Evolutionary Computation for Feature Selection and Feature Construction

Mengjie Zhang and Bing Xue
Evolutionary Computation Research Group
Victoria University of Wellington, New Zealand

Mengjie.Zhang@ecs.vuw.ac.nz and Bing.Xue@ecs.vuw.ac.nz

http://www.sigevo.org/gecco-2016/

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s). Copyright is held by the author/owner(s).

GECCO’16 Companion, July 20–24, 2016, Denver, CO, USA.
ACM 978-1-4503-4323-7/16/07.
http://dx.doi.org/10.1145/2908961.2927002
Instructors

Mengjie Zhang is a Professor of Computer Science at the School of Engineering and Computer Science, Victoria University of Wellington (VUW), New Zealand. His research is mainly focused on evolutionary computation, particularly genetic programming, particle swarm optimization and learning classifier systems with application areas of image analysis, multi-objective optimization, classification with unbalanced data, feature selection and reduction, and job shop scheduling. He has published over 400 academic papers in refereed international journals and conferences. He has been serving as an associated editor or editorial board member for five international journals (including IEEE Transactions on Evolutionary Computation and the Evolutionary Computation Journal) and as a reviewer of over fifteen international journals. He has been serving as a steering committee member and a program committee member for over eighty international conferences.

Bing Xue is a Lecturer at Victoria University of Wellington. She is with the Evolutionary Computation Research Group at VUW, and her research focuses mainly on evolutionary computation, machine learning and data mining, particularly, evolutionary computation for feature selection, feature construction, dimension reduction, symbolic regression, multi-objective optimisation, bioinformatics and big data. Bing is has been organising special sessions and issues on evolutionary computation for feature selection and construction. She is also the Chair of IEEE CIS Task Force on Evolutionary Computation for Feature Selection and Construction. Bing is a committee member of Evolutionary Computation Technical Committee, and Emergent Technologies Technical Committee, IEEE CIS. She has been serving as a guest editor, associated editor or editorial board member for international journals, and program chair, special session chair, symposium/special session organiser for a number of international conferences, and as reviewer for top international journals and conferences in the field.
Your Host
— Wellington
The Coolest Little Capital

IEEE Congress on Evolutionary Computation, 2019, Wellington, New Zealand

Dart River Jet Safaris
Queenstown
Milford Sound, Fiordland

Mike Heydon
Rob Suisted

Milford Sound, Fiordland
Outline

• Feature Selection and Feature Construction

• Evolutionary Computation (EC) for Feature Selection

• Feature Selection Methods

• Feature Construction Methods

• Application on Images

• Application on Biology

• Issues and Challenges
Monkeys performing classification task:
- Diagnostic features:
  ‣ Eye separation
  ‣ Eye height
- Non-Diagnostic features:
  ‣ Mouth height
  ‣ Nose length

Monkeys performing classification task
- Diagnostic features:
  - Eye separation
  - Eye height
- Non-diagnostic features:
  - Mouth height
  - Nose length

After Training: 72% (32/44) were selective to one or both of the diagnostic features (and not for the non-diagnostic features)

“The data from the present study indicate that neuronal selectivity was shaped by the most relevant subset of features during the categorisation training.”
— Nathasha Sigala, Nikos Logothetis

Data set (Classification) — Example 1

Credit card application:

- 7 applicants (examples/instances/observations)
- 2 classes: Approve, Reject
- 3 features/variables/attributes

<table>
<thead>
<tr>
<th></th>
<th>Job</th>
<th>Saving</th>
<th>Family</th>
<th>Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Applicant 1</td>
<td>true</td>
<td>high</td>
<td>single</td>
<td>Approve</td>
</tr>
<tr>
<td>Applicant 2</td>
<td>false</td>
<td>high</td>
<td>couple</td>
<td>Approve</td>
</tr>
<tr>
<td>Applicant 3</td>
<td>true</td>
<td>low</td>
<td>couple</td>
<td>Reject</td>
</tr>
<tr>
<td>Applicant 4</td>
<td>true</td>
<td>low</td>
<td>couple</td>
<td>Approve</td>
</tr>
<tr>
<td>Applicant 5</td>
<td>true</td>
<td>high</td>
<td>children</td>
<td>Reject</td>
</tr>
<tr>
<td>Applicant 6</td>
<td>false</td>
<td>low</td>
<td>single</td>
<td>Reject</td>
</tr>
<tr>
<td>Applicant 7</td>
<td>true</td>
<td>high</td>
<td>single</td>
<td>Approve</td>
</tr>
</tbody>
</table>
Cancer Diagnosis—Example 2

**LC-MS/MS**

**Liquid Chromatogram (LC)**

**Full scan MS (MS) at 189 min**

**Tandem MS of ion 444.90 (MS/MS)**
Feature Selection and Feature Construction

- Feature selection aims to pick a subset of relevant features to achieve similar or better classification performance than using all features.

- Feature construction is to construct new high-level features using original features to improve the classification performance.
Why Feature Selection?

- "Curse of the dimensionality"
  - Large number of features: 100s, 1000s, even millions
- Not all features are useful (relevant)
- Redundant or irrelevant features may reduce the performance (e.g. classification accuracy)
- Costly: time, memory, and money
- Feature selection
  - to select a small subset of relevant features from the original large set of features in order to maintain or even improve the performance
Why Feature Construction?

• The quality of input features can drastically affect the learning performance.

• Even if the quality of the original features is good, transformations might be required to make them usable for certain types of classifiers.

• Feature construction does not add to the cost of extracting (measuring) original features; it only carries computational cost.

• In some cases, feature construction can lead to dimensionality reduction or implicit feature selection.
What can FS/FC do?

- Reduce the dimensionality (No. of features)
- Improve the (classification) performance
- Simplify the learnt model
- Speed up the processing time
- Help visualisation and interpretation
- Reduce the cost, e.g. save memory
- and ?
Challenges in FS and FC

- **Large search space:** $2^n$ possible feature subsets
  - 1990: $n < 20$
  - 1998: $n \leq 50$
  - 2007: $n \approx 100s$
  - Now: 1000s, 1 000 000s

- **Feature interaction**
  - Relevant features may become redundant
  - Weakly relevant or irrelevant features may become highly useful

- **Slow** processing time, or even not possible

- **Multi-objective Problems**
General FS/FC System

Evolutionary Feature Selection/Construction → Constructed/Selected Feature(s)

Training Set → Unseen Test Set

Data Transformation

Classification Performance

Classification Algorithm

Transformed Training Set

Transformed Test Set
Feature FS/FC Process

On training set:

- Initialisation → Constructed/Selected Feature(s) → Subset → Feature(s) Evaluation → Goodness of the Subset
- If Goodness of the Subset is No, go back to Stop Criterion. If Yes, go to Results Evaluation.
Feature Selection Approaches

- Based on Evaluation —— learning algorithm
  - Three categories: Filter, Wrapper, Embedded
  - Hybrid (Combined)

![Diagram showing Feature Selection Approaches]

- Original Features
- Selected Features
- Evaluation (Measure)
- Learnt Classifier
- Selected Features
- Learnt Classifier
## Feature Selection Approaches

- Generally:

<table>
<thead>
<tr>
<th>Method</th>
<th>Classification Accuracy</th>
<th>Computational Cost</th>
<th>Generality (different classifiers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter</td>
<td>Low</td>
<td>Low</td>
<td>High</td>
</tr>
<tr>
<td>Embedded</td>
<td>Medium</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td>Wrapper</td>
<td>High</td>
<td>High</td>
<td>Low</td>
</tr>
</tbody>
</table>
Feature Selection
Feature Selection Approaches

- Conventional approaches
  - The Relief algorithm
    - Feature ranking method
  - The FOCUS algorithm
  - Sequential forward/backward selection
  - Sequential forward/backward floating selection
  - Statistical feature selection methods
- Evolutionary Computation (EC) based approaches

Evolutionary Computation (EC)

- A group of techniques inspired by the principles of biological evolution
Why Evolutionary Computation?

- Don't need domain knowledge
- Don’t make any assumption
  - e.g. differentiable, linearity, separability, equality
- Easy to handle constraints
- EC can simultaneously build model structures and optimise parameters
- Population based search is particularly suitable for multi-objective optimisation
EC for Feature Selection

- EC Paradigms
- Evaluation
- Number of Objectives

EC for Feature Selection

- Genetic algorithms (GAs), Genetic programming (GP)
- Particle swarm optimisation (PSO), ant colony optimisation (ACO)
- Differential evolution (DE), memetic algorithms, learning classifier systems (LCSs)

EC for Feature Selection

GAs for Feature Selection

- Over 25 years ago, first EC techniques
  - Filter, Wrapper, Single Objective, Multi-objective
- Representation
  - Binary string
- Search mechanisms
  - Genetic operators
- Multi-objective feature selection
- Scalability issue

GP for Feature Selection

- Implicit feature selection
  - Filter, Wrapper, Single Objective, Multi-objective

- Embedded feature selection

- Feature construction

- Computationally expensive

PSO for Feature Selection

- Very popular in recent years
  - Filter, Wrapper, Single Objective, Multi-objective
- Representation, continuous PSO vs Binary PSO
- Search mechanism
- Fitness function

- Scalability

ACO for Feature Selection

- Start from around 2003
  - Filter, Wrapper, Single Objective, Multi-objective
- Representation
- Search mechanism
- Filter approaches

- Scalability

DE, LCSs, and Memetic

- **DE:** since 2008
  - potential for large-scale
- **LCSs:**
  - implicit feature selection
  - embedded feature selection
- **memetic:**
  - population search + local search
  - Wrapper + filter


Related Areas (Applications)

- Biological and biomedical tasks
  - gene analysis, biomarker detection, cancer classification, and disease diagnosis
- Image and signal processing
  - image analysis, face recognition, human action recognition, EEG brain-computer-interface, speaker recognition, handwritten digit recognition, personal identification, and music instrument recognition.
- Network/web service
  - Web service composition and development, network security, and email spam detection.
- Business and financial problems
  - Financial crisis, credit card issuing in bank systems, and customer churn prediction.
- Others
  - power system optimisation, weed recognition in agriculture, melting point prediction in chemistry, and weather prediction.

Feature Selection
PSO for FS: initialisation and updating

- **Initialisation:**
  - Forward selection
  - Backward selection
  - Mixture of both

- **Updating:**
  - Consider the number of features in the pest and gbest updating

---

PSO FS: with backward elimination

Backward Elimination

1. Collect features selected by \textit{gbest}
2. For the first cluster:
   - \textit{gbest} selected more than \( \sqrt{m+1} \) features?
     - Yes: Go to the next cluster
     - No: Calculate fitness values of all selected features

   - All values larger than 0?
     - Yes: Remove the feature with the smallest fitness value
     - No: \textit{gbest} selected more than \( \sqrt{m+1} \) features?

3. Visit all clusters?
   - Yes: Update \textit{gbest}
   - No: \textit{x}: the position value in the \textit{i}th dimension

Filter Measure

\[
f'(s_i) = \frac{1}{x_i} \left( \frac{1}{|s|} \cdot \frac{1}{\text{Red}(s_i)} - \frac{1}{\text{Rel}(s_i)} \right)
\]

- By adding \( \frac{1}{x_i} \), \( f'(s_i) \) ensures that if two features have the same \( f(s) \) value, the one with a smaller position value (i.e., smaller probability) will be removed

- \( s_i \) is removed only when \( f'(s_i) < 0 \) and \( f'(s_i) \) is the smallest value

\[
\text{Rel}(s_i) = I(s_i; c) = \frac{1}{|S| - 1} \sum_{s_j \in S, s_j \neq s_i} I(s_i; s_j)
\]

\[
\text{Red}(s_i) = \frac{1}{|S| - 1} \sum_{s_j \in S, s_j \neq s_i} I(s_i; s_j)
\]

\text{Rel}(s_i): \text{relevance contribution of } s_i \text{ in } S

\text{Red}(s_i): \text{redundancy in } S \text{ caused by } s_i
Multi-objective PSO for FS

- Introduce and develop the first multi-objective PSO approach to feature selection
  - Simultaneously minimise the number of features and the error rate
  - ~121 citations since June 2013


Multi-objective PSO for FS

- Simultaneously minimise the number of features and the error rate

T-Test on Hypervolume Ratios on Training Accuracy

Multi-objective PSO for FS: Binary VS continuous

Example:
- Ave: (20, 41.5)
- Best: (20, 40)

Probability based BPSO (PBPSO)

- Updating equations:

\[ x_{id}(t + 1) = \begin{cases} 1 - x_{id}(t), & \text{if } \text{rand}() < p_{id} \\ x_{id}(t), & \text{otherwise} \end{cases} \]

\[ p_{id} = p_0 + p_{pd} + p_{gd} \]

\[ p_{pd} = \begin{cases} p_1, & \text{if } x_{id}(t) \neq y_{id}(t) \\ 0, & \text{otherwise} \end{cases} \]

\[ p_{gd} = \begin{cases} p_2, & \text{if } x_{id}(t) \neq \hat{y}_{id}(t) \\ 0, & \text{otherwise} \end{cases} \]

where \( y_{id}(t) \) represents \textit{pbest}, and \( \hat{y}_{id} \) represents \textit{gbest}.

\[ p_0 + p_1 + p_2 = 1 \]
### Probability based BPSO (PBPSO)

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Method</th>
<th>AveSize</th>
<th>Best Acc</th>
<th>AveAcc ± StdAcc</th>
<th>Test 1</th>
<th>Test 2</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hillvalley</td>
<td>All</td>
<td>100</td>
<td>56.59</td>
<td></td>
<td>=</td>
<td>+</td>
<td>50.28</td>
</tr>
<tr>
<td></td>
<td>BPSOfs</td>
<td>39.35</td>
<td>60.16</td>
<td>56.88 ± 1.6322</td>
<td>=</td>
<td>+</td>
<td>46.1</td>
</tr>
<tr>
<td></td>
<td>PBPSOfs</td>
<td>81.08</td>
<td>61.26</td>
<td>58.25 ± 1.8952</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Musk1</td>
<td>All</td>
<td>166</td>
<td>83.92</td>
<td></td>
<td>=</td>
<td>+</td>
<td>13.96</td>
</tr>
<tr>
<td></td>
<td>BPSOfs</td>
<td>75.52</td>
<td>90.91</td>
<td>84.21 ± 2.8401</td>
<td>=</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PBPSOfs</td>
<td>89.3</td>
<td>88.81</td>
<td>85.38 ± 1.8087</td>
<td>+</td>
<td>+</td>
<td>12.7</td>
</tr>
<tr>
<td>Arrhythmia</td>
<td>All</td>
<td>279</td>
<td>94.46</td>
<td></td>
<td>-</td>
<td>+</td>
<td>16.97</td>
</tr>
<tr>
<td></td>
<td>BPSOfs</td>
<td>99.7</td>
<td>95.14</td>
<td>94.21 ± 0.3937</td>
<td>-</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PBPSOfs</td>
<td>63.42</td>
<td>95.48</td>
<td>94.71 ± 0.3405</td>
<td>+</td>
<td>+</td>
<td>12.44</td>
</tr>
<tr>
<td>Madelon</td>
<td>All</td>
<td>500</td>
<td>70.9</td>
<td></td>
<td>+</td>
<td>+</td>
<td>991.6</td>
</tr>
<tr>
<td></td>
<td>BPSOfs</td>
<td>243.85</td>
<td>78.59</td>
<td>75.81 ± 1.4905</td>
<td>+</td>
<td>+</td>
<td>953.63</td>
</tr>
<tr>
<td></td>
<td>PBPSOfs</td>
<td>212.42</td>
<td>81.15</td>
<td>78.91 ± 1.2565</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Isolet5</td>
<td>All</td>
<td>617</td>
<td>98.45</td>
<td></td>
<td>-</td>
<td>+</td>
<td>384.16</td>
</tr>
<tr>
<td></td>
<td>BPSOfs</td>
<td>225.15</td>
<td>98.59</td>
<td>98.25 ± 0.1354</td>
<td>-</td>
<td>+</td>
<td>328.34</td>
</tr>
<tr>
<td></td>
<td>PBPSOfs</td>
<td>169.35</td>
<td>98.87</td>
<td>98.61 ± 0.1248</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
<tr>
<td>Multiple Features</td>
<td>All</td>
<td>649</td>
<td>98.63</td>
<td></td>
<td>+</td>
<td>+</td>
<td>692.54</td>
</tr>
<tr>
<td></td>
<td>BPSOfs</td>
<td>237.05</td>
<td>99.1</td>
<td>98.89 ± 0.0923</td>
<td>+</td>
<td>+</td>
<td>551.44</td>
</tr>
<tr>
<td></td>
<td>PBPSOfs</td>
<td>176.15</td>
<td>99.27</td>
<td>99.01 ± 0.1043</td>
<td>+</td>
<td>+</td>
<td></td>
</tr>
</tbody>
</table>

Introduce statistical feature clustering to feature selection and develop the first approach
- reduce the size of the search space
- \#features: from 600 to \sim 12
- implicitly consider feature interaction
- Example:
  - our method achieved accuracy 100\%: \{10, 7, 3\}
  - Single feature ranking: 7, 10, 12, 1, 9, 11, 6, 2, 13, 5, 4, 3
EC and Statistical Grouping for FS

- Development of four new particle position update algorithms that automatically select a single feature from each feature cluster
- As features are grouped by similarity, a single feature is expected to provide enough information about its feature cluster


Information Theory Feature Selection

- Information theory in evolutionary feature selection
  - Fast algorithm — mutual information
  - New measures, evaluate multiple features
  - Evolutionary multi-objective filter feature selection

<table>
<thead>
<tr>
<th></th>
<th>F-MI</th>
<th>F-E</th>
<th>F-RS</th>
<th>F-PRS</th>
<th>W-SVM</th>
<th>W-5NN</th>
<th>W-DT</th>
<th>W-NB</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>0.05</td>
<td>2.88</td>
<td>2.07</td>
<td>2.86</td>
<td>24.41</td>
<td>6.12</td>
<td>5.19</td>
<td>13.46</td>
</tr>
<tr>
<td>Value</td>
<td>0.05</td>
<td>97.7</td>
<td>2485.61</td>
<td>2766.29</td>
<td>5143.18</td>
<td>9311.59</td>
<td>189.43</td>
<td>304.08</td>
</tr>
<tr>
<td>Value</td>
<td>0.06</td>
<td>27.95</td>
<td>55.3</td>
<td>38.36</td>
<td>270.64</td>
<td>264.51</td>
<td>43.15</td>
<td>150.37</td>
</tr>
<tr>
<td>Value</td>
<td>0.07</td>
<td>9.85</td>
<td>14.81</td>
<td>9.95</td>
<td>118.37</td>
<td>72.72</td>
<td>47.87</td>
<td>19.42</td>
</tr>
<tr>
<td>Value</td>
<td>0.09</td>
<td>256.57</td>
<td>1372.93</td>
<td>1827.06</td>
<td>2441.21</td>
<td>4095.07</td>
<td>244.55</td>
<td>377.24</td>
</tr>
<tr>
<td>Value</td>
<td>0.15</td>
<td>2.96</td>
<td>6.9</td>
<td>0.68</td>
<td>5.4</td>
<td>1.68</td>
<td>3.82</td>
<td>4.13</td>
</tr>
<tr>
<td>Value</td>
<td>0.18</td>
<td>236.42</td>
<td>928.25</td>
<td>911.3</td>
<td>10937.87</td>
<td>1936.67</td>
<td>529.7</td>
<td>706.23</td>
</tr>
</tbody>
</table>


Feature Selection Though Data Discretisation

Bare-Bone Particle Swarm Optimisation

Two-stage (PSO-FS)

- Training Set
- Test Set
- Data discretisation
- Cut-points for all features
- Disc. Training
- Disc. Test
- Feature selection (PSO)
- Selected Feature Subset
- New Training
- New Test
- Transform Data
- Training a Classifier
- Test the trained Classifier
- Accuracy

One-stage (PSO-DFS)

- Training Set
- Test Set
- Data discretization and feature selection (PSO_DFS)
- Cut-points for selected features
- New Training
- New Test
- Transform Data
- Training a Classifier
- Test the trained Classifier
- Accuracy

Feature Selection Though Data Discretisation

<table>
<thead>
<tr>
<th>Min</th>
<th>Max</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.5</td>
<td>10.2</td>
</tr>
<tr>
<td>-9.2</td>
<td>20.5</td>
</tr>
<tr>
<td>7.6</td>
<td>12.7</td>
</tr>
<tr>
<td>-8.5</td>
<td>4.5</td>
</tr>
<tr>
<td>6.9</td>
<td>50.1</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>-9.3</td>
<td>-1.6</td>
</tr>
</tbody>
</table>

Particle's Position

| 7.2 | 10.4 | 12.7 | -3.8 | 6.9 | ... | -4.5 |

Candidate solution

Not selected

Filter FS based on Rough Set

- Promote rough set theory for feature selection
  - Others’: mainly < 200 features
  - Ours: more than 600 features
FS based on Rough Set

\[
\text{Fitness} = \frac{\sum_{i=1}^{n} |\text{aprP } U_i|}{|U|} + \frac{\sum_{x \in \{\text{EqC}\}} |x|}{|\text{EqC}|}
\]

- \(U\) is the universe or the whole dataset
- \(U_i\) is one class in the dataset
- \(\text{EqC}\): equivalence classes
- \(\text{aprP}\) is the lower approximation in probabilistic rough set theory
- A parameter \(\alpha\) to relax the definition of \(\text{aprP}\)


GP for Embedded Feature Selection

Feature Construction
Why Use GP for Feature Construction?

- GP is **flexible** in making mathematical and logical functions.

- There isn’t much structural (topological) information in the search space of possible functions, so using a meta-heuristic approach (such as evolutionary computation) seems reasonable.

![Diagram showing the process of feature construction from selected features to constructed features using GP.](image-url)
One constructed feature for one class

Defining a measure of goodness for a single feature:

- The interval of a class along a feature is determined by the dispersion of the instances of that class along the feature axis. The dispersion of instances themselves is related to the distribution of data points in that class.

- An interval $I$ is represented with a pair $(lower, upper)$ which shows the lower and upper boundaries of the interval. $I_c$ is used to indicate an interval for class $c$.

- The interval of class $c$ could be formulated as follows if the class distributions were normal.

$$I_c = [\mu_c - 3\sigma_c, \mu_c + 3\sigma_c]$$

- However, the normality assumption is not always satisfied.
GP for FC Measure: Examples of good and bad class intervals

- Overlapping intervals

- Non-overlapping intervals

• 4 features, 3 classes

• Construct multiple features from a single tree

Image Recognition/Classification
Image Recognition/Classification

- The traditional way
- Domain-specific pre-extracted features approach (DS-GP)

1. **Input**
   - The input is raw image pixel values

2. **Design**
   - The feature areas need to be designed by domain-experts

3. **Feature Extraction**
   - Transform the pixel values of the selected areas to a different domain

4. **Feature Selection**
   - Select a subset out of the extracted features (optional)

5. **Classification**
   - Feed the extracted features (with or without selection) to a GP-based classifier
Images: GP-Surff

- Improve domain-independent object classification in images by using GP techniques.

**Designing** a program representation that is capable of detecting sub-regions of the image that are rich in features;

**Constructing** a classification system to extract features from the selected regions and then use a SVM classifier and voting scheme to predict the class label; and

**Investigating** whether the regions detected by the new method are similar to those designed by domain experts.

Images: GP-Surff

- A program evolved on JAFFE, average over 95% test accuracy
- The program detects 4 interesting regions

• GP-HoG uses strongly typed GP to perform three tasks in the same tree structure.
• All layers are trained simultaneously and coherently.
• Output of the tree is thresholded.

Images: GP-HoG Method

- The below tree has 98% training and 95% test performance on the Jaffe dataset despite being very simple.

- The below tree has 95% training and 100% test performance on the Jaffe dataset.

Biology
Due to the nature, the MS data production process is very expensive (costs around 2,000 NZD daily) and time consuming (around two weeks to produce a single sample).

The number of samples available is very small and the number of features in each sample is extremely large.

Moreover, the features of interest are too small.

The classification of MS data is challenging.
### Biological Datasets

<table>
<thead>
<tr>
<th>Data set</th>
<th># Features</th>
<th># Samples</th>
<th># Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pancreatic Cancer</td>
<td>6771</td>
<td>181</td>
<td>2</td>
</tr>
<tr>
<td>Ovarian Cancer1</td>
<td>15154</td>
<td>253</td>
<td>2</td>
</tr>
<tr>
<td>Ovarian Cancer2</td>
<td>15000</td>
<td>216</td>
<td>2</td>
</tr>
<tr>
<td>Prostate Cancer</td>
<td>15000</td>
<td>322</td>
<td>4</td>
</tr>
<tr>
<td>Toxpath</td>
<td>7105</td>
<td>115</td>
<td>4</td>
</tr>
<tr>
<td>Arcene</td>
<td>10,000</td>
<td>200</td>
<td>2</td>
</tr>
<tr>
<td><strong>Apple-plus</strong></td>
<td><strong>773</strong></td>
<td><strong>40</strong></td>
<td><strong>4</strong></td>
</tr>
<tr>
<td>Apple-minus</td>
<td>365</td>
<td>40</td>
<td>4</td>
</tr>
</tbody>
</table>
Proteins

Metabolites

Sample

Ion Source

Mass Analyzer

Detector

Mass spectrometry

Data Analysis

Spectrum
Biology

Figure 1.1: Stages of the biomarker identification process

- Detection (Feature Manipulation)
- Verification (Prediction of detectable peptides)
- Experimental Validation

Biomarker Detection

- Single-objective
  - Ensemble Feature Ranking Ch.3
  - Multiple Feature Construction Ch.4
- Multi-objective
  - Multi-objective Feature selection and Construction Ch.5

Biomarker Verification Ch.6

Multiple Alignment of MS Data Ch.7

Soha Ahmed, Genetic Programming for Biomarker Detection in Classification of Mass Spectrometry Data, PhD thesis, 2015, School of Engineering and Computer Science, Victoria University of Wellington, New Zealand
Biology: Feature ranking and GP FS

Figure 3.6: Biomarker detection of the proposed method in comparison with IG and RF.

## Biomarker Identification

<table>
<thead>
<tr>
<th>m/z values in Apple–plus data set (12 biomarkers)</th>
<th>New Method (9 ✓)</th>
<th>Method 2 (3 ✓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>331.21</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>471.09</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>107.05, 169.05, 238.05, 275.09, 456.11, 459.13</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>456.62, 475.10</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>449.11</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>229.09</td>
<td>✓</td>
<td>✗</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Apple minus m/z (5 biomarkers)</th>
<th>New Method (5 ✓)</th>
<th>Method 2 (2 ✓)</th>
</tr>
</thead>
<tbody>
<tr>
<td>463.0</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>447.09</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>273.03</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>435.13</td>
<td>✓</td>
<td>✗</td>
</tr>
<tr>
<td>227.07</td>
<td>✓</td>
<td>✗</td>
</tr>
</tbody>
</table>
Biology: GP for Measuring Peptide Detectability

Biology: PSO with local search on $p_{best}$ and resetting $g_{best}$ (PSO-LSRG)

- Use a filter measure to identify:
  - **Relevant features**: correlated to the class label.
  - **Redundant features**: correlated with each other.
- Symmetric uncertainty (SU) is a normalised version of information gain (IG).
Biology: PSO with local search on pbest and resetting gbest (PSO-LSRG)

- A PSO based hybrid FS algorithm for high-dimensional classification.
- PSO-LSSU combines wrapper and filter measures:
  - The fitness function.
  - The local search.
- PSO-LSSU achieved much smaller feature subsets with significantly better classification performance than the compared methods in most cases.

5 - 6 times faster than PSO
• If the **whole dataset** is used during FS/FC process, the experiments(or evaluation) have **FS/FC Bias**

• What if only a small number of instances available ?
  - In classification, use **k-fold cross validation**
  - How to use **k-fold cross validation** in FS/FC to evaluate a FS/FC system?
Many works on bio-data containing feature selection which leads to biased results - conclusion might change.

Feature Selection Bias

<table>
<thead>
<tr>
<th>Experiment II</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Performance Evaluation</strong></td>
</tr>
<tr>
<td>To avoid feature selection bias and compare the performance of the algorithms with and without bias, the second set of experiments without feature selection bias have been conducted.</td>
</tr>
</tbody>
</table>
Binh Tran, Bing Xue and Mengjie Zhang. “Genetic Programming for Feature Construction and Selection in Classification on High-dimensional Data”, Memetic Computing, Accepted 4 December 2015. (DOI:10.1007/s12293-015-0173-y)
Work from ECRG
Weakness and Issues

• Search space:
  - Large search space: bit-string/vector with a length equal to the total number of features
  - Classification accuracy or existing filter measures in the fitness function, which often cannot lead to a smooth fitness landscape or with low locality

• Long computational time
  - A large number of evaluations
  - Wrapper: each evaluation involves a learning process of a machine learning or data mining algorithm
  - Filters are computationally cheaper than wrappers

• Poor scalability
  - the dimensionality of the search space often equals to the total number of features, thousands, or even millions
  - the number of instances is large
Weakness and Issues

- Feature selection or construction bias issue

- Generalisation issue
  - especially wrappers: selected or constructed features can easily overfit the wrapped learning algorithm and the training data, leading to poor performance on unseen test data
  - Feature construction
Future Directions

• Efficient and effective filter measure for the fitness function:
  - reduce the computational cost,
  - smooth the landscape of the search space,
  - improve the learning and generalisation performance, and
  - increase the interpretability/understandability of the obtained feature set

• Representation
  - Reduce the search space
  - Incorporate more information of about the features, e.g. relative importance of features, feature interactions or feature similarity
  - Embedded feature selection or construction
Future Directions

• Search mechanism
  - Evolutionary multi-objective optimisation
  - Combinatorial optimisation
  - Memetic computing
  - Large-scale optimisation
  - Surrogate models
  - Adaptive parameter control techniques

• Feature construction
  - both feature selection and feature construction

• Instance selection and construction

• Combining EC with *machine learning approaches*

• Feature selection and feature construction for other machine learning tasks: clustering and symbolic regression
Acknowledgement

- Thanks A/Prof Will Browne, A/Prof Peter Andreae, A/Prof Ivy (I-Ming) Liu, A/Prof Lin Shang, Dr Lifeng Peng, Dr Kourosh Neshatian, Dr Yi Mei, Dr Su Nguyen, Dr Soha Ahmed, Harith Al-Sahaf, Liam Cervante, Andrew Lensen, Hoai Bach Nguyen, Binh Tran, Qi Chen, Emrah Hancer, and others

- Funding Agents:
  1. Marsden Fund of New Zealand award number(s): VUW1209
  2. University Research Fund at Victoria University of Wellington award number(s): 210375/3557, 209861/3580
Acknowledgement

- Thanks everyone in the Evolutionary Computation Research Group at Victoria University of Wellington, New Zealand
Activities in 2016

- Task Force on Evolutionary Computation for Feature Selection and Construction, IEEE CIS
- IEEE Symposium on Computational Intelligence in Feature Analysis, Selection, and Learning in Image and Pattern Recognition (FASLIP) in IEEE SSCI 2016
- Australian Conference on Artificial Lift and Computational Intelligence (ACALCI 2017)
- Special session on Evolutionary Feature Selection and Construction in IEEE WCCI 2016 /CEC2016
- Special session on Transfer Learning in Evolutionary Computation in IEEE WCCI 2016 /CEC2016
- Special Issue on Evolutionary Optimisation, Feature Reduction and Learning, Soft Computing (Journal), Springer
Proposed Activities

- Task Force on Evolutionary Computation for Feature Selection and Construction, IEEE CIS
- Special session on Evolutionary Feature Selection and Construction in CEC2017
- Special session on Evolutionary Machine Learning in Image Analysis and Pattern Recognition in IES2017
- IEEE Symposium on Computational Intelligence in Feature Analysis, Selection, and Learning in Image and Pattern Recognition (FASLIP) in IEEE SSCI 2017
- Special session on Transfer Learning in Evolutionary Computation in IEEE WCCI 2016 /CEC2016
- The tutorial on EC for Feature Selection and Feature Construction, GECCO 2017
- The tutorial on EC for Feature Selection and Feature Construction, CEC2017
- Special Issue
Call for Papers

IEEE Symposium on Computational Intelligence in Feature Analysis, Selection, and Learning in Image and Pattern Recognition (FASLIP)

2016 IEEE Symposium Series on Computational Intelligence (SSCI 2016)
http://ssci2016.cs.surrey.ac.uk/

December 6-9, 2016, Athens, Greece

Deadline: 15 August 2016
PhD Scholarships Available in Evolutionary Computation

Join our internationally renowned and friendly research team:
  • Up to eight funded PhDs (fees + stipend) available 3 times a year,
  • 3yr duration in English, with expert supervision.

Five major EC strategic research directions:
  • Feature selection/construction for classification, regression, clustering
  • Combinatorial optimisation: scheduling, routing, web services
  • Computer vision and image analysis
  • Multi- and many- criteria optimisation
  • Transfer learning

Techniques include: Genetic Programming, Learning Classifier Systems, Particle Swarm Optimisation, Differential Evolution, and many others.

Wellington voted as ‘Coolest little capital in the World’!

VUUW is the top-rated research university in New Zealand.

Requirements: MSc/ME; GPA >= 3.5/4; research experience/publications

Come and find us after one of our many talks or apply at:
http://www.victoria.ac.nz/fgr/prospective-phds/scholarships

Find us: http://ecs.victoria.ac.nz/Groups/ECRG/WebHome or [Search: ‘ECRG VUU’]
Call for Postdoctoral Fellows

- Evolutionary Computation Research Group, Victoria University of Wellington, NZ
- Postdoc in Evolutionary Computations
- Salary: $70,000 – 85,000
- Areas:
  - Evolutionary Feature Selection and High Dimensionality Reduction
  - Evolutionary Image Analysis
  - Classification and Clustering
  - Transfer Learning
- Huawei NZ Funded Project
- Contact: Mengjie.Zhang@ecs.vuw.ac.nz or Bing.Xue@ecs.vuw.ac.nz
Thank you