# 18.303 notes on finite differences

S. G. Johnson

September 10, 2013

The most basic way to approximate a derivative on a computer is by a difference. In fact, you probably learned the *definition* of a derivative as being the limit of a difference:

$$u'(x) = \lim_{\Delta x \to 0} \frac{u(x + \Delta x) - u(x)}{\Delta x}.$$

To get an approximation, all we have to do is to remove the limit, instead using a small but non-infinitesimal $\Delta x$. In fact, there are at least three obvious variations (these are *not* the *only* possibilities) of such a difference formula:

$$\begin{aligned} u'(x) &\approx \frac{u(x+\Delta x) - u(x)}{\Delta x} && \text{forward difference} \\ &\approx \frac{u(x) - u(x-\Delta x)}{\Delta x} && \text{backward difference} \\ &\approx \frac{u(x+\Delta x) - u(x-\Delta x)}{2\Delta x} && \text{center difference,} \end{aligned}$$

with all three of course being equivalent in the $\Delta x \to 0$ limit (assuming a continuous derivative). Viewed as a *numerical method*, the key questions are:

- How big is the error from a nonzero $\Delta x$?
- How fast does the error vanish as $\Delta x \to 0$?
- How do the answers depend on the difference approximation, and how can we *analyze* and *design* these approximations?

Let's try these for a simple example: $u(x) = \sin(x)$, taking the derivative at $x = 1$ for a variety of $\Delta x$ values using each of the three difference formulas above. The exact derivative, of course, is $u'(1) = \cos(1)$, so we will compute the error $|\text{approximation} - \cos(1)|$

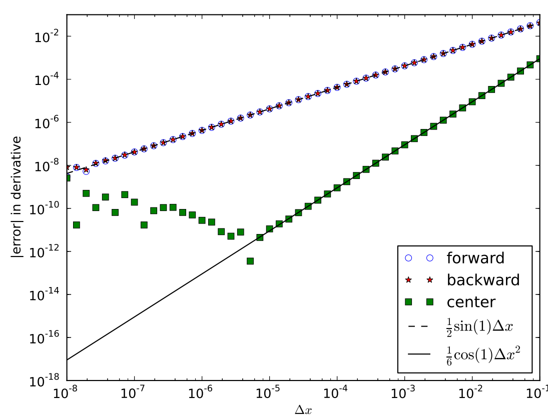


Figure 1: Error in forward- (blue circles), backward- (red stars), and center-difference (green squares) approximations for the derivative $u'(1)$ of $u(x) = \sin(x)$. Also plotted are the predicted errors (dashed and solid black lines) from a Taylor-series analysis. Note that, for small $\Delta x$, the center-difference accuracy ceases to decline because rounding errors dominate (15–16 significant digits for standard double precision).

versus $\Delta x$. This can be done in Julia with the following commands (which include analytical error estimates described below):

```
x = 1
dx = logspace(-8,-1,50)
f = (sin(x+dx) - sin(x)) ./ dx
b = (sin(x) - sin(x-dx)) ./ dx
c = (sin(x+dx) - sin(x-dx)) ./ (2*dx)
using PyPlot
loglog(dx, abs(cos(x) - f), "o",
       markerfacecolor="none",
       markeredgecolor="b")
loglog(dx, abs(cos(x) - b), "r*")
loglog(dx, abs(cos(x) - c), "gs")
loglog(dx, sin(x) * dx/2, "k--")
loglog(dx, cos(x) * dx.^2/6, "k-")
legend(["forward", "backward", "center",
   L"$\frac{1}{2}\sin(1) \Delta x$",
   L"$\frac{1}{6}\cos(1) \Delta x^2$"],
   "lower right")
xlabel(L"$\Delta x$")
ylabel("|error| in derivative")
```

The resulting plot is shown in Figure 1. The obvious conclusion is that the forward- and backward-difference approximations are about the same, but that center differences are dramatically more accurate—not only is the absolute value of the error smaller for the center differences, but the *rate* at which it goes to zero with $\Delta x$ is also qualitatively faster. Since this is a log–log plot, a straight line corresponds to a power law, and the forward/backward-difference errors shrink proportional to $\sim \Delta x$, while the center-difference errors shrink proportional to $\sim \Delta x^2$! For very small $\Delta x$, the error appears to go crazy—what you are seeing here is simply the effect of roundoff errors, which take over at this point because the computer rounds every operation to about 15–16 decimal digits.

We can understand this completely by analyzing the differences via *Taylor expansions* of $u(x)$. Recall that, for small $\Delta x$, we have

$$u(x+\Delta x) \approx u(x)+\Delta x\, u'(x)+\frac{\Delta x^2}{2}u''(x)+\frac{\Delta x^3}{3!}u'''(x)+\cdots .$$

$$u(x-\Delta x) \approx u(x)-\Delta x\, u'(x)+\frac{\Delta x^2}{2}u''(x)-\frac{\Delta x^3}{3!}u'''(x)+\cdots .$$

If we plug this into the difference formulas, after some algebra we find:

$$\text{forward difference} \approx u'(x)+\frac{\Delta x}{2}u''(x)+\frac{\Delta x^2}{3!}u'''(x)+\cdots ,$$

$$\text{backward difference} \approx u'(x)-\frac{\Delta x}{2}u''(x)+\frac{\Delta x^2}{3!}u'''(x)+\cdots ,$$

$$\text{center difference} \approx u'(x) + \frac{\Delta x^2}{3!}u'''(x) + \cdots .$$

For the forward and backward differences, the error in the difference approximation is dominated by the $u''(x)$ term in the Taylor series, which leads to an error that (for small $\Delta x$) scales linearly with $\Delta x$. For the *center*-difference formula, however, the $u''(x)$ term *cancelled* in $u(x+\Delta x) - u(x-\Delta x)$, leaving us with an error dominated by the $u'''(x)$ term, which scales as $\Delta x^2$.

In fact, we can even quantitatively predict the error magnitude: it should be about $\sin(1)\Delta x/2$ for the forward and backward differences, and about $\cos(1)\Delta x^2/6$ for the center differences. Precisely these predictions are shown as dotted and solid lines, respectively, in Figure 1, and match the computed errors almost exactly, until rounding errors take over.

Of course, these are not the only possible difference approximations. If the center difference is devised so as to exactly cancel the $u''(x)$ term, why not also add in additional terms to cancel the $u'''(x)$ term? Precisely this strategy can be pursued to obtain *higher-order difference approximations*, at the cost of making the differences more expensive to compute [more $u(x)$ terms]. Besides computational expense, there are several other considerations that can limit one in practice. Most notably, practical PDE problems often contain *discontinuities* (e.g. think of heat flow or waves with two or more materials), and in the face of these discontinuities the Taylor-series approximation is no longer correct, breaking the prediction of high-order accuracy in finite differences.