Notes on Support Vector Machines, COMP24111

Tingting Mu
School of Computer Science
University of Manchester
Manchester M13 9PL, UK

Editor: NA

We start from introducing the notations to be used in this notes. We denote a set of training samples by \{ \((x_1,y_1),(x_2,y_2),\ldots,(x_N,y_N)\) \}. The column vector \(x \in \mathbb{R}^d\) denotes the \(d\)-dimensional feature vector of the \(i\)-th training sample, which corresponds to a data point in a \(d\)-dimensional space. The scalar \(y_i \in \{-1,+1\}\) denotes the class label of the \(i\)-th training sample, for which +1 represents the positive class while -1 the negative class.\(^1\)

1. Support Vector Machines

A support vector machine (SVM) classifier (Cortes and Vapnik, 1995; Cristianini and Shawe-Taylor, 2000) finds a hyperplane to separate the two classes of data points with the widest separation margin. To find this separating hyperplane means to decide its direction \(w \in \mathbb{R}^d\) and its bias \(b \in \mathbb{R}\), such that \(w^T x + b = 0\). The separation margin is the region bounded by the two parallel hyperplanes \(w^T x + b = +1\) and \(w^T x + b = -1\), and is

\(^1\) We focus on the binary classification problem.
computed by:

\[
\text{margin} = 2\rho = 2 \frac{2}{\|w\|_2} = \frac{2}{\sqrt{w^T w}}.
\]

Figure 1 illustrates a separating hyperplane, as well as the two parallel hyperplanes associated with it, and its separation margin.

In addition to margin maximisation, the SVM classifier attempts to prevent the two classes of data points from falling into the margin. To achieve this, it forces \(w^T x + b \geq 1\) for a sample \((x, +1)\) from the positive class, while \(w^T x + b \leq 1\) for a sample \((x, -1)\) from the negative class. This is equivalent to forcing \(y (w^T x + b) \geq 1\) for any given sample \((x, y)\) where \(y \in \{-1, +1\}\).

1.1 Hard Margin SVM

To maximise the separation margin in Eq. (1) is equivalent to minimising the term \(w^T w\). To enforce all the training samples stay outside the margin is equivalent to setting the constraints of \(y_i (w^T x_i + b) \geq 1\) for \(i = 1, 2, \ldots, N\). Together, this gives the following constrained optimisation problem to solve:

\[
\min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \frac{1}{2} w^T w, \quad \text{subject to } y_i (w^T x_i + b) \geq 1, \text{ for } i = 1, 2, \ldots, N.
\]

Eq. (2) is referred to as the objective function of the optimisation problem. The purpose of introducing the scaling factor \(\frac{1}{2}\) in Eq. (2) is to have a neat gradient form (\(w\) instead of \(2w\)). The SVM training refers to the process of finding the optimal values of \(w\) and \(b\). The solution gives us the hard margin SVM classifier, which is also called the optimal margin classifier.

1.2 1-Norm Soft Margin SVM

When processing real-world data, we often need to deal with non-separable data patterns (see Figure 2 for example), for which it is impossible to find a hyperplane to separate perfectly the two classes. In this situation, it is helpful to allow the margin constraints in Eq. (3) to be violated.

We use the slack variable \(\xi_i \geq 0\) to measure the deviation from the ideal situation \(y_i (w^T x_i + b) \geq 1\) for the \(i\)-th training sample. A set of more relaxed constraints are used instead:

\[
y_i (w^T x_i + b) \geq 1 - \xi_i, \text{ for } i = 1, 2, \ldots, N.
\]

When \(0 < \xi_i < 1\), the corresponding sample point is allowed to fall within the margin region, but still has to be in the right side. When \(\xi_i = 1\), the corresponding point is allowed to stay on the decision boundary. When \(\xi_i > 1\), the point is allowed to stay in the wrong side of the decision boundary.

Although we relax the constraints to allow Eq. (3) to be violated, the violation, which is measured by the strength of \(\xi_i\) (e.g., \(|\xi_i|\)), should be small to maintain good accuracy. For instance, it is not wise to build a classifier that allows most training samples to fall in the wrong side of the separating hyperplane. Therefore, we minimise the term \(\sum_{i=1}^{N} |\xi_i|\), which is
equal to $\sum_{i=1}^{N} \xi_i$ since all the slack variables are positive numbers, together with the margin term $\frac{1}{2} w^T w$. The modified constrained optimisation problem becomes

$$\min_{w \in \mathbb{R}^d, b \in \mathbb{R}} \frac{1}{2} w^T w + C \sum_{i=1}^{N} \xi_i,$$

subject to

$$y_i \left( w^T x_i + b \right) \geq 1 - \xi_i, \text{ for } i = 1, 2, \ldots, N, \quad (5)$$

$$\xi_i \geq 0, \text{ for } i = 1, 2, \ldots, N, \quad (7)$$

where $C > 0$ is called the regularisation parameter. Let the column vector $\xi = [\xi_1, \xi_2, \ldots, \xi_N]^T$ store the slack variables and we refer to it as a slack vector. The second term in the objective function in Eq. (5) can be expressed via $l_1$-norm $C ||\xi||_1$. This modified SVM is subsequently named as the 1-norm soft margin SVM, and is often referred to as the $l_1$-SVM for simplification.

![Figure 2: Illustration of an case of non-separable data patterns.](image)

### 1.3 Support Vectors

The training samples that (1) distribute along one of the two parallel hyperplanes, or (2) fall within the margin, or (3) stay in the wrong side of the separating hyperplane are more challenging to classify. They contribute more significantly to the determination of the direction and position of the separating hyperplane. They are called support vectors.

The other samples that are not only stay in the right side of the separating hyperplane but also stay outside the margin are called non-support vectors, and they affect less the position of the separating hyperplane.

### 2. Dual Problem of SVM (Optional Reading)

An SVM classifier can be trained by solving a quadratic programming (QP) problem. This QP problem is called the dual problem of an SVM. A QP problem is a special type of optimisation problem, and there exist many sophisticated approaches for solving it. Its
objective function is a quadratic function of multiple input variables and its constraint functions are linear functions of these variables. We will explain in this section how to derive this dual problem by using the method of Lagrange multipliers.

2.1 Lagrange Duality

We start from introducing some knowledge on how to solve a constrained optimisation problem using the method of Lagrange multipliers. A general way to describe a constrained optimisation problem is

$$\min_{x \in \mathbb{R}^n} f(x),$$

subject to

$$g_i(x) \leq 0, \text{ for } i = 1, 2, \ldots, K,$$

$$h_i(x) = 0, \text{ for } i = 1, 2, \ldots, L.$$  

The above expression means that we would like to find the optimal value of the vector $x$ so that the value of the function $f(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ is minimised, meanwhile the optimal $x$ has to satisfy a total of $K$ different inequality constraints $\{g_i(x) \leq 0\}_{i=1}^K$ and a total of $L$ different equality constraints $\{h_i(x) = 0\}_{i=1}^L$. Here, $g_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ and $h_i(x) : \mathbb{R}^n \rightarrow \mathbb{R}$ are constraint functions, which take the same input variables as $f(x)$.

An effective way for solving the constrained optimisation problem in Eqs. (8)-(10) is the method of Lagrange multipliers. It first constructs a Lagrangian function by adding the constraint functions to the objective function, resulting in

$$L(x, \{\lambda_i\}_{i=1}^K, \{\beta_i\}_{i=1}^L) = f(x) + \sum_{i=1}^K \lambda_i g_i(x) + \sum_{i=1}^L \beta_i h_i(x),$$  

where we call $\{\lambda_i\}_{i=1}^K$ and $\{\beta_i\}_{i=1}^L$ Lagrange multipliers. Storing the Lagrange multipliers in two column vectors such that $\lambda = [\lambda_1, \lambda_2, \ldots, \lambda_K]^T$ and $\beta = [\beta_1, \beta_2, \ldots, \beta_L]^T$, the Lagrangian function is denoted by $L(x, \lambda, \beta)$.

The original constrained optimisation problem in Eqs. (8)-(10) is called the primal problem. To assist solving it, a dual problem is imposed using the above Lagrangian function. It is given as

$$\max_{\lambda \in \mathbb{R}^K, \beta \in \mathbb{R}^L} O(\lambda, \beta),$$

subject to

$$\lambda_i \geq 0, \text{ for } i = 1, 2, \ldots, K,$$

where

$$O(\lambda, \beta) = \min_{x \in \mathbb{R}^n} L(x, \lambda, \beta).$$  

Under certain assumptions\(^2\) on the objective and constraint functions $f$, $\{g_i\}_{i=1}^K$ and $\{h_i\}_{i=1}^L$, there must exist a setting of $x$ and $\lambda$ and $\beta$, denoted by $x^*$ and $\lambda^*$ and $\beta^*$, so that $x^*$ is the optimal solution of the primal problem in Eqs. (8)-(10), and $\{\lambda^*, \beta^*\}$ is the optimal solution of the dual problem.

\(^2\) We do not discuss details on the assumptions. It is proven that the objective and constraint functions of the SVM problems satisfy these assumptions.
optimal solution of the dual problem in Eqs. (12)- Eq. (13), also \(x^*\) is the optimal solution of Eq. (14). This means \(x^*\) and \(\lambda^*\) and \(\beta^*\) need to satisfy the following optimal conditions:

\[
\frac{\partial}{\partial x_i} L(x^*, \lambda^*, \beta^*) = 0, \text{ for } i = 1, 2, \ldots, N, \quad \text{(stationarity), (15)}
\]

\[
h_i(x^*) = 0, \text{ for } i = 1, 2, \ldots, L, \quad \text{(primal feasibility), (16)}
\]

\[
g_i(x^*) \leq 0, \text{ for } i = 1, 2, \ldots, K, \quad \text{(primal feasibility), (17)}
\]

\[
\lambda_i^* \geq 0, \text{ for } i = 1, 2, \ldots, K, \quad \text{(dual feasibility), (18)}
\]

\[
\lambda_i^* g_i(x^*) = 0, \text{ for } i = 1, 2, \ldots, K, \quad \text{(complementary slackness). (19)}
\]

These conditions are called the Karush-Kuhn-Tucker (KKT) conditions, where Eq. (15) is known as the stationarity condition, Eqs. (16) and (17) the primal feasibility, Eq. (18) the dual feasibility, and Eq. (19) the complementary slackness. If you can find some \(x\), \(\lambda\) and \(\beta\) that satisfy the KKT conditions, then these must be solutions to the primal and dual problems.

To summarise, to solve a constrained optimisation problem using the method of Lagrange multipliers, its Lagrangian function is firstly constructed based on Eq. (11). Then, the dual problem is constructed based on Eqs. (12)-(14). During the optimisation, the KKT conditions are examined.

### 2.2 Dual Problem of Hard Margin SVM

#### 2.2.1 Derive the Dual Problem

We derive the dual problem for the hard margin SVM in Eqs. (2)-(3) as a practice. The variables to be optimised for the hard margin SVM are stored in the vector \([w, b]\), constituting a \((d + 1)\)-dimensional vector. The objective function is \(\frac{1}{2}w^Tw\). There are a total of \(K = N\) inequality constraint functions, each formed for a training sample, given as

\[
1 - y_i (w^T x_i + b) \leq 0. \quad (20)
\]

There is no equality constraint in the SVM problem. You could also treat this as setting all the equality constraint functions as zeros: \(h_i(x) = 0\). Following Eq. (11), we define the Lagrangian function as

\[
L(w, b, \{\lambda_i\}_{i=1}^N) = \frac{1}{2}w^Tw + \sum_{i=1}^N \lambda_i \left[1 - y_i (w^T x_i + b)\right]. \quad (21)
\]

Next, we show how to construct the dual problem. We first compute the objective function \(O(\{\lambda_i\}_{i=1}^K)\) for the dual problem according to Eq. (14). We have learned that the minimum of a function is reached when its gradient reaches zero. Therefore, to minimise the Lagrangian function with respect to \(w\) and \(b\), we first calculate the gradient. The Lagrangian function can be re-written as

\[
L(w, b, \{\lambda_i\}_{i=1}^N) = \frac{1}{2}w^Tw - w^T \left(\sum_{i=1}^N \lambda_i y_i x_i\right) + \sum_{i=1}^N \lambda_i (1 - by_i). \quad (22)
\]
It includes one quadratic function $\frac{1}{2}w^Tw$, one linear function $w^T(\sum_{i=1}^{N}\lambda_iz_ix_i)$, and one constant function of $w$. Applying the gradient calculation rules for linear and quadratic functions as shown in the maths notes, we have

$$\frac{\partial}{\partial w}L(w, b, \{\lambda_i\}_{i=1}^{N}) = w - \sum_{i=1}^{N}\lambda_iz_ix_i. \quad (23)$$

Since the only term in the Lagrangian function that is related to $b$ is $-b\sum_{i=1}^{N}\lambda_iy_i$, we have

$$\frac{\partial}{\partial b}L(w, b, \{\lambda_i\}_{i=1}^{N}) = -\sum_{i=1}^{N}\lambda_iy_i. \quad (24)$$

Set the gradient to zero so that $\frac{\partial}{\partial w}L(w, b, \{\lambda_i\}_{i=1}^{N}) = 0$ and $\frac{\partial}{\partial b}L(w, b, \{\lambda_i\}_{i=1}^{N}) = 0$. This corresponds to the stationarity condition in Eq. (15), and it results in

$$w = \sum_{i=1}^{N}\lambda_iy_ix_i, \quad (25)$$

$$\sum_{i=1}^{N}\lambda_iy_i = 0. \quad (26)$$

Replacing $w$ with $\sum_{i=1}^{N}\lambda_iy_ix_i$ and replacing $\sum_{i=1}^{N}\lambda_iy_i$ with zero in the Lagrangian function in Eq. (21), the optimised Lagrangian function with respect to $w$ is obtained by

$$O(\{\lambda_i\}_{i=1}^{N}) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_jy_iy_jx_i^Tx_j - \sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_jy_iy_jx_j^Tx_i + \sum_{i=1}^{N}\lambda_i$$

$$= -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_jy_iy_jx_i^Tx_j + \sum_{i=1}^{N}\lambda_i. \quad (27)$$

This is a function of the Lagrange multipliers, which is the objective function of the dual problem of the hard margin SVM.

According to the dual feasibility condition in Eq. (18), $\lambda_i$ needs to be non-negative. Also Eq. (26) needs to hold. Subsequently, the complete formulation of the dual problem to be solved for the hard-margin SVM is given as

$$\max_{\{\lambda_i\}_{i=1}^{N}} -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\lambda_i\lambda_jy_iy_jx_i^Tx_j + \sum_{i=1}^{N}\lambda_i, \quad (28)$$

subject to $\lambda_i \geq 0$, for $i = 1, 2, \ldots, N$, \quad (29)

$$\sum_{i=1}^{N}\lambda_iy_i = 0. \quad (30)$$

This is a QP problem and its solution will be discussed later.

### 2.2.2 From Complementary Slackness Condition to Support Vectors

Additionally, the complementary slackness condition $\lambda_i^*[1 - y_i(x_i^Tw^* + b^*)] = 0$ needs to be satisfied for each training sample. This gives the concept of the support vector. After the
training, the training samples that are assigned non-zero Lagrange multipliers must satisfy
\[ 1 - y_i \left( x_i^T w^* + b^* \right) = 0. \] These sample points distribute along the two parallel hyperplanes, and are called support vectors. The remaining training samples with \( \lambda_i^* = 0 \) can have nonzero values of \( 1 - y_i \left( x_i^T w^* + b^* \right) \), more specifically \( y_i \left( x_i^T w^* + b^* \right) > 1 \) according to Eq. (20). These sample points stay outside the margin and are called non-support vectors.

2.2.3 FROM DUAL SOLUTION TO SVM DECISION FUNCTION

The primal feasibility condition in Eq. (17) tells us \( y_i (w^T x_i + b) \geq 1 \). Therefore, among the training samples from the positive class, the one possessing the lowest prediction function value, which is found by
\[ I_+ = \arg \min_{y_i = +1} x_i^T w^* + b^*, \quad (31) \]
should be the closest to the upper parallel hyperplane and should satisfy
\[ x_i^T w^* + b^* = 1, \quad (32) \]
Similarly, among the training samples from the negative class, the one possessing the highest prediction function value, which is found by
\[ I_- = \arg \max_{y_i = -1} x_i^T w^* + b^*, \quad (33) \]
should be the closest to the lower parallel hyperplane and should satisfy
\[ x_i^T w^* + b^* = -1. \quad (34) \]
The optimal value of \( b \) can then be approximated by adding Eqs. (32) and (34), which gives
\[ b^* = -\frac{x_i^T w^* + x_i^T w^*}{2}. \quad (35) \]
An equivalent expression of the above \( b^* \) is
\[ b^* = -\frac{\min_{y_i = +1} x_i^T w^* + \max_{y_i = -1} x_i^T w^*}{2}. \quad (36) \]
Incorporating Eq. (25), the final prediction function of the SVM is computed by
\[
\hat{y} = x^T w^* + b^* \\
= \sum_{i=1}^{N} \lambda_i^* y_i x_i^T x_i - \frac{\min_{y_i = +1} x_i^T w^* + \max_{y_i = -1} x_i^T w^*}{2} \\
= \sum_{i=1}^{N} \lambda_i^* y_i x_i^T x_i - \frac{\min_{y_i = +1} \sum_{i=1}^{N} \lambda_i^* y_i x_i^T x_i + \max_{y_i = -1} \sum_{i=1}^{N} \lambda_i^* y_i x_i^T x_i}{2}. \quad (37)
\]
The above equation shows how the optimal solution \( \{\lambda_i^*\}_{i=1}^{N} \) of the QP problem can be used to build a hard margin SVM classifier.
2.3 Dual Problem of 1-Norm Soft Margin SVM ($l_1$-SVM)

2.3.1 Derive the Dual Problem

In this section, we show how to derive the dual problem for the $l_1$-SVM. The variables to be optimised are stored in the vector \( \begin{bmatrix} w \\ \xi \\ b \end{bmatrix} \), constituting a \((d+N+1)\)-dimensional vector.

The objective function is \( \frac{1}{2} w^T w + C \| \xi \|_1 \). There are a total of \( K = 2N \) inequality constraint functions, where two inequality constraints are required for each of the \( N \) training samples:

\[
1 - \xi_i - y_i (w^T x_i + b) \leq 0, \\
-\xi_i \leq 0.
\]

Similar to the hard margin SVM, there is no equality constraint.

We first construct the Lagrangian function by following Eq. (11):

\[
L(w, \xi, b, \{\alpha_i\}_{i=1}^N, \{\beta_i\}_{i=1}^N) = \frac{1}{2} w^T w + C \| \xi \|_1 + \sum_{i=1}^N \alpha_i [1 - \xi_i - y_i (w^T x_i + b)] + \sum_{i=1}^N \beta_i (-\xi_i).
\]

Here, a total of \( 2N \) Lagrange multipliers \( \{\alpha_i, \beta_i \geq 0\}_{i=1}^N \) are used since there are a total of \( 2N \) inequality constraints. The vectors \( \alpha = [\alpha_1, \alpha_2, \ldots, \alpha_N]^T \) and \( \beta = [\beta_1, \beta_2, \ldots, \beta_N]^T \) are used to store the Lagrange multipliers.

Next we construct the dual problem. Because the objective function of the dual problem is obtained by minimising the Lagrangian function with respect to \( \begin{bmatrix} w \\ \xi \\ b \end{bmatrix} \), we need to compute the gradient of the Lagrangian function, which is given by

\[
\frac{\partial}{\partial w} L(w, \xi, b, \{\alpha_i\}_{i=1}^N, \{\beta_i\}_{i=1}^N) = w - \sum_{i=1}^N \lambda_i y_i x_i, \\
\frac{\partial}{\partial \xi_i} L(w, \xi, b, \{\alpha_i\}_{i=1}^N, \{\beta_i\}_{i=1}^N) = C - (\alpha_i + \beta_i), \quad \text{for } i = 1, 2, \ldots, N, \\
\frac{\partial}{\partial b} L(w, \xi, b, \{\alpha_i\}_{i=1}^N, \{\beta_i\}_{i=1}^N) = -\sum_{i=1}^N \alpha_i y_i.
\]

Here, we do not explain the gradient calculation in details, as it follows a similar procedure as in the hard margin SVM. Setting the gradient to zero, we have

\[
w = \sum_{i=1}^N \lambda_i y_i x_i, \\
C = \alpha_i + \beta_i, \quad \text{for } i = 1, 2, \ldots, N, \\
\sum_{i=1}^N \alpha_i y_i = 0.
\]

Replacing \( w \) with \( \sum_{i=1}^N \lambda_i y_i x_i \), replacing \( \alpha_i + \beta_i \) with \( C \), and \( \sum_{i=1}^N \alpha_i y_i \) with zero, the formulation of the optimised Lagrangian function with respect to \( w, \xi \) and \( b \) is obtained:

\[
O(\alpha, \beta) = -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^N \alpha_i.
\]
Eq. (45) indicates that the values of $\alpha_i$ and $\beta_i$ cannot exceed $C$, because the Lagrange multipliers $\alpha_i$ and $\beta_i$ are non-negative. This is used as an extra constraint for each Lagrange multiplier. The final dual problem of the $l_1$-SVM is given by

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

subject to

$$0 \leq \alpha_i \leq C, \text{ for } i = 1, 2, \ldots, N,$$

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

Comparing the dual problem of the $l_1$-SVM in Eqs. (48)-(50) and the dual problem of the hard margin SVM in Eqs. (28)-(30), it is interesting to see that they actually share the same objective function. The difference is that an extra upper bound $C$ is applied to the Lagrange multipliers of the $l_1$-SVM. The Lagrange multipliers $\{\beta_i\}_{i=1}^N$ of the $l_1$-SVM do not play any role in the optimisation.

2.3.2 From Complementary Slackness Condition to Support Vectors

The complementary slackness conditions to be satisfied for each training sample include

$$\alpha_i^* \left[ 1 - \xi_i^* - y_i (x_i^T w^* + b^*) \right] = 0,$$

$$\beta_i^* \xi_i = (C - \alpha_i^*) \xi_i = 0.$$

We observe the obtained value $\alpha_i^*$ for each training sample.

Figure 3: An illustration of support vectors for linear, non-separable patterns. The figure is from Mu and Nandi (2007).
When $0 < \alpha_i^* < C$, Eq. (52) indicates $\xi_i^* = 0$ and Eq. (51) indicates $1-\xi_i^*-y_i (x_i^T w^* + b^*) = 0$. Combining these two equations, we have $1 - y_i (x_i^T w^* + b^*) = 0$. Therefore, the training samples that possess $0 < \alpha_i^* < C$ distribute along one of the two parallel hyperplanes (either $x_i^T w^* + b^* = +1$ or $x_i^T w^* + b^* = -1$), which are called margin support vectors. Those training samples possessing $\alpha_i^* = C$ can have non-zero $\xi_i$, which then have to satisfy $1-\xi_i^*-y_i (x_i^T w^* + b^*) = 0$ according to Eq. (51). Therefore, they either fall within the margin or stay in the wrong side of the separating hyperplane, because $y_i (x_i^T w^* + b^*) = 1 - \xi_i^* < 1$. Such training samples are called non-margin support vectors. Those training samples possessing $\alpha_i^* = 0$ need to satisfy $(C - 0)\xi_i^* = 0$ according to Eq. (52), which gives $\xi_i = 0$. They also have $1 - \xi_i^*-y_i (x_i^T w^* + b^*) = 1 - y_i (x_i^T w^* + b^*) < 0$ according to Eqs. (51) and (38). Combining these, $y_i (x_i^T w^* + b^*) > 1$ hold for these training samples. This means that these samples all stay outside the margin and are in the right side. These are called non-support vectors. An illustration of support vectors for linear, non-separable patterns is provided in Figure 3.

3. Sequential Minimal Optimisation (Optional Reading)

We have shown in Section 2 that the training of an SVM, which corresponds to finding the optimal values of the direction $w$ and bias $b$ of the separating hyperplane, can be reduced to a dual problem of maximising a quadratic function subject to linear constraints as in Eqs. (28)-(30) or Eqs. (48)-(50), which is known as the QP problem. There are many approaches for solving a QP problem. We will explain one particular algorithm called sequential minimal optimisation (SMO) (Platt, 1998). It shows effective for solving the particular QP problem for SVM. We will explain this approach using the $l_1$-SVM as an example, and its dual problem is repeated below:

$$\max_{\{\alpha_i\}_{i=1}^N} \quad -\frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^N \alpha_i,$$

subject to

$$0 \leq \alpha_i \leq C, \text{ for } i = 1, 2, \ldots, N,$$

$$\sum_{i=1}^N \alpha_i y_i = 0.$$
The optimisation objective function that concerns only $\alpha_1$ and $\alpha_2$ is given as

$$O(\alpha_1, \alpha_2) = -\frac{1}{2} \sum_{i=1}^{2} \sum_{j=1}^{2} \alpha_i \alpha_j y_i y_j x_i^T x_j + \sum_{i=1}^{2} \alpha_i$$

$$= -\frac{1}{2} \left( \alpha_1^2 y_1^2 \|x_1\|^2 + 2 \alpha_1 \alpha_2 y_1 y_2 x_1^T x_2 + \alpha_2 y_2^2 \|x_2\|^2 \right) + \alpha_1 + \alpha_2$$

$$= -\frac{1}{2} \left( \alpha_1^2 \|x_1\|^2 + 2 \alpha_1 \alpha_2 y_1 y_2 x_1^T x_2 + \alpha_2 y_2^2 \|x_2\|^2 \right) + \alpha_1 + \alpha_2. \quad (53)$$

Given a previous setting of $N$ multipliers satisfying the constraint

$$\alpha_1^{(\text{old})} y_1 + \alpha_2^{(\text{old})} y_2 + \alpha_3^{(\text{old})} y_3 + \ldots + \alpha_N^{(\text{old})} y_N = 0.$$ 

When we modify one multiplier to a different value, the above constraint is immediately violated. Therefore, we will need to modify at least one more multiplier’s value to compensate the change. This is why we need to modify at least two multipliers’ values every time. Letting $\eta = -\left(\alpha_3^{(\text{old})} y_3 + \ldots + \alpha_N^{(\text{old})} y_N\right)$, the modified values of $\alpha_1$ and $\alpha_2$ satisfy

$$\alpha_1^{(\text{new})} y_1 + \alpha_2^{(\text{new})} y_2 = \alpha_1^{(\text{old})} y_1 + \alpha_2^{(\text{old})} y_2 = \eta. \quad (54)$$

Since $y_1, y_2 \in \{-1, +1\}$ are class labels, the above condition is equivalent to

$$\alpha_1^{(\text{new})} = y_1 \left(\eta - \alpha_2^{(\text{new})} y_2\right). \quad (55)$$

Incorporating Eq. (55) into Eq. (53), an objective function of one single variable is obtained:

$$O\left( \alpha_2^{(\text{new})} \right) = -\frac{1}{2} \left( y_1^2 \left(\eta - \alpha_2^{(\text{new})} y_2\right)^2 \|x_1\|^2 + 2 y_1 \left(\eta - \alpha_2^{(\text{new})} y_2\right) \alpha_2^{(\text{new})} y_2 x_1^T x_2 + \alpha_2^{(\text{new})^2} \|x_2\|^2 \right)$$

$$+ y_1 \left(\eta - \alpha_2^{(\text{new})} y_2\right) + \alpha_2^{(\text{new})}$$

$$= A_2 \alpha_2^{(\text{new})^2} + A_1 \alpha_2^{(\text{new})} + A_0, \quad (56)$$

where

$$A_2 = -\frac{1}{2} \left( \|x_1\|^2 - 2 x_1^T x_2 + \|x_2\|^2 \right) = -\frac{1}{2} \|x_1 - x_2\|^2, \quad (57)$$

$$A_1 = \eta y_2 \|x_1\|^2 - \eta y_2 x_1^T x_2 - y_1 y_2 + 1, \quad (58)$$

$$A_0 = -\frac{1}{2} \eta^2 \|x_1\|^2 + y_1 \eta. \quad (59)$$

It is very easy to find the maximum of a single-variable function like $f(x) = A_2 x^2 + A_1 x + A_0$ with $A_2 < 0$, without considering any constraint on $x$. The maximum is reached at $\frac{df}{dx} = 2A_2 x + A_1 = 0$, and this gives us a candidate value of the modified Lagrange multiplier:

$$\alpha_2^{(\text{new})} = -\frac{A_1}{2A_2}. \quad (60)$$
However, the modified value has to satisfy the constraint of \( 0 \leq \alpha^{(\text{new})}_2 \leq C \). The same constraint also applies to \( \alpha^{(\text{new})}_1 \):

\[
0 \leq \alpha^{(\text{new})}_1 = y_1 (\eta - \alpha^{(\text{new})}_2 y_2) \leq C,
\]

which is equivalent to

\[
y_1 \eta - C \leq \alpha^{(\text{new})}_2 y_1 y_2 \leq y_1 \eta.
\]

Combining it with \( 0 \leq \alpha^{(\text{new})}_2 \leq C \), we have

\[
\begin{cases}
\max(0, y_1 \eta - C) \leq \alpha^{(\text{new})}_2 \leq \min(y_1 \eta, C), & \text{if } y_1 = y_2, \\
\max(0, -y_1 \eta) \leq \alpha^{(\text{new})}_2 \leq \min(C - y_1 \eta, C), & \text{if } y_1 \neq y_2.
\end{cases}
\]

Since \( \alpha^{(\text{old})}_1 y_1 + \alpha^{(\text{old})}_2 y_2 = \eta \) as in Eq. (54), we rewrite \( y_1 \eta \) as

\[
y_1 \eta = y_1 (\alpha^{(\text{old})}_1 y_1 + \alpha^{(\text{old})}_2 y_2) = \alpha^{(\text{old})}_1 + \alpha^{(\text{old})}_2 y_1 y_2.
\]

Incorporate Eq. (64) into Eq. (63), it results in a final constraint for the modified value of \( \alpha^{(\text{new})}_2 \), expressed as \( L \leq \alpha^{(\text{new})}_2 \leq H \), where

\[
\begin{cases}
L = \max\left(0, \alpha^{(\text{old})}_1 + \alpha^{(\text{old})}_2 - C\right), H = \min\left(\alpha^{(\text{old})}_1 + \alpha^{(\text{old})}_2, C\right), & \text{if } y_1 = y_2, \\
L = \max\left(0, \alpha^{(\text{old})}_2 - \alpha^{(\text{old})}_1\right), H = \min\left(C - \alpha^{(\text{old})}_1 + \alpha^{(\text{old})}_2, C\right), & \text{if } y_1 \neq y_2.
\end{cases}
\]

We have computed earlier a candidate value for \( \alpha^{(\text{new})}_2 \) in Eq. (60). A projection approach can then be applied to re-shape \( \alpha^{(\text{new})}_2 \) so that it satisfies the constraint \( L \leq \alpha^{(\text{new})}_2 \leq H \). The re-shaped \( \alpha^{(\text{new})}_2 \) is given as

\[
\alpha^{(\text{new})}_2 = \begin{cases}
L, & \text{if } -\frac{A_1}{2A_2} < L, \\
H, & \text{if } -\frac{A_1}{2A_2} > H, \\
-\frac{A_1}{2A_2}, & \text{otherwise}.
\end{cases}
\]

The modified value \( \alpha^{(\text{new})}_1 \) can be computed from \( \alpha^{(\text{new})}_2 \) using Eq. (55).

Now, we briefly look at issue (2). Heuristic selection of the two multipliers is applied. The SMO algorithm employs two separate heuristics to choose the first and the second Lagrange multipliers to be optimised. The algorithm goes through the training samples and find one that violates the KKT conditions. The corresponding Lagrange multiplier of this chosen training sample will be treated as \( \alpha_1 \) to be optimised. The second multiplier is chosen in such a way that a modification on the pair \( \alpha_1 \) and \( \alpha_2 \) can cause a large change.

One way to evaluate the change is to first compute the difference between the value of the prediction function and the class label for each training sample (referred to as an error), e.g.,

\[
E_i = \sum_{i=1}^{N} \alpha_i y_i \mathbf{x}_i^T \mathbf{x}_i + b^* - y, \text{ for } i = 1, 2, \ldots, N.
\]

If the error for the chosen training sample in the first round of selection is positive \( (E_{\text{sel1}} > 0) \), the one with the smallest error in the remaining training samples is chosen in the second round of selection; otherwise, the one with the largest error is chosen. The corresponding Lagrange multiplier of the chosen sample in the second round is treated as \( \alpha_2 \).
References


