Cloud Scale, Machine Learning with FlexGP

Una-May O’Reilly
Evolutionary Design and Optimization Group
Computer Science and Artificial Intelligence Lab
MIT
HELP
Lots of Data Everywhere

The Internet

MOOCs

Healthcare

Engineering
Lots of Data Everywhere
The Cloud’s Role
The Cloud’s Role

+ Elasticity
+ Infinite resources on demand
+ Budget and time choice space

- Robustness
- Time to scale up
- Need interim solutions
- Algorithms need to exploit the positives
Agenda

• Strategies for cloud-scale machine learning with massive data

• FlexGP
  – Flexibly factored, flexibly scaled machine learning with Genetic Programming (GP)
  – Deeper Dives
    » Launch
    » Genetic programming learning engines for ML

• Beyond FlexGP
Strategies for Machine Learning

- Scaled up, existing algorithms are not completely sufficient

People who are really serious about software should make their own hardware. (Allan Kay)

The hardware is the cloud

ML algorithms should be designed with the assumption of infinite resources
Ensembles of Diverse Learners

Ensembles

– factoring
  » Heterogeneous learning engines
    ▪ Training data D
    ▪ Within Algorithm (PI)
      ❖ Model structure
      ❖ Objective
      ❖ Indicators/Explanatory vars
    ▪ Across algorithms

– filtering
  » Diverse models or classifiers or clusters

– Fusion
  » A robust result
Ensembles of Diverse Learners
Distributed sampling approaches

Population

Research Population

Random Samples
Agenda

- Strategies for cloud-scale machine learning
- **FlexGP**
  - Flexibly factored, flexibly scaled machine learning with Genetic Programming (GP)
  - Deeper Dives
    » Launch
    » Genetic programming learning engines for ML
- **Beyond FlexGP**
FlexGP

Signals, State, Ratings, Associations, Rankings, Relations, Categories

Problems
- Pattern Recognition
- Anomaly Detection
- Forecasting
- Planning
- Modeling

Machine Learning Framework
- Feature Extraction
- Statistical Analysis
- Ensemble Techniques
- Cross Validation
- Optimization

Classification
Regression
Clustering

FlexGP
Flexible Factoring and Scaling, Elasticity
Island Distributed Genetic Programming

Cloud

FlexGP Overview
Cloud with Learners

FlexGP Overview
Cloud with Networked Learners

FlexGP Overview
FlexGP Learning Engines

\[ L(\pi, \mathcal{D}) \]

\[ \pi_1 = \{ +, -, *, /, \sin, \cos, \tan, \sqrt{\cdot} \} \]

\[ \pi_2 = L^3 \quad \text{Objectives function} \]

\[ \pi_3 = (x_2, x_3, x_4) \quad \text{Explanatory vars} \]

Model or classifier

\[ \frac{\cos(x_4)}{\tan(x_2) + x_2} + \sqrt{x_3} \]
FlexGP Learning Engines

\[ \pi_1 = \{ + \, - \, * \, / \} \]

\[ \pi_2 = \text{mean squared error (L2)} \]

\[ \pi_3 = (x_1, x_2, x_3, x_5) \]

\[ f(x_1, x_2, x_3) = -\frac{x_2^2}{x_1} \]

<table>
<thead>
<tr>
<th>N</th>
<th>F</th>
<th>r</th>
<th>0.2</th>
<th>Picket for w =</th>
<th>0.4</th>
<th>0.6</th>
<th>0.8</th>
<th>Ave. comp. time (sec)</th>
<th>Ave. # of Pairs optimal schedules</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>10</td>
<td>2</td>
<td>11.30</td>
<td>13.35</td>
<td>11.82</td>
<td>9.54</td>
<td>0.00</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>11.20</td>
<td>12.30</td>
<td>10.84</td>
<td>9.26</td>
<td>0.00</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>12.70</td>
<td>14.36</td>
<td>13.05</td>
<td>11.24</td>
<td>0.06</td>
<td>8</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>11.46</td>
<td>13.22</td>
<td>12.90</td>
<td>10.10</td>
<td>0.05</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>15.81</td>
<td>16.90</td>
<td>15.86</td>
<td>13.82</td>
<td>0.03</td>
<td>6</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>14.68</td>
<td>15.61</td>
<td>15.14</td>
<td>12.36</td>
<td>0.02</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>2</td>
<td>10.46</td>
<td>11.48</td>
<td>10.23</td>
<td>8.92</td>
<td>0.00</td>
<td>14</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10.00</td>
<td>10.86</td>
<td>9.89</td>
<td>8.82</td>
<td>0.00</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>10.90</td>
<td>11.86</td>
<td>10.89</td>
<td>9.82</td>
<td>0.00</td>
<td>16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10.46</td>
<td>11.36</td>
<td>10.30</td>
<td>9.30</td>
<td>0.00</td>
<td>17</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>11.46</td>
<td>12.36</td>
<td>11.41</td>
<td>10.30</td>
<td>0.00</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>11.32</td>
<td>12.26</td>
<td>11.10</td>
<td>9.92</td>
<td>0.00</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>120</td>
<td>10</td>
<td>2</td>
<td>10.46</td>
<td>11.48</td>
<td>10.23</td>
<td>8.92</td>
<td>0.00</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>8.23</td>
<td>9.66</td>
<td>7.96</td>
<td>6.96</td>
<td>0.00</td>
<td>17</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>8.84</td>
<td>9.75</td>
<td>8.65</td>
<td>7.32</td>
<td>0.00</td>
<td>13</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>7.86</td>
<td>8.03</td>
<td>6.96</td>
<td>6.50</td>
<td>0.00</td>
<td>15</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>10.78</td>
<td>10.90</td>
<td>10.59</td>
<td>8.60</td>
<td>0.00</td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>9.86</td>
<td>10.29</td>
<td>9.04</td>
<td>7.82</td>
<td>0.00</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>2</td>
<td>10.46</td>
<td>11.48</td>
<td>10.23</td>
<td>8.92</td>
<td>0.00</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>6.23</td>
<td>7.86</td>
<td>6.76</td>
<td>5.76</td>
<td>0.00</td>
<td>13</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>2</td>
<td>7.12</td>
<td>8.66</td>
<td>7.28</td>
<td>6.32</td>
<td>0.00</td>
<td>18</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>5.95</td>
<td>8.26</td>
<td>6.28</td>
<td>5.89</td>
<td>0.00</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>2</td>
<td>9.19</td>
<td>9.81</td>
<td>8.23</td>
<td>7.06</td>
<td>0.10</td>
<td>17</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2.29</td>
<td>5.91</td>
<td>4.26</td>
<td>3.35</td>
<td>0.11</td>
<td>18</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
FlexGP Learning Engines

\[ \pi_1 = \{ +, -, *, /, \sin, \cos, \tan, \sqrt{\cdot} \} \]

\[ \pi_2 = L_3 \]

\[ \pi_3 = (x_2 \times x_3 \times x_4) \]

<table>
<thead>
<tr>
<th>( N )</th>
<th>( F )</th>
<th>( t )</th>
<th>( NR )</th>
<th>( \text{Av. run time}) (sec)</th>
<th>( \text{Av. no. of Pareto optimal schedules} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>10</td>
<td>2</td>
<td>11.30</td>
<td>13.25</td>
<td>11.62</td>
</tr>
<tr>
<td>4</td>
<td>11.30</td>
<td>12.36</td>
<td>10.94</td>
<td>9.26</td>
<td>0.00</td>
</tr>
<tr>
<td>14</td>
<td>12.78</td>
<td>14.26</td>
<td>13.00</td>
<td>11.24</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>11.46</td>
<td>13.32</td>
<td>12.30</td>
<td>10.10</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>15.81</td>
<td>16.56</td>
<td>15.68</td>
<td>13.62</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>14.68</td>
<td>15.61</td>
<td>15.14</td>
<td>12.35</td>
<td>0.00</td>
</tr>
<tr>
<td>100</td>
<td>10</td>
<td>2</td>
<td>11.46</td>
<td>11.46</td>
<td>10.37</td>
</tr>
<tr>
<td>4</td>
<td>10.06</td>
<td>10.06</td>
<td>9.26</td>
<td>8.05</td>
<td>0.00</td>
</tr>
<tr>
<td>14</td>
<td>12.66</td>
<td>14.26</td>
<td>13.00</td>
<td>11.24</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>11.46</td>
<td>13.32</td>
<td>12.30</td>
<td>10.10</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>16.81</td>
<td>16.56</td>
<td>15.68</td>
<td>13.62</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>15.68</td>
<td>15.61</td>
<td>15.14</td>
<td>12.35</td>
<td>0.00</td>
</tr>
<tr>
<td>120</td>
<td>10</td>
<td>2</td>
<td>11.30</td>
<td>12.36</td>
<td>11.62</td>
</tr>
<tr>
<td>4</td>
<td>11.30</td>
<td>12.36</td>
<td>11.62</td>
<td>9.54</td>
<td>0.00</td>
</tr>
<tr>
<td>14</td>
<td>6.22</td>
<td>6.00</td>
<td>6.09</td>
<td>6.00</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>6.00</td>
<td>6.00</td>
<td>6.09</td>
<td>6.00</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>7.88</td>
<td>9.03</td>
<td>4.36</td>
<td>7.00</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>8.00</td>
<td>9.03</td>
<td>4.36</td>
<td>7.00</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>10.38</td>
<td>10.54</td>
<td>9.04</td>
<td>8.62</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>8.88</td>
<td>10.54</td>
<td>9.04</td>
<td>8.62</td>
<td>0.00</td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>2</td>
<td>12.36</td>
<td>13.25</td>
<td>11.62</td>
</tr>
<tr>
<td>4</td>
<td>12.36</td>
<td>13.25</td>
<td>11.62</td>
<td>9.54</td>
<td>0.00</td>
</tr>
<tr>
<td>14</td>
<td>7.12</td>
<td>8.62</td>
<td>6.38</td>
<td>6.32</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>5.75</td>
<td>8.62</td>
<td>6.38</td>
<td>6.32</td>
<td>0.00</td>
</tr>
<tr>
<td>18</td>
<td>5.19</td>
<td>9.59</td>
<td>8.03</td>
<td>7.96</td>
<td>0.00</td>
</tr>
<tr>
<td>4</td>
<td>4.49</td>
<td>9.59</td>
<td>8.03</td>
<td>7.96</td>
<td>0.00</td>
</tr>
</tbody>
</table>
FlexGP Ensemble Fusion

\[ x_1 \sin(x_5) + x_2 \sqrt{x} \]
\[ \cos(x_4) / \sin(x_2) + \sqrt{x_3 - x_4} \]
\[ \frac{x_1}{\exp(x_2)} + x_5 x_3 + \frac{x_2}{x_3} \]
\[ \frac{\cos(x_4)}{\tan(x_2) + x_2} + \sqrt{x_3} \]

Filter to select diverse models

Fusion to derive an ensemble prediction

FlexGP Overview
FlexGP Demonstrated

Build a classifier that can discriminate between these signals?

Drop=0
Replicate 1
Not lubricated

Drop=3
Replicate 1

Drop =6
Replicate 1
Fully lubricated

Before Learning

After Learning

There is still scope
Agenda

• Strategies for cloud-scale machine learning
• FlexGP
  – Flexibly factored, flexibly scaled machine learning with Genetic Programming (GP)
  – Deeper Dives
    » Launch
    » Genetic programming learning engines for ML
• Beyond FlexGP
Cascading, Asynchronous Launch

“Start” node initiates recursive local launches
- Inputs are distributions of $\pi, D$ and cascading values: $N, k \rightarrow cl$

Each node
- **Phase 1:** launch $k$ other nodes if $cl > 0$
  - Each child is sent distributions $\pi, D$ and $k$, $cl = cl - 1$
  - Each child is sent ancestors’ IPs: IP-list

- **Phase 2:**
  - Thread 1: global IP discovery through gossip
    » Select an IP, dispatch IP-list
    » Return IP-list to any sender
  - Thread 2: $L(\pi, D)$ after sampling from distributions
π, D, N, k → 128.21.32.237
π, D, N, k → 128.21.32.237

π, D, k, cl, [IP-list]

128.21.32.238

π, D, k, cl, [IP-list]

128.21.32.239

FlexGP Launch
FlexGP Launch

128.21.32.238

128.21.32.239

Launch

Gossip

L(π, D)
FlexGP Launch
FlexGP Launch

Launch

Gossip

$L(\pi, D)$

128.21.32.123
128.21.31.512
128.21.31.542
128.21.31.6 12
128.21.31.332
128.21.31.832
128.21.31.812
128.21.41.832
FlexGP Launch

Launch
Gossip

$\mathcal{L}(\pi, \mathcal{D})$

128.21.32.123
128.21.31.512
128.21.31.542
128.21.31.612
128.21.31.332
128.21.31.832
128.21.41.832
FlexGP Launch

Launch

Gossip

$L(\pi, \mathcal{D})$
Launch complete!

... and ready to expand or contract (gossiping intermittently)
Genetic Programming

Goal: Model \( y = f(x_1, x_2, \ldots, x_n) \)

Form GP Trees
\[ \{+, -, *, /, \log, \sqrt{}, \exp\} \]
\[ \{x_1, x_2, \ldots, x_n\} \]

Execute trees

New trees

Selection and variation of trees

\[
\left( 2.2 - \left( \frac{x_2}{11} \right) \right) + (7 \times \cos(x_1))
\]

Transparent expression

GP Learning Engine
GP Tree Crossover

Parent 1

Child 1

Parent 2

Child 2

GP Learning Engine
Learning a classifier

\[ \log(x_1) + \exp(-x_2) + x_3 = [y_{gp}] \]

Area of overlap

\[ P(y_{gp} | C_2) \]

\[ P(y_{gp} | C_1) \]
Learning a classifier

$$x_1^2 + \exp(-x_2) + \log(x_3) = [y_{gp}]$$
Learning a regression model

\[ \log(x_1) + \exp(-x_2) + x_3 = [y_{gp}] \]

\[ \hat{y}_s = \frac{y_{gp} - y_{gpmin}}{y_{gpmax} - y_{gpmin}} \]

\[ \| \hat{y}_s - y_s \|_p \]

Scaling model via linear regression

Minimize p-norm

GP Algorithm Development
FlexGP Regression Model Diversity

Correlation of 1477 Individuals with MSE $\leq 0.0505$
FlexGP…

Is:
Flexibly factored, aggregating ML system
• Cascading launch
• Distributed scalable network protocol
• Cloud scale ensemble learning method

Delivers:
• Elasticity
• Scalability in computation size
• Large data strategy
• Innovation in machine learning with evolutionary computation
Automation

- "In the end, the biggest bottleneck is not data or CPU cycles, but human cycles."

Beyond FlexGP
Mass Customized Query Serving

Waveform database

<table>
<thead>
<tr>
<th>Feature</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total size</td>
<td>4 Terra Bytes</td>
</tr>
<tr>
<td>Waveform types</td>
<td>22</td>
</tr>
<tr>
<td>Signal sampling frequency</td>
<td>125 samples/sec</td>
</tr>
<tr>
<td>Number of samples</td>
<td>500m</td>
</tr>
</tbody>
</table>

Beyond FlexGP
Personalized Query Serving

Parameterizations:
- \( m \)- hours of past data used to forecast
- \( f \)- forecast window, lag
- \( p \)- period of forecast

\[ T_0 - m \text{ hours} \quad T_0 \quad T_0 + f \quad T_0 + f + p \]

Memory: sliding window
Forecast period
Forecast window
Fundamental Learning

- When the data overwhelms us...
  - We bundle it up
    » nb, this is not sampling!
  - We assume linearity and Gaussian distributions

- What are the intrinsic aggregations?
- What are the non-linearities and true distributions?

- Fundamental learning starts from the bottom up
  - Use unsupervised learning to propose features
  - Use features in a task
  - Pass performance feedback to feature learning
A time trajectory of GP-based machine learning
A time trajectory of GP-based machine learning
A time trajectory of GP-based machine learning
Acknowledgements

• Members of the Evolutionary Design and Optimization Group
  – Past and present
  – Dr. Kalyan Veeramachaneni: Research Scientist

GE Global Research

Industrial Machine Learning Lab, GEGR, Niskayuna