SchoolNova Computer Science 202

Homework 14 (due 1/23/2021)

Save your code as lastname_homework14.py and submit on Google Classroom

In this homework, your goal is to figure out a secret pattern based on a list of numbers, in which each number depends on the previous two numbers. We can accomplish this goal using machine learning.

Task 1

Download 'puzzle.csv' file to your computer and load in Python as a list. For example, you can save the file to a specific folder and then use numpy's loadtxt:

```
import numpy as np
import os
os.chdir('C:\\python')
X = list(np.loadtxt("puzzle.csv")) # after the file is saved to the folder above
```

Using list comprehension, change the type of each number in the list to integer.

Task 2

There are 1000 numbers in X. The numbers follow a particular (secret!) pattern. Each number is related to the previous two numbers (notice that the relationship is stochastic as there is a some 'noise' (randomness) and some relationships are [statistically] stronger than others).

For example, the first 10 numbers are [0, 0, 2, 0, 0, 3, 3, 1, 3, 3]. In this case, 2 follows (0, 0), 0 follows (0, 2), 0 follows (2, 0), 3 follows (0, 0), and so on.

Generate the dataset appropriate for the machine learning model (similar to what we did in class). In the data, a number *i* is a function of the numbers *i-2* and *i-1*. For the first 10 numbers your data would look like this:

```
[0, 0, 2],
[0, 2, 0],
[2, 0, 0],
[0, 0, 3],
[0, 3, 3],
[3, 3, 1],
[3, 1, 3],
[1, 3, 3]
```

If you cannot complete this Task, please, let me know and I will send you the solution since your next two Tasks depend on Task 2 successful completion.

Task 3

Split your data into X_train, X_test, y_train, y_test, devoting 70% of data to the training set (and 30% to the test set).

Using the sklearn's SVC classifier and the training sets that you just created, fit the model into the data.

Generate the machine learning prediction, using X_test.

Print the classification report and plot the confusion matrix (use classwork notes if you need help).

Task 4*

Generate a dictionary that shows the solution to the puzzle. It may look like this (for example):

{(0, 0): 1, (0, 1): 0, {0, 2}: 3 and so on} where the key is a tuple representing the previous two numbers and the value is the prediction.