Numerical Methods

Math 3338 – Spring 2022

Worksheet 10

Partial Derivatives and Gradient Descent

1 Reading

CP	5.11
NMEP	3.1, 3.2

Table 1: Sections Covered

2 Partial Derivatives

You basically just think coordinate wise. If $f_x(a,b) = \frac{\partial}{\partial x} f(x,y)|_{(a,b)}$, then

$$f_x(a,b) = \lim_{h \to 0} \frac{f(a+h,b) - f(a,b)}{h}$$

This should be fairly straightforward.

You can also do central differences, which is far more powerful.

$$f_x(a,b) = \lim_{h \to 0} \frac{f(a+h,b) - f(a-h,b)}{2h}$$

3 Gradient Descent

Once again, you need to think about this coordinate wise. The process is going to be exactly same as optimizing a single value function. However, you should be vectorizing the operations. Each coordinate is can be found using

$$x_i = x_i - \alpha f_{x_i}(x).$$

This is going to test your understanding of vectorizing.

Numerical Methods

Math 3338 – Spring 2022

Homework 10 (Due: Thursday, February 17)

Problem 1 (1 pt) Let $f : \mathbb{R}^n \to R$ we'll represent $f(x)$ as a function that takes in an array x. Write a function `partial(f,x,i,h=1e-8)` that will calculate the partial derivative, $f_i(x)$ at the point x.

Problem 2 (1 pt) Create a function `gradient(f,x,h=1e-8)` that returns a column vector so that the i'th coordinate is $f_i(x)$.

Problem 3 (1 pt) Create a function `gradient_descent(f,x,alpha=.1,max_iter=50,h=1e-8)` that approximates the minimum of f using the starting point x with learning rate α and a maximum number of iterations.

Problem 4 (1 pt) Use your function to approximate the minimum of $f(x, y) = x^2 + y^2$. You should know what the minimum is so you can verify if you got it right. Call your function `problem_4` (instead of f). Try several different starting values and learning rates.

Problem 5 (1 pt) Use your function to approximate the minimum of $f(x_1, x_2, x_3, x_4, x_5) = \sum x_i^2$. You should know what the minimum is so you can verify if you got it right. Call your function `problem_5` (instead of f, although if you do Problem 4 "correctly" you can use the same function, but still have problem_5).

Problem 6 (1 pt) Use your function to approximate the minimum of $f(x_1, x_2, x_3, x_4, x_5) = \sum x_i^2 + 2x_i$. You should know what the minimum is so you can verify if you got it right. Call your function `problem_6` (instead of f)

Problem 7 (1 pt) Gradient descent will only find a single minimum, so there could be an issue with local minimums. This turns out to not be an issue for very high dimensional data. Can you think of a justification why this would be true? Submit your answer as a PDF.

Problem 8 (1 pt) This is an optional problem. Explore how to plot 3d functions in Python. Plot a function, use gradient descent to find the minimum, and plot the point on the graph. Hopefully the minimum you find coincides with the minimum.