# Structure Learning: Structured SVM

Chunpai Wang

February 25, 2018

## 1 Problem

The goal is to learn a mapping from input $\mathbf{x} \in X$ to discrete outputs $\mathbf{y} \in Y$ based on a training sample of input-output pairs $(\mathbf{x}_1, \mathbf{y}_1), ..., (\mathbf{x}_2, \mathbf{y}_n) \in (X \times Y)$ drawn from some fixed but unknown probability distribution. Note, here we consider structured output space $Y$, and $\mathbf{y} \in Y$ may be sequences, strings, labeled trees, or graphs. The structured SVM generalize the general **large margin methods**, more specifically multi-class SVMs to the broader problem of learning structured responses y.

Note: general margin is defined as $\mathbf{y}_i \cdot f(\mathbf{x}_i)$, recall the functional margin and geometry margin in standard SVM. This idea may be used in the formulation. Question: what is the formulation of multi-class SVM ?

## 2 Ideas

- The simple idea is to generalize multi-class SVM, and treat each structure as a separate class. However, this is intractable, because too many classes
- We need to specify **discriminant functions** that exploit the structure and dependencies within Y. Question: how does this idea arise ?
- The key idea is the **generalization of the maximum-margin principle**.

## 3 Multi-class SVM

First, let's introduce the one of multi-class SVM method that is different from One-against-the-Rest and One-against-One classifiers, which use a combination of binary classification rules. This method is a direct generalization of the binary classification SV method[2].

Recall, the formulation of standard binary SVM is

$$\begin{aligned} \phi(\mathbf{w}, \xi) &= \frac{1}{2}(\mathbf{w}^\top \mathbf{w}) + C\sum_{i=1}^{n} \xi_i \\ s.t.\ \ & y_i((\mathbf{w}^\top \varphi(\mathbf{x}_i)) + b) \geq 1 - \xi_i, \quad i = 1, ..., n \\ & \xi_i \geq 0, \quad i = 1, ..., n. \end{aligned} \tag{1}$$

If we view hyperplane in binary SVM as a decision function, then a more natural way to solve k-class problems is to construct a decision function by considering all classes at once. Hence, one can generalize the standard binary SVM to the following.

Given a labeled training set $S = \{(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n)\}$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{1, ..., k\}$, the formulation given by Crammer and Singer is [4]

$$\begin{aligned} \min_{\mathbf{w}_m \in \mathcal{H}, \xi \in \mathbb{R}^{n \times k}} \phi(\mathbf{w}, \xi) &= \frac{1}{2}\sum_{m=1}^{k} (\mathbf{w}_m^\top \mathbf{w}_m) + C\sum_{i=1}^{n} \xi_i \\ s.t.\ \ & (\mathbf{w}_{y_i}^\top \varphi(\mathbf{x}_i)) + \delta_{y_i,m} \geq (\mathbf{w}_m^\top \varphi(\mathbf{x}_i)) + 1 - \xi_i, \\ & \xi_i \geq 0, \\ & i = 1, ..., n; \end{aligned} \tag{2}$$

where $\delta_{y_i,m} = 1$ if $y_i = m$, and 0 otherwise.

Weston and Watkins[2] introduce $n|Y| - n$ slack variables, But here, we only use $n$ slack variables. Please check extra note on relation between multi-class SVM and logistic regression model for details. I think we can use other scalar to replace 1, but it will have several optimal $\mathbf{w}$ by simply rescaling it. Please check standard SVM, functional margin and geometry margin. We can simply view 1 as the margin between the true label $y_i$ and the best $\in Y \backslash y_i$.

In the point of view of multi-class classification, we replace the misclassification error of an example with the following piecewise linear bound:

$$\max_{m \in Y \backslash y_i} \{\mathbf{w}_m \cdot \varphi(\mathbf{x}_i) + 1 - \delta_{y_i,m}\} - \mathbf{w}_{y_i} \cdot \varphi(\mathbf{x}_i) \tag{3}$$

where $\delta_{p,q} = 1$ if $p = q$ and 0 otherwise. We can call the value of the inner product of $\mathbf{w}_i$ with the instance $\varphi(\mathbf{x})$ the **confidence** and the **similarity score** for the $i^{th}$ class. Note, the above bound is 0 if the confidence value for the correct label is larger by at least one than the confidences assigned to the rest of the labels. Otherwise, we suffer a loss which is linearly proportionally to the difference between the confidence of the correct label and the maximum among the confidences of the other labels.

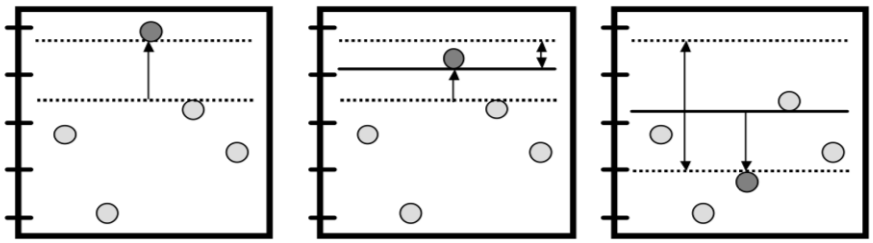

Figure 1: Illustration of the margin bound employed in multi-class SVM. Correct label is plotted in dark grey. The height of each label designates its confidence. The left plot corresponds to the case when the margin is larger than one, and therefore the error bound equals zero, and hence the example is correctly classified. The middle figure shows a case where the example is correctly classified but with small margin and we suffer some loss. The right plot depicts the loss of a misclassified example. In the middle and right plots, double-arrow denotes the error bound, and single-arrow denotes the loss. Note that error and loss are two different things in the context.

Summing over all the examples in $S$ we get an upper bound on the empirical loss

$$E_S \leq \frac{1}{n} \sum_{i=1}^{n} [\max_{m \in Y \backslash y_i} \{\mathbf{w}_m \cdot \varphi(\mathbf{x}_i) + 1 - \delta_{y_i,m}\} - \mathbf{w}_{y_i} \cdot \varphi(\mathbf{x}_i)] \tag{4}$$

We say that a sample $S$ is linearly separable by multi-class machine if there exists a set of $\{\mathbf{w}_i | i = 1, ..., k\}$ such that the above loss is equal to 0, that is

$$\forall i \max_{m \in Y \backslash y_i} \{\mathbf{w}_m \cdot \varphi(\mathbf{x}_i) + 1 - \delta_{y_i,m}\} - \mathbf{w}_{y_i} \cdot \varphi(\mathbf{x}_i) = 0 \tag{5}$$

which is same as

$$\forall i, m \qquad \mathbf{w}_{y_i} \cdot \varphi(\mathbf{x}_i) + \delta_{y_i,m} - \mathbf{w}_m \cdot \varphi(\mathbf{x}_i) \geq 1 \tag{6}$$

In the general case the sample $S$ might not be linearly separable by a multi-class machine. We therefore add slack variables $\xi_i \geq 0$ and modify the constraints to be,

$$\forall i, m \qquad \mathbf{w}_{y_i} \cdot \varphi(\mathbf{x}_i) + \delta_{y_i,m} - \mathbf{w}_m \cdot \varphi(\mathbf{x}_i) \geq 1 - \xi_i \tag{7}$$

# 4 Discriminant Functions

In the structured SVM, we can further generalize the multi-class SVM. First, we can define discriminant functions as

$$F : X \times Y \to \mathbb{R} \tag{8}$$

Note, the domain is input $\times$ output space, and the approach we pursue is to learn a discriminant function $F$ over input $\times$ output pairs from which we can derive a prediction by maximizing $F$ over the response variable $\mathbf{y}$ for a specific given input $\mathbf{x}$. The number of possible $\mathbf{y}$ corresponds to a fixed $x$ should be tractable.

General Form of Our Hypothesis:

$$f(\mathbf{x}, \mathbf{w}) = \arg\max_{\mathbf{y} \in Y} F(\mathbf{x}, \mathbf{y}; \mathbf{w}) \tag{9}$$

where $w$ denotes a **parameter** vector. We assume $F$ to be linear in some combined feature representation of inputs and output $\Psi(\mathbf{x}, \mathbf{y})$,

$$F(\mathbf{x}, \mathbf{y}; \mathbf{w}) = \langle \mathbf{w}, \Psi(\mathbf{x}, \mathbf{y}) \rangle \tag{10}$$

Note, the specific form of $\Psi$ depends on the nature of the problem. We can view $\Psi(\mathbf{x}, \mathbf{y})$ as combining and mapping input and output features to a space, where we can apply numerical optimization.

### 4.1 Example of $\Psi(x, y)$

Assuming we have a list of predefined context free grammars (rules), and each rule $g_j$ has a corresponding weight $w_j$. Now, feeding a sentence, we can build a parse tree based on CFGs (see figure). Here, we can design the function $\Psi(\mathbf{x}, \mathbf{y})$ as a histogram vector counting how often each grammar rule $g_j$ occurs or used in the tree $\mathbf{y}$.

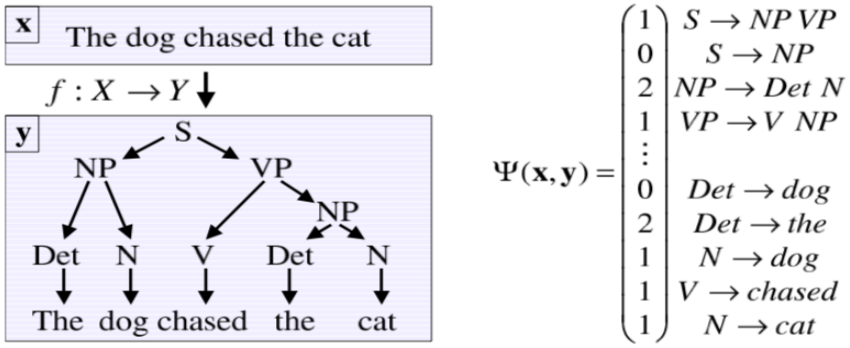


Figure 2: Illustration of natural language parsing model

In this example, $f(\mathbf{x}; \mathbf{w})$ can be efficiently computed by finding the structure $\mathbf{y} \in Y$ that maximize $F(\mathbf{x}, \mathbf{y}; \mathbf{w})$ via the CKY algorithm.

Question: so here $\mathbf{w}$ is predefined in this example ? No, $\mathbf{w}$ will be learned when we feeding training samples $S = \{(\mathbf{x}_i, \mathbf{y}_i)\}$, such that training error is minimized. For example, in multi-class SVM we have $k$ classes, then each class corresponds to one $\mathbf{w}_k$. In structured SVM, we are looking for only one $\mathbf{w}$ that maximize the determinant function.

## 5 Loss Function

We assume availability of a bounded loss function

$$\triangle : Y \times Y \to \mathbb{R} \tag{11}$$

where $\triangle(\mathbf{y}, \hat{\mathbf{y}})$ quantifies the loss associated with a prediction $\hat{\mathbf{y}}$ to the true output value $\mathbf{y}$. For example, in natural language parsing, there exists a metric to measure the difference from the predicted parse tree to the correct parse tree.

**The Generalization Error or Risk** of our predicted hypothesis $f$ can be formulated as

$$\mathcal{R}_P^{\triangle}(f) = \int_{X \times Y} \triangle(\mathbf{y}, f(\mathbf{x})) dP(\mathbf{x}, \mathbf{y}) \tag{12}$$

where $P(\mathbf{x}, \mathbf{y})$ is the unknown true distribution of input-output pairs.

**The Empirical Risk** of our predicted hypothesis $f$ can be formulated as

$$\mathcal{R}_S^{\triangle}(f) = \sum_{(\mathbf{x}_i, \mathbf{y}_i) \in S} \triangle(\mathbf{y}_i, f(\mathbf{x}_i)) \tag{13}$$

where $S$ is a set of training samples. Note, in our case, we use the discriminant function, which is parameterized by vector $\mathbf{w}$. Thus, we will also write $\mathcal{R}_P^{\triangle}(\mathbf{w}) = \mathcal{R}_P^{\triangle}(f(\cdot; \mathbf{w}))$.

# 6 Margins and Margin Maximization

## 6.1 Hard Margin Case

In separable case, there exists a ERM hypothesis $f$, such that for all $(\mathbf{x}_i, \mathbf{y}_i) \in S$,

$$f(\mathbf{x}_i; \mathbf{w}) = \mathbf{y}_i = \arg\max_{\mathbf{y} \in Y} F(\mathbf{x}_i, \mathbf{y}; \mathbf{w}) = \arg\max_{\mathbf{y} \in Y} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle \tag{14}$$

Thus, we have

$$\begin{aligned} \forall i : \max_{\mathbf{y} \in Y \backslash \mathbf{y}_i} F(\mathbf{x}_i, \mathbf{y}; \mathbf{w}) < F(\mathbf{x}_i, \mathbf{y}_i; \mathbf{w}) \\ \forall i : \max_{\mathbf{y} \in Y \backslash \mathbf{y}_i} \{\langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle\} < \{\langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle\} \end{aligned} \tag{15}$$

If we define

$$\delta\Psi_i(\mathbf{y}) := \Psi(\mathbf{x}_i, \mathbf{y}_i) - \Psi(\mathbf{x}_i, \mathbf{y}) \tag{16}$$

then we can replace $n$ nonlinear constraint in (11) with $n|Y| - n$ linear constraints

$$\forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y}_i : \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}) \rangle > 0 \tag{17}$$

If the set of $n|Y| - n$ inequalities is feasible, there will typically be more than one solution $\mathbf{w}^*$, because we can scale the $\mathbf{w}*$. Therefore, we need to select the $\mathbf{w}$ with $\|w\| \leq 1$ for which the score of the correct label $\mathbf{y}_i$ is uniformly most different from the closest runner-up $\hat{\mathbf{y}}_i(\mathbf{w}) = \arg\max_{\mathbf{y} \neq \mathbf{y}_i} \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle$ . This generalizes the maximum-margin principle employed in SVMs to the more general case here.

Question: how to interpret the red sentence ? Note that, in large-margin method, large margin implies small $\mathbf{w}$, and further implies regularized solution and good generalization.[7]

Let's go back to the origin problem that we want to maximize the margin (or difference) between true label $\mathbf{y}_i$ and the best $\mathbf{y} \in Y \backslash \mathbf{y}_i$, and we define the difference as

$$\hat{\gamma} = \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle - \max_{\mathbf{y} \in Y \backslash \mathbf{y}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i, \mathbf{y}) \rangle \tag{18}$$

Intuitively, We want the difference to be as large as possible to make our prediction on testing data to be confident and correct. However, we can scale the $\mathbf{w}$ to make $\gamma$ large, but it does not change our confidence, because the difference between **arbitrary** two $\langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle$ is also scaled, and it does not change the order or location. Therefore, we would like to replace $\mathbf{w}$ with $\frac{\mathbf{w}}{\|w\|}$, and define

$$\gamma = \langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle - \max_{\mathbf{y} \in Y \backslash \mathbf{y}_i} \langle \frac{\mathbf{w}}{\|\mathbf{w}\|}, \Phi(\mathbf{x}_i, \mathbf{y}) \rangle \tag{19}$$

and if $\|\mathbf{w}\| = 1$, then $\gamma = \hat{\gamma}$

Thus, our objective is to maximize the $\gamma$, and we can formulate our initial problem as

$$\begin{aligned} &\max_{\gamma, \mathbf{w}} \gamma \\ &s.t.\ \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle - \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle \geq \gamma \quad \forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y}_i \\ &\|\mathbf{w}\| = 1 \end{aligned} \tag{20}$$

The $\|\mathbf{w}\| = 1$ constraint moreover ensures that $\gamma = \hat{\gamma}$, so we are also guaranteed that all $\langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle - \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle \geq \gamma$.

However, the $\|\mathbf{w}\| = 1$ constraint is a nasty non-convex one.

$$\begin{aligned} &\max_{\hat{\gamma}, \mathbf{w}} \frac{\hat{\gamma}}{\|\mathbf{w}\|} \\ &s.t.\ \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle - \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}) \rangle \geq \hat{\gamma} \quad \forall i,\ \forall \mathbf{y} \in Y \backslash \mathbf{y}_i \end{aligned} \tag{21}$$

Now, we remove the constraint $\|\mathbf{w}\| = 1$, but the objective becomes non-convex.

Recall that we can add an arbitrary scaling constraint on $\mathbf{w}$ without changing the layout of each $\Psi(\mathbf{x}_i, \mathbf{y}_i)$. Thus, we would like introduce the scaling constraint on $\mathbf{w}$ such that

$$\hat{\gamma} = \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i) \rangle - \max_{\mathbf{y} \in Y \backslash \mathbf{y}_i} \langle \mathbf{w}, \Phi(\mathbf{x}_i, \mathbf{y}) \rangle = 1 \tag{22}$$

Thus, our formulation becomes

$$\max_{\mathbf{w}} \frac{1}{\|\mathbf{w}\|} \\ s.t.\ \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i)\rangle - \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y})\rangle \geq 1 \quad \forall i,\ \forall \mathbf{y} \in Y \backslash \mathbf{y}_i \tag{23}$$

The resulting hard-margin optimization problem is

$$[\text{SVM}_0] \quad \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 \\ s.t. \langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle \geq 1 \quad \forall i,\ \forall \mathbf{y} \in Y \backslash \mathbf{y}_i \tag{24}$$

where $\delta\Psi_i(\mathbf{y}) = \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y}_i)\rangle - \langle \mathbf{w}, \Psi(\mathbf{x}_i, \mathbf{y})\rangle$.

## 6.2 Soft Margin Case

We introduce one slack variable for every non-linear constraint (11), and add a penalty term that is linear in the slack variables to the objective results in the quadratic program [4]

$$[\text{SVM}_1] \quad \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n} \xi_i \quad s.t.\ \ \forall i, \xi_i \geq 0 \\ s.t. \forall i \quad \forall \mathbf{y} \in Y \backslash \mathbf{y}_i : \langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle \geq 1 - \xi_i \tag{25}$$

where $C > 0$ is a constant that controls the trade-off between training error minimization and margin maximization. Adding the slack variable will result in a upper bound on the empirical risk and offers some additional algorithmic advantages.

Question: Is $\xi_i$ hinge loss here? $\xi_i$ is used to count the number of errors, and we can use surrogate loss to approximate the 0-1 loss, such as hinge loss and logistic function. We can replace the $\frac{C}{n}\sum_{i=1}^{n}\xi_i$ with $\frac{C}{2n}\xi_i^2$ to smooth the function. The front can be think as sparse learning.

## 6.3 Generalization to Arbitrary Loss Function

The formulation (19) implicitly considers the zero-one classification loss. That means if predicted $y$ is same as $y_i$, then loss is $\triangle(\mathbf{y}, \mathbf{y}_i) = 0$, and $\triangle(\mathbf{y}, \mathbf{y}_i) = 1$ otherwise. Thus $\xi_i$ is same as $\frac{\xi_i}{\triangle(\mathbf{y}, \mathbf{y}_i)}$. However, in the structured data setting, we would like the prediction loss is associated with $\triangle(\mathbf{y}_i, \mathbf{y})$. In general, violating a margin constraint involving a $\mathbf{y} \neq \mathbf{y}_i$ with high loss $\triangle(\mathbf{y}_i, \mathbf{y})$ should be penalized more severely than smaller loss $\triangle(\mathbf{y}_i, \mathbf{y})$. Otherwise, violating the margin constraint for $\mathbf{y}_i$ and $\mathbf{y}_j$ may have same penalty ($\xi_i = \xi_j$), but very different loss $\triangle(\mathbf{y}_i, \mathbf{y})$ and $\triangle(\mathbf{y}_j, y)$. Therefore, we use 3 methods belows

$$[\text{SVM}_1^{\triangle s}] \quad \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n} \xi_i \quad s.t.\ \ \forall i, \xi_i \geq 0 \\ s.t. \forall i \quad \forall \mathbf{y} \in Y \backslash \mathbf{y}_i : \langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle \geq 1 - \frac{\xi_i}{\triangle(\mathbf{y}_i, \mathbf{y})} \tag{26}$$

$$[\text{SVM}_2^{\triangle s}] \quad \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n} \xi_i \quad s.t.\ \ \forall i, \xi_i \geq 0 \\ s.t. \forall i \quad \forall \mathbf{y} \in Y \backslash \mathbf{y}_i : \langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle \geq 1 - \frac{\xi_i}{\sqrt{\triangle(\mathbf{y}_i, \mathbf{y})}} \tag{27}$$

This approach is to rescale the margin for the special case of the Hamming loss

$$\begin{aligned}[\text{SVM}_2^{\triangle m}] \quad & \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n}\xi_i \quad s.t. \;\; \forall i, \xi_i \geq 0 \\ & s.t. \forall i \quad \forall \mathbf{y} \in Y \backslash \mathbf{y}_i : \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}) \rangle \geq \triangle(\mathbf{y}_i, \mathbf{y}) - \xi_i \end{aligned} \tag{28}$$

# 7 Dual Programs

New problem: the primal problems have a large number of margin constraints, and we cannot use standard QP solvers to solve them. Hence, we have to seek helps from the dual problem. By applying Lagrangian dual, we can get the dual program of $\text{SVM}_0$ as

$$\begin{aligned} \max_{\alpha} & \sum_{i, \mathbf{y} \neq \mathbf{y}_i} \alpha_{i\mathbf{y}} - \frac{1}{2} \sum_{i, \mathbf{y} \neq \mathbf{y}_i j, \bar{\mathbf{y}} \neq \mathbf{y}_j} \alpha_{i\mathbf{y}} \alpha_{j\mathbf{y}} \langle \delta\Phi_i(\mathbf{y}), \delta\Phi_j(\bar{\mathbf{y}}) \rangle \\ s.t. \;\; & \forall i, \forall \mathbf{y} \neq Y \backslash \mathbf{y}_i : \alpha_{i\mathbf{y}} \geq 0 \end{aligned} \tag{29}$$

It should be $\forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y}_i : \alpha_{i\mathbf{y}} \geq 0$ ?

First, we can get the Lagrangian for $\text{SVM}_0$

$$\begin{aligned} L(\mathbf{w}, \alpha) &= \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{i=1}^{m} \alpha_{i\mathbf{y}} [1 - \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}) \rangle] \\ &= \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{m} \alpha_{i\mathbf{y}} [\langle \mathbf{w}, \delta\Psi_i(\mathbf{y}) \rangle - 1] \qquad \forall i, \mathbf{y} \in Y \backslash \mathbf{y}_i \end{aligned} \tag{30}$$

Note, the size of $\alpha$ is $n|Y| - n$ and $\alpha \geq 0$. Then, we have Lagrangian dual

$$g(\alpha) = \min_{\mathbf{w}} L(\mathbf{w}, \alpha) \tag{31}$$

We denote $p^*$ is the optimal value of primal problem $\text{SVM}_0$, then we have fact that

$$g(\alpha) \leq p^* \tag{32}$$

and we can formulate the dual problem as

$$\begin{aligned} \max_{\alpha} g(\alpha) &= \max_{\alpha} \min_{\mathbf{w}} L(\mathbf{w}, \alpha) \\ s.t. \quad & \alpha \geq 0 \end{aligned} \tag{33}$$

We denote by d* the optimal value of dual problem. Thus we can present weak duality and strong duality as

Weak duality :

$$d^* = \max_{\alpha \geq 0} \min_{\mathbf{w}} L(\mathbf{w}, \alpha) \leq \min_{\mathbf{w}} \max_{\alpha \geq 0} L(\mathbf{w}, \alpha) = p^* \tag{34}$$

Strong duality:

$$d^* = p^* \tag{35}$$

Since the primal problem is convex, and slater condition always holds for affine constraints, we can conclude strong duality holds. Also, if strong duality holds and $\mathbf{w}^*$ and $\alpha^*$ are optimal, then they must satisfy the KKT conditions, which are

1. primal constraints: $f_i\mathbf{y}(\mathbf{w}^*) = 1 - \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}) \rangle \leq 0 \quad \forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y}_i$
2. dual constraints: $\alpha^*_{i\mathbf{y}} \geq 0 \quad \forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y}$
3. complementary slackness: $\alpha_{i\mathbf{y}} f_{i\mathbf{y}}(\mathbf{w}^*) = 0 \quad \forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y}$, that is

$$\alpha^*_{i\mathbf{y}}(1 - \langle \mathbf{w}, \delta\Psi_i(\mathbf{y}) \rangle) = 0 \quad \forall i, \forall \mathbf{y} \in Y \backslash \mathbf{y} \tag{36}$$

4. gradient of Lagrangian with respect to $\mathbf{w}^*$ vanishes

$$\nabla f_0(\mathbf{w}^*) + \sum_i \sum_{\mathbf{y}\in Y\setminus\mathbf{y}_i} \alpha_{i\mathbf{y}} \nabla f_{i\mathbf{y}}(\mathbf{w}^*) = 0$$

that is

$$\mathbf{w}^* - \sum_i \sum_{\mathbf{y}\in Y\setminus\mathbf{y}_i} \alpha_{i\mathbf{y}} \delta\Psi_i(\mathbf{y}) = 0 \tag{37}$$

Now, we can formulate our dual problem of $\text{SVM}_0$

$$\max_\alpha \min_{\mathbf{w}} L(\mathbf{w}, \alpha) \tag{38}$$

$$= \max_\alpha \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^n \sum_{\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} [\langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle - 1] \tag{39}$$

$$= \max_\alpha \frac{1}{2}\|\mathbf{w}^*\|^2 - \sum_{i=1}^n \sum_{\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} [\langle \mathbf{w}^*, \delta\Psi_i(\mathbf{y})\rangle - 1] \tag{40}$$

$$= \max_\alpha \frac{1}{2}\| \sum_j \sum_{\mathbf{y}\in Y\setminus\mathbf{y}_j} \alpha_{j\bar{\mathbf{y}}} \delta\Psi_j(\bar{\mathbf{y}})\|^2 - \sum_i \sum_{\mathbf{y}\in Y\setminus\mathbf{y}_i} \alpha_{i\mathbf{y}} [\langle \sum_j \sum_{\bar{\mathbf{y}}\in Y\setminus\mathbf{y}_j} \alpha_{j\bar{\mathbf{y}}} \delta\Psi_j(\bar{\mathbf{y}}), \delta\Psi_i(\mathbf{y})\rangle - 1] \tag{41}$$

$$= \max_\alpha \sum_{i,\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} - \frac{1}{2} \sum_{\substack{i,\mathbf{y}\neq\mathbf{y}_i \\ j,\bar{\mathbf{y}}\neq\mathbf{y}_j}} \alpha_{i\mathbf{y}} \alpha_{j\bar{\mathbf{y}}} \langle \delta\Psi_i(\mathbf{y}), \delta\Psi_j(\bar{\mathbf{y}})\rangle \tag{42}$$

$$= \max_\alpha \sum_{i,\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} - \frac{1}{2} \sum_{\substack{i,\mathbf{y}\neq\mathbf{y}_i \\ j,\bar{\mathbf{y}}\neq\mathbf{y}_j}} \alpha_{i\mathbf{y}} \alpha_{j\bar{\mathbf{y}}} K \tag{43}$$

where $\alpha_{i\mathbf{y}} \geq 0$. The (41) is by 4th of KKT conditions; Since there is inner product of mapping function $\delta\Psi(\mathbf{x}, \mathbf{y})$, we can use $K((\mathbf{x}, \mathbf{y}), (\mathbf{x}', \mathbf{y}'))$ (a kernel) which is positive definite matrix to replace the inner product to reduce the computation complexity.

Question: $K$ is used to replace inner product of one mapping $\delta\Psi$ or two mappings $\phi(\delta\Psi)$? If for one mapping , does it mean we do not need to design the function $\Psi$ ?

For soft-margin optimization with slack rescaling and linear penalties ($\text{SVM}_1^{\triangle s}$), we can also formulate its dual problem

$$\max_\alpha \min_{\mathbf{w},\xi} L(\mathbf{w}, \alpha) \quad s.t.\ \xi_i \geq 0; \quad \alpha_{i\mathbf{y}\geq 0} \quad \forall i, \forall \mathbf{y}\in Y\setminus\mathbf{y}_i \tag{44}$$

$$= \max_{\alpha,\beta} \min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^n \xi_i - \sum_{i=1}^n \sum_{\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} [\langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle - 1 + \frac{\xi_i}{\triangle(\mathbf{y}_i, \mathbf{y})}] - \beta_i \xi_i \quad s.t.\ \beta_i \geq 0 \tag{45}$$

If we take derivative with respect to $\xi_i$, we will get the implicit constraint

$$\frac{C}{n} - \sum_{i,\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} \cdot \frac{1}{\triangle(\mathbf{y}_i, \mathbf{y})} - \beta_i = 0 \tag{46}$$

since $\beta_i \geq 0$, we will explicit constraints

$$\begin{aligned} \frac{C}{n} - \sum_{\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} \cdot \frac{1}{\triangle(\mathbf{y}_i, \mathbf{y})} &\geq 0 \quad \forall i \\ \sum_{\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} \cdot \frac{1}{\triangle(\mathbf{y}_i, \mathbf{y})} &\leq \frac{C}{n} \quad \forall i \\ n \sum_{\mathbf{y}\neq\mathbf{y}_i} \frac{\alpha_{i\mathbf{y}}}{\triangle(\mathbf{y}_i, \mathbf{y})} &\leq C \quad \forall i \end{aligned} \tag{47}$$

We can arrange the terms and apply the $\beta_i = \frac{C}{n} - \sum_{i,\mathbf{y}\neq\mathbf{y}_i} \alpha_{i\mathbf{y}} \cdot \frac{1}{\triangle(\mathbf{y}_i,\mathbf{y})}$ in formula (45), then we get dual problem of $\text{SVM}_1^{\triangle s}$ as

$$\max_{\alpha,\beta} \min_{\mathbf{w},\xi} L(\mathbf{w},\alpha,\beta) \tag{48}$$

$$= \max_{\alpha,\beta} \min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 + \frac{C}{n}\sum_{i=1}^{n}\xi_i - \sum_{i=1}^{n}\sum_{\mathbf{y}\neq\mathbf{y}_i}\alpha_{i\mathbf{y}}[\langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle - 1 + \frac{\xi_i}{\triangle(\mathbf{y}_i,\mathbf{y})}] - \sum_{i=1}\beta_i\xi_i \tag{49}$$

$$= \max_{\alpha,\beta} \min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n}\sum_{\mathbf{y}\neq\mathbf{y}_i}\alpha_{i\mathbf{y}}[\langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle + 1] + \underbrace{\frac{C}{n}\sum_{i=1}^{n}\xi_i - \sum_{i}\sum_{\mathbf{y}\neq\mathbf{y}_i}\frac{\alpha_{i\mathbf{y}}\xi_i}{\triangle(\mathbf{y}_i,\mathbf{y})} - \sum_{i=1}\beta_i\xi_i}_{=0} \tag{50}$$

$$= \max_{\alpha,\beta} \min_{\mathbf{w},\xi} \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n}\sum_{\mathbf{y}\neq\mathbf{y}_i}\alpha_{i\mathbf{y}}[\langle \mathbf{w}, \delta\Psi_i(\mathbf{y})\rangle + 1] \tag{51}$$

$$= \max_{\alpha} \frac{1}{2}\|\mathbf{w}^*\|^2 - \sum_{i=1}^{n}\sum_{\mathbf{y}\neq\mathbf{y}_i}\alpha_{i\mathbf{y}}[\langle \mathbf{w}^*, \delta\Psi_i(\mathbf{y})\rangle - 1] \tag{52}$$

$$= \max_{\alpha} \frac{1}{2}\|\sum_{j}\sum_{\mathbf{y}\in Y\setminus\mathbf{y}_j}\alpha_{j\bar{\mathbf{y}}}\delta\Psi_j(\bar{\mathbf{y}})\|^2 - \sum_{i}\sum_{\mathbf{y}\in Y\setminus\mathbf{y}_i}\alpha_{i\mathbf{y}}[\langle\sum_{j}\sum_{\bar{\mathbf{y}}\in Y\setminus\mathbf{y}_j}\alpha_{j\bar{\mathbf{y}}}\delta\Psi_j(\bar{\mathbf{y}}), \delta\Psi_i(\mathbf{y})\rangle - 1] \tag{53}$$

$$= \max_{\alpha} \sum_{i,\mathbf{y}\neq\mathbf{y}_i}\alpha_{i\mathbf{y}} - \frac{1}{2}\sum_{\substack{i,\mathbf{y}\neq\mathbf{y}_i\\ j,\bar{\mathbf{y}}\neq\mathbf{y}_j}}\alpha_{i\mathbf{y}}\alpha_{j\bar{\mathbf{y}}}\langle\delta\Psi_i(\mathbf{y}), \delta\Psi_j(\bar{\mathbf{y}})\rangle \tag{54}$$

$$= \max_{\alpha} \sum_{i,\mathbf{y}\neq\mathbf{y}_i}\alpha_{i\mathbf{y}} - \frac{1}{2}\sum_{\substack{i,\mathbf{y}\neq\mathbf{y}_i\\ j,\bar{\mathbf{y}}\neq\mathbf{y}_j}}\alpha_{i\mathbf{y}}\alpha_{j\bar{\mathbf{y}}}K \tag{55}$$

same dual objective but with extra constraints

$$n\sum_{\mathbf{y}\neq\mathbf{y}_i}\frac{\alpha_{i\mathbf{y}}}{\triangle(\mathbf{y}_i,\mathbf{y})} \leq C \quad \forall i$$

We can also apply similar procedures to other variant primal problems to get their dual problems.

# 8 Algorithm

**Algorithm 1** Algorithm for solving $\text{SVM}_0$ and the loss re-scaling formulations $\text{SVM}_1^{\triangle s}$ and $\text{SVM}_2^{\triangle s}$

Input: $(\mathbf{x}_1,\mathbf{y}_1),\ldots,(\mathbf{x}_n,\mathbf{y}_n)$, $C$, $\epsilon$
$S_i \leftarrow \emptyset$ for all $i = 1,\ldots,n$
**repeat**
  **for** $i = 1,\ldots,n$ **do**
    set up cost function
    $\text{SVM}_1^{\triangle s}$: $H(\mathbf{y}) \equiv (1 - \langle\delta\Psi_i(\mathbf{y}), \mathbf{w}\rangle)\,\triangle(\mathbf{y}_i,\mathbf{y})$
    $\text{SVM}_2^{\triangle s}$: $H(\mathbf{y}) \equiv (1 - \langle\delta\Psi_i(\mathbf{y}), \mathbf{w}\rangle)\sqrt{\triangle(\mathbf{y}_i,\mathbf{y})}$
    $\text{SVM}_1^{\triangle m}$: $H(\mathbf{y}) \equiv \triangle(\mathbf{y}_i,\mathbf{y}) - \langle\delta\Psi_i(\mathbf{y}), \mathbf{w}\rangle$
    $\text{SVM}_2^{\triangle m}$: $H(\mathbf{y}) \equiv \sqrt{\triangle(\mathbf{y}_i,\mathbf{y})} - \langle\delta\Psi_i(\mathbf{y}), \mathbf{w}\rangle$
    where $\mathbf{w} \equiv \sum_j \sum_{\mathbf{y}'\in S_j}\alpha_{j\mathbf{y}'}\delta\Psi_j(\mathbf{y}')$.
    compute $\hat{\mathbf{y}} = \arg\max_{\mathbf{y}\in Y} H(\mathbf{y})$
    compute $\xi_i = \max\{0, \max_{\mathbf{y}\in S_i} H(\mathbf{y})\}$
    **if** $H(\hat{\mathbf{y}}) > \xi_i + \epsilon$ **then**
      $S_i \leftarrow S_i \cup \{\hat{\mathbf{y}}\}$
      $\alpha_S \leftarrow$ optimize dual over $S$, $S = \cup_i S_i$.
    **end if**
  **end for**
**until** no $S_i$ has changed during iteration

This algorithm uses a cutting plane method, which is to iteratively tighter the relaxations of the original problem. Solving the dual QP problem is more feasible than the primal QP problem, because

- it depends on inner products in the joint feature space defined by $\Psi$, hence allowing the user of kernel functions, and we don;t need to design the function $\Psi$
- the constraint matrix of the dual supports a natural problem decomposition, since it is block diagonal, where each block corresponds to a specific training instance, and each block has row size $|Y| - 1$

# References


[1] Tsochantaridis, Ioannis, et al. "Support vector machine learning for interdependent and structured output spaces." Proceedings of the twenty-first international conference on Machine learning. ACM, 2004.(1999): 988-999.

[2] Weston, Jason, and Chris Watkins. Multi-class support vector machines. Technical Report CSD-TR-98-04, Department of Computer Science, Royal Holloway, University of London, May, 1998.

[3] Weston, Jason, and Chris Watkins. "Support vector machines for multi-class pattern recognition." Esann. Vol. 99. 1999.

[4] Crammer, Koby, and Yoram Singer. "On the algorithmic implementation of multiclass kernel-based vector machines." Journal of machine learning research 2.Dec (2001): 265-292.

[5] Lecture 18: Multiclass Support Vector Machines - Arizona Math http://math.arizona.edu/~hzhang/math574m/2017Lect18_msvm.pdf

[6] Wang, Zhe, and Xiangyang Xue. "Multi-class support vector machine." Support Vector Machines Applications. Springer, Cham, 2014. 23-48.

[7] The Support Vector Machine and regularization - MIT OpenCourseWare https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-867-machine-learning-fall-2006/lecture-notes/lec4.pdf

[8] Joachims, Thorsten, Thomas Finley, and Chun-Nam John Yu. "Cutting-Plane Training of Structural SVMs." https://www.cs.cornell.edu/people/tj/publications/joachims_etal_09a.pdf

[9] http://mat.gsia.cmu.edu/classes/integer/node11.html

[10] http://www.seas.ucla.edu/~vandenbe/236C/lectures/localization.pdf

[11] https://stanford.edu/class/ee364b/lectures/localization_methods_slides.pdf