

Homework 1

Problem 1 [5pts]:

Suppose there are 4 independent variables. And 3 operations, i.e., addition, multiplication, and power, are allowed in connecting them into an expression. How many different expression can we have? For example, let the 4 variables be A, B, C, and D. Then $(A+B) * C ^D$ is one expression while $(A^B) * C+D$ is another. The order of variables matters, e.g., $A+B$ and $B+A$ are two different expressions.

The purpose of this problem is to understand how many different parametric equations can exist among a set of variables, and thus using traditional scientific discovery way to model the relationship between high-dimensional variables is very challenging.

Problem 2 [5pts]:

In the Jupyter Notebook example for Unit 1, there is a demo that the score of a neural network changes along with the maximal number of iterations (i.e., the `max_iter` argument in the function `test_NN`). Now, let's visualize the change by writing a plot function `learning_curve` that takes 3 arguments: a. X, 1-D numpy array, e.g., `[1,2,3]` not `[[1],[2], [3]]` b. y, 1-D numpy array c. filename, a string such that X and y are the input and corresponding output for a supervised learning task and filename specifies the PNG file to save the plot. The X and y will be used to train a neural network for multiple times, with different maximal numbers of iterations. Scan the maximal number of iterations from 50 to 2,000 with a step of 50 while logging the corresponding score sequentially (the return of `test_NN`). Make a line plot between the maximal number of iterations and the score of the NN, and save as a PNG file. Also, return the two vectors (as 1-D numpy array or 1-D list) used for plotting (first the sequence of maximal numbers of iterations, and then the scores).

Submit one .py file with the answer to Problem 1 as a comment line at the top of the file. Then include the function definition to `learning_curve` below. Do NOT include other code in your submission.