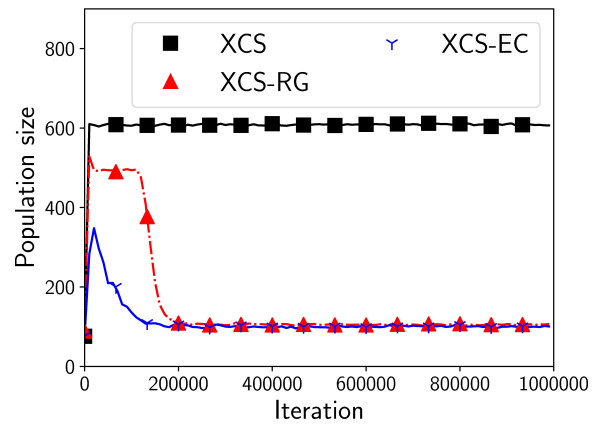


(b) 分類子数の平均

図 7.24 ケース V: $\sup\{\sigma_R\} = 200$

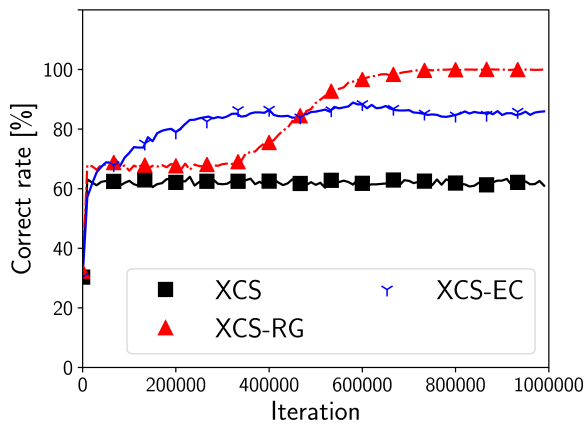


(a) 正答率の平均

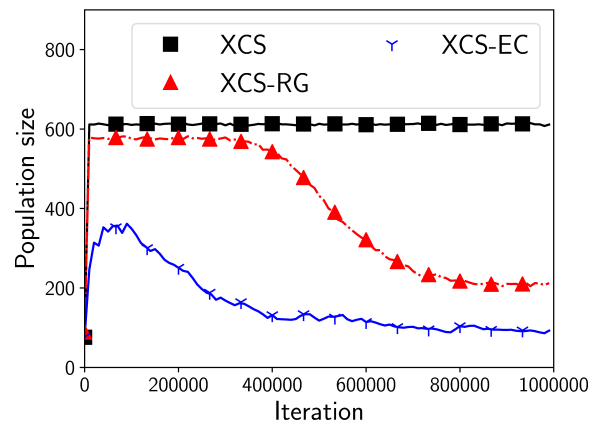


(b) 分類子数の平均

図 7.25 ケース V: $\sup\{\sigma_R\} = 700$



(a) 正答率の平均



(b) 分類子数の平均

図 7.26 ケース V: $\sup\{\sigma_R\} = 1500$

表 7.8 ケース U: $P_O = 0.2$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	52.62
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	49 (98%)	32

表 7.9 ケース U: $P_O = 0.3$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.74
XCS-RG	0 (0%)	0 (0%)	2.40
XCS-EC	39 (78%)	31 (62%)	33.22

表 7.10 ケース U: $P_O = 0.4$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	47.06
XCS-RG	0 (0%)	0 (0%)	2.06
XCS-EC	0 (0%)	0 (0%)	46.00

判定の信憑性における不確実性を伴う環境において入出力における不確実性を伴う場合と類似した結果が得られた。XCS-RG, XCS-EC とともにケース U で $\sigma_R = 200$ の場合に、ケース V では $\sup\{\sigma_R\} = 200, 700$ の場合に、正答率の平均, 分類子数の平均, [O] の獲得試行数が高い水準を維持しており (XCS に対して $p < 0.01$), 最大 F の分類子数の平均は 32 に近い値を取っていた。また, $\sigma_R = 700$ および $\sup\{\sigma_R\} = 1500$ の場合に XCS-RG は一部の試行を除いて [O] を獲得している。

5) 入力における不確実性および出力における不確実性を伴う場合

入力における不確実性と出力における不確実性の組み合わせ ($(P_I, P_O) = (0.05, 0.15), (0.05, 0.2)$) の場合の正答率の平均と分類子数の平均を図 7.27 および 7.28 に, [O] の獲得試行数と最大 F の分類子数の平均を表 7.20 および 7.21 に示す。図の横軸は Iteration を表し, 縦軸は正答率の平均あるいは分類子数の平均を表す。

XCS-EC は $(P_I, P_O) = (0.05, 0.15)$ および $(0.05, 0.2)$ のどちらの場合においても学習も分類子の正しい一般化もでき, 正答率の平均, 分類子数の平均, [O] の獲得試行数および最大 F の分類子数の平均全てが高い水準を維持している (XCS に対して $p < 0.01$)。XCS-RG は $(P_I, P_O) = (0.05, 0.15)$ の場合においてのみ学習および分類子の正しい一般化が行われたが (XCS に対して $p < 0.01$), $(P_I, P_O) = (0.05, 0.2)$ の場合において正答率が低く, 分類子数の平均が多く [O] の獲得試行数が少なく, 最大 F の分類子数の平均が

表 7.11 ケース V: $\sup\{P_O\} = 0.2$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	59.14
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

表 7.12 ケース V: $\sup\{P_O\} = 0.3$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	53.52
XCS-RG	7 (14%)	0 (0%)	52.62
XCS-EC	50 (100%)	50 (100%)	32

表 7.13 ケース V: $\sup\{P_O\} = 0.4$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0 (0%)	0 (0%)	50.96
XCS-RG	0 (0%)	0 (0%)	2.14
XCS-EC	50 (100%)	49 (98%)	32

32 よりも少ないことから学習が収束していないことが分かる。

6) 入力における不確実性および判定の信憑性における不確実性を伴う場合

入力における不確実性と判定の信憑性における不確実性の組み合わせ $((P_I, \sigma_R) = (0.03, 500), (0.1, 200))$ の場合の正答率の平均と分類子数の平均を図 7.29 および 7.30 に、[O] の獲得試行数と最大 F の分類子数の平均を表 7.22 および 7.23 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。

XCS-EC は $(P_I, \sigma_R) = (0.03, 500)$ および $(0.1, 200)$ のどちらの場合においても学習も分類子の正しい一般化もでき、正答率の平均、分類子数の平均、[O] の獲得試行数および最大 F の分類子数の平均全てが高い水準を維持している (XCS に対して $p < 0.01$)。XCS-RG は $(P_I, \sigma_R) = (0.03, 500)$ の場合においてのみ学習および分類子の正しい一般化が行われたが (XCS に対して $p < 0.01$)、 $(P_I, \sigma_R) = (0.1, 200)$ の場合において正答率が低く、分類子数の平均が多く [O] の獲得試行数が少なく、最大 F の分類子数の平均が 32 よりも少ないことから学習が収束していないことが分かる。

7) 出力における不確実性および判定の信憑性における不確実性を伴う場合

出力における不確実性と判定の信憑性における不確実性の組み合わせ $((P_O, \sigma_R) = (0.1, 500), (0.2, 100))$ の場合の正答率の平均と分類子数の平均を図 7.31 および 7.32 に、

表 7.14 ケース U: $\sigma_R = 200$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	59.98
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

表 7.15 ケース U: $\sigma_R = 700$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.22
XCS-RG	50 (100%)	49 (98%)	32
XCS-EC	43 (86%)	43 (86%)	31.24

表 7.16 ケース U: $\sigma_R = 1500$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	47.16
XCS-RG	0 (0%)	0 (0%)	2.42
XCS-EC	1 (2%)	0 (0%)	26.86

[O] の獲得試行数と最大 F の分類子数の平均を表 7.24 および 7.25 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。

XCS-RG は $(P_I, \sigma_R) = (0.2, 100)$ および $(0.1, 500)$ のどちらの場合においても学習も分類子の正しい一般化もでき、正答率の平均、分類子数の平均、[O] の獲得試行数および最大 F の分類子数の平均全てが高い水準を維持している (XCS に対して $p < 0.01$)。XCS-EC は $(P_I, \sigma_R) = (0.2, 100)$ の場合においてのみ学習および分類子の正しい一般化が行われたが (XCS に対して $p < 0.01$)、 $(P_I, \sigma_R) = (0.1, 500)$ の場合において 1 試行で正答率が低く、分類子数の平均が多く [O] の獲得試行数が少なく、最大 F の分類子数の平均が 32 よりも少ないことから適切に学習できず、正しく分類子を一般化できていない。XCS-EC は XCS-ER と同様に性能に安定性がない。

8) 入力における不確実性, 出力における不確実性および判定の信憑性における不確実性を伴う場合

入力における不確実性, 出力における不確実性および判定の信憑性における不確実性の組み合わせ ($(P_I, P_O, \sigma_R) = (0.03, 0.05, 200), (0.03, 0.15, 500)$) の場合の正答率の平均と分類子数の平均および [O] の獲得試行数と最大 F の分類子数の平均を図 7.33 および 7.34 に、[O] の獲得試行数と最大 F の分類子数の平均を表 7.26 および 7.27 に示す。

XCS-EC, XCS-RG 共に $(P_I, P_O, \sigma_R) = (0.03, 0.05, 200)$ の場合のみ、正答率が 100%

表 7.17 ケース V: $\sup\{\sigma_R\} = 200$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	51.66
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	50 (100%)	50 (100%)	32

表 7.18 ケース V: $\sup\{\sigma_R\} = 700$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	54.96
XCS-RG	50 (100%)	50 (100%)	32
XCS-EC	48 (96%)	48 (96%)	31.88

表 7.19 ケース V: $\sup\{\sigma_R\} = 1500$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.58
XCS-RG	44 (88%)	20 (40%)	31.50
XCS-EC	21 (42%)	19 (38%)	22.16

表 7.20 $(P_I, P_O) = (0.05, 0.15)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	51.6
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

表 7.21 $(P_I, P_O) = (0.05, 0.2)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	49.8
XCS-RG	2 (40%)	2 (40%)	19.8
XCS-EC	5 (100%)	5 (100%)	31.8

に達し、分類子の正しい一般化が行われ (XCS に対して $p < 0.01$), 全試行において [O] を獲得し、最大 F の分類子数の平均が 32 となった. $(P_I, P_O, \sigma_R) = (0.03, 0.15, 500)$ の場合には XCS-EC の方が正答率が高く、4 試行において [O] を獲得するなど分類子の一般化も進んでいる.

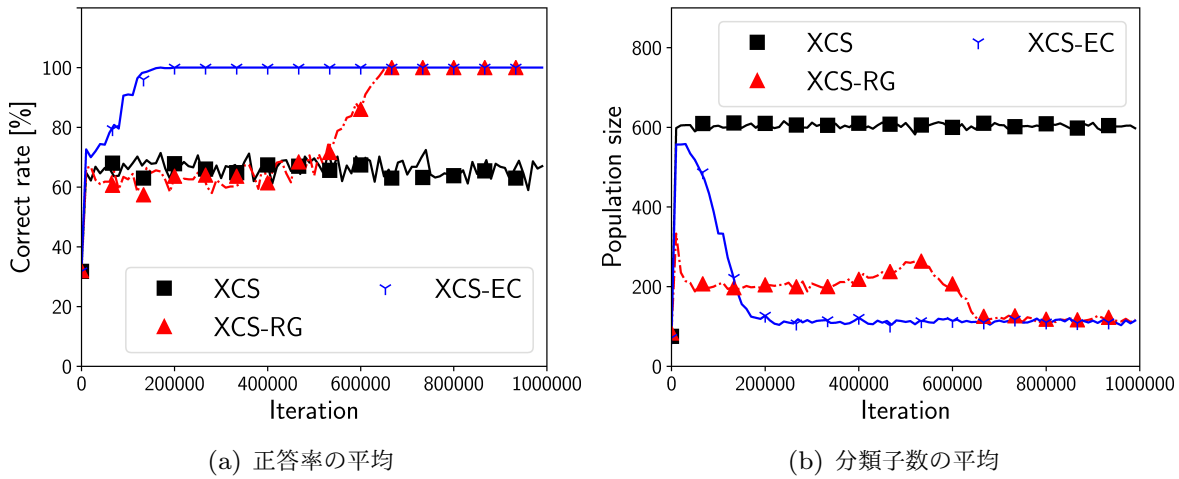


図 7.27 $(P_I, P_O) = (0.05, 0.15)$

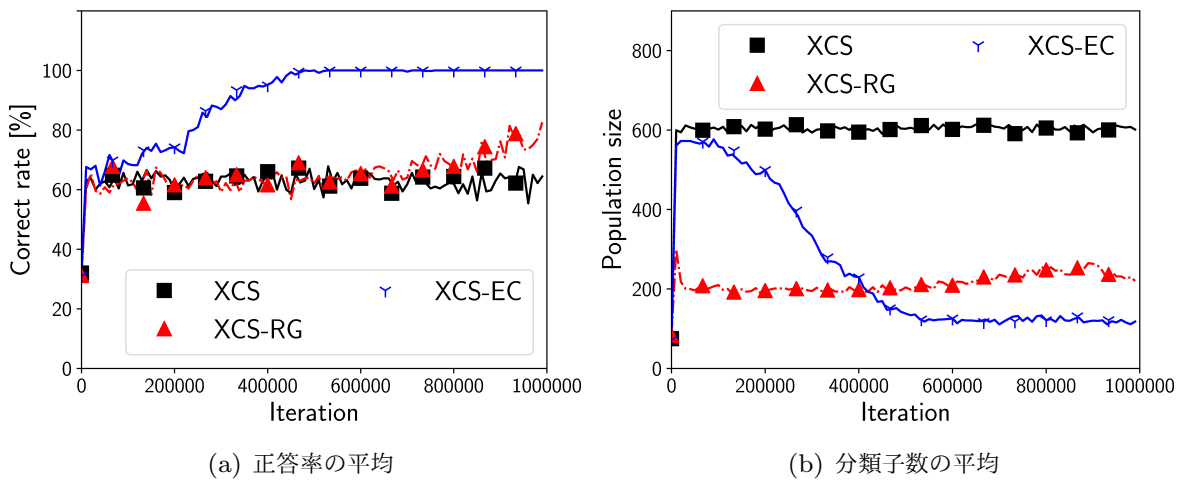


図 7.28 $(P_I, P_O) = (0.05, 0.2)$

7.5 考察

7.5.1 XCS-RG の収束速度

正答率の平均と分類子数の平均の収束速度について XCS-RG は他の手法と比べて明らかに収束速度が遅い。これは、式 (7.3), (7.4) に基づいて d_r および r_{cl} を求めるために Recording table の全状態行動対 (s, a) において平均 $M(s, a)$ をある程度正しく求める必要があるためである。さらに、伴う不確実性の強度が増すごとに収束速度が遅くなる。強度が大きな不確実性が伴うと (i) 入出力における不確実性が伴う環境においては認識した入力と出力の組における正誤と異なる報酬を獲得する頻度が多くなり、(ii) 判定の信憑性

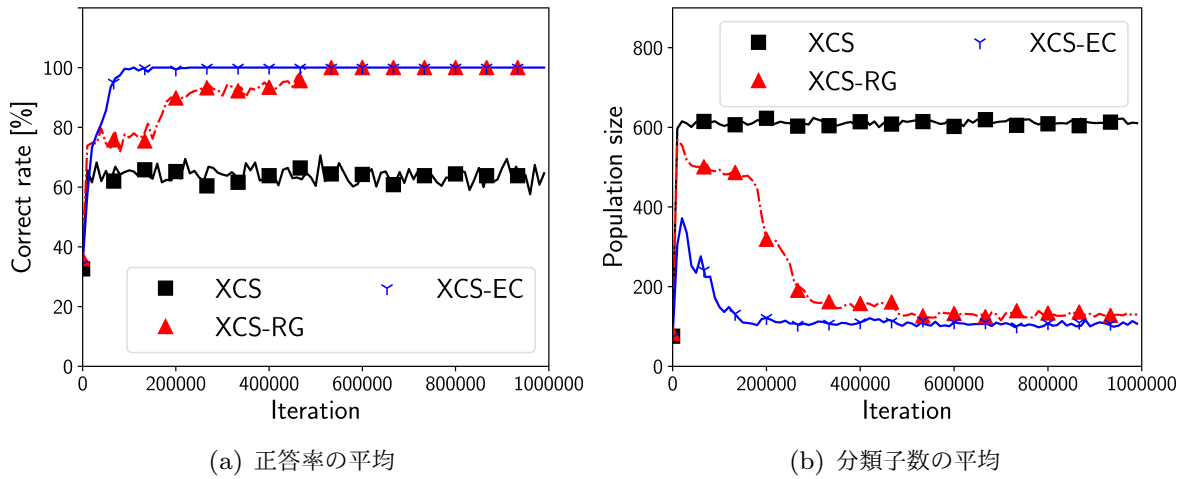


図 7.29 $(P_I, \sigma_R) = (0.03, 500)$

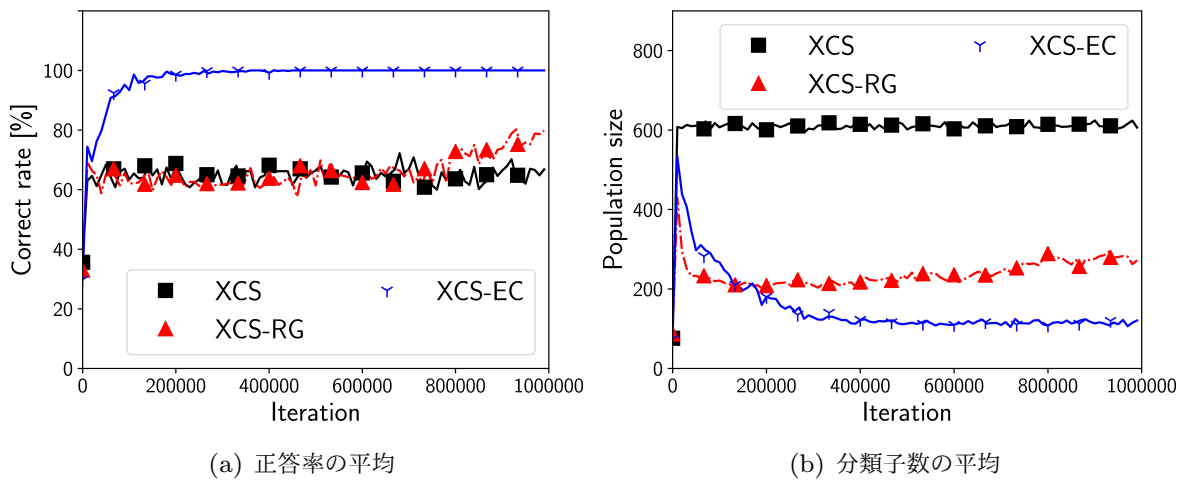


図 7.30 $(P_I, \sigma_R) = (0.1, 200)$

における不確実性が伴う環境では本来の報酬の値から離れた値の報酬を獲得しやすくなる。そのため、平均 $M(s, a)$ が分類子の正確性を正しく判定するのに十分な精度になるまでに時間がかかる。

7.5.2 XCS-RG の包摂条件

包摂条件 $r_{cl} < d_r$ について考察する。図 7.35, 7.36, 7.37 に入力における不確実性 $P_I = 0.05$, 出力における不確実性 $P_O = 0.2$, 判定の信憑性における不確実性 $\sigma_R = 200$ を伴う場合におけるある 1 シードの基準報酬間距離 d_r および正確な分類子と不正確な分類子の獲得報酬範囲 r_{cl} を示す。横軸は Iteration を、縦軸は d_r および r_{cl} を表し、線が d_r を、丸い点が正確な分類子の r_{cl} を、三角形の点が不正確な分類子の r_{cl} を表す。この

表 7.22 $(P_I, \sigma_R) = (0.03, 500)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	51.6
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

表 7.23 $(P_I, \sigma_R) = (0.1, 200)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	45.4
XCS-RG	0 (0%)	0 (0%)	18.0
XCS-EC	5 (100%)	5 (100%)	32

表 7.24 $(P_O, \sigma_R) = (0.1, 500)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.6
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	4 (80%)	4 (80%)	30.0

表 7.25 $(P_O, \sigma_R) = (0.2, 100)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	52.0
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

シードにおいては正確な分類子の r_{cl} は常に d_r を下回り、不正確な分類子の r_{cl} は常に d_r を上回っている。しかし、50 試行全体を見ると正確な分類子に対する包摂条件 $r_{cl} < d_r$ に誤判定が生じる場合もあった。この誤判定は 3 つのシードのみで起こっており、学習初期 (約 15 万 Iteration 付近まで) に特定の入力において、雑音の影響で入力異なっても獲得する報酬が近い値を取ったため d_r が正確な分類子の r_{cl} を下回った。評価回数が増えると、 d_r の値が増加し誤判定は起こらなくなった。また、不正確な分類子に対する包摂条件 $r_{cl} < d_r$ において誤判定は無かった。不正確な分類子を正確であると誤判定することがなく、正確な分類子のほとんどを正確であると判断可能なため、包摂条件 $r_{cl} < d_r$ は分類子の正確性の判定に適切である。

さらに実験的限界として入力における不確実性を伴う場合には $P_I = 0.05$ までであれば全ての試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し [O] を獲得す

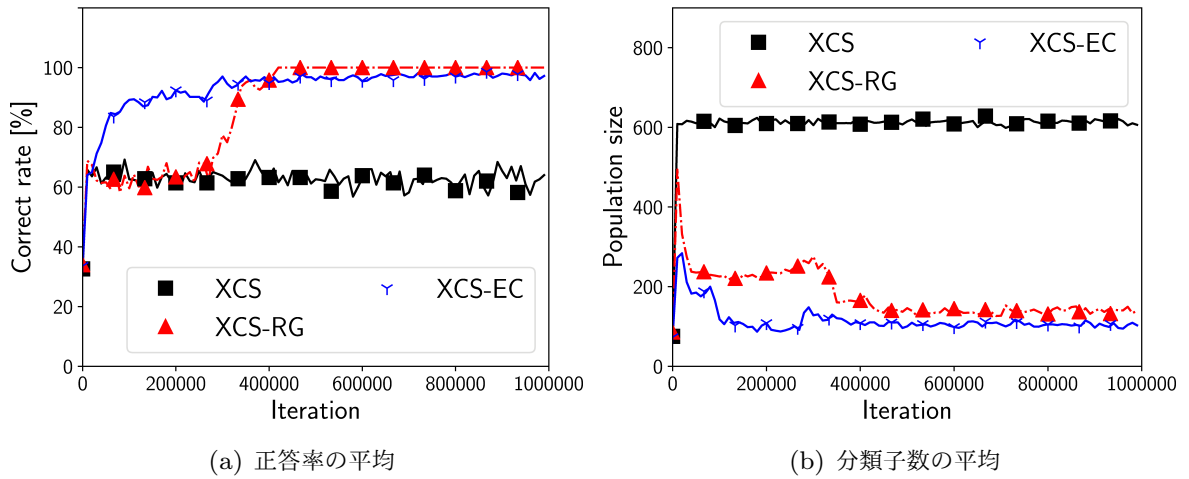


図 7.31 $(P_O, \sigma_R) = (0.1, 500)$

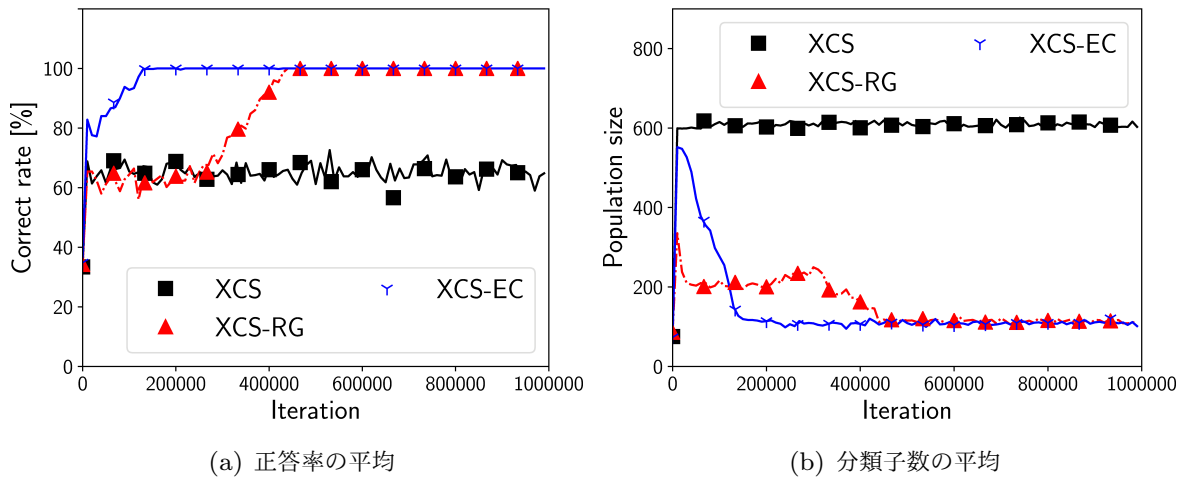


図 7.32 $(P_O, \sigma_R) = (0.2, 100)$

る。出力における不確実性を伴う場合には $P_O = 0.2$ までであれば全ての試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し $[O]$ を獲得する。判定の信憑性における不確実性を伴う場合には $\sigma_R = 200$ までであれば全ての試行で、 $\sigma_R = 700$ までであれば大半の試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し $[O]$ を獲得する。

7.5.3 XCS-EC の正誤推測の有効性とその限界

図 7.38 に判定における不確実性を伴う場合の正誤推測機構の例を示す。図 5.20 と同様に判定における不確実性を伴う場合であってもその時の入力の評価が他の入力における評価よりも多くなることおよび正確な分類子ほど exp が多くなる傾向があることからその

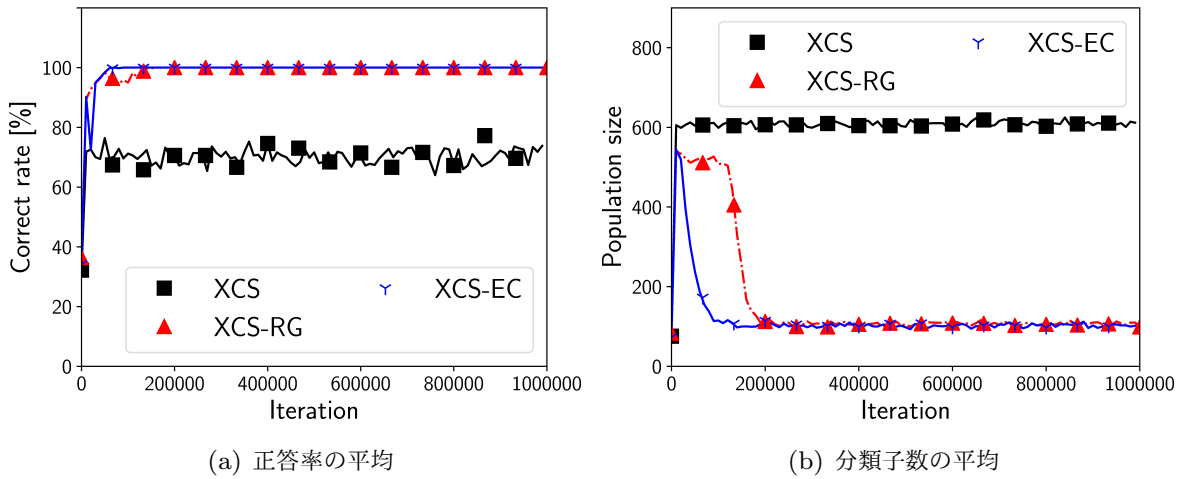


図 7.33 $(P_I, P_O, \sigma_R) = (0.03, 0.5, 200)$

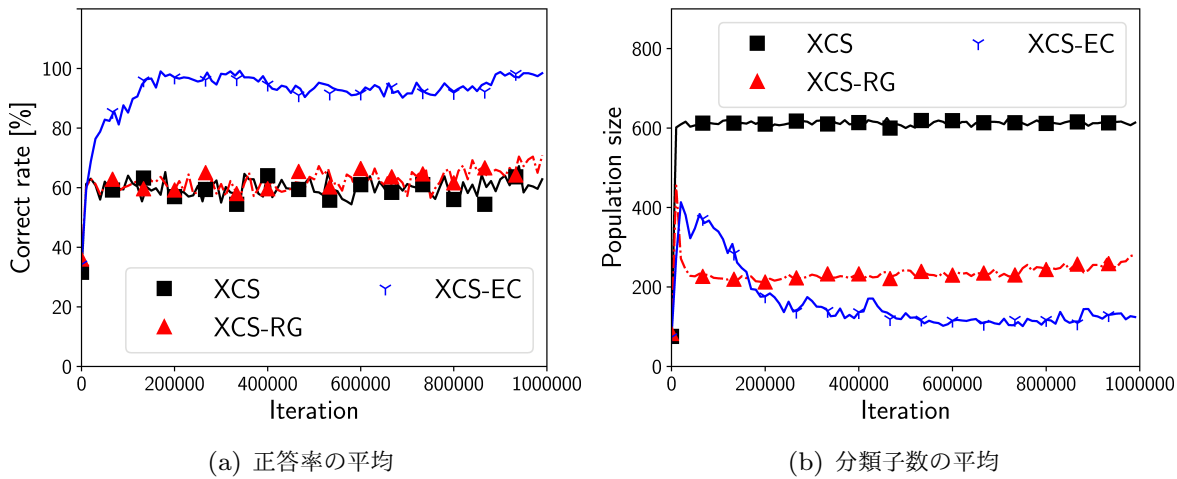


図 7.34 $(P_I, P_O, \sigma_R) = (0.03, 0.15, 500)$

時の入力における獲得報酬を推測可能である。

図 7.39(a), 7.39(b) および 7.40 に $P_I = 0.05$, $P_O = 0.2$ および $\sigma_R = 200$ における $[O]$ を獲得した試行の XCS-EC に入力された状態に対する報酬推測機構の正誤推測の正答率を示す。この推測率は評価指標の正答率の平均と同様に 100Iteration で移動平均を取ったものである。縦軸は正誤推測の正答率、横軸は Iteration を表す。 $P_I = 0.05$, $P_O = 0.2$, $\sigma_R = 200$ 全ての場合において 100% に達している。これらの結果から、XCS-EC の正誤推測機構は入力された状態に対して正解となる出力を選択することができる事が分かる。

さらに実験的限界として入力における不確実性を伴う場合には $P_I = 0.05$ までであれば全ての試行で、 $P_I = 0.1$ までであれば大半の試行で入力に対して正しい出力を選択

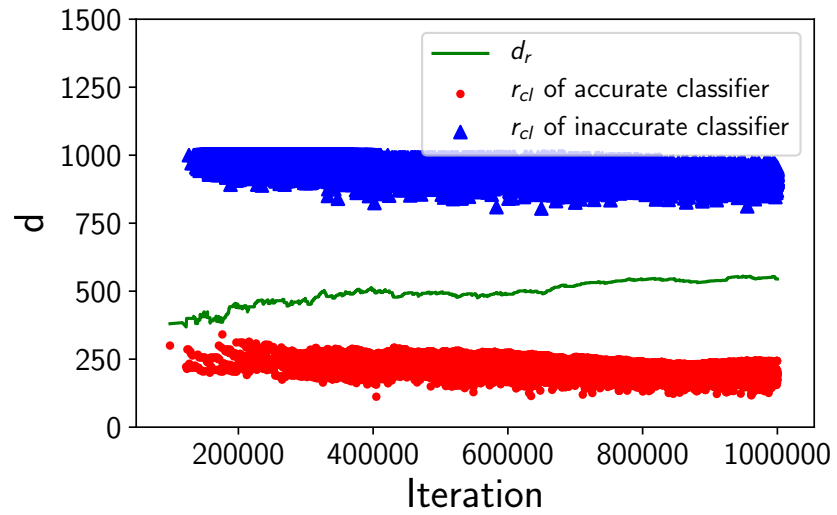


図 7.35 入力における不確実性下での d_r および r_{cl}

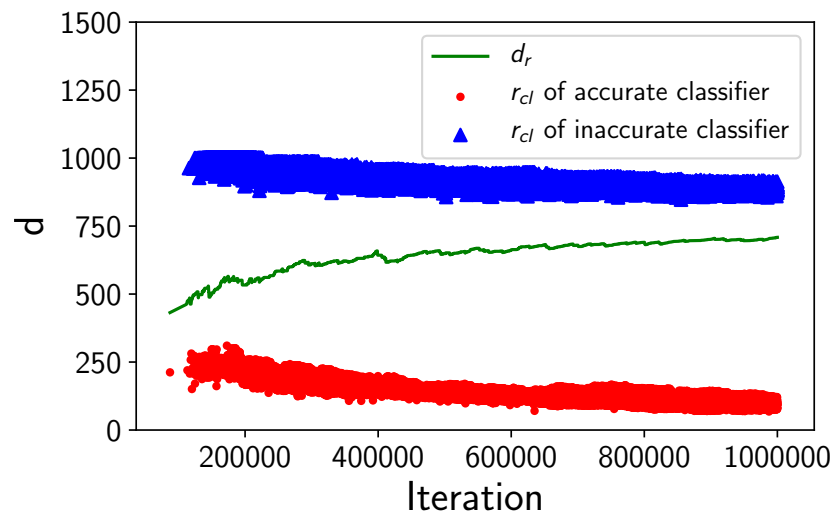


図 7.36 出力における不確実性下での d_r および r_{cl}

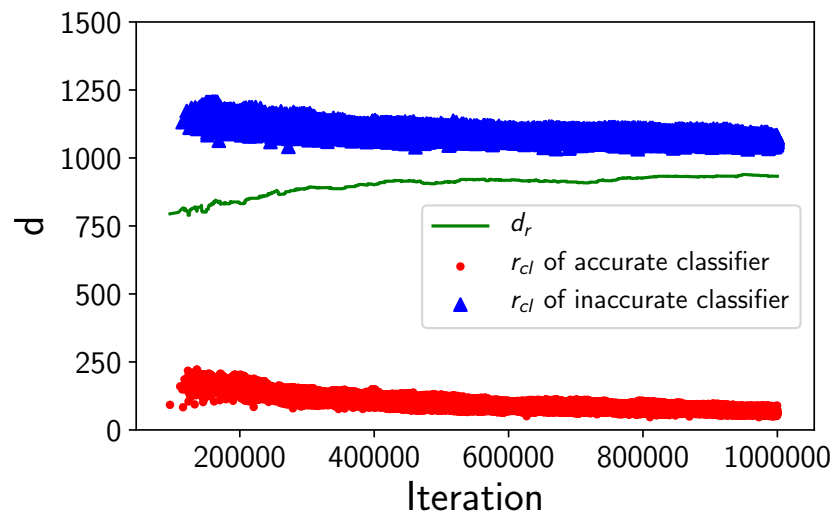


図 7.37 判定の信憑性における不確実性下での d_r および r_{cl}

表 7.26 $(P_I, P_O, \sigma_R) = (0.03, 0.5, 200)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	46.8
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

表 7.27 $(P_I, P_O, \sigma_R) = (0.03, 0.15, 500)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	46.8
XCS-RG	0 (0%)	0 (0%)	8.8
XCS-EC	4 (80%)	8 (80%)	30.4

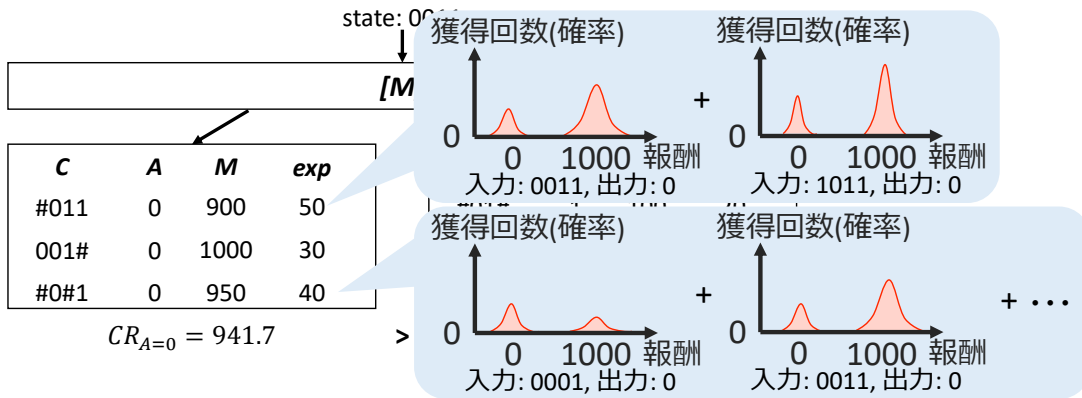
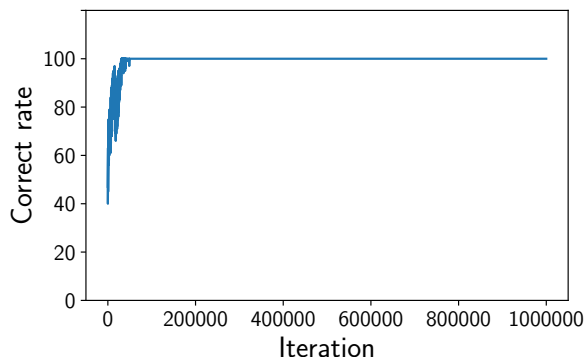


図 7.38 XCS-EC の判定における不確実性を伴う場合の正誤推測機構

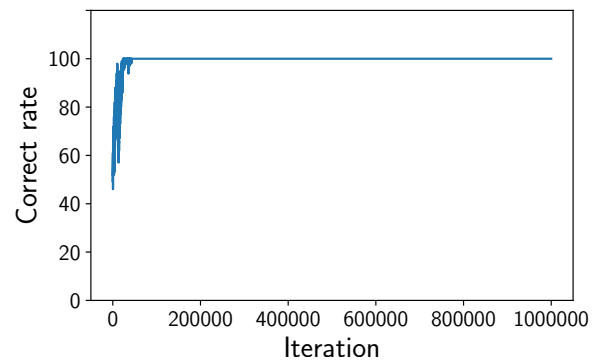
でき、分類子を正しく一般化し [O] を獲得する。出力における不確実性を伴う場合には $P_O = 0.3$ までであれば大半の試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し [O] を獲得する。判定の信憑性における不確実性を伴う場合には $\sigma_R = 200$ までであれば全ての試行で、 $\sigma_R = 700$ までであれば大半の試行で入力に対して正しい出力を選択でき、分類子を正しく一般化し [O] を獲得する。

7.5.4 XCS-EC の新設されたパラメータ

XCS-EC はハイパーパラメータ θ_{EC} を持つ。ここでは θ_{EC} の設定について述べる。図 7.41 から 7.44 に出力における不確実性 ($P_O = 0.2$) を伴う場合に θ_{EC} を 0, 2, 5, 10 に設定した際の正答率の平均と分類子数の平均を、図 7.45 から 7.48 に判定の信憑性における不確実性 ($\sigma_R = 200$) を伴う場合に θ_{EC} を 0, 2, 5, 10 に設定した際の正答率の平均と分類子数の平均を示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類



(a) $P_I = 0.05$ における正誤推測の正答率



(b) $P_O = 0.2$ における正誤推測の正答率

図 7.39 正誤推測の正答率

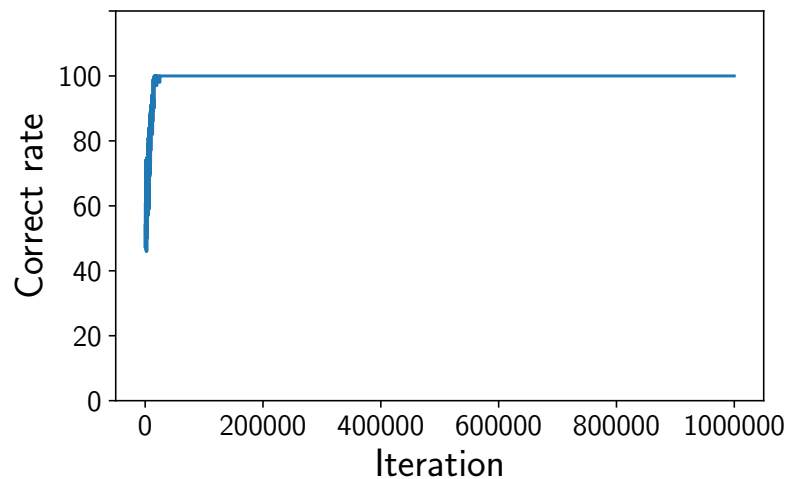
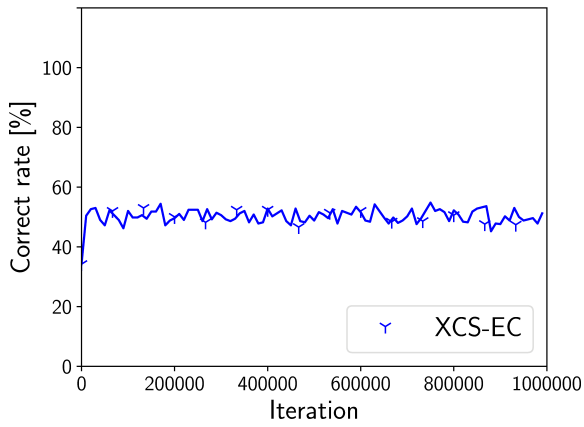


図 7.40 $\sigma_R = 200$ における正誤推測の正答率

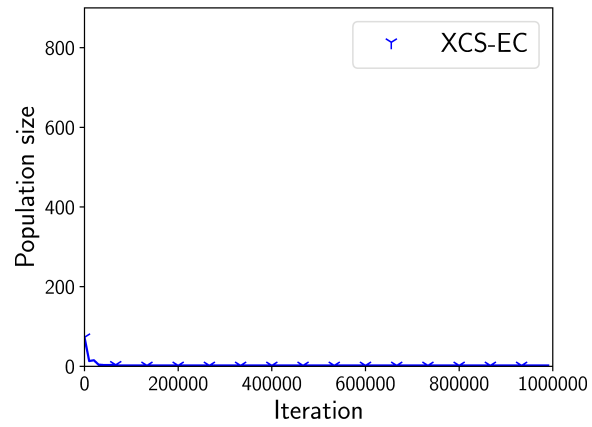
子数の平均を表す。 $\theta_{EC} = 0$ の場合において正答率の平均が約 50% で分類子数の平均が 2 と分類子が過剰に一般化されている。 $\theta_{EC} = 0$ では評価回数が少なくても ϵ_0 のための $Max\epsilon$ の計算対象に入る。そのため、不正確な分類子が評価回数が少ない内に $Max\epsilon$ の対象となり ϵ_0 が大きく設定され、不正確な分類子が正確と判定される。

$\theta_{EC} = 2$ の場合において、正答率の平均が 100% に達し、分類子数の平均は 100 程度と適切に学習ができている。

$\theta_{EC} = 5, 10$ の場合において正答率の平均は 100% には届かず、分類子数の平均が 200 程度と分類子の一般化が適切に行われていない。 θ_{EC} が大きいと分類子が包摂元になる条件を満たすまでに多くの評価回数を要することになり、正確な分類子による包摂が行われず n が増えにくく、包摂が行われるときに比べ削除されやすくなる。また、 θ_{EC} が大きい

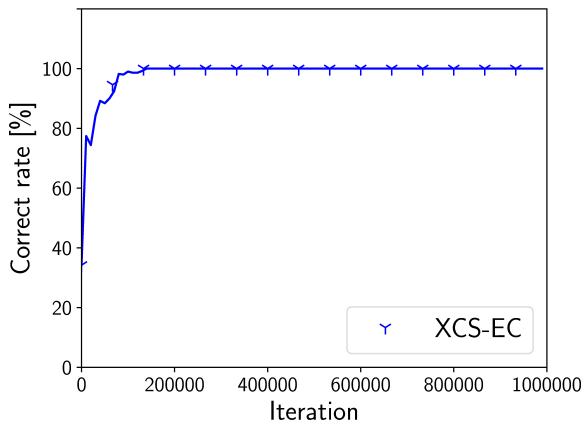


(a) 正答率の平均

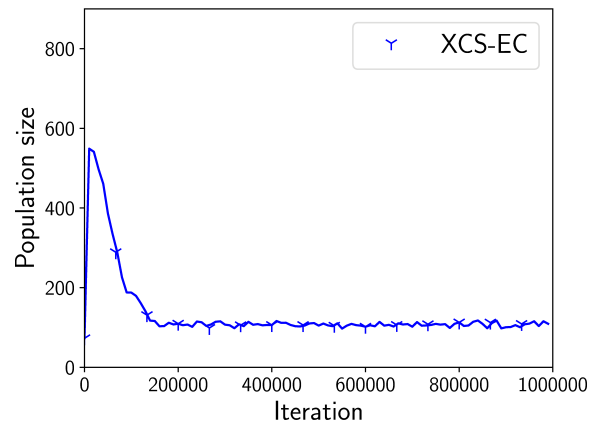


(b) 分類子数の平均

図 7.41 $\theta_{EC} = 0$ ($P_O = 0.2$)



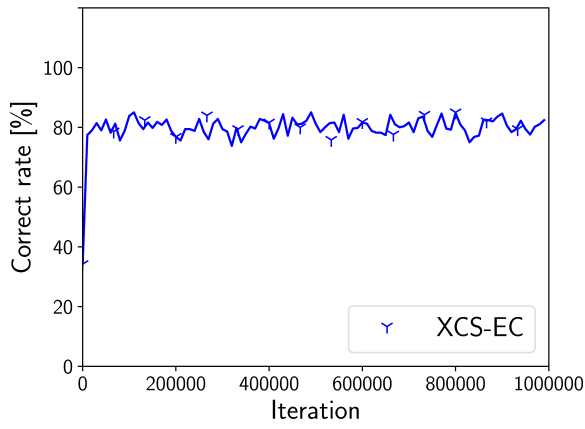
(a) 正答率の平均



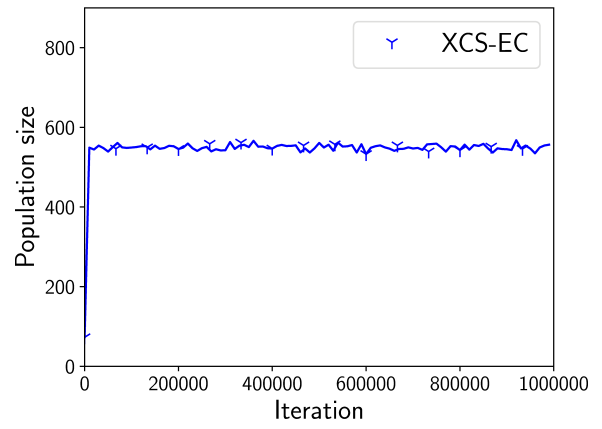
(b) 分類子数の平均

図 7.42 $\theta_{EC} = 2$ ($P_O = 0.2$)

いため過剰な一般化が行われず θ_{EC} が小さい場合と比べて正答率の平均の低下度合は大きくない。これらの結果から θ_{EC} を 2 程度に設定するのが好ましい。

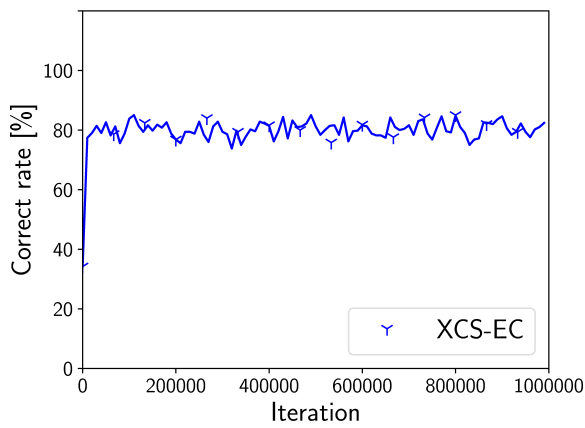


(a) 正答率の平均

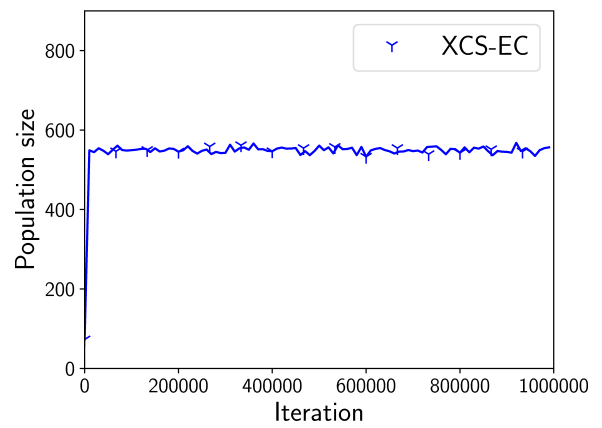


(b) 分類子数の平均

図 7.43 $\theta_{EC} = 5$ ($P_O = 0.2$)

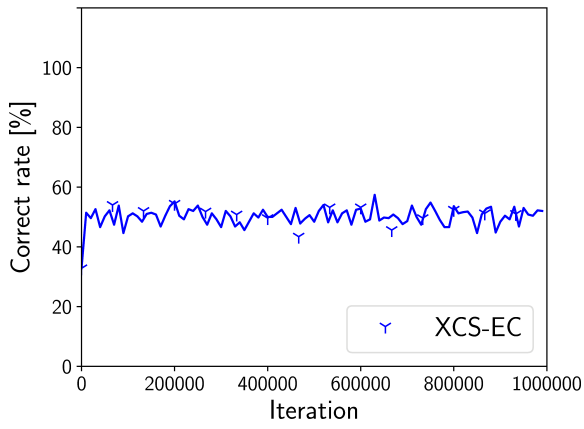


(a) 正答率の平均

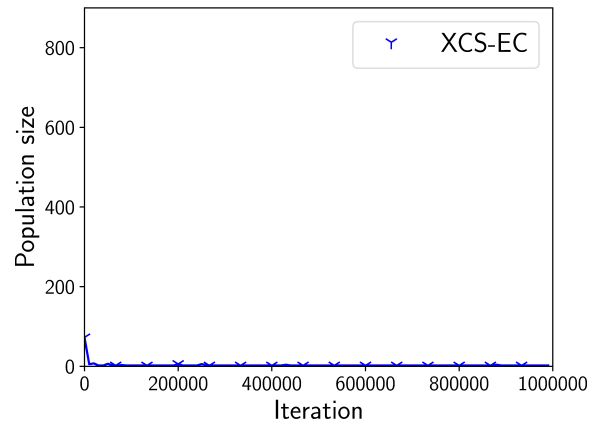


(b) 分類子数の平均

図 7.44 $\theta_{EC} = 10$ ($P_O = 0.2$)

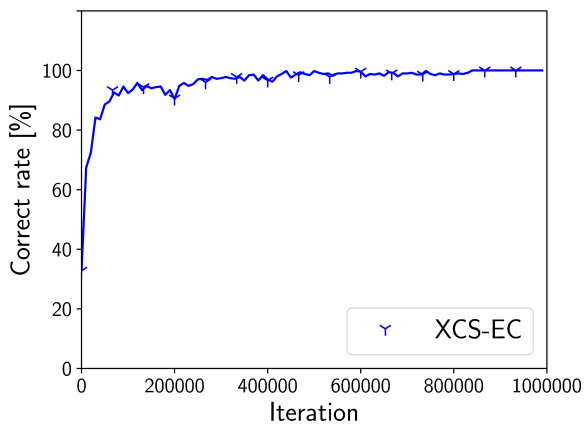


(a) 正答率の平均

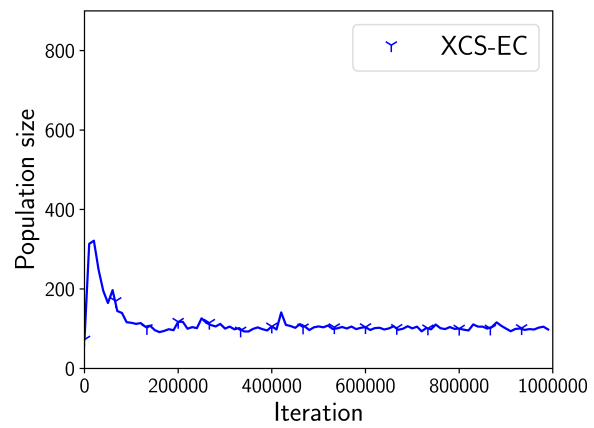


(b) 分類子数の平均

図 7.45 $\theta_{EC} = 0$ ($\sigma_R = 200$)

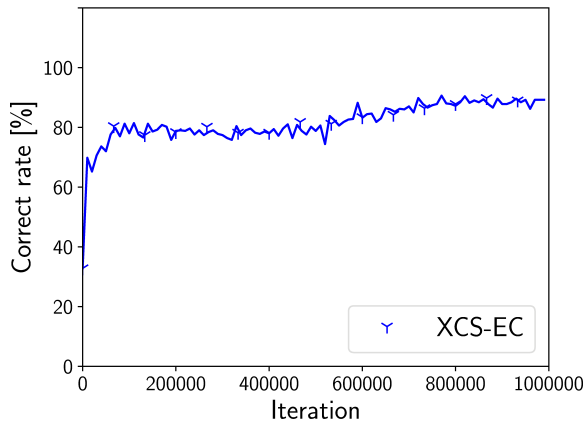


(a) 正答率の平均

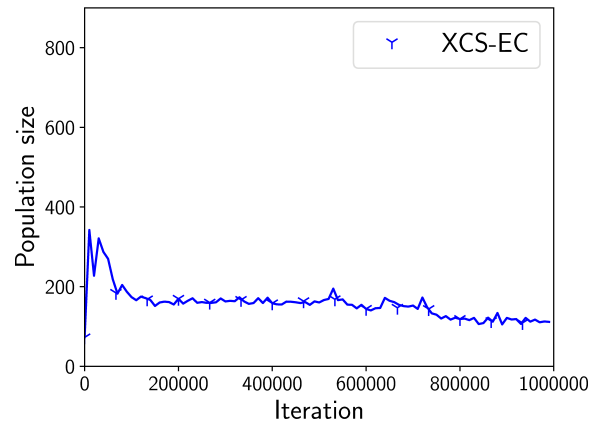


(b) 分類子数の平均

図 7.46 $\theta_{EC} = 2$ ($\sigma_R = 200$)

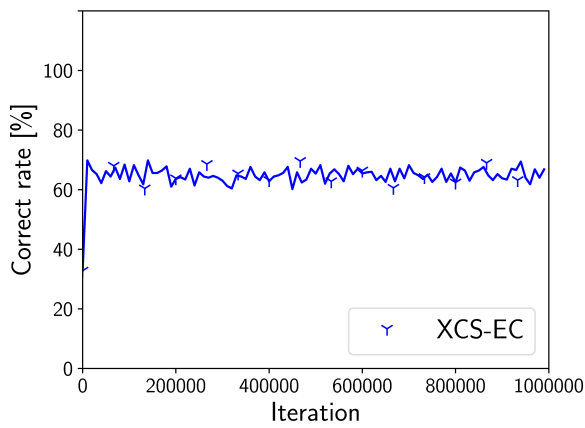


(a) 正答率の平均

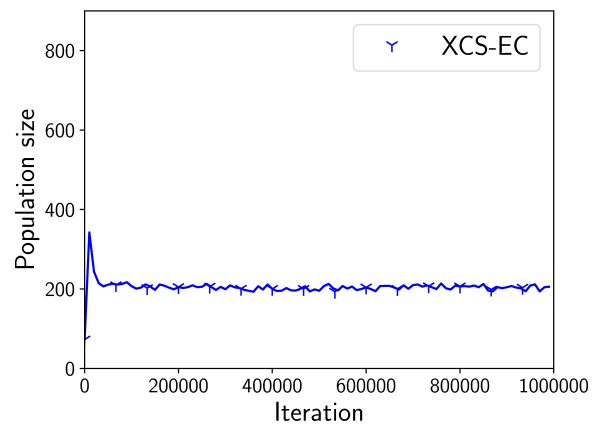


(b) 分類子数の平均

図 7.47 $\theta_{EC} = 5$ ($\sigma_R = 200$)



(a) 正答率の平均



(b) 分類子数の平均

図 7.48 $\theta_{EC} = 10$ ($\sigma_R = 200$)

第 8 章

様々な不確実性を伴う場合に対する適応性

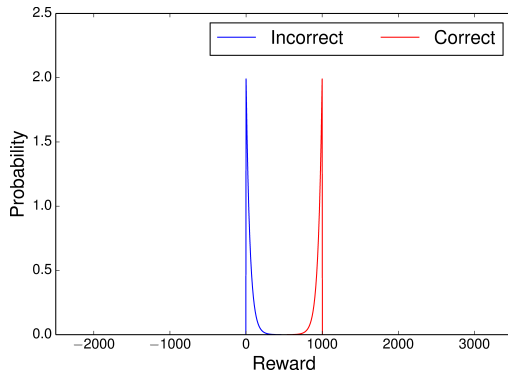
実問題では判定の信憑性における不確実性を伴うとしても、その不確実性の分布が正規分布とは限らない。不確実性を伴った報酬の分布の形状にかかわらず提案メカニズムが不確実性に対応し、適切に学習を行い知識を獲得できるかを調べる。

8.1 実験

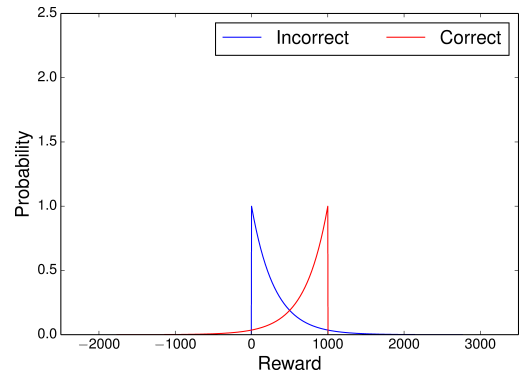
8.1.1 実験内容

XCS および提案手法の内判定の信憑性における不確実性に対応可能な XCS-RR, XCS-EC, XCS-RG を以下の不確実性を伴う場合に適用する。左右非対称かつ正規分布よりも裾が厚い分布である指数分布と左右対称だが上限と下限がある一様分布を扱う。正規分布, 指数分布および一様分布の結果を比較することで、獲得報酬の値が平均から遠い値を取る割合の多寡および正解時と不正解時の獲得報酬の重複の割合の多寡による学習および分類子の一般化に対する影響を調べる。指数分布は母数 λ の指数分布を x 倍したものとす。また、一様分布は 0 または R を中心として $\pm w$ の幅にしたものと、0 または R を基準（端）として幅 w にしたものとする。それぞれの分布の倍率 x や幅 w は不正解時の分布を基準としており、正解時にはその分布にマイナスを掛けて反転させたものとする。具体的には指数分布は $(\lambda, x) = (2, 100), (1, 300), (1, 400), (1, -100), (1, -500)$, 一様分布は $w = \pm 700, \pm 1500, \pm 2000, -2000, 500, 700$ に加えて $\sup\{w\} = 700$ （ケース V に相当）を上限にランダムに設定したものを扱う。図 8.1(a) から 8.2(f) にそれぞれの分布を示す。横軸は獲得報酬の値を表し、縦軸はその獲得報酬の確率密度を表す。赤色の線および青色の線はそれぞれ正解時および不正解時の報酬分布を表す。

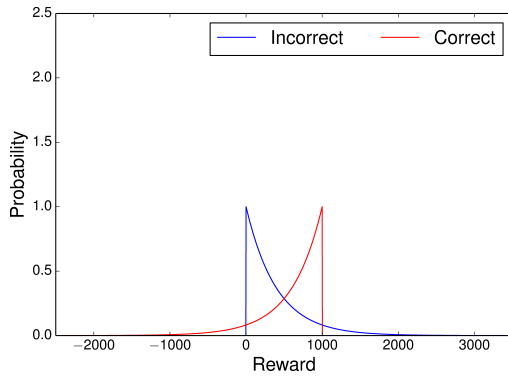
さらに、獲得報酬の分布の形状が常に同一とは限らない。分布の形状の非定常性に対応



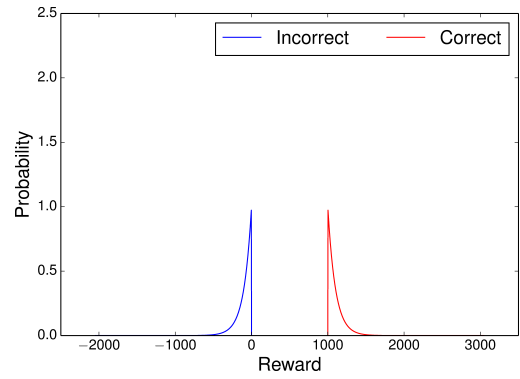
(a) $(\lambda, x) = (2, 100)$



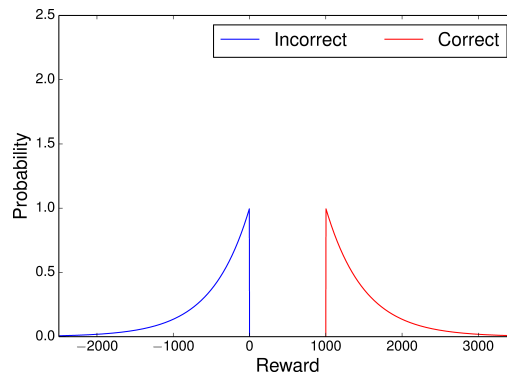
(b) $(\lambda, x) = (1, 300)$



(c) $(\lambda, x) = (1, 400)$



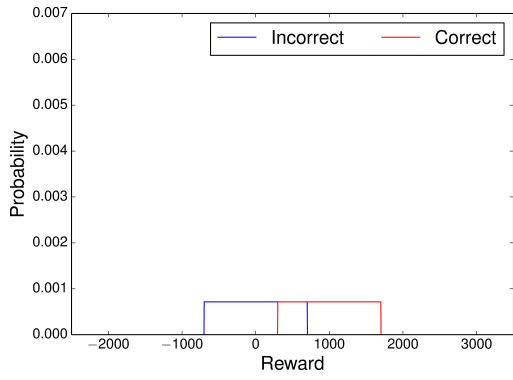
(d) $(\lambda, x) = (1, -100)$



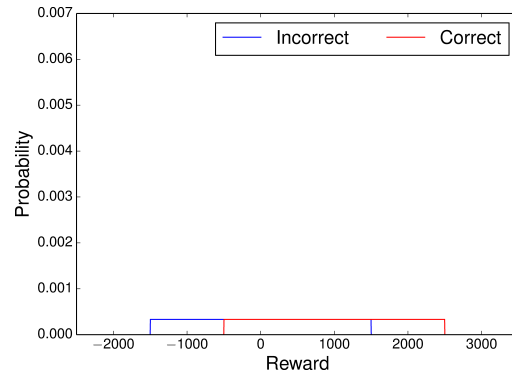
(e) $(\lambda, x) = (1, -500)$

図 8.1 指数分布

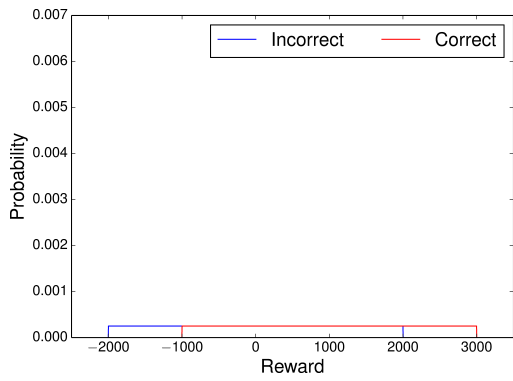
可能かを調べるために、正規分布 ($\sigma_R = 200$), 指数分布 ($(\lambda, x) = (1, 200)$) および一様分布 ($w = \pm 900$) が Iteration ごとにランダムに変わる場合についても適用する。ただし、正規分布 ($\sigma_R = 200$) および一様分布 ($w = \pm 900$) は平均が 0 の分布であるが、指数分布 ($(\lambda, x) = (1, 200)$) は平均が 200 の分布であるため、指数分布の場合は 200 だけスライドし平均を 0 に揃える。



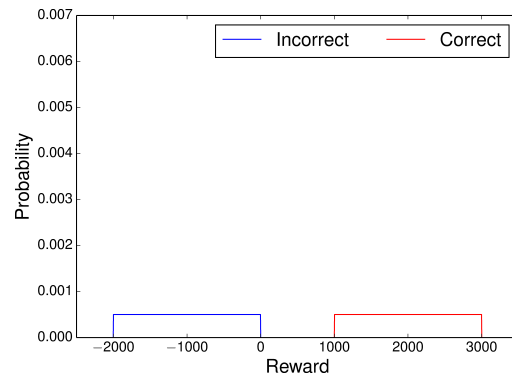
(a) $w = \pm 700$



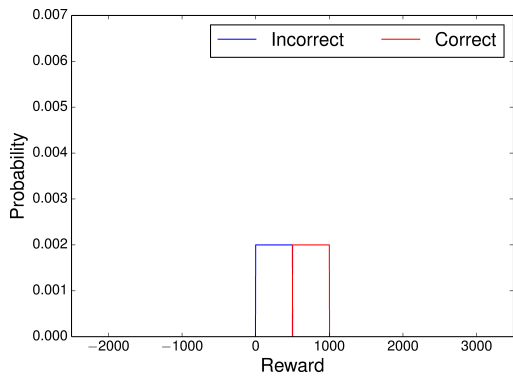
(b) $w = \pm 1500$



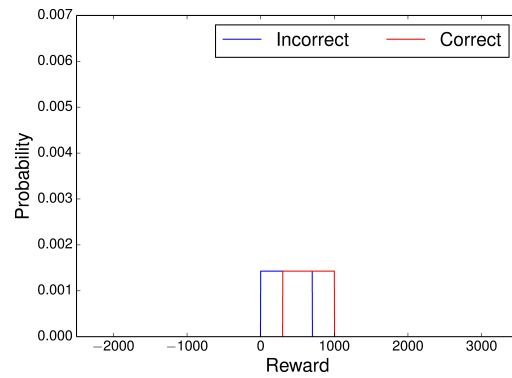
(c) $w = \pm 2000$



(d) $w = -2000$



(e) $w = 500$

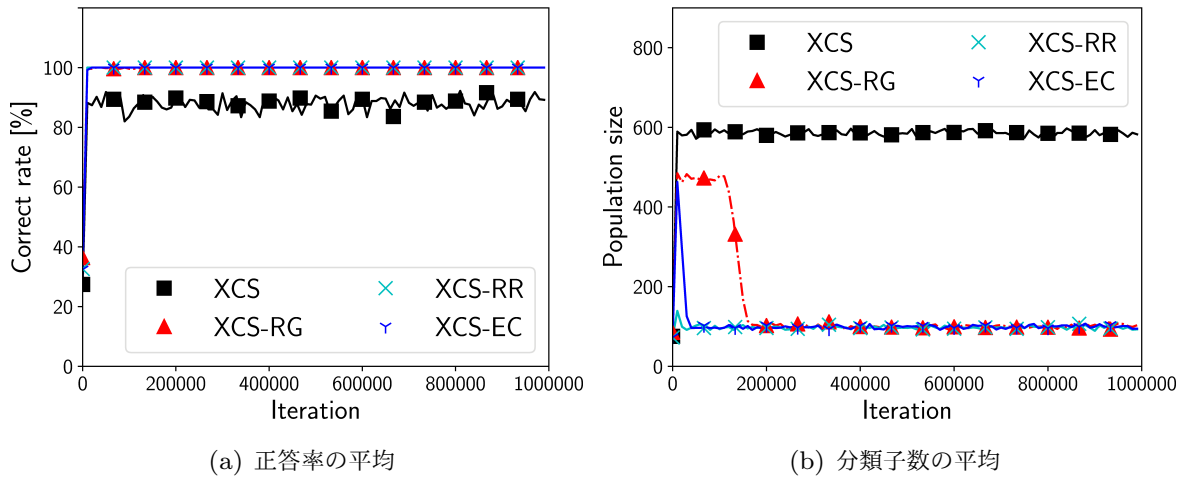


(f) $w = 700$

図 8.2 一様分布

8.1.2 実験設定

XCS および提案手法で用いるパラメータは第 5 章から第 7 章の実験で用いた値と同様の $\theta_{mna} = 2$, $P_{\#} = 0.35$, $p_I = 0.01$, $\epsilon_I = 0.01$, $F_I = 0.01$, $p_{explr} = 1.0$, $\beta = 0.2$, $\epsilon_0 = 10$, $\alpha = 0.1$, $\nu = 10$, $\theta_{sub} = 20$, $\theta_{GA} = 25$, $\chi = 0.8$, $\mu = 0.04$, $N = 800$, $\theta_{del} = 20$,



(a) 正答率の平均

(b) 分類子数の平均

図 8.3 $(\lambda, x) = (2, 100)$

$\delta = 0.1$, $\theta_{RR} = 10$, $\theta_{EC} = 2$, $\theta_{LL} = 25$ である。また、評価基準は第 4.3 節で説明した正答率の平均、分類子数の平均および最適解集合 $[O]$ を獲得した試行数および最大 F の分類子数の平均である。ただし、試行数は 5 である。

8.1.3 実験結果

指数分布 $(\lambda, x) = (2, 100), (1, 300), (1, 400), (1, -100), (1, -500)$ の場合の正答率の平均と分類子数の平均および $[O]$ の獲得試行数と最大 F の分類子数の平均を図 8.3 から 8.7 および表 8.1 から 8.5 に、一様分布 $w = \pm 700, \pm 1500, \pm 2000, -2000, 500, 700$, $\sup\{w\} = 700$ (ケース V) の場合の正答率の平均と分類子数の平均および $[O]$ の獲得試行数と最大 F の分類子数の平均を図 8.8 から 8.14 および表 8.6 から 8.12 に、正規分布 ($\sigma_R = 200$), 指数分布 ($(\lambda, x) = (1, 200)$) および一様分布 ($w = \pm 900$) が Iteration ごとに変わる場合の正答率の平均と分類子数の平均および $[O]$ の獲得試行数と最大 F の分類子数の平均を図 8.15 および表 8.13 に示す。図の横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。

判定の信憑性における不確実性が指数分布の場合

XCS は全ての場合において適切に学習できず、分類子の一般化ができていない。XCS-RR は $(\lambda, x) = (2, 100), (1, -100), (1, -500)$ の場合においてのみ適切に学習し分類子を一般化することで、 $[O]$ を獲得した。XCS-RG は $(\lambda, x) = (1, 400)$ の場合を除いて、適切に学習し分類子を一般化することで、 $[O]$ を獲得した。XCS-EC は $(\lambda, x) = (2, 100), (1, -100)$ の全ての試行と $(\lambda, x) = (1, -500)$ の一部の試行においてのみ適切に学習し分類子を一般化することで、 $[O]$ を獲得した。

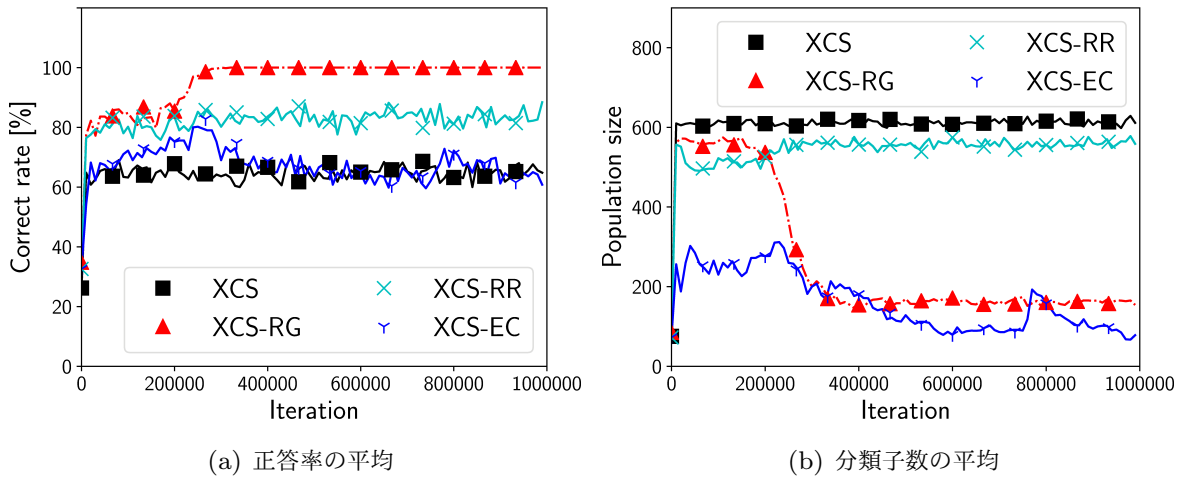


図 8.4 $(\lambda, x) = (1, 300)$

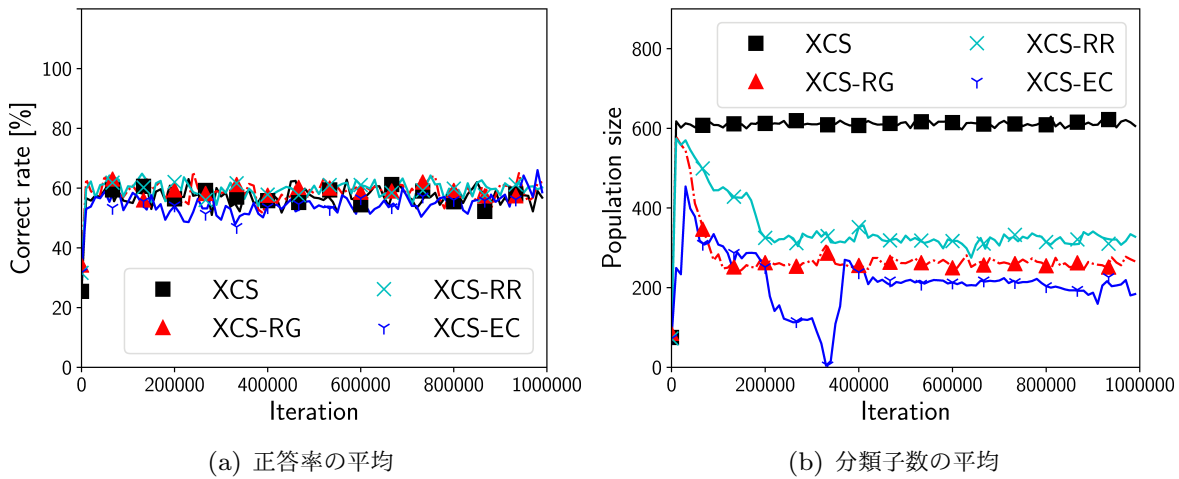


図 8.5 $(\lambda, x) = (1, 400)$

判定の信憑性における不確実性が一様分布の場合

XCS は全ての場合において適切に学習できず、分類子の一般化ができていない。XCS-RR は $w = \pm 700, 500, \sup\{w\} = 700$ (ケース V) の場合においてのみ適切に学習し分類子を一般化することで、[O] を獲得した。XCS-RG は $w = \pm 700, \pm 1500, 500, 700$ の場合においてのみ適切に学習し分類子を一般化することで、[O] を獲得した。XCS-EC は $w = \pm 2000, -2000$ の一部の試行を除き、適切に学習し分類子を一般化することで、[O] を獲得した。

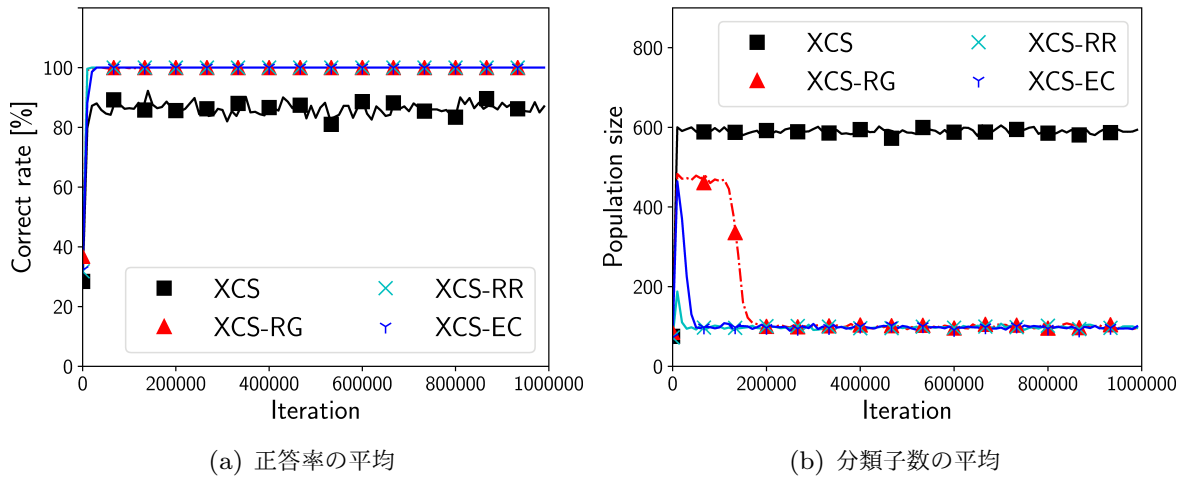


図 8.6 $(\lambda, x) = (1, -100)$

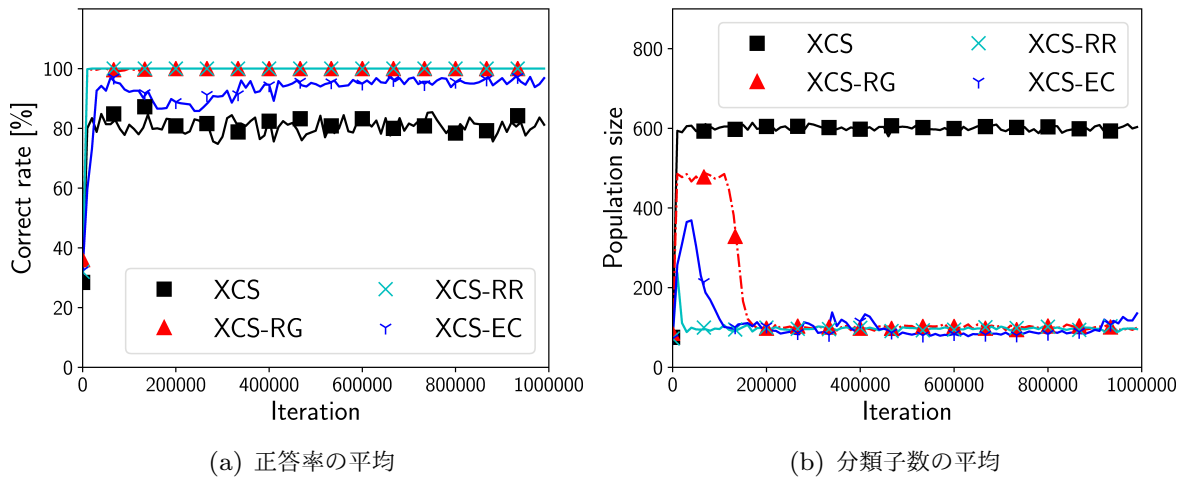


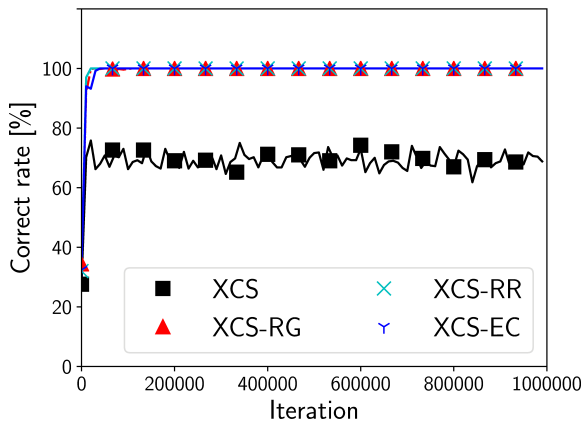
図 8.7 $(\lambda, x) = (1, -500)$

判定の信憑性における不確実性の種類が変化する場合

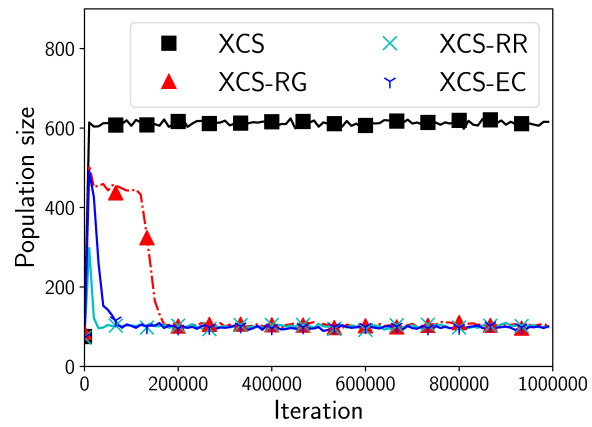
XCS はこれまでの結果と同様に適切に学習できず、分類子の一般化ができていないが、3つの提案手法は適切に学習し分類子を一般化することで、[O] を獲得した。3つの提案手法は獲得する報酬分布の形状の定常性を必要としない。

8.2 考察

獲得報酬の分布形状の違いによる提案手法の対応の得手不得手について述べる。図 8.16 に獲得報酬の分布が指数分布および一様分布の場合における各提案手法の正確性判定への

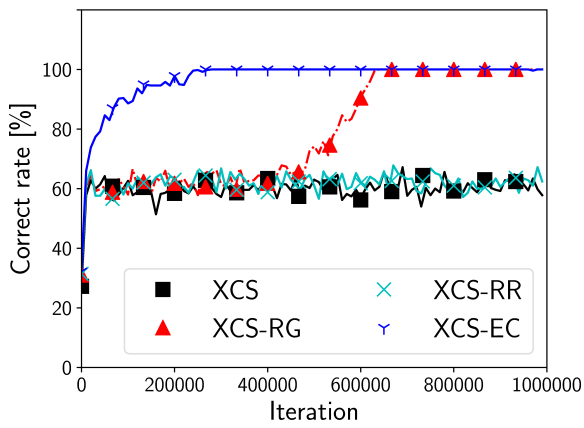


(a) 正答率の平均

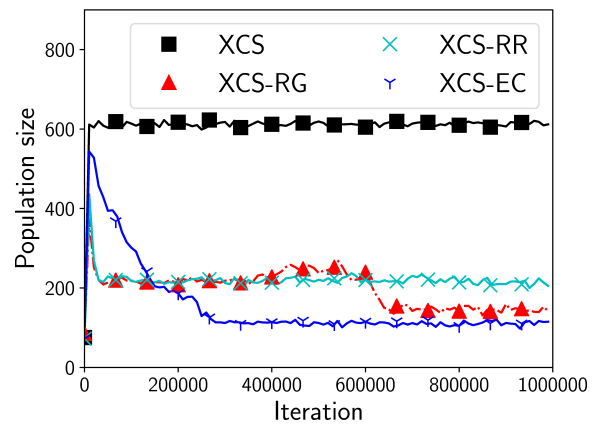


(b) 分類子数の平均

図 8.8 $w = \pm 700$

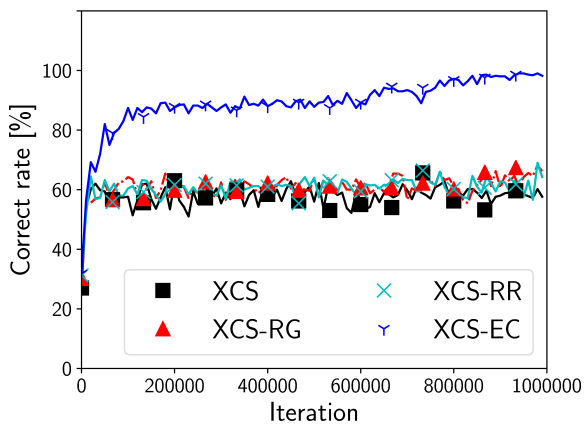


(a) 正答率の平均

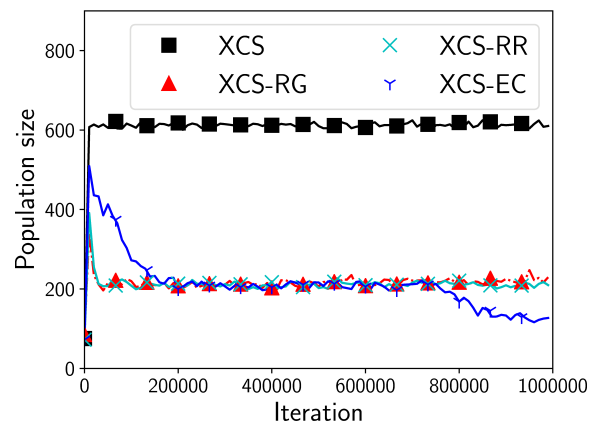


(b) 分類子数の平均

図 8.9 $w = \pm 1500$

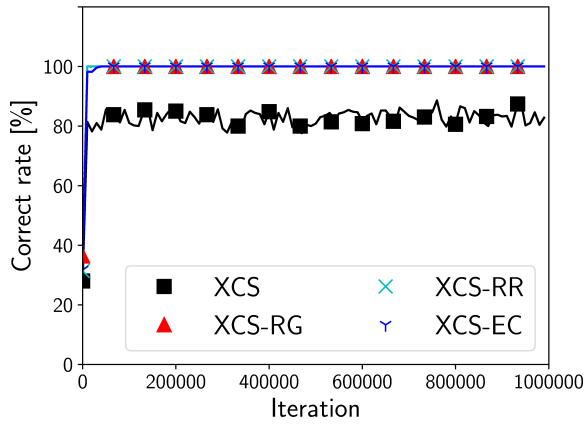


(a) 正答率の平均

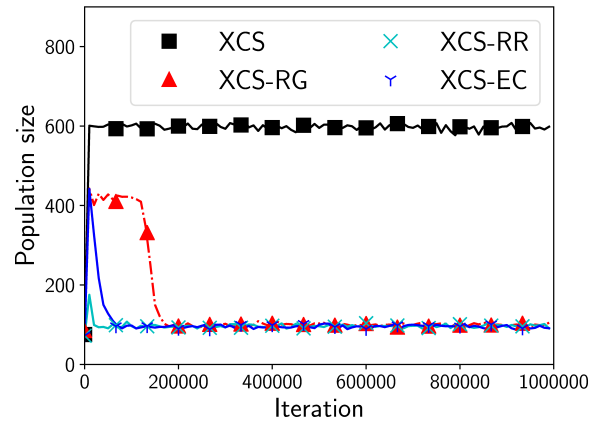


(b) 分類子数の平均

図 8.10 $w = \pm 2000$

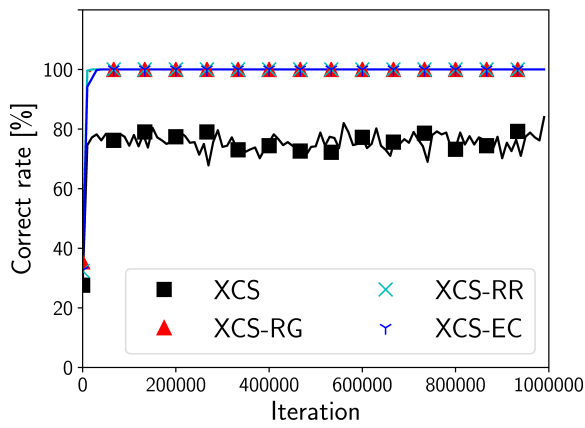


(a) 正答率の平均

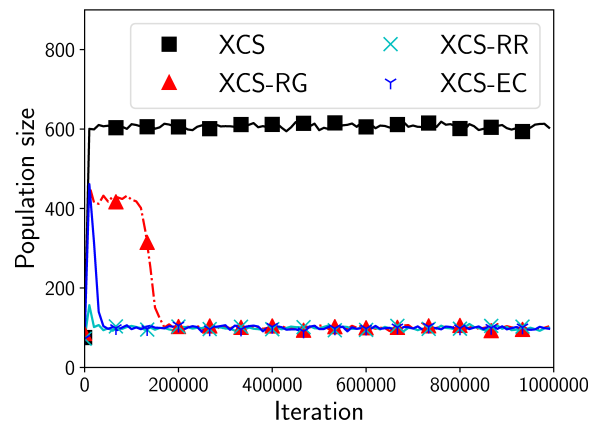


(b) 分類子数の平均

図 8.11 $w = -2000$

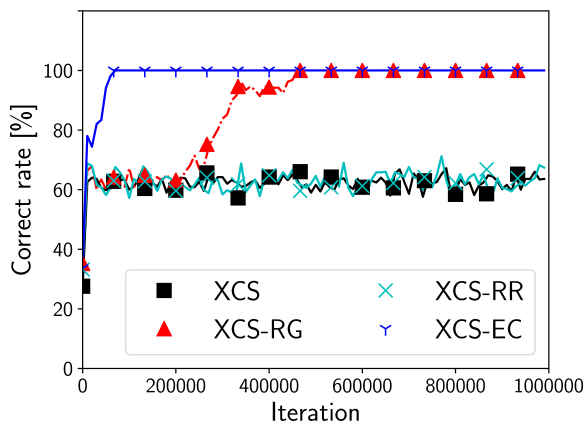


(a) 正答率の平均

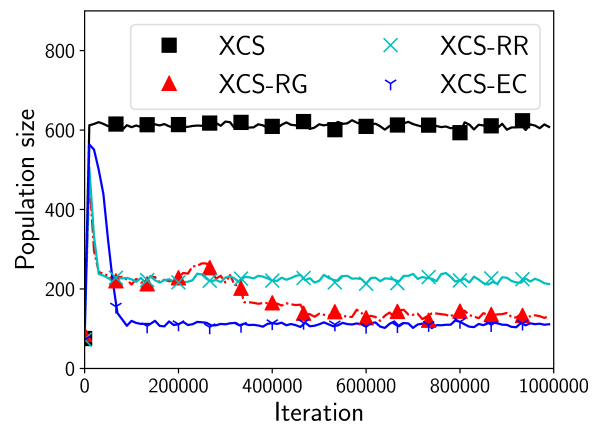


(b) 分類子数の平均

図 8.12 $w = 500$



(a) 正答率の平均



(b) 分類子数の平均

図 8.13 $w = 700$

表 8.1 $(\lambda, x) = (2, 100)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	59.6
XCS-RR	5 (100%)	5 (100%)	32
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

表 8.2 $(\lambda, x) = (1, 300)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	56.6
XCS-RR	0 (0%)	0 (0%)	48.6
XCS-RG	5 (100%)	4 (80%)	31.6
XCS-EC	0 (0%)	0 (0%)	7.6

表 8.3 $(\lambda, x) = (1, 400)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.4
XCS-RR	0 (0%)	0 (0%)	8.0
XCS-RG	0 (0%)	0 (0%)	3.0
XCS-EC	0 (0%)	0 (0%)	14.8

表 8.4 $(\lambda, x) = (1, -100)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	64.8
XCS-RR	5 (100%)	5 (100%)	32
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

影響を示す。横軸は獲得報酬の値を表し、縦軸はその獲得報酬の確率密度を表す。

XCS-RR は分類子が獲得する報酬の最大値あるいは最小値の値により正解を導く正確な分類子と不正解を導く正確な分類子を識別する。つまり、不正解時に非常に低い値の報酬を獲得しても、正解時に非常に高い値の報酬を獲得しても分類子の正確性に影響を及ぼさない。そのため、図 8.16(a) の指数分布の様に重複がない場合には適切に学習し、[O] を獲得できる。一方で図 8.16(b) の一様分布の様に重複の割合が高い場合には正確な分類

表 8.5 $(\lambda, x) = (1, -500)$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	62.4
XCS-RR	5 (100%)	5 (100%)	32
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	2 (40%)	2 (40%)	25.2

表 8.6 $w = \pm 700$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	54.4
XCS-RR	5 (100%)	5 (100%)	32
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

表 8.7 $w = \pm 1500$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	48.8
XCS-RR	0 (0%)	0 (0%)	2.2
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	4 (80%)	32

表 8.8 $w = \pm 2000$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	46.8
XCS-RR	0 (0%)	0 (0%)	2.0
XCS-RG	0 (0%)	0 (0%)	3.2
XCS-EC	3 (60%)	3 (60%)	31.6

子を識別できないため、正しく学習できない。

XCS-RG は Recording table に状態行動対ごとの獲得した報酬の平均を記録し、その値が同一の状態行動対の組み合わせからなる分類子を正確とみなす。そのため、図 8.16(a) の指数分布の場合でも、図 8.16(b) の一様分布の場合でも、正解の場合に獲得される報酬の平均と不正解の場合に獲得される報酬の平均が明確に分かれていれば適切に学習し、[O] を獲得できる。ただし、 $\sup\{w\} = 700$ (ケース V) (図 8.14 および表 8.12) の場合ではランダムの影響で状態行動対ごとに平均が異なるため、正解の場合に獲得される報酬

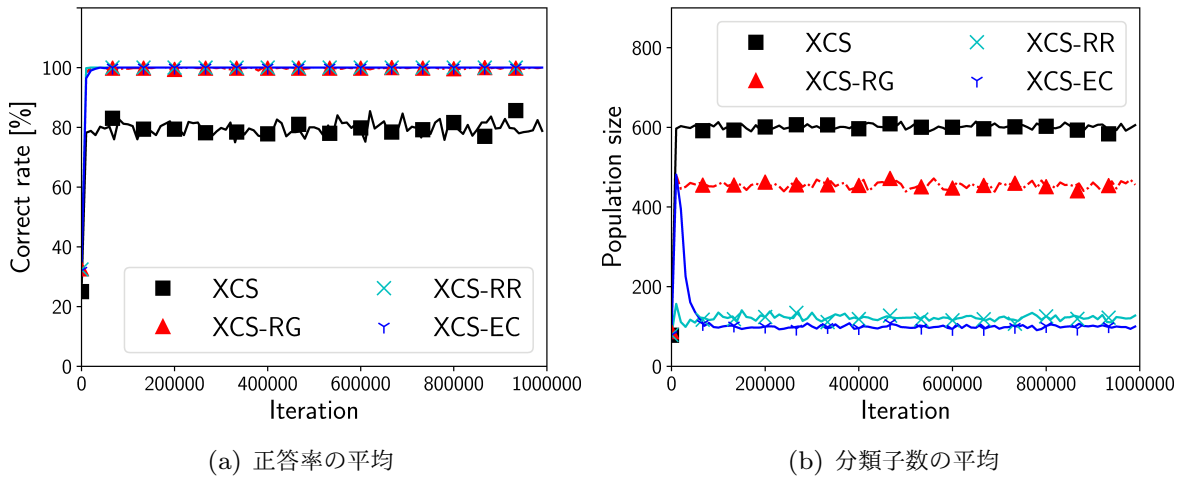


図 8.14 $\sup\{w\} = 700$ (ケース V)

表 8.9 $w = -2000$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	46.8
XCS-RR	0 (0%)	0 (0%)	2.0
XCS-RG	0 (0%)	0 (0%)	3.2
XCS-EC	3 (60%)	3 (60%)	31.6

表 8.10 $w = 500$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	52.8
XCS-RR	5 (100%)	5 (100%)	32
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

の平均と不正解の場合に獲得される報酬の平均が明確に分かれず、正解時の平均の分布と不正解時の平均の分布を正しく見積もれなかったためである。

XCS-EC は分類子が獲得した報酬の平均の加重平均によってその時の入出力における正誤評価を推測する。そのため、図 8.16(b) の一様分布の様に正解時に獲得する報酬 R および不正解時に獲得する報酬 0 から遠い報酬を獲得しない場合には、正しくその時の入出力における正誤評価を推測し、適切に学習し、[O] を獲得できる。一方で図 8.16(a) の指数分布の様に正解時に獲得する報酬 R および不正解時に獲得する報酬 0 から遠い報酬を獲得する場合には、その時の入出力における正誤評価の推測が誤る場合があるため、正し

表 8.11 $w = 700$ における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	56.6
XCS-RR	0 (0%)	0 (0%)	2.6
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	4 (80%)	32

表 8.12 $\sup\{w\} = 700$ (ケース V) における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	58.6
XCS-RR	5 (100%)	0 (0%)	32.2
XCS-RG	0 (0%)	0 (0%)	58.2
XCS-EC	5 (100%)	5 (100%)	32

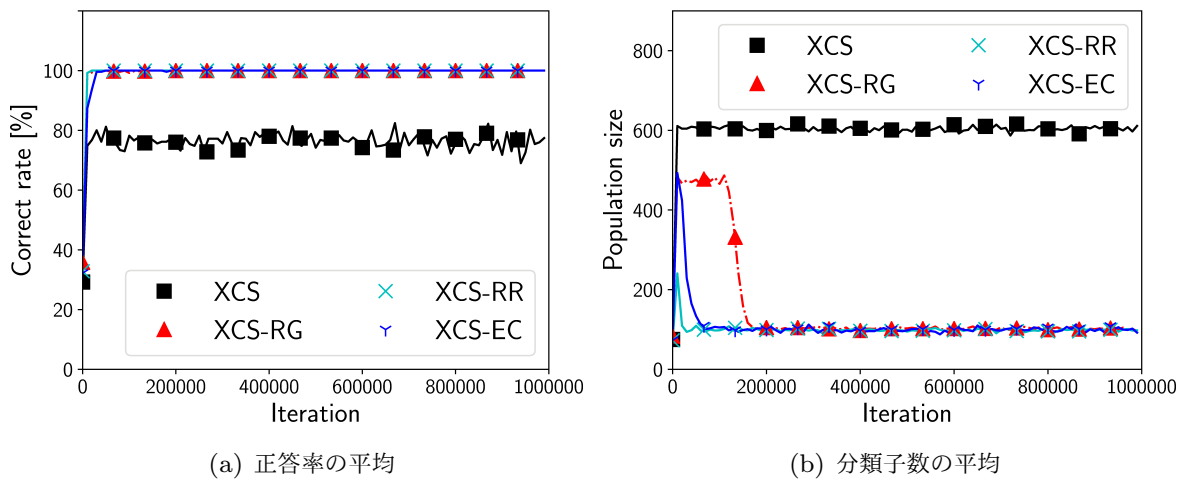


図 8.15 正規分布, 指数分布および一様分布が Iteration ごとに変わる場合

く学習できない.

表 8.13 正規分布, 指数分布および一様分布が Iteration ごとに変わる場合における最適解の獲得

手法	[O] の獲得試行数	上位 32 個での [O] の獲得試行数	最大 F の分類子数の平均
XCS	0(0%)	0 (0%)	50.4
XCS-RR	5 (100%)	5 (100%)	32
XCS-RG	5 (100%)	5 (100%)	32
XCS-EC	5 (100%)	5 (100%)	32

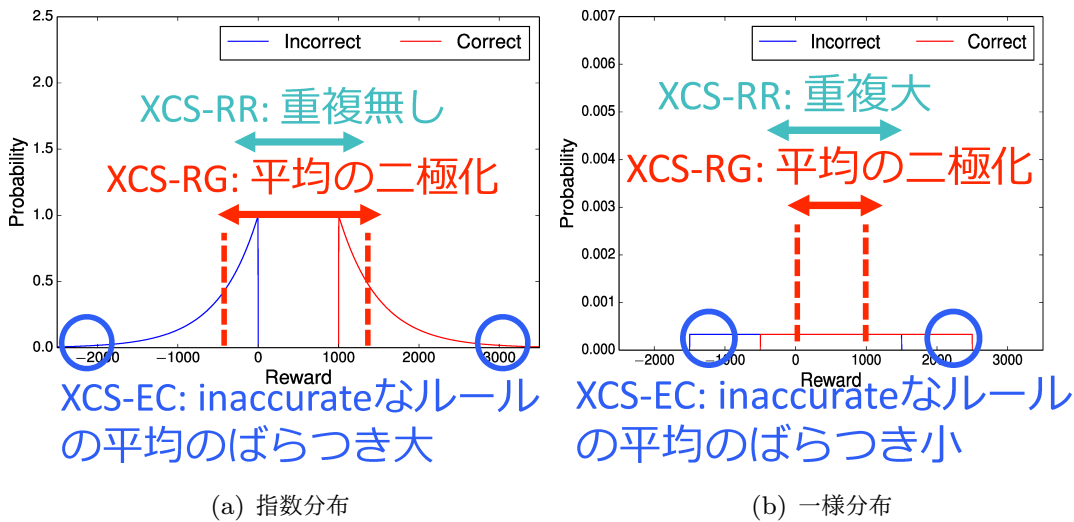


図 8.16 分布形状と提案手法における正確性判定への影響

第 9 章

不確実性に対応可能な 学習分類子システムの設計論

9.1 不確実性を伴うデータから知識を獲得するメカニズムの設計

データセット全体を簡潔なルールで表す知識を獲得するためには評価が同一な複数のデータをまとめ、その共通点を見いだす必要がある。正しくルールが生成できたかを確認するためには、A) 生成されたルールが常に単一の評価を獲得することを確認するか、B) ルールが照合するデータ全てが同一の評価を獲得することを確認する必要がある。どちらのアプローチを採用したとしても、データに不確実性を伴うと評価にばらつきが生じるので“同一”の評価では正しくルールが生成されたかを評価できない。本研究では XCS を題材に A) 生成されたルールが常に単一の評価を獲得することを確認するアプローチにおいて不確実性に対応を図った XCS-ER, XCS-RR および XCS-EC を提案し、B) ルールが照合するデータ全てが同一の評価を獲得することを確認するアプローチにおいて不確実性に対応を図った XCS-RG を提案した。これらの手法を基に (i) 入出力における不確実性あるいは (ii) 判定の信憑性における不確実性を伴うデータから知識を獲得するメカニズムの設計について述べる。

9.1.1 入出力における不確実性を伴うデータに対応するメカニズムの設計

入出力における不確実性を伴うと他の入力あるいは他の出力に由来する評価が混在し、複数種類の離散的な評価を獲得する。獲得した複数の評価の内、対象となる入出力対の正しい評価を採用し、他の入力あるいは他の出力に由来する評価を除く必要がある。XCS はルールベースであり逐次的に更新するオンライン学習の手法であるため、学習途中の段階において獲得している分類子の情報を用いることができる。具体的には、獲得している

分類子は照合する入力と出力に対する評価の期待値を持っているため、ある入力において照合する分類子を用いることにより出力毎の評価の期待値を求め、その大小関係から出力の（正誤）評価を推測できる。本研究においては XCS-ER および XCS-EC にあたる。機械学習には学習データを一度に適用するバッチ学習の手法も存在するため、学習途中の情報に依らない方法として学習データ全体における傾向を把握するものが挙げられる。具体的には、全ての入出力対において獲得報酬の平均を記録し、同一の入力における出力の違いによる獲得報酬の平均の大小関係から出力の（正誤）評価を推測する。本研究においては XCS-RG にあたる。

9.1.2 判定の信憑性における不確実性を伴うデータに対応するメカニズムの設計

判定の信憑性における不確実性を伴うと評価が連続的な数値となる。ひとつひとつの評価の数値の信頼性は低いため、その評価の分布として正解の評価なのか、不正解の評価なのか、正解と不正解が混在した評価なのか識別する必要がある。評価の分布によって正誤を把握するためには平均を用いる方法や最大値や最小値を用いる方法が考えられる。平均を用いる方法は上記の入出力における不確実性を伴う場合と同様である。獲得評価の最大値や最小値を用いる方法では、不確実性が大きくない時あるいは正誤によって不確実性の強度に違いが無い時に、正解の場合の獲得評価の最大値や最小値はそれぞれ不正解の場合の最大値や最小値よりも大きくなることを利用し、獲得した評価の正誤を判定する。本研究においては XCS-RR にあたる。

9.1.3 アプローチによる不確実性への対応の差異

不確実性への対応メカニズムをまとめると表 9.1 となる。平均値（期待値）は入出力における不確実性および判定の信憑性における不確実性のどちらを伴っても正解時と不正解時に差が生じることが期待できるため、両方の不確実性に対応可能である。最大値や最小値は入出力における不確実性のみを伴う場合には正解時と不正解時に同一の値となるため対応できないが、判定の信憑性における不確実性を伴う場合には正解時と不正解時に差が生じることが期待できるため対応可能である。○は対応を示し、×は未対応を示す。

提案手法の使い分け

KDD に用いられる問題が持つ性質として

- 問題の種類
 - － 分類問題
 - － 回帰問題

表 9.1 メカニズムと対応する不確実性

メカニズム	(i) 入出力における不確実性	(ii) 判定の信憑性における不確実性
平均値（期待値）の比較	○	○
最大値や最小値の比較	×	○

- 特徴量
 - － 離散的（少数の種類）
 - － 連続的（様々な値）
- データ量（網羅性）
 - － 全入力（と出力の組み合わせ）にデータが十分にある
 - － データが無い入力（と出力の組み合わせ）がある
- 伴う不確実性
 - － 無し
 - － 入出力における不確実性のみ
 - － 判定の信憑性における不確実性のみ
 - － 入出力における不確実性および判定の信憑性における不確実性の複合

が挙げられる。また、第 5 章から第 8 章までの実験の結果から以下のような提案手法の特徴がある。XCS-ER は入出力における不確実性のみが伴う場合に適用範囲が最も広く、最も収束速度が速い。XCS-EC 及び XCS-RG はどのような不確実性を伴っても対応が可能だが、報酬分布が指数分布の様に裾が厚い分布の場合には XCS-RG が、状態行動対ごとの平均の値のばらつきが大きい場合には XCS-EC が適している。これらのことから提案手法の使い分けは以下の通りとなる。

問題の種類が

- 分類問題の場合、かつ
データ量（網羅性）が
 - － 全ての入力と出力の組み合わせにデータがある場合: XCS-RG
 - － データが無い入力と出力の組み合わせがある場合、かつ
伴う不確実性が
 - * 入出力における不確実性のみの場合: XCS-ER
 - * 判定の信憑性における不確実性のみの場合: XCS-RR
 - * 双方の不確実性を複合する場合あるいはオンライン処理などで不明な場合: XCS-EC
- 回帰問題の場合、かつ

伴う不確実性が

- 判定の信憑性における不確実性のみで値が2極化している場合: XCS-RR
- 入出力における不確実性も含む場合あるいはオンライン処理などで不明な場合: XCS-EC

ここで上記の使い分けを基に UCI Machine Learning Repository[90] に登録されているデータセットの中から代表的なものに対してグループ分けすると以下の様になる。

- XCS-RG が適したデータ
分類問題かつ全ての入出力対のデータが含まれる場合 (例: Car Evaluation)
- XCS-ER が適したデータ
分類問題かつ一部の入出力対のデータが含まれない場合 (例: Iris, Bank Marketing, Wine, Breast Cancer Wisconsin (Diagnostic), Heart disease, Abalone, Adult, Human Activity Recognition Using Smartphones, Wine Quality)
- XCS-EC が適したデータ
回帰問題かつ値が釣り鐘型の場合 (例: Forest Fires, Student Performance, Auto MPG, Computer Hardware, SGEMM GPU kernel performance, Solar Flare)

ただし, XCS-RR は回帰問題かつ値が2極化していることを必要とするが, 回帰問題において値が正規分布や指数分布の様に釣り鐘型となっており適用できる例は代表的なものの中から見つけることはできなかった。

9.2 提案手法の解析的適用範囲

解析的に提案メカニズムの適用範囲を示すために, 図 9.1 の様に獲得する報酬の分布をモデル化する。正解時および不正解時に獲得する報酬 X の真 (母集団) の平均 μ_C, μ_I の差が R であるとする。また, 正解時および不正解時に獲得する報酬の真 (母集団) の分散を σ^2 とする。サンプル数 n の時の標本平均 \bar{X} の期待値 ($E[\bar{X}]$) および分散 ($V[\bar{X}]$) は以下の様になる。

$$\begin{aligned} E[\bar{X}] &= E\left[\frac{1}{n}(X_1 + X_2 + \dots + X_n)\right] \\ &= \frac{1}{n}E[X_1 + X_2 + \dots + X_n] \\ &= E[X](= \mu) \end{aligned} \tag{9.1}$$

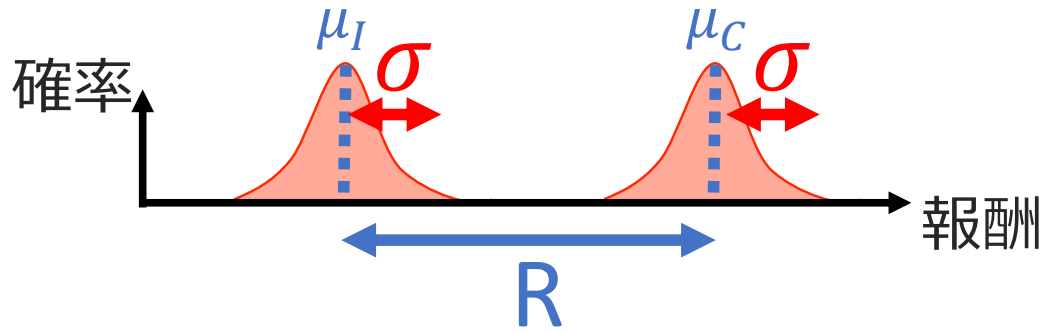


図 9.1 報酬分布のモデル化

$$\begin{aligned}
 V[\bar{X}] &= V\left[\frac{1}{n}(X_1 + X_2 + \dots + X_n)\right] \\
 &= \frac{1}{n^2}V[X_1 + X_2 + \dots + X_n] \\
 &= \frac{V[X]}{n} (= \frac{\sigma^2}{n})
 \end{aligned} \tag{9.2}$$

信頼区間を $\alpha\sigma$ (ただし, α は正の数とする.) とすると, 正解時および不正解時に獲得する報酬の平均は以下の範囲に収まることになる.

- 正解時: $\mu_C \pm \alpha\sigma/\sqrt{n}$
- 不正解時: $\mu_I \pm \alpha\sigma/\sqrt{n}$

これを用いて最低限満たさなければならない条件を示すことで, 解析的に提案メカニズムの適用範囲を示す. ただし, 計算の簡単のためサンプル数 n は対象となる値の内の最小の値として, 全て同一の値として扱う.

9.2.1 XCS-ER

入出力における不確実性を伴う場合

XCS-ER は $[M]$ 内の分類子を出力毎に分けて評価回数を重みとした獲得報酬の平均の加重平均の大小関係で出力の正誤を推測する. 正しく出力の正誤を推測するためには, 正確かつ正解の分類子が獲得した報酬の平均と正確かつ不正解の分類子が獲得した報酬の平均の大小関係が正しい必要がある. これを式に表すと

$$2\alpha \frac{\sigma}{\sqrt{n}} < R \tag{9.3}$$

となる. ここで, 入出力における不確実性を伴う場合における母集団の標準偏差 σ_p を求める. p は不確実性による正誤が変化する確率を表す. p における報酬の平均の距離 R_p は, $p = 0$ の場合の正誤の報酬の平均の距離を R とすると,

$$R_p = \mu_c - \mu_I = (1 - 2p)R \tag{9.4}$$

表 9.2 入出力における不確実性に対する XCS-ER の適用範囲（正誤が変化する確率 p ）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	0.35	0.23	0.16	0.11	0.08	0.06
	50	0.43	0.36	0.30	0.25	0.21	0.18
	100	0.45	0.40	0.36	0.31	0.28	0.24
	150	0.46	0.42	0.38	0.34	0.31	0.28
	200	0.46	0.43	0.40	0.36	0.33	0.30

表 9.3 入出力における不確実性を伴う場合の XCS-ER の最適解の獲得試行数

	$P_O = 0.3$	$P_O = 0.35$
[O] の獲得試行数	5 (100%)	4 (80%)
上位 32 個での [O] の獲得試行数	5 (100%)	3 (60%)
最大 F の分類子数の平均	32	31.8

となり，ここから σ_p は

$$\sigma_p = \sqrt{p(1-p)}R \quad (9.5)$$

となる．これを式 (9.3) に当てはめると，

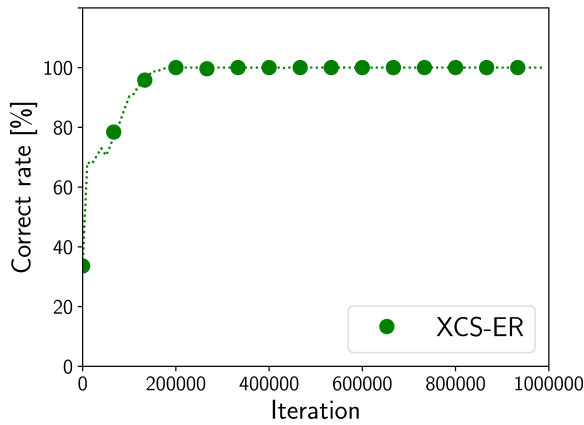
$$\begin{aligned} \frac{2\alpha\sqrt{p(1-p)}R}{\sqrt{n}} &< (1-2p)R \\ 4\alpha^2p(1-p) &< n(4p^2 - 4p + 1) \\ p^2 - p + \frac{n}{4n + 4\alpha^2} &> 0 \end{aligned} \quad (9.6)$$

これを解くと $p < 0.5$ の制約から

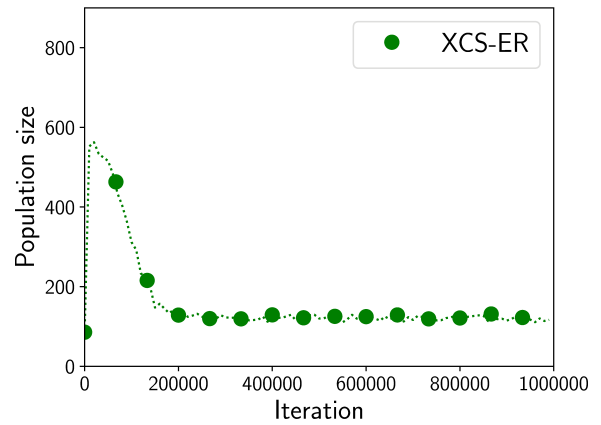
$$p < \frac{1}{2} \left(1 - \sqrt{1 - \frac{n}{n + \alpha^2}} \right) \quad (9.7)$$

が求まる．表 9.2 に $(1 - \sqrt{1 - n/(n + \alpha^2)})/2$ の値を示す．行は分類子の評価回数 n を列は信頼区間 α を示す．

図 9.11 および 9.12 に $P_O = 0.3, 0.35$ の場合の正答率の平均と分類子数の平均を示す．横軸は Iteration を表し，縦軸は正答率の平均あるいは分類子数の平均を表す．表 9.3 に [O] を獲得した試行数および最大 F の分類子数の平均を示す． $P_O = 0.3$ の場合では正答率の平均が 100% に達し，全ての試行で [O] を獲得する． $P_O = 0.35$ の場合では正答率の平均が 100% よりも小さく，[O] を獲得する試行が少ない．この結果から，表 9.2 の太

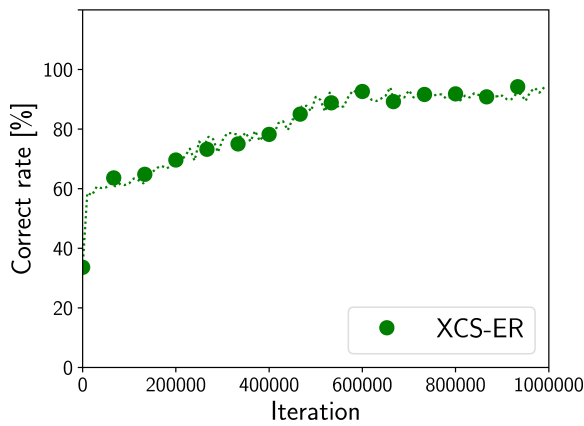


(a) 正答率の平均

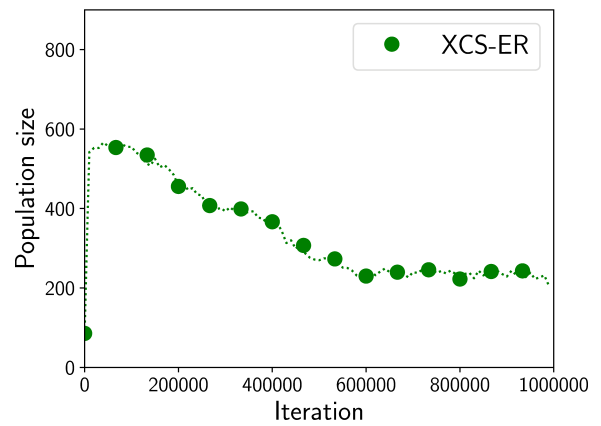


(b) 分類子数の平均

図 9.2 $P_O = 0.3$



(a) 正答率の平均



(b) 分類子数の平均

図 9.3 $P_O = 0.35$

字の部分に示す様に、信頼区間 $4\sigma \sim 5\sigma$ 程度かつ分類子の評価回数が 100 程度の値と合致する。

中心極限定理より分類子の評価回数 n が多くなればなるほど分類子が獲得する報酬の平均は真の平均に近づくため、判定の信憑性における不確実性においては式 (9.7) より、対応可能な不確実性の強度が大きくなる。しかし、XCS-ER はその時点において保持している分類子に基づいて報酬や正誤を推測するため、評価回数 n を増やすには限界がある。

9.2.2 XCS-RR

XCS-RR は正確かつ正解の分類子が獲得した報酬の最小値が正確かつ不正解の分類子が獲得した報酬の平均値を上回ることと正確かつ不正解の分類子が獲得した報酬の最大値

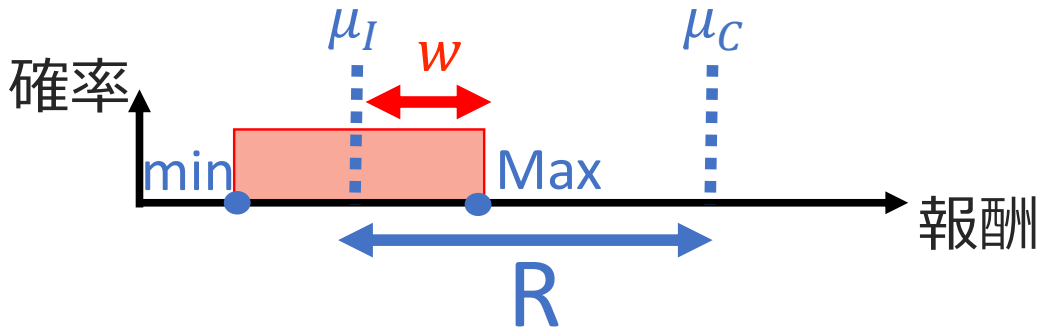


図 9.4 XCS-RR の正確な分類子が満たす条件

が正確かつ正解の分類子が獲得した報酬の平均値を下回ることが必要である。つまり図 9.4 で示すような報酬の分布である必要がある。

判定の信憑性における不確実性（一様分布）を伴う場合

最大値および最小値に制限がある一様分布 ($\pm w$) を扱う。図 9.4 の例では Max が μ_C を下回る様に

$$w + \frac{\alpha\sigma}{\sqrt{n}} < R \quad (9.8)$$

を満たす必要がある。ここで一様分布の標準偏差は $\sigma = w/\sqrt{3}$ なので

$$\begin{aligned} w\left(1 + \frac{\alpha}{\sqrt{3n}}\right) &< R \\ w &< \frac{\sqrt{3n}}{\sqrt{3n} + \alpha} R \end{aligned} \quad (9.9)$$

となる。表 9.4 に $\sqrt{3n}/(\sqrt{3n} + \alpha)$ の値を示す。行は分類子の評価回数 n を列は信頼区間 α を示す。さらに、獲得する報酬の範囲は μ_C, μ_I から $\pm w$ の範囲なので $\alpha\sigma/\sqrt{n} < w$ となり、

$$w + w < R \quad (9.10)$$

つまり、 $w < R/2$ のとき n の値にかかわらず、上記条件を満たす。

図 9.5 および 9.6 に $w = \pm 900, \pm 1000$ の場合の正答率の平均と分類子数の平均を示す。横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。表 9.5 に [O] を獲得した試行数および最大 F の分類子数の平均を示す。 $w = \pm 900$ の場合では正答率の平均が 100% に達し、全ての試行で [O] を獲得する。 $w = \pm 1000$ の場合では正答率の平均が 100% よりも小さく、[O] を獲得する試行が少ない。この結果から、表 9.4 の太字の部分に示す様に、信頼区間 $4\sigma \sim 5\sigma$ 程度かつ分類子の評価回数が 500 程度の値と合致する。

表 9.4 判定の信憑性における不確実性（一様分布）に対する XCS-RR の適用範囲（ R に対する倍率）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	0.85	0.73	0.65	0.58	0.52	0.48
	50	0.92	0.86	0.80	0.75	0.71	0.67
	100	0.95	0.90	0.85	0.81	0.78	0.74
	200	0.96	0.92	0.89	0.86	0.83	0.80
	300	0.97	0.94	0.91	0.88	0.86	0.83
	400	0.97	0.95	0.92	0.90	0.87	0.85
	500	0.97	0.95	0.93	0.91	0.89	0.87

表 9.5 判定の信憑性における不確実性（一様分布）を伴う場合の XCS-RR の最適解の獲得試行数

	$w = \pm 900$	$w = \pm 1000$
[O] の獲得試行数	5 (100%)	1 (20%)
上位 32 個での [O] の獲得試行数	5 (100%)	1 (20%)
最大 F の分類子数の平均	32	8.2

判定の信憑性における不確実性（正規分布）を伴う場合

正規分布の台（値域）は $(-\infty, \infty)$ であるが、平均値から遠い値ほど出現する確率が低くなる。確率が著しく低い値を無視し、台が $(-\infty, \infty)$ となる正規分布であっても、最大値および最小値があるとみなす。正規分布に従う報酬の最大値および最小値が平均から $\pm\alpha\sigma$ の範囲内にあるとすると w を $\alpha\sigma$ に置き換えて

$$\alpha\sigma + \frac{\alpha\sigma}{\sqrt{n}} < R$$

$$\sigma < \frac{R}{\alpha(\alpha + \frac{1}{\sqrt{n}})} \quad (9.11)$$

となる。表 9.6 に $1/(\alpha(\alpha + 1/\sqrt{n}))$ の値を示す。行は分類子の評価回数 n を列は信頼区間 α を示す。

図 9.7 および 9.8 に $\sigma_R = 200, 300$ の場合の正答率の平均と分類子数の平均を示す。横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。表 9.7 に [O] を獲得した試行数および最大 F の分類子数の平均を示す。 $\sigma_R = 200$ の場合では正答率の平均が 100% に近く、全ての試行で [O] を獲得する。 $\sigma_R = 300$ の場合では正答率の平均

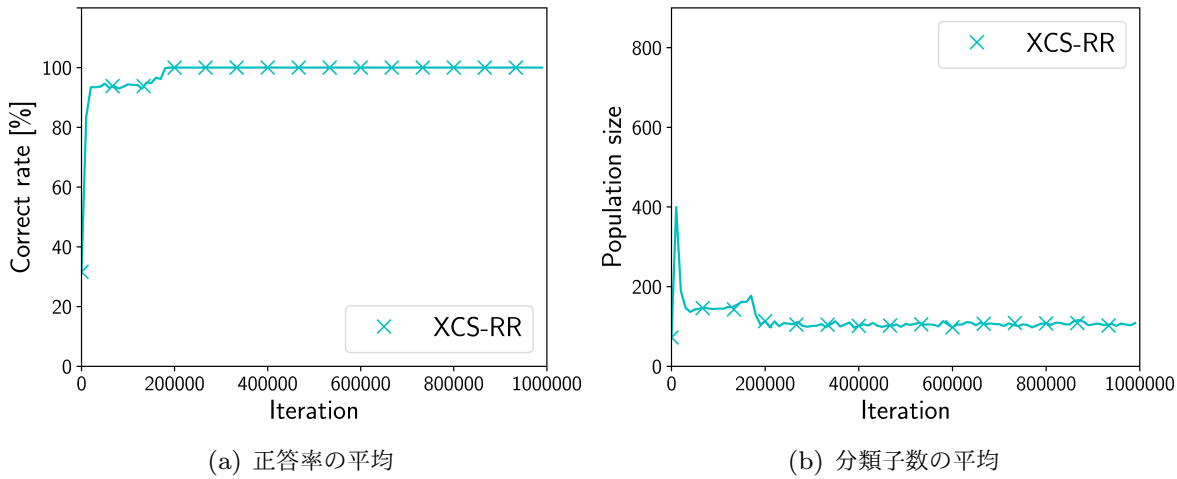


図 9.5 $w = \pm 900$

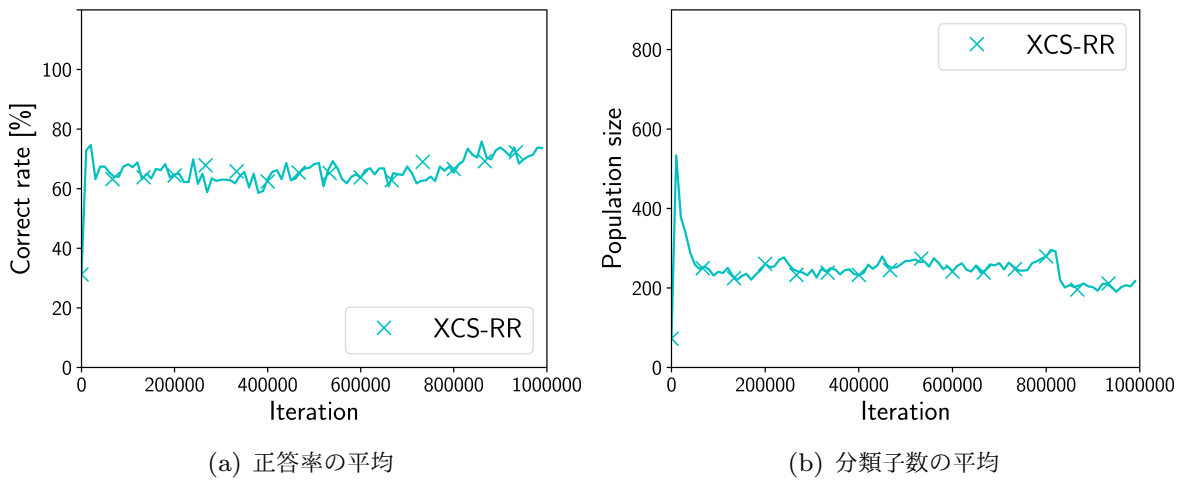


図 9.6 $w = \pm 1000$

が 100% に近いが, $[O]$ を獲得する試行がない. この結果から, 表 9.6 の太字の部分に示す様に, 信頼区間 $4\sigma \sim 5\sigma$ 程度かつ分類子の評価回数が 500 程度の値と合致する.

Multiplexer 問題において最適解よりも $\#$ が多い分類子の中で不正確な度合いが一番低いのは $\{C : 00\#0\#\#\#\#\#\#, A : 1\}$ の様に $1/4$ だけ正解あるいは不正解が混在するものである. この分類子が評価回数 n だけ評価されても上記の条件を満たしてしまう確率は, $3/4$ の評価と混在する $1/4$ の評価の中で平均を下回る ($1/4$ 正解の分類子) あるいは上回る ($1/4$ 不正解の分類子) 場合なので $(7/8)^n$ となる. これを n ごとに表 9.8 に表す. 表 9.8 より n が 40 程度で 3σ に, 100 程度で 5σ に相当する確率になるため, これまでに求めた一様分布および正規分布の場合の n が 500 程度はこの条件を満たす.

分類子が獲得した最大値や最小値から分類子の正確性を判定するため, 分類子の評価回

表 9.6 判定の信憑性における不確実性（正規分布）に対する XCS-RR の適用範囲（ R に対する倍率）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	0.76	0.38	0.25	0.19	0.15	0.13
	50	0.88	0.44	0.29	0.22	0.18	0.15
	100	0.91	0.45	0.30	0.23	0.18	0.15
	200	0.93	0.47	0.31	0.23	0.19	0.16
	300	0.95	0.47	0.32	0.24	0.19	0.16
	400	0.95	0.48	0.32	0.24	0.19	0.16
	500	0.96	0.48	0.32	0.24	0.19	0.16

表 9.7 判定の信憑性における不確実性（正規分布）を伴う場合の XCS-RR の最適解の獲得試行数

	$\sigma_R = 200$	$\sigma_R = 300$
[O] の獲得試行数	5 (100%)	5 (100%)
上位 32 個での [O] の獲得試行数	5 (100%)	0 (0%)
最大 F の分類子数の平均	32	32.8

表 9.8 $(7/8)^n$ の値

評価回数 n	20	40	50	100	200	500
確率	6.9×10^{-2}	4.8×10^{-3}	1.3×10^{-3}	1.6×10^{-6}	2.5×10^{-12}	1.0×10^{-29}

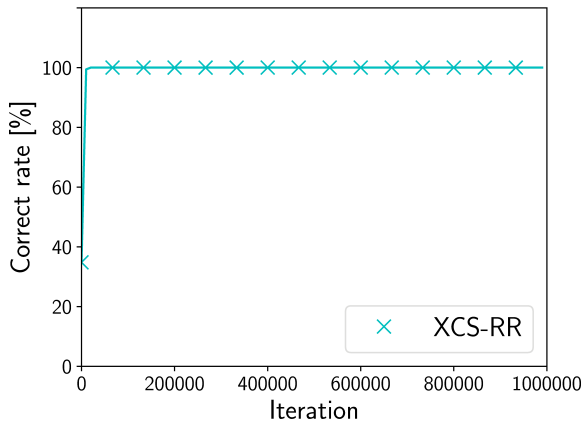
数 n に依らず対応可能な不確実の強度に制限がある。

9.2.3 XCS-EC

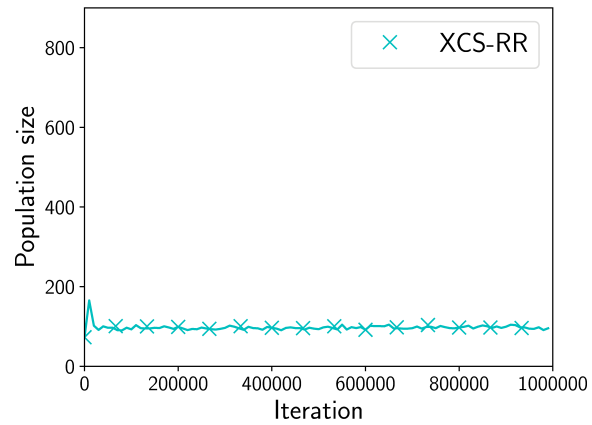
判定の信憑性における不確実性（正規分布）を伴う場合

XCS-EC は $[M]$ 内の分類子を出力毎に分けて評価回数を重みとした獲得報酬の平均の加重平均の大小関係で出力の正誤を推測する。正しく出力の正誤を推測するためには、正確かつ正解の分類子の平均と正確かつ不正解の分類子の平均の大小関係が正しい必要がある。これを式に表すと

$$2\alpha \frac{\sigma}{\sqrt{n}} < R \quad (9.12)$$

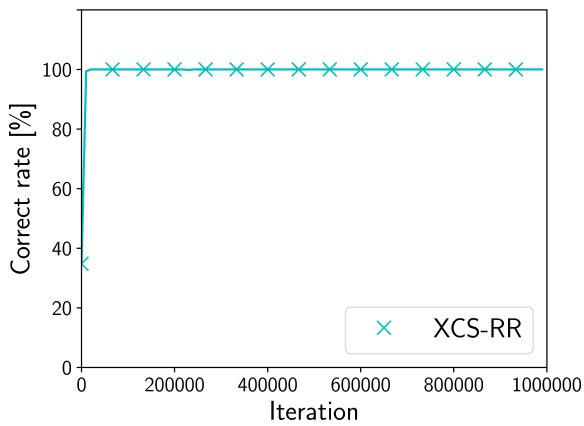


(a) 正答率の平均

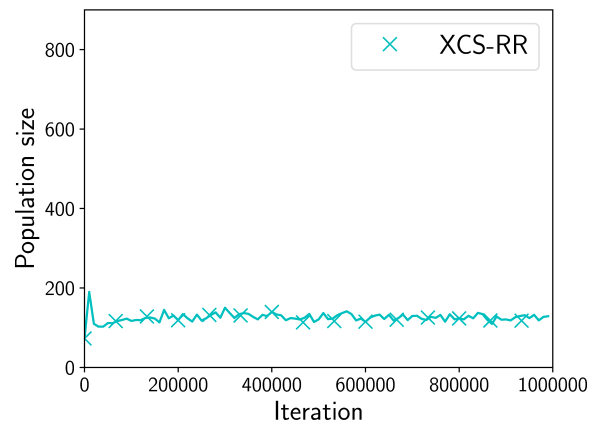


(b) 分類子数の平均

図 9.7 $\sigma_R = 200$



(a) 正答率の平均



(b) 分類子数の平均

図 9.8 $\sigma_R = 300$

となる。これを变形して

$$\sigma < \frac{\sqrt{n}}{2\alpha} R \quad (9.13)$$

となる。表 9.9 に $\sqrt{n}/2\alpha$ の値を示す。行は分類子の評価回数 n を列は信頼区間 α を示す。

図 9.9 および 9.10 に $\sigma_R = 900, 1000$ の場合の正答率の平均と分類子数の平均を示す。横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。表 9.10 に [O] を獲得した試行数および最大 F の分類子数の平均を示す。 $\sigma_R = 900$ の場合では正答率の平均が 100% に近く、ほとんどの試行で [O] を獲得する。 $\sigma_R = 1000$ の場合では正答率の平均が 100% よりも小さく、[O] を獲得する試行が少なく、最大 F の分類子数の平均も大きく低下する。この結果から、表 9.9 の太字の部分に示す様に、信頼区間 $4\sigma \sim 5\sigma$ 程

表 9.9 判定の信憑性における不確実性（正規分布）に対する XCS-EC の適用範囲（ R に対する倍率）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	1.58	0.79	0.53	0.40	0.32	0.26
	50	3.54	1.77	1.18	0.88	0.71	0.59
	100	5.00	2.50	1.67	1.25	1.00	0.83
	150	6.12	3.06	2.04	1.53	1.22	1.02
	200	7.07	3.54	2.36	1.77	1.41	1.18

表 9.10 判定の信憑性における不確実性（正規分布）を伴う場合の XCS-EC の最適解の獲得試行数

	$\sigma_R = 900$	$\sigma_R = 1000$
[O] の獲得試行数	4 (80%)	3 (60%)
上位 32 個での [O] の獲得試行数	4 (80%)	3 (60%)
最大 F の分類子数の平均	30.4	24.4

度かつ分類子の評価回数が 100 程度の値と合致する。

入出力における不確実性を伴う場合

入出力における不確実性を伴う場合には正誤が変化する確率 p により正誤の報酬の平均の距離 R_p が異なる。 $p = 0$ の場合の正誤の報酬の平均の距離を R とすると、

$$R_p = \mu_c - \mu_I = (1 - 2p)R \quad (9.14)$$

$$\sigma_p = \sqrt{p(1-p)}R \quad (9.15)$$

となる。これを式 (9.12) に当てはめると、

$$\begin{aligned} \frac{2\alpha\sqrt{p(1-p)}R}{\sqrt{n}} &< (1 - 2p)R \\ 4\alpha^2 p(1-p) &< n(4p^2 - 4p + 1) \\ p^2 - p + \frac{n}{4n + 4\alpha^2} &> 0 \end{aligned} \quad (9.16)$$

これを解くと $p < 0.5$ の制約から

$$p < \frac{1}{2} \left(1 - \sqrt{1 - \frac{n}{n + \alpha^2}} \right) \quad (9.17)$$

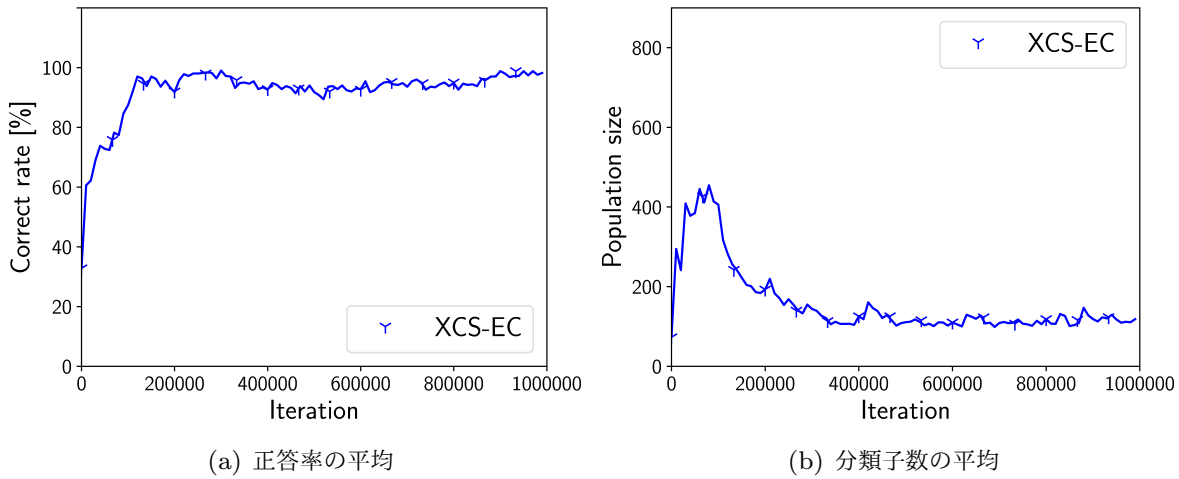


図 9.9 $\sigma_R = 900$

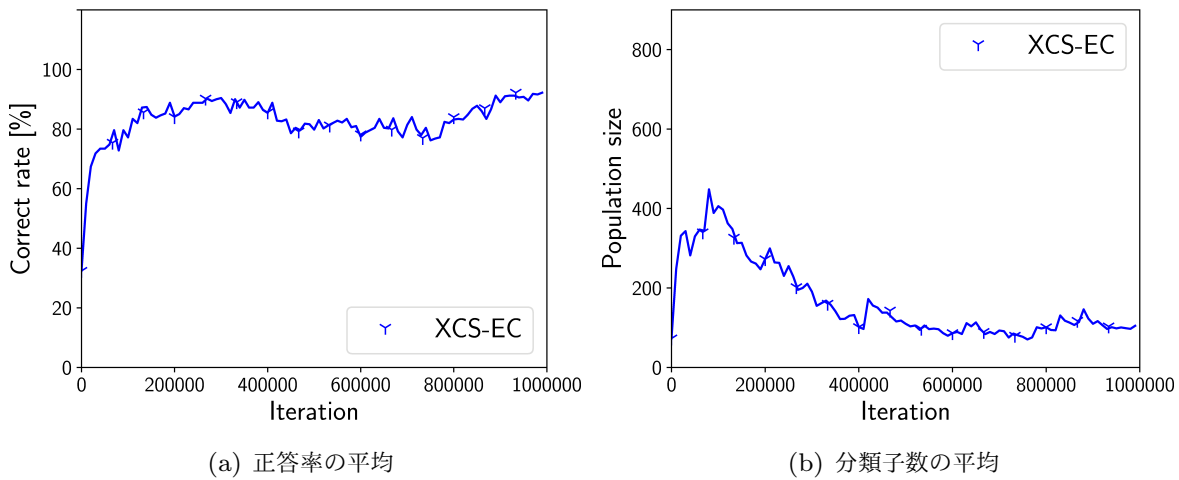


図 9.10 $\sigma_R = 1000$

となる．表 9.11 に $(1 - \sqrt{1 - n/(n + \alpha^2)})/2$ の値を示す．行は分類子の評価回数 n を列は信頼区間 α を示す．

図 9.11 および 9.12 に $P_O = 0.3, 0.35$ の場合の正答率の平均と分類子数の平均を示す．横軸は Iteration を表し，縦軸は正答率の平均あるいは分類子数の平均を表す．表 9.12 に $[O]$ を獲得した試行数および最大 F の分類子数の平均を示す． $P_O = 0.3$ の場合では正答率の平均が 100% に達し，ほとんどの試行で $[O]$ を獲得する． $P_O = 0.35$ の場合では正答率の平均が 100% よりも小さく， $[O]$ を獲得する試行が少ない．この結果から，表 9.11 の太字の部分に示す様に，信頼区間 $4\sigma \sim 5\sigma$ 程度かつ分類子の評価回数が 100 程度の値と合致する．

表 9.11 入出力における不確実性に対する XCS-EC の適用範囲（正誤が変化する確率 p ）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	0.35	0.23	0.16	0.11	0.08	0.06
	50	0.43	0.36	0.30	0.25	0.21	0.18
	100	0.45	0.40	0.36	0.31	0.28	0.24
	150	0.46	0.42	0.38	0.34	0.31	0.28
	200	0.46	0.43	0.40	0.36	0.33	0.30

表 9.12 入出力における不確実性を伴う場合の XCS-EC の最適解の獲得試行数

	$P_O = 0.3$	$P_O = 0.35$
[O] の獲得試行数	5 (100%)	1 (20%)
上位 32 個での [O] の獲得試行数	4 (80%)	0 (0%)
最大 F の分類子数の平均	32.4	33.8

入出力における不確実性および判定の信憑性における不確実性の両方を伴う場合

$p = 0, \sigma_R = 0$ の場合の正誤の報酬の平均の距離を R とし、不確実性を伴った場合の正誤の報酬の平均の距離 R_{p,σ_R} および標準偏差 σ_{p,σ_R} は

$$R_{p,\sigma_R} = \mu_c - \mu_I = (1 - 2p)R \quad (9.18)$$

$$\sigma_{p,\sigma_R} = \sqrt{p(1-p)R^2 + \sigma_R^2} \quad (9.19)$$

となり、式 (9.12) に当てはめると、

$$\begin{aligned} \frac{2\alpha\sqrt{p(1-p)R^2 + \sigma_R^2}}{\sqrt{n}} &< (1 - 2p)R \\ 4\alpha^2\{p(1-p)R^2 + \sigma_R^2\} &< n(4p^2 - 4p + 1)R^2 \\ p^2 - p + \frac{nR^2 - 4\alpha^2\sigma_R^2}{4R^2(n + \alpha^2)} &> 0 \end{aligned} \quad (9.20)$$

これを解くと $p < 0.5$ の制約から

$$p < \frac{1}{2} \left(1 - \sqrt{1 - \frac{nR^2 - 4\alpha^2\sigma_R^2}{R^2(n + \alpha^2)}} \right) \quad (9.21)$$

となる。これまでの実験結果と照らし合わせると信頼区間 $4\sigma \sim 5\sigma$ 程度かつ入出力の組み合わせの評価回数が 100 程度と関連するため表 9.13 に (1 -

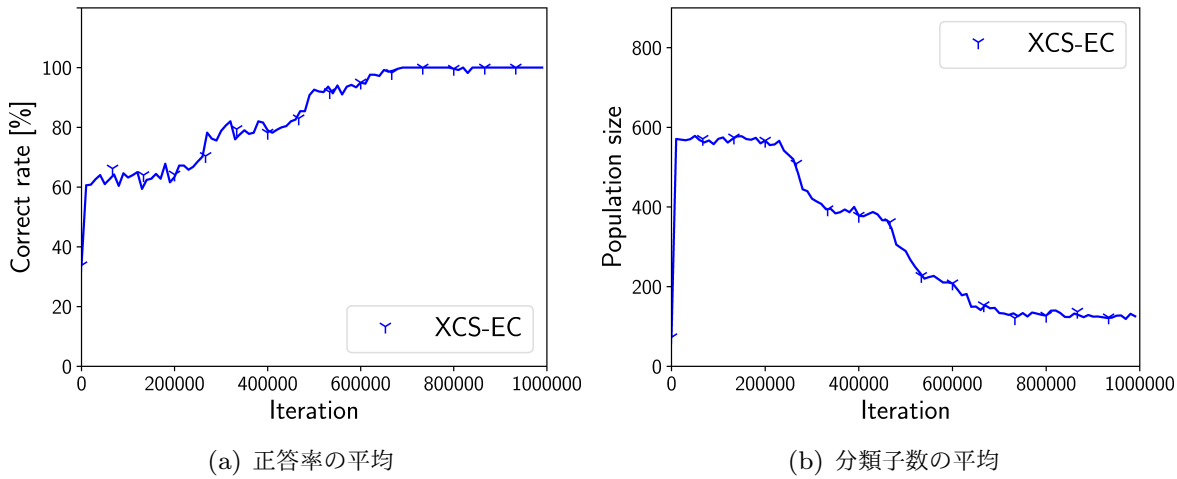


図 9.11 $P_O = 0.3$

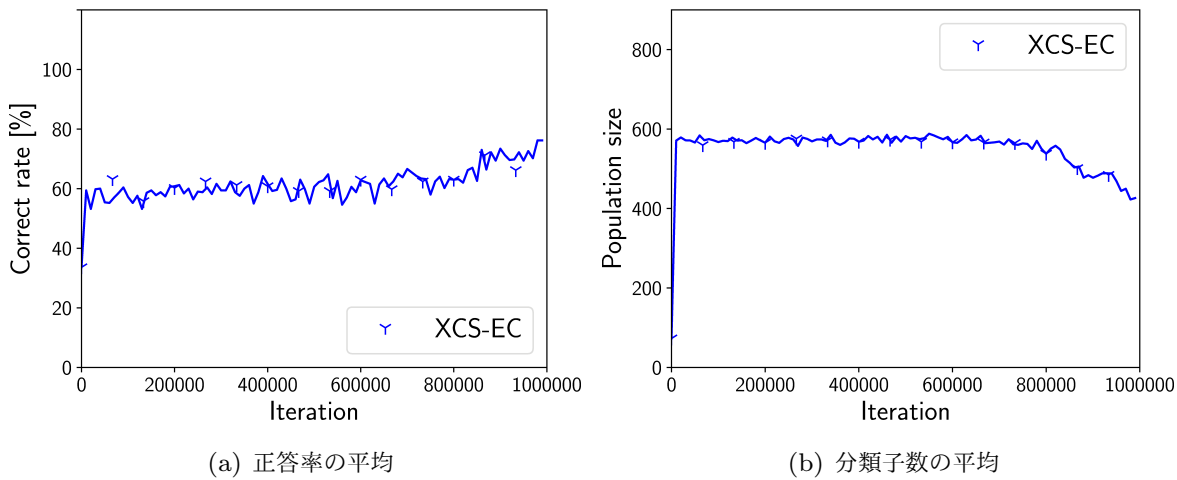


図 9.12 $P_O = 0.35$

$\sqrt{1 - (nR^2 - 4\alpha^2\sigma_R^2)/(R^2(n + \alpha^2))}/2$ に対し, $n = 100$ とした時の値を示す. 行は信頼区間 n を列は判定の信憑性における不確実性 σ_R を示す.

図 9.13 および 9.14 に $(P_O, \sigma_R) = (0.25, 200), (0.3, 200)$ の場合の正答率の平均と分類子数の平均を示す. 横軸は Iteration を表し, 縦軸は正答率の平均あるいは分類子数の平均を表す. 表 9.14 に $[O]$ を獲得した試行数および最大 F の分類子数の平均を示す. $(P_O, \sigma_R) = (0.25, 200)$ の場合では正答率の平均が 100% に達し, 全ての試行で $[O]$ を獲得する. $(P_O, \sigma_R) = (0.3, 200)$ の場合では正答率の平均が 100% よりも小さく, $[O]$ を獲得する試行が少ない. この結果から, 表 9.13 の太字の部分に示す様に, 信頼区間 $4\sigma \sim 5\sigma$ 程度の値と合致する.

中心極限定理より分類子の評価回数 n が多くなればなるほど分類子が獲得する報酬の

表 9.13 入出力における不確実性および判定の信憑性における不確実性を伴う場合に対する XCS-EC の適用範囲 (R に対する倍率)

		判定の信憑性における不確実性 σ_R						
		0	100	200	300	400	500	600
信頼区間 α	1	0.45	0.45	0.45	0.44	0.44	0.43	0.42
	2	0.40	0.40	0.39	0.39	0.37	0.36	0.35
	3	0.36	0.35	0.35	0.33	0.32	0.30	0.28
	4	0.31	0.31	0.30	0.28	0.26	0.24	0.21
	5	0.28	0.27	0.26	0.24	0.21	0.18	0.15
	6	0.24	0.24	0.22	0.20	0.17	0.14	0.10

表 9.14 入出力における不確実性および判定の信憑性を伴う場合の XCS-EC の最適解の獲得試行数

	$(P_O, \sigma_R) = (0.25, 200)$	$(P_O, \sigma_R) = (0.3, 200)$
[O] の獲得試行数	5 (100%)	3 (60%)
上位 32 個での [O] の獲得試行数	5 (100%)	2 (40%)
最大 F の分類子数の平均	32	30.4

平均は真の平均に近づくため、判定の信憑性における不確実性においては式 (9.13) より、対応可能な不確実性の強度が大きくなる。しかし、XCS-EC はその時点において保持している分類子に基づいて報酬や正誤を推測するため、評価回数 n を増やすには限界がある。

9.2.4 XCS-RG

判定の信憑性における不確実性（正規分布）を伴う場合

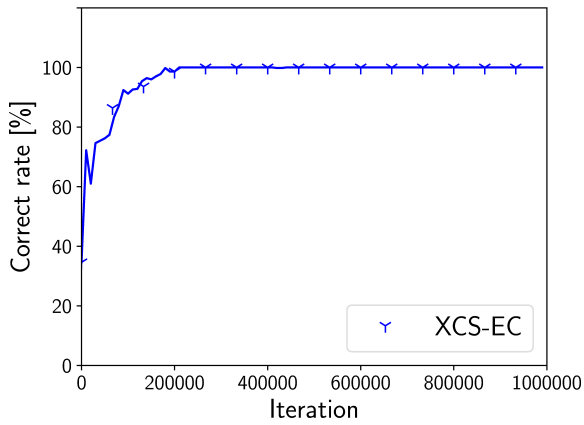
XCS-RG は状態行動対毎に獲得した報酬の平均を求め、その分布によって分類子の正確性を判定する。つまり分類子に照合する状態行動対が図 9.15 の μ_C の周辺の範囲内のみあるいは μ_I の周辺の範囲内だけに収まっていることと、それぞれの範囲が d よりも小さければ正確となる。つまり、

$$\frac{2\alpha\sigma}{\sqrt{n}} < d \quad (9.22)$$

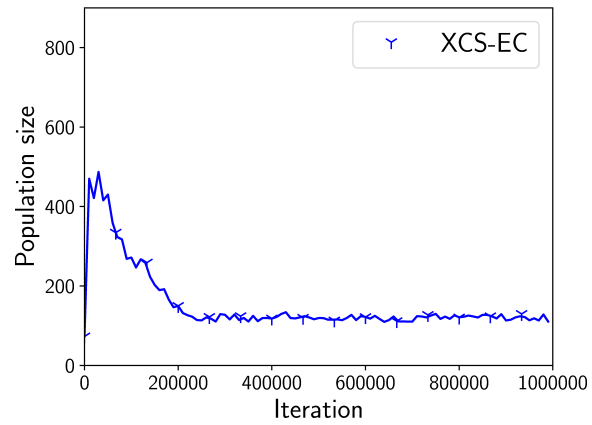
の条件を満たせば良い。ここで $d = R - 2\alpha\sigma/\sqrt{n}$ から

$$\frac{2\alpha\sigma}{\sqrt{n}} < R - \frac{2\alpha\sigma}{\sqrt{n}} \quad (9.23)$$

$$\sigma < \frac{\sqrt{n}}{4\alpha} R \quad (9.24)$$

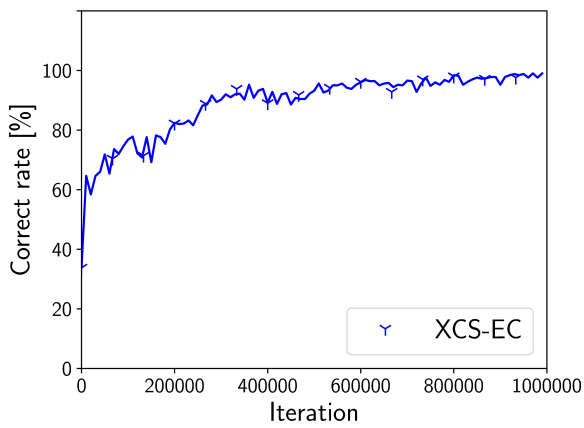


(a) 正答率の平均

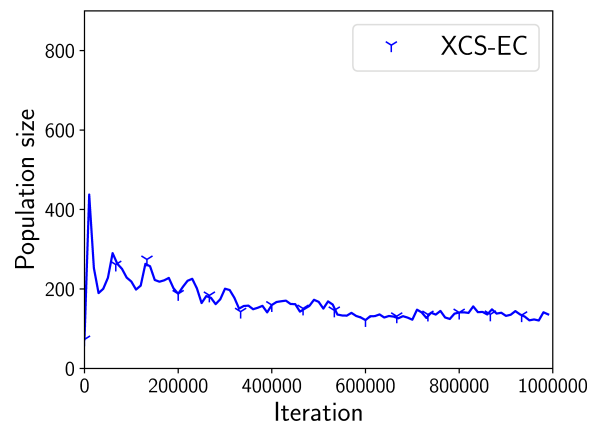


(b) 分類子数の平均

図 9.13 $(P_O, \sigma_R) = (0.25, 200)$



(a) 正答率の平均



(b) 分類子数の平均

図 9.14 $(P_O, \sigma_R) = (0.3, 200)$

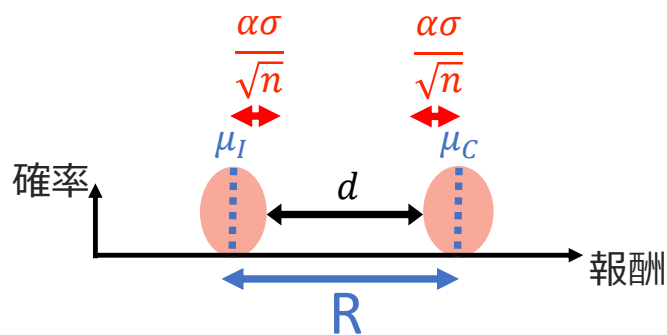


図 9.15 XCS-RG の正確な分類子が満たす条件

表 9.15 判定の信憑性における不確実性（正規分布）に対する XCS-RG の適用範囲（ R に対する倍率）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	0.79	0.40	0.26	0.20	0.16	0.13
	50	1.77	0.88	0.59	0.44	0.35	0.29
	100	2.50	1.25	0.83	0.63	0.50	0.42
	200	3.54	1.77	1.18	0.88	0.71	0.59
	250	3.95	1.98	1.32	0.99	0.79	0.66
	300	4.33	2.17	1.44	1.08	0.87	0.72

表 9.16 判定の信憑性における不確実性（正規分布）を伴う場合の XCS-RG の最適解の獲得試行数

	$\sigma_R = 1000$	$\sigma_R = 1100$
[O] の獲得試行数	4 (80%)	0 (0%)
上位 32 個での [O] の獲得試行数	3 (60%)	0 (0%)
最大 F の分類子数の平均	31.8	20.6

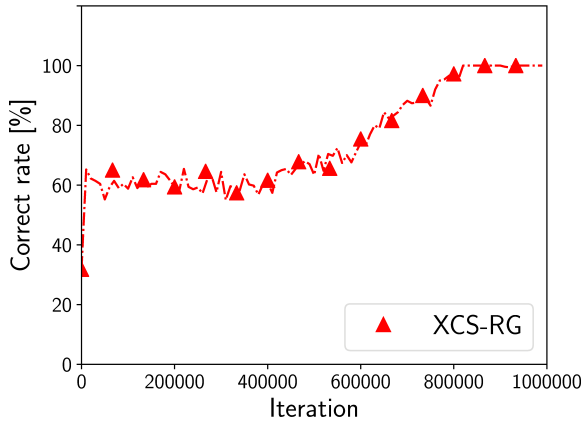
となる．表 9.15 に $\sqrt{n}/(4\alpha)$ の値を示す．行は状態行動対の評価回数 n を列は信頼区間 α を示す．

図 9.16 および 9.17 に $\sigma_R = 1000, 1100$ の場合の正答率の平均と分類子数の平均を示す．横軸は Iteration を表し，縦軸は正答率の平均あるいは分類子数の平均を表す．表 9.16 に [O] を獲得した試行数および最大 F の分類子数の平均を示す． $\sigma_R = 1000$ の場合では正答率の平均が 100% に近く，ほとんどの試行で [O] を獲得する． $\sigma_R = 1100$ の場合では正答率の平均が 100% よりも小さく，[O] を獲得する試行がなく最大 F の分類子数の平均が大きく低下している．この結果から，表 9.15 の太字の部分に示す様に，信頼区間 $4\sigma \sim 5\sigma$ 程度かつ分類子の評価回数が 250 程度 ($1,000,000/(2^{11} \times 2)$) の値と合致する．ただし，11-Multiplexer 問題において入力の種類は 2^{11} 通りで出力の種類は 2 通りであるため入出力の組み合わせは 4,096 通りとなり，Iteration は 1,000,000 なので各入出力の組み合わせは 250 回程度学習データに含まれる．

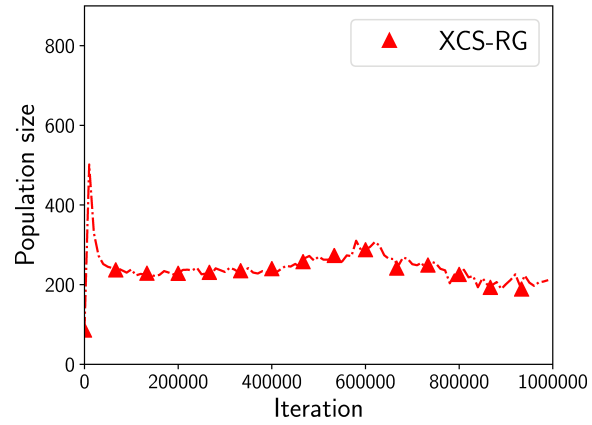
入出力における不確実性を伴う場合

$p = 0$ の場合の正誤の報酬の平均の距離を R とし，不確実性を伴った場合の正誤の報酬の平均の距離 R_p および標準偏差 σ_p は

$$R_p = \mu_c - \mu_I = (1 - 2p)R \quad (9.25)$$

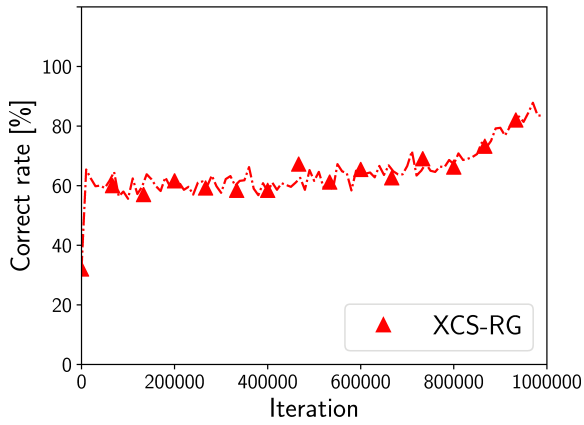


(a) 正答率の平均

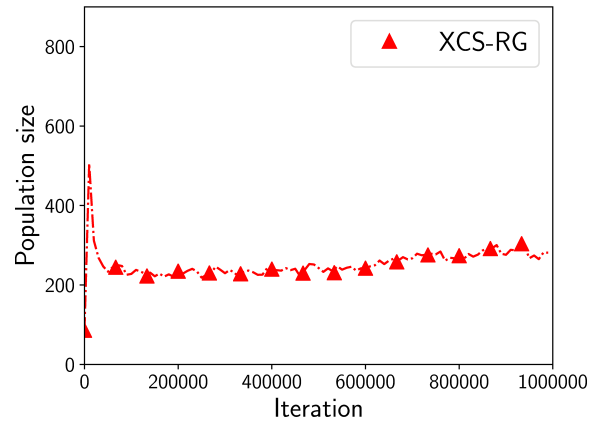


(b) 分類子数の平均

図 9.16 $\sigma_R = 1000$



(a) 正答率の平均



(b) 分類子数の平均

図 9.17 $\sigma_R = 1100$

$$\sigma_p = \sqrt{p(1-p)}R \quad (9.26)$$

となり，式 (9.23) に当てはめると，

$$\begin{aligned} \frac{2\alpha\sqrt{p(1-p)}R}{\sqrt{n}} &< (1-2p)R - \frac{2\alpha\sqrt{p(1-p)}R}{\sqrt{n}} \\ 16\alpha^2 p(1-p) &< n(4p^2 - 4p + 1) \\ p^2 - p + \frac{n}{16n + 4\alpha^2} &> 0 \end{aligned} \quad (9.27)$$

これを解くと $p < 0.5$ の制約から

$$p < \frac{1}{2} \left(1 - \sqrt{1 - \frac{n}{n + 4\alpha^2}} \right) \quad (9.28)$$

表 9.17 入出力における不確実性に対する XCS-RG の適用範囲（正誤が変化する確率 p ）

		信頼区間 α					
		1	2	3	4	5	6
評価回数 n	10	0.23	0.11	0.06	0.04	0.02	0.02
	50	0.36	0.25	0.18	0.13	0.09	0.07
	100	0.40	0.31	0.24	0.19	0.15	0.12
	200	0.43	0.36	0.30	0.25	0.21	0.18
	250	0.44	0.38	0.32	0.27	0.23	0.20
	300	0.44	0.39	0.34	0.29	0.25	0.22

表 9.18 入出力における不確実性を伴う場合の XCS-RG の最適解の獲得試行数

	$P_O = 0.25$	$P_O = 0.3$
[O] の獲得試行数	5 (100%)	0 (0%)
上位 32 個での [O] の獲得試行数	5 (100%)	0 (0%)
最大 F の分類子数の平均	32	2.0

となる．表 9.17 に $(1 - \sqrt{1 - n/(n + 4\alpha^2)})/2$ の値を示す．行は状態行動対の評価回数 n を列は信頼区間 α を示す．

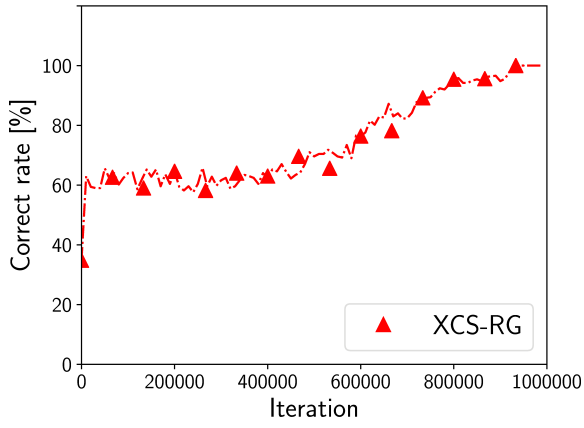
図 9.18 および 9.19 に $P_O = 0.25, 0.3$ の場合の正答率の平均と分類子数の平均を示す．横軸は Iteration を表し，縦軸は正答率の平均あるいは分類子数の平均を表す．表 9.18 に [O] を獲得した試行数および最大 F の分類子数の平均を示す． $P_O = 0.25$ の場合では正答率の平均が 100% に達し，全ての試行で [O] を獲得する． $P_O = 0.3$ の場合では正答率の平均が 100% よりも小さく，[O] を獲得する試行がない．この結果から，表 9.17 の太字の部分に示す様に，信頼区間 $4\sigma \sim 5\sigma$ 程度かつ分類子の評価回数が 250 程度 $(1,000,000/(2^{11} \times 2))$ の値と合致する．

入出力における不確実性と判定の信憑性における不確実性の両方を伴う場合

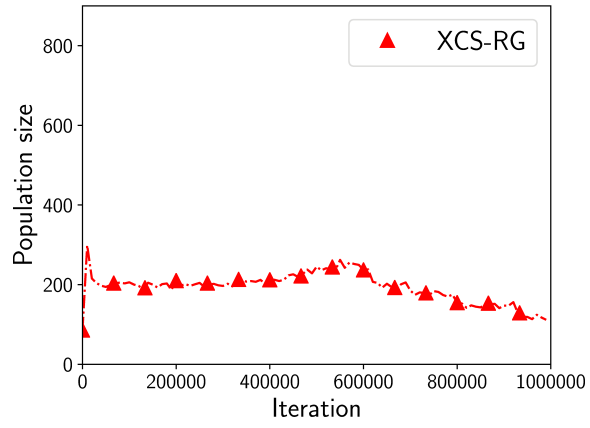
$p = 0, \sigma_R = 0$ の場合の正誤の報酬の平均の距離を R とし，不確実性を伴った場合の正誤の報酬の平均の距離 R_{p,σ_R} および標準偏差 σ_{p,σ_R} は

$$R_{p,\sigma_R} = \mu_c - \mu_I = (1 - 2p)R \quad (9.29)$$

$$\sigma_{p,\sigma_R} = \sqrt{p(1-p)R^2 + \sigma_R^2} \quad (9.30)$$

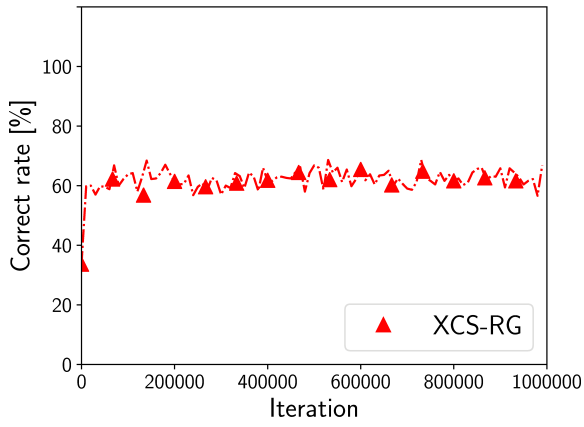


(a) 正答率の平均

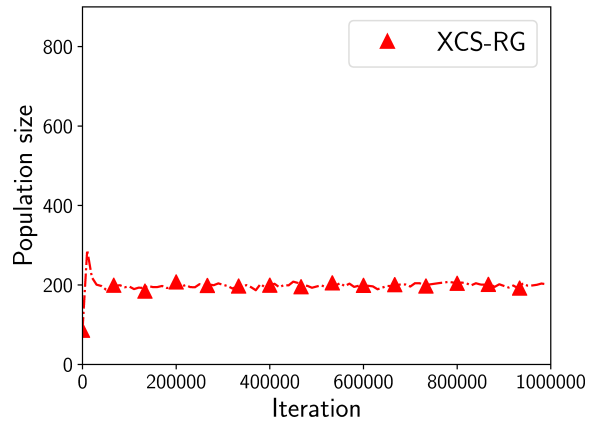


(b) 分類子数の平均

図 9.18 $P_O = 0.25$



(a) 正答率の平均



(b) 分類子数の平均

図 9.19 $P_O = 0.3$

となり，式 (9.23) に当てはめると，

$$\begin{aligned} \frac{2\alpha\sqrt{p(1-p)R^2 + \sigma_R^2}}{\sqrt{n}} &< (1-2p)R - \frac{2\alpha\sqrt{p(1-p)R^2 + \sigma_R^2}}{\sqrt{n}} \\ 16\alpha^2\{p(1-p)R^2 + \sigma_R^2\} &< n(4p^2 - 4p + 1)R^2 \\ p^2 - p + \frac{nR^2 - 16\alpha^2\sigma_R^2}{4R^2(n + 4\alpha^2)} &> 0 \end{aligned} \quad (9.31)$$

となる．これを解くと $p < 0.5$ の制約から

$$p < \frac{1}{2}\left(1 - \sqrt{1 - \frac{nR^2 - 16\alpha^2\sigma_R^2}{R^2(n + 4\alpha^2)}}\right) \quad (9.32)$$

表 9.19 入出力における不確実性および判定の信憑性における不確実性を伴う場合に対する XCS-RG の適用範囲 (R に対する倍率)

		判定の信憑性における不確実性 σ_R						
		0	100	200	300	400	500	600
	1	0.44	0.44	0.43	0.43	0.42	0.41	0.40
	2	0.38	0.37	0.37	0.36	0.34	0.33	0.31
	3	0.32	0.32	0.31	0.29	0.27	0.25	0.22
	4	0.27	0.27	0.26	0.24	0.21	0.18	0.15
	5	0.23	0.23	0.21	0.19	0.16	0.12	0.08
	6	0.20	0.19	0.17	0.15	0.11	0.07	0.03

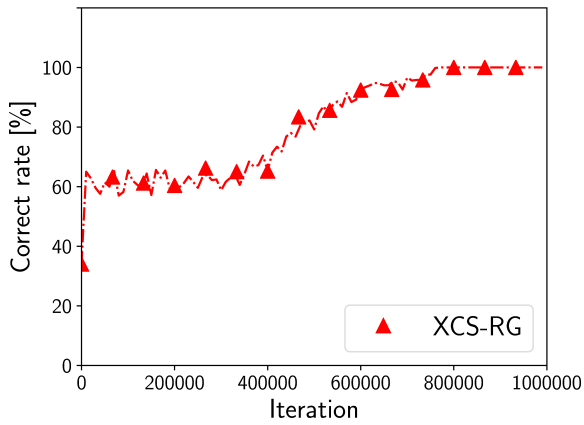
表 9.20 入出力における不確実性および判定の信憑性を伴う場合の XCS-RG の最適解の獲得試行数

	$(P_O, \sigma_R) = (0.2, 300)$	$(P_O, \sigma_R) = (0.25, 300)$
[O] の獲得試行数	5 (100%)	0 (0%)
上位 32 個での [O] の獲得試行数	5 (100%)	0 (0%)
最大 F の分類子数の平均	32	3.8

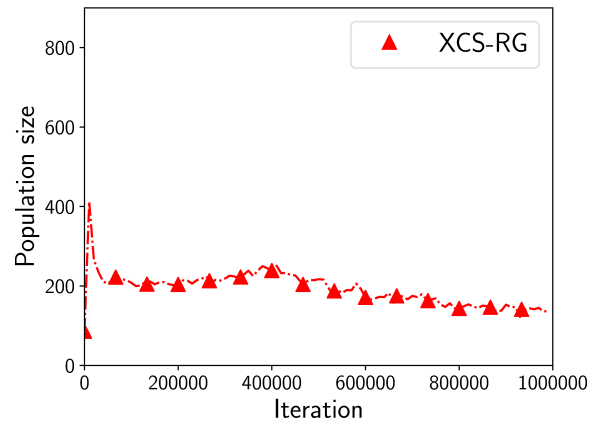
となる。これまでの実験結果と照らし合わせると信頼区間 $4\sigma \sim 5\sigma$ 程度かつ入出力の組み合わせの評価回数が 250 程度 ($1,000,000/(2^{11} \times 2)$) と関連するため表 9.19 に $(1 - \sqrt{1 - (nR^2 - 16\alpha^2\sigma_R^2)/(R^2(n + 4\alpha^2))})/2$ に対し、 $n = 250$ とした時の値を示す。行は信頼区間 n を列は判定の信憑性における不確実性 σ_R を示す。

図 9.20 および 9.21 に $(P_O, \sigma_R) = (0.2, 300), (0.25, 300)$ の場合の正答率の平均と分類子数の平均を示す。横軸は Iteration を表し、縦軸は正答率の平均あるいは分類子数の平均を表す。表 9.20 に [O] を獲得した試行数および最大 F の分類子数の平均を示す。 $(P_O, \sigma_R) = (0.2, 300)$ の場合では正答率の平均が 100% に達し、全ての試行で [O] を獲得する。 $(P_O, \sigma_R) = (0.25, 300)$ の場合では正答率の平均が 100% よりも小さく、[O] を獲得する試行がない。この結果から、表 9.19 の太字の部分に示す様に、信頼区間 $4\sigma \sim 5\sigma$ 程度の値と合致する。

中心極限定理より分類子の評価回数 n が多くなればなるほど分類子が獲得する報酬の平均は真の平均に近づくため、判定の信憑性における不確実性においては式 (9.24) より、対応可能な不確実性の強度が大きくなる。XCS-RG は Recording table に獲得報酬の平均記録することから分類子の生成や削除と関係ないため、データ数を増やせば評価回数 n を増加させることができる。

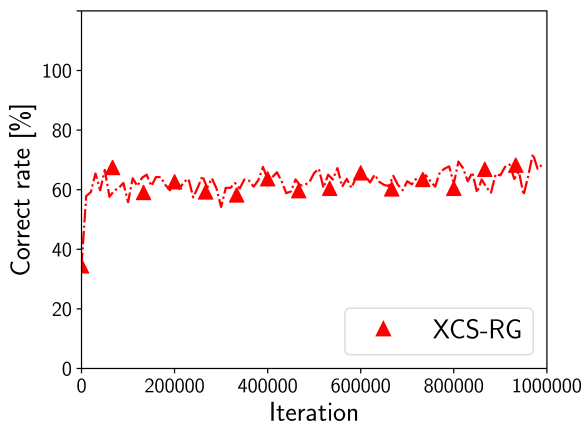


(a) 正答率の平均

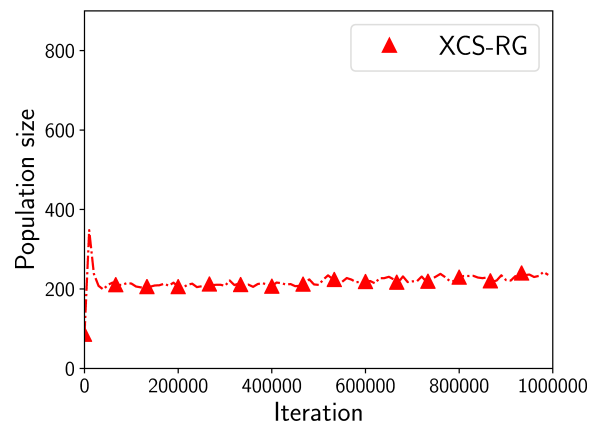


(b) 分類子数の平均

図 9.20 $(P_O, \sigma_R) = (0.2, 300)$



(a) 正答率の平均



(b) 分類子数の平均

図 9.21 $(P_O, \sigma_R) = (0.25, 300)$

第 10 章

結言

10.1 本研究の成果

本研究では学習分類子システム (Learning Classifier System: LCS) において、不確実性を伴うデータから不確実性の種類やその強度に関する情報なしに、不確実性を取り除いた場合と同等の知識を発見するメカニズムを構築し、その有効性を Multiplexer 問題に対し不確実性を模擬したノイズを付加したデータに適用し検証することを目的とした。その目的に向け不確実性を (i) 入出力における不確実性および (ii) 判定の信憑性における不確実性に分け、(i) 入出力における不確実性のみに対応可能なメカニズム、(ii) 判定の信憑性における不確実性のみに対応可能なメカニズムおよび 2 つの (i) 入出力における不確実性と (ii) 判定の信憑性における不確実性の双方に対応可能なメカニズムを提案した。(i) 入出力における不確実性のみに対応可能な手法として推測報酬に基づく学習分類子システム (XCS based on Estimated Reward: XCS-ER) を (ii) 判定の信憑性における不確実性のみに対応可能な手法として報酬の範囲に基づく学習分類子システム (XCS based on Range of Reward: XCS-RR) を (i) 入出力における不確実性および (ii) 判定の信憑性における不確実性の双方に対応可能な手法として報酬の組分けに基づく学習分類子システム (XCS based on Reward Group: XCS-RG) および正誤推測に基づく学習分類子システム (XCS based on Estimated Correctness: XCS-EC) を提案した。提案メカニズムの有効性を検証するために、LCS において一般的なベンチマーク問題である Multiplexer 問題において (i) 入出力における不確実性および (ii) 判定の信憑性における不確実性を模擬したノイズを付加した問題を設計し、提案メカニズムを適用したところ以下の知見を得た。

- 入出力対における尤もらしい評価の採用による (i) 入出力における不確実性への対応 (第 5 章)

構築した XCS-ER は、学習中における分類子を用いて入出力の組み合わせにおける尤もらしい評価を推測することで、入出力における不確実性に対応を図った。

入出力における不確実性の強度 (P_I, P_O) について把握していなくても、入力に対して出力毎に求めた評価の期待値の大小関係から正しい評価を推測し、正しく分類子を一般化可能である。期待値（平均値）を用いて推測するため、対応可能な不確実性は強度の最大値よりも平均的な値に依存する。

- 評価を包括的に扱うことによる (ii) 判定の信憑性における不確実性への対応（第 6 章）

構築した XCS-RR は、分類子が獲得した評価全体を他の分類子が獲得した評価と比較することで、判定の信憑性における不確実性に対応を図った。判定の信憑性における不確実性の強度 (σ_R) について把握していなくても、分類子が獲得した評価の最大値および最小値の他の分類子の値との大小関係から正しく分類子の正確性を判定し、正しく分類子を一般化可能である。最大値や最小値を用いて正確性を判定するため、対応可能な不確実性は強度の最大値に依存する。

- 上記のアプローチを組み合わせることによる (i) 入出力における不確実性および (ii) 判定の信憑性における不確実性への対応（第 7 章および第 8 章）

第 5 章の入出力対における尤もらしい評価を採用するアプローチと第 6 章の評価を包括的に扱うアプローチの組み合わせによって構築した XCS-EC および XCS-RG は入出力における不確実性および判定の信憑性における不確実性への対応を図った。XCS-EC の結果から、学習中における分類子を用いて入出力の組み合わせにおける尤もらしい評価を推測することで、不確実性の種類およびその強度について知らなくても、両方の不確実性に対応可能であることが示された。さらに XCS-RG の結果から、分類子の生成や削除とは別に獲得した評価に関する情報を記録することで、より安定して不確実性に対応可能であることが示された。

- 不確実性を伴うデータから知識を獲得するメカニズムの設計（第 9 章）

これまでに構築した XCS-ER, XCS-RR, XCS-EC および XCS-RG から入出力における不確実性あるいは判定の信憑性における不確実性に対応するために必要となるメカニズムを明らかにした。さらに、そのメカニズムおよびそれを実装した 4 手法の特徴と問題が持つ特徴との関係から、それぞれの手法の使い分けを示した。

- 構築した手法の解析的な適用範囲の導出（第 9 章）

正確だとみなすべき分類子が獲得する報酬の条件から適応限界となる不確実性の強度を導出し、実際の結果と整合性があることが示された。求めた適用範囲は XCS-RR については最大値や最小値から分類子の正確性を判定するため対応可能な不確実性の強度に限界があること、XCS-ER および XCS-EC には分類子の評価回数 n を増やすには限界があるため対応可能な不確実性の強度に実質的な限界があること、XCS-RG は Recording table に獲得報酬の平均を記録するためデータ数（入出力対の評価回数 n ）に応じて対応可能な不確実性の強度が決まることが明

らかになった.

10.2 今後の課題

今後の展開として本研究で提案したメカニズムを実問題を含めた多くの問題に適用できるよう適応可能範囲を広げるため、また更なる精度や安定性の向上のため以下の課題に取り組む必要がある.

10.2.1 複数の手法に共通する問題

パラメータの動的な設定

XCS-ER において ϵ_0 や包摂条件に関わるパラメータとして θ_{RE} を, XCS-RR において分類子の正確性判定に関わるパラメータとして θ_{RR} を, XCS-EC において ϵ_0 に関わるパラメータとして θ_{EC} を新設した. また, XCS には θ_{mna} , $P_{\#}$, p_I , ϵ , F_I , p_{explr} , β , ϵ_0 , α , ν , θ_{sub} , θ_{GA} , χ , μ , N , θ_{del} , δ と大量のパラメータがある. これらのパラメータは問題における入出力空間の広さや伴う不確実性の種類とその強度によって適切な値が異なる. 提案メカニズムの学習の安定性や適応可能範囲の向上のためにはそれらのパラメータの動的な設定が必要となる. これらのパラメータの値を不確実性の種類やその強度を考慮した値に設定することによる不確実性に対する適応性を向上することが考えられる. 不確実性への適応性の向上には, 不正確な分類子を生成を抑えることおよび不正確な分類子の早急な削除が考えられる. 分類子の正確性は照合する入出力対の組み合わせによって決まるため, XCS-RG の様にそれぞれの入出力対の獲得した報酬の分布を知れば不正確な分類子の生成を抑えることは可能になるが, 遺伝的アルゴリズムに用いられるパラメータの設定では調整することが難しい. しかし, 不正確な分類子を早急に削除し評価を安定させることは可能である. 分類子の早急な削除に貢献するパラメータとして式 (3.8) で用いられる α および ν がある. 不正確な度合いが高い分類子ほど正答率や XCS-ER および XCS-EC の正解となる行動の予測に影響を及ぼすので早急に削除すべきである. そのため, ν の設定がこの目的に沿う. 以下に ν の設定について述べる. 図 10.1 に ν を 10 から 5 に変更したときの $\sigma_R = 200$ (ケース U) の結果を示す. 図 7.21 と比較して, この結果では XCS-EC が正答率の平均が 100% まで達していない違いがある. ν が 5 に小さくなることで式 (3.8) から不正確な分類子の正確性 κ が大きくなり, 不正確な分類子が ν が大きいときに比べ削除されにくくなる. XCS-EC は分類子が獲得した報酬の加重平均から分類子の正確性を判定するため, 特に不正確な分類子の影響を受けやすい. この様に, XCS と共通するパラメータであっても, 不確実性に対する学習性能や知識獲得性能に影響を受ける. Nakata らは不確実性を伴わない理想的な環境におけるハイパーパラメータの設定

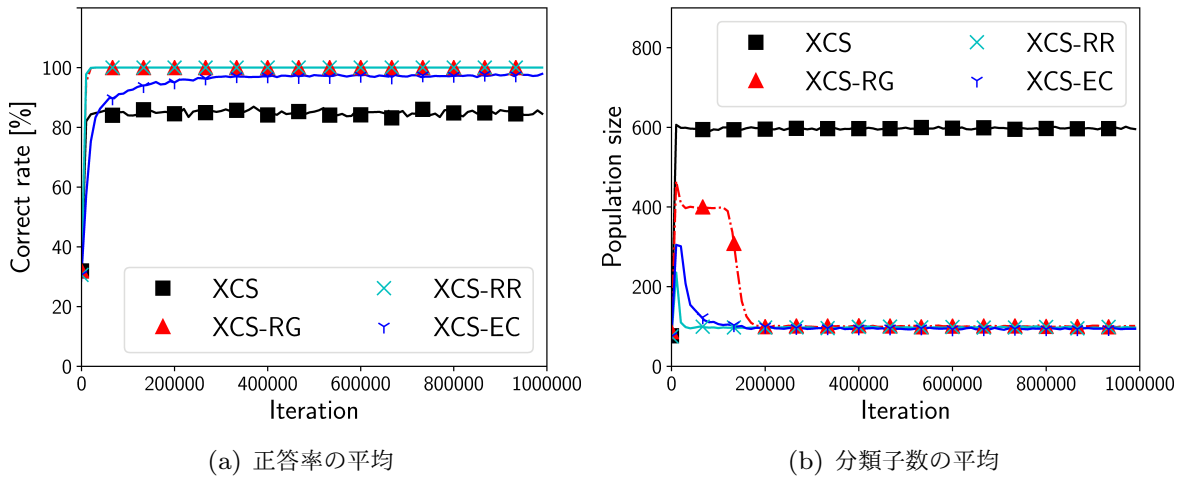


図 10.1 $\sigma_R = 200$ ($\nu = 5$)

について述べている [89] ので、この文献を参考に不確実性の種類や強度を限定することで適切なハイパーパラメータの設定を目指す。

一時的な不確実性の強度が変化する場合への対処

第 5, 6, 7, 8 章の結果から提案メカニズムには適用可能な不確実性の種類と強度に制限がある。それぞれのメカニズムの適応可能範囲は恒常的に伴う不確実性の強度を示しており、適応可能な恒常的な不確実性の強度を広げることは必要である。しかしながら、外れ値のような突発的な不確実性を伴う場合が考えられる。突発的な不確実性を伴うと分類子が獲得する報酬から算出される平均や分散などの値が大きく変化し、正確と判定されるべき分類子が不正確であると判定される場合が考えられる。これに対処するため、恒常的な不確実性を捉えながら突発的な不確実性を除く異常検知のような仕組みを導入し、分類子の正確性判定に影響を及ぼさない仕組みの導入が考えられる。

また、表 5.3 の様に [O] を獲得した試行と [O] を獲得できない試行が混在する場合がある。それは学習初期において不確実性を伴うデータの多寡によるものである。[O] を獲得できた試行は学習初期において学習データに不確実性を伴う割合が少なく、[O] あるいは [O] に近い度合で一般化された正確な分類子が生成され、安定して評価ができた試行である。[O] を獲得できない試行は学習初期において特定の入力あるいは入力全体において学習データに不確実性を伴う割合が多く、分類子の正確性を正しく判定できなかった試行である。学習初期は用いれるデータが少なく情報が不足しているため難しいが、上述した仕組みが導入されれば [O] を獲得する試行を増やすことができる。

実問題への適用

今回用いた Multiplexer 問題の様に一般化を考慮して人工的に作られた問題以外には、一般化によって得られる最適解集合 $[O]$ が定義されておらず、 $[O]$ が求まるとも限らない。その原因として全ての状態行動対が学習データにおいて網羅されていない場合があること、状態行動対ごとに学習データにおけるインスタンス数が大幅に異なる場合があることが挙げられる。

学習データに含まれない状態行動対は不確実性の有無にかかわらず学習に悪影響を与える。提案メカニズムを含め XCS は正確と判定される限り分類子を最大限に一般化する。学習データに含まれない状態行動対は評価データにおける獲得報酬値に関わらず、学習データにおいて分類子が獲得する報酬に影響を及ぼさないため、その行動対が含まれる様に一般化される。そのため学習データに含まれない状態行動対に照合する過剰に一般化された分類子が生成され、評価データにおける正答率が低下する。

全ての状態行動対が学習データに含まれていたとしても、極端にインスタンス数が少ない状態行動対があると不確実性を伴う場合に学習に悪影響がある場合がある。インスタンス数が少ないと伴う不確実性の種類やその強度を把握することは困難であり、学習データにおける振る舞い（正誤判定）と評価データにおける振る舞いが異なる場合がある。例えば、ある状態行動対は学習データに1つのインスタンスしかなく、そのインスタンスが正誤判定における不確実性を伴い正誤判定が逆転していた場合、誤った学習が行われる。この問題は LCS に限ったものではないが、正確性を分類子の評価とする XCS や提案メカニズムには重大な問題である。インスタンス数が少ないため全体に占める割合が小さく、他の学習手法では正解率が少し下がる程度であるが、XCS においては正確性の判定では不正確と判定され、知識として用いられない場合がある。

上記の2つの問題を解決した後に、比較に用いた XCS および提案した4つのメカニズムを適用し、獲得された分類子から得られた知識を比較することにより、問題の特徴を明らかにすることやその結果からどのような問題にどのメカニズムを適用すべきかの指針を拡充することが望まれる。

10.2.2 手法固有の問題

XCS-EC および XCS-ER の安定性の向上

XCS-RR や XCS-RG は学習途中の場合を除き、学習や知識の獲得が成功するか失敗するかは試行による違いはほとんど無い。しかし、XCS-EC および XCS-ER は一部の環境においては4割から6割の試行でのみ学習や知識の獲得が成功する結果が得られた。これは複数の分類子の獲得した報酬の平均を基に出力の正誤を推測するため、学習の初期に正しく正誤を推測できるか否かが重要であるため、試行により学習や知識の獲得の成功に

違いが出る。XCS-EC および XCS-ER を実際に用いるためには試行による学習や知識の獲得の成否の違いを抑える必要がある。現在の手法のままで安定性の向上を図るには、学習データの並び順を変えて複数回学習するなどの対策を検討する必要がある。

XCS-RR の適応範囲の拡大

XCS-RR は分類子が獲得した報酬の最大値および最小値によって分類子の正確性を判定するため、XCS-EC や XCS-RG と比べて獲得報酬の範囲が広い（裾が厚い分布）に対応できない。不正解の場合に獲得される報酬の最大値が正解の場合に獲得される報酬の平均値に比べて小さい場合および正解の場合に獲得される報酬の最小値が不正解の場合に獲得される報酬の平均値に比べて大きい場合に対応できるため、機械的に $R/2$ で報酬を切り分けるよりは適応範囲は広いが、適応範囲が狭い。これを拡大するためには最大値や最小値を求めるのに比べるとコストがかかるが四分位点を用いるなど最大値や最小値以外の指標を活用することを検討する必要がある。

XCS-EC の外れ値への対応

第 8.2 節で示す様に XCS-EC は XCS-RR や XCS-RG よりも裾が厚い分布あるいは外れ値がある分布に弱い。XCS-RG の Recording table に比べ評価回数の少ない分類子の獲得報酬の平均は大きくばらつくため、評価の推測が誤りやすく、適切に学習できない。それに対応するために、手法に外れ値を除く機構を導入し適応性の向上を図る必要がある。

XCS-RG の入出力空間が広い問題への対応

第 7 章の結果から XCS-RG は他の 3 手法に比べて正答率の平均および分類子数の平均が収束するまでに時間がかかる。それは、第 7.5.1 項で述べた様に Recording table の全状態行動対 (s, a) において平均 $M(s, a)$ をある程度正しく求める必要があるためである。そのため、入出力空間が広くなり状態行動対 (s, a) の数が増えると、正答率の平均および分類子数の平均の収束にさらにデータを要する。また、人工的にデータを生成できない実問題などでは全状態行動対 (s, a) においてデータが得られるとは限らない。この場合に全状態行動対 (s, a) において平均 $M(s, a)$ を求めることは困難であるため、類似する状態行動対 (s, a) から平均 $M(s, a)$ を推測する機構を導入し、正答率の平均および分類子数の平均の収束の高速化を図る必要がある。

謝辞

本論文をまとめるにあたり，終始多大なるご指導とご教示をいただいた主任指導教員の高玉圭樹教授，指導教員の西野哲朗教授，佐藤寛之准教授に心より感謝の意を表します．博士論文の審査をしていただいた田野俊一教授，庄野逸教授，研究室に招聘された際に数日間にわたり一対一で研究議論していただいた Tim Kovacs 博士，International Workshop on Learning Classifier Systems 2019 に co-organizer として迎え入れるとともに研究議論していただいた横浜国立大学の中田雅也准教授，アウクスブルク大学の Anthony Stein 博士，特別研究員制度によって研究費の面で助成いただいた日本学術振興会様，並びに研究室の人々にこの場を借りてあらためて深く感謝申し上げます．最後にここまで私を支えてくれた家族にも感謝の意を表します．

参考文献

- [1] M. T. Ribeiro, S. Singh, and C. Guestrin. Why Should I Trust You?: Explaining the Predictions of Any Classifier. In *Proceedings of 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 1135–1144, 2016.
- [2] U. Fayyad, G. Piatetsky-Shapiro, and P. Smyth. From Data Mining to Knowledge Discovery in Databases. *AI magazine*, Vol. 17, No. 3, pp. 37–37, 1996.
- [3] G. Piatetski and W. Frawley. *Knowledge Discovery in Databases*. MIT press, 1991.
- [4] X. Glorot, A. Bordes, and Y. Bengio. Domain Adaptation for Large-scale Sentiment Classification: A Deep Learning Approach. In *Proceedings of 28th International Conference on Machine Learning (ICML-11)*, pp. 513–520, 2011.
- [5] B. E. Boser, I. M. Guyon, and V. N. Vapnik. A Training Algorithm for Optimal Margin Classifiers. In *Proceedings of Fifth Annual Workshop on Computational Learning Theory*, pp. 144–152, 1992.
- [6] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston. Random Forest: a Classification and Regression Tool for Compound Classification and QSAR Modeling. *Journal of Chemical Information and Computer Sciences*, Vol. 43, No. 6, pp. 1947–1958, 2003.
- [7] T. Chen and C. Guestrin. Xgboost: A Scalable Tree Boosting System. In *Proceedings of 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, 2016.
- [8] R. Agrawal and R. Srikant. Fast Algorithms for Mining Association Rules. In *Proceedings of 20th International Conference Very Large Data Bases*, Vol. 1215, pp. 487–499, 1994.
- [9] J. H. Holland. Escaping Brittleness: The Possibilities of General-Purpose Learning Algorithms Applied to Parallel Rule-Based Systems. *Machine learning*, Vol. 2, pp. 593–623, 1986.

- [10] S. W. Wilson. Classifier Fitness Based on Accuracy. *Evolutionary Computation*, Vol. 3, No. 2, pp. 149–175, 1995.
- [11] X. Wu. *Knowledge Acquisition from Databases*. Intellect books, 1995.
- [12] K. Orr. Data Quality and Systems Theory. *Communications of ACM*, Vol. 41, No. 2, pp. 66–71, 1998.
- [13] J. I. Maletic and A. Marcus. Data Cleansing: Beyond Integrity Analysis. In *Proceedings of 2000 Conference on Information Quality (IQ2000)*, pp. 200–209, 2000.
- [14] B. Frenay and M. Verleysen. Classification in the Presence of Label Noise: A Survey. *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 25, No. 5, pp. 845–869, 2014.
- [15] P. L. Lanzi and S. W. Wilson. Toward Optimal Classifier System Performance in Non-Markov Environments. *Evolutionary Computation*, Vol. 8, No. 4, pp. 393–418, 2000.
- [16] P. L. Lanzi and M. Colombetti. An Extension to the XCS Classifier System for Stochastic Environments. In *Proceedings of 1999 Genetic and Evolutionary Computation Conference (GECCO 1999)*, pp. 353–360, 1999.
- [17] J. Sun, F. Zhao, C. Wang, and S. Chen. Identifying and correcting mislabeled training instances. In *Proceedings of 2007 Future Generation Communication and Networking (FGCN 2007)*, Vol. 1, pp. 244–250, 2007.
- [18] J. R. Quinlan. Induction of decision trees. *Machine Learning*, Vol. 1, No. 1, pp. 81–106, 1986.
- [19] C. E. Brodley and M. A. Friedl. Identifying mislabeled training data. *Journal of artificial intelligence research*, Vol. 11, pp. 131–167, 1999.
- [20] J. Abellán and A. R. Masegosa. Bagging Decision Trees on Data Sets with Classification Noise. In *Proceedings of 2010 Foundations of Information and Knowledge Systems*, pp. 248–265, 2010.
- [21] G. L. Libralon, A. C. P. d. L. F. d. Carvalho, and A. C. Lorena. Pre-processing for Noise Detection in Gene Expression Classification Data. *Journal of the Brazilian Computer Society*, Vol. 15, No. 1, pp. 3–11, 2009.
- [22] D. Gamberger, N. Lavrač, and S. Džeroski. Noise Elimination in Inductive Concept Learning: A Case Study in Medical Diagnosis. In *Proceedings of 1996 Algorithmic Learning Theory*, pp. 199–212, 1996.
- [23] D. Gamberger and N. Lavrač. Conditions for Occam’s Razor Applicability and Noise Elimination. In *Proceedings of 1997 European Conference on Machine*

- Learning (ECML 1997)*, pp. 108–123, 1997.
- [24] D. Gamberger and N. Lavrač. Noise Detection and Elimination Applied to Noise Handling in a KRK Chess Endgame. In *Proceedings of 1997 Inductive Logic Programming*, pp. 72–88, 1997.
- [25] D. Gamberger, N. Lavrac, and S. Dzeroski. Noise Detection and Elimination in Data Preprocessing: Experiments in Medical Domains. *Applied Artificial Intelligence*, Vol. 14, No. 2, pp. 205–223, 2000.
- [26] T. M. Khoshgoftaar and P. Rebour. Generating Multiple Noise Elimination Filters with the Ensemble-partitioning Filter. In *Proceedings of 2004 IEEE International Conference on Information Reuse and Integration*, pp. 369–375, 2004.
- [27] J. Thongkam, G. Xu, Y. Zhang, and F. Huang. Support Vector Machine for Outlier Detection in Breast Cancer Survivability Prediction. In *Proceedings of 2008 Advanced Web and Network Technologies, and Applications*, pp. 99–109, 2008.
- [28] D. Garca-Gil, J. Luengo, S. Garca, and F. Herrera. Enabling Smart Data: Noise Filtering in Big Data Classification. *Information Sciences*, Vol. 479, pp. 135 – 152, 2019.
- [29] P. Yang, J. T. Ormerod, W. Liu, C. Ma, A. Y. Zomaya, and J. Y. H. Yang. AdaSampling for Positive-unlabeled and Label Noise Learning with Bioinformatics Applications. *IEEE transactions on cybernetics*, Vol. 49, No. 5, pp. 1932–1943, 2018.
- [30] G. H. John. Robust Decision Trees: Removing Outliers from Databases. In *Proceedings of 1995 Knowledge Discovery and Data Mining*, Vol. 95, pp. 174–179, 1995.
- [31] E. Blanzieri and F. Melgani. Nearest Neighbor Classification of Remote Sensing Images with the Maximal Margin Principle. *IEEE Transactions on geoscience and remote sensing*, Vol. 46, No. 6, pp. 1804–1811, 2008.
- [32] N. Segata, E. Blanzieri, and P. Cunningham. A Scalable Noise Reduction Technique for Large Case-Based Systems. In *Proceedings of 2009 Case-Based Reasoning Research and Development*, pp. 328–342, 2009.
- [33] N. Segata, E. Blanzieri, S. J. Delany, and P. Cunningham. Noise Reduction for Instance-based Learning with a Local Maximal Margin Approach. *Journal of Intelligent Information Systems*, Vol. 35, No. 2, pp. 301–331, 2010.
- [34] C. E. Brodley and M. A. Friedl. Identifying Mislabeled Training Data. *Journal of Artificial Intelligence Research*, Vol. 11, No. 1, pp. 131–167, 1999.

- [35] C. E. Brodley and M. A. Friedl. Identifying and Eliminating Mislabeled Training Instances. In *Proceedings of Thirteenth National Conference on Artificial Intelligence (AAAI 1996)*, pp. 799–805, 1996.
- [36] C. E. Brodley and M. A. Friedl. Improving Automated Land Cover Mapping by Identifying and Eliminating Mislabeled Observations from Training Data. In *1996 International Geoscience and Remote Sensing Symposium (IGARSS 1996)*, Vol. 2, pp. 1379–1381, 1996.
- [37] K. H. Lee, X. He, L. Zhang, and L. Yang. Cleannet: Transfer Learning for Scalable Image Classifier Training with Label Noise. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5447–5456, 2018.
- [38] W. Zhang, D. Wang, and X. Tan. Robust Class-Specific Autoencoder for Data Cleaning and Classification in the Presence of Label Noise. *Neural Processing Letters*, Vol. 50, No. 2, pp. 1845–1860, 2019.
- [39] D. L. Wilson. Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-2, No. 3, pp. 408–421, 1972.
- [40] N. P. Hughes, S. J. Roberts, and L. Tarassenko. Semi-supervised Learning of Probabilistic Models for ECG Segmentation. In *Proceedings of 26th Annual International Conference of IEEE Engineering in Medicine and Biology Society*, Vol. 1, pp. 434–437, 2004.
- [41] L. Joseph, T. W. Gyorkos, and L. Coupal. Bayesian Estimation of Disease Prevalence and the Parameters of Diagnostic Tests in the Absence of a Gold Standard. *American Journal of Epidemiology*, Vol. 141, No. 3, pp. 263–272, 1995.
- [42] D. A. Berry, K. M. Chaloner, and J.K. Geweke. *Bayesian analysis in statistics and econometrics: Essays in honor of Arnold Zellner*. John Wiley & Sons, 1996.
- [43] T. B. Swartz, Y. Haitovsky, A. Vexler, and T. Y. Yang. Bayesian Identifiability and Misclassification in Multinomial Data. *Canadian Journal of Statistics*, Vol. 32, No. 3, pp. 285–302, 2004.
- [44] U. Rebbapragada and C. E. Brodley. Class Noise Mitigation Through Instance Weighting. In *Proceedings of 2007 European Conference on Machine Learning (ECML 2007)*, pp. 708–715, 2007.
- [45] X. Zhou, P. L. K. Ding, and B. Li. Improving Robustness of Random Forest Under Label Noise. In *Proceedings of 2019 IEEE Winter Conference on Applications of Computer Vision (WACV 2019)*, pp. 950–958, 2019.
- [46] A. Ganapathiraju and J. Picone. Support Vector Machines for Automatic Data

- Cleanup. In *Proceedings of Sixth International Conference on Spoken Language Processing*, Vol. 4, pp. 210–213, 2000.
- [47] A. Kowalczyk, A. J. Smola, and R. C. Williamson. Kernel Machines and Boolean Functions. In *Proceedings of 14th International Conference on Neural Information Processing Systems: Natural and Synthetic*, pp. 439–446, 2001.
- [48] Y. Li and P. M. Long. The Relaxed Online Maximum Margin Algorithm. In *Proceedings of 12th International Conference on Neural Information Processing Systems*, pp. 498–504, 1999.
- [49] S. Nazmi and A. Homaifar. Possibility Rule-Based Classification Using Function Approximation. In *Proceedings of 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2018)*, pp. 668–674, 2018.
- [50] O. N. Granichin. Estimating the Parameters of Linear Regression in an Arbitrary Noise. *Automation and Remote Control*, Vol. 63, No. 1, pp. 25–35, 2002.
- [51] O. N. Granichin. Randomized algorithms of stochastic approximation under arbitrary noise. *Automation and Remote Control*, Vol. 63, No. 2, pp. 209–219, 2002.
- [52] M. Gershoff and S. Schulenburg. Collective Behavior Based Hierarchical XCS. In *Proceedings of 9th Annual Conference Companion on Genetic and Evolutionary Computation (GECCO 2007)*, pp. 2695–2700, 2007.
- [53] M. Abedini and M. Kirley. CoXCS: A Coevolutionary Learning Classifier Based on Feature Space Partitioning. In *Proceedings of Advances in Artificial Intelligence (AI 2009)*, pp. 360–369, 2009.
- [54] F. Zhu and S. Guan. Cooperative Co-evolution of GA-based Classifiers based on Input Decomposition. *Engineering Applications of Artificial Intelligence*, Vol. 21, No. 8, pp. 1360 – 1369, 2008.
- [55] E. Debie, K. Shafi, C. Lokan, and K. Merrick. Performance Analysis of Rough Set Ensemble of Learning Classifier Systems with Differential Evolution based Rule Discovery. *Evolutionary Intelligence*, Vol. 6, No. 2, pp. 109–126, 2013.
- [56] Y. Liu, W. N. Browne, and B. Xue. Hierarchical Learning Classifier Systems for Polymorphism in Heterogeneous Niches. In *Proceedings of Advances in Artificial Intelligence (AI 2018)*, pp. 397–409, 2018.
- [57] Y. Liu, W. N. Browne, and B. Xue. Adapting Bagging and Boosting to Learning Classifier Systems. In *Proceedings of Applications of Evolutionary Computation*, pp. 405–420, 2018.
- [58] Y. Gao, J. Z. Huang, H. Rong, and D. Gu. LCSE: Learning Classifier System

- Ensemble for Incremental Medical Instances. In *Proceedings of 2007 International workshop on Learning Classifier Systems*, pp. 93–103, 2007.
- [59] Y. Gao, J. Z. Huang, and L. Wu. Learning Classifier System Ensemble and Compact Rule Set. *Connection Science*, Vol. 19, No. 4, pp. 321–337, 2007.
- [60] Y. Gao, J. Z. Huang, H. Rong, and D. Gu. Learning Classifier System Ensemble for Data Mining. In *Proceedings of 7th Annual Workshop on Genetic and Evolutionary Computation (GECCO 2005)*, pp. 63–66, 2005.
- [61] P. Rojanavasu, H. H. Dam, H. A. Abbass, C. Lokan, and O. Pinngern. A Self-Organized, Distributed, and Adaptive Rule-Based Induction System. *IEEE Transactions on Neural Networks*, Vol. 20, No. 3, pp. 446–459, 2009.
- [62] P. L. Lanzi, D. Loiacono, and M. Zanini. Evolving Classifier Ensembles with Voting Predictors. In *Proceedings of 2008 IEEE Congress on Evolutionary Computation (CEC 2008)*, pp. 3760–3767, 2008.
- [63] P. L. Lanzi, D. Loiacono, and M. Zanini. Evolving Classifiers Ensembles with Heterogeneous Predictors. In *Proceedings of 2008 International workshop on Learning Classifier Systems*, pp. 218–234, 2008.
- [64] L. Guo and L. Ljung. Exponential Stability of General Tracking Algorithms. *IEEE Transactions on Automatic Control*, Vol. 40, No. 8, pp. 1376–1387, 1995.
- [65] L. Guo. Stability of Recursive Stochastic Tracking Algorithms. *Society for Industrial and Applied Mathematics Journal on Control and Optimization*, Vol. 32, No. 5, pp. 1195–1225, 1994.
- [66] L. Ljung and P. Priouret. A Result on the Mean Square Error Obtained Using General Tracking Algorithms. *International Journal of Adaptive Control and Signal Processing*, Vol. 5, No. 4, pp. 231–248, 1991.
- [67] V. Solo and X. Kong. *Adaptive Signal Processing Algorithms: Stability and Performance*. Prentice-Hall, Inc., 1994.
- [68] R. S. Sutton. Learning to Predict by the Methods of Temporal Differences. *Machine Learning*, Vol. 3, No. 1, pp. 9–44, 1988.
- [69] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Longman Publishing Co., Inc., 1st edition, 1989.
- [70] J. H. Holland and J. S. Reitman. Cognitive Systems Based on Adaptive Algorithms. In D. A. Waterman and F. Hayes-Roth, editors, *Pattern-directed Inference Systems*, pp. 313–329. 1978.
- [71] J. H. Holland. The Possibilities of General-purpose Learning Algorithms Applied to Parallel Rule-based Systems. *Machine Learning, An Artificial Intelligence*

- Approach*, Vol. 2, pp. 593–623, 1986.
- [72] S. F. Smith. *A Learning System Based on Genetic Adaptive Algorithms*. PhD thesis, University of Pittsburgh, 1980.
- [73] E. Bernadó-Mansilla and J. M. Garrell-Guiu. Accuracy-Based Learning Classifier Systems: Models, Analysis and Applications to Classification Tasks. *Evolutionary Computation*, Vol. 11, No. 3, pp. 209–238, 2003.
- [74] S. W. Wilson. ZCS: A Zeroth Level Classifier System. *Evolutionary computation*, Vol. 2, No. 1, pp. 1–18, 1994.
- [75] R. J. Urbanowicz and W. N. Browne. *Introduction to Learning Classifier System*. Springer, 1st. edition, 2017.
- [76] K. A. D. Jong, W. A. Spears, and D. F. Gordon. Using genetic algorithms for concept learning. *Machine Learning*, Vol. 13, No. 2, pp. 161–188, 1993.
- [77] J. Bacardit. *Pittsburgh Genetics-based Machine Learning in the Data Mining Era: Representations, Generalization, and Run-time*. PhD thesis, Ramon Llull University, 2004.
- [78] J. Bacardit and N. Krasnogor. A Mixed Discrete-continuous Attribute List Representation for Large Scale Classification Domains. In *Proceedings of 11th Annual Conference on Genetic and Evolutionary Computation (GECCO 2009)*, pp. 1155–1162, 2009.
- [79] T. Kovacs. *Strength or Accuracy: Credit Assignment in Learning Classifier Systems*. SpringerVerlag, 2003.
- [80] M. V. Butz, P. L. Lanzi, and S. W. Wilson. Function Approximation With XCS: Hyperellipsoidal Conditions, Recursive Least Squares, and Compaction. *IEEE Transactions on Evolutionary Computation*, Vol. 12, No. 3, pp. 355–376, 2008.
- [81] A. Stein, S. Menssen, and J. Hahner. What About Interpolation?: A Radial Basis Function Approach to Classifier Prediction Modeling in XCSF. In *Proceedings of Genetic and Evolutionary Computation Conference (GECCO 2018)*, pp. 537–544, 2018.
- [82] X. Cheng, W. N. Browne, and M. Zhang. Decomposition Based Multi-Objective Evolutionary Algorithm in XCS for Multi-Objective Reinforcement Learning. In *Proceedings of 2018 IEEE Congress on Evolutionary Computation (CEC 2018)*, pp. 1–8, 2018.
- [83] X. Cheng, G. Chen, and M. Zhang. An XCS-based algorithm for multi-objective reinforcement learning. In *Proceedings of 2016 IEEE Congress on Evolutionary Computation (CEC 2016)*, pp. 4007–4014, 2016.

- [84] F. Rudzisk. A Multi-objective Genetic Optimization of Interpretability-oriented Fuzzy Rule-based Classifiers. *Applied Soft Computing*, Vol. 38, pp. 118–133, 2016.
- [85] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT press, 1998.
- [86] M. V. Butz and S. W. Wilson. An Algorithmic Description of XCS. *Soft Computing*, Vol. 6, No. 3-4, pp. 144–153, 2002.
- [87] J. R. Koza. A Genetic Approach to the Truck Backer Upper Problem and the Inter-twined Spiral Problem. In *International Joint Conference on Neural Networks*, Vol. 4, pp. 310–318, 1992.
- [88] P. W. Dixon, D. W. Corne, and M. J. Oates. A Ruleset Reduction Algorithm for the XCS Learning Classifier System. In *Proceedings of 2003 International workshop on Learning Classifier Systems*, pp. 20–29, 2003.
- [89] M. Nakata, W. Browne, T. Hamagami, and K. Takadama. Theoretical XCS Parameter Settings of Learning Accurate Classifiers. In *Proceedings of 2017 Genetic and Evolutionary Computation Conference (GECCO 2017)*, pp. 473–480, 2017.
- [90] D. Dua and C. Graff. UCI machine learning repository. <http://archive.ics.uci.edu/ml>, 2017.

関連論文の印刷公表の方法およびその時期

1. 全著者名 : Takato Tatsumi, Hiroyuki Sato, and Keiki Takadama
論文題目 : Learning Classifier System Based on Mean of Reward
印刷公表の方法及びその時期 : *Journal of Advanced Computational Intelligence and Intelligent Informatics Vol. 21, No. 5, pp. 895-906, September 2017*
(第7章に関連)
2. 全著者名 : 辰巳 嵩豊, 小峯 嵩裕, 中田 雅也, 佐藤 寛之, 高玉 圭樹
論文題目 : 許容誤差を自己適応可能な学習分類子システム
印刷公表の方法及びその時期 : *進化計算学会論文誌 Vol. 6, No. 2, pp. 90-103, 2015年11月*
(第7.2節に関連)
3. 全著者名 : Takato Tatsumi and Keiki Takadama
論文題目 : XCS-CR for handling input, output, and reward noise
印刷公表の方法及びその時期 : *International Workshop on Learning Classifier Systems in Genetic and Evolutionary Computation Conference (GECCO), pp. 1303-1311, July 2019*
(第7章に関連)
4. 全著者名 : Takato Tatsumi and Keiki Takadama
論文題目 : Comparison of Statistical Table- and Non-Statistical Table-based XCS in Noisy Environments
印刷公表の方法及びその時期 : *IEEE Congress on Evolutionary Computation (CEC), pp. 1875-1882, June 2019*
(第9章に関連)
5. 全著者名 : Takato Tatsumi, Tim Kovacs and Keiki Takadama
論文題目 : XCS-CR: Determining Accuracy of Classifier by its Collective Reward in Action Set toward Environment with Action Noise

- 印刷公表の方法及びその時期 : *International Workshop on Learning Classifier Systems in Genetic and Evolutionary Computation Conference (GECCO)*, pp. 1457-1464, July 2018
(第 5 章に関連)
6. 全著者名 : Takato Tatsumi, Hiroyuki Sato and Keiki Takadama
論文題目 : Automatic Adjustment of Selection Pressure based on Range of Reward in Learning Classifier System
印刷公表の方法及びその時期 : *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 505-512, July 2017
(第 6 章に関連)
7. 全著者名 : Takato Tatsumi, Hiroyuki Sato, Tim Kovacs and Keiki Takadama
論文題目 : Applying Variance-based Learning Classifier System without Convergence of Reward Estimation into Various Reward Distribution
印刷公表の方法及びその時期 : *IEEE Congress on Evolutionary Computation (CEC)*, pp. 2630-2637, June 2017
(第 7 章に関連)
8. 全著者名 : Takato Tatsumi, Takahiro Komine, Masaya Nakata, Hiroyuki Sato, Tim Kovacs and Keiki Takadama
論文題目 : Variance-based Learning Classifier System without Convergence of Reward Estimation
印刷公表の方法及びその時期 : *Genetic and Evolutionary Computation Conference (GECCO)*, pp. 67-68, July 2016
(第 7 章に関連)
9. 全著者名 : Takato Tatsumi, Takahiro Komine, Hiroyuki Sato and Keiki Takadama
論文題目 : Handling Different Level of Unstable Reward Environment Through an Estimation of Reward Distribution in XCS
印刷公表の方法及びその時期 : *IEEE Congress on Evolutionary Computation (CEC)*, pp. 2973-2980, May 2015
(第 7.2 節に関連)
10. 全著者名 : 辰巳嵩豊, 小峯嵩裕, 中田雅也, Tim Kovacs, 佐藤寛之, 高玉圭樹
論文題目 : 報酬分散の収束に依らない分散に基づく学習分類子システム
印刷公表の方法及びその時期 : *進化計算学会進化計算シンポジウム 2015*, P3-06, 2015 年 12 月
(第 7 章に関連)

11. 全著者名：辰巳嵩豊, 小峯嵩裕, 佐藤寛之, 高玉圭樹

論文題目：不安定報酬環境下における学習分類子システム

印刷公表の方法及びその時期：進化計算学会進化計算シンポジウム 2014, P2-01,

2014年12月

(第7.2節に関連)