

# Genetic Programming—GP

- Genetic Programming (GP) is evolution applied to machine learning
- That is given some examples of data we want to learn a rule
- In GP we evolve a program or more usually a formula to learn the rule
- The bigger vision would be to evolve programs to perform tasks rather than hire programmers
- I think it is fair to say that GP is still at an experimental level and has some way to prove its usefulness against more established techniques

# Genetic Programming—GP

- Genetic Programming (GP) is evolution applied to machine learning
- That is given some examples of data we want to learn a rule
- In GP we evolve a program or more usually a formula to learn the rule
- The bigger vision would be to evolve programs to perform tasks rather than hire programmers
- I think it is fair to say that GP is still at an experimental level and has some way to prove its usefulness against more established techniques

# Genetic Programming—GP

- Genetic Programming (GP) is evolution applied to machine learning
- That is given some examples of data we want to learn a rule
- In GP we evolve a program or more usually a formula to learn the rule
- The bigger vision would be to evolve programs to perform tasks rather than hire programmers
- I think it is fair to say that GP is still at an experimental level and has some way to prove its usefulness against more established techniques

# Genetic Programming—GP

- Genetic Programming (GP) is evolution applied to machine learning
- That is given some examples of data we want to learn a rule
- In GP we evolve a program or more usually a formula to learn the rule
- The bigger vision would be to evolve programs to perform tasks rather than hire programmers
- I think it is fair to say that GP is still at an experimental level and has some way to prove its usefulness against more established techniques

# Genetic Programming—GP

- Genetic Programming (GP) is evolution applied to machine learning
- That is given some examples of data we want to learn a rule
- In GP we evolve a program or more usually a formula to learn the rule
- The bigger vision would be to evolve programs to perform tasks rather than hire programmers
- I think it is fair to say that GP is still at an experimental level and has some way to prove its usefulness against more established techniques

# Typical Application

- Suppose you have a chemical plant with many control parameters and some observables (things you measure)
- One of the classic control problems is *system identification*
- That is, finding an equation which describes the system
- GP has been used to obtain formulae describing how the observables depend on the control parameters.
- One advantage of GP over many other machine learning techniques is that the formula is often interpretable by humans

# Typical Application

- Suppose you have a chemical plant with many control parameters and some observables (things you measure)
- One of the classic control problems is *system identification*
- That is, finding an equation which describes the system
- GP has been used to obtain formulae describing how the observables depend on the control parameters.
- One advantage of GP over many other machine learning techniques is that the formula is often interpretable by humans

# Typical Application

- Suppose you have a chemical plant with many control parameters and some observables (things you measure)
- One of the classic control problems is *system identification*
- That is, finding an equation which describes the system
- GP has been used to obtain formulae describing how the observables depend on the control parameters.
- One advantage of GP over many other machine learning techniques is that the formula is often interpretable by humans



# Typical Application

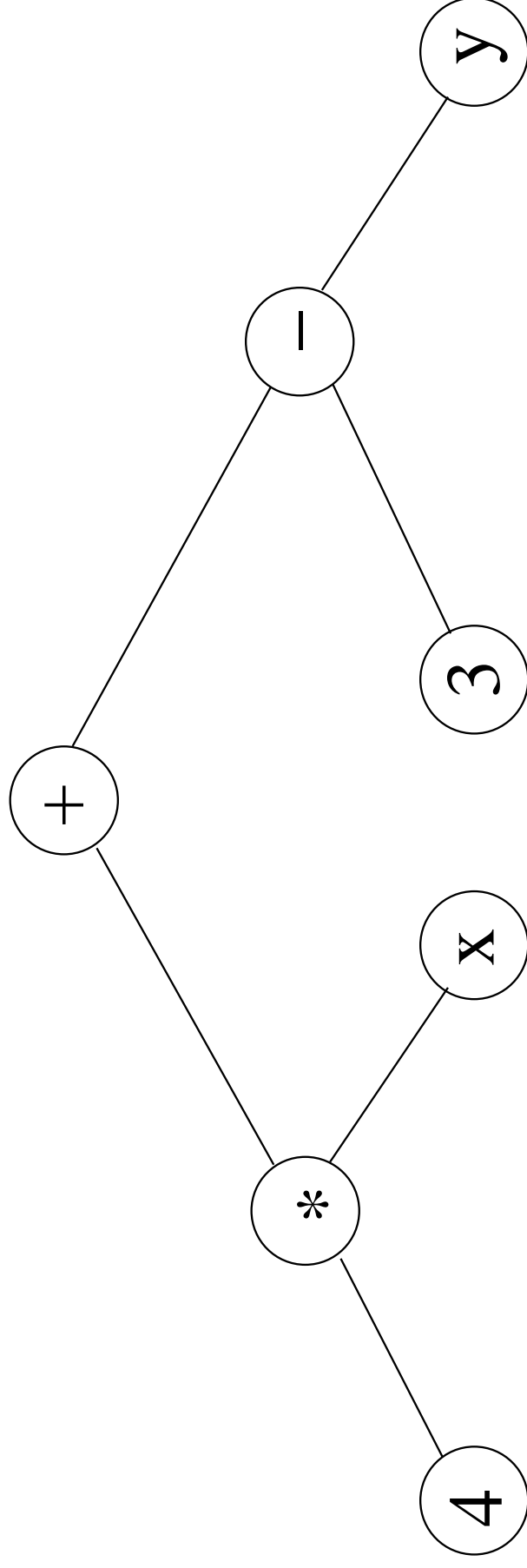
- Suppose you have a chemical plant with many control parameters and some observables (things you measure)
- One of the classic control problems is *system identification*
- That is, finding an equation which describes the system
- GP has been used to obtain formulae describing how the observables depend on the control parameters.
- One advantage of GP over many other machine learning techniques is that the formula is often interpretable by humans

# Typical Application

- Suppose you have a chemical plant with many control parameters and some observables (things you measure)
- One of the classic control problems is *system identification*
- That is, finding an equation which describes the system
- GP has been used to obtain formulae describing how the observables depend on the control parameters.
- One advantage of GP over many other machine learning techniques is that the formula is often interpretable by humans

# Representing Rules

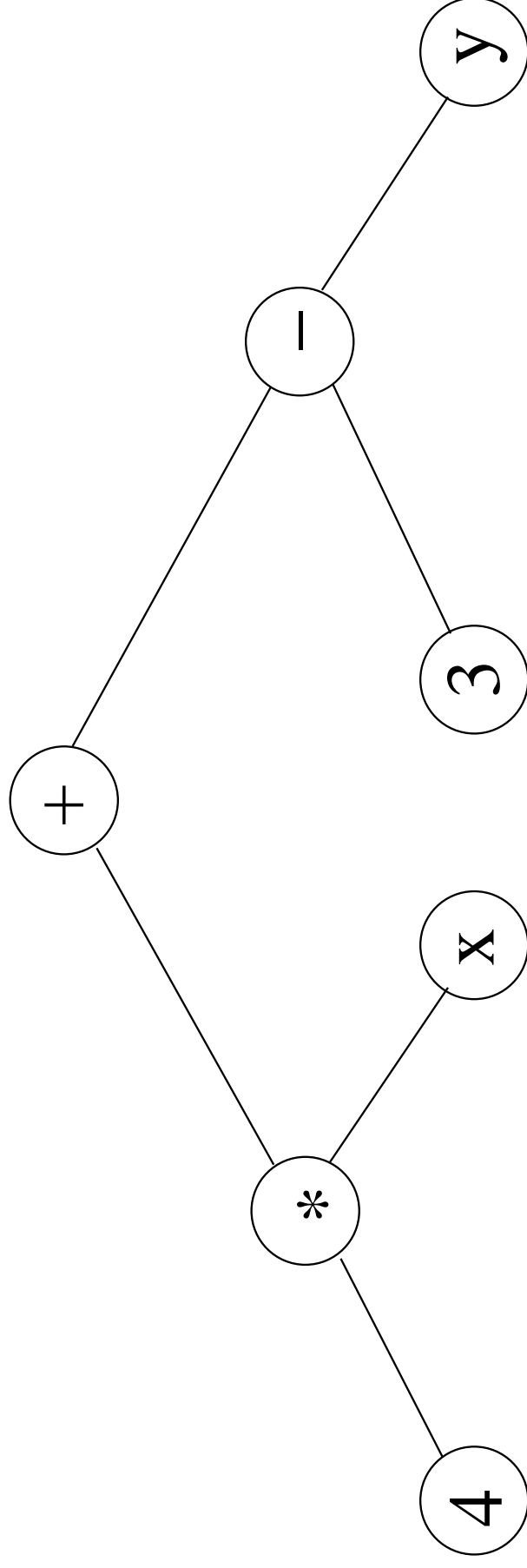
- The first challenge in GP is deciding how to represent a rule or formula
- Different approaches, most commonly use expression trees



$$(4 * x) + (3 - y)$$

# Representing Rules

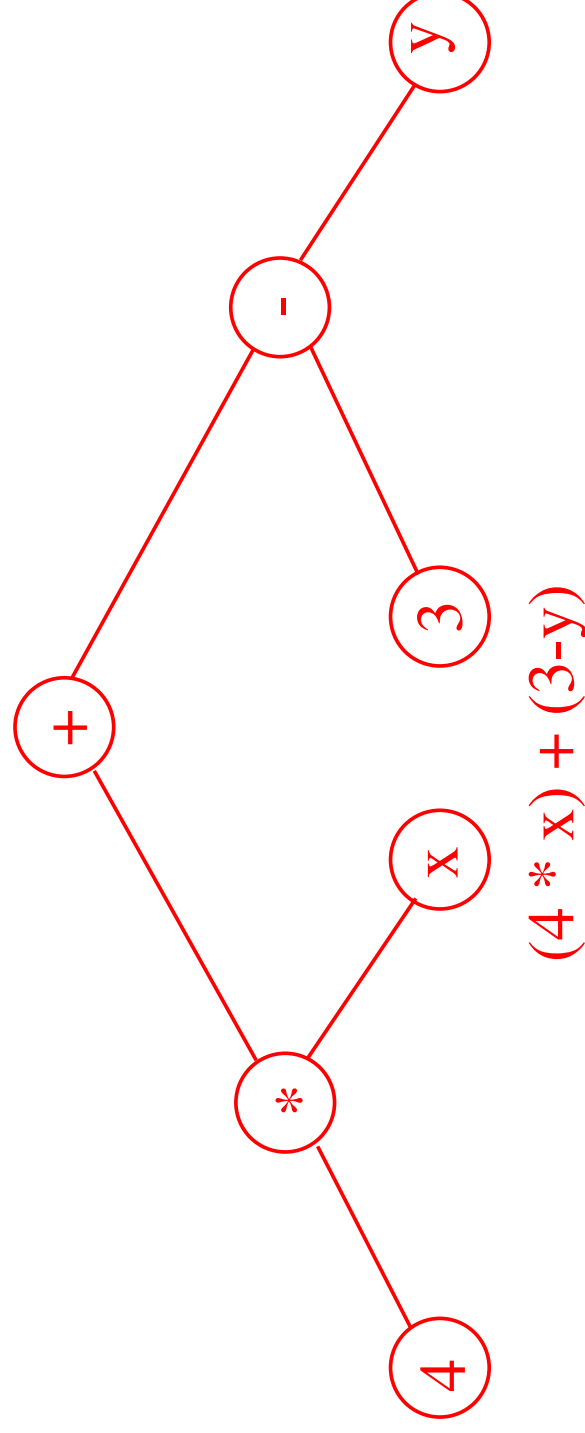
- The first challenge in GP is deciding how to represent a rule or formula
- Different approaches, most commonly use expression trees



$$(4 * x) + (3 - y)$$

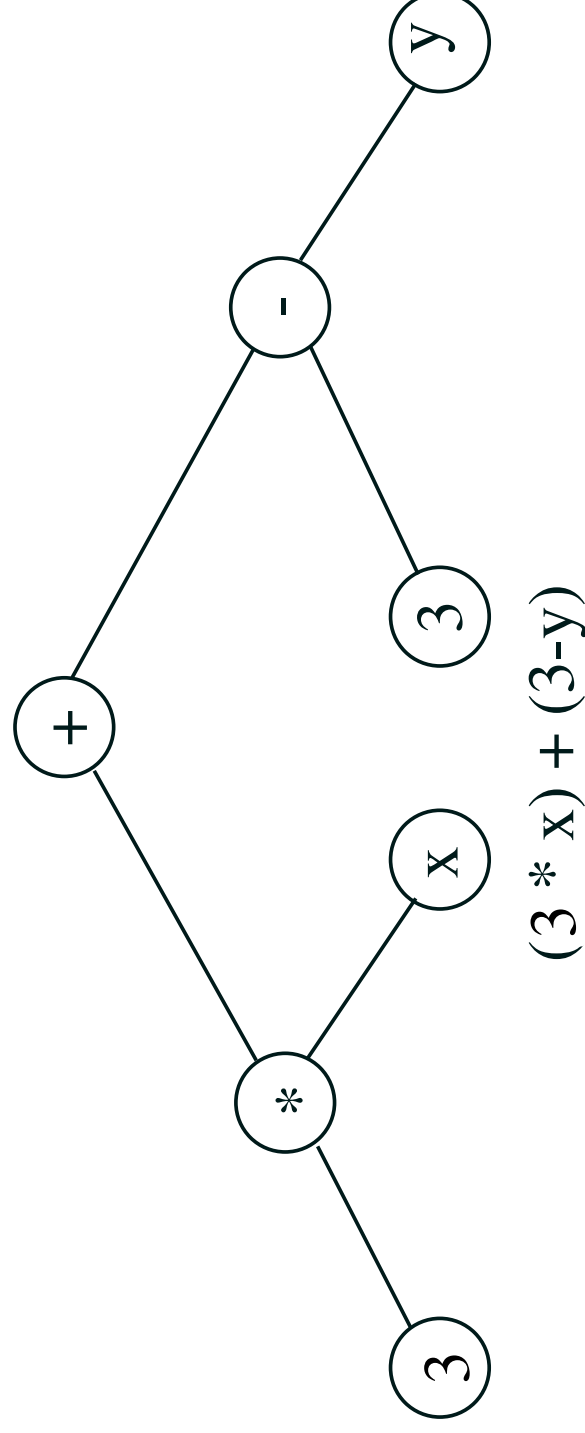
# Mutations

- Mutations usually correspond to either
  - ★ changing a leaf or
  - ★ changing a binary node



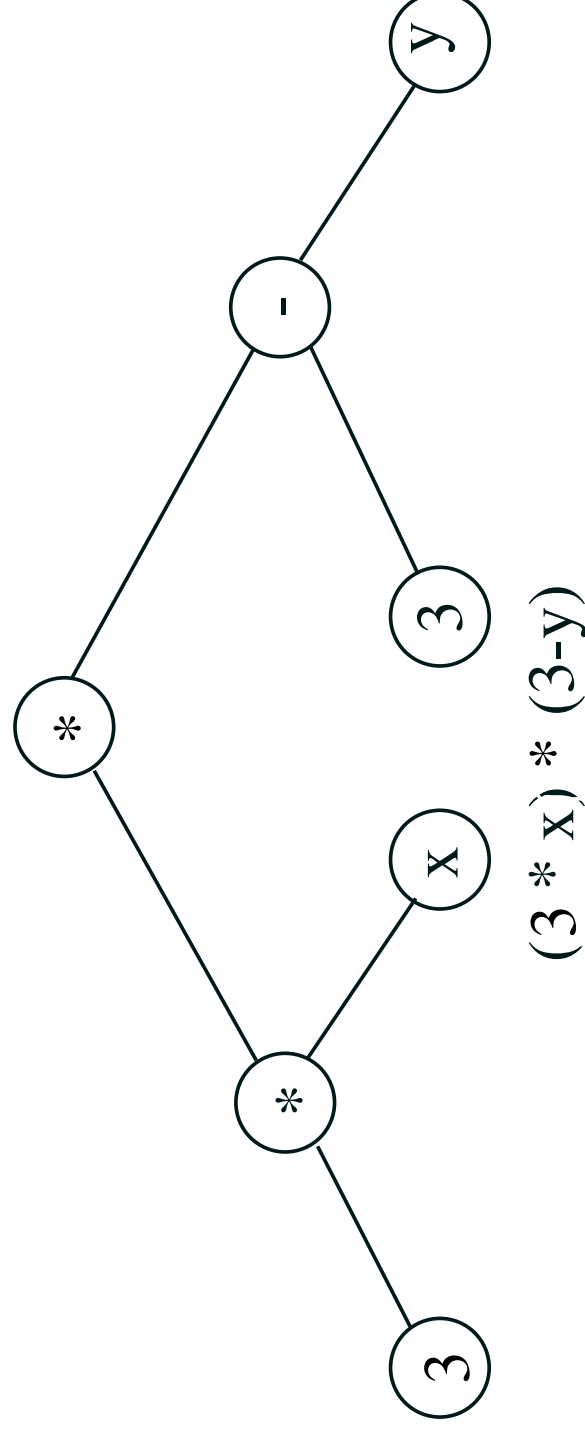
# Mutations

- Mutations usually correspond to either
  - ★ changing a leaf or
  - ★ changing a binary node



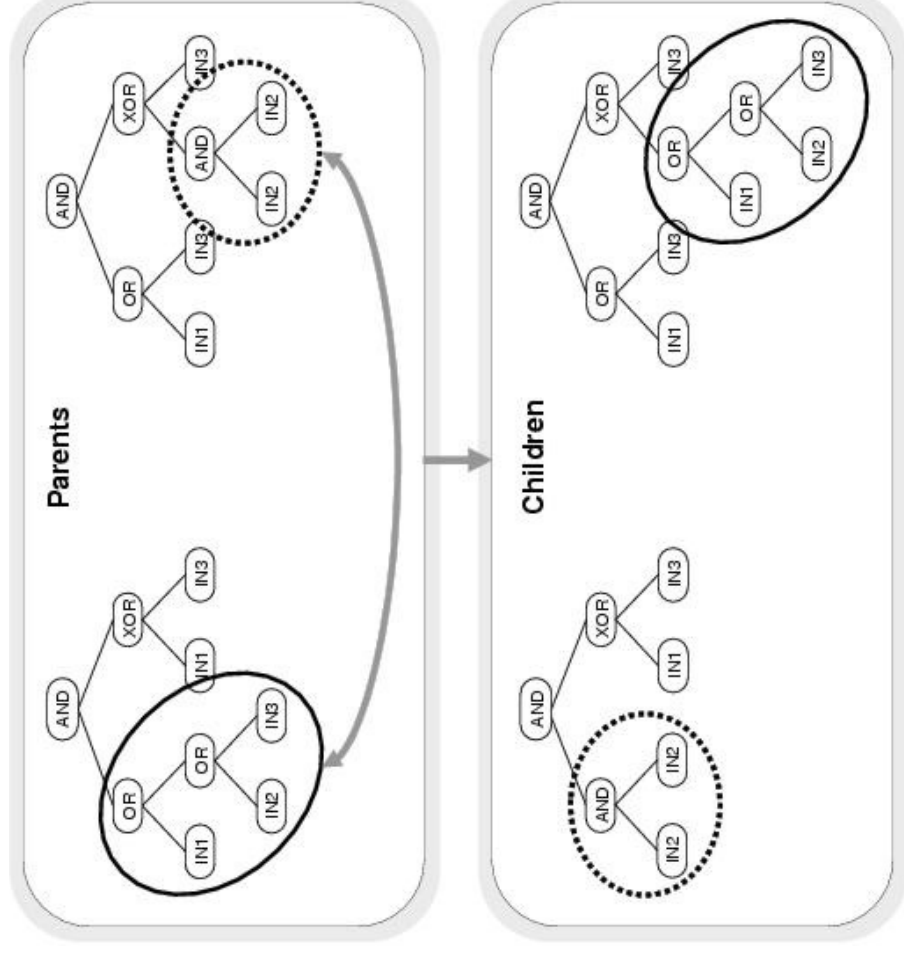
# Mutations

- Mutations usually correspond to either
  - ★ changing a leaf or
  - ★ changing a binary node



# Crossover

- The most common form of crossover is exchanging subtrees





# Disruption

- Mutation and crossover can be quite disruptive

$$(29 * x + 33) - (5 + x) \\ (29 * x + 33)^{(5+x)}$$

- However, in a binary expression tree most nodes are at the leaves and this will usually cause much smaller changes

# Disruption

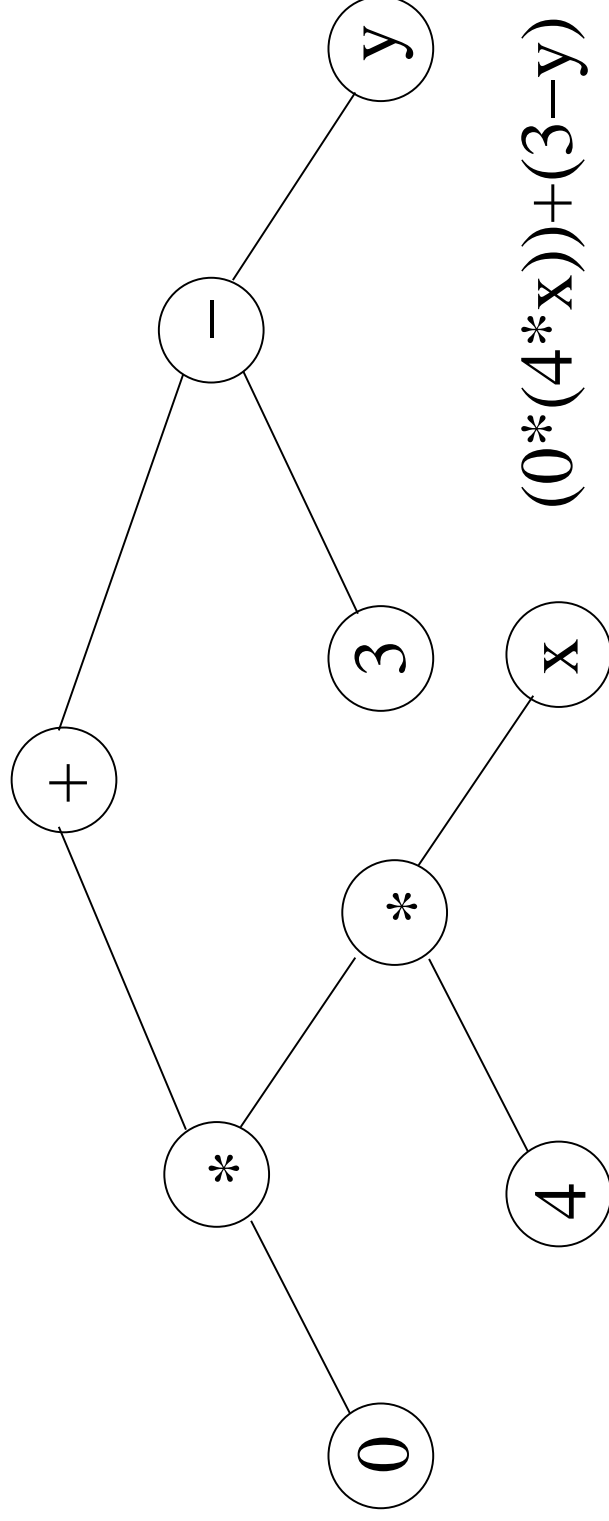
- Mutation and crossover can be quite disruptive

$$(29 * x + 33) - (5 + x)$$
$$(29 * x + 33)^{(5+x)}$$

- However, in a binary expression tree most nodes are at the leaves and this will usually cause much smaller changes

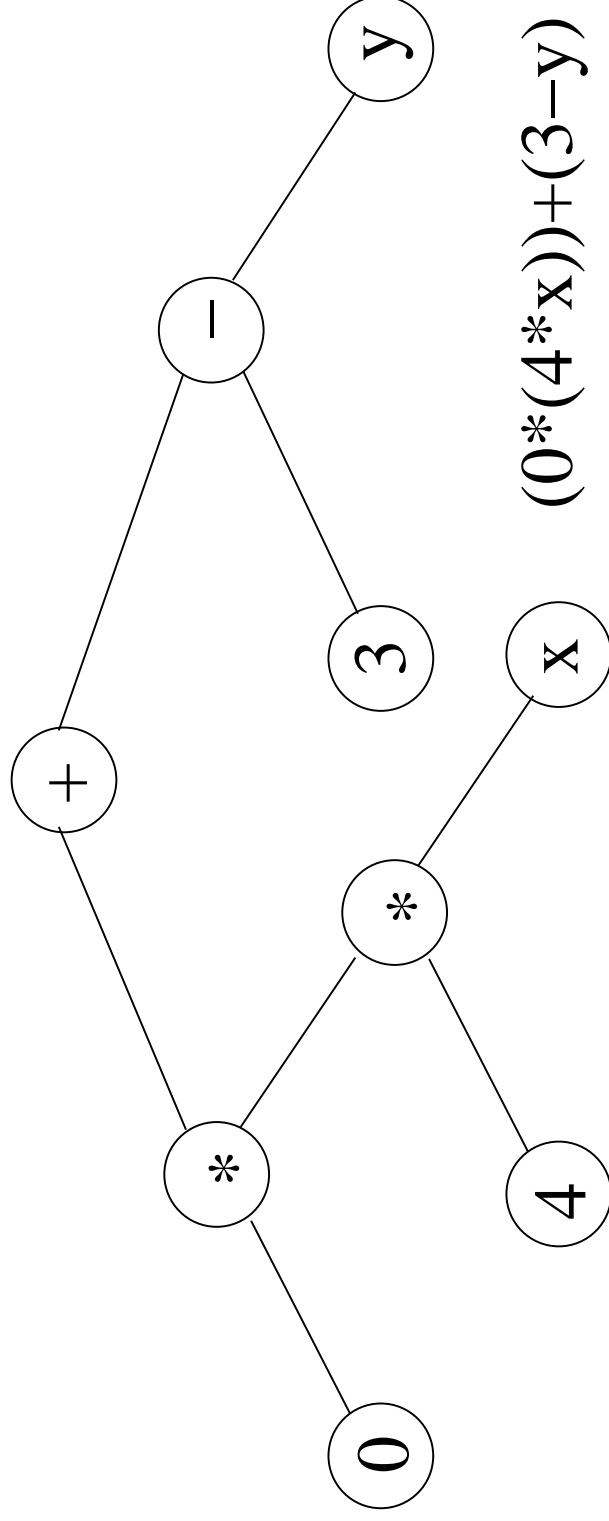
# Dysfunction Subtrees

- Often a subtree will not contribute to the formulae



# Dysfunction Subtrees

- Often a subtree will not contribute to the formulae



# Bloat

- Over time expression trees tend to grow
- This is known as **bloat**
- Much of the tree might be dysfunctional
- One possible advantage of bloat is that it can reduce the disruptiveness of mutation—it decreases the amount of mutation suffered by the functional part of the expression tree

# Bloat

- Over time expression trees tend to grow
- This is known as **bloat**
- Much of the tree might be dysfunctional
- One possible advantage of bloat is that it can reduce the disruptiveness of mutation—it decreases the amount of mutation suffered by the functional part of the expression tree

# Bloat

- Over time expression trees tend to grow
- This is known as **bloat**
- Much of the tree might be dysfunctional
- One possible advantage of bloat is that it can reduce the disruptiveness of mutation—it decreases the amount of mutation suffered by the functional part of the expression tree

# Bloat

- Over time expression trees tend to grow
- This is known as **bloat**
- Much of the tree might be dysfunctional
- One possible advantage of bloat is that it can reduce the disruptiveness of mutation—it decreases the amount of mutation suffered by the functional part of the expression tree



# Parsimony Pressure

- Bloat has a two downsides though
  - ★ It reduces the interpretability of the formulae (a major attraction of using GP)
  - ★ It might produce over-complicated formulae (thus over-fitting the data and increasing the generalisation error)
- A common strategy to reduce the amount of bloat is to punish (through a fitness term) large expression trees

# Parsimony Pressure

- Bloat has a two downsides though
  - ★ It reduces the interpretability of the formulae (a major attraction of using GP)
  - ★ It might produce over-complicated formulae (thus over-fitting the data and increasing the generalisation error)
- A common strategy to reduce the amount of bloat is to punish (through a fitness term) large expression trees

# Parsimony Pressure

- Bloat has a two downsides though
  - ★ It reduces the interpretability of the formulae (a major attraction of using GP)
  - ★ It might produce over-complicated formulae (thus over-fitting the data and increasing the generalisation error)
- A common strategy to reduce the amount of bloat is to punish (through a fitness term) large expression trees

# Parsimony Pressure

- Bloat has a two downsides though
  - ★ It reduces the interpretability of the formulae (a major attraction of using GP)
  - ★ It might produce over-complicated formulae (thus over-fitting the data and increasing the generalisation error)
- A common strategy to reduce the amount of bloat is to punish (through a fitness term) large expression trees

# Summary of GP

- GP is a big research field
- Advocates claim a few successes in finding phenomenological formulae for describing complex systems
- It is usually very compute intensive—evolving very large populations for a large number of generations
- There are a lot of challenges in really understanding how GP works and improving it
  - ★ Inheriting subroutines
  - ★ Intelligent guidance of search
  - ★ Controlling over-fitting