CS 559: Machine Learning: Fundamentals and Applications

Due: 2/21/2024 Wednesday 11:59 p.m.

The assignment must be individual work and must not be copied or shared. Any tendency to cheat/copy evidence will lead to a 0 mark for the assignment. Students must only use Pandas, NumPy, and Spacy if the coding problem does not specify libraries/packages. Use of other libraries than specified will be penalized. All problems must be submitted in a single notebook file.

1 LDA [40 pts]

This problem expands the binary classifiers discussed in the lecture to the multi-classification case. Use the following code to generate the train data set. The code will generate a random data set with four features and classes.

```
from sklearn import datasets
X, y = datasets.make_blobs(n_samples = 400, n_features= 4,
    centers = 4, cluster_std= 2, random_state= 100)
```

a. [20 pts] Modify the **within-class** ($\mathbf{S_w}$) and **between-class** ($\mathbf{S_B}$) functions in the lecture note to calculate the scatter matrices.

b. [5 pts] Reduce the dimensions to 2-D by finding the eigenvalues, λ, and their corresponding eigenvectors using S_B and S_w matrices found in a). Projectile points and visualize the result.

c. [10 pts] Within the reduced space, perform LDA with the provided implementation code to classify the target. Report the accuracy.

d. [5 pts] Perform scikit-learn LDA to classify the target in the original space by using the original data). Compare the results from c).

2 Perceptron Classification [40 pts]

Generate the data using the following code:

```
from sklearn import datasets
X, y = datasets.make_blobs(n_samples = 400, n_features= 4,
    centers = 2, cluster_std= 1, random_state= 100)
```

a. [20 pts] Finish **perceptron_fit()** code in the lecture notebook.

b. [15 pts] Set an arbitrary w vector to pass to **perceptron_fit()**. Classify the target by increasing the epoch number with an interval of 25. Make a visualization showing the improvement of accuracy as the epoch number grows.

c. [5 pts] Run scikit-learn Perceptron to classify the target and compare the **w** vector with the obtained in b). Did you expect them to be the same? Explain why or why not.