Integration via Sums

Timothy Hanson

Department of Statistics, University of South Carolina

Stat 740: Statistical Computing

Integration

Integration is a fundamental operation in statistics. Means, variances, and probabilities are integrals; quantiles satify an integral equation; and mixed model likelihoods are obtained via integration.

Only the simplest of integrals can be easily be found in closed form. Programs like *Mathematica* can symbolically integrate many complex functions. Try

```
Integrate[x^2*Sin[x]]
```

at `http://www.wolframalpha.com/`.

However, many intergrals we are interested cannot be found in closed form. We will discuss several numerical approximations to integrals used in statistical computing.

Direct approximations

The simplest approximation to a univariate integral is a *Riemann sum*

$$\int_a^b f(x)dx \approx \Delta \sum_{j=1}^{J} f(x_j), \quad \Delta = \tfrac{b-a}{J}, \quad x_j = a + \Delta(j - \tfrac{1}{2}).$$

This version uses the midpoint of the interval.

Instead of assuming $f(\cdot)$ is approximately constant over subintervals, we can instead try to approximate the function over an interval with a simple polynomial; this leads to the trapezoidal rule (linear approximation):

$$\int_a^b f(x)dx \approx \tfrac{1}{2}\Delta[f(a) + f(b)] + \Delta \sum_{j=1}^{J-1} f(a + j\Delta),$$

and Simpson's rule (quadratic, p. 137 in G & H, 2013).

Direct methods

- Integrals of the form $\int_{-\infty}^{\infty} f(x)g(x)dx$ where $f(x)$ is a density can be integrated over the density's *effective range* for "reasonable" functions $g(x)$, e.g. polynomials. An effective range is a finite interval (a, b) such that $\int_a^b f(x)dx \approx 1$.
- For $\int_{-\infty}^{\infty} f(x)g(x)dx$ with density $f(x)$, if $g(x)$ has tails that die down as quickly (or more quickly) than $f(x)$, the effective range depends on $f(x)g(x)$, *be careful!*
- Of course as J increases the approximations become more accurate. There are methods for bounding the apporimxation error for numerical integration.
- Note that the Riemann sum, trapazoidal rule, and Simpson's rule can all be written $\int_a^b f(x)dx \approx \sum_{j=1}^{J} w_j f(x_j)$.

Example: $E(X^4)$ for $X \sim N(1, 2^2)$.

Quadrature

Quadrature rules also approximate the integral with a sum

$$\int_a^b f(x)dx \approx \sum_{j=1}^{J} w_j f(x_j).$$

Any quadrature rule picks $w_1, \ldots, w_J$ and $x_1, \ldots, x_J$ to provide *an exact result* for polynomials of at most degree $2J - 1$. See

`https://en.wikipedia.org/wiki/Gaussian_quadrature.`

The nodes x_j are the roots of a polynomial from a class of orthogonal polynomials; see 142–148 in G & H.

R has `integrate` built-in, which calls the QUADPACK routines QAGS and QAGI for finite and infinite intervals. *Mathematica* has `Integrate` and NIntegrate.

Quadrature

- Integrals of the form $\int_a^b f(x)dx$ for intervals (a, b) for finite (a, b) are transformed to $\frac{b-a}{2}\int_{-1}^{1} f\left(\frac{b-a}{2}x + \frac{a+b}{2}\right) dx$ and typically evaluated using Gauss-Legendre quadrature.
- Integrals of the form $\int_{-1}^{1} \frac{1}{\sqrt{1-x^2}} g(x)dx$ can be evaluated via Chebyshev-Gauss, $\int_0^\infty e^{-x} g(x)dx$ via Gauss-Laguerre, and $\int_{-\infty}^{\infty} e^{-x^2} g(x)dx$ via Gauss-Hermite.
- If the $g(x)$ is highly localized relative to the weight function, the quadrature breaks down and essentially only uses one point/weight. Adaptive quadrature can help.
- Remember: quadrature is *exact* for polynomials $g(x)$ up to degree $2J - 1$. Useful for obtaining moments! Also recall smooth functions can be approximated by polynomials via Taylor's (differentiable) and Weierstrass (continuous) theorems.

Gauss-Hermite quadrature

The most common is integration with respect to a normal density

$$\int_{-\infty}^{\infty} g(x)\phi(x|\mu,\sigma^2)dx.$$

`fastGHQuad` (univariate) and `MultiGHQuad` (multivariate) perform Gauss-Hermite quadrature and multivariate versions.

`fastGHQuad` computes

$$\int_{-\infty}^{\infty} g(x)e^{-x^2}dx \approx \sum_{j=1}^{J} w_j g(x_j),$$

you pick the number of points/weights J. Note

$$\int_{-\infty}^{\infty} g(y)\frac{1}{\sqrt{2\pi}\sigma}\exp\{-\frac{1}{2\sigma^2}(y-\mu)^2\} = \frac{1}{\sqrt{\pi}}\int_{-\infty}^{\infty} g(\mu+\sqrt{2}\sigma x)e^{-x^2}dx.$$

Example: $E(X^4)$ for $X \sim N(1,2^2)$. What the smallest J so this is exact?

Generalized linear mixed models

GLMMs are one place numerical integration is often employed.

Repeated measures Bernoulli data often takes the form $\{(y_{ij}, \mathbf{x}_{ij})\}$ where $i = 1, \ldots, m$ clusters and $j = 1, \ldots, n_i$ repeated measurements within a cluster. Positive correlation is induced across the $\mathbf{y}_i = (y_{i1}, \ldots, y_{in_i})'$ via random effect u_i

$$\text{logit} P(y_{ij} = 1 | u_i) = \mathbf{x}_i' \beta + u_i, \quad u_1, \ldots, u_m \overset{iid}{\sim} N(0, \sigma^2).$$

The ith likelihood contribution is

$$L_i = p(\mathbf{y}_i | \boldsymbol{\beta}, \sigma) = \int_{-\infty}^{\infty} \underbrace{\left[\prod_{j=1}^{n_i} \frac{\exp\{(\mathbf{x}_{ij}' \boldsymbol{\beta} + u) y_{ij}\}}{1 + \exp\{\mathbf{x}_{ij}' \boldsymbol{\beta} + u\}} \right]}_{p(\mathbf{y}_i | \boldsymbol{\beta}, u)} \underbrace{\frac{1}{\sqrt{2\pi}\sigma} \exp\{-\frac{1}{2\sigma^2} u^2\}}_{p(u|\sigma)} du.$$

Poisson data

Consider the Ache hunting data $\{(m_i, t_i, a_i)\}_{i=1}^{47}$ with hunter-specific random effects:

$$m_i \sim \text{Pois}\{t_i \exp(\beta_0 + \beta_1 a_i + u_i)\}, \;\; i = 1, \ldots, 47.$$

Hunter i's likelihood contribution is, where $\mathbf{x}_i = (1, a_i)'$,

$$L_i = p(m_i|\boldsymbol{\beta}, \sigma) = \int_{-\infty}^{\infty} \exp\{-e^{\mathbf{x}_i'\boldsymbol{\beta}+u} t_i\} e^{m_i(\mathbf{x}_i'\boldsymbol{\beta}+u)} \frac{1}{\sqrt{2\pi}\sigma} \exp\{-\tfrac{1}{2\sigma^2} u^2\} du.$$

Note that $\frac{t^{m_i}}{m_i!}$ is not needed.

How to improve Gauss-Hermite quadrature?

Note that each likelihood contribution L_i consists of two portions as functions of u: one is $N(0, \sigma^2)$ and the other is *approximately* normal when either n_i or t_i are large.

The product of two Gaussians is an unnormalized Gaussian. The multivariate version is

$$\phi_d(\mathbf{x}|\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1)\phi_d(\mathbf{x}|\boldsymbol{\mu}_2, \boldsymbol{\Sigma}_2) \propto \phi_d(\mathbf{x}|\mathbf{V}[\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_2^{-1}\mu_2], \mathbf{V}),$$

$$\mathbf{V} = [\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1}]^{-1}.$$

This implies that G-H quadrature around the origin may not be the most efficient/accurate approximation.

Laplace approximations

Want to approximate $\int_{\boldsymbol{\theta} \in \mathbb{R}^k} f(\boldsymbol{\theta}) d\boldsymbol{\theta}$.

Multivariate Taylor's theorem 2nd-order approximation:

$$g(\boldsymbol{\theta}) \approx g(\hat{\boldsymbol{\theta}}) + (\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})[\nabla g(\hat{\boldsymbol{\theta}})] + \tfrac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})'[\nabla^2 g(\hat{\boldsymbol{\theta}})](\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}).$$

Take $g(\boldsymbol{\theta}) = \log f(\boldsymbol{\theta})$ and find $\hat{\boldsymbol{\theta}}$ such that $\nabla g(\hat{\boldsymbol{\theta}}) = \mathbf{0}$. Then

$$\log f(\boldsymbol{\theta}) \approx \log f(\hat{\boldsymbol{\theta}}) - \tfrac{1}{2}(\boldsymbol{\theta} - \hat{\boldsymbol{\theta}})'[-\nabla^2 \log f(\hat{\boldsymbol{\theta}})](\boldsymbol{\theta} - \hat{\boldsymbol{\theta}}).$$

Exponentiating, recognizing a multivariate normal kernel, and integrating gives

$$\int_{\boldsymbol{\theta} \in \mathbb{R}^k} f(\boldsymbol{\theta}) d\boldsymbol{\theta} \approx f(\hat{\boldsymbol{\theta}})(2\pi)^{k/2} \left| -\nabla^2 \log f(\hat{\boldsymbol{\theta}}) \right|^{1/2}.$$

The more "Gaussian shaped" $f(\cdot)$ is, the better this approximation works.

Adaptive quadrature

One can combine the Laplace approximation idea with quadrature to provide highly accurate approximations. Essentially, this method finds $\hat{\boldsymbol{\theta}}$ s.t. $\nabla \log f(\hat{\boldsymbol{\theta}}) = \mathbf{0}$ and uses the approximation

$$f(\boldsymbol{\theta}) \propto \phi_k(\boldsymbol{\theta}|\hat{\boldsymbol{\theta}}, [-\nabla^2 \log f(\hat{\boldsymbol{\theta}})]^{-1}),$$

to help "guide" the quadrature points.

SAS `proc glimmix` documentation has an excellent overview of how this works.

`fastGHQuad` for univariate integrals

In one dimension, `fastGHQuad` provides a function to automate adaptive quadrature. Say we want

$$\int_{-\infty}^{\infty} g(x) dx,$$

where $g(\cdot)$ is somewhat Gaussian shaped. Find the mode $\hat{x}$ of $g(x)$, e.g. $\frac{d}{dx} \log g(\hat{x}) = 0$ (perhaps via Newton-Raphson) as well as the scale $\sqrt{1/[-\frac{d^2}{dx^2} \log g(\hat{x})]}$. Then `aghQuad(g,mode,scale,rule)` approximates the integral.

`rule=gaussHermiteData(1)` gives a Laplace approximation, otherwise `rule=gaussHermiteData(J)` for $J > 1$ uses Gauss-Hermite quadrature.

Ache Poisson regression

We'll show for

$$p(m_i|\boldsymbol{\beta},\sigma) = \int_{-\infty}^{\infty} \underbrace{\exp\{-e^{\mathbf{x}_i'\boldsymbol{\beta}+u} t_i\} e^{m_i(\mathbf{x}_i'\boldsymbol{\beta}+u)} \frac{1}{\sqrt{2\pi}\sigma} \exp\{-\frac{1}{2\sigma^2} u^2\}}_{g(u)} \, du,$$

that

$$\frac{d}{dx} \log g(u) = -t_i e^{\mathbf{x}_i'\boldsymbol{\beta}+u} + m_i - \frac{u}{\sigma^2},$$

and

$$\frac{d^2}{dx^2} \log g(u) = -t_i e^{\mathbf{x}_i'\boldsymbol{\beta}+u} - \frac{1}{\sigma^2}.$$

Thus, for a given $\boldsymbol{\beta}$, we can perform Newton-Raphson as to find $\hat{u}_i$ via

$$u_j = u_{j-1} - \frac{-t_i e^{\mathbf{x}_i'\boldsymbol{\beta}+u_{j-1}} + m_i - \frac{u_{j-1}}{\sigma^2}}{-t_i e^{\mathbf{x}_i'\boldsymbol{\beta}+u_{j-1}} - \frac{1}{\sigma^2}}.$$

Example: Ache hunting data using adaptive G-H quadrature.

Quadrature & Laplace approximations

- Laplace approximation used in SAS `proc glimmix` with `method=laplace`; also used in the `glmer` function in the `lme4` package as default.
- `proc glimmix` also implements adaptive quadrature via `method=quad(qpoints=6)` using the "guiding" idea on the previous slide. If you omit the `qpoints` option `glimmix` also adaptively chooses the *number* of points/weights. `qpoints=1` is equivalent to `laplace`, but the latter allows fitting more general models in SAS.
- `glmer` also can perform adaptive quadrature via, e.g. `nAGQ=100`.
- Quadrature is also used in SAS `proc nlmixed` to fit general models with random effects.

- Quadrature and/or Laplace approximations can be used in lower-dimensional problems but break down in moderate to high-dimensional problems, e.g. $k \geq 5$.
- MCMC works well in high-dimensional problems, especially for multi-modal likelihoods/posteriors.
- `INLA` is an R package to fit generalized linear mixed models (including models with spatiotemporal information) using *integrated nested Laplace approximations*...very powerful and very fast. Uses Newton-Raphson to obtain posterior mode.
- `BayesX` is another free package to fit generalized linear mixed models, as well as semiparametric survival models, competing risks, etc. Allows for additive predictors, varying coefficient models and spatially-varying random effects. `R2BayesX` package allows fitting in R. Both approximate (fast) inference and more exact (MCMC, slower) inference is possible.
- Next topic: Monte Carlo integration.