Effects of Constant Optimization by Nonlinear Least Squares Minimization in Symbolic Regression

Michael Kommenda, Gabriel Kronberger, Stephan Winkler, Michael Affenzeller, and Stefan Wagner

Contact:
Michael Kommenda
Heuristic and Evolutionary Algorithms Lab (HEAL)
Softwarepark 11
A-4232 Hagenberg

e-mail: michael.kommenda@fh-hagenberg.at
Web: http://heal.heuristiclab.com
http://heureka.heuristiclab.com
Model a relationship between input variables $x$ and target variable $y$ without any predefined structure

$$y = f(x, w) + \varepsilon$$

Minimization of $\varepsilon$ using an evolutionary algorithm

- Model structure
- Used variables
  - Constants / weights
The correct model structure is found during the algorithm execution, but not recognized due to misleading / wrong constants.
Constants in Symbolic Regression

- **Ephemeral Random Constants**
  - Randomly initialized constants
  - Remain fixed during the algorithm run

- **Evolutionary Constants**
  - Updated by mutation
    - $C_{\text{new}} = C_{\text{old}} + N(0, \sigma)$
    - $C_{\text{new}} = C_{\text{old}} * N(1, \sigma)$

- **Finding correct constants**
  - combination of existing values
  - mutation of constant symbol nodes
    - undirected changes to values

Effects of Constant Optimization by Nonlinear Least Squares Minimization
Faster genetic programming based on **local gradient search** of numeric leaf values (Topchy and Punch, GECCO 2001)

Improving gene expression programming performance by using **differential evolution** (Zhang et al., ICMLA 2007)

**Evolution Strategies** for Constants Optimization in Genetic Programming (Alonso, ICTAI 2009)

**Differential Evolution** of Constants in Genetic Programming Improves Efficacy and Bloat (Mukherjee and Eppstein, GECCO 2012)
Improving Symbolic Regression with Interval Arithmetic and Linear Scaling (Keijzer, EuroGP 2003)

- Use Pearson’s $R^2$ as fitness function and perform linear scaling
  - Removes necessity to find correct offset and scale
  - Computationally efficient

Outperforms the local gradient search
Concept

- Treat all constants as parameters
- Local optimization step
- Multidimensional optimization

Levenberg-Marquardt Algorithm

- Least squares fitting of model parameters to empirical data
- \[ \text{Minimize } Q(\beta) = \sum_{i=1}^{m} [y_i - f(x_i, \beta)]^2 \]
- Uses gradient and Jacobian matrix information
- Implemented e.g. by ALGLIB
**Gradient Calculation**

- **Transformation of symbolic expression tree**
  - Extract initial numerical values (starting point)
  - Add scaling tree nodes

- **Automatic differentiation**
  - Provided e.g. by AutoDiff
  - Numerical gradient calculation in one pass
  - Faster compared to symbolic differentiation

\[
\nabla f = \left( \frac{\partial f}{\partial \beta_1}, \frac{\partial f}{\partial \beta_2}, \ldots, \frac{\partial f}{\partial \beta_n} \right)
\]

- **Update tree with optimized values**
  - Optionally calculate new fitness

Effects of Constant Optimization by Nonlinear Least Squares Minimization
**Improvement** = \( \text{Quality}_{\text{optimized}} - \text{Quality}_{\text{original}} \)

**Exemplary GP Run**

- Average & median improvement stays constantly low
- Maximum improvement almost reaches the best quality found
- Crossover worsens good individuals
- The quality of few individuals can be dramatically increased
### Symbolic regression benchmarks

- Better GP Benchmarks: Community Survey Results and Proposals (White et al., GPEM 2013)

<table>
<thead>
<tr>
<th>Problem</th>
<th>Function</th>
<th>Training</th>
<th>Test</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nguyen-7</td>
<td>$f(x) = \ln(x + 1) + \ln(x^2 + 1)$</td>
<td>20</td>
<td>500</td>
</tr>
<tr>
<td>Keijzer-6</td>
<td>$f(x, y, z) = \frac{30xz}{(x - 10)y^2}$</td>
<td>20</td>
<td>120</td>
</tr>
<tr>
<td>Vladislavleva-4</td>
<td>$f(x_1, \ldots, x_5) = \frac{10}{5 + \sum(x_i - 30)^2}$</td>
<td>1024</td>
<td>5000</td>
</tr>
<tr>
<td>Pagie-1</td>
<td>$f(x, y) = \frac{1}{1 + x^{-4}} + \frac{1}{1 + y^{-4}}$</td>
<td>676</td>
<td>1000</td>
</tr>
<tr>
<td>Poly-10</td>
<td>$f(x_1, \ldots, x_{10}) = x_1x_2 + x_3x_4 + x_5x_6 + x_1x_7x_9 + x_3x_6x_{10}$</td>
<td>250</td>
<td>250</td>
</tr>
<tr>
<td>Friedman-2</td>
<td>$f(x_1, \ldots, x_{10}) = 10 \sin(\pi x_1x_2) + 20(x_3 - 0.5)^2 + 10x_4 + 5x_5 + N(0,1)$</td>
<td>500</td>
<td>5000</td>
</tr>
<tr>
<td>Tower</td>
<td>Real world data</td>
<td>3136</td>
<td>1863</td>
</tr>
</tbody>
</table>

Effects of Constant Optimization by Nonlinear Least Squares Minimization
Algorithm Configurations

Genetic Programming with strict offspring selection
- Only child individuals with better quality compared to the fitter parent are accepted in the new generation

Varying parameters
- Population size of 500, 1000, and 5000 for runs without constant optimization
- Probability for constant optimization 25%, 50%, and 100% (population size 500)

All others parameters were not modified
- Maximum selection pressure of 100 was used as termination criterion
- Size constraints of tree length 50 and depth 12
- Mutation rate of 25%
- Function set consists solely of arithmetic functions (except Nguyen-7)
Results - Quality

Success rate (test $R^2 > 0.99$)

- OSGP 500
- OSGP 1000
- OGSP 5000
- CoOp 25%
- CoOp 50%
- CoOp 100%

Effect of Constant Optimization by Nonlinear Least Squares Minimization
## Results - Quality

**Noisy datasets**

- Success rate not applicable
- $R^2$ of best training solution ($\mu \pm \sigma$)

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Friedman-2</th>
<th>Tower</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Training</td>
<td>Test</td>
</tr>
<tr>
<td>OSGP 500</td>
<td>0.836 ± 0.027</td>
<td>0.768 ± 0.172</td>
</tr>
<tr>
<td>OSGP 1000</td>
<td>0.857 ± 0.036</td>
<td>0.831 ± 0.102</td>
</tr>
<tr>
<td>OSGP 5000</td>
<td>0.908 ± 0.035</td>
<td>0.836 ± 0.191</td>
</tr>
<tr>
<td>CoOp 25%</td>
<td>0.959 ± 0.001</td>
<td>0.871 ± 0.151</td>
</tr>
<tr>
<td>CoOp 50%</td>
<td>0.967 ± 0.000</td>
<td>0.920 ± 0.086</td>
</tr>
<tr>
<td>CoOp 100%</td>
<td>0.964 ± 0.000</td>
<td>0.864 ± 0.142</td>
</tr>
</tbody>
</table>
Results – LM Iterations

Constant optimization probability of 50%
Varying iterations for the LM algorithm (3x, 5x, 10x)

- success rate
- respectively test $R^2$ for noisy datasets

<table>
<thead>
<tr>
<th>Problem</th>
<th>OGSP 5000</th>
<th>CoOp 50% 3x</th>
<th>CoOp 50% 5x</th>
<th>CoOp 50% 10x</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nguyen-7</td>
<td>1.00</td>
<td>0.92</td>
<td>0.92</td>
<td>0.94</td>
</tr>
<tr>
<td>Keijzer-6</td>
<td>0.74</td>
<td>0.92</td>
<td>0.88</td>
<td>0.94</td>
</tr>
<tr>
<td>Vladislavleva-4</td>
<td>0.48</td>
<td>0.56</td>
<td>0.82</td>
<td>0.86</td>
</tr>
<tr>
<td>Pagie-1</td>
<td>0.20</td>
<td>0.26</td>
<td>0.52</td>
<td>0.74</td>
</tr>
<tr>
<td>Poly-10</td>
<td>0.62</td>
<td>0.78</td>
<td>0.88</td>
<td>0.94</td>
</tr>
<tr>
<td>Friedman-2</td>
<td>0.836 ± 0.191</td>
<td>0.946 ± 0.046</td>
<td>0.943 ± 0.076</td>
<td>0.920 ± 0.086</td>
</tr>
<tr>
<td>Tower</td>
<td>0.890 ± 0.009</td>
<td>0.902 ± 0.010</td>
<td>0.912 ± 0.008</td>
<td>0.921 ± 0.006</td>
</tr>
</tbody>
</table>
Feature Selection Problems

**Artificial datasets**
- 100 input variables \( N(0,1) \)
- Linear combination of 10/25 variables with weights \( U(0,10) \)
- noisy \( \rightarrow \) max \( R^2 = 0.90 \)
- Training 120 rows, Test 500 rows
- Population size 500
- Constant optimization 50% 5x

**Observation**
- Constant optimization can lead to overfitting
- Selection of correct features is also an issue
Conclusion

Constant optimization improves the success rate and quality of models
- Better results with smaller population size
- Especially useful for post-processing of models

Removes the effort of evolving correct constants
- Genetic programming can concentrate on the model structure and feature selection

Ready-to-use implementation in HeuristicLab
- Configurable probability, iterations, random sampling
- All experiments available for download
- [http://dev.heuristiclab.com/AdditionalMaterial](http://dev.heuristiclab.com/AdditionalMaterial)
Effects of Constant Optimization by Nonlinear Least Squares Minimization in Symbolic Regression

Michael Kommenda, Gabriel Kronberger, Stephan Winkler, Michael Affenzeller, and Stefan Wagner