DATA 1010
In-class exercises
Samuel S. Watson
19 September 2018

Problem 1

Consider the sequence $\{\mathrm{mod}(3 \cdot 2^n, 11)\}_{n=1}^{100}$. Use Julia to show that each number from 1 to 10 appears exactly 10 times in this sequence. Also, use Julia to show that a_{2k} is smaller than a_{2k-1} for far more than half the values of k from 1 to 50. Hint: `countmap(a)` tells you how many times each element in the collection `a` appears. To use this function, do `using StatsBase` first.

Repeat these tests on the sequence whose kth term is the kth digit in the decimal representation of π: `reverse(digits(floor(BigInt,big(10)^99*pi)))`.

Solution

Only 10 of the 50 pairs have the even-indexed term larger than the odd-indexed term:

```
using StatsBase
a = [6]
for i=1:99
    push!(a,mod(2a[end],11))
end
countmap(a) # every number appears 10 times
sum(a[2i] < a[2i-1] for i=1:50) # returns 40
```

Repeating with the digits of π, we find that 27 of the first hundred blocks of 2 have even-indexed digit smaller than the one before it.

Problem 2

Use difference quotients to approximate the derivative of $f(x) = x^2$ at $x = \frac{2}{3}$, with $\epsilon = 2^k$ as k ranges from -60 to -20. What is the least error over these values of k? How does that error compare to machine epsilon?

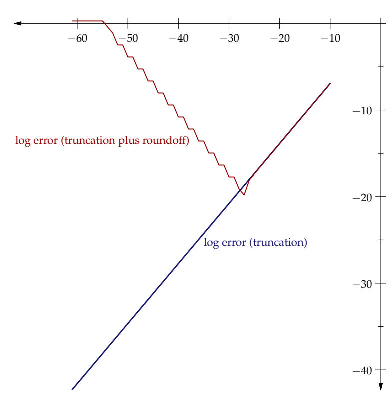
Solution

The block

```
diffquotient(f,x,ϵ) = (f(x+ϵ) - f(x))/ϵ
m = minimum([abs(diffquotient(x->x^2,2/3,2.0^k) - 4/3)
                                        for k = -60:-20])
```

returns 2.48×10^{-9}. This error is more than *ten million* times larger than we could hope for just from rounding to the nearest float.

```
m / (nextfloat(4/3) - 4/3)
```



Problem 3

In this exercise, we will explain why

$$f\left(\begin{bmatrix} x & 1 \\ 0 & x \end{bmatrix}\right) = \begin{bmatrix} f(x) & f'(x) \\ 0 & f(x) \end{bmatrix}, \tag{3.1}$$

for any polynomial f.

(i) Check that (3.1) holds for the identity function (the function which returns its input) and for the function which returns the multiplicative identity.

(ii) Check that if (3.1) holds for two differentiable functions f and g, then it holds for the sum $f + g$ and the product fg.

(iii) Explain why (3.1) holds for any polynomial function $f(x)$.

Solution

(i) If f is the identity function, then both sides of (3.1) reduce to $\begin{bmatrix} x & 1 \\ 0 & x \end{bmatrix}$. If f returns the multiplicative identity, then both sides reduce to the identity matrix.

(ii) We have

$$\begin{bmatrix} f(x) & f'(x) \\ 0 & f(x) \end{bmatrix} \begin{bmatrix} g(x) & g'(x) \\ 0 & g(x) \end{bmatrix} = \begin{bmatrix} f(x)g(x) & f'(x)g(x) + f(x)g'(x) \\ 0 & f(x)g(x) \end{bmatrix},$$

which in turn is equal to $\begin{bmatrix} f(x)g(x) & (f(x)g(x))' \\ 0 & f(x)g(x) \end{bmatrix}$ by the product rule. The result for $f + g$ works similarly, with linearity of the derivative in place of the product rule.

(iii) The set of functions which satisfies (3.1) includes 1 and x and is closed under multiplication and addition. Therefore, this set of functions at least includes all polynomials.

Problem 4

Use automatic differentiation to find the derivative of $f(x) = (x^4 - 2x^3 - x^2 + 3x - 1)e^{-x^4/4}$ at the point $x = 2$. Compare your answer to the true value of $f'(2)$.

Hint: You'll want to define f using

```julia
using LinearAlgebra
f(t) = exp(-t^2/4)*(t^4 - 2t^3 - t^2 + 3t - I)
```

where `I` is an object which is defined to behave like multiplicative identity (note that subtracting the identity matrix is the appropriate matrix analogue of subtracting 1 from a real number).

Also, to help check your answer, here's the symbolic derivative of f:

```julia
df(t) = (-t^5 + 2*t^4 + 9*t^3 - 15*t^2 - 3*t + 6)*exp(-t^2/4)/2
```

Solution

We define f as suggested and then calculate `f([2 1; 0 2])[1,2]`. The result is *exactly the same* as `df(2)` and 7.46×10^{-17} different from `df(big(2))`. So we see that automatic differentiation gives a major improvement over the difference quotient approach in this instance.