Regression

Re-submit Assignment

Due Oct 1 by 11:59pm **Points** 100 **Submitting** a file upload
File Types pdf and html

You are tasked to develop a regression model that best fits the data below. You will turn in a Jupyter notebook in PDF format that describes your method, and provides and demonstrates your code.

All data sets are curated from the UCI Machine Learning repository.

- Airfoil Self-Noise: **https://archive.ics.uci.edu/ml/datasets/Airfoil+Self-Noise (https://archive.ics.uci.edu/ml/datasets/Airfoil+Self-Noise)**
- Yacht Hydrodynamics: **https://archive.ics.uci.edu/ml/datasets/Yacht+Hydrodynamics (https://archive.ics.uci.edu/ml/datasets/Yacht+Hydrodynamics)**
- Concrete Slump: **https://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test (https://archive.ics.uci.edu/ml/datasets/Concrete+Slump+Test)** (note that this data set has 3 output variables)

1) Develop a function with the following first line:

def my_regression(trainX, testX, noutputs)

The input variables are:

- trainX - an [ntrain x (nfeature + noutputs)] array that contains the features in the first 'nfeature' columns and the outputs in the last 'noutput' columns
- testX - an [ntest x nfeature] array of test data for which the predictions are made
- noutputs - the number of output columns in trainX

The output should be an [ntest x noutputs] array, which contains the prediction values for the testX data.

Your my_regression code should do some kind of cross-validation to determine the right model for the training data, e.g., linear vs. polynomial vs. radial basis functions. Then this model is applied to the test data to make a prediction.

Rules:

- Your code should not use any regression library, but can use NumPy as this will be useful for array handling.
- No other inputs are allowed.
- You may not use the testX data to train your model. Notice that outputs are not passed in for testing data.
- Your code can perform data scaling (scaling features to the interval [0,1] or z-score scaling), can use any basis functions you want, cross-validation on the training data, regularization, and model selection.

Present your findings for each data set as a Jupyter notebook. Submit as a PDF. Present your squared error for several cross-validation fold of each data set.

Rubric:

- (30 points) - The regression works and you produce some kind of result.
- (30 points) - Are the results reasonable? Do you get squared errors that are close to mine? (I'll release linear basis function sqaured errors)

- (20 points) - Can you do better than linear basis function prediction?
- (20 points) - Is your code commented? Is your analysis strong? Do your plots look nice? Does your overall document look nice?