Fall 2024

COSC 522 – Homework #2 (Due 9/19)

Problem 1: (25 pts.) On Gradient Descent (GD). Find the global minimum of f(t) = 50 * sin(t) + t^2 over -10<=t<=10. This problem intends to give you a hands-on experience on how gradient descent works and how it can get trapped at the local minima.

a) (1) Plot this function. Visualize the multiple local minima and the global minimum.

b) (25) Implement gradient descent in Python to find the local minimum.

   i) (10) Pick a starting point at t=7. What's the minimum? Show the convergence path. Experimenting with different learning rate.

   ii) (10) Pick a starting point at t=1. What's the minimum? Show the convergence path. Experimenting with different learning rate.

   iii) (4) Comment on the results from the above experiments

Problem 2: (35 pts) On Perceptron.

a) (15 pts) This part should be done on pencil and paper. Use the truth table for OR as the input, assuming the initial values of the weights and bias are w_1=0.1, w_2=0.2, w_0=0.5. Show details for the first epoch using both online learning and batch learning.

b) (15 pts) Use Perceptron to implement the OR logic. Show output from each iteration (that is, the two inputs, the targeted output, and the Perceptron output) with the maximum number of iterations being 10.

c) (5 pts) Use Perceptron to implement the XOR logic. Show output from each iteration (that is, the two inputs, the targeted output, and the Perceptron output) with the maximum number of iterations being 10.

Problem 3: (40 pts.) Using SVM for XOR. The four training samples are [[-1, -1], [-1, 1], [1, -1], [1, 1]], with the corresponding label being [-1, 1, 1, -1]. Obviously, the decision boundary would not be linear. We use the kernel trick to solve this problem. Kernel tricks use a kernel function to project the data from the original space (in this case, 2-d space) to a higher-dimensional space where a linear boundary can be found to perfectly separate the samples.

(a) (10 pts) Suppose the kernel function is a 2nd-degree polynomial, i.e., $K(\mathbf{x}, \mathbf{y}) = (x_1y_1 + x_2y_2 + C)^2$, where $\mathbf{x} = [x_1\ x_2]^T$, $\mathbf{y} = [y_1\ y_2]^T$. Derive the basis functions $\phi(x)$, that is, $K(\mathbf{x}, \mathbf{y}) = \phi^T(\mathbf{x}).\phi(\mathbf{y})$

(b) (10 pts) Using the derived basis function, what is the higher-dimensional space that the 2-d sample should be mapped to? Provide the higher-dimensional counterpart to the four 2-d samples.

(c) (20 pts) Apply Perceptron on the higher-dimensional samples and see if you can find a linear decision boundary using the higher-dimensional samples. Output the weights learned. Project the learned hyperplane onto 2-D space and show the decision boundary.