Lab - 9
CSL2010: Introduction To Machine Learning
AY 2022-23

Perceptron and SVM

(Due: 27 Oct 2022, 11:59 PM)

General Instructions

1. You need to upload a zip ***<Lab9_Your_Roll_No>.zip***, which contains one file for the task in ***<Lab9_Your_Roll_No>.py*** format and the report for the entire assignment in ***<Lab9_Your_Roll_No>.pdf*** format.
2. Provide your colab file link in the report. **Make sure that your file is accessible**.
3. Submit a single report mentioning your observations for all the tasks.
4. Report/Cite any resources you have used while attempting the assignment.
5. Attempt Q.1 and Q.2 from Part-A, and Part-B during the lab.

[Part-A: 45 Marks]: Perceptron

The dataset is available here. You need to build a prediction model to identify fake notes based on several features extracted from bank-notes' specimen pictures. You need to utilize the perceptron algorithm to model the problem.

Q.1 Pre-Process the dataset by handling missing values and normalizing the data. Split in the ratio 70:20:10 for train-test-validation. [5 Marks]

Q.2 Train a Perceptron model (using the sklearn library). Use K-Fold cross-validation with appropriate K to choose optimal regularization parameter(s). What is the effect of regularization here? [15 Marks]

Q.3 Implement the naive Perceptron algorithm from scratch on the same data. While training, if you observe that the misclassification rate on the validation set is not reducing beyond a certain value (>0), you may stop training. Or, you may also put a threshold on the maximum number of iterations over the training set. Try with different values of "learning rate". Evaluate the learned model and compare it with the results obtained in the previous question. [25 Marks]

[Part-B: 20 Marks]: SVM

Q.1 Using the dataset from the previous question, train an SVM classifier. Analyze the performance (classification accuracy) for different values of 'C'. Also compare your results with those obtained in part-A. [20 Marks]

PRACTICE QUESTION:
Train SVM classifiers using different kernel functions and compare their performance.

Resources

1. SVM (sklearn)
2. Perceptron (sklearn)
3. Reference Implementation of Perceptron from scratch