Date __/__/____
Page 1

11/8/2021

Summary: Simplified Sequential Minimal Optimization

The SMO algorithm is an optimization algorithm used for training support vector machines.

The SMO algorithm tries to optimize the SVM function of the form:

$$\max_{\alpha} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} y_i y_j \alpha_i \alpha_j \langle x_i, x_j \rangle \rightarrow (1)$$

S.T:

$$0 \leq \alpha_i \leq C, \; i = 1, \ldots, m \rightarrow (2)$$

$$\sum_{i=1}^{m} \alpha_i y_i = 0 \rightarrow (3)$$

This is the dual form of a regularized SVM classifier ($f(x) = w^T x + b$).

The point of convergence can be checked using the KKT conditions:

$$\alpha_i = 0 \Rightarrow y_i (w^T x_i + b) \geq 1 \rightarrow (4)$$

(correctly classified points)

$$\alpha_i = C \Rightarrow y_i (w^T x_i + b) \leq 1 \rightarrow (5)$$

(points inside margin)

$$0 < \alpha_i < C \Leftrightarrow y_i (w^T x_i + b) = 1 \rightarrow (6)$$

(on margin)

Working of the SMO algorithm :

The algorithm selects two parameters α_i and α_j and optimizes for both these values. It then adjusts the parameter b for the new α's. This processes is repeated until all α's converge.

Step 1: Selecting α's

- The simplified SMO iterates over all $\alpha_i = 1, \ldots m$.
- If α_i doesn't fulfill KKT conditions, it selects random α_j from the remaining $m-1$ α's and tries to optimize both α_i and α_j jointly.
- If the α's havent changed after a few iterations, the algorithm terminates.

Step 2: Optimizing α_i and α_j

- We define constraints on α_i and α_j then ~~so to~~ solve the constrained maximization problem.

- We define bounds L and H on α_j: $L \le \alpha_j \le H$, which must hold for α_j to satisfy $0 \le \alpha_j \le C$. Those being:

(i) If $y_i \neq y_j$ (different classes),

$$L = \max(0, \alpha_j - \alpha_i), \quad H = \min(C, C + \alpha_j - \alpha_i) \rightarrow (7)$$

(ii) If $y_i = y_j$ (same class),

$$L = \max(0, \alpha_i + \alpha_j - C), \quad H = \min(C, \alpha_i + \alpha_j) \rightarrow (8)$$

- Now we find α_j that maximizes the function,

$$\alpha_j = \alpha_j - \frac{y_j (E_i - E_j)}{\eta} \rightarrow (9)$$

Where, $E_k = f(x_k) - y_k \rightarrow (10)$

$$\eta = 2\langle x_i, x_j \rangle - \langle x_i, x_i \rangle - \langle x_j, x_j \rangle \rightarrow (11)$$

Here E_k is the error between SVM prediction and true label.

- We now clip the value of α_j if it is lower than L or greater than H.

$$\alpha_j = \begin{cases} H & \text{if } \alpha_j > H \\ \alpha_j & \text{if } L \leq \alpha_j \leq H \\ L & \text{if } \alpha_j < L \end{cases}$$

- After having solved for α_j, we compute α_i

$$\alpha_i = \alpha_i + y_i y_j (\underbrace{\alpha_j^{old}}_{\text{value of } \alpha_j \text{ before optimization.}} - \alpha_j)$$

Step 3: Computing threshold b

- Now that we have optimal α_i and α_j, we select b such that KKT conditions are satisfied for i^{th} and j^{th} examples.

- Now, if

(i) After optimization, $\alpha_i < 0$ or $\alpha_i \geq C$,

then b_1 is valid

$$b_1 = b - E_i - y_i(\alpha_i - \alpha_i^{old})\langle x_i, x_i\rangle - y_j(\alpha_j - \alpha_j^{old})\langle x_i, x_j\rangle$$

(ii) After optimization, $0 < \alpha_j < C$,

then b_2 is valid.

$$b_2 = b - E_j - y_i(\alpha_i - \alpha_i^{old})\langle x_i, x_j\rangle - y_j(\alpha_j - \alpha_j^{old})\langle x_j, x_j\rangle$$

(iii) If both α_i and α_j are within bounds 0 and C, then they will be equal.

(iv) If both α's are exactly at the bounds then all thresholds between b_1 and b_2 satisfy the KKT conditions.

We can select $b = \frac{b_1 + b_2}{2}$ here.

- Therefore, we have a final equation for b :-

$$b = \begin{cases} b_1 & \text{if } 0 < \alpha_i < C \\ b_2 & \text{if } 0 < \alpha_j < C \\ (b_1+b_2)/2 & \text{otherwise} \end{cases}$$

This completes the SMO algorithm