Root findings : equations f(x) = 0

Algorithm used to find roots of an equation use iterations, and a numerical criterion to accept a solution when a sufficiently accurate value is reached. The rate of convergence depends on the used method and the function f(x). Some methods (Newton-Raphson) need the derivative of the function f(x).

- Polynomial equations : Bairstow's method is an efficient algorithm for finding the roots of a real polynomial of arbitrary degree
  - Polynomials in NumPy
  - polynomial module, including polyroots© to compute the roots of a polynomial.
- Bisection method (dichotomy) : very simple and robust method, but relatively slow. It assumes continuity of the function, and obtain one roots. The algorithm is based on a loop invariant property : an interval [a, b] is said to bracket a root if f(a) and f(b) have opposite signs.
- Secant method (retains the last two computed points)
- Regula falsi (retains the points which preserve bracketing)
- Newton-Raphson method
- Chapter 9 in the book “Numerical Recipes” : Root finding an nonlinear sets of equations
  - 9.0 Introduction
  - 9.1 Bracketing and Bisection
  - 9.2 Secant Method, False Position Method, and Ridders' Method
  - 9.4 Newton-Raphson Method Using Derivative
  - 9.5 Roots of Polynomials
- Python NumPy library : SciPy Reference
  - scipy.optimize package (root)

Applications

- ...

Références

- Numerical recipes, The Art of Scientific Computing 3rd Edition, William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, 2007, isbn: 9780521880688
  - http://numerical.recipes/
    - in C : http://apps.nrbook.com/c/index.html
  - http://www2.units.it/ipl/students_area/imm2/files/Numerical_Recipes.pdf, p 347...
  - http://apps.nrbook.com/empanel/index.html#
    - Chapter 9 : Root finding an nonlinear sets of equations