Computational Methods in Finance, Lecture 3, Finite Difference Methods for Diffusion Equations.

Jonathan Goodman *
Courant Institute of Mathematical Sciences, NYU

September 24, 1997

1 Introduction

We go more deeply into finite difference methods for solving diffusion equations. We cover the basic ideas of numerical analysis for this problem: order of accuracy. We discuss ways to compute derivatives of computed quantities with respect to problem parameters. This "sensitivity analysis" is very important in financial applications and can be tricky. Much of this material is also covered in the Scientific Computing course and will not be repeated in this course in future years.

2 Finite Differences and Order of Accuracy

It is often helpful to know the "order of accuracy" of a computational method. For one thing, computational methods with a higher order of accuracy tend to give more accurate results. Also, verifying that a code produces answers with

*goodman@cims.nyu.edu, or http://www.math.nyu.edu/faculty/goodman,

the theoretical order of accuracy is a use check of the correctness of the code. For example, if boundary conditions are accidentally applied at grid point one instead of grid point zero, the order of accuracy of the forward Euler method will go from 2 to 1. This buggy code would still converge to the right answer as $\Delta x \to 0$, but fixing the bug would lead to large improvements in accuracy for a given Δx.

We start by examining the one sided difference formula (8) from Lecture 2. The Taylor series expansion for $g(s+h)$ about $h=0$, is

$$g(s+h) \approx g(s) + hg'(s) + \frac{h^2}{2}g''(s) + \frac{h^3}{6}g'''(s) + \cdots \quad .$$

Substituting this into the difference approximation gives

$$g'(s) - \frac{g(s+h)-g(s)}{h} \approx \frac{h}{2}g''(s) + \frac{h^2}{6}g'''(s) + \cdots \quad . \tag{1}$$

When h is small, the right hand side is almost equal to the first term. In particular, it is roughly proportional to h. If we carry the Taylor series to fifth order, we get a similar analysis of the centered approximation to the first derivative:

$$\frac{g(s+h)-g(s-h)}{2h} - g'(s) \approx \frac{h^2}{3}g'''(s) + \frac{h^4}{60}g^{(5)}(s) + \cdots \quad . \tag{2}$$

For the approximation to the second derivative we find

$$\frac{g(s+h)-2g(s)+g(s-h)}{h^2} - g''(s) \approx \frac{h^2}{12}g^{(4)}(s) + \frac{h^4}{360}g^{(6)}(s) + \cdots \quad . \tag{3}$$

These Taylor series calculations can be tedious, but they are not very tedious and we don't have to do them very often.

The expressions (1), (2), and (3), are examples of "asymptotic error expansions". In the general theory of Asymptotic error expansions, we suppose that there is a quantity, A, that we wish to calculate (the "answer"). Given a step, h, we compute an approximation, $A(h)$. An asymptotic error expansion is an expression

$$A(h) - A \approx A_1 h^{p_1} + A_2 h^{p_2} + \cdots \quad . \tag{4}$$

The first term on the right will dominate if h is small and the powers of h are increasing: $p_1 < p_2 < p_3 < \cdots$. The leading power, p_1 determines the order of accuracy of the approximation. In this notation, (1) says that the one sided approximation to the first derivative is first order accurate, the powers being $p_1 = 1$, $p_2 = 2$, and so on. The centered approximation to the first derivative is second order accurate, as is the three point approximation to the second derivative. In both those cases, $p_1 = 2$, $p_2 = 4$, and so on.

This discussion of order of accuracy focuses on what is common to a large class of approximations rather than details that could only be known on a case by case basis, or, in practice, not at all. In this case, the powers, p_1, p_2, and so on, depend only on the approximation formula, while the "error constants", A_1, A_2, ..., depend on the function we are working with. In solving partial differential equations (PDE's from now on), the difference approximations are used to compute the function in the first place, making it unlikely that we would know the error constant in advance. We *can* determine the order of accuracy in advance.

It is possible that an approximation has a different order of accuracy than we expect on the basis of these considerations. The actual order could be higher if, by some fluke, the first error constant were zero. For example, (1) shows that the one sided approximation to the first derivative would be second order accurate at a point s if $g''(s) = 0$. Happy accidents like this can happen in unfortunately chosen model problems, but are rare in actual applications. More commonly, the order of accuracy is lower than we expect because the error constant is infinite or the function is singular. For example, if $g(s) = |s|^{3/2}$ then the one sided approximation to $g'(0)$ at $s = 0$ has order of accuracy $p_1 = 1/2$.[1] This situation is common in financial applications. Unfortunately, the impact of occasional lowering of the order of accuracy intermediate computations on the final accuracy can be hard to determine theoretically. This makes computational convergence analysis (see below) particularly attractive.

There are two important applications of the asymptotic error analysis that depend on the form of the expansion (4) but not on knowing the error constants. They are "convergence analysis" and "Richardson extrapolation". Convergence analysis is a way to determine p_1. Richardson extrapolation is a way to remove the principle error term $A_1 h^{p_1}$ from the approximation, assuming that p_1 (but not A_1) is known.

Convergence analysis is a bit simpler if the exact answer, A, is known.

[1] The reader should be able to check this.

This may be the case in a model problem that we use to validate our code even if it is not in actual application. We run the code with step size h and again with step size $2h$ and get output $A(h)$ and $A(2h)$. A little algebra using (4) shows that

$$\frac{A(2h) - A}{A(h) - A} \approx 2^{p_1} + (2^{p_2} - 2^{p_1})\, h^{p_2 - p_1} A_2/A_1 + \cdots \quad . \tag{5}$$

This shows that if an approximation really is order p_1, then the empirical ratio on the left side of (5) will get closer and closer to 2^{p_1} as h gets smaller and smaller. The most common values for the ratio are 2, for first order, 4, for second order, 16, for fourth order, or 1.414 for order 1/2. Last one occurs most often in solution of stochastic differential equations rather than PDE's. If the exact answer is not known, then (5) can be replaced by

$$\frac{A(4h) - A(2h)}{A(2h) - A(h)} \approx 2^{p_1} + \frac{(2^{p_2} - 2^{p_1})(2^{p_2} - 1)A_2}{(2^{p_1} - 1)A_1} h^{p_2 - p_1} \quad . \tag{6}$$

With this, we can estimate the order of accuracy using empirical data from three runs of the code.

Richardson extrapolation combines output from two code runs to eliminate the leading error term. Let us write expressions for $A(h)$ and $A(2h)$ using (4):

$$\begin{aligned} A(h) &\approx A + A_1 h^{p_1} + A_2 h^{p_2} + \cdots \\ A(2h) &\approx A + 2^{p_1} A_1 h^{p_1} + 2^{p_2} A_2 h^{p_2} + \cdots \quad . \end{aligned}$$

We want a linear combination of these expressions that approximates A with error terms proportional to h^{p_1} eliminated. To do this, we multiply the top equation by 2^{p_1}, subtract the second equation, and rearrange. The result is

$$A = (2^{p_1} A(h) - A(2h)) \,/\, (2^{p_1} - 1) + *** \cdot h^{p_2} + \cdots \quad . \tag{7}$$

The extrapolated estimate on the right side of (7) is of order p_2, which is a higher order than p_1. Applying Richardson extrapolation to the one sided approximation to the first derivative gives a second order approximation (which is different from the centered difference approximation). Applying Richardson extrapolation to the three point approximation to the second derivative gives a fourth order accurate approximation. We would probably use this fourth order approximation if we wanted a fourth order finite difference approximation to the diffusion equation. How this would work is described below.

3 The accuracy of the forward Euler method

In order to get (approximately) the right answer, a finite difference method for a diffusion equation must be *stable* and *accurate*. For the forward Euler method, the stability condition is that all the coefficients should be positive. This section addresses accuracy, more precisely, the order of accuracy of the Forward Euler method. We will show that forward Euler is second order.

We want to determine the order of magnitude[2] of the error. The exact solution, $u(x,t)$, exactly satisfies the heat equation, while the computer approximation, U_j^n, satisfies the forward Euler difference equations exactly. However, $u(x,t)$ does not satisfy the difference equations exactly, nor does U satisfy the heat equation.

The size of the error is determined by the extent to which $u(x,t)$ fails to satisfy the difference equations[3]. For this purpose, define $V_j^n = u(x_j, t_n)$. These are the exact solution sampled at the grid points. The *residual*, ρ_j^n, is the extent to which the exact solution fails to satisfy the difference equations. That is

$$\rho_j^n = -\frac{V_j^{n+1} - V_j^n}{\Delta t} + \frac{1}{2}\frac{V_{j+1}^n - 2V_j^n + V_{j-1}^n}{\Delta x^x} \quad .$$

To estimate the residuals, we us the asymptotic error expansions for finite difference approximations to derivatives, and that fact that u satisfies $-\partial_t u + \frac{1}{2}\partial_x^2 u = 0$. First:

$$\begin{aligned} \frac{V_j^{n+1} - V_j^n}{\Delta t} &= \frac{u(x_j, t_n + \Delta t) - u(x_j, t_n)}{\Delta t} \\ &\approx \partial_t u + \frac{\Delta t}{2}\partial_t^2 u \quad . \end{aligned}$$

I often write simply u for $x(x_j, t_n)$, $\partial_t u$ for $\partial_t u(x_j, t_n)$, and so on. Continuing:

$$\frac{V_{j+1}^n - 2V_j^n + V_{j-1}^n}{\Delta x^2} \approx \frac{\Delta x^2}{12}\partial_x^4 u \quad .$$

Combining these:

$$\rho_j^n \approx \frac{\Delta t}{2}\partial_t^2 u + \frac{\Delta x^2}{24}\partial_x^4 u \quad .$$

[2] In the mathematicians' sense of powers of h rather than in the scientists' sense as a power of 10.

[3] It would be harder to say now the numbers U_j^n fail to satisfy the heat equation, for, among other reasons, these numbers are not even defined at every point.

To simplify the algebra, we now suppose that Δt and $Deltax$ are chosen so that

$$\Delta t = \lambda \Delta x^2 \quad . \tag{8}$$

That is, if we reduce Δx, we reduce Δt by the amount mandated by (8). It is very natural to do this. We already know that $\Delta t \leq \Delta x^2$ is necessary: violating this constraint gives an unstable method. Taking Δt less than this would increase the computer work without much increase inaccuracy. Using (8) and the relation $\partial_2^2 u = \frac{1}{4}\partial_x^4 u$ (which comes from the differentiating the heat equation with respect to t and fiddling), you get

$$\rho_j^n \approx \frac{\Delta x^2}{8}\left(\frac{1}{3} - \lambda\right)\partial_x^4 u \quad . \tag{9}$$

From this we wee that the truncation error (another term for residual) has an expansion whose principal term is proportional to Δx^2. It looks as though the truncation error will be smaller if $\lambda = 1/3$. This is true for the heat equation itself, but remember that this is a model problem. Most of the actual problems have features, such as variable coefficients of lower order terms, that spoil this.

The residual determines the error through the dynamics. Let us define the error

$$E_j^n = V_j^n - U_j^n \quad .$$

We know about the E_j^n that (i) they satisfy the forward Euler difference equations with forcing from the truncation error[4]

$$\frac{E_j^{n+1} - E_j^n}{\Delta t} = \frac{1}{2}\frac{E_{j+1}^n - 2E_j^n + E_{j-1}^n}{\Delta x^2} + \rho_j^n \quad , \tag{10}$$

and that (ii) $E_j^0 = 0$ (we naturally take $U_j^0 = u_0(x_j)$, why make deliberate errors in the initial data?[5]). We imagine that the errors E_j^n are approximately samplings from a smooth error function, $e(x,t)$, in the sense that

$$E_j^n \approx e(x_j, t_n) \quad .$$

Then, the difference equation (9) is approximated by

$$\partial_t e = \frac{1}{2}\partial_x^2 e + \rho \quad . \tag{11}$$

[4]Subtract the U from the V equation get this.

[5]It may be wise not to do this if the initial data is singular.

From (10) it follows that e, and therefore also E, has the same power of Δx as ρ, second order in the case of forward Euler. This is an explanation of the second order accuracy of the forward Euler method for the heat equation.

4 Forward Euler for more general diffusion equations

More general diffusion equations differ from the heat equation in several ways.

(i) Time may go backwards instead of forwards.

(ii) There may be variable coefficients.

(iii) There may be lower order terms in the equation.

(iv) There may be more than one "factor" (space dimension) and mixed partial derivatives.

(v) We may want to use a non uniform grid.

I will discuss these point by point.

(i) To evolve forward in time, we want to compute the approximation at time $n+1$ in terms of the approximation at time n. This makes the numerical approximation "evolve" in the same sense in which the solution to the diffusion equation does. If the diffusion equation is a backward equation (as it is most of the time in financial applications), then we want to compute an approximation at time $n-1$ from the already computed approximation at them n. This can be done by using a "backward" time derivative

$$\partial_t f \approx \frac{F_j^n - F_j^{n-1}}{\Delta t} \tag{12}$$

in place of the forward derivative

$$\partial_t u \approx \frac{U_j^{n+1} - U_j^n}{\Delta t} \quad .$$

The right side of (11) will allow us to compute the F_j^{n-1} in terms of the F_j^n if all the space derivatives are discretized using F values at time

n. This marching backwards in time is still called the *forward* Euler method. The backward Euler method will be discussed in Lecture 4. The order of accuracy is still 2.

(ii) We can generally find simple second order approximations to variable coefficient derivative terms such as

$$\mu(x)\partial_x^2 u \approx \mu(x_j)\frac{U_{j+1}^n - 2U_j^n + U_{j-1}^n}{\Delta x^2} \quad ,$$

$$\partial_x^2\left(\mu(x)f\right) \approx \frac{\mu(x_{j+1})U_{j+1}^n - 2\mu(x_j)U_j^n + \mu(x_{j-1})U_{j-1}^n}{\Delta x^2} \quad ,$$

$$\partial_x\left(\mu\partial_x u\right) \approx \frac{1}{\Delta x}\left(\mu(x_{j+1/2})\frac{U_{j+1}^n - U_j^n}{\Delta x} - \mu(x_{j-1/2})\frac{U_j^n - U_{j-1}^n}{\Delta x}\right) \quad .$$

If μ is independent of x, these approximations are all the same as the one for constant μ we used for the heat equation. Coefficients effect the stability conditions. For constant μ, the condition is

$$\Delta t \leq \mu\Delta x^2 \quad . \tag{13}$$

This makes dimensional sense because μ has units *length*^{2}/*time*. For constant μ, the forward Euler method is fourth order accurate if $\Delta t = \mu\Delta x^2/3$, but it is generally impossible to achieve this for all x if μ is not constant. For variable μ, we should make (12) hold for all μ values in the computational domain. Always, this stability condition can be derived from the condition that the formula for U_j^{n+1} in terms of the U_j^n has all positive coefficients.

(iii) To achieve second order accuracy, we must use second order approximations to the first derivative terms. For example

$$\partial_x\left(a(x)u\right) \approx \frac{1}{2\Delta x}\left(a(x_{j+1})U_{j+1}^n - a(x_{j-1})U_{j-1}^n\right) \quad ,$$

$$a(x)\partial_x u \approx a(x_j)\frac{U_{j+1}^n - U_{j-1}^n}{2\Delta x} \quad .$$

the first derivative term will usually not effect the stability of the method unless a is very large relative to μ.

(iv) For multidimensional problems we have more spatial derivatives to discretize. Most of the terms involve derivatives with respect to just one variable. For example, to compute $u(x, y, t)$ (a two factor model), take $x_j = j\Delta x$, $y_k = k\Delta y$, $U^n_{jk} \approx u(x_j, t_k, t_n)$, and

$$\partial_x^2 u \approx \frac{U^n_{j+1,k} - 2U_{jk} + U^n_{j-1,k}}{\Delta y^2} \; .$$

A second order approximation to the mixed partial is

$$\partial_x \partial_y u \approx \frac{U^n_{i+1,j+1} - U^n_{i+1,j-1} - U^n_{i-1,j+1} + U^n_{i-1,j-1}}{\Delta x \Delta y} \; .$$

If the "off diagonal" diffusion coefficients (such as $\mu_1 2$) are not zero[6] using these formulas will lead to a forward Euler method that is stable for $\Delta t / Deltax^2$ and $\Delta t / Deltay^2$ small enough. However, the coefficients will not all be positive. There is a second order approximation with positive coefficients but it takes longer to describe and involves discretizing all the second derivative terms together rather than adding separate approximations to the individual terms.

(v) Very often it makes sense not to take all the space steps the same size. For example, in Black-Scholes, we often fix $\Delta S/S$ rather than ΔS. It can be cumbersome to find simple second order accurate difference approximations in these cases. I recommend the following. First decide on a change of variable, such as $S = e^x$. Then write the equation in terms of the new variable[7]. Then discretize the transformed equation with uniform space intervals.

5 Computing sensitivities

Sensitivities are derivatives of the answer with respect to parameters in the problem. For example, the Black-Scholes problem has a volatility parameter, σ. Vega ($= \Lambda$) is the derivative of f with respect to σ. One way to compute Λ and other sensitivities is to compute, for example, $f(\Sigma)$, $f(\sigma + \Delta\sigma)$ and use

[6] a situation that occurs when the noise sources for several factors are correlated

[7] This uses the chain rule a bit.

a finite difference approximation to $\partial_\sigma f$[8]. This approach has the drawback that the errors made in approximating f may be amplified when we divide by $\Delta\sigma$. This problem may be worse when $\Delta\sigma$ is smaller. In other cases the approximation error for f depends smoothly on σ, and therefore, largely cancels out when computing $f(\sigma+\Delta\sigma)-f(\sigma)$. For Monte Carlo approximation, the errors generally do not cancel. For simple difference approximations, European options, smooth initial data, the errors may well cancel.

A different approach to estimating sensitivities is to calculate the PDE the sensitivity satisfies and solve it directly. For the Black-Scholes equation:

$$\partial_t f + rS\partial_S f + \frac{1}{2}\sigma^2 S^2 \partial_S^2 f = rf \quad ,$$

this gives

$$\partial_t \frac{\partial f}{\partial_\sigma} + rS\partial_S \frac{\partial f}{\partial_\sigma} + \frac{1}{2}\sigma^2 S^2 \partial_S^2 \frac{\partial f}{\partial_\sigma} + \sigma S^2 \partial_S^2 f = r\frac{\partial f}{\partial_\sigma} \quad , \tag{14}$$

At the final time, $\partial_\sigma f = 0$. Thus, we can solve (13) to estimate $\partial_\sigma f$ at the same time we compute f itself. One potential problem is that $\partial_S^2 f$ might be very large. possibly leading to inaccurate numerical results for $\partial_\sigma f$.

[8] It would probably be better to compute $f(\sigma+\Delta\sigma)$ and $f(\sigma-\Delta\sigma)$ and use a centered approximation.