Technische Universität München
Lehrstuhl für Informatik V
Dr. Tobias Neckel
M. Sc. Ionuț Farcaș

SS 2017

May 24, 2017

Algorithms for Uncertainty Quantification

Tutorial 5: Repetition of interpolation and quadrature

In this worksheet, we focus on aspects regarding interpolation and quadrature.

Lagrange interpolation

In the lecture we saw that given a function $f : [-1, 1] \to \mathbb{R}$ and a grid of $N + 1$ interpolation nodes $G = (x_i)_{i=0}^N$, the Lagrange interpolant on this grid reads

$$I_N^G f(x) = \sum_{i=0}^{N} f(x_i) L_i^G(x), \tag{1}$$

where $L_i^G(x)$ is the i^{th} *Lagrange (cardinal) polynomial*

$$L_i^G(x) = \prod_{j=0, j\neq i}^{N} \frac{x - x_j}{x_i - x_j}, \quad i = 0, \ldots, N.$$

An alternative formulation, called the (first) barycentric form of interpolation, reads

$$I_N^G f(x) = L^G(x) \sum_{i=0}^{N} f(x_i) \frac{w_i}{x - x_i}, \tag{2}$$

where

$$L^G(x) = \prod_{i=0}^{N} (x - x_i), w_i = \frac{1}{\prod_{k\neq i}(x_i - x_k)}.$$

Assignment 1

Let $f : [-1, 1] \to \mathbb{R}, f(x) = \sin(10x)$. This is usually Interpolate f on a uniform grid in $[-1, 1]$ consisting in $N = [4, 8, 12, 16]$ points using Eq. (1) and (2). How many operations do the two techniques (roughly) require with respect to the grid size? Plot both the original function and the two interpolants. What do you observe? *Hint: in* ***python****, you can use* ***numpy.linspace(start, stop, num, endpoint, retstep, dtype)*** *to generate a uniform grid.*

Gaussian quadrature

In the lecture, we saw how Gaussian quadrature is defined. Given $g : \mathbb{D} \to \mathbb{R}, \mathbb{D} \subset \mathbb{R}$ and a weight function $w : \mathbb{D} \to \mathbb{R}$, an $N + 1$ Gaussian quadrature scheme estimates the integral $\int_{\mathbb{D}} f(x)w(x)dx$ as a weighted sum of the form

$$\int_{\mathbb{D}} f(x)w(x)dx = \sum_{i=0}^{N} w_i f(x_i) + \texttt{error},$$

where $\{(x_i, w_i)\}_{i=0}^{N}$ are pairs of quadrature nodes and weights. The nodes x_i are the zeros of polynomials p_{N+1} of degree $N + 1$ that are orthogonal with respect to $w(x)$, i.e.

$$\int_{\mathbb{D}} p_i(x)p_j(x)w(x)dx = k\delta_{ij},$$

where $k \in \mathbb{R}$ and δ_{ij} if Kronecker's delta function, i.e. $\delta_{ij} = 1$ when $i = j$, 0 otherwise. Furthermore, the weights w_i are computed as

$$w_j = \int_{\mathbb{D}} L_j^G(x)w(x)dx, \tag{3}$$

where $L_j^G(x)$ is the j^{th} order Lagrange polynomial constructed on grid consisting of the nodes $\{x_i\}_{j=0}^{N}$. Note that in our context, $w(x)$ is the input probability distribution.
In `chaospy`, it is very easy to generate Gaussian nodes and wights via the function `generate_quadrature`. Provided that `distr` is the underlying input probability distribution, the syntax for generating $N + 1$ Gaussian nodes and weights is

```
import chaospy as cp

# distr = is the underlying probability distribution
```

```
# the first arguments is the order of the quadrature scheme
# the nodes are numbered from 0...order
nodes, weights = cp.generate_quadrature(N, distr, rule='G')
```

Assignment 2

Consider $f(x) = \sum_{i=0}^{7} x^i$. Find the analytical solution to $\int_0^1 f(x)dx$. Approximate $\int_0^1 f(x)dx$ using Gaussian quadrature of degree $N = [2, 3, 4]$. What do you observe?

Assignment 3

Consider $f(x) = x$ and $g(x) = x^2$. Compute $\frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} f(x) \exp(-\frac{x^2}{2})dx$ and $\frac{1}{\sqrt{2\pi}} \int_{\mathbb{R}} g(x) \exp(-\frac{x^2}{2})dx$ using Gaussian quadrature of degree $N = 2$. What do you observe? What are the analytical results?