

Emergence in Complex Systems

Genetic Programming





Purpose of Genetic Programming

Genetic Programming is a way to artificially generate programs by a process of simulated evolution.

A completely random program generation is impossible because of the size of the strict syntax of the languages.

Genetic Programming adapts the tools of Genetic Algorithms to the generation of programs with a strict syntax.





Global idea of Genetic Programming

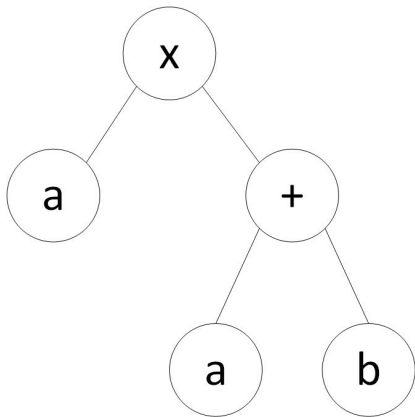
- A program can be represented as a tree
- Each subtree is a program and is considered as a gene
- A *terminal* is leaf: it consists in constants or arguments (or 0-arity functions).
- A *function* is an internal node. Its children are its arguments.
- A population of programs will evolve genetically.

From now on, we will only consider that the programs model real functions $f : \mathbb{R}^d \mapsto \mathbb{R}$





Example



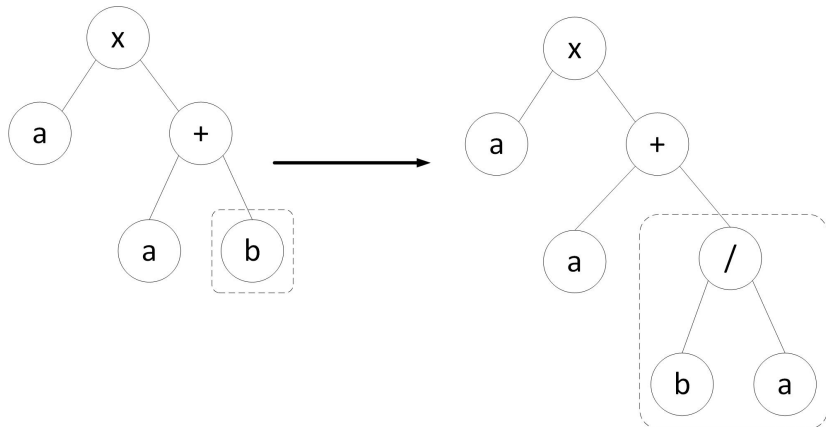
This program corresponds to the function $f(a, b) = a(a + b)$





Genetic operators

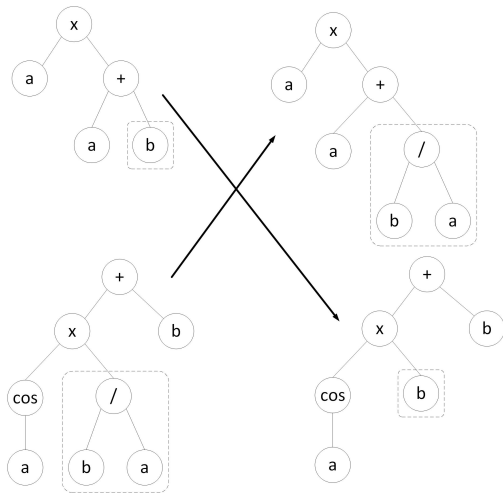
Mutation





Genetic operators

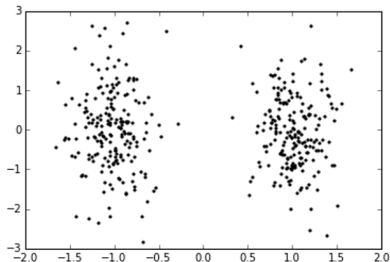
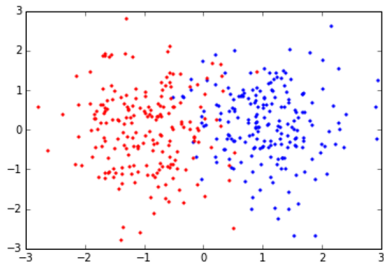
Crossover





Classification in Machine Learning

Classification consists in learning a labeling rule for incoming inputs based on already labeled inputs (called *training data*).





Learning a classification function

Training data: $\{(X_i, Y_i)\}$ with $X_i \in \mathcal{X}$ and $Y_i \in \{-1, 1\}$

The purpose of classification is to learn a function $h : \mathcal{X} \mapsto \{-1, 1\}$.
In practice, we often learn a function $f : \mathcal{X} \mapsto \mathbb{R}$ and choose $h = \text{sign}(f)$

The criterion used to measure the quality of the learning is the number of errors done on the training data:




$$E(f) = \frac{1}{n} \sum_{i=1}^n \mathbb{I}(Y_i \neq h(X_i))$$

Genetic Programming can be used to learn f





References

-  F.-M De Rainville, F.-A. Fortin, M.-A. Gardner, M. Parizeau, and C. Gagné, *Deap: A python framework for evolutionary algorithms*, Proceedings of the 14th Annual Conference Companion on Genetic and Evolutionary Computation (New York, NY, USA), GECCO '12, ACM, 2012, pp. 85–92.
-  Melanie Mitchell, *An introduction to genetic algorithms*, MIT Press, Cambridge, MA, USA, 1998.
-  Riccardo Poli, William B. Langdon, and Nicholas Freitag McPhee, *A field guide to genetic programming*, Lulu Enterprises, UK Ltd, 2008.





Licence de droits d'usage



Contexte public } sans modifications

Par le téléchargement ou la consultation de ce document, l'utilisateur accepte la licence d'utilisation qui y est attachée, telle que détaillée dans les dispositions suivantes, et s'engage à la respecter intégralement.

La licence confère à l'utilisateur un droit d'usage sur le document consulté ou téléchargé, totalement ou en partie, dans les conditions définies ci-après et à l'exclusion expresse de toute utilisation commerciale.

Le droit d'usage défini par la licence autorise un usage à destination de tout public qui comprend :

- Le droit de reproduire tout ou partie du document sur support informatique ou papier,
- Le droit de diffuser tout ou partie du document au public sur support papier ou informatique, y compris par la mise à la disposition du public sur un réseau numérique.

Aucune modification du document dans son contenu, sa forme ou sa présentation n'est autorisée.

Les mentions relatives à la source du document et/ou à son auteur doivent être conservées dans leur intégralité.

Le droit d'usage défini par la licence est personnel, non exclusif et non transmissible.

Tout autre usage que ceux prévus par la licence est soumis à autorisation préalable et expresse de l'auteur : sitopedago@telecom-paristech.fr

