PHYS-GA 2000 Computational Physics PS5

Kaixuan Zhou

November 11, 2023

1 Problem 1

In problem 1, we implement Brent's 1D minimization method and verified that with the function:

$$y = (x - 0.3)^2 e^x \tag{1}$$

I created a function with the golden section search. This method is a technique to find extremum for a function in a setting interval. We set up the starting point and step, by iterations we approaches the extremum in this problem. Here I will refer the psuedocode given in the hints to describe the optimization process:

def Brent(f, a, b, delta = 1e-7):

if $|f(a)| < |f(b)|$:

swap a and b

c = a

flag = True denotes whether bisection (you'll want to replace this with golden mean search!) is used

while $|a - b| < delta$:

s = $\mathrm{s}_quad_interp(a, b, c)$

if any of the following conditions are true:

Condition 1: $s \geq b$ s is outside of bracket: then resort to bisection

Condition 2: flag = True and $|s - b| \geq |b - c|$ —b-c— is size of previous step, bisection was previous step

Condition 3: flag = False and $|s - b| \geq |b - c|$ previous step not bisection and step before the previous step is larger smaller than the current step

then resort to bisection minimization: you'll want to replace this with golden mean search!

s = (a+b)/2

flag = True we used bisection

else: keep the Brent estimate

flag = False we used Brent

now iterate

d = c

c = b

a = s

if —f(a)—¡—f(b)—: again swap a and b if needed
swap a and b
return b return the minimum

That is the psuedocode of the progress. We compare this Brent method written by me and the scipy package, here's the result:

```
In [33]: print(optimized)
         print(minimized)

0.2999999225139618
0.29999857467940383
```

```
In [34]: print('The difference between my optimization with scipy package is',abs(minimized -optimized))

The difference between my optimization with scipy package is 1.3478345579631679e-06
```

We see the error is small. It is a valid method.

2 Problem 2

In this problem, we extract data from the file firstly, then model the probability with given function:

$$p(x) = \frac{1}{1 + exp[-(\beta_0 + \beta_1 x)]} \tag{2}$$

Then we calculates the likelihood function:

$$\mathcal{L} = log f(x|\theta) = \prod_{i=1}^{n} \rho(x_i|\theta) \tag{3}$$

And we use the minimize function from scipy. Following is the error and covariance result:

```
Optimal parameters and error:
        p:  [-5.62023318  0.10956342]
        dp:  [0.07135651 0.00426131]
Covariance matrix of optimal parameters:
        C:  [[ 5.09175175e-03 -1.17826017e-04]
 [-1.17826017e-04  1.81587401e-05]]
```

Figure 1: It is the figure of Optimal parameters and error and Covariance matrix of optimal parameters

and the figure of the distribution:
This all work well for the likelihood function.
GitHub account:luminousxuan

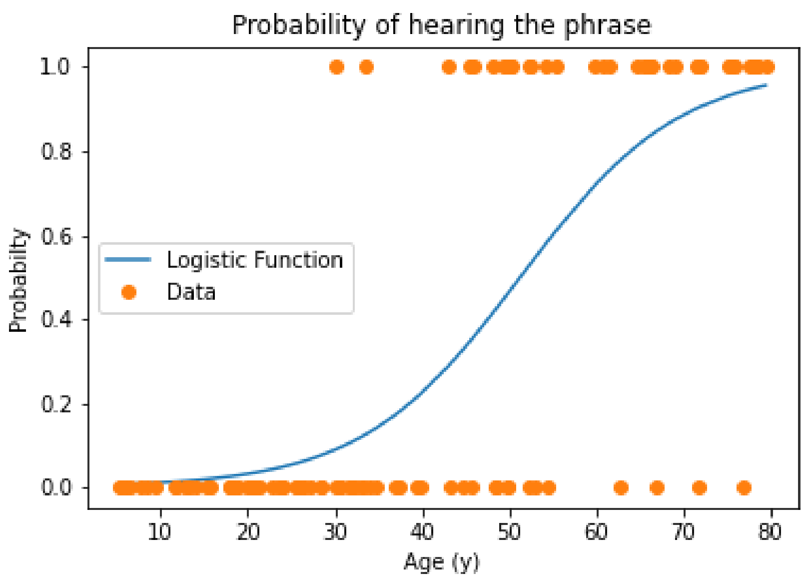

Figure 2: It is the figure of the distribution over the age(year).