Dr JanV's c++ header-library

Generated by Doxygen 1.9.3

Chapter 1

Namespace Documentation

1.1 drjanv Namespace Reference

Functions

- double Legendre (int N, double x)
- double dLegendredx (int N, double x)
- double d2Legendredx2 (int N, double x)
- std::vector< double > LegendreRoots (unsigned int N, unsigned int max_iters=1000, double tol=1.0e-12)
- std::pair< std::vector< double >, std::vector< double > > GaussLegendreQuadrature (unsigned int N, bool verbose=false, unsigned int max_iters=1000, double tol=1.0e-12)
- template<typename T , typename D = int>
  std::vector< T > Range (T start, T end, D delta=1)

1.1.1 Function Documentation

1.1.1.1 Legendre()

```
double drjanv::Legendre (
              int N,
              double x )
```

Provides the function evaluation of the Legendre polynomial P_N at value x.

Parameters

N	int Order of the Legendre polynomial.
x	double The evaluation point.

1.1.1.2 dLegendredx()

```
double drjanv::dLegendredx (
             int N,
             double x )
```

Provides the function evaluation of the derivative of the Legendre polynomial $\frac{dP_N}{dx}$ at value x.

Parameters

N	int Order of the Legendre polynomial.
x	double The evaluation point.

1.1.1.3 d2Legendredx2()

```
double drjanv::d2Legendredx2 (
             int N,
             double x )
```

Provides the function evaluation of the second derivative of the Legendre polynomial $\frac{d^2P_N}{dx^2}$ at value x.

Parameters

N	int Order of the Legendre polynomial.
x	double The evaluation point.

1.1.1.4 LegendreRoots()

```
std::vector< double > drjanv::LegendreRoots (
             unsigned int N,
             unsigned int max_iters = 1000,
             double tol = 1.0e-12 )
```

Finds the roots of the Legendre polynomial.

The algorithm is that depicted in:

[1] Barrera-Figueroa, et al., "Multiple root finder algorithm for Legendre and Chebyshev polynomials via Newton's method", Annales Mathematicae et Informaticae, 33 (2006) pp. 3-13.

Parameters

N	Is the order of the polynomial.
roots	Is a reference to the roots.
max_iters	Maximum newton iterations to perform for each root. Default: 1000.
tol	Tolerance at which the newton iteration will be terminated. Default: 1.0e-12.

Returns

A `std::vector<double>` containg a sorted list of roots in the interval [-1,1].

Author

Jan

1.1.1.5 GaussLegendreQuadrature()

```
std::pair< std::vector< double >, std::vector< double > > drjanv::GaussLegendreQuadrature (
            unsigned int N,
            bool verbose = false,
            unsigned int max_iters = 1000,
            double tol = 1.0e-12 )
```

Populates the abscissae and weights for a Gauss-Legendre quadrature given the number of desired quadrature points.

Parameters

N	Is the number of quadrature points.
roots	Is a reference to the roots.
max_iters	Maximum newton iterations to perform for each root. Default: 1000.
tol	Tolerance at which the newton iteration will be terminated. Default: 1.0e-12.

Returns

A pair with each part of type `std::vector<double>` and equal in size. The first part is a vector of quadrature points and the second part is a vector of weights.

Author

Jan

1.1.1.6 Range()

```
template<typename T , typename D = int>
std::vector< T > drjanv::Range (
            T start,
            T end,
            D delta = 1 )
```

Returns a range of number according to the logic of the parameters.

Parameters

start	First number in the sequence.
end	Termination criteria. If the delta is positive then the sequence will terminate if i>=end, otherwise if the delta is negative the sequence will terminate if i<=end
delta	Cannot be 0. Default 1. Can be negative.

Returns

A `std::vector` of template type `T` containing the range according to the logic.

Example:

```
auto iorder = drjanv::Range<int>(0, 10); // 0,1,...,9
auto iorder = drjanv::Range<int>(9,-1,-1); //9,8,...,0
```

Index