A brief note on sparse quadrature

Alen Alexanderian*

Abstract

We provide a brief introduction to sparse grids for computing integrals of multivariate functions.

1 Multivariate quadrature formulas

Consider \( f : S \rightarrow \mathbb{R} \), with \( S = [-1,1]^d \), where \( d \) is an integer greater than one. We consider the problem of computing \( \int_S f(x) \, dx \). We use univariate quadrature formulas of the form

\[
I^1_i(g) = \sum_{i=1}^{N_i} w_i g(x_i),
\]

as the building blocks for constructing higher dimensional quadrature formulas. Here \( \ell \) indicates the resolution level of the quadrature rule. The most straightforward approach is a full-tensor construction. For simplicity, let’s consider the case where we use the \( I^1_i \) in each coordinate direction. We can define,

\[
I^d_{FT}(f) = \sum_{i_1=1}^{N_1} \cdots \sum_{i_d=1}^{N_d} \left( \prod_{k=1}^{d} w^{(i_k,k)} \right) f(x_1^{(i_1,k)}, \ldots, x_d^{(i_d,k)}) =: \left( I^1_1 \otimes \cdots \otimes I^1_d \right) f.
\]

Notice that the number of nodes in this quadrature formula is \((N_i)^d\); this exponential scaling of the number of quadrature nodes with respect to \( d \) is often referred to as the curse of dimensionality.

As a concrete example of full-tensor grid, we can use the univariate \( n \)-point Gauss–Legendre rule, \( I^1_i(g) = \sum_{j=1}^{n} w^{(j)} g(x^{(j)}) \) to construct a full-tensor quadrature. In the case of \( d = 2 \), the full-tensor quadrature formula is as follows:

\[
I^2_{FT,n}(f) = \sum_{i=1}^{n} \sum_{j=1}^{n} w^{(i)} w^{(j)} f(x_1^{(i)}, x_2^{(j)}).
\]

Due to the curse of dimensionality, applicability of full-tensor quadrature formulas is limited to small dimensions (e.g., \( d \leq 8 \) or so).

2 Sparse quadrature

Consider univariate rules of the form (1.1) for approximating \( I(g) = \int_{-1}^{1} g(x) \, dx \). For \( \ell \geq 1 \), we define difference formulas

\[
\Delta^1_1(g) = I^1_i(g) - I^1_{i-1}(g),
\]

with \( I^1_0 = 0 \). Notice that (1.1) can be written as \( I^1_i(g) = \sum_{i=1}^{\ell} \Delta^1_{i}(g) \). Also, assuming \( E_{\ell}(g) = I^1_{\ell}(g) - I(g) \to 0 \), as \( \ell \to \infty \), we also have \( \Delta^1_{i}(g) \to 0 \), as \( \ell \to \infty \). In particular, when using Gaussian quadrature formulas, the smoother \( g \), the faster \( \Delta^1_1(g) \) approaches zero, as \( \ell \) increases.

Now, let us rewrite (1.2) using the difference formula just introduced:

\[
I^d_{FT}(f) = \left( I^1_i \otimes \cdots \otimes I^1_i \right) f = \left( \prod_{i=1}^{\ell} (\sum_{i_{\ell}=1}^{\ell} \Delta^1_{i_{\ell}}) \otimes \cdots \otimes (\sum_{i_1=1}^{\ell} \Delta^1_{i_1}) \right) f
\]

\[
= \sum_{i_1=1}^{\ell} \cdots \sum_{i_{\ell}=1}^{\ell} (\Delta^1_{i_1} \otimes \cdots \otimes \Delta^1_{i_{\ell}}) f = \sum_{|i| \leq \ell} (\Delta^1_{i_1} \otimes \cdots \otimes \Delta^1_{i_{\ell}}) f.
\]

*North Carolina State University, Raleigh, NC, USA. E-mail: alexanderian@ncsu.edu

Last revised: March 3, 2018.
A brief note on sparse quadrature

Here $|i|_\infty = \max_{1 \leq k \leq d} i_k$. Sparse quadrature rules are obtained by retaining a subset of the terms of the above summation. Denoting $|i|_1 = \sum_{k=1}^d i_k$, the classical Smolyak construction [1] is given by,

$$I_{\text{smol}, \ell}^d(f) = \sum_{|i|_1 \leq \ell + d - 1} (\Delta_{i_1}^1 \otimes \cdots \otimes \Delta_{i_d}^1) f.$$

For detailed description of computing the nodes and weights of a sparse quadrature formula, see [1]. As an illustration, we demonstrate the construction of a sparse quadrature formula for $d = 2$ and $\ell = 3$:

$$I_{\text{smol}, 3}^2 = \sum_{|i|_1 \leq 4} (\Delta_{i_1}^1 \otimes \Delta_{i_2}^1) = (\Delta_1^1 \otimes \Delta_1^1) + (\Delta_1^1 \otimes \Delta_2^1) + (\Delta_2^1 \otimes \Delta_1^1) + (\Delta_2^1 \otimes \Delta_2^1) + (\Delta_3^1 \otimes \Delta_1^1) + (\Delta_1^1 \otimes \Delta_3^1).$$

We show in Figure 1 (top left) the nodes of level 3 sparse quadrature, obtained by using univariate Gauss-Kronrod-Patterson rule that is a nested variant of Gaussian quadrature [2]. The corresponding full-tensor grid is displayed in Figure 1 (top middle), with the sparse grid superimposed. The corresponding multi-index sets are depicted in Figure 1 (top right). The nodes of the univariate quadrature formulas corresponding to levels $\ell = 1, 2,$ and $3$ are shown in Figure 1 (bottom).

3 Further thoughts

In practice, the nodes and weights of full-tensor or sparse quadrature formulas are typically precomputed and ordered so as to write the quadrature formulas as

$$\int_S f(x) \, dx \approx \sum_{j=1}^N W_j f(X_j),$$

where $X_j \in S$, and $W_j, j = 1, \ldots, N$ are the multi-dimensional quadrature nodes and weights.

The textbook [3] has a detailed, yet accessible discussion of sparse quadrature. It is also worth pointing out that sparse grid interpolation schemes can be developed analogously to the sparse grid quadrature rules; see e.g., [3]. Note that, while sparse grid formulas provide a significant improvement over the full-tensor quadrature (and interpolation) formulas as dimension increases (see e.g., Figure 2), they do not solve the curse of dimensionality; sparse quadrature merely tempers the curse of dimensionality, and becomes impractical before $d$ reaches around 40 or so, which is still not really a very high-dimensional problem in real applications. Adaptive sparse grids [4] can enable integration (interpolation) of
higher-dimensional functions. However, for computing integrals of very high dimensional functions quasi Monte Carlo and Monte Carlo integration techniques can provide viable options, especially in the cases of non-smooth integrands.

References


