GP-COACH:

A Novel Genetic Cooperative-Competitive Fuzzy Rule Based Learning Method using Genetic Programming for High-Dimensional Problems

Francisco José Berlanga

Dept. of Computer Science, University of Jaén, Spain

In collaboration with

María José del Jesus and Francisco Herrera
Index

- Introduction
- GP-COACH Algorithm
- Experimental Framework and Results
- Conclusions
Introduction

Fuzzy Rule-Based Classification System (FRBCS)

Knowledge Base

Rule Base  Data Base

Problem Instance  Fuzzy Inference  Class
**Fuzzy Rule-Based Classification System (FRBCS)**

R1: If $X_1$ is High and $X_2$ is Low then Class is A
R2: If $X_1$ is Low and $X_2$ is Low then Class is B
Learning of FRBCSs for high-dimensional problems

- Number of training examples
- Number of input features
- Number of rules
- Interpretability
Learning of FRBCSs for high-dimensional problems

1. Carry out a feature selection \((a \text{ priori or embedded})\) process:
   - Using information measures
   - Using GFSs
   - Neuro-fuzzy systems
   - Other different techniques

2. Compact and reduce a previously learned rule set in a postprocessing stage:
   - Using GFSs
   - Orthogonal transformation methods
   - Similarity measures
FRBSs learning. Carrying out a feature selection process (I):

- Using information measures
  - Q. Shen and R. Jensen, Selecting informative features with fuzzy-rough sets and its application for complex systems monitoring, Pattern Recognition, vol. 37, no. 7, pp. 1351–1363, 2004

- Using GFSs
FRBSs learning. Carrying out a feature selection process (II):

- **Neuro-fuzzy systems**

- **Other different techniques**
FRBSs learning. Compacting and reducing a previous rule set (I):

- Using GFSs

- Orthogonal transformation methods
Introduction

FRBSs learning. Compacting and reducing a previous rule set (II):

- Similarity measures
Introduction

Genetic Programming (GP)

S → x_1 | x_2 | N | + S S | * S S
N → ℝ

Chromosome: + * x_2 x_1 * 0.8 * x_1 x_1
Expression: x_2 * x_1 + 0.8 * x_1^2
FRBSs learning. GP-based approaches (I):

FRBSs learning. GP-based approaches (II):


Introduction

GP-COACH Algorithm

Experimental Framework and Results

Conclusions
Main Features

A. Uses a context-free grammar that allow the learning of DNF fuzzy rules and the absence of some of the input features

B. Follows the genetic cooperative-competitive learning (GCCL) approach which encodes a single rule per individual in the population and the complete RB is formed by the whole population

- **Local fitness function** ➔ evaluates individuals (rules)
- **Global fitness function** ➔ evaluates one population (rule set)

C. Includes a mechanism to maintain the diversity in the population that eliminates irrelevant rules: *Token Competition*

D. Uses a two level hierarchical inference process
If $X_1$ is L then Class is $C_1$

Context-free Grammar

- start $\rightarrow$ [If], antec, [then], conseq, [.]
- antec $\rightarrow$ descriptor1, [and], descriptor2.
- descriptor1 $\rightarrow$ [any].
- descriptor1 $\rightarrow$ [$X_1$ is] label.
- descriptor2 $\rightarrow$ [any].
- descriptor2 $\rightarrow$ [$X_2$ is] label.
- label $\rightarrow$ {member(?a, [L, M, H, L or M, L or H, M or H, L or M or H])}, [?a].
- conseq $\rightarrow$ [Class is] descriptorClass
- descriptorClass $\rightarrow$ {member(?a, [C_1, C_2, C_3])}, [?a].
GP-COACH Algorithm

Local Fitness Function

\[
\text{raw_fitness} = (\alpha \times \text{Confidence}) + ((1 - \alpha) \times \text{Support})
\]

Confidence:

\[
\text{Confidence} = \frac{\mu_{tp}}{\left( \mu_{tp} + \mu_{fp} \right)}
\]

Support:

\[
\text{Support} = \frac{\mu_{tp}}{N_{C^k}^{\text{k}}}
\]

- \(\mu_{tp}\) ➔ is the sum of the matching degree for true positives
- \(\mu_{fp}\) ➔ is the sum of the matching degree for false positives
- \(N_{C^k}^{\text{k}}\) ➔ is the number of examples belonging to the same class that the one indicated in the consequent of the rule
Global Fitness Function

\[
\text{Global\_fitness} = (w_1 \times \%Tra) + (w_2 \times \#V) + (w_3 \times \#C) + (w_4 \times \#R)
\]

\[
\#A = 1 - \#A \quad \text{and} \quad \#A = \{\#V, \#C, \#R\}
\]

\%Tra \rightarrow \text{the normalized value of the correct percentage on training examples}

\#V \rightarrow \text{the normalized value of the number of variables per individual (rule) in the population}

\#C \rightarrow \text{the normalized value of the number of labels (or conditions) per individual}

\#R \rightarrow \text{the normalized value of the number of rules in the population}
Token Competition

Modified_fitness = raw_fitness \times \text{count} / \text{ideal}
GP-COACH Algorithm

Two level hierarchical inference process

1. **Primary rules:** Strong and general rules created by the genetic operators. They cover most of the examples

2. **Secondary rules:** Weaker and more specific rules, generated after the Token Competition procedure in order to increase the diversity in the population

- Secondary rules are only taken into account if there not exist any primary rule matching with some of the examples

- It is also possible that GP-COACH learns rule sets having no secondary rules, because their primary rules are strong enough to cover all the given examples
Genetic Operators

1. **Crossover**: Produces one child from two parents. A part in the first parent is randomly selected and exchanged by another part, randomly selected, in the second one.

2. **Mutation**: Operates on label sets level
   - Add a label to the set
   - Remove a label from the set
   - Exchange a label in the set by another one not included in it

3. **Insertion**: Adds a variable to a rule

4. **Dropping Condition**: Remove a variable from a rule
GP-COACH Algorithm

Pseudocode

```pseudocode
Initialize(Initial_pop)
For each Initial_pop[i] do
    Initial_pop[i].fitness <- Evaluate(Initial_pop[i])
End for
Copy Initial_pop to Best_pop
Best_pop.global_fitness <- Global_score(Best_pop)
Copy Initial_pop to Current_pop
While (not termination-condition) do
    Offspring_pop = ∅
    While (Size(Offspring_pop) ≠ Size(Current_pop)) do
        Parent <- Binary_tournament(Current_pop)
        Child <- Genetic_operator(Parent)
        Evaluate(Child)
        Add Child to Offspring_pop
    End while
    Joint_pop <- Current_pop U Offspring_pop
    New_current_pop <- Token_competition(Joint_pop)
    New_current_pop.global_fitness <- Global_score(New_current_pop)
    If New_current_pop.global_fitness > Best_pop.global_fitness then
        Best_pop.global_fitness <- New_current_pop.global_fitness
        Copy New_current_pop to Best_pop
    End if
    Copy New_current_pop to Current_pop
End while
Return(Best_pop)
```
Introduction

GP-COACH Algorithm

Experimental Framework and Results

Conclusions
### Experimental Framework

#### Datasets

<table>
<thead>
<tr>
<th>Name</th>
<th>N. Instances</th>
<th>N. Features</th>
<th>N. Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ionosphere</td>
<td>351</td>
<td>34</td>
<td>2</td>
</tr>
<tr>
<td>Pen-based</td>
<td>10992</td>
<td>16</td>
<td>10</td>
</tr>
<tr>
<td>Spambase</td>
<td>4597</td>
<td>57</td>
<td>2</td>
</tr>
<tr>
<td>Thyroid</td>
<td>7200</td>
<td>21</td>
<td>3</td>
</tr>
<tr>
<td>Vehicle</td>
<td>846</td>
<td>18</td>
<td>4</td>
</tr>
<tr>
<td>Wdbc</td>
<td>569</td>
<td>30</td>
<td>2</td>
</tr>
</tbody>
</table>
Experimental Framework

Comparative Methods

1. **2SLAVE:**
   - GA-based method (IRL approach) for learning DNF fuzzy rules
   - Feature selection process during the learning phase

2. **Tsakonas:**
   - GP-based method for learning FRBCSs
   - Context-free grammar to generate complete rule sets per individual in the population (Pittsburgh approach)

3. **FRBCS_GP:**
   - Does not use a two level hierarchical inference process
   - Population size does not change during the evolutionary process
   - Does not use any kind of global fitness score to keep the best evolved population
   - Uses a crisp fitness function, which uses the number of positive and negative examples
   - Uses a ranking selection scheme to select parents from the population, and different genetic operators to generate new children
### Results (I)

**Ionosphere**

<table>
<thead>
<tr>
<th>Method</th>
<th>#R</th>
<th>#V</th>
<th>#C</th>
<th>% Tra</th>
<th>% Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>2SLAVE(_1)</td>
<td>6</td>
<td>12.73</td>
<td>25.16</td>
<td>82.73</td>
<td>81.88</td>
</tr>
<tr>
<td>2SLAVE(_2)</td>
<td>6</td>
<td>12.73</td>
<td>25.16</td>
<td>85.46</td>
<td>84.14</td>
</tr>
<tr>
<td>Tsakonas(_1)</td>
<td>35.4</td>
<td>2.44</td>
<td><strong>2.92</strong></td>
<td>40.37</td>
<td>39.69</td>
</tr>
<tr>
<td>Tsakonas(_2)</td>
<td>35.4</td>
<td>2.44</td>
<td><strong>2.92</strong></td>
<td>51.05</td>
<td>52.42</td>
</tr>
<tr>
<td>FRBCS(_{GP})_1</td>
<td>14.03</td>
<td>2.41</td>
<td>6.63</td>
<td>83.65</td>
<td>82.81</td>
</tr>
<tr>
<td>FRBCS(_{GP})_2</td>
<td>14.03</td>
<td>2.41</td>
<td>6.63</td>
<td>82.86</td>
<td>81.48</td>
</tr>
<tr>
<td>GP-COACH(_1)</td>
<td><strong>5.9</strong></td>
<td>2.14</td>
<td>5.20</td>
<td>93.61</td>
<td>92.14</td>
</tr>
<tr>
<td>GP-COACH(_2)</td>
<td>5.97</td>
<td><strong>2.06</strong></td>
<td>5.07</td>
<td><strong>94.26</strong></td>
<td><strong>92.31</strong></td>
</tr>
</tbody>
</table>

**Pen-based**

<table>
<thead>
<tr>
<th>Method</th>
<th>#R</th>
<th>#V</th>
<th>#C</th>
<th>% Tra</th>
<th>% Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

• **#R**: Average rule number  
• **#V**: Average antecedent variables per rule  
• **#C**: Average conditions number per rule  
• **% Tra**: Correct percentage with training  
• **% Test**: Correct percentage with test

**Subscripts**: Related to the fuzzy reasoning method (FRM) used:
- **1**: Classical FRM (max-min)  
- **2**: Normalized sum FRM
Results (II)

<table>
<thead>
<tr>
<th>Method</th>
<th>#R</th>
<th>#V</th>
<th>#C</th>
<th>% Tra</th>
<th>% Test</th>
<th>#R</th>
<th>#V</th>
<th>#C</th>
<th>% Tra</th>
<th>% Test</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2SLAVE₁</td>
<td>7.9</td>
<td>22.25</td>
<td>49.19</td>
<td>73.66</td>
<td>73.74</td>
<td>3.7</td>
<td>9.23</td>
<td>20.52</td>
<td>92.81</td>
<td>92.87</td>
</tr>
<tr>
<td>2SLAVE₂</td>
<td>7.9</td>
<td>22.25</td>
<td>49.19</td>
<td>69.87</td>
<td>70.14</td>
<td>3.7</td>
<td>9.23</td>
<td>20.52</td>
<td>92.85</td>
<td>92.88</td>
</tr>
<tr>
<td>Tsakonas₁</td>
<td>27.43</td>
<td>3.03</td>
<td>3.68</td>
<td>39.63</td>
<td>39.93</td>
<td>27.27</td>
<td>2.02</td>
<td>2.56</td>
<td>80.31</td>
<td>80.38</td>
</tr>
<tr>
<td>Tsakonas₂</td>
<td>27.43</td>
<td>3.03</td>
<td>3.68</td>
<td>49.63</td>
<td>49.49</td>
<td>27.27</td>
<td>2.02</td>
<td>2.56</td>
<td>43.23</td>
<td>43.77</td>
</tr>
<tr>
<td>FRBCS_GP₁</td>
<td>43.93</td>
<td>13.46</td>
<td>31.57</td>
<td>73.72</td>
<td>73.50</td>
<td>29.33</td>
<td>12.01</td>
<td>34.16</td>
<td>92.90</td>
<td>92.89</td>
</tr>
<tr>
<td>FRBCS_GP₂</td>
<td>43.93</td>
<td>13.46</td>
<td>31.57</td>
<td>75.03</td>
<td>74.55</td>
<td>29.33</td>
<td>12.01</td>
<td>34.16</td>
<td>92.82</td>
<td>92.82</td>
</tr>
<tr>
<td>GP-COACH₁</td>
<td>14</td>
<td>5.27</td>
<td>9.63</td>
<td>77.29</td>
<td>76.62</td>
<td>6.13</td>
<td>1.59</td>
<td>2.73</td>
<td>93.27</td>
<td>93.26</td>
</tr>
<tr>
<td>GP-COACH₂</td>
<td>10.77</td>
<td>3.93</td>
<td>7.96</td>
<td>83.70</td>
<td>83.59</td>
<td>6.3</td>
<td>1.60</td>
<td>2.80</td>
<td>93.26</td>
<td>93.25</td>
</tr>
</tbody>
</table>

**Subscripts**: Related to the fuzzy reasoning method (FRM) used:
- 1: Classical FRM (max-min)
- 2: Normalized sum FRM

- **#R**: Average rule number
- **#V**: Average antecedent variables per rule
- **#C**: Average conditions number per rule
- **%Tra**: Correct percentage with training
- **%Test**: Correct percentage with test
## Results (III)

<table>
<thead>
<tr>
<th>Method</th>
<th>#R</th>
<th>#V</th>
<th>#C</th>
<th>% Tra</th>
<th>% Test</th>
<th>#R</th>
<th>#V</th>
<th>#C</th>
<th>% Tra</th>
<th>% Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>2SLAVE&lt;sub&gt;1&lt;/sub&gt;</td>
<td>11.73</td>
<td>11.07</td>
<td>24.96</td>
<td>49.11</td>
<td>47.30</td>
<td>5.2</td>
<td>10.31</td>
<td>20.22</td>
<td>92.59</td>
<td>92.74</td>
</tr>
<tr>
<td>2SLAVE&lt;sub&gt;2&lt;/sub&gt;</td>
<td>11.73</td>
<td>11.07</td>
<td>24.96</td>
<td>49.21</td>
<td>47.38</td>
<td>5.2</td>
<td>10.31</td>
<td>20.22</td>
<td>92.43</td>
<td>92.33</td>
</tr>
<tr>
<td>Tsakonas&lt;sub&gt;1&lt;/sub&gt;</td>
<td>37.73</td>
<td>1.89</td>
<td>2.48</td>
<td>26.15</td>
<td>26.09</td>
<td>27.8</td>
<td>2.76</td>
<td>3.69</td>
<td>51.89</td>
<td>53.28</td>
</tr>
<tr>
<td>Tsakonas&lt;sub&gt;2&lt;/sub&gt;</td>
<td>37.73</td>
<td>1.89</td>
<td>2.48</td>
<td>26.32</td>
<td>27.00</td>
<td>27.8</td>
<td>2.76</td>
<td>3.69</td>
<td>52.39</td>
<td>52.71</td>
</tr>
<tr>
<td>FRBCS_GP&lt;sub&gt;1&lt;/sub&gt;</td>
<td>55.23</td>
<td>6.10</td>
<td>15.84</td>
<td>53.13</td>
<td>52.60</td>
<td>14.9</td>
<td>3.59</td>
<td>8.85</td>
<td>94.48</td>
<td>93.49</td>
</tr>
<tr>
<td>FRBCS_GP&lt;sub&gt;2&lt;/sub&gt;</td>
<td>55.23</td>
<td>6.10</td>
<td>15.84</td>
<td>56.47</td>
<td>55.05</td>
<td>14.9</td>
<td>3.59</td>
<td>8.85</td>
<td>95.27</td>
<td>94.37</td>
</tr>
<tr>
<td>GP-COACH&lt;sub&gt;1&lt;/sub&gt;</td>
<td>34.03</td>
<td>4.08</td>
<td>9.17</td>
<td>54.81</td>
<td>53.55</td>
<td>4.97</td>
<td>1.11</td>
<td>2.61</td>
<td>94.19</td>
<td>92.32</td>
</tr>
<tr>
<td>GP-COACH&lt;sub&gt;2&lt;/sub&gt;</td>
<td>33.53</td>
<td>4.22</td>
<td>9.57</td>
<td>58.33</td>
<td>55.09</td>
<td>4.8</td>
<td>1.09</td>
<td>2.67</td>
<td>95.03</td>
<td>93.38</td>
</tr>
</tbody>
</table>

- **#R**: Average rule number
- **#V**: Average antecedent variables per rule
- **#C**: Average conditions number per rule
- **% Tra**: Correct percentage with training
- **% Test**: Correct percentage with test

**Subscripts**: Related to the fuzzy reasoning method (FRM) used:
- **1**: Classical FRM (max-min)
- **2**: Normalized sum FRM
Compactness:

- 2SLAVE algorithm usually obtains the rule sets with the lower number of rules. However, these rules present a high number of variables and conditions per variable, that makes the interpretability of the rule sets decrease.

- GP-COACH learns rule sets with a quite small number of rules, having them also a low number of variables and condition per variable.

- Tsakonas algorithm obtains rule sets with a low number of variables and conditions per variable, but with a high number of fuzzy rules.

- GP-COACH is able to learn more compact rule sets than the ones obtained by FRBCS_GP.
Results’ Analysis (II)

Accuracy:

- GP-COACH is the algorithm that obtains the best test accuracy results with all the data sets, except with the Wdbc data set, where FRBCS_GP slightly outperforms its results.

- 2SLAVE obtains good test accuracy results, comparable to the ones obtained by FRBCS_GP, but they do not outperform the ones obtained by GP-COACH.

- Tsakonas algorithm obtains the worst results on test accuracy because of the simplicity of its learned rules.
Index

- Introduction
- GP-COACH Algorithm
- Experimental Framework and Results
- Conclusions
Conclusions

- We have proposed GP-COACH, a Genetic Programming based method to obtain COmpact and ACcurate FRBCSs for High-dimensional problems:
  - Evolutionary process (with GCCL codification approach) that uses a context-free grammar to learn DNF rules \( \Rightarrow \) rules with lower antecedent conditions
  - Niche formation mechanism (Token Competition) and a two types of rules (primary and secondary) to increase the diversity into the population \( \Rightarrow \) gives out a fewer number of rules with a high generalization capability

- The experimental study has demonstrated the good performance of our proposal:
  - GP-COACH is able to learn rule sets with a small number of rules, variables and conditions per variable for high-dimensional problems
  - GP-COACH has outperformed the rest of algorithms with regard to the accuracy results on training and test data, showing a high performance for high-dimensional problems
Thank you for your attention!