559

Numerical Differentiation Methods

Fish 559; Lecture 11

Symbolic vs Numerical Differentiation

- Differentiation is algorithmic - there is no function that cannot be differentiated (given patience and a large piece of paper).
- Many packages (including R) include symbolic differentiation routines. These apply an algorithm to provide code which computes the derivatives analytically.
- Numerical differentiation involves applying *approximations* to calculate the value of the derivative at a given point.

Symbolic differentiation in R-I

- The function "D", called as follows, does symbolic differentiation:

```
my.deriv <- function(mathfunc, var) {
  temp <- substitute(mathfunc)
  name <- deparse(substitute(var))
  D(temp, name)
}
```

- This is NOT a a very smart function:

```
> my.deriv(x*x^2,x)
x^2 + x * (2 * x)
```

Symbolic differentiation in R-II

- D can, however, be quite useful:

```
> my.deriv(x*(1-x)-a*x*v/(x+d),x)
(1 - x) - x - (a * v/(x + d) - a * x * v/(x + d)^2)
> my.deriv(x*(1-x)-a*x*v/(x+d),v)
 - (a * x/(x + d))
```

- However, D may crash for complicated formulae.

The Deriv function-I

- Numerical methods often require the derivatives of functions. R includes the function “deriv” which calculates gradients:

  ff <- deriv(expression, name, function name)

- For example:

```
ff<-deriv(~x*(1-x)-a*x*v/(x+d),c("x","v"),
function(x,v,a,d) NULL, formal=T)
```

- This creates a function *ff*.

The Deriv function-II

```
function(x, v, a, d)
{
        .expr1 <- 1 - x
        .expr3 <- a * x
        .expr4 <- .expr3 * v
        .expr5 <- x + d
        .value <- (x * .expr1) - (.expr4/.expr5)
        .grad <- array(0, c(length(.value), 2), list(NULL, c("x", "v")))
        .grad[, "x"] <- (.expr1 - x) - (((a * v)/.expr5) -
(.expr4/(.expr5^2)))
        .grad[, "v"] <-  - (.expr3/.expr5)
        attr(.value, "gradient") <- .grad
        .value
}
```

Calculating Derivatives Numerically-I

- Sometimes calculating derivatives analytically can get rather tedious. For example, for the dynamic Schaefer model:

$$SSQ = \sum_t [\ell n I_t - \ell n(q\tilde{B}_t)]; \qquad \tilde{B}_{t+1} = \tilde{B}_t + r\,\tilde{B}_t\,(1-\tilde{B}_t / K) - C_t$$

$$\frac{dSSQ}{dr} = \frac{d}{dr}\sum_t [\ell n I_t - \ell n(q\tilde{B}_t)]^2 = -2\sum_t [\ell n I_t - \ell n(q\tilde{B}_t)]\frac{1}{\tilde{B}_t}\frac{d\tilde{B}_t}{dr}$$

$$\frac{d\tilde{B}_t}{dr} = \frac{d}{dr}[B_{t-1} + \tfrac{r}{2}B_{t-1}(1-B_{t-1}/K) - \frac{C_{t-1}}{2}] = \frac{d}{dr}[(1+\tfrac{r}{2})B_{t-1} - \tfrac{r}{2K}(B_{t-1})^2 - \frac{C_{t-1}}{2}]$$

$$= \tfrac{B_{t-1}}{2} + (1+\tfrac{r}{2})\frac{dB_{t-1}}{dr} - \tfrac{r}{k}B_{t-1}\frac{dB_{t-1}}{dr}$$

- I think you get the picture…

Calculating Derivatives Numerically-II

- Methods of numerical differentiation rely on Taylor series' approximations to functions, i.e.:

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2} f''(x) + \frac{h^3}{6} f'''(x) + \ldots.$$

$$g(x+h, y+k) = g(x,y) + hg_x(x,y) + kg_y(x,y) + 0.5(hhg_{xx}(x,y) + hkg_{xy}(x,y) + kkg_{yy}(x,y)) + ..$$

Calculating Derivatives Numerically-III

- Two common approximations to $f'(x)$ exist. Both rely on two function evaluations – which is to be preferred?

$$f'(x) \approx \frac{f(x+h) - f(x-h)}{2h} \qquad\qquad f'(x) \approx \frac{f(x+h) - f(x)}{h}$$

- We can compare them in terms of how well they mimic the Taylor series expansion of *f*

Calculating Derivatives Numerically-IV

$$f(x+h)-f(x-h)$$
$$=[f(x)+hf'(x)+\tfrac{h^2}{2}f''(x)+\tfrac{h^3}{6}f'''(x)+..]-[f(x)-hf'(x)+\tfrac{h^2}{2}f''(x)-\tfrac{h^3}{6}f'''(x)+...]$$
$$=2hf'(x)+\tfrac{h^3}{3}f'''(x)+...$$

Therefore:

$$\frac{f(x+h)-f(x-h)}{2h}=f'(x)+\tfrac{h^2}{6}f'''(x)+...$$

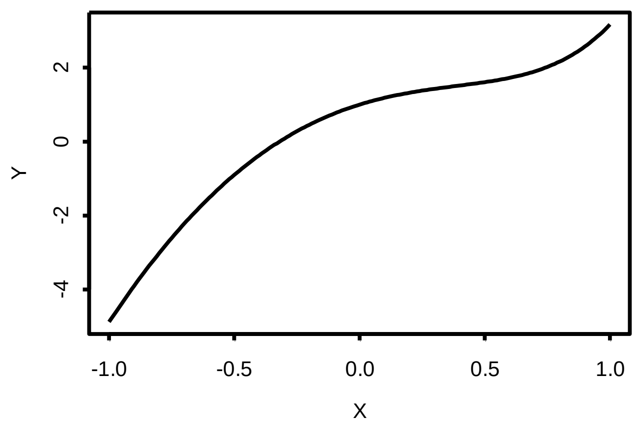
Applying the same approach to $f'(x)\approx\dfrac{f(x+h)-f(x)}{h}$ leads to:

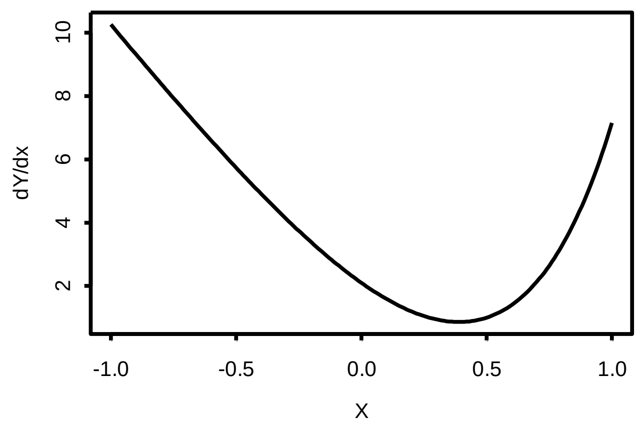
$$\frac{f(x+h)-f(x)}{h}=f'(x)+\tfrac{h}{2}f''(x)+...$$

The "central difference" approach is therefore clearly preferable if "h is small".

Does the Previous Result Hold up in Reality-I?

$$y(x) = \exp(2.1x) - 5x^2$$





Derivative at x=0.5 (h=0.0001)

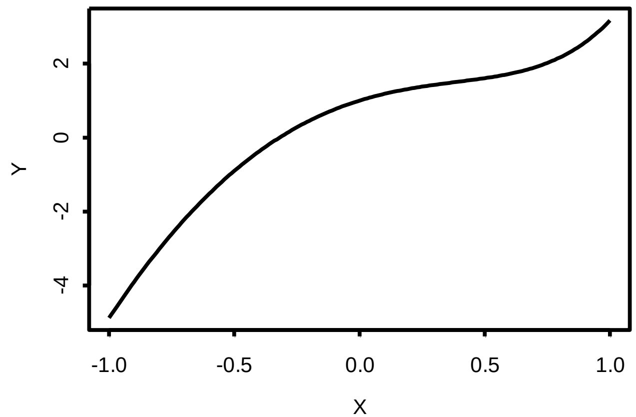
True: 1.001067

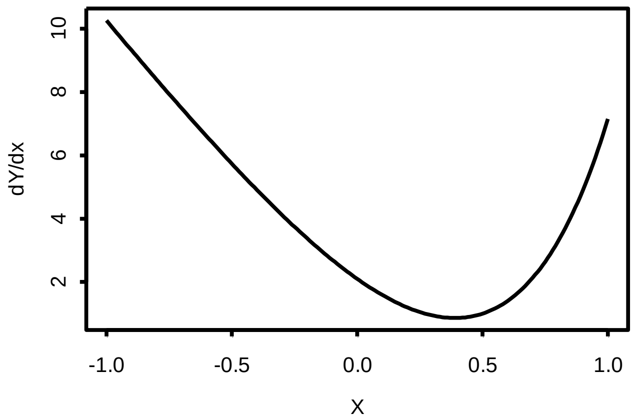
Central difference: 1.001067

Right difference: 1.001198

Does the Previous Result Hold up in Reality-II?

$$y(x) = \exp(2.1x) - 5x^2$$





Derivative at x=0.5 (h=h <- 0.000000000000001)

True: 1.001067

Central difference: 2.220446

Right difference: 4.440892

So what happened?

559

Calculating Derivatives Numerically-V

- When I need an accurate approximation to a derivative, I have tended to use the four-point "central difference" approximation.
- Central differences perform badly when calculating derivatives on a boundary.

Calculating Derivatives Numerically

$$\frac{df^{2}(x)}{dx^{2}} \approx \frac{f(x+h) - 2f(x) + f(x-h)}{h^{2}}$$

$$\frac{df^{2}(x,y)}{dx\,dy} \approx \frac{f(x+h,y+k) - f(x-h,y+k) - f(x+h,y-k) + f(x-h,y-k)}{4hk}$$

- The accuracy of the approximation depends on the number of terms and the size of *h* / *k* (smaller – but not too small – is better)