Notes on Support Vector Machines

Tianjian Li

February 2022

1 Question Formulation

Support Vector Machines aim to solve this problem: Given training data

$$D = (x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$$

We want to find a hyperplane which seperates different types of training data.

Any plane can be described with such an equation:

$$w^\top x + b = 0$$

The distance between any sample x and this plane is:

$$r = \frac{|w^\top x + b|}{||w||}$$

Assume that the hyperplane correctly classifies every sample, that is to say

$$w^\top x_i + b \geq +1, \forall y_i = +1$$

and

$$w^\top x_i + b \leq -1, \forall y_i = -1$$

The samples that are the closest to the hyperplane are called **support vectors**. The distance between the positive and negative support vectors are called the **margin**. The margin is

$$\gamma = \frac{2}{||w||}$$

Our goal is to find the hyperplane that maximizes the margin. A mathematical denotation of our problem is

$$\max_{w,b} \frac{2}{||w||}$$
$$s.t.\ y_i(w^\top x_i + b) \geq 1, \forall i$$

Apparently, to maximize the margin, we need to maximize $||w||^{-1}$, this is equivalent to minimizing $||w||^2$, therefore the above equation can be rewritten as

$$\min_{w,b} \frac{1}{2}||w||^2$$
$$s.t.\ y_i(w^\top x_i + b) \geq 1, \forall i$$

2 Derivation

We use the **Lagrange Multiplier Method** to solve this problem. Assigning a multiplier α_i to each constraint, we have

$$L(w, b, \alpha) = \frac{1}{2}||w||^2 + \sum_{i=1}^{m} \alpha_i (1 - y_i(w^T x_i + b))$$

Let the partial derivative of L w.r.t. w and b equal to zero, we have

$$w = \sum_{i=1}^{m} \alpha_i x_i y_i$$

$$0 = \sum \alpha_i y_i$$

Therefore, the problem we need to solve can be reformulated as

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i^{\top} x_j$$

$$s.t. \sum_{i=1}^{m} \alpha_i y_i = 0$$

2.1 Karush-Kuhn-Tucker Conditions

In order for the convex optimization problem to have a solution, the problem must satisfy the **Karush-Kuhn-Tucker Conditions**(KKT Conditions). The KKT conditions for the SVM problem are

$$\alpha_i \geq 0$$

$$y_i f(x_i) - 1 \geq 0$$

$$\alpha_i (y_i f(x_i) - 1) = 0$$

The KKT conditions are satisfied because for any sample (x_i, y_i), either $\alpha_i = 0$ or $y_i f(x_i) = 1$. If $\alpha_i = 0$, then the sample is not on a support vector. If $\alpha_i > 0$, then we have $y_i f(x_i) = 1$. The sample is on a support vector. This property reveals the essence of SVM: the resulting hyperplane only depends on the support vectors.

3 Kernel Function

In the previous sections, we assume that we can find such a hyperplane that correctly classifies every node in the sample space. However this is not always the case. We use a function ϕ to embed our input feature vector x into a higher

dimension.

Let $\phi(x)$ denote the embedded vector. The SVM problem becomes

$$\min_{w,b} \frac{1}{2}||w||^2$$

$$s.t.\ y_i(w^\top \phi(x_i) + b) \geq 1, \forall i$$

and

$$\max_{\alpha} \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j \phi(x_i)^\top \phi(x_j)$$

$$s.t.\ \sum_{i=1}^{m} \alpha_i y_i = 0$$

It is computationally expensive to compute $\phi(x_i)^\top \phi(x_j)$. Therefore we use a function to compute it.

$$\kappa(x_i, x_j) = \phi(x_i)^\top \phi(x_j)$$

The function κ is called the **kernel function**. Popular choices of kernel functions are

1. $\kappa(x_i, x_j) = x_i^\top x_j$
2. $\kappa(x_i, x_j) = (x_i^\top x_j)^d$
3. $\kappa(x_i, x_j) = \exp(-\frac{||x_i - x_j||^2}{2\sigma^2})$
4. $\kappa(x_i, x_j) = \exp(-\frac{||x_i - x_j||}{\sigma}$
5. $\kappa(x_i, x_j) = tanh(\beta x_i^\top x_j + \theta)$

4 Soft Margins

In the previous sections, we assume that there exists such a hyperplane that correctly classifies every sample. However this is not always the case in real life, and even if we do find such a hyperplane, it might be the result of overfitting. To solve this issue, we can find a hyperplane that might incorrectly classifies some of the samples. We introduce **soft margin**.

When all of the samples need to be classified correctly, we have

$$y_i(w^\top x_i + b) \geq 1, \forall i$$

When using a soft margin, we allow some of the samples to be incorrectly classified. The optimization objective becomes

$$\min_{w,b} \frac{1}{2}||w||^2 + C \sum_{i=1}^{m} l_{0/1}(y_i(w^\top x_i + b) - 1)$$

Where C is a constant. When C is sufficiently large, the optimization problem forces all our samples to be correctly classified. When C is small, we allow some of the samples to be incorrectly classified. $l_{0/1}$ is a 0-1 loss function.

$$l_{0/1}(z) = \begin{cases} 1, & \text{if } z < 0. \\ 0, & \text{otherwise.} \end{cases}$$

However, $l_{0/1}$ is not a convex and continuous function. So we use other loss functions to approximate $l_{0/1}$. The most common ones are

1. hinge loss: $l_{hinge} = \max(0, 1 - z)$
2. exponential loss: $l_{\exp} = \exp(-z)$
3. log loss: $l_{\log} = \log(1 + \exp(-z))$

If we use hinge loss, the optimization objective becomes

$$\min_{w,b} \frac{1}{2}||w||^2 + C \sum_{i=1}^{m} \max(0, 1 - y_i(w^\top x_i + b))$$