Interior-point methods

Release 1.0.0

Victor Barbarich, Adelina Tsoi

Apr 26, 2022

CONTENTS:

CHAPTER ONE

INNER_POINT

CHAPTER TWO

ALGORITHMS MODULE

`gradient`(*f*, *x*, *delta_x=1e-08*)

Returns the gradient of the function at a specific point x

A two-point finite difference formula that approximates the derivative

$$\frac{\partial f}{\partial x} \approx \frac{f(x+h) - f(x-h)}{2h} \tag{2.1}$$

Gradient

$$\nabla f = \left[\frac{\partial f}{\partial x_1} \ \frac{\partial f}{\partial x_2} \ \dots \ \frac{\partial f}{\partial x_n}\right]^\top \tag{2.2}$$

Parameters

- **f** (*Callable[[numpy.ndarray], numbers.Real]*) – function which depends on n variables from x
- **x** (*numpy.ndarray*) – n - dimensional array
- **delta_x** (*numbers.Real*) – precision of two-point formula above (delta_x = h)

Returns

Return type numpy.ndarray

`jacobian`(*f_vector*, *x*, *delta_x=1e-08*)

Returns the Jacobian matrix of a sequence of m functions from f_vector by n variables from x.

$$\nabla f = \left[\frac{\partial f}{\partial x_1} \ \frac{\partial f}{\partial x_2} \ \dots \ \frac{\partial f}{\partial x_n}\right]^\top \tag{2.3}$$

Parameters

- **f_vector** (*Sequence[Callable[[numpy.ndarray], numbers.Real]]*) – a flat sequence, list or tuple or other containing m functions
- **x** (*numpy.ndarray*) – an n-dimensional array. The specific point at which we will calculate the Jacobian
- **delta_x** (*numbers.Real*) – precision of gradient

Returns the Jacobian matrix according to the above formula. Matrix n x m

Return type numpy.ndarray

PYTHON MODULE INDEX

INDEX