CS4210 Fall 2024 Project Assignment 1

Total points: 100

Due date: Friday, 10/04/2024

Purposes:

1. Warm up your Python programming skills.
2. Understand the key concepts of machine learning.
3. Get familiar with linear regression and use Scikit-learn library.
4. Master the training loop based on gradient descent optimization
    - A figure about the training loop is shown on page 2 of this document.

Task Description:

In this assignment, you will use linear regression to study diabetes data, which has 10 baseline variables (age, sex, body mass index, average blood pressure, and six blood serum measurements) were obtained for each of 442 diabetes patients, and the response of interest (a quantitative measure of disease progression one year after baseline).

Please implement the following tasks,

- **(10 pts) Task 1: Prepare 3 datasets:** training dataset, validation dataset, and testing dataset.
    - Note that a good starting point on the ratio between training dataset, validation dataset, and testing dataset is 60% training data, 20% validation data, and 20% testing data.
- **(20 pts) Task 2: Use** LinearRegression() from Scikit-learn.
    - **print** the resulting value of the bias and the weights
    - **print** the trained model's errors on training dataset, validation dataset, and testing dataset, respectively.
- **(30 pts) Task 3: Implement** basic gradient descent to perform linear regression. Please tune the parameters to get close to the accuracy of the linear regression model from scikit-learn.
    - **print** the resulting value of the bias and the weights
    - **use** matplotlib to plot the learning curves showing how training error and validation errors along iterations.
    - **print** the trained model's errors on training dataset, validation dataset, and testing dataset, respectively.
- **(40 pts) Task 4: Implement** stochastic gradient descent method to perform linear regression. Please tune the parameters to get close to the accuracy of the linear regression model from scikit-learn.
    - **print** the resulting value of the bias and the weights
    - **use** matplotlib to plot the learning curves showing how training error and validation errors along batches.
    - **print** the trained model's errors on training dataset, validation dataset, and testing dataset, respectively.

In each of the above tasks, please ensure to use the following loss function to compute the errors,

$$\ell(w) = \frac{1}{2N}\sum_{i=1}^{N}[t^{(i)} - y(x^{(i)})]^2$$

Note that please use **matrix/vector operations** to evaluate the above loss function, rather than a *for* loop.

What to Submit (on Canvas)?

1. A iPython notebook that contains your codes. A template can be found in the zipped folder of this assignment. Notes:
    1. non-executable programs result in a grade of zero.
    2. regular Python program file with ".py" is not acceptable.
    3. properly comment your programs.
    4. name your file using the following format: "*yourLastName_yourFirstName_assignment1.ipynb*" and submit it on Canvas

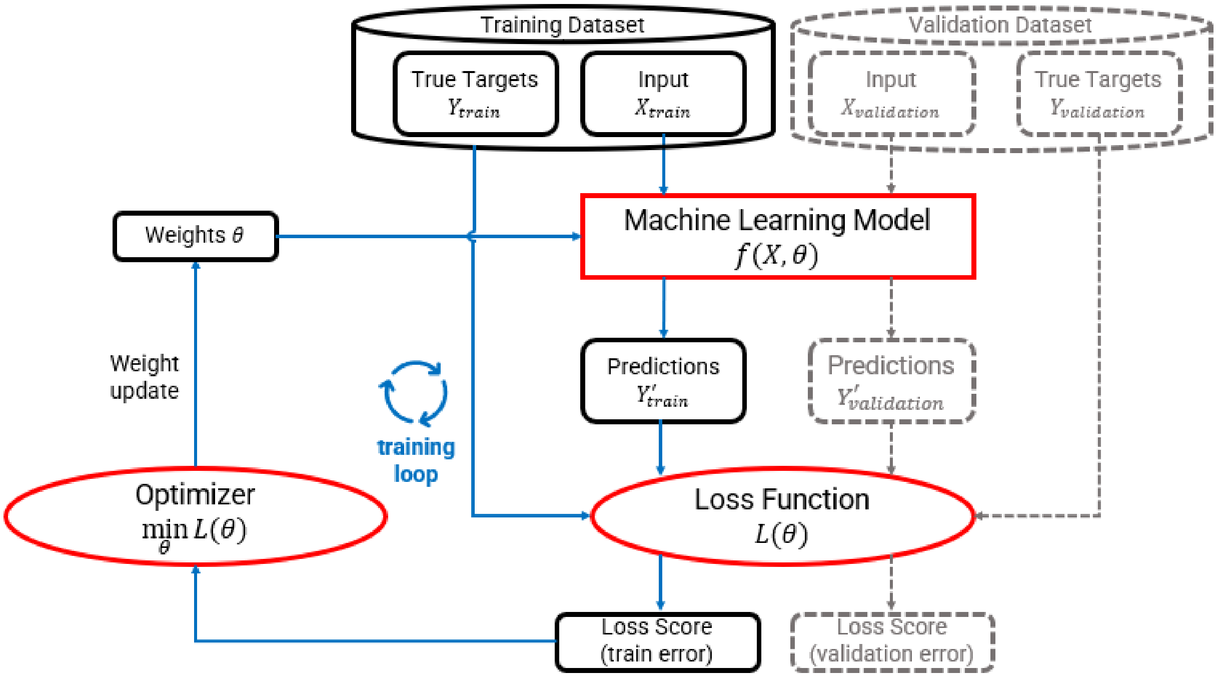

Fig.1: The training loop in machine learning