TP 1 : Hard margin SVC and Soft margin SVC

Objective:

The objective of this lab is to gain an understanding of how Support Vector Machines (SVMs) work for binary classification and to train the model using the Sequential Minimal Optimization (SMO) algorithm. The implementation of both Hard margin SVC and Soft margin SVC will be carried out using a Python library such as NumPy, with visualization by Matplotlib.

Hard margin SVC and Soft margin SVC

We give the mathematical models of the dual forms of Hard margin SVC and Soft margin SVC respectively:

Hard margin SVC:

- Input: training set : $S=\{(x_i,y_i)\}_{i=1}^n$ where $x_i \in \mathbb{R}^d, y_i \in \{1,-1\}$
- Contruct and solve the optimzation problem Dual $-SVC$ obtaining $\lambda^* = (\lambda_1^*, \ldots, \lambda_n^*)$

$$\text{Dual } -SVC: \begin{cases} \text{Max} \quad \sum_{i=1}^n \lambda_i - \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \lambda_j \lambda_i y_j y_i \left(x_j^T x_i\right) \\ \text{s.t} \quad \sum_{i=1}^n \lambda_i y_i = 0 \\ \lambda_i \geq 0 i = 1, \ldots, n \end{cases}$$

- Choose a possitive component of λ^*, λ_j^*, and Compute

$$w^* = \sum_{i=1}^n \lambda_i^* y_i x_i \text{ and } b^* = y_j - \sum_{i=1}^n \lambda_i^* y_i \left(x_j^T x_i\right) \rightarrow h_{w^*,b^*}(x) = \left(w^*\right)^T x + b^*$$

- Contruit the separating hyperplane $\left(w^*\right)^T x + b^*$ and the decision function is

$$h_S(x) = \operatorname{sign}\left(h_{w^*,b^*}(x)\right) \rightarrow L_S\left(h_S\right) = \frac{1}{n}\sum_{i=1}^n 1_{\left\{h_{w^*,b^*}(x_i) \neq y_i\right\}}$$

Soft margin SVC:

- Input: training set : $\{(x_i,y_i)\}_{i=1}^n$ where $x_i \in \mathbb{R}^d, y_i \in \{1,-1\}$
- Choose an approriate penalty parameter $C > 0$
- Contruct and solve the optimzation problem Dual: $\mathrm{C}-SVC$ obtaining $\lambda^* = (\lambda_1^*, \ldots, \lambda_n^*)$

$$\text{Dual } : -SVC \begin{cases} \text{Max} \sum_{i=1}^n \lambda_i - \frac{1}{2}\sum_{i=1}^n \sum_{j=1}^n \lambda_j \lambda_i y_j y_i \left(x_j^T x_i\right) \\ \text{s.t } \sum_{i=1}^n \lambda_i y_i = 0 \\ C \geq \lambda_i \geq 0, i = 1, \ldots, n \end{cases}$$

- Choose a possitive component of $\lambda^*, \lambda_j^* \in (0, C)$, and Compute

$$w^* = \sum_{i=1}^{n} \lambda_i^* y_i x_i \text{ and } b^* = y_j - \sum_{i=1}^{n} \lambda_i^* y_i \left({x_j}^T x_i \right)$$

- Contruit the separating hyperplane $(w^*)^T x + b^*$ and the decision function is

$$h(x) = \operatorname{sign}\left((w^*)^T x + b^* \right) \rightarrow L_S(h_S) = \frac{1}{n} \sum_{i=1}^{n} 1_{\{h_{w^*,b^*}(x_i) \neq y_i\}}$$

Method:

1. Choose two simple dataset of binary classification, these datasets data must be with classes that are separated linearly with a bit of noise, one which is $2\,\mathrm{D}$ and the other $3\,\mathrm{D}$.
2. Perform exploratory analysis of the data and preprocess the data if necessary.
3. Split data into train and test
4. Implement the SMO optimization algorithm.
5. Implement Hard margin SVC and Soft margin SVC.
6. Train the two models, and compare their accuracy, what can we conclude?
7. Visualize the classification results (separation hyperplane in $2\,\mathrm{D}$ or $3\,\mathrm{D}$ depending on the type of dataset).