Support Vector Machine Notes

J. S.

2025.02.18

Abstract

This is my personal study notes on support vector machine (SVM) [Cortes and Vapnik, 1995].

Contents

1 Mathematical Foundations of Support Vector Machine

1.1 Problem Formulation

Consider a training dataset

$$\{(\mathbf{x}_1, c_1), (\mathbf{x}_2, c_2), \cdots, (\mathbf{x}_m, c_m)\}$$

where $\mathbf{x}_i \in \mathbb{R}^n$ are feature vectors and $c_i \in \{-1, +1\}$ are the corresponding class labels. The goal of SVM is to find the hyperplane that maximally separates the data points of the two classes.

The hyperplane is defined by the equation

$$\mathbf{w}^T\mathbf{x} + b = 0$$

where $\mathbf{w} \in \mathbb{R}^n$ is the normal vector (not necessarily standardized) to the hyperplane and $b \in \mathbb{R}$ is the bias term which determines the distance to the origin.

If the hyperplane classifies the data points correctly, then $c_i = +1 \Leftrightarrow \mathbf{w}^T\mathbf{x}_i + b > 0$, $c_i = -1 \Leftrightarrow \mathbf{w}^T\mathbf{x}_i + b < 0$. We scale $\mathbf{w}$ and b such that

$$\begin{cases} \mathbf{w}^T\mathbf{x}_i + b > +1, & c_i = +1, \\ \mathbf{w}^T\mathbf{x}_i + b < -1, & c_i = -1. \end{cases}$$

To unify the notations, we can use $c_i(\mathbf{w}^T\mathbf{x}_i + b)$ to measure the distance from $\mathbf{x}_i$ to the decision boundary. There are some nearest points where $c_i(\mathbf{w}^T\mathbf{x}_i + b) = 1$, which is denoted as "support vector".

The sum of distance between the hyperplane and the support vectors from two different classes is $\gamma = \frac{2}{\|\mathbf{w}\|}$, which is called "margin". To find the hyperplane with maximum margin is to find $\mathbf{w}$ and b that maximizes γ:

$$\begin{aligned} \max_{\mathbf{w},b} \quad & \frac{2}{\|\mathbf{w}\|} \\ \text{s.t.} \quad & c_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, i = 1, 2, \cdots, m. \end{aligned}$$

1.2 Solution to SVM in Separable Case

We reformulate the problem into the form of a classical constrained optimization problem:

$$\begin{aligned} \min_{\mathbf{w},b} \quad & \|\mathbf{w}\|_2^2 \\ \text{s.t.} \quad & c_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, i = 1, 2, \cdots, m. \end{aligned}$$

By KKT theory[1], we define the Lagrangian function with Lagrangian multipliers $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_m)$:

$$\begin{aligned} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) &= \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{m} \lambda_i \Big[1 - c_i(\mathbf{w}^T\mathbf{x}_i + b)\Big] \\ &= \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^{m} \lambda_i - \sum_{i=1}^{m} \lambda_i c_i \mathbf{w}^T\mathbf{x}_i - \sum_{i=1}^{m} \lambda_i c_i b. \end{aligned}$$

The KKT conditions are

$$1 - c_i(\mathbf{w}^T\mathbf{x}_i + b) \leq 0 \quad \text{(Primal Feasibility)},$$

$$\lambda_i \geq 0 \quad \text{(Dual Feasibility)},$$

$$\lambda_i \Big[1 - c_i(\mathbf{w}^T\mathbf{x}_i + b)\Big] = 0 \quad \text{(Complementary Slackness)},$$

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) = \mathbf{w} - \sum_{i=1}^{m} \lambda_i c_i \mathbf{x}_i = 0 \quad \text{(Stationarity)},$$

$$\nabla_b\mathcal{L}(\mathbf{w}, b, \boldsymbol{\lambda}) = -\sum_{i=1}^{m} \lambda_i c_i = 0 \quad \text{(Stationarity)}.$$

Solving the conditions, we can get the solution to the SVM problem.

[1] Details can be found in Section 2.1.

By plugging back the KKT conditions, we have

$$\begin{aligned}
\mathcal{L}(w, b, \lambda) &= \frac{1}{2}\|\mathbf{w}\|_2^2 - \mathbf{w}^T \sum_{i=1}^{m} \lambda_i c_i \mathbf{x}_i - b \sum_{i=1}^{m} \lambda_i c_i + \sum_{i=1}^{m} \lambda_i \\
&= \frac{1}{2}\|\mathbf{w}\|_2^2 - \mathbf{w}^T \mathbf{w} - 0 + \sum_{i=1}^{m} \lambda_i \\
&= -\frac{1}{2}\mathbf{w}^T \mathbf{w} + \sum_{i=1}^{m} \lambda_i \\
&= -\frac{1}{2}\Big(\sum_{i=1}^{m} \lambda_i c_i \mathbf{x}_i^T\Big)\Big(\sum_{j=1}^{m} \lambda_j c_j \mathbf{x}_j\Big) + \sum_{j=1}^{m} \lambda_i \\
&= -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \lambda_i \lambda_j c_i c_j \mathbf{x}_i^T \mathbf{x}_j + \sum_{i=1}^{m} \lambda_i.
\end{aligned}$$

So the Lagrangian can be rewritten as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\lambda}) &= \sum_{i=1}^{m} \lambda_i - \frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m} \lambda_i \lambda_j c_i c_j \mathbf{x}_i^T \mathbf{x}_j, \\
\text{s.t.} \quad & \begin{cases} \lambda_i \geq 0, \quad i = 1, 2, \cdots, m, \\ \sum_{i=1}^{m} \lambda_i c_i = 0. \end{cases}
\end{aligned}$$

By maximizing $\mathcal{L}(\boldsymbol{\lambda})$, we get $\lambda_1^*, \lambda_2^*, \cdots, \lambda_m^*$. Apply the KKT conditions back, we have

$$\mathbf{w}^* = \sum_{i=1}^{m} \lambda_i^* c_i \mathbf{x}_i.$$

To simplify the calculation, we can take advantage of the property of support vectors. Analyze the three KKT conditions:

$$1 - c_i(\mathbf{w}^T \mathbf{x}_i + b) \leq 0 \quad \text{(Primal Feasibility)},$$

$$\lambda_i \geq 0 \quad \text{(Dual Feasibility)},$$

$$\lambda_i \Big[1 - c_i(\mathbf{w}^T \mathbf{x}_i + b)\Big] = 0 \quad \text{(Complementary Slackness)}.$$

It's easy to derive that

$$c_i(\mathbf{w}^T \mathbf{x}_i + b) = 1 \Longleftrightarrow \lambda_i > 0,$$

$$c_i(\mathbf{w}^T \mathbf{x}_i + b) > 1 \Longleftrightarrow \lambda_i = 0.$$

So we have $\lambda_i > 0$ only for support vectors, and $\lambda_i = 0$ for all other vectors.

By taking a positive support vector $\mathbf{x}_k$ where $c_k = +1$, we have

$$b^* = 1 - \mathbf{w}^{*T} \mathbf{x}_k.$$

Written in another way, since the support vectors have the smallest margins, we have

$$b^* = 1 - \min_{i:c_i=+1} \mathbf{w}^T \mathbf{x}_i.$$

1.3 Solution to SVM in Nonseparable Case

If there is no separating hyperplane, there is no feasible solution to the problem we wrote above. We can make slight adjustments to fit the nonseparable case. Mistakes are allowed now, but we add a penalty.

We change our primal problem to this new primal problem:

$$\begin{aligned} \min_{\mathbf{w},b,\boldsymbol{\xi}} \quad & \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{m}\xi_i, \\ \text{s.t.} \quad & \begin{cases} c_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1 - \xi_i, & i = 1, 2, \cdots, m, \\ \xi_i > 0, & i = 1, 2, \cdots, m. \end{cases} \end{aligned}$$

So the constraints allow some slack of size ξ_i, but we pay a price for it in the objective. That is, if $c_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$, then ξ_i is set to 0, and penalty is 0. Otherwise, if $c_i(\mathbf{w}^T\mathbf{x}_i + b) = 1 - \xi_i$, penalty is set to ξ_i. Parameter $C \in \mathbb{R}^+$ adjusts the scale of slackness penalty.

To minimize the above object function, we can form the Lagrangian:

$$\begin{aligned} \mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{m}\xi_i + \sum_{i=1}^{m}\alpha_i\Big[1 - \xi_i - c_i(\mathbf{w}^T\mathbf{x}_i + b)\Big] + \sum_{i=1}^{m}\beta_i(-\xi_i) \\ &= \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{m}\xi_i + \sum_{i=1}^{m}\alpha_i - \sum_{i=1}^{m}\alpha_i\xi_i - \sum_{i=1}^{m}\beta_i\xi_i - \mathbf{w}^T\sum_{i=1}^{m}\alpha_i c_i\mathbf{x}_i - b\sum_{i=1}^{m}\alpha_i c_i \end{aligned}$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_m)$ and $\boldsymbol{\beta} = (\beta_1, \beta_2, \cdots, \beta_m)$ are Lagrange multipliers.

The KKT conditions are:

$$1 - \xi_i - c_i(\mathbf{w}^T\mathbf{x}_i + b) \leq 0 \quad \text{(Primal Feasibility)},$$

$$\alpha_i \geq 0 \quad \text{(Dual Feasibility)},$$

$$\beta_i \geq 0 \quad \text{(Dual Feasibility)},$$

$$\alpha_i\Big[1 - \xi_i - c_i(\mathbf{w}^T\mathbf{x}_i + b)\Big] = 0 \quad \text{(Complementary Slackness)},$$

$$\beta_i\xi_i = 0 \quad \text{(Complementary Slackness)},$$

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \mathbf{w} - \sum_{i=1}^{m}\alpha_i c_i\mathbf{x}_i = 0 \quad \text{(Stationarity)},$$

$$\nabla_b\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = -\sum_{i=1}^{m}\alpha_i c_i = 0 \quad \text{(Stationarity)},$$

$$\nabla_{\xi_i}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = C - \alpha_i - \beta_i = 0 \quad \text{(Stationarity)}.$$

Since $\alpha_i \geq 0, \beta_i \geq 0, C - \alpha_i - \beta_i = 0$, we have $\alpha_i, \beta_i \in [0, C]$.

By plugging back the KKT conditions, we have

$$\begin{aligned}
\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) &= \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^m \xi_i + \sum_{i=1}^m \alpha_i - \sum_{i=1}^m \alpha_i\xi_i - \sum_{i=1}^m \beta_i\xi_i - \mathbf{w}^T\sum_{i=1}^m \alpha_i c_i \mathbf{x}_i - b\sum_{i=1}^m \alpha_i c_i \\
&= \frac{1}{2}\|\mathbf{w}\|_2^2 + \sum_{i=1}^m \alpha_i + \left(C\sum_{i=1}^m \xi_i - \sum_{i=1}^m \alpha_i\xi_i - \sum_{i=1}^m \beta_i\xi_i\right) - \mathbf{w}^T\mathbf{w} - 0 \\
&= \frac{1}{2}\|\mathbf{w}\|_2^2 - \mathbf{w}^T\mathbf{w} + \sum_{i=1}^m \alpha_i + \sum_{i=1}^m \xi_i\Big(C - \alpha_i - \beta_i\Big) \\
&= -\frac{1}{2}\mathbf{w}^T\mathbf{w} + \sum_{i=1}^m \alpha_i + 0 \\
&= -\frac{1}{2}\Big(\sum_{i=1}^m \alpha_i c_i \mathbf{x}_i^T\Big)\Big(\sum_{j=1}^m \alpha_j c_j \mathbf{x}_j\Big) + \sum_{j=1}^m \alpha_i \\
&= -\frac{1}{2}\sum_{i=1}^m\sum_{j=1}^m \alpha_i\alpha_j c_i c_j \mathbf{x}_i^T\mathbf{x}_j + \sum_{i=1}^m \alpha_i.
\end{aligned}$$

So the Lagrangian function can be rewritten as

$$\begin{aligned}
\mathcal{L}(\boldsymbol{\alpha}) &= \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m\sum_{j=1}^m \alpha_i\alpha_j c_i c_j \mathbf{x}_i^T\mathbf{x}_j, \\
\text{s.t.} \quad &\begin{cases} 0 \le \alpha_i \le C, & i = 1, 2, \cdots, m, \\ \sum_{i=1}^m \alpha_i c_i = 0. \end{cases}
\end{aligned}$$

So the only difference from the original problem's Lagrangian is that the constraint $\alpha_i \ge 0$ is changed to $0 \le \alpha_i \le C$. By maximizing $\mathcal{L}(\boldsymbol{\alpha})$, we can get $\alpha_1^*, \alpha_2^*, \cdots, \alpha_m^*$, and $\mathbf{w}^*$ and b^* can be derived by the same process as previouly introduced.

1.4 SVM with Kernels

Basic idea: If we can't separate positives from negatives in a low-dimensional space using a hyperplane, we can map everything to a higher dimensional space where the samples can be separated.

The kernel trick is that if we have an algorithm (like SVM) where the samples appear only in inner products, we can freely replace the inner product with a different one. Assume there is a map $\Phi(\cdot) : \mathbb{R}^n \to \mathbb{R}^{n'}$. Instead of applying SVM to the original $\mathbf{x}$, we could apply it to $\Phi(\mathbf{x})$.

In the previous discussion, we derived the Lagrangian of SVM:

$$\mathcal{L} = \sum_{i=1}^m \alpha_i - \frac{1}{2}\sum_{i=1}^m\sum_{j=1}^m \alpha_i\alpha_j c_i c_j \mathbf{x}_i^T\mathbf{x}_j.$$

We can replace the inner product with another one, without even knowing the exact form of Φ. In other words, we can replace $\mathbf{x}^T\mathbf{z}$ with $\kappa(\mathbf{x}, \mathbf{z})$, where $\kappa(\mathbf{x}, \mathbf{z}) = \langle\Phi(\mathbf{x}), \Phi(\mathbf{z})\rangle$. Here $\kappa(\cdot,\cdot) : \mathbb{R}^{n'} \times \mathbb{R}^{n'} \to \mathbb{R}$ is called the *kernel*.

Common kernels:

- Linear kernel:

$$\kappa(\mathbf{x}, \mathbf{z}) = \mathbf{x}^T\mathbf{z};$$

- Polynomial kernel:
$$\kappa(\mathbf{x}, \mathbf{z}) = (\mathbf{x}^T\mathbf{z} + c)^d, \quad d \in \mathbb{N}^*;$$
- Gaussian kernel:
$$\kappa(\mathbf{x}, \mathbf{z}) = \exp\Big(-\frac{\|\mathbf{x} - \mathbf{z}\|_2^2}{2\sigma^2}\Big), \quad \sigma > 0;$$
- Laplace kernel:
$$\kappa(\mathbf{x}, \mathbf{z}) = \exp\Big(-\frac{\|\mathbf{x} - \mathbf{z}\|_1}{\sigma}\Big), \quad \sigma > 0;$$
- Sigmoid kernel:
$$\kappa(\mathbf{x}, \mathbf{z}) = \tanh(\beta\mathbf{x}^T\mathbf{z} + \theta), \quad \beta > 0, \quad \theta < 0.$$

1.5 Support Vector Regression (SVR)

Consider a training dataset
$$\{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), \cdots, (\mathbf{x}_m, y_m)\}$$
where $\mathbf{x}_i \in \mathbb{R}^n$ are feature vectors and $y_i \in \mathbb{R}$ are response variables.

SVR aims to find a function $f(\mathbf{x}) = \mathbf{w}^T\mathbf{x} + b$ that for each $\mathbf{x}_i$, $f(\mathbf{x}_i)$ is close to y_i. Denote $\epsilon \in \mathbb{R}^+$, the goal can be written as
$$|y_i - (\mathbf{w}^T\mathbf{x}_i + b)| \leq \varepsilon.$$
It is possible that no such function $f(\mathbf{x})$ exists to satisfy these constraints for all points. To deal with otherwise infeasible constraints, introduce slack variables $\boldsymbol{\xi} = (\xi_1, \xi_2, \cdots, \xi_m)$ (for positive residual) and $\boldsymbol{\xi}^* = (\xi_1^*, \xi_2^*, \cdots, \xi_m^*)$ (for negative residual) for each point.

The primal problem of SVR can be formulated as
$$\min_{\mathbf{w}, b, \boldsymbol{\xi}, \boldsymbol{\xi}^*} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{m}(\xi_i + \xi_i^*),$$
$$\text{s.t.} \quad \begin{cases} -\varepsilon - \xi_i^* \leq y - f(\mathbf{x}) \leq \varepsilon + \xi_i, & i = 1, 2, \cdots, m, \\ \xi_i, \xi_i^* \geq 0, & i = 1, 2, \cdots, m. \end{cases}$$

The constant C is the box constraint, a positive numeric value that controls the penalty imposed on observations that lie outside the epsilon margin (ε) and helps to prevent overfitting.

Actually, the performance of fitting is measured by ε-insensitive loss, which ignores errors that are within ϵ distance of the observed value by treating them as 0. The form is as follows:
$$\xi_i \text{ or } \tilde{\xi}_i = L_\varepsilon(f(\mathbf{x}_i), y_i) = \begin{cases} 0, & \text{if} \quad |y - f(\mathbf{x})| \leq \varepsilon, \\ |y_i - f(\mathbf{x}_i)| - \varepsilon & \text{otherwise.} \end{cases}$$

To obtain the dual formula, construct a Lagrangian function from the primal function by introducing

nonnegative multipliers $\alpha_i, \tilde{\alpha_i}, \beta_i, \tilde{\beta}_i$ for each observation $\mathbf{x}_i$:

$$\begin{aligned}
\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}) &= \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{m}(\xi_i + \tilde{\xi}_i) \\
&\quad + \sum_{i=1}^{m}\alpha_i(y_i - \mathbf{w}^T\mathbf{x}_i - b - \xi - \varepsilon) - \sum_{i=1}^{m}\beta_i\xi_i \\
&\quad + \sum_{i=1}^{m}\tilde{\alpha}_i(-y_i + \mathbf{w}^T\mathbf{x}_i + b - \tilde{\xi} - \varepsilon) - \sum_{i=1}^{m}\tilde{\beta}_i\tilde{\xi}_i \\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{m}(\xi_i + \tilde{\xi}_i) + \sum_{i=1}^{m}\Big(\alpha_i - \tilde{\alpha}_i\Big)y_i - \mathbf{w}^T\sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)\mathbf{x}_i \\
&\quad - b\sum_{i=1}^{m}\Big(\alpha_i - \tilde{\alpha}_i\Big) - \sum_{i=1}^{m}\alpha_i\xi_i - \sum_{i=1}^{m}\tilde{\alpha}_i\tilde{\xi}_i - \varepsilon\sum_{i=1}^{m}\Big(\alpha_i + \tilde{\alpha}_i\Big) - \sum_{i=1}^{m}\beta_i\xi_i - \sum_{i=1}^{m}\tilde{\beta}_i\tilde{\xi}_i \\
&= \frac{1}{2}\mathbf{w}^T\mathbf{w} - \mathbf{w}^T\sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)\mathbf{x} + \left[\sum_{i=1}^{m}(C - \alpha_i - \beta_i)\xi_i + \sum_{i=1}^{m}(C - \tilde{\alpha}_i - \tilde{\beta}_i)\tilde{\xi}_i\right] \\
&\quad + \sum_{i=1}^{m}\Big(\alpha_i - \tilde{\alpha}_i\Big)y_i - b\sum_{i=1}^{m}\Big(\alpha_i - \tilde{\alpha}_i\Big) - \varepsilon\sum_{i=1}^{m}\Big(\alpha_i + \tilde{\alpha}_i\Big).
\end{aligned}$$

The KKT conditions are:

$$y_i - \mathbf{w}^T\mathbf{x}_i - b - \varepsilon - \xi_i \le 0 \quad \text{(Primal Feasibility)},$$

$$-y_i + \mathbf{w}^T\mathbf{x}_i + b - \varepsilon - \xi_i \le 0 \quad \text{(Primal Feasibility)},$$

$$\alpha_i, \tilde{\alpha_i}, \beta_i, \tilde{\beta}_i \ge 0 \quad \text{(Dual Feasibility)},$$

$$\sum_{i=1}^{m}\alpha_i(y_i - \mathbf{w}^T\mathbf{x}_i - b - \xi - \varepsilon) = 0 \quad \text{(Complementary Slackness)},$$

$$\sum_{i=1}^{m}\tilde{\alpha}_i(-y_i + \mathbf{w}^T\mathbf{x}_i + b - \tilde{\xi} - \varepsilon) = 0 \quad \text{(Complementary Slackness)},$$

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}) = \mathbf{w} - \sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)\mathbf{x} = 0 \quad \text{(Stationarity)},$$

$$\nabla_b\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}) = \sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i) = 0 \quad \text{(Stationarity)},$$

$$\nabla_{\xi_i}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}) = C - \alpha_i - \beta_i = 0 \quad \text{(Stationarity)},$$

$$\nabla_{\tilde{\xi}_i}\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}) = C - \tilde{\alpha}_i - \tilde{\beta}_i = 0 \quad \text{(Stationarity)}.$$

By plugging the KKT conditions back, we have

$$\begin{aligned}
\mathcal{L}(\mathbf{w}, b, \boldsymbol{\xi}, \tilde{\boldsymbol{\xi}}, \boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}, \boldsymbol{\beta}, \tilde{\boldsymbol{\beta}}) &= -\frac{1}{2}\mathbf{w}^T\mathbf{w} + 0 + \sum_{i=1}^{m}\Big(\alpha_i - \tilde{\alpha}_i\Big)y_i - 0 - \varepsilon\sum_{i=1}^{m}\Big(\alpha_i + \tilde{\alpha}_i\Big) \\
&= -\frac{1}{2}\Big[\sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)\mathbf{x}_i^T\Big]\Big[\sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)\mathbf{x}_i\Big] + \sum_{i=1}^{m}\Big(\alpha_i - \tilde{\alpha}_i\Big)y_i - \varepsilon\sum_{i=1}^{m}\Big(\alpha_i + \tilde{\alpha}_i\Big) \\
&= -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i - \tilde{\alpha}_i)(\alpha_j - \tilde{\alpha}_j)\mathbf{x}_i^T\mathbf{x}_j + \sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)y_i - \epsilon\sum_{i=1}^{m}(\alpha_i + \tilde{\alpha}_i).
\end{aligned}$$

So the Lagrangian function can be rewritten as

$$\mathcal{L}(\boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}}) = -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i - \tilde{\alpha}_i)(\alpha_j - \tilde{\alpha}_j)\mathbf{x}_i^T\mathbf{x}_j + \sum_{i=1}^{m}(\alpha_i - \tilde{\alpha}_i)y_i - \epsilon\sum_{i=1}^{m}(\alpha_i + \tilde{\alpha}_i),$$

$$s.t. \quad \begin{cases} \sum_{i=1}^{n}(\alpha_i - \tilde{\alpha}_i) = 0, & i = 1, 2, \cdots, m, \\ 0 \le \alpha_i, \tilde{\alpha}_i \le C, & i = 1, 2, \cdots, m. \end{cases}$$

By maximizing $\mathcal{L}(\boldsymbol{\alpha}, \tilde{\boldsymbol{\alpha}})$, we can get the optimal Lagrangian multipliers: $\boldsymbol{\alpha}^* = (\alpha_1^*, \alpha_2^*, \cdots, \alpha_m^*)$, $\tilde{\boldsymbol{\alpha}}^* = (\tilde{\alpha}_1^*, \tilde{\alpha}_2^*, \cdots, \tilde{\alpha}_m^*)$. $\mathbf{w}^*$ and b^* can be derived by the same process as previouly introduced.

Generally, the data may not have a clear linear relationship, so we may use a kernel $\kappa(\cdot, \cdot) : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$. The regression function can be rewritten as

$$f(\mathbf{x}) = \sum_{i=1}^{m}(\alpha_i^* - \tilde{\alpha}_i^*)\kappa(\mathbf{x}, \mathbf{x}_i) + b.$$

Only the support vectors (with $\alpha_i^* - \tilde{\alpha_i}^* \neq 0$) contribute to the prediction.

1.6 LS-SVM

Least Squares Support Vector Machine (LS-SVM) [Suykens and Vandewalle, 1999] is obtained by reformulating the minimization problem as

$$\begin{aligned} &\min \frac{1}{2}\|\mathbf{w}\|_2^2 + \frac{\gamma}{2}\sum_{i=1}^{m} e_i^2, \\ &\text{s.t.} \quad y_i = \mathbf{w}^T\phi(\mathbf{x}_i) + b + e_i, \quad i = 1, 2, \cdots, m. \end{aligned}$$

The Lagrangian is given by

$$\begin{aligned} \mathcal{L}(\mathbf{w}, b, \mathbf{e}, \boldsymbol{\alpha}) &= \frac{1}{2}\|\mathbf{w}\|_2^2 + \frac{\gamma}{2}\sum_{i=1}^{m} e_i^2 + \sum_{i=1}^{m}\alpha_i\Big(\mathbf{w}^T\phi(\mathbf{x}_i) + b + e_i - y_i\Big) \\ &= \frac{1}{2}\mathbf{w}^T\mathbf{w} + \frac{\gamma}{2}\sum_{i=1}^{m} e_i^2 + \mathbf{w}^T\sum_{i=1}^{m}\alpha_i\phi(\mathbf{x}_i) + b\sum_{i=1}^{m}\alpha_i + \sum_{i=1}^{m}\alpha_i e_i - \sum_{i=1}^{m}\alpha_i y_i. \end{aligned}$$

The KKT conditions are

$$\mathbf{w}^T\phi(\mathbf{x}_i) + b + e_i - y_i = 0 \quad \text{(Primal Feasibility)},$$

$$\nabla_{\mathbf{w}}\mathcal{L}(\mathbf{w}, b, \mathbf{e}, \boldsymbol{\alpha}) = \mathbf{w} - \sum_{i=1}^{m}\alpha_i\phi(\mathbf{x}_i) = 0 \quad \text{(Stationarity)},$$

$$\nabla_b\mathcal{L}(\mathbf{w}, b, \mathbf{e}, \boldsymbol{\alpha}) = \sum_{i=1}^{m}\alpha_i = 0 \quad \text{(Stationarity)},$$

$$\nabla_{e_i}\mathcal{L}(\mathbf{w}, b, \mathbf{e}, \boldsymbol{\alpha}) = \gamma e_i - \alpha_i = 0 \quad \text{(Stationarity)},$$

The conditions can be expressed in linear equation system:

$$\begin{bmatrix} 0 & \mathbf{1}^T \\ \mathbf{1} & \mathbf{K} + \gamma^{-1}\mathbf{I}_m \end{bmatrix}\begin{bmatrix} b \\ \boldsymbol{\alpha} \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix}$$

where $\mathbf{y} = (y_1, y_2, \cdots .y_m)^T$, $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \cdots, \alpha_m)^T$; $\mathbf{1} = (1, 1, \cdots, 1)^T \in \mathbb{R}^{m\times 1}$ is the one-vector; $\mathbf{I}_m = \text{diag}(1, 1, \cdots, 1) \in \mathbb{R}^{m\times m}$ is the identity matrix; $\mathbf{K} \in \mathbb{R}^{m\times m}$ is the kernel matrix defined by $K_{ij} = \kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$.

Compared to classical SVM, LS-SVM is faster in computation, since it finds the solution by solving a set of linear equations instead of a convex quadratic programming problem. However, LS-SVM uses L2 loss and thus lacks sparseness, i.e. not several support vectors but the majority of training samples contribute to the prediction.

References

Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995. URL https://link.springer.com/article/10.1007/Bf00994018.

Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999. URL https://link.springer.com/article/10.1023/A:1018628609742.

MIT OpenCourseWare. Mit 15.097 lecture 12: Support vector machines, 2012a. URL https://ocw.mit.edu/courses/15-097-prediction-machine-learning-and-statistics-spring-2012/resources/mit15_097s12_lec12/.

MIT OpenCourseWare. Mit 15.097 lecture 13: Kernels, 2012b. URL https://ocw.mit.edu/courses/15-097-prediction-machine-learning-and-statistics-spring-2012/resources/mit15_097s12_lec13/.

MathWorks. Mathworks - understanding support vector machine regression. URL https://www.mathworks.com/help/stats/understanding-support-vector-machine-regression.html.

Wikipedia. Wikipedia - least-squares support vector machine. URL https://en.wikipedia.org/wiki/Least-squares_support_vector_machine.

2 Appendix

2.1 Karush-Kuhn-Tucker (KKT) Method

Given a constrained optimization problem:

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \quad & f(\mathbf{x}). \\ \text{s.t.} \quad & \mathbf{x} \in \chi = \{g_i(\mathbf{x}) < 0, h_j(\mathbf{x}) = 0\}_{i,j}. \end{aligned}$$

The problem is equivalent to

$$\begin{aligned} \min_{\mathbf{x}\in\mathbb{R}^n} \max_{\boldsymbol{\lambda},\boldsymbol{\mu};\boldsymbol{\lambda}>0} \quad & L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu}) \\ = \min_{\mathbf{x}\in\mathbb{R}^n} \max_{\boldsymbol{\lambda},\boldsymbol{\mu};\boldsymbol{\lambda}>0} & \left\{ f(\mathbf{x}) + \sum_{i=1}^{k} \lambda_i g_i(\mathbf{x}) + \sum_{j=1}^{l} \mu_j h_j(\mathbf{x}) \right\} \end{aligned}$$

where $L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})$ is called the Lagrangian function, and $\boldsymbol{\lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_k) \in \mathbb{R}^k$, $\boldsymbol{\mu} = (\mu_1, \mu_2, \cdots, \mu_l) \in \mathbb{R}^l$ are called KKT multipliers.

If $\mathbf{x}^*$ is an optimal solution, there exists KKT multipliers that satisfy KKT conditions:

- Primal Feasibility: The point $\mathbf{x}^*$ must satisfy all the original constraints:

$$g_i(\mathbf{x}^*) \leq 0, \quad \forall i = 1, \cdots, k,$$

$$h_j(\mathbf{x}^*) = 0, \quad \forall j = 1, \cdots, l;$$

- Dual Feasibility: The Lagrange multipliers with respect to the inequality constraints must be non-negative:

$$\lambda_i \geq 0, \quad \forall i = 1, \cdots, k;$$

- Complementary Slackness: Either the Lagrange multiplier or the corresponding inequality constraint is zero:

$$\lambda_i g_i(\mathbf{x}^*) = 0, \quad \forall i = 1, \ldots, k;$$

- Stationarity: The gradient of the Lagrangian function at $\mathbf{x}^*$ must be zero:

$$\nabla_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}, \boldsymbol{\mu})\Big|_{\mathbf{x}=\mathbf{x}^*} = \nabla f(\mathbf{x}^*) + \sum_{i=1}^{k} \lambda_i \nabla g_i(\mathbf{x}^*) + \sum_{j=1}^{l} \mu_j \nabla h_j(\mathbf{x}^*) = 0.$$

Created on 20250218;
Finished on 20250319;
Proofread on 20251031.