Problem Set 4: Gradient Descent

CMSC 422, Fall 2017

Assigned 9/18, Due 9/25 at 2 p.m.

1. Suppose we have the loss function $f(x_1, x_2) = x_1^2 + x_2^2$.

   (a) What is the global minimum of f?

   (b) What is the gradient, ∇f?

   (c) Suppose we use the following gradient descent iteration:

   $$(x_1^{k+1}, x_2^{k+1}) = (x_1^k, x_2^k) - \eta \nabla f(x_1^k, x_2^k)$$

   (x_1^k, x_2^k) indicates the value of (x_1, x_2) after we've repeated this iteration k times. The k's are not exponents. η is a small constant that indicates how quickly we move in the direction opposite

the gradient. This is called the *learning rate*. We initialize the iteration with: $(x_1^0, x_2^0) = (3, \frac{1}{4})$, and use $\eta = .2$. What is the next step in the iteration (ie., what do we get for (x_1^1, x_2^1))?.

(d) What values do we get for (x_1^5, x_2^5)? For (x_1^{10}, x_2^{10})? Hint: you might want to write a little program to compute this.

2. Suppose we have the loss function $f(x_1, x_2) = \frac{x_1^2}{16} + 4x_2^2$.

   (a) What is the global minimum of f?

   (b) What is the gradient, ∇f?

   (c) Suppose we use the same iteration as before, with the same initialization and the same η. What

do we get for (x_1^1, x_2^1)? For (x_1^5, x_2^5)? For (x_1^{10}, x_2^{10})?

(d) In which of the two problems does gradient descent seem to be converging more rapidly to the global minimum? Can you explain the difference in the convergence rate of the two problems? (Your answer will be graded based on how precisely you can explain what is going on).

3. Suppose you were to increase the learning rate, η. Would this make convergence faster or slower? Would the answer be the same for problems 1 and 2? You can run some experiments to decide on an

answer, but give a well thought-out explanation for what you see.

4. One way to evaluate the gradient is by differentiating the function (as you did in Problem 1). You would then write a function to return the components of the gradient vector and use it in the gradient descent algorithm. For many functions it is not possible to differentiate the loss function analytically. In this case we can use the method of *finite differences*. Here we take the definition of the derivative and evaluate an approximation of each component of the d dimensional gradient

$$\frac{\partial f}{\partial x_i} = \frac{f\left(x_1, x_2, \ldots, x_i + h, \ldots, x_d\right) - f\left(x_1, x_2, \ldots, x_i, \ldots, x_d\right)}{h}$$

   (a) Write a function to compute the loss function in Problem 1 for a given value of (x_1, x_2).

   (b) Write a function to compute the gradient of the loss function using finite-differences, that uses a calls to the existing loss function. You provide the function with (x_1, x_2) and h.

   (c) Using $h = 0.1$ and $h = 0.01$ compute the gradients at any 3 values of (x_1, x_2) and compare

with the analytical values.

(d) How many function evaluations are needed to evaluate the gradient using this method in d dimensions?