Sin(x) = x Fixed Point Iteration

Cody Frisby

9/11/2017

First I define a function called `fixed_plot` where I combine fixed point iteration and plotting (I'm a visual guy).

```
fixed_plot <- function(x, FUN = cos, n = 10) {
  xmin <- -abs(x)
  xmax <- abs(x)
  plot(1:abs(x), xlim = c(xmin - 1, xmax + 1),
       ylim = c(-2, 2), type = "n", xlab = "", ylab = "")
  grid()
  abline(h = 0, lty = 3) # x axis
  abline(v = 0, lty = 3)# y axis
  abline(coef = c(0, 1), col = "dodgerblue", lty = 3)
  curve(FUN(x), from = xmin - 2, xmax + 2, lwd = 2, add = TRUE)
  x0 <- x
  i <- 1
  m <- matrix(ncol = 2)
  for(i in 1:n) {
    x0 <- FUN(x) * x^3/6 # calculate f(x)
    r <- x0/x
    #a <- 1 - (x^2/6)
    points(x, x0, col = i)
    x <- x0
    #print(x); print(r); print(a)
    #Sys.sleep(0.1)
    m <- rbind(m, c(x0, r))
  }
  points(x0, FUN(x0), pch = "X", cex = 2)
  m <- m[-1, ] # remove the stupid NA row
  colnames(m) <- c("x0", "ratio errors")
  return(m)
}
```

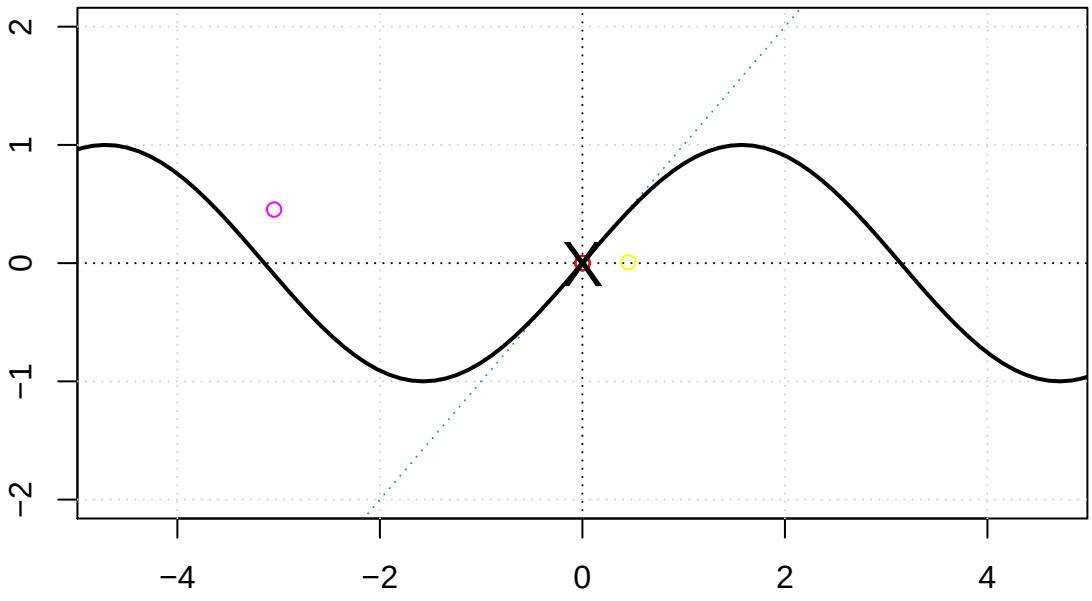
Notice that I have multiplied $sin(x)$ by $\frac{x^3}{6}$ which makes

$$f(x) = sin(x)\frac{x^3}{6}$$

which you can easily identify as the second term in the taylor series expansion of $sin(x)$.

So, this little "fudge function" makes my function converge super fast now, but ONLY if my starting point is less than $|3.6|$. If I choose any number greater than that my iteration diverges, and rapidly. Here's some examples with only 5 iterations.

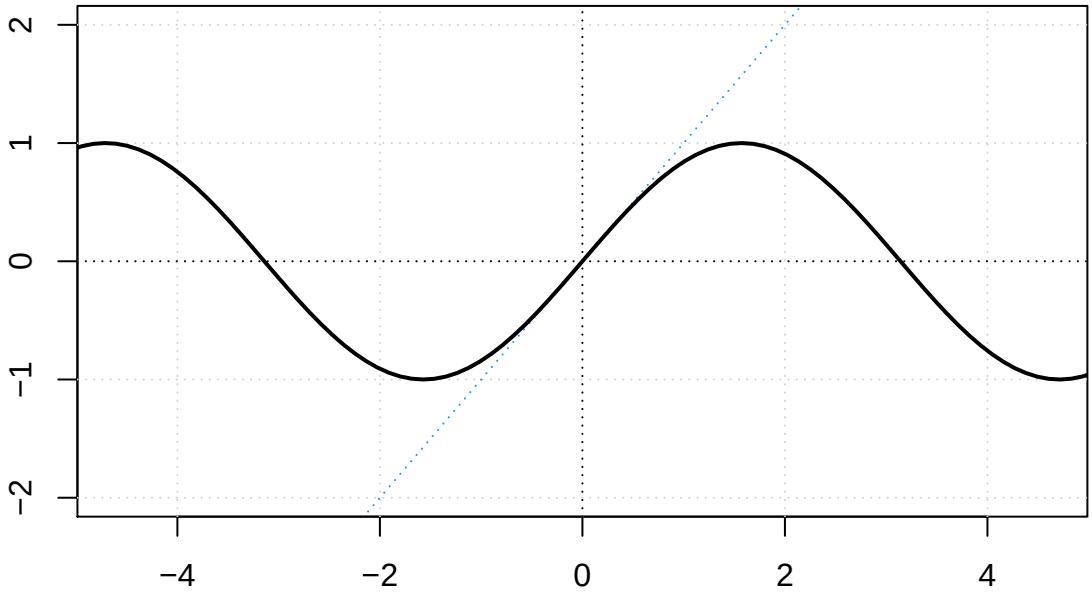
```
fixed_plot(3.6178, FUN = sin, n = 10)
```




```
                 x0   ratio errors
 [1,]  -3.617747e+00  -9.999854e-01
 [2,]  -3.617217e+00   9.998535e-01
 [3,]  -3.611910e+00   9.985329e-01
 [4,]  -3.558934e+00   9.853331e-01
 [5,]  -3.045225e+00   8.556564e-01
 [6,]   4.528619e-01  -1.487121e-01
 [7,]   6.772742e-03   1.495543e-02
 [8,]   3.506741e-10   5.177726e-08
 [9,]   2.520364e-39   7.187199e-30
[10,]  6.725149e-156  2.668324e-117
```

I'm at **zero** super fast as you can see. Now try a starting guess greater than that.

```
fixed_plot(3.61781, FUN = sin, n = 10)
```




```
      x0  ratio errors
```

```
 [1,] -3.617847e+00 -1.000010e+00
 [2,] -3.618220e+00  1.000103e+00
 [3,] -3.621951e+00  1.001031e+00
 [4,] -3.659396e+00  1.010338e+00
 [5,] -4.042573e+00  1.104711e+00
 [6,] -8.631834e+00  2.135233e+00
 [7,]  7.636520e+01 -8.846927e+00
 [8,]  6.109804e+04  8.000769e+02
 [9,]  1.278146e+13  2.091959e+08
[10,]  2.211242e+38  1.730039e+25
```

And we diverge super fast.

What's going on here?