Mohammad Motamed
MATH 505, Fall 2022 – HW # **3**, Posted: Oct. 28, 2022

Upload your report as a PDF-file on Canvas before Sunday November 20, 11:59pm.

A problem of function approximation: Consider a traget function

$$u(x) = (\sin(4\,x) + \cos(6\,x))\,(1 + x^2), \qquad x \in [-1, 1].$$

We use its closed form only to generate data, and after collecting data we assume that the target function is not given. Our goal is to approximate the target function using the collected data and some of the optimization techniques we have learned in class.

Data collection: Let the number of data be n. We uniformy discretize the interval $[-1, 1]$ into n points:

$$x_i = -1 + h(i-1), \qquad i = 1, 2, \ldots, n, \qquad h := \frac{2}{n-1}.$$

Then use the closed form of the target function and generate n output points

$$y_i = u(x_i), \qquad i = 1, 2, \ldots, n.$$

This will give us a set of n input-output data points $\{x_i, y_i\}_{i=1}^n$.

Mathematical model: We will use a simple mathematical model with two parameters

$$v(x; \boldsymbol{\theta}) = (\sin(\theta_1\,x) + \cos(\theta_2\,x))\,(1 + x^2), \qquad x \in [-1, 1], \qquad \boldsymbol{\theta} = (\theta_1, \theta_2).$$

Optimization problem: Our goal is to use the collected data and solve the following minimization problem to obtain the model parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$:

$$\min_{\boldsymbol{\theta}} f(\boldsymbol{\theta}),$$

where the objective function that we want to minimize reads

$$f(\boldsymbol{\theta}) := \frac{1}{n} \sum_{i=1}^{n} \frac{1}{2} (y_i - v(x_i; \boldsymbol{\theta}))^2$$

Your tasks:

1. (50 points) Let n be small (not too small), e.g. $n = 101$. Implement a code that uses Gradient Descent (GD) algorithm to approximate the minimizer:

   $$\boldsymbol{\theta}^{(k+1)} = \boldsymbol{\theta}^{(k)} - \lambda_k \nabla_{\boldsymbol{\theta}} f(\boldsymbol{\theta}^{(k)}).$$

   You may need to try different learning rates λ_k. Note that the learning rate should not be too small or too large. For each learning rate, monitor $f(\boldsymbol{\theta}^{(k)})$ versus iterations $k = 1, 2, \ldots$. If the objective function does not decay, it means that your learning rate is not good. Try your code with different number of data points n. After you obtain a good approximation of the parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$, plot the approximant $v(x; \boldsymbol{\theta})$ together with the target function $u(x)$ for $x \in [-1, 1]$ in the same plot. Make sure your plot has labels and legend. Also include plots showing the decay of the objective function versus k. Discuss your observations and compare with the theoretical rate of convergence (or decay in objective function) that you expect.

2. (50 points) Repeat with Stochastic Gradient Descent (SGD). You should notice that here you can more efficiently handle larger number of data points (larger n) compared to when you use GD. Why is so?

3. (Optional with 40 bonus points) Repeat with Accelerated Gradient Descent (AGD). Specifically discuss the convergence rate and compare with both GD and SGD. What are the major benefits and drawbacks of AGD compared to GD and SGD.