Analysis of a Genetic Programming Algorithm for Association Studies

Robin Nunkesser

Department of Computer Science, TU Dortmund

GECCO 2008
Outline

1. Introduction
   - Motivation
   - Problem Definition

2. The Algorithm
   - Definition
   - Experimental Justification

3. Results
   - Logic Minimization
   - Genetic Association Studies
Genetic Association Studies

Task: Identify genetic factors that may contribute to a medical condition

- Data sets on genetic factors of individuals (cases and controls)
- SNPs are the genetic factors typically considered

Single Nucleotide Polymorphism

DNA sequence variation

Single: located at a single base/nucleotide pair

Nucleotide: sugar, phosphate and base Adenine, Thymine, Cytosine, or Guanine

Polymorphism: nucleotide variant with a frequency of $\geq 1\%$

$\approx 90\%$ of genetic variations are SNPs
Introduction

The Algorithm

Results

Summary and Outlook

Single Nucleotide Polymorphisms

- Most SNPs are biallelic (two variants exist)
- Let \( A \) be the major/reference allele and \( a \) the minor allele
- Biallelic SNPs can be divided into three types determined by the mother chromosome and the father chromosome:
  - homozygous reference \( AA \) (coded as 0)
  - heterozygous variant \( aA \) or \( Aa \) (1)
  - homozygous variant \( aa \) (2)

Data Example

<table>
<thead>
<tr>
<th>SNP_1</th>
<th>SNP_2</th>
<th>SNP_3</th>
<th>SNP_4</th>
<th>SNP_5</th>
<th>SNP_6</th>
<th>case/control</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>
Logic Minimization

Observation: Multi-valued logic minimization on incompletely specified truth tables works on the same data

Major difference: Input has to be fitted exactly

Some Related Work

- Standard multi-valued logic minimization tool: Espresso MV (Rudell and Sangiovanni-Vincentelli, 1987)
- Boolean case: GP algorithm by Droste (1997)
Problem Definition

Output in $B := \{0, 1\}$, inputs from $P := \{0, \ldots, p\}$

**Goal:** Find the function $f : P^n \rightarrow B$ that “fits” the input best

**Our approach:** Map multiple-valued variables to all distinct Boolean variables

$$X^a := \begin{cases} 1, & \text{if } X = a \\ 0, & \text{otherwise} \end{cases}$$

and $\overline{X^a}$ (or $(X = a)$ and $(X \neq a)$) and search for functions in DNF

Example of an individual
**GPAS (Nunkesser et al., 2007)**

1. Create an initial random population consisting of two individuals

2. Perform the following steps on the current generation:
   1. Select all individuals in the population, and reproduce them
   2. Conduct mutations and crossovers to uniformly at random selected individuals in the population
   3. Evaluate the fitness value of the adapted and reproduced individuals with fitness functions evaluating
      1. Number of predicted controls
      2. Number of predicted cases
      3. Size of the logic expression
   4. Select all adapted and reproduced individuals that are not pareto dominated for the next generation

3. If the termination criterion is fulfilled, then output the final population. Otherwise, set the next generation as current and go to step 2
Crossover and Mutations

Original individual

```
         OR
        /   \
      AND  SNP2 = 0
     /    \
SNP1 = 2  SNP3 = 0
```

Individual chosen for crossover

```
         OR
        /   \
      AND  SNP2 = 0
     /    \
SNP1 = 2  SNP3 = 0
```

Crossover

```
         OR
        /   \
      AND  SNP1 = 2
     /    \
 SNP3 = 0  SNP2 = 2
```

Replace a literal by a new literal

```
         OR
        /   \
      AND  SNP1 = 2
     /    \
SNP2 = 0  SNP3 = 0
```

Delete a monomial

```
         OR
        /   \
      AND  SNP1 = 2
     /    \
SNP2 = 0  SNP3 = 0
```

Delete a literal

```
         OR
        /   \
      AND  SNP1 = 2
     /    \
SNP2 = 0  SNP3 = 0
```

Insert a new literal

```
         OR
        /   \
      AND  SNP1 = 2
     /    \
 SNP2 = 0  SNP3 = 0
```

Insert a new literal as a new monomial

```
         OR
        /   \
      AND  SNP1 = 2
     /    \
SNP2 = 0  SNP3 = 0
```
Experimental Justification

- Wilcoxon signed rank tests based on 100 runs of two different algorithms
- For more algorithms: Friedman rank sum test and Bonferroni corrected Wilcoxon tests
- Simulated Data

Results

- Three objectives outperform one or two objectives
- Crossover speeds the algorithm up
- Crossover leads to better results upon stagnation
First Experiment for Logic Minimization

- DNF of size 5 is considered for different values of the number of variables $n \in \{10, 20, \ldots, 50\}$ and the number of input data $m \in \{100, 1000, 10000, 25000, 50000\}$
- Maximum run time 1 hour
- GPAS finds the optimal solution in all cases, Standard GP (Koza, 1992) in no case

### Results for Espresso MV

<table>
<thead>
<tr>
<th>$n$</th>
<th>$m = 100$</th>
<th>1000</th>
<th>10000</th>
<th>25000</th>
<th>50000</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>20</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
</tr>
<tr>
<td>30</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>×</td>
<td>×</td>
</tr>
<tr>
<td>40</td>
<td>×</td>
<td>×</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
</tr>
<tr>
<td>50</td>
<td>×</td>
<td>×</td>
<td>❌</td>
<td>❌</td>
<td>❌</td>
</tr>
</tbody>
</table>
Second Experiment

- Good performance on short DNF is not surprising
- We consider DNF sizes in \( \{5, 10, \ldots, 50\} \) now

Results for different DNF sizes

- Thinner boxplots indicate unsuccessful (in the given time) runs
Multiplexer Experiment

Definition

The *11-multiplexer* \( \text{MUX}11 : B^{11} \to B \) is defined as

\[
\text{MUX}11(a_2, a_1, a_0, d_7, \ldots, d_0) := d_{a_22^2 + a_12^1 + a_02^0}
\]

- Espresso MV and GPAS find the DNF of \( \text{MUX}11 \) in seconds
- To test the ability for logic concept learning we consider sampled training sets of sizes \( \{2^6, 2^7, \ldots, 2^{10}\} \)
- The misclassification rate (MCR) is the ratio of inputs the learned function delivers a wrong result for

<table>
<thead>
<tr>
<th>( m )</th>
<th>64</th>
<th>128</th>
<th>256</th>
<th>512</th>
<th>1024</th>
</tr>
</thead>
<tbody>
<tr>
<td>mean MCR</td>
<td>28.12</td>
<td>8.59</td>
<td>0.4</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>
Overfitting

- One of the main problems in genetic association studies
- Major difference to logic minimization

Goal: Determine the correct model size (here 5) in the training data
Avoid Overfitting

Main ideas:

- Only consider individuals on the convex hull in relation to MCR and length
- Consider the ratio of MCR improvement per length unit

**MCR improvement per length unit for points on the convex hull**

![Graph showing MCR improvement per length unit for points on the convex hull.](image)

Observation: Correct model size 5 becomes easier to spot
Automated Rules

Two main ideas:

- Use a threshold function to determine the correct model size
- Use a gap statistic to find the right model size (see Tibshirani et al., 2001)

Here: Constant threshold

### MCR of discrimination on simulated data

<table>
<thead>
<tr>
<th></th>
<th>GP Algorithm</th>
<th>Logic Regression</th>
<th>CART</th>
<th>Bagging</th>
<th>Random Forests</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean MCR</td>
<td>0.329</td>
<td>0.342</td>
<td>0.371</td>
<td>0.382</td>
<td>0.379</td>
</tr>
<tr>
<td>St. Dev.</td>
<td>0.018</td>
<td>0.022</td>
<td>0.025</td>
<td>0.018</td>
<td>0.018</td>
</tr>
</tbody>
</table>

### Constant threshold for model sizes between 1 and 12

<table>
<thead>
<tr>
<th></th>
<th>Correct size</th>
<th>Correct size ±1</th>
<th>Different size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Size chosen</td>
<td>70.41%</td>
<td>14.42%</td>
<td>15.17%</td>
</tr>
</tbody>
</table>
Results on Real Data

**GENICA**
- **Gene ENvironment Interaction and Breast CAncer in Germany**
- Survey on genetic and environmental influence factors on sporadic breast cancer
- Here: 63 SNPs from 1258 women

**HapMap**
- Original purpose: Search for common haplotypes of four different ethnies
- Here: Discriminate between two ethnies
- 121774 SNPs from 90 individuals
Results on GENICA

MCR of LR and the GP algorithm for a restricted number of variables

MCR and running times of discrimination

<table>
<thead>
<tr>
<th></th>
<th>GP Algorithm</th>
<th>Logic Regression</th>
<th>CART</th>
<th>Bagging</th>
<th>Random Forests</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>MCR</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>0.392</td>
<td>0.405</td>
<td>0.429</td>
<td>0.457</td>
<td>0.450</td>
</tr>
<tr>
<td></td>
<td>Runtime</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6.31</td>
<td>11.75</td>
<td>1.37</td>
<td>21.77</td>
<td>9.03</td>
</tr>
</tbody>
</table>

Robin Nunkesser

Robin Nunkesser

GP for Association Studies

GECCO 2008
### Results on HapMap

#### MCR and running times of discrimination on significant genes

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>GP Algorithm</th>
<th>Logic Regression</th>
<th>CART</th>
<th>Bagging</th>
<th>Random Forests</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCR</td>
<td>0.011</td>
<td>0.144</td>
<td>0.356</td>
<td>0.022</td>
<td>0.011</td>
</tr>
<tr>
<td>Runtime</td>
<td>1.1 (89.3)</td>
<td>1.15</td>
<td>0.83</td>
<td>5.01</td>
<td>0.3</td>
</tr>
</tbody>
</table>

#### Search on BRLMM genotypes

- Number of generations vs. runtime.
- Running Time:
  - About 8 minutes for 10,000 generations.
  - Other approaches do not work on a standard PC.
Summary

Genetic Programming for Association Studies

- also works for specific logic minimization tasks,
- provides easy to interpret models,
- is the only of the considered methods working on more than 100,000 SNPs,
- provides the best results in the considered applications,
- is very flexible and adaptable,
- provides results on the fly.
Bibliography


