Assignment 5 Supervised Learning
Due Date: 1400/Bahman/05

Foundations of Data Science
Supervisor Dr. SaeedReza Kheradpisheh
Teaching Assistants Hesam Damghanian, Ali Rahimi

Shahid **B**eheshti **U**niversity

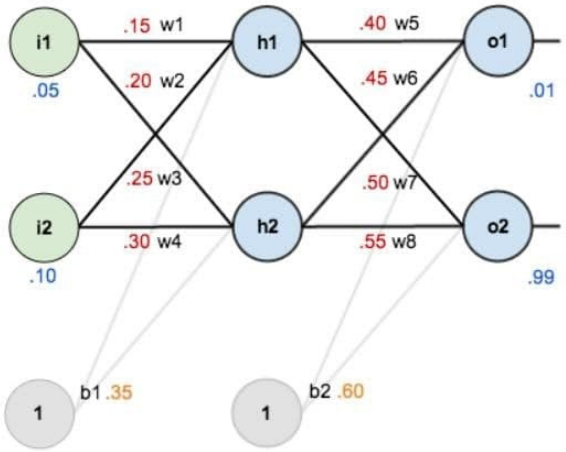
Optimization (3/20)

Theory

1. Discuss these questions in-length:
    a. Why do we usually consider cost function as a negative of log-likelihood?
    b. Explain L1 and L2 regularization; compare them to each other.
    c. What is the effect of the momentum in a learning method?
2. Learning in a Neural network is done by updating the weights. “Backward propagation of errors” known as backpropagation is the common method to do the updating efficiently. Now consider the following neural network with Sigmoid as neurons’ activation function and “Mean Squared Error” as the cost function.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y_i})^2$$

    a. Calculate the network error after one step of feed-forward.
    b. Calculate one step of backpropagation with a learning rate equal to 0.3.



3. Suppose batch gradient descent in a deep network is taking excessively long to find a value of the parameters that achieves a small value for the cost function J(W[1],b[1],...,W[L],b[L]). Which of the following techniques could help find parameter values that attain a small value forJ? (Check all that apply)
    - ☐ Try better random initialization for the weights
    - ☐ Try mini-batch gradient descent
    - ☐ Try using Adam
    - ☐ Try initializing all the weights to zero
    - ☐ Try tuning the learning rate α

Implementation

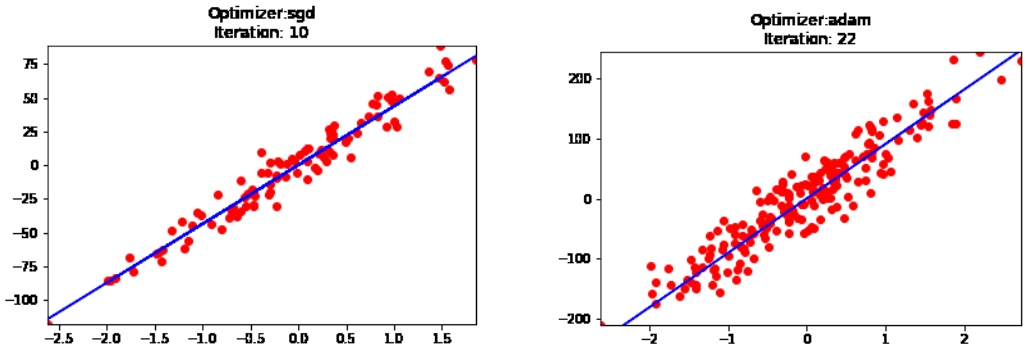
In the last assignment, you implemented the gradient descent algorithm for optimizing linear regression, this time you are asked to implement the following optimization algorithms in python for the same model as previous.

1. SGD
2. SGD with momentum
3. AdaGrad
4. RMSProp
5. Adam

Tasks:

1. Use the template code provided, only modify the TODO parts and for reproducibility reasons, do not change the random seed.
2. You should find and **report** the right hyperparameters for these algorithms to converge, i.e., learning rate and **explain** each hyperparameter for each algorithm (1-2 lines).
3. Discuss and compare each algorithm's pros and cons (2-3 lines each).

You should expect outputs like these:



Bonus:

1. Visualize the gradient descent process (Check out the template code for guides).
2. Implement new optimization methods to solve the toy regression problem, from recent papers, and test your model with 2d inputs; in addition, visualize the regression process on 3d plots.

See here for more information about optimization algorithms.