5. SVM(sparse kernel)

In the previous notes, we've discussed the model based on the **feature** of data(Linear hypothesis) and the **relationship** of data(kernel).

Now, we want to make a model based on geometry attributes: find a hyperplane $\theta^T x + \theta_0 = 0$ that can seperate the data set.

This means that the margin from the data to the hyperplane should be bigger than 0, and the bigger the better.

5.1 Distance from a Point to a Hyperplane

The distance from a point $x \in R^d$ to a hyperplane $\theta^T x + \theta_0 = 0$ is :

$$r = \frac{\theta^T x + \theta_0}{||\theta||}$$

that's because the inner product of two vectors is the product of the lengths of their projection, which means that $\theta^T x$ equals to $distance \times ||\theta||$, where distance is the distance from this x to the hyperplane across the origin and parallel to the current hyperplane. So, we need to subtract the distance bwtween origin and this line, $dist_o$. Like $distance$, $dist_o = \frac{\theta^T 0^d}{||\theta||} = \frac{-\theta_0}{||\theta||}$. As a result, $r = distance - dist_o = \frac{\theta^T x}{||\theta||} - \frac{-\theta_0}{||\theta||} = \frac{\theta^T x + \theta_0}{||\theta||}$.

5.2 SVM

5.2.1 Objective Optimization

In binary classification problem, we usually have the positive data points and negative data points, so we change r to:

$$r = y\frac{\theta^T x + \theta_0}{||\theta||}$$

Then, r will always be greater than 0, if no prediction mistake happens.

While training, we try to make sure that r always greater than 0 and the minimum r represents the really gap between two classes, so we get our optimization problem:

$$max_{\theta,\theta_0} min[y^i \frac{\theta^T x^i + \theta_0}{||\theta||}]$$

Since we can rescaling $\theta \rightarrow k\theta$ and $\theta_0 \rightarrow k\theta_0$ while maintain the margin, we can define a scaling factor such that

$$min[y^i(\theta^T x^i + \theta_0)] = 1$$

Then the optimization problem turns into:

$$
\begin{aligned}
& max_{\theta,\theta_0} \frac{1}{||\theta||} \\
s.t. \quad & \\
& y^i(\theta^T x^i + \theta_0) \geq 1
\end{aligned}
$$

5.2.2 Support Vector

With Lagrange Multipliers, we get:

$$L(\theta, \theta_0, \alpha) = \frac{1}{2}||\theta||^2 - \sum_{i=1}^{m} \alpha^i[(\theta^T x^i + \theta_0)y^i - 1]$$

From that, we can get:

> 1. $\frac{\partial L}{\partial \theta} = 0 \implies \theta = \sum_{i=1}^{m} \alpha^i y^i x^i$
> 2. $\frac{\partial L}{\partial \theta_0} = 0 \implies \sum_{i=1}^{m} \alpha^i y^i = 0$
> 3. KKT $\implies \alpha^i[y^i(\theta^T x^i + \theta_0) - 1] = 0$

From the third result, we know that for i such that $\alpha^i > 0$, $y^i(\theta^T x^i + \theta_0) - 1 = 0$. Thest are points that right on the decision boundary, named **support vector**. For the other points $y^i(\theta^T x^i + \theta_0) - 1 > 0$, we can get $\alpha^i = 0$, which means they are useless.

5.3 Softmargin SVM

The above discussion is based on datasets that can be perfectly seperated. But the truth is that most datasets cannot meet that.

So, now we allow some points can be misclassified, and we define ξ^i is the distance of a point x^i from its margin. The margin is a hyperplane for this class, $\theta^T x + \theta_0 = 1 \; or \; -1$.

Now, the optimization problem changed:

$$
\begin{aligned}
& min_{\theta,\theta_0} \frac{1}{2}||\theta||^2 + C\sum_{i=1}^{m} \xi^i \\
s.t. \quad & \\
& y^i(\theta^T x^i + \theta_0) \geq 1 - \xi^i \\
& \xi^i \geq 0
\end{aligned}
$$

Let's try to avoid ξ, focus on the constraints first.

Define $h(x) = \theta^T x + \theta_0$, we get $y^i h(x^i) \geq 1 - \xi^i$. That's $\xi^i \geq 1 - y^i h(x^i)$. Combined with $\xi^i \geq 0$, we now have $\xi^i = max(0, 1 - y^i h(x^i))$.

Now, the problem is :

$$min_{\theta,\theta_0} \frac{1}{2}||\theta||^2 + C\sum_{i=1}^{m} max(0, 1 - y^i h(x^i))$$

As usual, we can use gradient descent here to get θ.

$$J(\theta) = \frac{1}{2}||\theta||^2 + C\sum_{i=1}^{m} max(0, 1 - y^i h(x^i))$$

$$\frac{\partial J}{\partial \theta} = -y^i x^i \quad (y^i h(x^i) < 1)$$

5.4 Conclusion

I once was confused about this SVM algorithm.

Because it says that it only needs support vectors to define a classifier, but how can I get support vector without having the decision boudary, also named θ. Since support vectors are the points that stands on the margin.

That means, after all, we have to use all the data to find θ, and then we can decide which points are support vectors.

Explanation:

First of all, as shown in the formular in gradient descent, the points such that $y^i h(x^i) \geq 1$ won't make any change on θ.

Secondly, θ itself represents the support vector, since it can represent the margin. That means, it's correct that you're finding θ, but it's not the hyperplane that you're finding, it's the support vectors.

I'm not sure I was all right on this, I just record my thoughs. Please correct me if anything goes wrong.