COMP 6915 - Machine Learning
Assignment 3 Report

by
© Soroush Baghernezhad
& Ramyar Zarza
& Mantra .

Instructor: Karteek Popuri
Department of Computer Science & Scientific Computing
Memorial University of Newfoundland

January 2025

St. John's Newfoundland

In this assignment, we have to train SVM classifiers to classify between the stable normal controls (sNC) group and the stable DAT (sDAT) group. The given dataset consists of 474 data samples with binary labels. There are 8 features corresponding to glucose metabolism in different regions of the brain, namely:

- ctx-lh-inferiorparietal
- ctx-lh-inferiortemporal
- ctx-lh-isthmuscingulate
- ctx-lh-middletemporal
- ctx-lh-posteriorcingulate
- ctx-lh-precuneus
- ctx-rh-isthmuscingulate
- ctx-rh-posteriorcingulate
- ctx-rh-inferiorparietal
- ctx-rh-middletemporal
- ctx-rh-precuneus
- ctx-rh-inferiortemporal
- ctx-lh-entorhinal
- ctx-lh-supramarginal

An SVM is a supervised learning algorithm used primarily for classification tasks. It aims to find the optimal hyperplane that maximizes the margin between two classes of data points. The points closest to the decision boundary, called support vectors, determine this optimal hyperplane.

Mathematically, for a binary classification with labels $y_i \in \{-1, +1\}$, the SVM finds the hyperplane by solving the following optimization problem:

$$\min_{w,b} \frac{1}{2}\|w\|^2 \quad \text{subject to} \quad y_i(w \cdot x_i + b) \geq 1, \quad \forall i$$

This optimization can be solved using its dual form via Lagrangian multipliers:

$$\max_{\alpha} \left(\sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \right)$$

subject to constraints:

$$\sum_{i=1}^{n} \alpha_i y_i = 0, \quad 0 \le \alpha_i \le C$$

Here, C controls the trade-off between margin maximization and classification errors. Points for which $\alpha_i > 0$ are termed support vectors.

Question 1:

In this question, we trained an SVM model with a linear kernel using various fold values for cross-validation. Specifically, we used 50 logarithmically spaced points for the hyperparameter C, ranging from 10^{-3} to 10^4. The performance of the model was evaluated across different fold values k, as depicted in Figure 1. After identifying the optimal C for each fold scenario, we retrained the SVM using this best-performing parameter, resulting in the following outcomes:

```
Best C: 5, Best Degree: 2
performance metrics for linear kerenl SVM:
Accuracy: 0.84
Precision: 0.59
Recall : 0.93
Specificity: 0.81
Balanced Accuracy: 0.87
```

In Figure 1 it can be seen that all different folds converged around 85% accuracy with the optimal C value ranging between 2 yo 5.18.

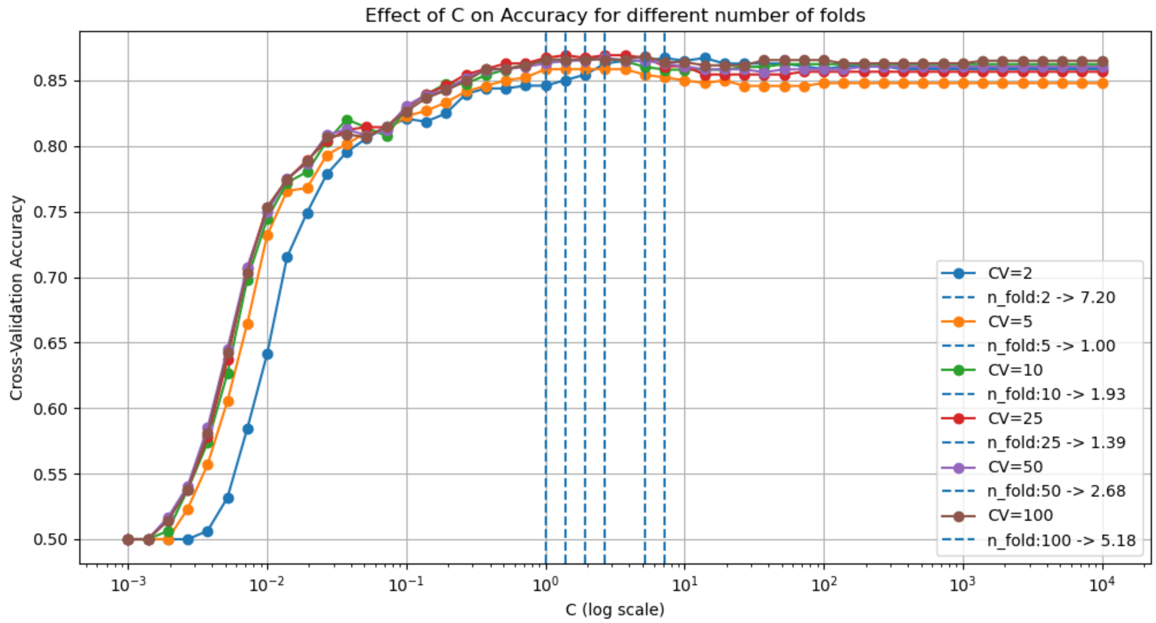

Figure 1

Question 2

In Part 1, we trained SVM classifiers using a polynomial kernel. Support Vector Machines (SVM) with a polynomial kernel address nonlinear classification by mapping input data into a higher-dimensional space. The polynomial kernel is defined as:

$$K(x_i, x_j) = (\gamma x_i^T x_j + r)^d$$

where x_i, x_j are input vectors, γ (scaling parameter), r (coefficient), and d (degree) are kernel parameters. We performed hyperparameter tuning by conducting grid search over two hyperparameters: the regularization parameter (C) and the polynomial degree of the kernel function. The considered values for C were $\{0.1, 0.2, 0.5, 0.7, 1, 1.5, 2, 3, 5, 10, 100\}$, and for the polynomial degree, we tested values from 1 to 6. The optimal combination of these parameters was selected based on cross-validation accuracy. The resulting optimal parameters were then used to retrain the SVM classifier, yielding the following results.

```
performance metric for poly kernel SVM:
Accuracy: 0.83
Precision: 0.58
Recall : 0.94
Specificity: 0.80
Balanced Accuracy: 0.87
```

```
Best C: 10, Best gamma: 1.0
```

In Figure 2, we can observe two linear trends for the optimal values of C and the polynomial degree. In certain regions of the grid search, increasing both the degree and the C value simultaneously results in higher accuracy. It can be noted that, in most cases, accuracy remains consistent along a diagonal pattern.

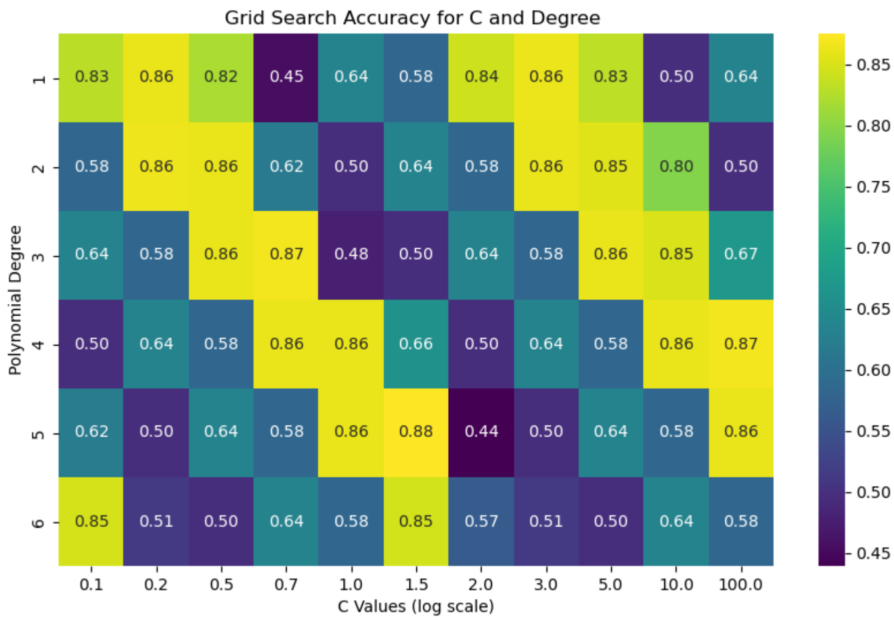

Figure 2

Question 3:

In this question, we find the best C and γ parameters for our RBF kernel SVM, where the RBF kernel function is defined as:

$$K(x_i, x) = \exp\left(-\gamma \|x_i - x\|^2\right)$$

with $\gamma > 0$ as a hyperparameter controlling the spread of the kernel.

The values we searched in our 2D grid search for C and γ are:

$$C : [0.1, 0.2, 0.5, 0.7, 1, 1.5, 2, 3, 5, 10, 100]$$

$$\gamma : \text{np.logspace}(-3, 0, 10)$$

After finding the best parameters, we retrained our RBF model and obtained the following results:

```
    Best C: 10, Best gamma: 1.0
performance metric for rbf kernel SVM:
Accuracy: 0.87
Precision: 0.65
Recall : 0.94
Specificity: 0.85
Balanced Accuracy: 0.90
```

Similar to Figure 2, in Figure 3, the anti-diagonal elements of our grid search matrix tend to have the same accuracies, suggesting that there is an inverse relationship between the C and γ values. As C increases, we need to decrease γ to achieve high accuracy.

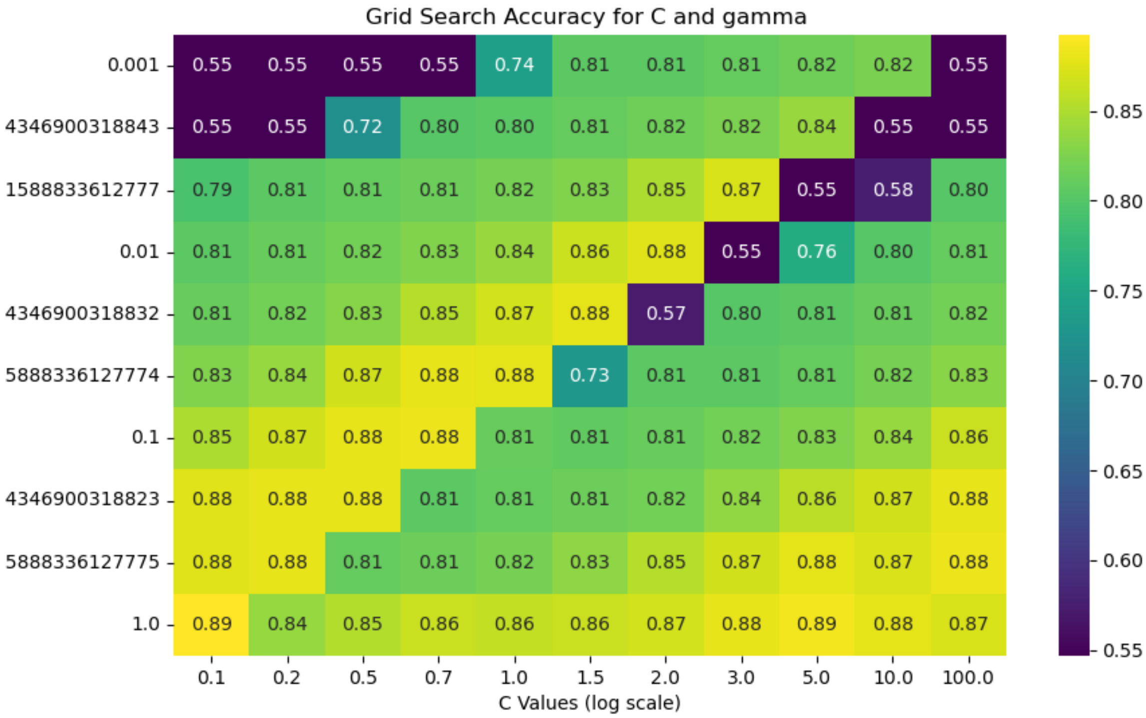

Figure 3

Question 4:

In Question 4, we used same RBF kernel method which we had best results with, then we added normalization to our data and we achieved following results:

```
Best C: 100, Best gamma: 0.21544346900318823
performance metric for rbf kernel with normalization SVM:
Accuracy: 0.87
Precision: 0.65
Recall : 0.94
Specificity: 0.85
Balanced Accuracy: 0.90
```