Week 6 - SVM Classification

DS3010 - Introduction to Machine Learning

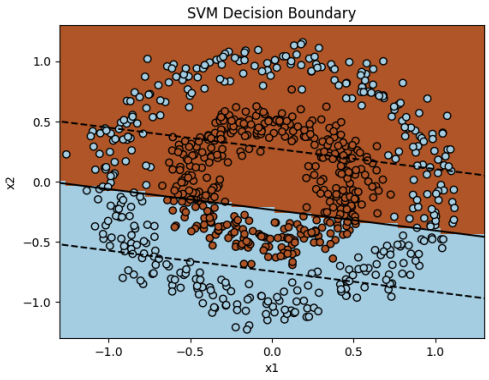
Instructions

1. Provide commented, indented code. Variables should have meaningful names.
2. Submit one .ipynb file containing all answers. The name should be [student name][roll_number] assignment[number].ipynb
3. Read the questions carefully before answering. If a question asks to follow a particular approach or to use a specific data structure, then it must be followed.
4. Write questions in separate text blocks in Jupyter Notebook before the code blocks containing answers.
5. All plots should have appropriate axis labels, titles, and legends.

Tasks for the Lab

1. Linear Classifier (4)

A. Load the given train and test data CSV files.
B. Plot the data points in a 2D plot with different colors for the two classes. 1
C. Create an instance of the SVC (kernel = 'linear', C=0.2) and fit the model. 1
D. Plot the SVM boundary regions (as shown in following figure) learned by the classifier. 1



E. Print the classification report for training data.

F. Predict the labels y for the test data and store them in a new column named 'svm_prediction' in this given test CSV file. 1

2. Non-Linear Classifier (8)

A. Create an instance of the SVC (C=0.2) for different kernels ("sigmoid", "poly","rbf") and fit the model. 1
B. Plot the SVM boundary regions learned by each kernel classifier. 3
C. Print the classification report for training data for each kernel. 1
D. Predict the labels y for the data points provided in test_data.csv and store them in this CSV file's new column named '[kernel]_kernel_prediction'. 1
E. Submit the updated test CSV file. 1
F. Write your observations with comparisons for linear and nonlinear classifiers. 1

3. Hyperparameter Tuning (8)

A. Load the CSV file (UniversalBank.csv) and use the train_test_split function to create the train and test splits. 1
B. Do the standardization and required pre-processing on the dataset. 2
C. Define a param_grid dictionary with the list of permissible values for the hyper-parameters "C"," kernel", "gamma", and "degree". 3

- Kernel - ['linear', 'poly', 'rbf','sigmoid']
- Vary C between 1e-5 to 10 in multiples of 10.
- Gamma - ['scale',' auto']
- Degree - [2,3,4,5]

D. Select the best hyperparameter value to train the final model. 1

E. Print the classification report for test data. 1