G12 Computer Science Wed Oct 2 2019

Assignment 2

Due Date: Mo. Oct. 7th 2019

This assignment develops on the topic discussed in the last class on the algorithms of finding roots of a function and finding its extremal points. See the notebook here (https://nbviewer.jupyter.org/url/evermeet.cx/~user055/Dragon/Lessons/CompSci/G12/G12-CS-20190930_111000.ipynb?flush_cache=true)

Problem statement

Write a function called `newton(f,df_dx,xo,epsilon,alpha)` such that

1. it returns ***a triplet*** in the following order:
   A. the x^* value such that f is an extremum, i.e., either a minimum or a maximum, at x^*.
   B. the minimum value of f, $f(x^*)$, i.e., the value of the function at x^*
   C. the value of its derivative at that same point, i.e., $df_dx(x^*)$
2. The value of x^* is determined with a precision of ϵ. Make sure to declare the function such that it is $\epsilon = 0.1$ **by default**.
3. The learning rate is $\alpha = 0.05$ **by default**.
4. The starting point for the search is x_o
5. `df_dx` is another function provided by the user and corresponds to the derivative function of f with respect to x.
6. Consider the example worked out for our last day's homework $f(x) = 5 \cdot (x-7)^2 + 14$. What would happen if $x_o = 7$? Make sure your code try its best to detect such a case.

Explicit example

Work out and provide the code for an explicit example. That is, include a cell where you define a function f, its derivative $f'(x)$ and then use your code for `newton` to obtain the extremum of f.

Additional requirements

Make sure to

1. use exactly the same notation as in this problem statement.
2. make your code as ***modular*** as possible by using different functions that tackle separate little problems, but combined give you the solution you want.
3. strictly follow and write down the four steps for writing algorithms.
4. make sure you include your notes obtained during the intuition gathering step, as well as the logic of the

program and its pseudo_code as a multiline comment at the beginning of the function definitioin. Do one such comment for each and every function you define

Sample input/outputs

In all these following examples, it is $f(x) = 5 \cdot (x-7)^2 + 14$ and $df \equiv$ df_dx $\equiv f'(x) = 10 \cdot (x-7)$.

1. `newton(f,df,7)`:
   (7,14.0,0.0)
2. `newton(f,df,9)`:
   (7.0625, 14.01953125, 0.625)
3. `newton(f,df,9,epsilon=0.01)`:
   (7.0078125, 14.00030517578125, 0.078125)
4. `newton(f,df,9,epsilon=0.01,alpha=0.1)`:
   (7.0, 14.0, 0.0)
5. `newton(f,df,9,epsilon=0.01,alpha=1)`:
   OverflowError: (34, 'Result too large')

Solution