AI1104: Programming for AI, Spring/Summer 2021 (56)

Indian Institute of Technology Hyderabad
HW 1, Assigned: Saturday 17.07.2021.
Due: Friday 23.07.2021 at 11:59 pm.

The Force is with you, young Programmer. You are not a Jedi yet!
Don't be drawn to the Dark Side of online solutions!

1. Implement a linear regressor using: (50)

   (a) $\hat{y} = f(x;\theta) = \beta_0 + \beta_1 x$,

   (b) $\hat{y} = f(x;\theta) = \beta_0 + \beta_1 x + \ldots + \beta_m x^m$.

   Carry out the following steps:

   - Use the following code snippet to generate training and test samples.

     ```
     import numpy as np
     # Number of training/test samples
     N = 10
     # Generate equispaced floats in the interval [0, 2π]
     x = np.linspace(0, 2*np.pi, N)
     # Generate noise
     mean = 0
     std = 0.05
     # Generate some numbers from the sine function
     y = np.sin(x)
     # Add noise
     y += np.random.normal(mean, std, N)
     ```

   - Assuming MSE loss, train each of the above models with N training samples. Choose $N = 10, 100, 1000$ and train your model for each case.
   - For the polynomial model (in x), experiment with $m = 2, 3, 4, 5$.
   - Report your model's performance using T test samples also generated using the above code. Specifically, report performance for *each* value of N chosen for training the models, and for each value of m for the polynomial function (in x).
     - Performance must be reported *qualitatively* by plotting the following pairs of points: $\{(x_1^{\text{test}}, \hat{y}_1^{\text{test}}), (x_2^{\text{test}}, \hat{y}_2^{\text{test}}), \ldots, (x_T^{\text{test}}, \hat{y}_T^{\text{test}})\}$. Clearly label your plots.
     - Performance must be reported *quantitatively* in terms of the MSE between the predicted and ground truth labels.
     - Print any important observations (like overfitting, underfitting, matrix ill-conditioning etc.) from your experiments.

2. Update the demo code of the K-means clustering algorithm to accept the number of clusters K and the input dimension p as user inputs. In other words, generalize the code to work with user-specified values for K and p. You can use the same process for data generation as in the demo code. (50)