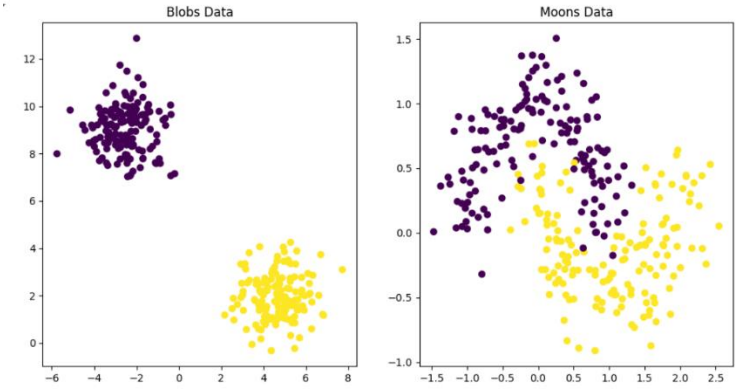
یاسمین طیبی 99442209

SVM document

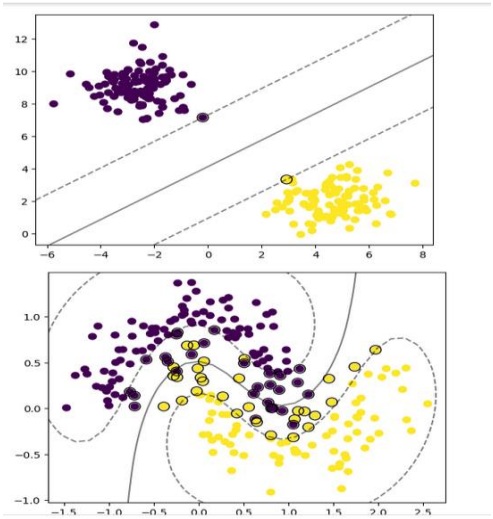
First Part:

Generate Synthetic Data: We'll generate points in 2D space using random functions to create datasets with distinct features and shapes. We will use functions that can generate different patterns each time for diverse experimentation.



Classification Using SVM: We'll classify the generated data using Support Vector Machines (SVM) with and without kernels, testing various kernels and their parameters.

Plotting Decision Boundaries and Margins: We'll plot the decision boundary found by the SVM alongside the training points and also include the margins. Common libraries typically provide functions for this purpose.



Increasing Data Complexity: We'll examine how increasing the complexity of the data affects kernel selection and parameter tuning. The results and analysis will be documented.

Increasing the complexity of the data typically requires more complex kernels like RBF to achieve good classification results. Linear kernels may perform well on simple data but struggle with more complex patterns. By using SVM with different kernels, we can effectively classify 2D data. The choice of kernel depends on the

complexity of the data, and more complex data usually benefits from more sophisticated kernels like RBF. These implementations should be run in an interactive environment like Jupyter Notebook to visualize the results graphically.

Second Part:

I used the MNIST dataset for the fifth part of the network project.

Here are the steps:

1. Load the MNIST Dataset
2. Train SVM with Different Kernels(linear, sigmoid, polynomial)
3. Predict and Evaluate
4. Visualization

Third Part:

First, you need to collect images of license plate characters.

Convert the images to an appropriate format (e.g., grayscale) and resize them.

If needed, use data augmentation methods such as rotation, scaling, etc., to increase data diversity.

Data Splitting:

Split the data into two sets: a training set and a test set. Typically, 70-80% of the data is allocated to the training set and 20-30% to the test set.

Feature Extraction:

Use various methods to extract suitable features from the images. These features can include simple features like Histogram of Oriented Gradients (HOG), pixel values, or more complex features extracted by Convolutional Neural Networks (CNN).

SVM Model Training:

Train the SVM model using the training data and the extracted features.

Tune the SVM model parameters (such as C parameter and kernel type) using techniques like Grid Search or Random Search.

Model Evaluation:

Evaluate the trained model using the test data and calculate metrics such as accuracy.