Statistical Theories for Brain and Parallel Computing

Assignment No.3

17M18819

ZHU LICHENG

## Description of Assignment

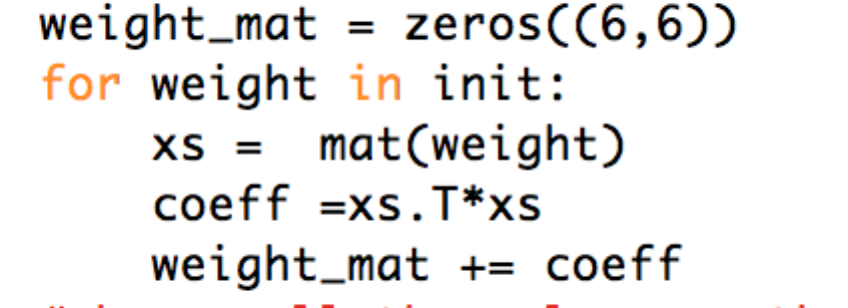
In assignment 3, I need to construct a suitable RNN to solve a linear equation problem. The RNN should be set to have the minimum energy at the solution.

## Solution

***Deterministic and Binary Model***

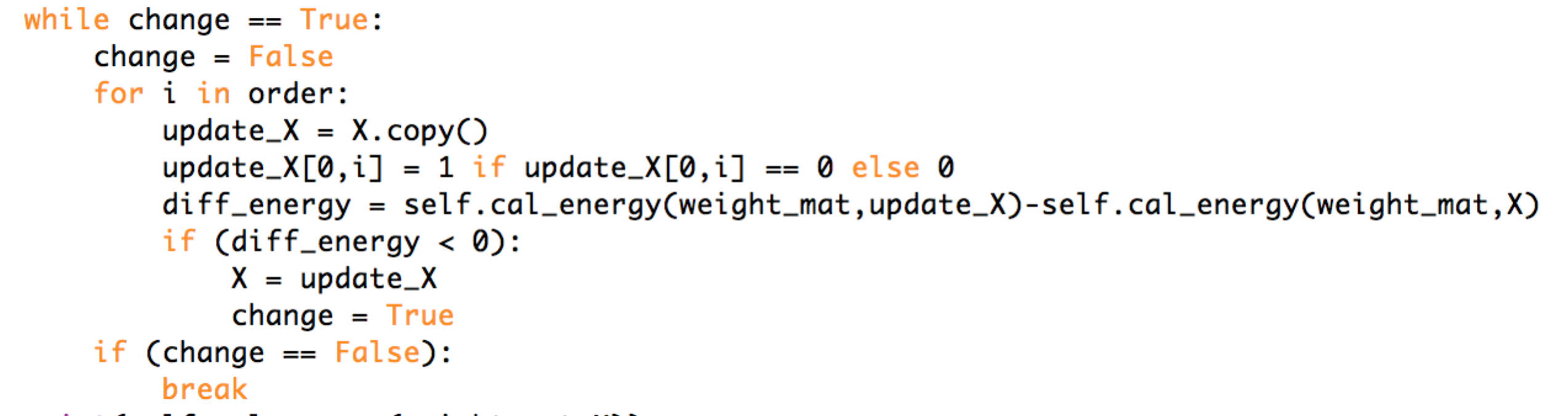
I used python to implement this task. I created one class called **ass3\_determin.py** where we can call the function **determin\_rec(init)** where init is all the coefficients of the linear equation. In this case, I focus on one linear equation below.

First of all, I would like to calculate out all the coefficient of this linear equation as follows:

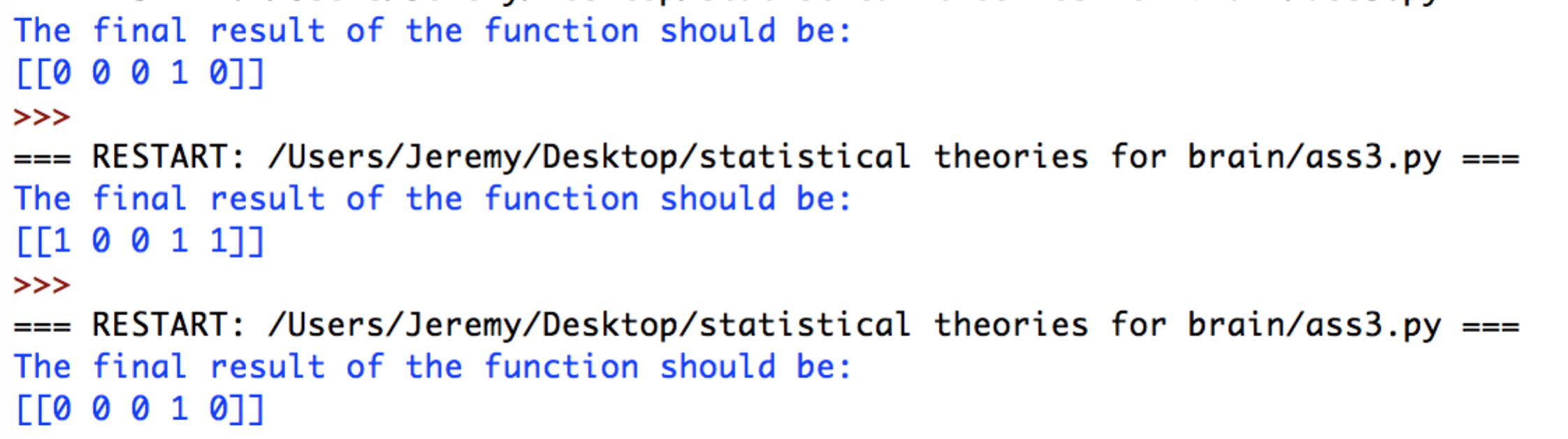


After all the weight of connection and thresholds have been calculated out , I can call function **cal\_energy** to calculate the energy of current status.

Then I initialized the initial solution of X as [0,0,0,0,0] and randomly choose one of five neurons for updating at a time:



The updating process will stop when the energy reaches a global minimum or a local minimum. Actually, in this test case, I will accidentally run into some local minimum like: [0,0,0,1,0] when the real solution is [1,0,0,1,1] because [0,0,0,1,0] is a local minimum with energy equal to -15 while energy of [1,0,0,1,1] is a global minimum which is equal to -19:



Therefore, we may use probabilistic model to overcome the disadvantage of deterministic model later.

(The following parameters can be changed by hand when calling function **determin\_rec()**)

*determin\_rec([[1,-2,1,2,-1,-2],[-1,1,1,-1,-1,3],[2,1,-1,-2,-2,2],[1,2,-1,-1,-1,1],[-1,-1,1,1,1,-1]])*