BIOE 210, Spring 2022

Homework 8

Due Monday, 3/21/2022 by 5:00pm.
Upload your answers to Gradescope. If submitting a single PDF, you must mark the location of all answers.

Part I (8 points)

We want to find a root for the nonlinear system

$$\mathbf{g}(\mathbf{x}) = \begin{pmatrix} x_2 \cos x_1 \\ 2x_2^2 - 1 \end{pmatrix}$$

We will use Newton's method to find a vector $\mathbf{x}$ such that $\mathbf{g}(\mathbf{x}) = \mathbf{0}$ starting from the initial guess $\mathbf{x}^{(0)} = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$.

1. Write the Jacobian matrix $\mathbf{J}(\mathbf{x})$ for the system of equations.
2. Show that $\mathbf{x}^{(0)}$ is not already a root by verifying that $\mathbf{g}(\mathbf{x}^{(0)}) \neq \mathbf{0}$.
3. Using Newton's method, find a new guess $\mathbf{x}^{(1)}$ using $\mathbf{x}^{(0)}$. Calculate $\mathbf{g}(\mathbf{x}^{(1)})$. You are welcome to use MATLAB or a calculator to invert $\mathbf{J}(\mathbf{x})$ and perform any matrix multiplication.
4. Perform two more iterations to find $\mathbf{x}^{(2)}$ and $\mathbf{x}^{(3)}$. Show that the values $\mathbf{g}(\mathbf{x}^{(2)})$ and $\mathbf{g}(\mathbf{x}^{(3)})$ approach $\mathbf{0}$.

Part II (12 points)

You goal is to solve the problem

$$\min_{\mathbf{x}} f(\mathbf{x}) = (x_1 - 2)^2 + (x_2 + 3)^2 + (x_3 - x_1)^2$$

using gradient descent with $\mathbf{x}^{(0)} = \mathbf{0}$ and $\alpha = 0.01$.

1. Define functions for the function f and its gradient $\mathbf{g}$. One options is to use anonymous function in MATLAB. For example, f can be defined

   ```
   f = @(x) (x(1)-2)^2 + (x(2)+3)^2 + (x(3)-x(1))^2;
   ```

   allowing you to evaluate `f(x)` for a vector `x`. The gradient $\mathbf{g}$ can be defined similarly, returning a vector instead of a single value.
2. You will perform 1000 iterations of gradient descent and store the iterates $\mathbf{x}^{(k)}$ and function values $y^{(k)} = f(\mathbf{x}^{(k)}$ for each iteration. Use the `zeros` function to initialize a 1000×3 matrix `X` for the iterates and a 1000×1 matrix `y` to store the function values.
3. Write a for loop to calculate the new iterate, storing the iterate and the function value at each iteration.
4. Plot the function values $y^{(k)}$ vs. k.

5. Plot all three entries of $\mathbf{x}$ (x_1, x_2, and x_3) vs. k on the same plot. You can use `plot(X)` to quickly plot the columns of a matrix.

6. Re-run your code using step sizes $\alpha = 0.1$ and $\alpha = 0.001$. Show the plots of $\mathbf{x}$ vs. k for each value of α. (You do not need to include new plots for y.) Discuss why changing α has the effect it does.