optimization

Generated by Doxygen 1.9.1

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 2

File Documentation

2.1 optimization.hpp File Reference

```
#include <stdexcept>
#include <functional>
#include <optional>
#include "armadillo"
```

Typedefs

- using **f64** = double

Functions

- template<typename T >
  Mat< T > **jacobian** (const function< Col< T >(Col< T >)> &func, const Col< T > &x)

  Numerically approximate the jacobian matrix for a vector function Compute the numerical approximation for the Jacobian matrix using the centered-difference approximation. This requires the size of the function 'f' to be the number of input parameters so that the corresponding Jacobian matrix is square.
- template<typename T >
  optional< Col< T > > **root** (const function< Col< T >(Col< T >)> &func, const function< Mat< T >(Col< T >)> &jac, const Col< T > &x0)

  Determine the root of a general vector function Determine the root of a general vector function using Newton-Raphson iteration. Newton-Raphson iteration requires evaluating a Jacobian and then solving the resulting linear system. The iterative method looks like: x^{i+1} = x^{i} - J(x^{i})^{-1}f(x^{i}) This method has quadratic convergence, though may be unstable and therefore requires an initial guess that is sufficiently close to the root.

Variables

- const unsigned int **MAX_ITERATIONS** = 50

2.1.1 Typedef Documentation

2.1.1.1 f64

```
using f64 = double
```

Maximum number of iterations before failing

2.1.2 Function Documentation

2.1.2.1 jacobian()

```
template<typename T >
Mat<T> jacobian (
            const function< Col< T >(Col< T >)> & func,
            const Col< T > & x )
```

Numerically approximate the jacobian matrix for a vector function Compute the numerical approximation for the Jacobian matrix using the centered-difference approximation. This requires the size of the function 'f' to be the number of input parameters so that the corresponding Jacobian matrix is square.

Template Parameters

T	A floating point data type

Parameters

func	The continuous function whose Jacobian to approximate
x	The value at which to approximate the Jacobian of 'f'

Returns

Mat<T>

2.1.2.2 root()

```
template<typename T >
optional<Col<T> > root (
            const function< Col< T >(Col< T >)> & func,
            const function< Mat< T >(Col< T >)> & jac,
            const Col< T > & x0 )
```

Determine the root of a general vector function Determine the root of a general vector function using Newton-↩ Raphson iteration. Newton-Raphson iteration requires evaluating a Jacobian and then solving the resulting linear system. The iterative method looks like: x^{i+1} = x^{i} - J(x^{i})^{-1}f(x^{i}) This method has quadratic convergence, though may be unstable and therefore requires an initial guess that is sufficiently close to the root.

Template Parameters

T	

Parameters

func	The function to find the root of
jac	The function that generates the jacobian matrix
x0	The initial guess for the root

Returns

optional<Col<T>>

Index