AI534 WA2

Yu-Hao. Shih.

shihyuh@oregonstate.edu.

93450399

1. Cubic Kernal:

$K(X_i, X_j) = (X_i \cdot X_j + 1)^2 \quad X = (X_1, X_2) \in R^2$

$\Phi(X) = (X_1^2, X_2^2, \sqrt{2}X_1X_2, \sqrt{2}X_1, \sqrt{2}X_2, 1)$

$K(X_i, X_j) = (X_i \cdot X_j + 1)^3$. let $X_i = (X_{i1}, X_{i2}), X_j = (X_{j1}, X_{j2})$

$\Rightarrow X_i \cdot X_j = X_{i1}X_{j1} + X_{i2}X_{j2}$

so $K(X_i, X_j) = (X_{i1}X_{j1} + X_{i2}X_{j2} + 1)^3$

$(a+b+c)^3 = a^3 + b^3 + c^3 + 3a^2b + 3ab^2 + 3a^2c + 3ac^2 + 3b^2c + 3bc^2 + 6abc$

$a = X_{i1}X_{j1}, \quad b = X_{i2}X_{j2}, \quad c = 1$

$$\Rightarrow (X_{i1}X_{j1} + X_{i2}X_{j2} + 1)^3 = X_{i1}^3X_{j1}^3 + X_{i2}^3X_{j2}^3 + 1 + 3X_{i1}^2X_{j1}^2X_{i2}X_{j2} + 3X_{i1}X_{j1}X_{i2}^2X_{j2}^2 + 3X_{i1}^2X_{j1}^2 + 3X_{i2}^2X_{j2}^2 + 3X_{i1}X_{j2} + 3X_{i2}X_{j2} + 6X_{i1}X_{j1}X_{i2}X_{j2}$$

recombination to high degree space projection.

$X_1^3, X_2^3, \sqrt{3}X_1^2X_2, \sqrt{3}X_1X_2^2, \sqrt{3}X_1^2, \sqrt{3}X_2^2, \sqrt{6}X_1X_2, \sqrt{3}X_1, \sqrt{3}X_2, 1$

$\Phi(X) = (X_1^3, X_2^3, \sqrt{3}X_1^2X_2, \sqrt{3}X_1X_2^2, \sqrt{3}X_1^2, \sqrt{3}X_2^2, \sqrt{6}X_1X_2, \sqrt{3}X_1, \sqrt{3}X_2, 1)$

2. Kernal or not:

In Kernal function, for two K_1 and K_2, there linear combination and positive times are still a kernal function. So, when K_1 and K_2 are kernal functions, and C_1, C_2 are positive, there linear combination $K(x, z) = C_1k_1(x, z) + C_2k_2(x, z)$ is still a kernal function.

Because we suppose k_1 and k_2 are kernals. we know:

$$k_1(x, z) = \phi_1(x) \cdot \phi_1(z)$$

$$k_2(x. z) = \phi_2(x) \cdot \phi_2(z)$$

Redefine $\phi(x)$ as weights combination of $\phi_1(x)$ and $\phi_2(x)$

$$\Rightarrow \phi(x) = (\sqrt{C_1}\, \phi_1(x), \sqrt{C_2}\, \phi_2(x))$$

Valid features projection: Is $\phi(x) \cdot \phi(z) = k(x, z)$?

$$\phi(x) \cdot \phi(z) = (\sqrt{C_1}\, \phi_1(x), \sqrt{C_2}\, \phi_2(x)) \cdot (\sqrt{C_1}\, \phi_1(z), \sqrt{C_2}\, \phi_2(z))$$

$$\Rightarrow \phi(x) \cdot \phi(z) = C_1(\phi_1(x) \cdot \phi_1(z)) + C_2(\phi_2(x) \cdot \phi_2(z))$$

We know $\phi_1(x) \cdot \phi_1(z) = k_1(x, z)$ and $\phi_2(x) \cdot \phi_2(z) = k_2(x, z)$

$$\Rightarrow \phi(x) \cdot \phi(z) = C_1 k_1(x, z) + C_2 k_2(x, z) = k(x. z)$$

Hence. we know $k(x. z) = C_1 k_1(x. z) + C_2 k_2(x. z)$ is a kernal function.

the features projection is $\phi(x) = (\sqrt{C_1}\, \phi_1(x), \sqrt{C_2}\, \phi_2(x))$

3. Kernalizing Logistic Regression:

(a).

Set our weight vector w as linear combination of training samples.

$w = \sum_{i=1}^{N} \alpha_i x_i$ (α_i are weights consistancy) (for logistic regression)

$\Rightarrow$ brings to loss function: $L(w) = -\sum_{i=1}^{N} [y_i \log(\sigma(w^T x_i)) + (1-y_i) \log(1-\sigma(w^T x_i))]$

because w can represent the sum of weights. so when we brings in $w = \sum_{j=1}^{N} \alpha_j x_j$, then we can express in kernal space.

Dot form of kernal logistic regression, than we can only calculate the dot x_i and x_j. Then we can express as kernal function without specific high degree features projection. Hence, the solution of logistic regression can represent as the sum of sample's weights.

(b).

First we have to initial a coefficient vector α, and the same length with training samples N. and set all $\alpha_i = 0$.

And update the new kernal gradient function:

$$w \leftarrow w + \gamma \left(y_i - \sigma(w^T x_i) \right) x_i$$

Let w = weight sum of samples $\Rightarrow w = \sum_{j=1}^{N} \alpha_j x_j$,

and replace the inner dot $w^T x_i$ with $\sum_{j=1}^{N} \alpha_j K(x_j, x_i)$, $K(x_j, x_i)$ is a kernal function, (inner dot of features space in high degree)

Than update α vector.

$$\alpha_i \leftarrow \alpha_i + \gamma \left(y_i - \sigma\left(\sum_{j=1}^{N} \alpha_j K(x_j, x_i) \right) \right)$$

Thus. We only have to update α and we don't have to calculate w. which makes the calculate process happen in kernal space.

Repeating these processes until converge.

4. Hard margin SVM:

Definition: those point who are most close to the decision boundary. (SVM)

(a).

For positive (blue points): (1,1) (1,0.5) are SVM.
and for negative (red points): (2,1) (2,1.5) are SVM.
The decision boundary should be in the middle of two classes. (blue and red. positive and negative) which is $x_1 = 1.5$, and this decision boundary will seperate the blue points and red points.

Interval boundary located on the both side of decision boundary. and pass the SVM via (1,1). (1,0.5), (2,1) and (2,1.5), the interval boundary are $x_1 = 1$ and $x_1 = 2$.

(b)

Since the decision boundary is $x_1 = 1.5$, if we assume only W_1 has a value, and $W_2 = 0$, the decision boundary can be.

$$\Rightarrow W_1 X_1 + b = 0$$

Brings the support vectors (1,1) and (1,0.5) to solve positive class

$$\Rightarrow W_1 \cdot 1 + b = 1$$

bring (2, 1) and (2, 1.5) to solve negative class.

$\Rightarrow w_1 \cdot 2 + b = -1.$

$\Rightarrow \begin{cases} w_1 + b = 1 \\ 2w_1 + b = -1 \end{cases} \quad \Rightarrow \begin{aligned} w_1 &= -2 \\ b &= 3 \end{aligned}$

why assume $w_2 = 0$ is because the decision boundary is vertical

if $w_2 x_2 + b = 0$, it's a horizontal

5. L_2 SVM.

SVM with L_2 panelty term.

(a).

Proof by contradiction.

Assume $\xi_i < 0$. Suppose in the optimal solution (ξ^*), there exists some $\xi_i < 0$. However, because ξ_i is the panelty term for misclassification, if $\xi_i < 0$, then it would reduce the penalty, making the objective smaller, which contradicts the meaning of optimizing. because ξ_i should provide a non-negative penalty for misclassification. Thus, the assumption is incorrect, ξ_i must be greater than or equal to "0".

Hence, this shows that even without the constraint $\xi_i \geq 0$, the optimal solution will still satisfy $\xi_i \geq 0$.

(b).

After removing the $\xi_i \geq 0$ constraint, there is only one optimization problem remain:

$$y_i(W^T x_i + b) \geq 1 - \xi_i$$

Lagrangian Function:

$$\Rightarrow L(w, b, \xi, \alpha) = \frac{1}{2} W^T W + C \sum_{i=1}^{N} \xi_i^2 - \sum_{i=1}^{N} \alpha_i (y_i (W^T x_i + b) - 1 + \xi_i)$$

$\alpha_i \geq 0$ are Lagrangian multipliers associated with the constraints.

(c). partial $L(w, b, \xi, \alpha)$. and set as zero.

$$\frac{\partial}{\partial w} \Rightarrow W - \sum_{i=1}^{N} \alpha_i y_i x_i = 0 \Rightarrow W = \sum_{i=1}^{N} \alpha_i y_i x_i$$

$$\frac{\partial}{\partial b} \Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0$$

$$\frac{\partial}{\partial \xi_i} \Rightarrow 2C \xi_i - \alpha_i = 0, \quad \xi_i = \frac{\alpha_i}{2C}$$

brings them into Lagrangian function:

$$\frac{1}{2} W^T W = \frac{1}{2} \left(\sum_{i=1}^{N} \alpha_i y_i x_i\right)^T \left(\sum_{i=1}^{N} \alpha_i y_i x_i\right)$$

$$= \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j x_i^T x_j$$

Kernal: $\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \cdot K(x_i, x_j)$

$$C \sum_{i=1}^{N} \xi_i^2 = C \sum_{i=1}^{N} \left(\frac{\alpha_i}{2C}\right)^2 = \frac{1}{4C} \sum_{i=1}^{N} \alpha_i^2$$

$$\Rightarrow L(w, b, \xi, \alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j K(x_i, x_j) - \frac{1}{4C} \sum_{i=1}^{N} \alpha_i^2$$

[$y_i(W^T x_i + b) = 1$. so it was eliminated]

We got: $\max\limits_{\alpha} \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j \cdot K(x_1, x_2) - \frac{1}{4C} \sum_{i=1}^{N} \alpha_i^2$

$$\left(\sum_{i=1}^{N} \alpha_i y_i = 0, \quad \alpha_i \geq 0\right)$$