# **Assignment 3:** *Support Vector Machine*

Date: *Feb 23, 2025*
Due : *Mar 09, 2025*

**Acknowledgment** This assignment has been adapted in part from the materials of the courses ECE421 by N. Papernot and ECE1513 by S. Emara.

**Code of Honor** Assignments are designed to enhance your understanding and advance your skills, constituting a significant portion of your final assessment. They must be completed individually, as engaging in any form of academic dishonesty violates the principles of the Code of Honor. If you encounter any challenges while solving the assignments, please contact the instructional team for guidance.

**How to Submit** This assignment has 3 questions. For each question, you need to upload one file on Crowdmark. If the question asks for your code, you need to copy and paste your code in the file that you submit or print your code and its output terminal as PDF.

**Grading** The grades add up to 100 and comprise roughly 10% of the final mark.

**Deadline** The deadline for your submission is on **Mar 09, 2025 at 11:59 PM**. Please note that this deadline is strict and **no late submission will be accepted.**

## Questions

Question 1 [40 Points] **(Model Parameters for SVC)** Consider the following dataset with four samples

$$\mathbb{D} = \{(\boldsymbol{x}, v) : ([1,1], +1), ([-1,-1], -1), ([1,0], -1), ([0,1], -1)\}$$

We intend to find the SVC both geometrically and via standard approach.

1. Plot the dataset and draw the maximum-margin line. Use geometry to give the equation of this line. You may use it to verify your answers at the end of this question. Note that the values are not exactly the same; however, the ratio will be the same, i.e., you can find SVC solution by a different scaling factor.
2. Specify the support vector.
3. Recall the optimization problem we defined for the SVC. We now consider it with a *bias*, i.e.,

$$\min_{\mathbf{w}, b} \|\mathbf{w}\|^2 \quad \textit{subject to} \quad v_n(\mathbf{w}^\mathsf{T} \cdot \boldsymbol{x}_n + b) \geq 1 \textit{ for } \; n = 1, \dots, 4. \tag{0.1}$$

   Using the method of Lagrange multipliers, specify the dual Lagrangian $\ell(\mathbf{w}, b, \boldsymbol{\lambda})$ for this constrained optimization, where $\boldsymbol{\lambda} \in \mathbb{R}_+^4$ is the vector of Lagrangian multipliers all being non-negative.

4. Write the dual problem as

$$\max_{\boldsymbol{\lambda}} \min_{\mathbf{w},b} \ell\left(\mathbf{w}, b, \boldsymbol{\lambda}\right) \tag{Dual}$$

and solve the inner minimization for a fixed $\boldsymbol{\lambda}$. Note that in this case, we have to equations; namely,

$$\nabla_{\mathbf{w}} \ell\left(\mathbf{w}, b, \boldsymbol{\lambda}\right) = 0$$
$$\frac{\partial}{\partial b} \ell\left(\mathbf{w}, b, \boldsymbol{\lambda}\right) = 0$$

5. Use the second equation for finding the stationary point, i.e.,

$$\frac{\partial}{\partial b} \ell\left(\mathbf{w}, b, \boldsymbol{\lambda}\right) = 0$$

to find a relationship between the dual variables $\lambda_n$.

6. Replace the stationary point computed in the last part in (Dual) to find the dual optimization as

$$\max_{\boldsymbol{\lambda} \geq 0} F\left(\boldsymbol{\lambda}\right) \tag{0.2}$$

Determine the expression $F\left(\boldsymbol{\lambda}\right)$.

7. We now get back to our toy dataset $\mathbb{D}$. Substitute the dataset into the dual problem, and solve it to find the value for dual variables $\lambda_n$. You may also use complementary slackness to find the dual variable.

8. Check the dual variables and confirm their value for support vectors recalling that they are only non-zero for support vectors.

9. Determine $\mathbf{w}$ and $b$, and compare your solution with what you found geometrically.

QUESTION 2 [40 Points] **(Linear Classification vs. SVC)** In this problem, we compare the performance of a binary linear classifier to a SVC. Because the purpose of this problem is to compare the performance of the two classifiers, we can use their implementation in `sklearn`.
Consider the `iris` dataset included in `sklearn`. For all parts of this question (except the last one), we will only consider the first 100 entries of the dataset. With the help of `train_test_split` from `sklearn.model_selection`, split the dataset into a training set and a test set. For now, you can do so by setting the argument `test_size` to 0.8 in `train_test_split`. To ensure results below are comparable and reproducible, set the random state argument of your `train_test_split` calls to 0: this controls the shuffling applied to the data before applying the split.

1. Implement a binary linear classifier on the first two dimensions (sepal length and width) of the `iris` dataset and plot its decision boundary.
   **Hint:** `sklearn` *refers to the binary linear classifier as Logistic Regression.*

2. Report the accuracy of your binary linear classifier on both the training and test sets. You may use `from sklearn.metrics import accuracy_score`.

3. Implement the SVC on the first two dimensions. Plot the decision boundary of the classifier and its margins. Circle the support vectors.

4. Report the accuracy of your SVC classifier on both the training and test sets.

5. What is the value of the margin? Show how are you getting it.

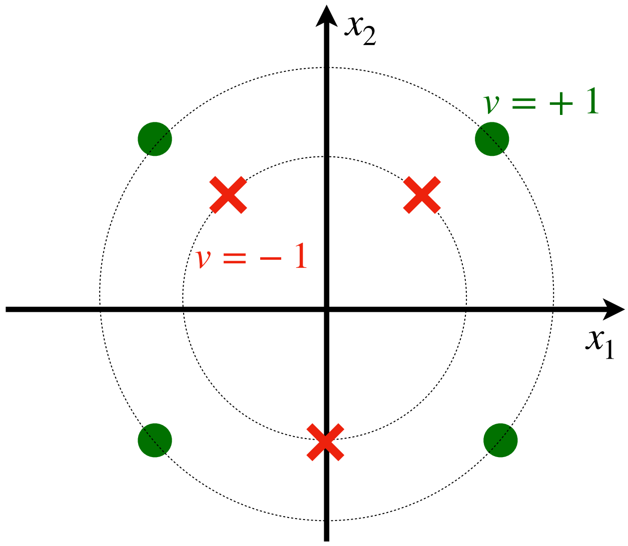
6. Which vector is orthogonal to the decision boundary?

7. Do the binary linear classifier and SVC have the same decision boundaries?

8. Split the `iris` dataset again into a training and test set, this time setting `test_size` to 0.4 when calling `train_test_split`. Train the SVC again. Does the decision boundary change? How about the test accuracy? Justify what you observe.
**Hint:** *Think about the support vectors.*

9. Now, consider all 150 entries in the iris dataset, which introduces a new class. Retrain the SVC and test it with `test_size=0.4`. You should find that the data points are not linearly separable. How can you deal with it? Justify your answer and plot the decision boundary of your new proposed classifier.

QUESTION 3 [20 Points] **(Nonlinear SVM)** A dataset consists of 7 samples with three of them being labeled by $v = -1$ (crosses) and four others with $v = 1$ (circles), as shown in the figure below.




Note that as shown in the figure, the samples lie on two circles.

- Find a *nonlinear* classifier, such that the dataset gets classified perfectly. In case needed, call the radius of the inner and outer circles, $r_1$ and $r_2$, respectively.

- Suggest an embedding which can map the samples to a 3-dimensional space, in which the points are linearly separable.

- Determine the kernel for your suggested embedding, and explain how you can use it to find the SVM. You do not need to find the SVM, just explain the procedure in words.