Genetic Programming and Symbolic Regression

Trent McConaghy, PhD
Mysteries of the universe..

What does AI encompass?

Is Deep Learning cool or what?

WTF is genetic programming or symbolic regression? Why should I care?

How does Google find furry robots?
What *is* AI anyway?
Classification, in 2D

Credit profile:
- Paid bills
- Didn’t pay
Regression, in 1D

Age

Job satisfaction

Age

Job satisfaction
Regression, in 2D

How: Polynomials, splines, neural networks, support vector machines, Gaussian process models, boosted trees, ... [many refs]
Symbolic Regression (SR)
(Like regression, but output a symbolic model too)

Symbolic model:
\[ y = 50.2 + 9.1 \times x + 3.2 \times \max(0, x^2) \]

[e.g. McConaghy 2005; McConaghy 2011]
Example: SR on Circuits

![Circuit Diagram]

<table>
<thead>
<tr>
<th>Perf.</th>
<th>Expression</th>
</tr>
</thead>
</table>
| $A_{LF}$ | $-10.3 + 7.08e-5 / id1$  
$+ 1.87 * \ln( -1.95e+9 + 1.00e+10 / (vsg1*vsg3) + 1.42e+9 *(vds2*vsd5) / (vsg1*vgs2*vsg5*id2))$ |
| $f_u$ | $10^\left( 5.68 - 0.03 * vsg1 / vds2 - 55.43 * id1+ 5.63e-6 / id1 \right)$ |
| PM | $90.5 + 190.6 * id1 / vsg1 + 22.2 * id2 / vds2$ |
| $V_{offset}$ | $-2.00e-3$ |
| $SR_p$ | $2.36e+7 + 1.95e+4 * id2 / id1 - 104.69 / id2 + 2.15e+9 * id2 + 4.63e+8 * id1$ |
| $SR_n$ | $-5.72e+7 - 2.50e+11 * (id1*id2) / vgs2 + 5.53e+6 * vds2 / vgs2 + 109.72 / id1$ |

[McConaghy 2005]
SR Problem Definition, Redux

• Given \((X,y)\)

• Find whitebox *models*

• That minimize *error-complexity tradeoff*
AI Has a Toolbox of Ways to Solve...

- Classification – Fraud detection, spam filtering ...
- Regression – Stock prediction, sensitivity analysis ...
- Whitebox regression – Scientific discovery ...
- Optimization – Airfoil design, circuit simulation ...
- Structural synthesis – Analog synthesis, robotics ...
- Pattern recognition – Face recognition, object recog ...
- System identification – Scientific discovery ...
- Ranking – Web search, ad serving, social discovery ...
- Control – Auto-driving autos, spacecraft trajectories ...
- ...
AI Sub-fields

- machine learning
- neural networks
- evolutionary computation
- fuzzy logic
- data mining
- artificial general intelligence
- pattern recognition
- ..
- (nee) nonlinear programming
- (nee) databases
- ..
AI Sub-fields of sub fields

• machine learning + neural networks
  – recurrent neural networks
  – sparse linear regression
  – deep learning
  – ..

• evolutionary computation
  – evolutionary programming, evolution strategies
  – genetic algorithms
  – genetic programming

• ..
Genetic Programming (GP):
A branch of a branch of AI
But a super-cool one..
The Cycle of Evolution

Replication → Selection → Fitness → Mutation → Replication
Genetic Programming

How can we get a computer to know what to do?

Genetic Intelligence

High Return Human Competitive

Machine Intelligence

That will never work!

Scalable

Reuse

Problem Independent

Divide + Conquer

10 - 1 hour's law jump

Mathematical discontinuity

REUSE

REUSE

PROBABILISTIC

Decompose problems into subproblems

Red

Divide and Conquer

Evolution driven by high level specs

Amenable to Beowulf

15% of companies exploring for use

Leading edge

-Brad Holtz

Logic Based Representations

From toy problems to G.P. creating programs automatically creating random programs

Representation

Selection

Evolution

50's

C.O.F.E.S

Shift to
GP for SR

“A function is a tree”

\[ f(x) = 4.8x_3 + \sqrt{x_2} \]

Searches through the space of trees:

1. Initial random population; evaluate
2. Create children from parents via operators; evaluate
3. Select best; goto 2
GP for SR: Crossover Operator

[Diagram showing the crossover of two GP trees]
SR with Vanilla GP.. And Problems

\[
(+ (- (% (RLOG (COS X))) (* (RLOG 0.48800004)
  (* (+ (- X X) (COS -0.8))
  X))))
\]
\[
(- (COS -0.8) (COS -0.8)))
\]
\[
(* (COS (- (COS (COS (+ (RLOG X)
(RLOG (COS X))))))
(RLOG X)))))
\]
\[
(* (COS (- (COS -0.8) (RLOG X))))
\]
\[
(* (- (% (RLOG (COS X)))
(* (RLOG 0.48800004)
  (* (+ (- X X) (COS -0.8)) X)))
(SIN X))
(RLOG (COS (RLOG X))))))).
\]
SR with Vanilla GP.. And Problems (2)

(+ (* (* -0.5403 (+ 0.5741 -0.8861)) (% (*
0.29690000000000016 0.0808999999999997) (+ (% (% (-
-0.5962000000000001 0.3902000000000001) (- (+ (% (* (+(*
0.23550000000000004 0.1506000000000007) (* *
-0.1028999999999999 -0.7332) 0.7723)) (*
0.23550000000000004 0.1506000000000007)) + 0.6026 (+ (+
(% (- 0.37250000000000005 -0.3490999999999997) (- -0.776
-0.6013)) (- -0.5250999999999999 -0.0900000000000008))
(% (- 0.29690000000000016 -0.3490999999999997) (- -0.776
-0.6013))))) (* (+ -0.8661 (% -0.6019999999999992
0.051100000000000145)) (% -0.6019999999999992
0.051100000000000145))) (% -0.49659999999999993 0.4475))
(+ (% (* (+ -0.1943999999999999 0.4366000000000001) (*
0.23550000000000004 0.1506000000000007)) + 0.6026 (* (*
(+ (* -0.5403 -0.0171999999999993) (%
-0.06019999999999992 0.051100000000000145)) (% (* (+
-0.1943999999999999 0.4366000000000001) (*
0.23550000000000004 0.1506000000000007)) (%
0.42100000000000004 -0.4275) (- -0.4816000000000003
0.5708))) (0.7723)) (- -0.8395 -0.1986)) (- (-
0.37250000000000005 -0.3490999999999997) (- -0.776
-0.6013))))) (% (+ 0.6698000000000002
0.8714000000000002) (% (- -0.829 -0.636) (-
0.7635000000000001 -0.1589999999999992)) (- (- (*
-0.5403 -0.0171999999999993) (- -0.8395 -0.1986)) (- (*
(* -0.5403 -0.0171999999999993) (- 0.6004 -0.4343)) (-
-0.951 (% (* 0.7803 0.9777) 0.31920000000000015)))))
(+ (* (* -0.5403 -0.0171999999999993
-0.1924000000000002) (+ (+ -0.1333999999999996 0.7944)
0.6004)).).

[Koza1992]
CAFFEINE Approach

CAFFEINE = _Canonical form functions in evolution_

Grammar to describe the canonical forms:

```
REPVC => VC | REPVC * REPOP | REPOP
REPOP => REPOP * REPOP | OP_1ARG ( W + REPADD ) | OP_2ARG ( 2ARGS ) | ... 3OP, 4OP
2ARGS => W + REPADD, MAYBEW | MAYBEW, W + REPADD
MAYBEW => W | W + REPADD
REPADD => W * REPVC | REPADD + REPADD
OP_2ARG => DIVIDE | POW | MAX | ...
OP_1ARG => INV | LOG10 | ...
```

Search the space with **grammatically-constrained GP** [Whig1995]
Benchmarks: Experimental Setup

- High Speed amplifier
- 13 design variables
  - $V_{ds}$, $V_{gs}$, $I_{ds}$ (operating-point driven formulation)
- orthogonal hypercube sampling
- 243 training samples
- 243 testing samples
Example: GP for SR on Circuits

<table>
<thead>
<tr>
<th>Perf.</th>
<th>Expression</th>
</tr>
</thead>
</table>
| $A_{LF}$ | -10.3 + 7.08e-5 / id1  
|        | + 1.87 * ln(-1.95e+9 + 1.00e+10 / (vsg1*vsg3) + 1.42e+9 *(vds2*vsd5) / (vsg1*vgs2*vsg5*id2)) |
| $f_u$  | 10^(5.68 - 0.03 * vsg1 / vds2 - 55.43 * id1 + 5.63e-6 / id1) |
| PM    | 90.5 + 190.6 * id1 / vsg1 + 22.2 * id2 / vds2 |
| $V_{offset}$ | - 2.00e-3 |
| $SR_p$ | 2.36e+7 + 1.95e+4 * id2 / id1 - 104.69 / id2 + 2.15e+9 * id2 + 4.63e+8 * id1 |
| $SR_n$ | - 5.72e+7 - 2.50e+11 * (id1*id2) / vgs2 + 5.53e+6 * vds2 / vgs2 + 109.72 / id1 |

[McConaghy 2005]
<table>
<thead>
<tr>
<th>Perf.</th>
<th>Target % error</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>A&lt;sub&gt;LF&lt;/sub&gt;</td>
<td>10 10</td>
<td>$-10.3 + 7.08\times10^{-5} / id1 + 1.87 \times \ln\left(-1.95\times10^9 + 1.00\times10^{10} / (vsg1\times vsg3) + 1.42\times10^9 \times (vds2\times vsd5) / (vsg1\times vgs2\times vsg5\times id2)\right)$</td>
</tr>
<tr>
<td>f&lt;sub&gt;U&lt;/sub&gt;</td>
<td>10 10</td>
<td>$10^{(5.68 - 0.03 \times vsg1 / vds2 - 55.43 \times id1 + 5.63\times10^{-6} / id1)}$</td>
</tr>
<tr>
<td>PM</td>
<td>10 10</td>
<td>$90.5 + 190.6 \times id1 / vsg1 + 22.2 \times id2 / vds2$</td>
</tr>
<tr>
<td>v&lt;sub&gt;offset&lt;/sub&gt;</td>
<td>10 10</td>
<td>$-2.00\times10^{-3}$</td>
</tr>
<tr>
<td>SR&lt;sub&gt;p&lt;/sub&gt;</td>
<td>10 10</td>
<td>$2.36\times10^7 + 1.95\times10^4 \times id2 / id1 - 104.69 / id2 + 2.15\times10^9 \times id2 + 4.63\times10^8 \times id1$</td>
</tr>
<tr>
<td>SR&lt;sub&gt;n&lt;/sub&gt;</td>
<td>10 10</td>
<td>$-5.72\times10^7 - 2.50\times10^{11} \times (id1\times id2) / vgs2 + 5.53\times10^6 \times vds2 / vgs2$</td>
</tr>
</tbody>
</table>

CAFFEINE models with <10% error
CAFFEINE Prediction Performance

Predicts better than several state-of-the-art blackbox regression techniques on circuits benchmark suite (and gives whitebox models).

![Graph showing prediction error comparison between different models](image-url)

- Constant (Ref.)
- Linear
- Stepwise Posynomial
- Projection-based quadratic
- Full quadratic
- MARS (Stepwise PWP)
- CAFFEINE
- FFNN
- Boosted FFNN
- SVM
- Kriging

Legend:
- fu
- offsetn
- srp
- srn
- lfgain
- pm

Avg. prediction error
Conclusion

What does AI encompass?

Is Deep Learning cool or what?

WTF is genetic programming or symbolic regression? Why should I care?

How does Google find furry robots?