## 1: First derivative matrix

(a,b)

$$f_1{}' = \frac{-3f_1 + 4f_2 - f_3}{2\Delta} + \frac{1}{3}\Delta^2 f_1{}''' + \ldots, \tag{1}$$

and similar for $f_N{}'$ . Everyone got the coefficients right and showed that the error is $\sim \Delta^2$.

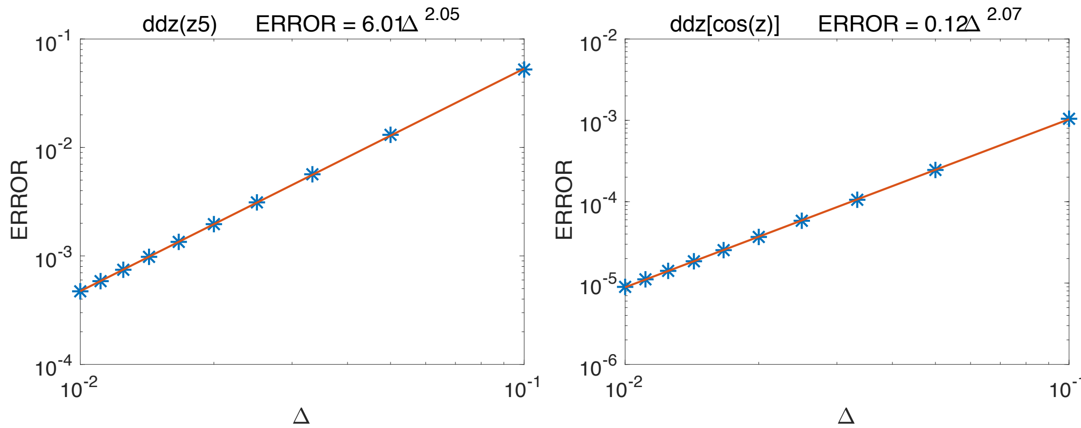


**Figure 1:** Approximating the first derivative of $f = z^5$ and $f = \cos z$. The error is the absolute difference between the approximation and the true value. Orange line = linear fit.

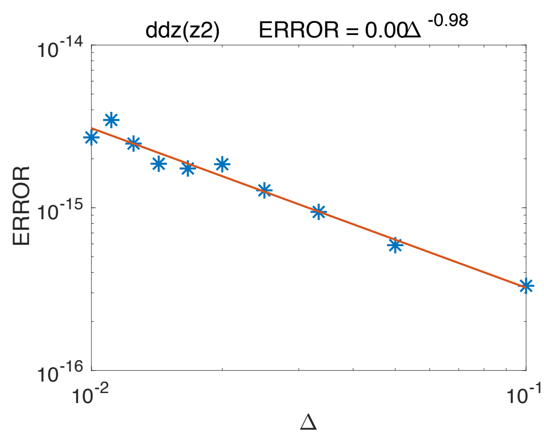


**Figure 2:** Approximating the first derivative of $f = z^2$.

(c,d) Numerical results: The error test shows that the error $\propto \Delta^2$ in every case (figure 1) except $f = z^2$ (figure 2).

These calculations have two sources of error: *truncation error* and *roundoff error*.

- *Truncation error* results from the fact that we *truncate* the Taylor series expansions when designing our finite difference formulas. If we could somehow use the whole, infinite Taylor series there would be no truncation error.
- *Roundoff error* occurs because the computer can only store real numbers up to a certain number of significant digits, i.e., it must round them off. Try calculating $2^{1/2} - 4^{1/4}$ in Matlab (or whatever you use). You should get zero, but you don't. (We get 2.2204e-16.) That's roundoff error.

Usually, truncation error is much larger than roundoff error. But when we approximate the derivative of a low-order polynomial like $z^2$, the terms that we neglected (truncated) are all zero; hence, so is the truncation error. In that case only roundoff error remains. Roundoff error is basically a stream of very small random numbers that all get divided by $\Delta$ in the process of computing $D_{ij} f_i$. The result is, on average, proportional to $1/\Delta$. That's what you see in figure 2.