# Learning Classifier Systems

Andrew Cannon
Angeline Honggowarsito

# Contents

# Rule-Based Agents

- Represented by rule-based agents.
  - Agents - Single Components
- IF condition THEN action
- Use system’s environment information to make decision

# Metaphor

- Two biological Metaphors:
  - Evolution
  - Learning
- Genetic Algorithm & Learning Mechanism
- Environment of the system
- Example
  - Robots navigating maze environment

# The Driving Mechanism

- Discovery - The Genetic Algorithm
  - Rule Discovery
  - Apply Genetic Algorithm
    - The fitness function quantifies the optimality of a given rule
  - Classification Accuracy most widely used as metric of fitness

# The Driving Mechanism

- Learning
  - “The improvement of performance in some environment through the acquisition of knowledge resulting from experience in that environment.”
  - Each classifier has one or more parameters
  - Iteratively update the parameters

# Learning

- Purposes:
  - Identify useful classifiers
  - Discovery of better rules
- Different problem domains require different styles of learning.
- Learning based on the information provided
  - Batch Learning
    - Training instances presented simultaneously.
    - End result: rule set that does not change with respect to time.

# Learning

- Incremental learning
  - One training instances at a time
  - End result:
    - Rule set that changes continuously
- Learning based on type of feedback
  - Supervised Learning
  - Reinforcement Learning

# Minimal Classifier System

- Basic LCS Implementation
- Developed by Larry Bull
- Advancing LCS theory
- Designed to understand more complex implementations, instead of solving real world problems.

# Minimal Classifier System

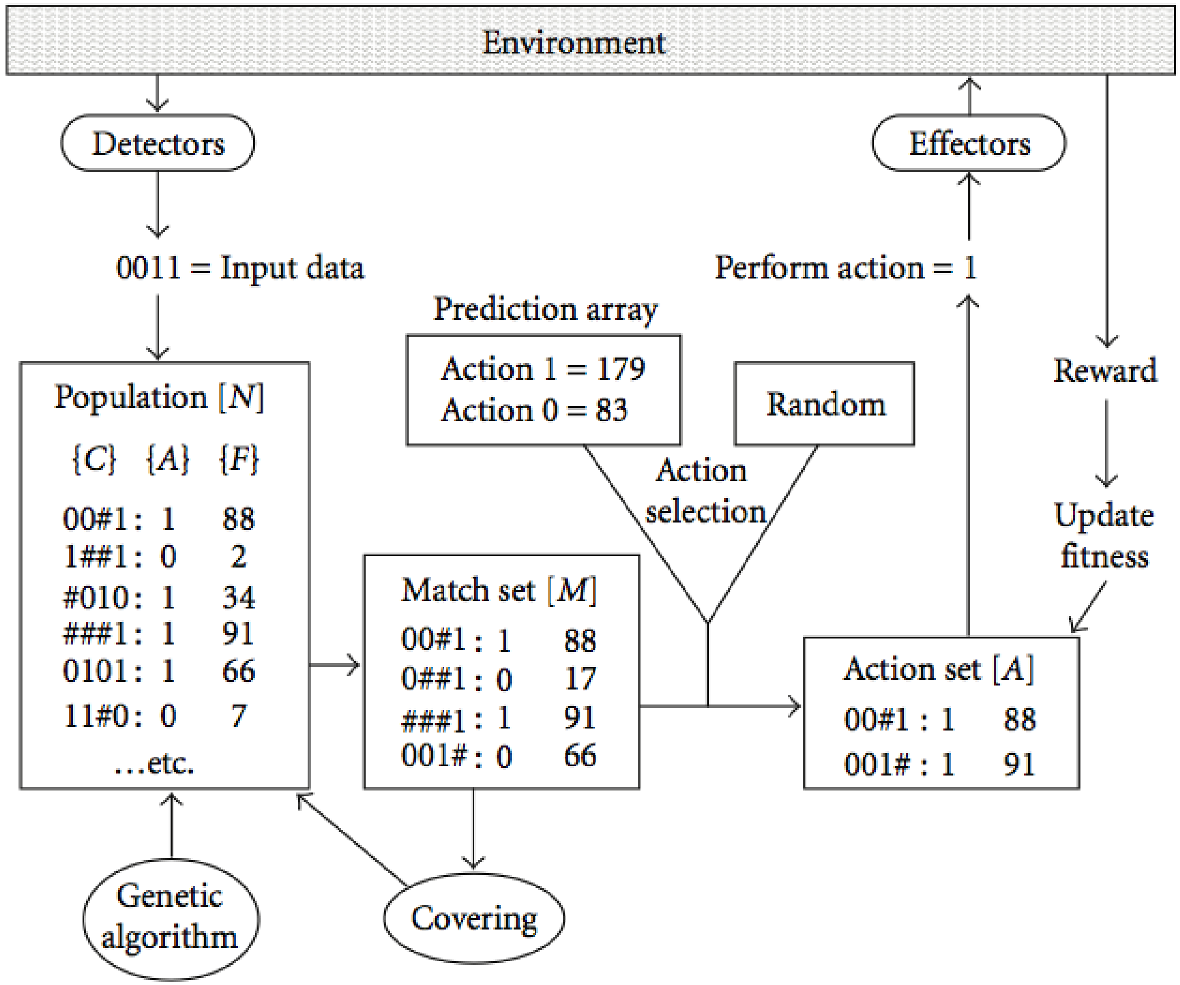


FIGURE 3: MCS algorithm—an example iteration.

# Minimal Classifier System

- Input Data
  - 4 Digit binary Number
- Learning Iteratively, one instance at a time
- Population [N]
  - Condition {C}
  - Action {A}
  - Fitness Parameter {F}
- Population is randomly initialized

# Population

- Condition
  - String of “0, 1, #”
  - 00#1 matches 0011 or 0001
- Action
  - Which action is possible (0 or 1)
- Fitness Parameter
  - How good is the classifier

0011

| C | A | F |
|---|---|---|
| 00#1 | 1 | 88 |
| 0##1 | 0 | 17 |
| #010 | 1 | 34 |
| 001# | 1 | 91 |
| 0#11 | 0 | 66 |
| 11#0 | 0 | 7 |

# Match Set

- Population Scanned
- Match Set : List of rules whose condition matches the input string at each position
- Input = 0011

| C | A | F |
|---|---|---|
| 00#1 | 1 | 88 |
| 0##1 | 0 | 17 |
| #010 | 1 | 34 |
| 001# | 1 | 91 |
| 0#11 | 0 | 66 |
| 11#0 | 0 | 7 |

Population

| C | A | F |
|---|---|---|
| 00#1 | 1 | 88 |
| 0##1 | 0 | 17 |
| 001# | 1 | 91 |
| 0#11 | 0 | 66 |

Match set

# Action Set

- Action Set established using explore/exploit scheme by alternating between:
  - Select action found in M (Explore)
  - Select deterministically with prediction array

Match set

| C | A | F |
|---|---|---|
| 00#1 | 1 | 88 |
| 0##1 | 0 | 17 |
| 001# | 1 | 91 |
| 001# | 0 | 66 |

Action set

| | | |
|---|---|---|
| 00#1 | 1 | 88 |
| 001# | 1 | 91 |

Prediction array

| | |
|---|---|
| Action 1 | 179 |
| Action 0 | 83 |

# Minimal Classifier System

- Prediction array: List of prediction values calculated for each action
- Prediction value: sum of fitness values found in the subset of M advocating the same action
- Learning starts when the reward is received

# Contents

- Categories of LCS
  - Accuracy based (XCS)

# Accuracy based (XCS)

- Extended classifier system
- Most studied and widely used family of LCS

- Each rule predicts a particular reward (and error)
- Each rule has a particular fitness
- Retain rules that predict lower rewards as long as those predictions are accurate

# Accuracy based (XCS)

- Population of rules (initially empty but bounded to some size P) specifying actions in response to conditions
- Match set formed in response to stimuli from environment
- Action selected from match set
  - Highest fitness
  - Roulette wheel selection
  - Alternation between exploration and exploitation
- Rules advocating the same action form the action set

# Accuracy based (XCS)

- Receive a reward r from the environment for executing the specified action
- Update the predicted reward for each rule in the action set
  - $p \leftarrow p + \beta\,(r-p)$
- Update the predicted error for each rule in the action set
  - $\varepsilon \leftarrow \varepsilon + \beta\,(|r-p| - \varepsilon)$

  - $\beta$ = estimation rate

# Accuracy based (XCS)

- If $\varepsilon < \varepsilon_0$, set prediction accuracy k=1
- Otherwise, set prediction accuracy
  - $k = \alpha(\varepsilon_0/\varepsilon)^v$ for some $\alpha, v > 0$
- Calculate relative prediction accuracy
  - k' = k(rule) / (sum of k for all rules in action set)
- Update the fitness of each rule
  - $f \leftarrow f + \beta\,(k' - f)$

  - $\alpha$ = learning rate
  - $\beta$ = estimation rate

# Accuracy based (XCS)

- Run genetic algorithm to introduce diversity and increase fitness of population
- Run every $\theta_{GA}$ time steps
- Run on members of action set (rather than members of global population)
- Favours accurate classifiers
- Two parents selected via roulette wheel selection (based on fitness) produce two offspring via mutation and crossover

# Accuracy based (XCS)

- Covering operator adds new rules when no rules match the current environmental condition (possibly generalised)
- If the population exceeds its bounded size, the requisite number of rules are deleted via roulette wheel selection based on the average size of the action sets containing each rule

# Accuracy based (XCS)

- Sometimes, when a new rule is added, it is checked whether or not a more general rule already exists – if it does, another copy of the more general rule is added instead of the more specific rule
- Favours generalisation
- Computationally expensive

# Accuracy based (XCS)

- Suggested parameters (Butz and Wilson 2002)
  - Maximum population size P=800 or P=2000
  - Learning rate $\alpha = 0.1$
  - Estimation rate $\beta = 0.2$
  - Genetic algorithm run every $\theta_{GA} = 25$ time steps