Genetic Programming and Symbolic Regression

Trent McConaghy, PhD

ascribe

solido
DESIGN AUTOMATION

Mysteries of the universe..

What does AI encompass?

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

Is Deep Learning cool or what?

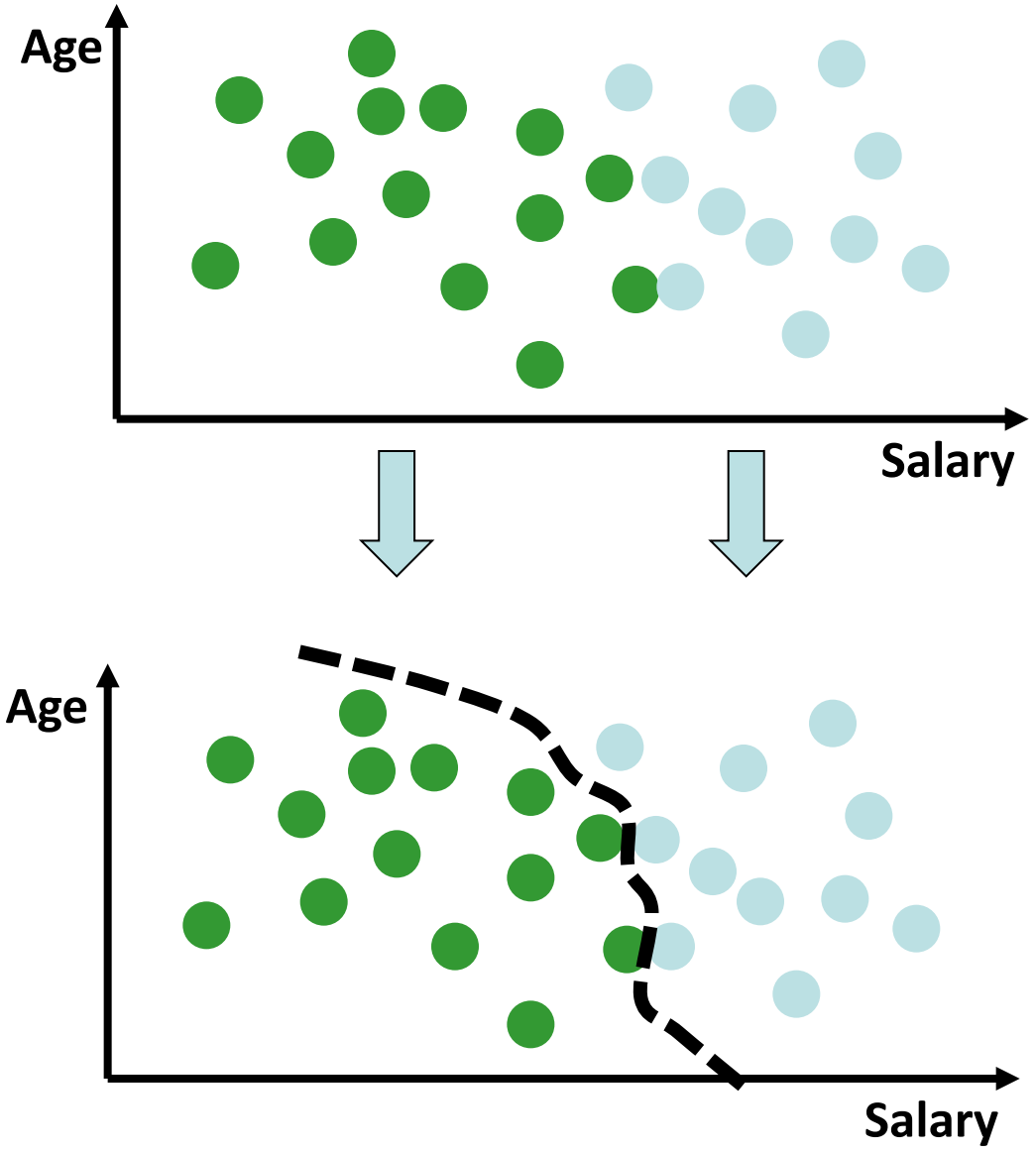
How *does* Google find furry robots?

What *is* AI anyway?

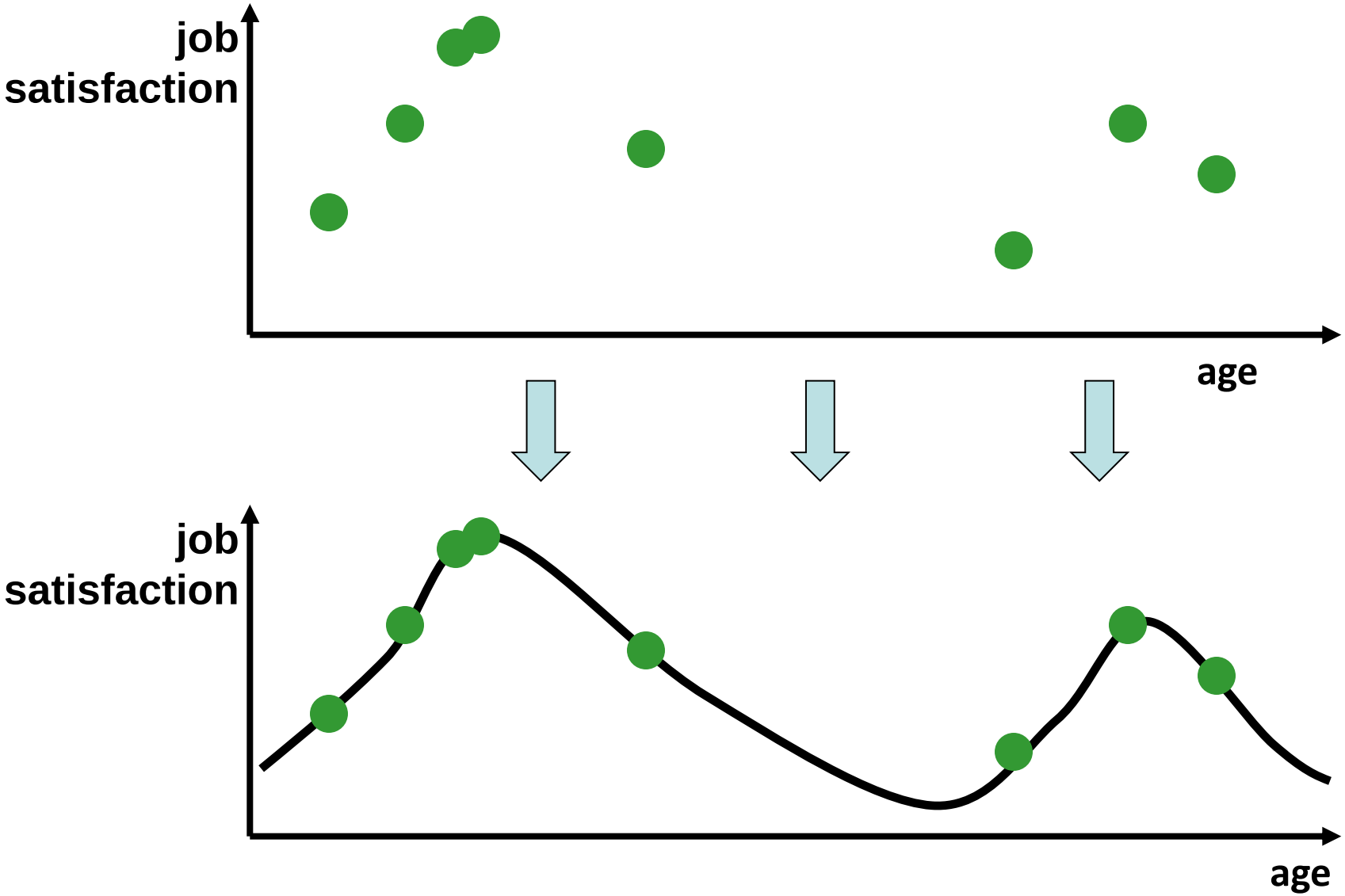
Classification, in 2D

Credit profile:

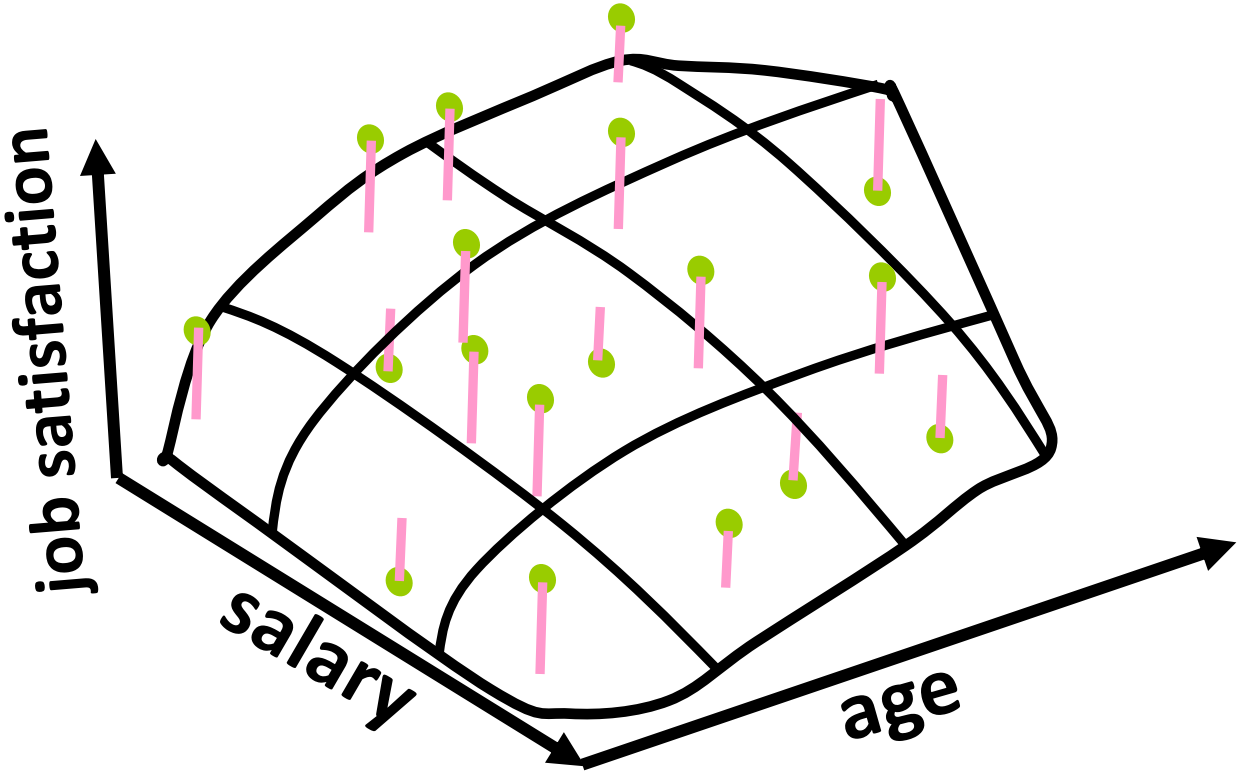
- **Paid bills**
- **Didn't pay**



Regression, in 1D



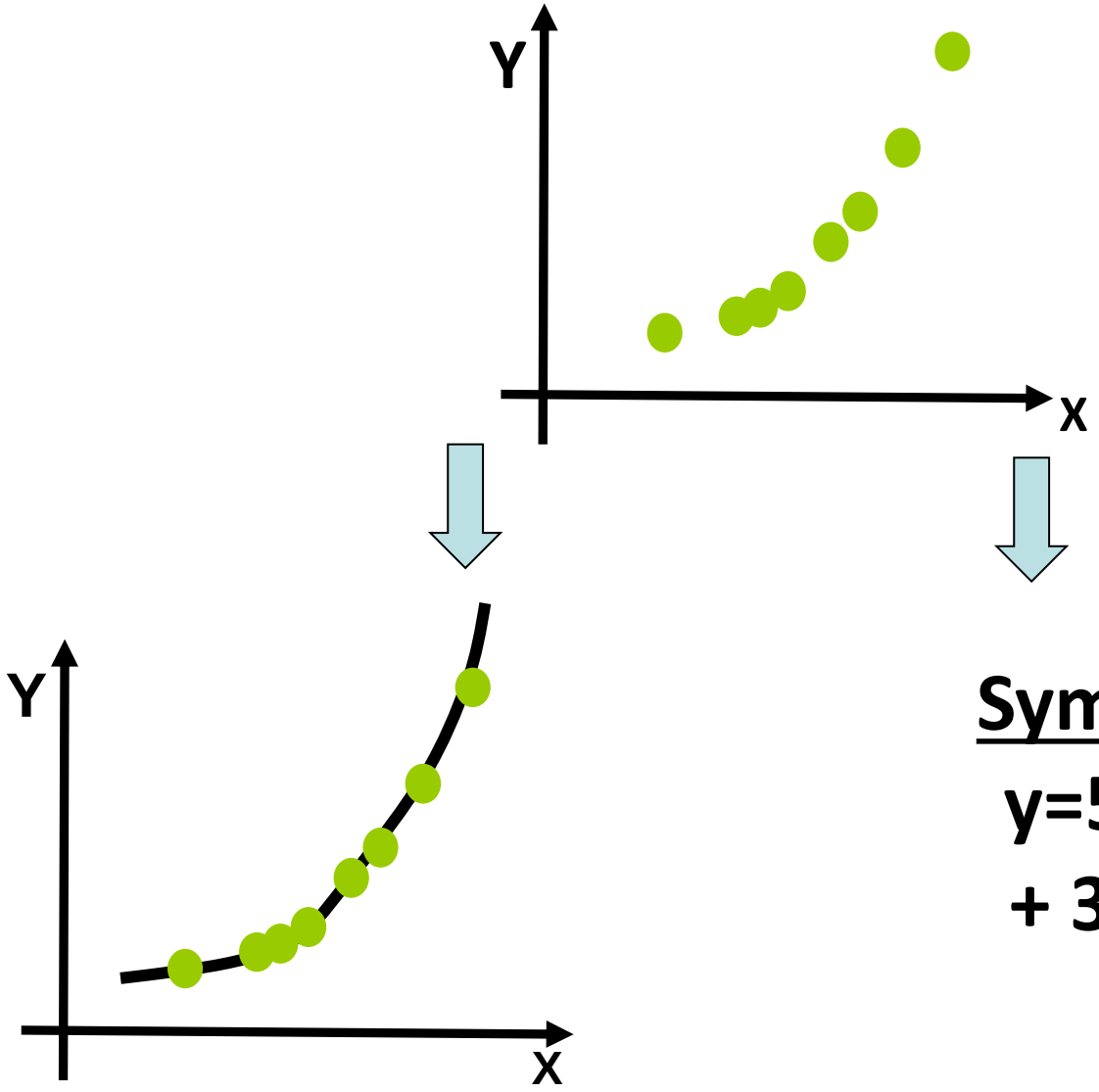
Regression, in 2D



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)

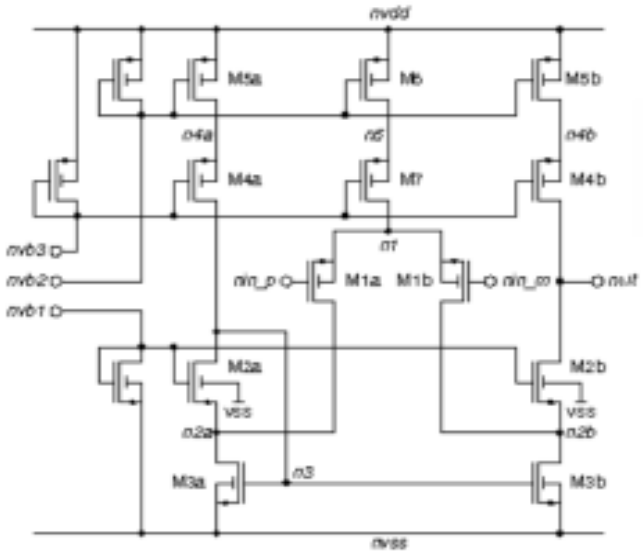


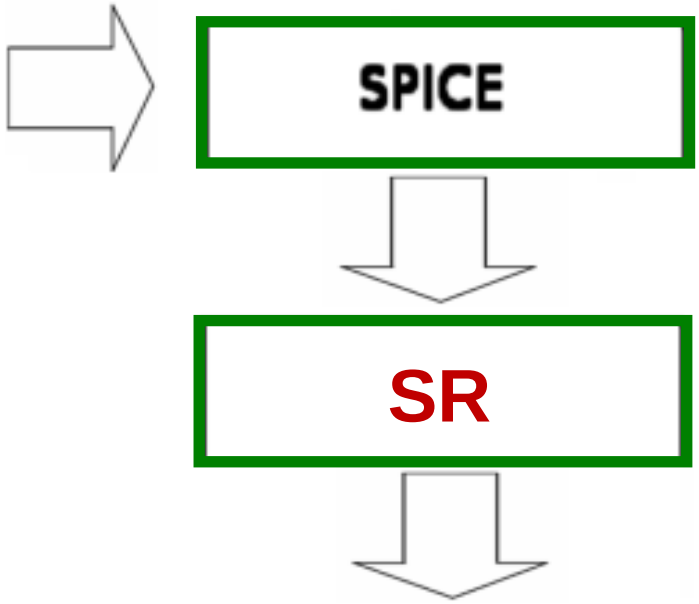
Symbolic model:

$$y = 50.2 + 9.1 \cdot x + 3.2 \cdot \max(0, x^2)$$

[e.g. McConaghy 2005; McConaghy 2011]

Example: SR on Circuits



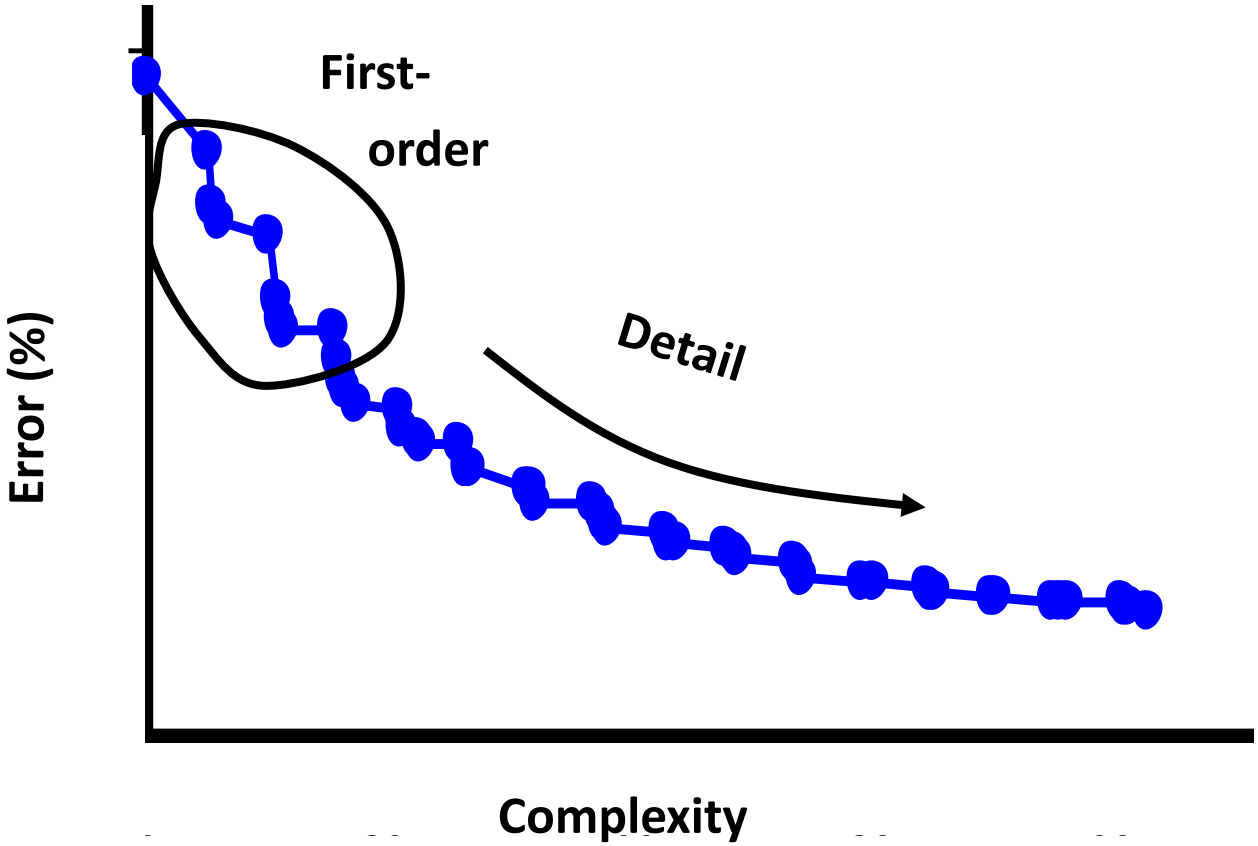


Perf.	Expression
A_{LF}	-10.3 + 7.08e-5 / id1 + 1.87 * ln(-1.95e+9 + 1.00e+10 / (vsg1*vsg3)+ 1.42e+9 *(vds2*vsd5) / (vsg1*vgs2*vsg5*id2))
f_u	10^(5.68 - 0.03 * vsg1 / vds2 - 55.43 * id1+ 5.63e-6 / id1)
PM	90.5 + 190.6 * id1 / vsg1 + 22.2 * id2 / vds2
v_{offset}	- 2.00e-3
SR_p	2.36e+7 + 1.95e+4 * id2 / 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]

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

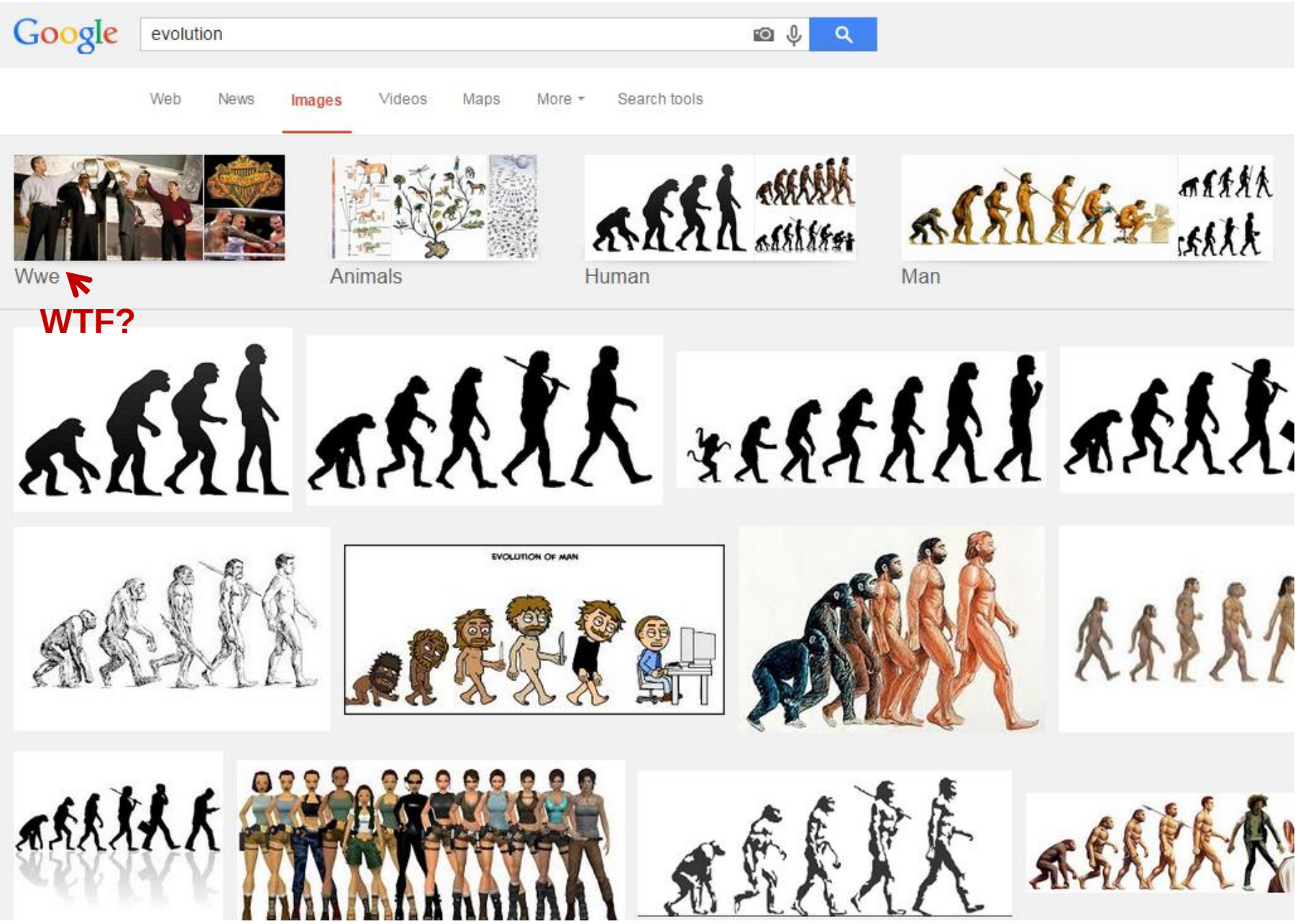
- machine learning
- neural networks
- evolutionary computation
- fuzzy logic
- data mining
- artificial general intelligence
- pattern recognition
- ..
- (nee) nonlinear programming
- (nee) databases
- ..

AI Sub-fields of sub fields

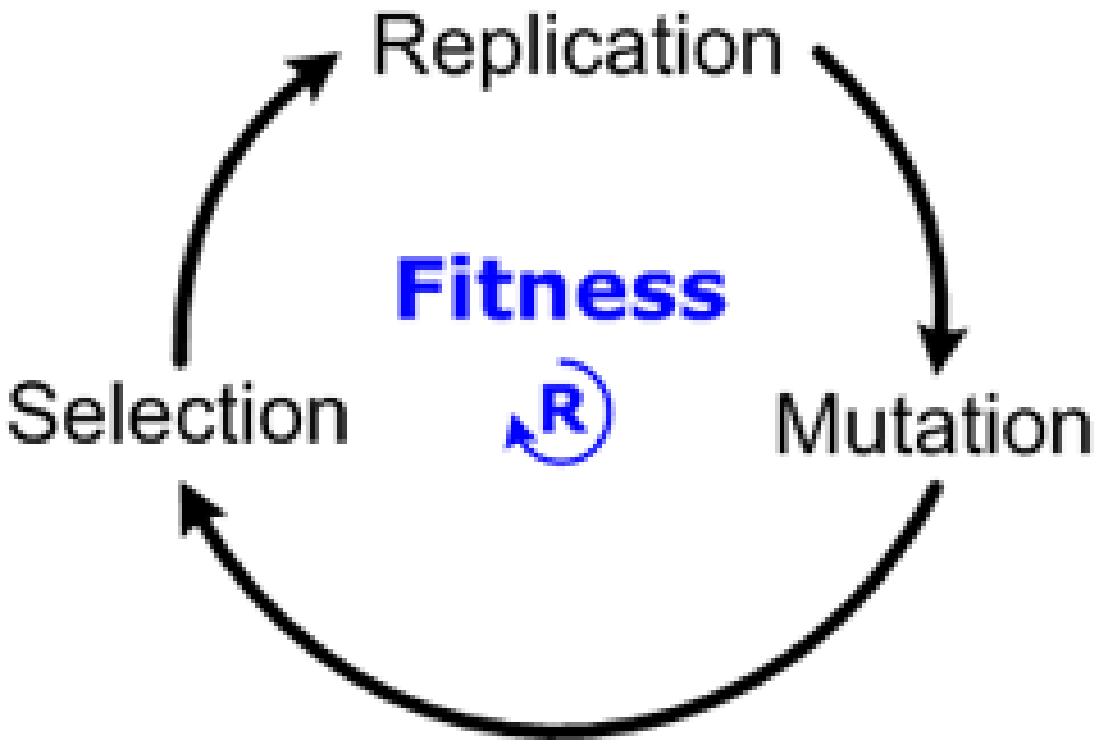
- machine learning + neural networks
  - recurrent neural networks
  - sparse linear regression
  - deep learning
  - ..
- evolutionary computation
  - evolutionary programming, evolution strategies
  - genetic algorithms
  - genetic programming
- ..

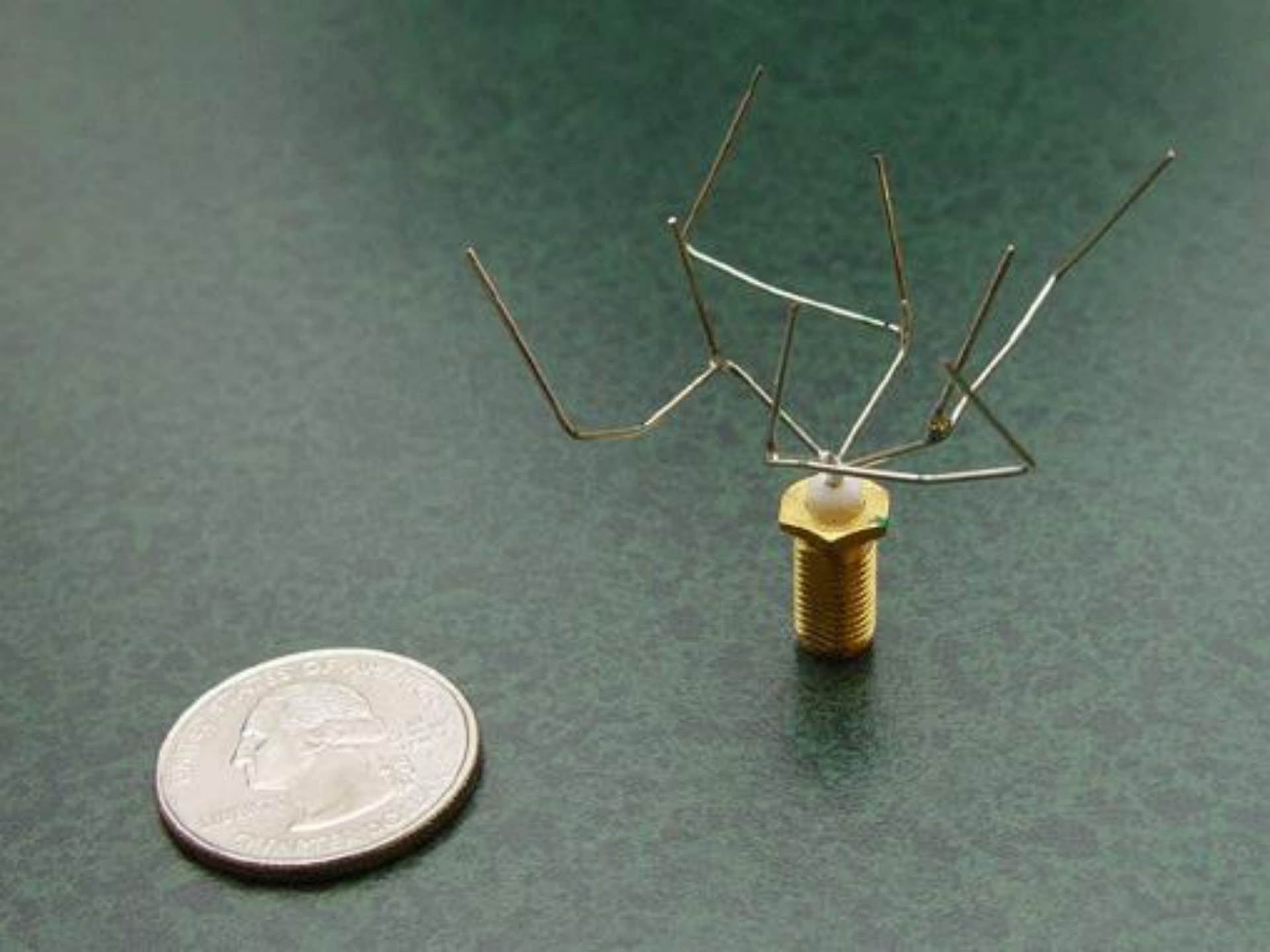
Genetic Programming (GP):
A branch of a branch of AI
But a super-cool one..

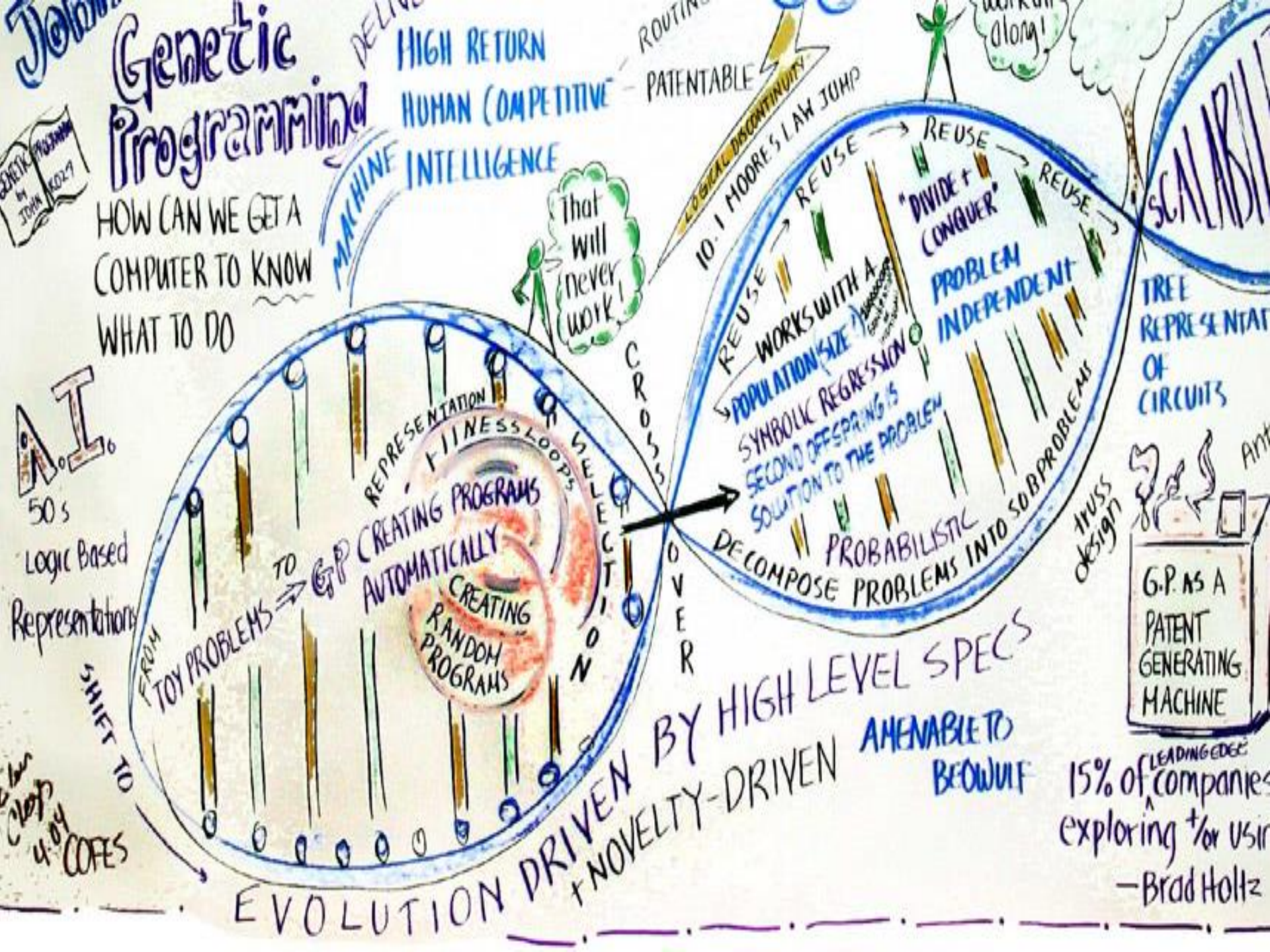
Evolution


Google
evolution
Web
News
Images
Videos
Maps
More
Search tools
Wwe
Animals
Human
Man
WTF?
EVOLUTION OF MAN

The Cycle of Evolution



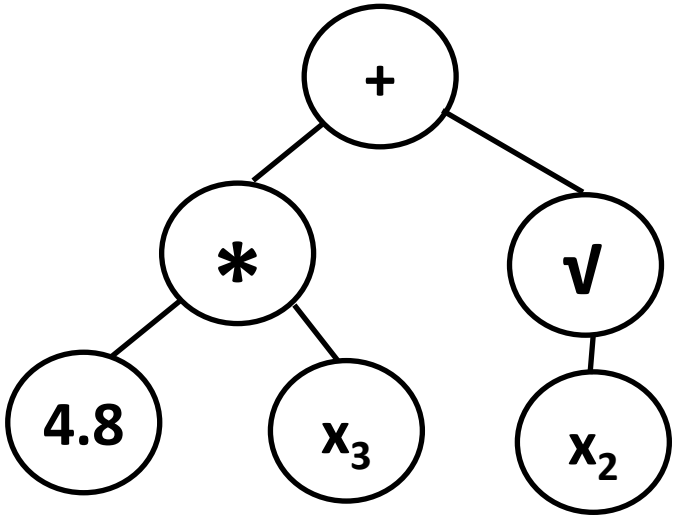



Genetic Programming
HOW CAN WE GET A COMPUTER TO KNOW WHAT TO DO
MACHINE INTELLIGENCE
HIGH RETURN
HUMAN COMPETITIVE
PATENTABLE
ROUTINE
That will never work!
LOGICAL DISCONTINUITY
10.1 MOORE'S LAW JUMP
AI
50's
Logic Based
Representations
SHIFT TO
FROM TOY PROBLEMS TO GP CREATING PROGRAMS AUTOMATICALLY
REPRESENTATION
FITNESS LOOPS
SELECTION
CREATING RANDOM PROGRAMS
CROSS OVER
REUSE
WORKS WITH A POPULATION SIZE
SYMBOLIC REGRESSION
SECOND OFFSPRING IS SOLUTION TO THE PROBLEM
"DIVIDE & CONQUER"
PROBLEM INDEPENDENT
PROBABILISTIC DECOMPOSE PROBLEMS INTO SUBPROBLEMS
SCALABI
TREE REPRESENTATION OF CIRCUITS
truss design
G.P. AS A PATENT GENERATING MACHINE
EVOLUTION DRIVEN BY HIGH LEVEL SPECS
+NOVELTY-DRIVEN
AMENABLE TO BEOWULF
15% of LEADING EDGE companies
exploring +/or usin
—Brad Holtz

GP for SR

"A function is a *tree*"

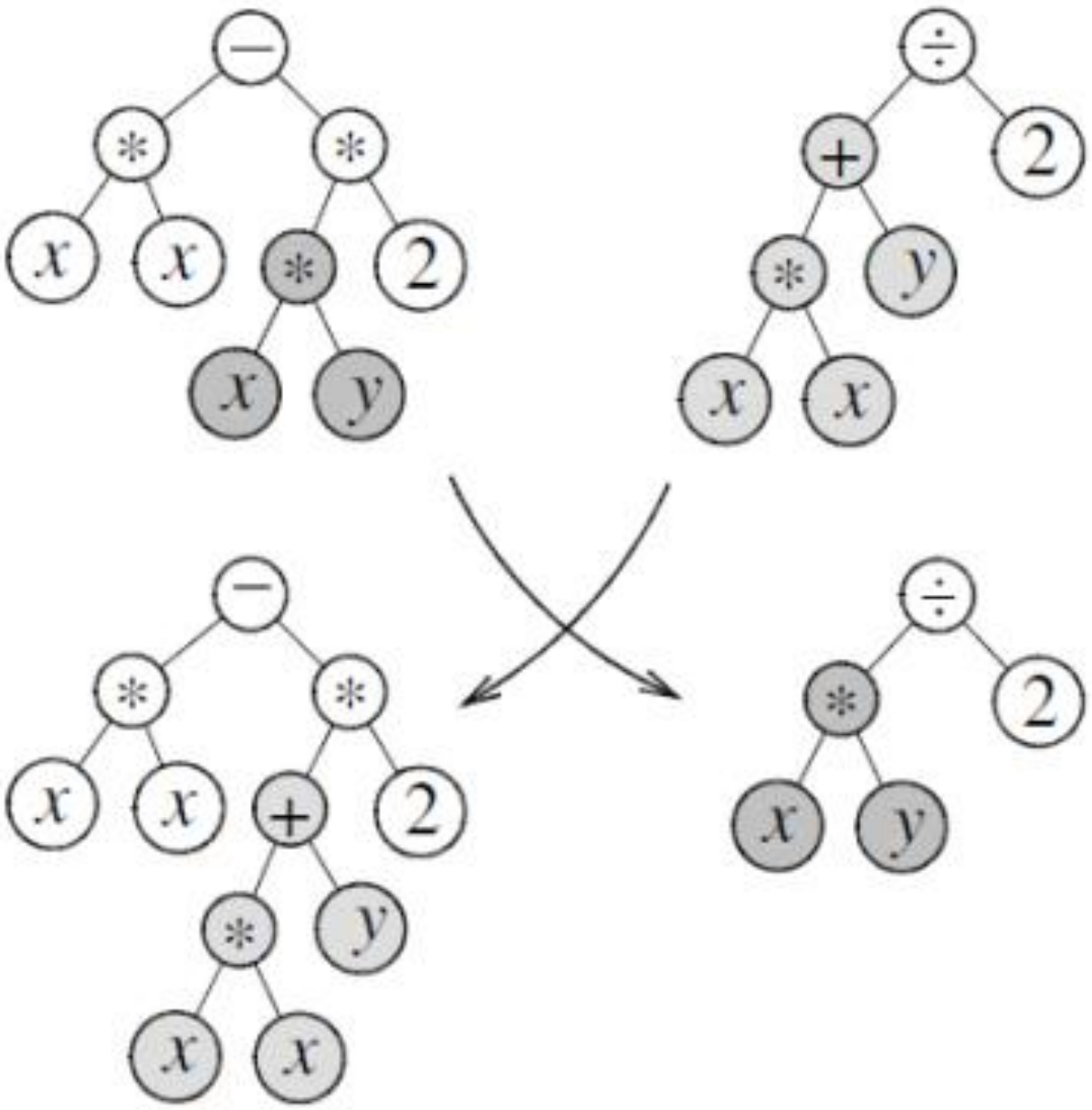
$f(x) = 4.8 * x_3 + \sqrt{x_2}$



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



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)))))))).
```

[Koza1992]

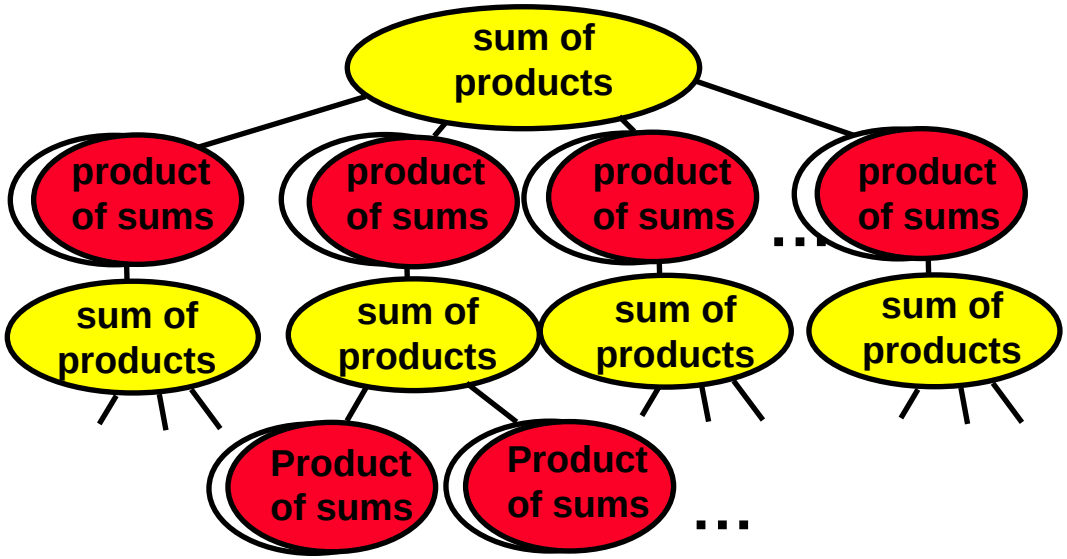
SR with Vanilla GP.. And Problems (2)

```
(+ (* (* -0.5403 (+ 0.5741 -0.8861)) (% (*
0.29690000000000016 0.08089999999999997) (+ (% (% (-
-0.5962000000000001 0.3902000000000001) (- (+ (% (* (+ (*
0.23550000000000004 0.15060000000000007) (* (*
-0.10289999999999999 -0.7332) 0.7723)) (*
 0.23550000000000004 0.15060000000000007)) (+ 0.6026 (+ (+
 (% (- 0.37250000000000005 -0.34909999999999997) (- -0.776
 -0.6013)) (- -0.5250999999999999 -0.009000000000000008))
 (% (- 0.29690000000000016 -0.34909999999999997) (- -0.776
 -0.6013))))) (* (+ -0.8861 (% -0.06019999999999992
 0.051100000000000145)) (% -0.06019999999999992
 0.051100000000000145))) (% -0.49659999999999993 0.4475)))
 (+ (% (% (* (+ -0.1943999999999999 0.4366000000000001) (*
 0.23550000000000004 0.15060000000000007)) (+ 0.6026 (* (*
 (+ (* -0.5403 -0.017199999999999993) (%
 -0.06019999999999992 0.051100000000000145)) (% (* (+
 -0.1943999999999999 0.4366000000000001) (*
 0.23550000000000004 0.15060000000000007)) (% (%
 0.42100000000000004 -0.4275) (- -0.48160000000000003
 0.5708)))) 0.7723))) (- -0.8395 -0.1986)) (% (-
 0.37250000000000005 -0.34909999999999997) (- -0.776
 -0.6013)))) (% (% (+ 0.6698000000000002
 0.8714000000000002) (% (- -0.829 -0.636) (-
 0.7635000000000001 -0.15899999999999992))) (- (- (*
 -0.5403 -0.017199999999999993) (- -0.8395 -0.1986)) (- (*
 (* -0.5403 -0.017199999999999993) (- 0.6004 -0.4343)) (-
 -0.951 (* (% 0.7803 0.9777) 0.31920000000000015)))))))
 (+ (* (* -0.5403 -0.017199999999999993)
 -0.19240000000000002) (+ (+ -0.13339999999999996 0.7944)
 0.6004))).
```

[Koza1992]

CAFFEINE Approach

CAFFEINE = Canonical form functions in evolution



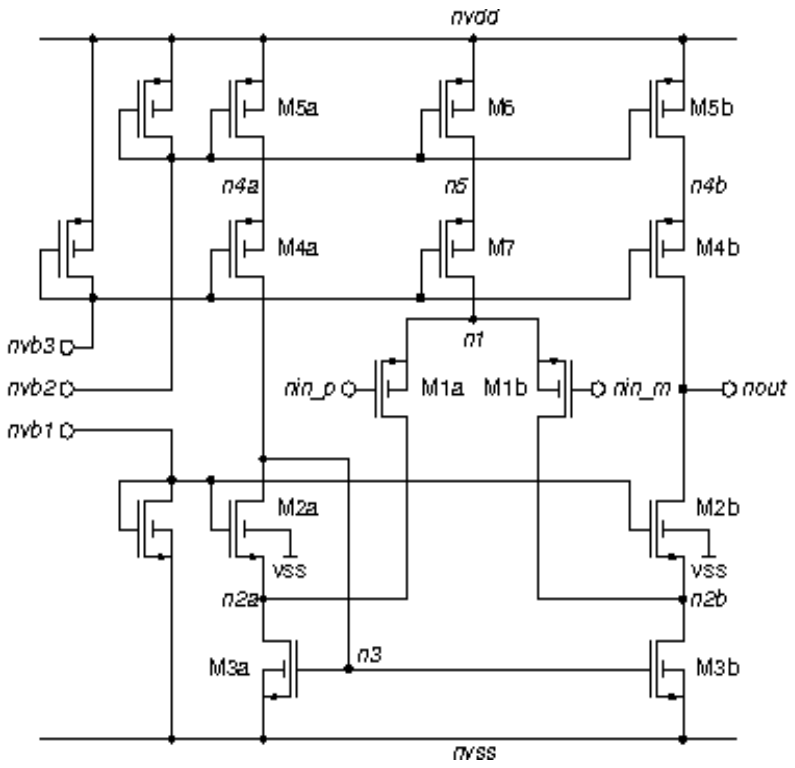
Grammar to describe the canonical forms:

```
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 | ...
```

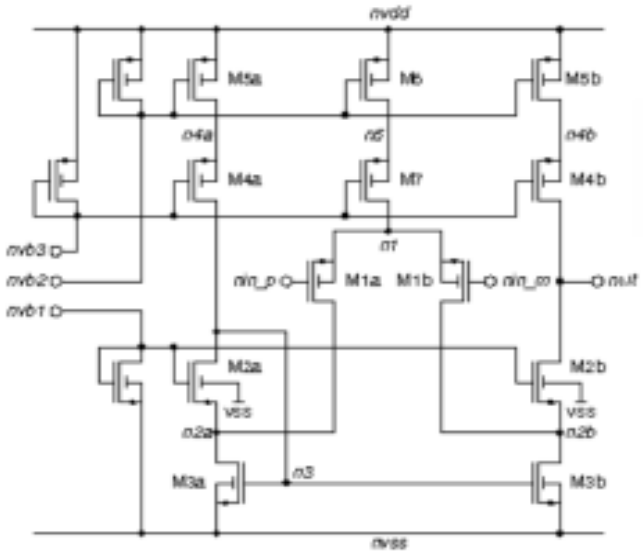
Search the space with *grammatically-constrained* GP [Whig1995]

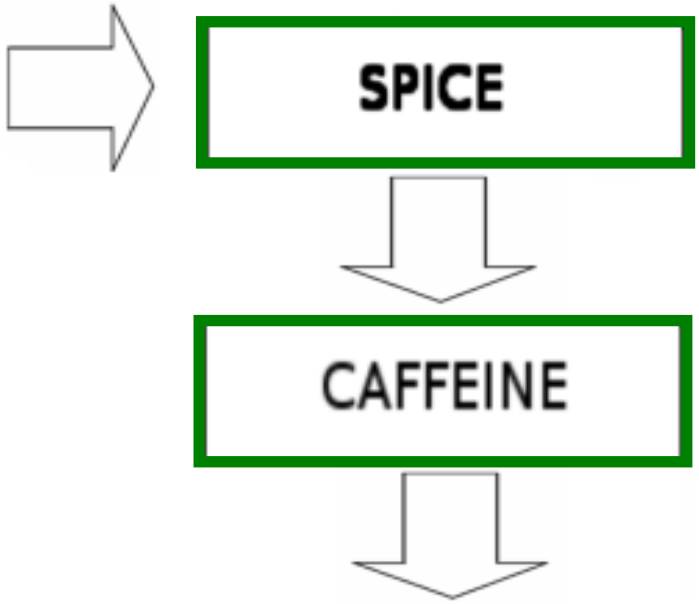
Benchmarks: Experimental Setup

- High Speed amplifier
- 13 design variables
  - Vds, Vgs, Ids (operating-point driven formulation)
- orthogonal hypercube sampling
- 243 training samples
- 243 testing samples



Example: GP for SR on Circuits





Perf.	Expression
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

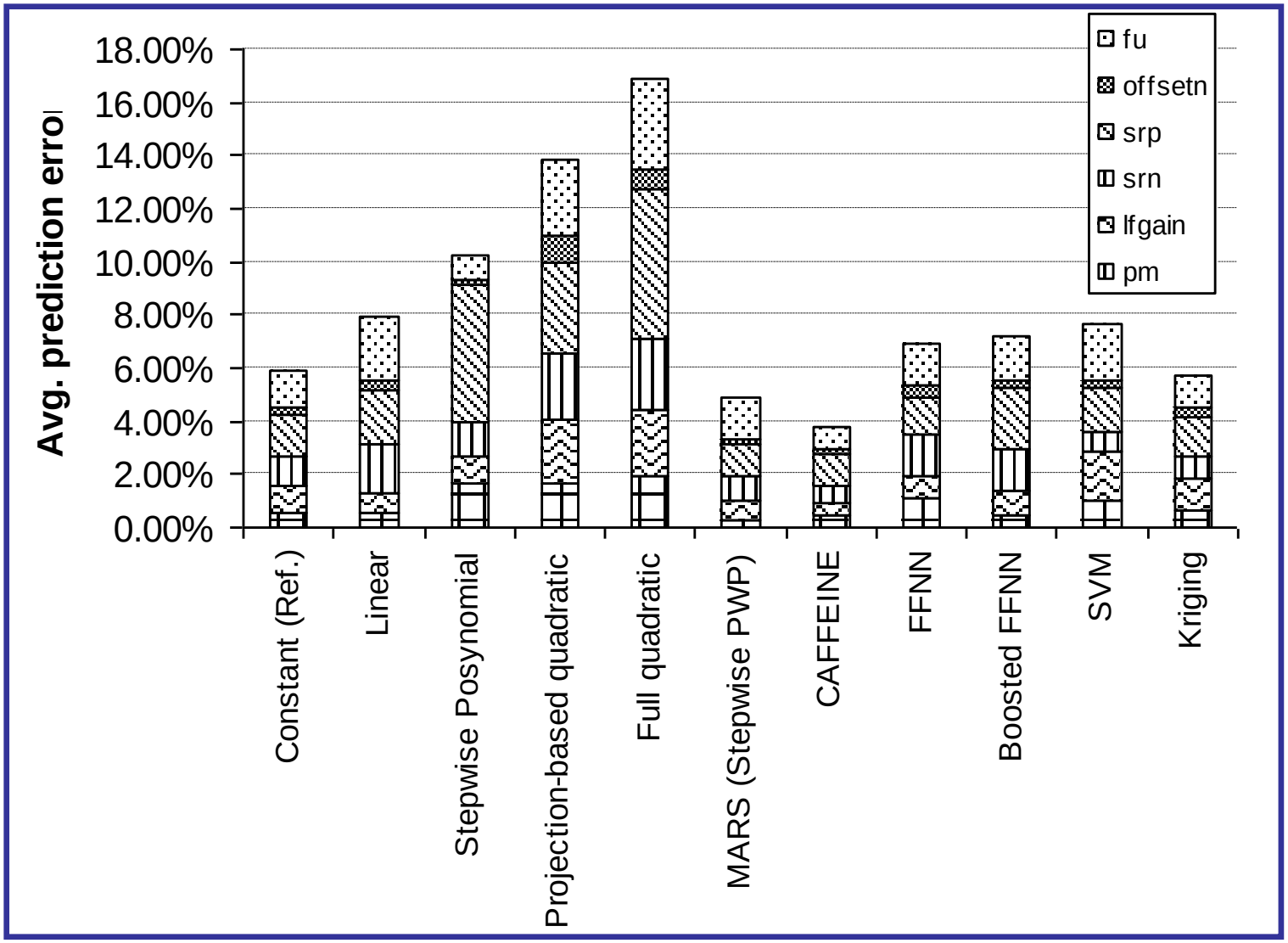
[McConaghy 2005]

CAFFEINE models with <10% error

Perf.	Target % error		Expression
	trn	tst	
A_{LF}	10	10	-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	10	10^(5.68 - 0.03 * vsg1 / vds2 - 55.43 * id1+ 5.63e-6 / id1)
PM	10	10	90.5 + 190.6 * id1 / vsg1 + 22.2 * id2 / vds2
v_{offset}	10	10	- 2.00e-3
SR_p	10	10	2.36e+7 + 1.95e+4 * id2 / id1 - 104.69 / id2 + 2.15e+9 * id2 + 4.63e+8 * id1
SR_n	10	10	- 5.72e+7 - 2.50e+11 * (id1*id2) / vgs2 + 5.53e+6 * vds2 / vgs2

CAFFEINE Prediction Performance

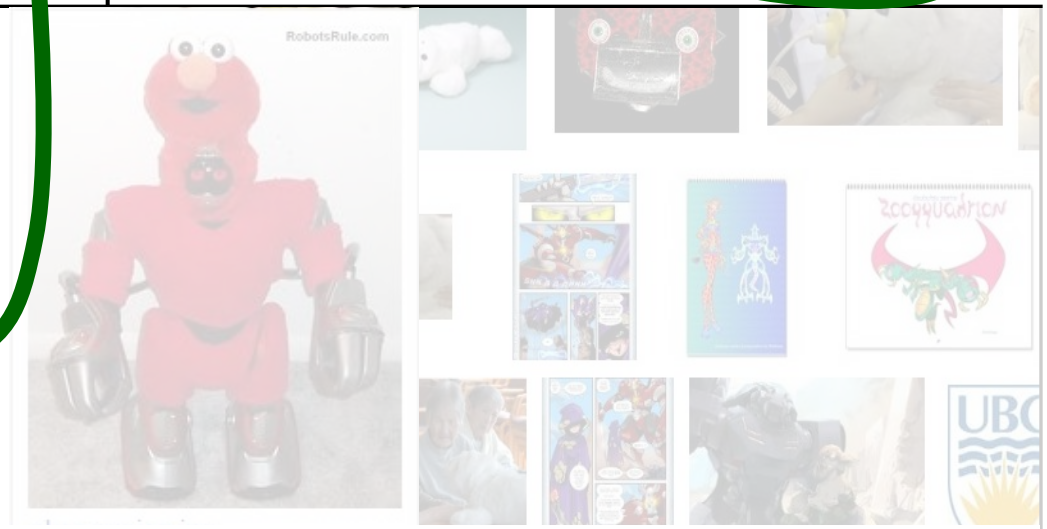
Predicts better than several state-of-the-art blackbox regression techniques on circuits benchmark suite (*and* gives whitebox models).



Conclusion

What does AI encompass?

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



Is Deep Learning cool or what?

How *does* Google find furry robots?