Bisection Method

Andrés Mateo Otálvaro, Santiago Suarez Perez,Daniel Ermilson Velásquez

September 11, 2015

Algorithm 1 Bisection

```
1: procedure BISECTION
2:     X_i
3:     X_s
4:     tol
5:     iter
6:     Y_i = f(X_i)
7:     Y_s = f(X_s)
8:     if Y_i * Y_s > 0 then
9:         "The roots are not equal"
10:    else if Y_i = 0 then
11:        "X_i is a root"
12:    else if Y_s = 0 then
13:        "X_s is a root"
14:    else
15:        X_m = (Y_i*X_s − Y_s*X_i) / (Y_i − Y_s)
16:        Y_m = f(X_m)
17:        Error = tol *2
18:        Cont = 1 //Because an X_m was already compute
19:        while Y_m ≠ 0 & error > tol & cont ≤ iter do
20:            if Y_m * Y_i < 0 then
21:                X_s = X_m
22:                Y_s = Y_m
23:            else
24:                X_i = X_n
25:                Y_i = Y_m
26:            end if
27:            Aux = X_m
28:            X_m = (X_i + X_s) / 2
29:            err = ||X_m − Aux||
30:            Cont = Cont + 1
31:        end while
32:        if Y_m = 0 then
33:            "X_m is a root"
34:        else if error < tol then
35:            X_m is a root with an error of error
36:        else if Cont > iter then
37:            "We have ran out of iterations"
38:        end if
39:    end if
40: end procedure
```