Machine Learning (PG)

Monsoon 2020

TOTAL MARKS: 80 ASSIGNMENT 5 DUE DATE: 25 NOV, 2020

Instructions:

(1) The assignment is to be attempted in groups.
(2) You can use only Python as the programming language.
(3) You are free to use math libraries like Numpy, Pandas, SciPy, sklearn, etc.; any library is allowed for visualizations; and utility libraries like os, pickle etc. are fine.
(4) Usage instructions regarding the other libraries is provided in the questions. **Do not use any ML module that is not allowed.**
(5) Create a '.pdf' report that contains your approach, pre-processing, assumptions etc. Add all the analysis related to the question in the written format in the report, **anything not in the report will not be marked**. Use plots wherever required.
(6) Implement code that is modular in nature. Only python (*.py) files should be submitted.
(7) Submit code, readme and analysis files in ZIP format with naming convention '**A5_groupno.zip**' (one submission per group). This nomenclature has to be followed strictly.
(8) You should be able to replicate your results during the demo, failing which will fetch zero marks.
(9) There will be no deadline extension under any circumstances. According to course policies, no late submissions will be considered. So, start early.

Question 1: kNN Algorithm

Use the satellite dataset for this question.

(1) Load the dataset and perform splitting into training and validation sets with 70:30 ratio. Use tsne plot to visualise the dataset. **5 Points**
(2) Implement the kNN algorithm from scratch . You need to find the optimal number of k using the grid search. You may use sklearn for grid search. Plot the error vs number of neighbours graph (k). Report the optimal number of neighbours. **30 Points**
(3) Report the training and the validation accuracy only with optimal value of k using sklearn kNN function. Comment on the accuracy obtained for optimal value of k for both the methods i.e, your implementation and the inbuilt sklearn function. **10 Points**

Question 2: Neural Networks

Use the MNIST subset data for this question.

(1) Split the data into a train and test set with 80:20 (use seed 42). The test set should be held out. **5 Points**
(2) Implement a NN architecture using sklearn with 3 hidden layers - [100, 50, 50]. Assume a Sigmoid activation function in each layer. Report the accuracy and loss. **15 Points**
(3) Use the same architecture as in (2) and plot the decision boundary for 3 different values of *'alpha'* (2 extreme and 1 middle value) b/w the training samples. **15 Points**