Homework - 9
COEN 240-Machine Learning

Pujitha Kallu
ID : W1653660
pkallu@scu.edu

Problem 1

Find the solution (x^*, y^*) to the following problem.

$$\text{optimize } xy$$
$$\text{subject to } x + y = 10$$

Sol:

Solution 1 :

Standard form :⇒

optimize xy

subject to $x+y-10=0$

Lagrangain : $L(x,y,\beta)$

$= xy + \beta(x+y-10)$

Partial derivative :⇒

$$\nabla_x L(x,y,\beta) = y+\beta = 0 \rightarrow \text{Eq } ①$$
$$\nabla_y L(x,y,\beta) = x+\beta = 0 \rightarrow \text{Eq } ②$$
$$\nabla_\beta L(x,y,\beta) = x+y-10 = 0 \rightarrow \text{Eq } ③$$

Now, we can solve for x,y :⇒

from Eq① & Eq② :

$\Rightarrow \beta = -y$ — ① & $\Rightarrow \beta = -x$ — ②

now Equating ① & ②

we get $-y = -x$

$\Rightarrow y = x.$

So from Eq ③

$x+y-10=0$

$\Rightarrow 2x = 10$

$x=5$ or $y=5$

$$\boxed{\therefore \quad x^* = 5 \quad \& \quad y^* = 5}$$

:

Problem 2

The SVM optimization can be defined by the primal form:

$$\min_w \frac{1}{2}\|\mathbf{w}\|^2$$

$$\text{subject to } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1, \qquad i = 1, \ldots, N$$

Or by its the dual form:

$$\max_\alpha J(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \alpha_i\alpha_j y_i y_j \left(\mathbf{x}_i^T\mathbf{x}_j\right)$$

$$\text{subject to } \alpha_i \geq 0, i = 1, \ldots N \text{ and } \sum_{i=1}^{N} \alpha_i y_i = 0$$

What is the Lagrangian function $L(\mathbf{w}, b, \alpha)$ evaluated at $\mathbf{w}$ that minimizes that function?

Note this is the objective function $J(\alpha)$.

Hints:

1. Write the primal problem in standard form
2. Form the Lagrangian function $L\ (\mathbf{w}, b, \alpha)$
3. Find $\mathbf{w}$ and b that minimize $L\ (\mathbf{w}, b, \alpha)$
4. Plug the results back into $L\ (\mathbf{w}, b, \alpha)$

Sol:

Solution 2 :

tep 1. First we write the primal problem in standard form:

$$\min_{\omega} 1/2 \, \|w\|^2 \text{ subject to } g_i(\omega) = -y_i \, (\omega^T x_i + b) + 1 \leq 0$$

tep 2. Second we form the Lagrangian function:

$$L(\omega, b, \alpha) = 1/2 \, \|w\|^2 + \sum_{i=1}^{N} \alpha_i \left[-y_i \, (\omega^T x_i + b) + 1\right] \rightarrow Eq(1$$

Here there is no β_i because there are no quality constraints.

ep 3. Now we find w and b that minimize $L(\omega, b, \alpha)$

$$\nabla_\omega L(w, b, \alpha) = w - \sum_{i=1}^{N} \alpha_i y_i x_i = 0$$

$$w = \sum_{i=1}^{N} \alpha_i y_i x_i \longrightarrow Eq(2)$$

Now $\nabla_b L(\omega, b, \alpha) = \sum_{i=1}^{N} \alpha_i y_i = 0 \longrightarrow Eq(3)$

p 4: Now we plug in Eq (2) & Eq(3) into Eq(1)

$$L(\omega, b, \alpha) = 1/2 \, \omega^T \omega + \sum_{i=1}^{N} \alpha_i \left[-y_i \, (\omega^T x_i + b) + 1\right]$$

$$L(\omega, b, \alpha) = \frac{1}{2} \sum_{i=1}^{N} \alpha_i y_i x_i^T \cdot \sum_{j=1}^{N} \alpha_j y_j x_j$$

$$L(\omega, b, \alpha) = \frac{1}{2} \sum_{i=1}^{N} \alpha_i y_j x_i^T \cdot \sum_{j=1}^{N} \alpha_j y_j x_j + \sum_{i=1}^{N} \alpha_i \left[-y_i \left(\left(\sum_{j=1}^{N} \alpha_j y_j x_j^T \right) x_i + b \right) + 1 \right]$$

$$\Rightarrow \quad L(\omega, b, \alpha) = 1/2 \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j)$$

$$- \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_j^T x_i) - \sum_{i=1}^{N} \alpha_i y_i b + \sum_{i=1}^{N} \alpha_i$$

But,

$$\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_j^T x_i) = \sum_{i=1}^{N} \sum_{j=1} \alpha_i \alpha_j \cdot y_i y_j (x_i^T x_j)$$

And from Eq (3) $\Rightarrow \sum_{i=1}^{N} \alpha_i y_i = 0$

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

Therefore,

$$J(a) = L(\omega, b, \alpha)$$

$$\boxed{J(\alpha) = \sum_{i=1}^{N} \alpha_i - 1/2 \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j y_i y_j (x_i^T x_j)}$$

References:

1. Class notes:Support vector machines (SVMs)
2. https://www.user.tu-berlin.de/mtoussai/teaching/13-Optimization/03-constrainedOpt.pdf
3. https://stats.stackexchange.com/questions/171676/models-for-machine-learning-constrained-optimization
4. https://georgian.io/constrained-optimization-how-to-do-more-with-less/