COMP4131: Data Modelling and Analysis

Lecture 10: Support Vector Machines and Kernel Methods

Daokun Zhang

University of Nottingham Ningbo China

daokun.zhang@nottingham.edu.cn

April 21, 2025

FAQs for Coursework 2

- Q1: Should I compare my solutions with the state-of-the-art baselines?
- A1: Comparison with the state-of-the-art is encouraged but not compulsory.
- Q2: Should I work on the classification and regression tasks simultaneously?
- A2: Please focus on only one task. You can choose one the three tasks: classification, regression and clustering.
- Q3: Can I choose a method that has not been covered by our module?
- A3: Yes. The solution methods are not limited to what we have learned. We encourage you to explore more advanced machine learning techniques to solve your problem.

FAQs for Coursework 2

- Q4: How many solution methods should I compare?
- A4: It's expected that more than 3 methods should be investigated. Generally, the more the better, but enough words and space have to be used for describing data pre-processing details, justifying the reason for choosing the used methods, showcasing and analyzing the experimental results.
- Q5: What data scale is expected?
- A5: Generally, a dataset with more than 2000 samples is favored. If the dataset you identified contains too many samples, like in million or billion scales, you can do some down-sampling to reduce the scale.
- Q6: Can I use image or textual data?
- A6: No. Please use the tabular data.

FAQs for Coursework 2

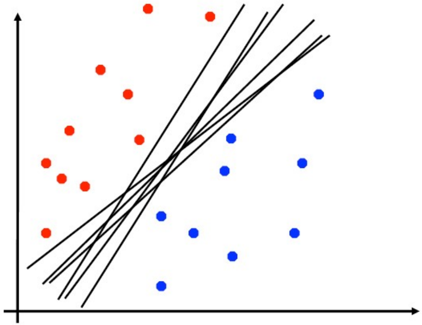
- Q7: Can I re-use the datasets of our lab sessions?
- A7: No. Please identify some new datasets.
- Q8: Can I use the same dataset with some other student?
- A8: No. Please be different to avoid the suspicion of plagiarism.

Overview

Support Vector Machines

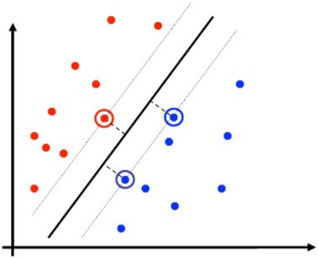
Linear Separators

- If training data is linearly separable, we can find some linear separator to distinguish two classes of examples.
- Which of these is optimal?



Support Vector Machines (SVMs)

- SVMs (Vapnik, 1990's) choose the linear separator with the largest margin.



Robust to outliers!

Vladimir Vapnik

- Good according to intuition, theory, practice.
- SVMs became famous when, using images as input, it gave accuracy comparable to neural-network with hand-designed features in a handwriting recognition task.

Geometry of Linear Separators

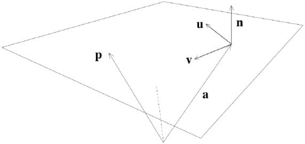
In a high-dimensional Euclidean space, a linear separator is determined by a hyperplane that can be specified as the set of points given by

$$\mathbf{p} = \mathbf{a} + s\mathbf{u} + t\mathbf{v}, \qquad s, t \in \mathbb{R},$$

where $\mathbf{a}$ is a vector from origin to a point in the plane, and $\mathbf{u}$ and $\mathbf{v}$ denote two non-parallel directions in the plane.

Alternatively, the points can be specified as

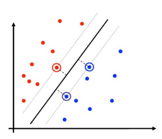
$$(\mathbf{p} - \mathbf{a}) \cdot \mathbf{n} = 0 \Leftrightarrow \mathbf{p} \cdot \mathbf{n} = \mathbf{a} \cdot \mathbf{n},$$



where $\mathbf{n}$ is the norm vector, which is also noted as $\mathbf{w}$, and the dot product $\mathbf{a} \cdot \mathbf{n}$ can be treated as a scalar to be specified, donated as the offset b.

So, the hyperplane can be represented as $\mathbf{w}^{\mathrm{T}}\mathbf{x} + b = 0$.

Hard-SVM



Suppose we are given a set of training samples $\{(\mathbf{x}_1, y_1), \cdots, (\mathbf{x}_N, y_N)\}$ that are linearly separable, the distance between the data point $\mathbf{x}_n$ with class label $y_n \in \{+1, -1\}$ and the hyperplane $\mathbf{w}^{\mathrm{T}}\mathbf{x} + b = 0$ is

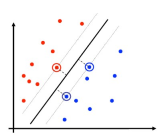
$$\frac{|\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b|}{\|\mathbf{w}\|},$$

where $\|\cdot\|$ is the 2-norm of a vector.

As we are only interested in the linear separators for which all data points are correctly classified, implying that $\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b \geq 0$ for $y_n = +1$ and $\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b < 0$ for $y_n = -1$, i.e., $y_n(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b) \geq 0$, the distance between the data point $\mathbf{x}_n$ and the hyperplane can be rewritten as

$$\frac{y_n(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b)}{\|\mathbf{w}\|}.$$

Hard-SVM



The margin between the hyperplane and all training samples is determined by the data points closest to the hyperplane, whose value is calculated as

$$\min_n \frac{y_n(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b)}{\|\mathbf{w}\|}.$$

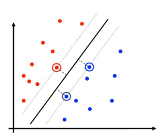
The hyperplane corresponding to the largest margin can be specified as

$$\arg\max_{\mathbf{w},b} \left\{ \frac{1}{\|\mathbf{w}\|} \min_n \left[y_n \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b \right) \right] \right\},$$

where we have taken the factor $1/\|\mathbf{w}\|$ outside the optimization over n because $\mathbf{w}$ does not depend on n.

Direct solution of this optimization problem would be very complex, and so we shall convert it into an equivalent problem that is much easier to solve.

Hard-SVM



We note that if we make the rescaling $\mathbf{w} \to \kappa \mathbf{w}$ and $b \to \kappa b$, the distance between any point $\mathbf{x}_n$ to the hyperplane, given by $y_n(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b)/\|\mathbf{w}\|$, remains unchanged. We can use this freedom to set

$$y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) = 1$$

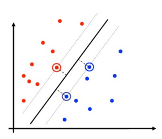
for the point that is closest to the hyperplane. In this case, all data points will satisfy the constraints

$$y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geqslant 1, \quad n = 1, \cdots, N.$$

In the case of data points for which the equality holds, the constraints are said to be active, whereas for the remainder they are said to be inactive.

By definition, there will always be at least one active constraint, because there will always be a closest point, and once the margin has been maximized there will be at least two active constraints.

Hard-SVM



The optimization problem then simply requires that we maximize $\|\mathbf{w}\|^{-1}$, which is equivalent to minimizing $\|\mathbf{w}\|^2$, and so we have to solve the optimization problem

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \quad \text{s.t.} \quad y_n \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geqslant 1, \quad n = 1, \cdots, N.$$

The factor of $1/2$ is included for mathematical convenience. This is a quadratic programming problem in which we are trying to minimize a quadratic function subject to a set of linear inequality constraints.

Soft-SVM

The Hard-SVM formulation assumes that the training set is linearly separable, which is a rather strong assumption.

Soft-SVM can be viewed as a relaxation of the Hard-SVM rule that can be applied even if the training set is not linearly separable.

The optimization problem for Hard-SVM enforces the hard constraints $y_n \left(\mathbf{w}^{\mathrm{T}} \mathbf{x}_n + b\right) \geqslant 1$ for all n.

A natural relaxation is to allow the constraint to be violated for some of the examples in the training set.

This can be modeled by introducing nonnegative slack variables, $\xi_1, \cdots, \xi_N$, and replacing each constraint $y_n \left(\mathbf{w}^{\mathrm{T}} \mathbf{x}_n + b\right) \geqslant 1$ by the constraint $y_n \left(\mathbf{w}^{\mathrm{T}} \mathbf{x}_n + b\right) \geqslant 1 - \xi_n$.

That is, ξ_n measures by how much the constraint $y_n \left(\mathbf{w}^{\mathrm{T}} \mathbf{x}_n + b\right) \geqslant 1$ is being violated.

Soft-SVM

Soft-SVM jointly minimizes the norm of **w** (corresponding to the margin) and the average of ξ_n (corresponding to the violations of the constraints).

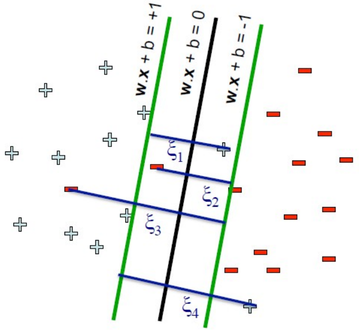
The tradeoff between the two terms is controlled by a parameter C. The Soft-SVM optimization problem can be formulated as

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n$$

$$\text{s.t.} \quad y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geq 1 - \xi_n$$

$$\text{and } \xi_n \geq 0,$$

$$n = 1, 2, \cdots, N.$$



Hinge Loss

For the constraint $y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geq 1 - \xi_n$, if $y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geq 1$ is satisfied, the minimization over ξ_n with the constraint $\xi_n \geq 0$ would make $\xi_n = 0$; otherwise, $\xi_n = 1 - y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)$. In this sense, ξ_n can be expressed as

$$\xi_n = \max\{0, 1 - y(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b)\}.$$

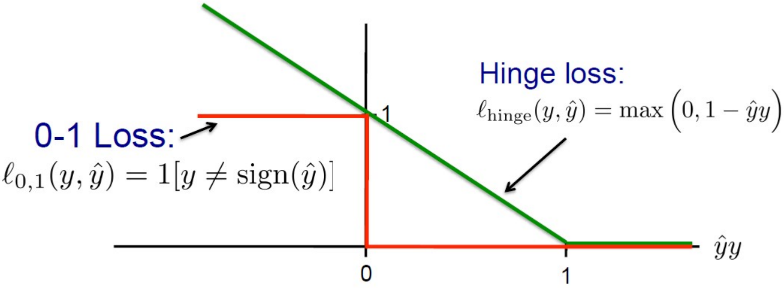
Then the Soft-SVM optimization problem can be re-formulated as

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N} \max\{0, 1 - y(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b)\}.$$

By denoting $\hat{y}_n = \mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b$, ξ_n defines the hinge loss to measure the discrepancy between the prediction $\hat{y}_n$ and the ground truth y_n:

$$\ell_{\text{hinge}}\left(y_n, \hat{y}_n\right) = \max(0, 1 - \hat{y}_n y_n).$$

Hinge Loss



Hinge loss upper bounds 0-1 loss!

It is the tightest *convex* upper bound on the 0-1 loss.

Dual Form for Hard-SVM

Given the Hard-SVM's primal optimization problem

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|^2 \quad s.t. \quad y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geqslant 1, \quad n = 1, \cdots, N,$$

which might be difficult to solve. We can derive its equivalent dual form, by starting with constructing a function

$$\begin{aligned} g(\mathbf{w}, b) &= \max_{\boldsymbol{\alpha}\in\mathbb{R}^N:\boldsymbol{\alpha}\geq\mathbf{0}} \sum_{n=1}^{N} \alpha_n \left\{1 - y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} \\ &= \begin{cases} 0 & \text{if } \forall n, y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geq 1 \\ \infty & \text{otherwise} \end{cases}. \end{aligned}$$

Dual Form for Hard-SVM

The Hard-SVM's primal optimization problem can be re-formulated as

$$\min_{\mathbf{w},b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + g(\mathbf{w}, b) \right\}, \text{ i.e.,}$$

$$\min_{\mathbf{w},b} \max_{\boldsymbol{\alpha}\in\mathbb{R}^N:\boldsymbol{\alpha}\geq\mathbf{0}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} \alpha_n \left\{1 - y_n \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} \right\}.$$

Now suppose that we flip the order of min and max in the above equation. This can only decrease the objective value, and we have

$$\begin{aligned} &\min_{\mathbf{w},b} \max_{\boldsymbol{\alpha}\in\mathbb{R}^N:\boldsymbol{\alpha}\geq\mathbf{0}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} \alpha_n \left\{1 - y_n \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} \right\} \\ &\geq \max_{\boldsymbol{\alpha}\in\mathbb{R}^N:\boldsymbol{\alpha}\geq\mathbf{0}} \min_{\mathbf{w},b} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} \alpha_n \left\{1 - y_n \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} \right\}. \end{aligned}$$

Dual Form for Hard-SVM

The inequality ($\geq$) is called weak duality. It turns out that in our case, strong duality also holds; namely, the inequality holds with equality. Therefore, the dual problem is

$$\max_{\boldsymbol{\alpha} \in \mathbb{R}^N : \boldsymbol{\alpha} \geq \mathbf{0}} \min_{\mathbf{w}, b} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + \sum_{n=1}^{N} \alpha_n \left\{ 1 - y_n \left(\mathbf{w}^{\mathrm{T}} \mathbf{x}_n + b \right) \right\} \right\}.$$

We can simplify the dual problem by noting that once $\boldsymbol{\alpha}$ is fixed, the optimization problem with respect to $\mathbf{w}$ is unconstrained and the objective is differentiable; thus, at the optimum, the gradient equals zero:

$$\mathbf{w} - \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n = 0 \Rightarrow \mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n.$$

Dual Form for Hard-SVM

However, as $\frac{1}{2}\|\mathbf{w}\|^2 + \sum_{n=1}^{N} \alpha_n \left\{1 - y_n \left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\}$ is a linear function of b with coefficient $-\sum_{n=1}^{N} \alpha_n y_n$, its minimum value would be $-\infty$ for all $\boldsymbol{\alpha}$ except for the $\boldsymbol{\alpha}$ satisfying

$$\sum_{n=1}^{N} \alpha_n y_n = 0,$$

which is a necessary condition to make sure the maximization dual problem has a valid optimal solution.

The Hard-SVM's dual optimization problem can be finally formulated as

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \left\{-\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m + \sum_{n=1}^{N}\alpha_n\right\}$$

$$\text{s.t. } \alpha_n \geq 0 \text{ for } n = 1, 2, \cdots, N \text{ and } \sum_{n=1}^{N} \alpha_n y_n = 0.$$

Dual Form for Hard-SVM

Once we find the optimal solution $\boldsymbol{\alpha}^*$ to the dual optimization problem, how shall we convert it to the optimal solution $\mathbf{w}^*$ and b^* to the primal problem?

The optimal $\mathbf{w}^*$ is straightforward with $\mathbf{w}^* = \sum_{n=1}^{N} \alpha_n^* y_n \mathbf{x}_n$.

The optimal b^* is a bit difficult to be derived. At the optimal solution with $\mathbf{w}^*$ and $\boldsymbol{\alpha}^*$ having been specified, b^* should be the optimal solution to the optimization problem

$$\min_{b} \left\{ \frac{1}{2} \|\mathbf{w}^*\|^2 + \sum_{n=1}^{N} \alpha_n^* \left\{ 1 - y_n \left({\mathbf{w}^*}^{\mathrm{T}} \mathbf{x}_n + b \right) \right\} \right\}.$$

If $\alpha_n^* > 0$ for only one n, the value of $\alpha_n^* \left\{ 1 - y_n \left({\mathbf{w}^*}^{\mathrm{T}} \mathbf{x}_n + b \right) \right\}$ should be minimized with the optimal b^*.

As it is impossible for $1 - y_n \left({\mathbf{w}^*}^{\mathrm{T}} \mathbf{x}_n + b^* \right) < 0$ (otherwise α_n^* would be zero for the optimality), b^* should satisfy $1 - y_n \left({\mathbf{w}^*}^{\mathrm{T}} \mathbf{x}_n + b^* \right) = 0$.

Dual Form for Hard-SVM

More rigorously, it can be proved that

1. There exists $\alpha_n^* > 0$, and
2. $1 - y_n\left(\mathbf{w}^{*\mathrm{T}}\mathbf{x}_n + b^*\right) = 0$ for all $\alpha_n^* > 0$ (KKT condition).

The data point $\mathbf{x}_n$ with $\alpha_n > 0$ are termed as support vectors. Multiplying the left and right sides of the equations $1 - y_n\left(\mathbf{w}^{*\mathrm{T}}\mathbf{x}_n + b^*\right) = 0$ by y_n, we can solve b^* as

$$
\begin{aligned}
b^* &= \frac{\sum_{n=1}^N \mathbb{I}(\alpha_n^* > 0) \cdot \left(y_n - \mathbf{w}^{*\mathrm{T}}\mathbf{x}_n\right)}{\sum_{n=1}^N \mathbb{I}(\alpha_n^* > 0)} \\
&= \frac{\sum_{n=1}^N \mathbb{I}(\alpha_n^* > 0) \cdot \left(y_n - \sum_{m=1}^N \alpha_m^* y_m \mathbf{x}_m^{\mathrm{T}}\mathbf{x}_n\right)}{\sum_{n=1}^N \mathbb{I}(\alpha_n^* > 0)},
\end{aligned}
$$

where $\mathbb{I}(\alpha_n^* > 0)$ is an indicator function, whose value is 1 if $\alpha_n^* > 0$, and 0 otherwise.

Dual Form for Hard-SVM

Given the feature vector of a new test sample $\mathbf{x}$, its label $\hat{y}$ can be predicted as

$$\hat{y} = \begin{cases} +1 & \text{if } \ \mathbf{w}^{*\mathrm{T}}\mathbf{x} + b^* \geq 0 \\ -1 & \text{if } \ \mathbf{w}^{*\mathrm{T}}\mathbf{x} + b^* < 0 \end{cases}.$$

Equivalently,

$$\hat{y} = \begin{cases} +1 & \text{if } \ \sum_{n=1}^{N} \alpha_n^* y_n \mathbf{x}_n^{\mathrm{T}}\mathbf{x} + b^* \geq 0 \\ -1 & \text{if } \ \sum_{n=1}^{N} \alpha_n^* y_n \mathbf{x}_n^{\mathrm{T}}\mathbf{x} + b^* < 0 \end{cases}.$$

Dual Form for Soft-SVM

Given the Soft-SVM's primal optimization problem

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n$$
$$\text{s.t.} \quad y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geq 1 - \xi_n \text{ and } \xi_n \geq 0,$$
$$n = 1, 2, \cdots, N.$$

We can also derive its equivalent dual form, by starting with constructing a function

$$\begin{aligned} g(\mathbf{w}, b, \boldsymbol{\xi}) &= \max_{\boldsymbol{\alpha}\geq\mathbf{0},\boldsymbol{\mu}\geq\mathbf{0}} \left\{\sum_{n=1}^{N}\alpha_n\left\{1 - \xi_n - y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} - \sum_{n=1}^{N}\mu_n\xi_n\right\} \\ &= \begin{cases} 0 & \text{if } \forall n, y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right) \geq 1 - \xi_n \text{ and } \xi_n \geq 0 \\ \infty & \text{otherwise} \end{cases}. \end{aligned}$$

Dual Form for Soft-SVM

The Soft-SVM's primal optimization problem can be re-formulated as

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n + g(\mathbf{w},b,\boldsymbol{\xi}) \right\}, \text{ i.e.,}$$

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \max_{\boldsymbol{\alpha}\geq\mathbf{0},\boldsymbol{\mu}\geq\mathbf{0}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n + \sum_{n=1}^{N}\alpha_n\left\{1-\xi_n-y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n+b\right)\right\} - \sum_{n=1}^{N}\mu_n\xi_n \right\}.$$

Now suppose that we flip the order of min and max in the above equation. This can only decrease the objective value, and we have

$$\min_{\mathbf{w},b,\boldsymbol{\xi}} \max_{\boldsymbol{\alpha}\geq\mathbf{0},\boldsymbol{\mu}\geq\mathbf{0}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n + \sum_{n=1}^{N}\alpha_n\left\{1-\xi_n-y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n+b\right)\right\} - \sum_{n=1}^{N}\mu_n\xi_n \right\}$$

$$\geq$$

$$\max_{\boldsymbol{\alpha}\geq\mathbf{0},\boldsymbol{\mu}\geq\mathbf{0}} \min_{\mathbf{w},b,\boldsymbol{\xi}} \left\{ \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^{N}\xi_n + \sum_{n=1}^{N}\alpha_n\left\{1-\xi_n-y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n+b\right)\right\} - \sum_{n=1}^{N}\mu_n\xi_n \right\}.$$

Dual Form for Soft-SVM

The inequality ($\geq$) is called weak duality. It turns out that in our case, strong duality also holds; namely, the inequality holds with equality. Therefore, the dual problem is

$$\max_{\boldsymbol{\alpha} \geq \mathbf{0}, \boldsymbol{\mu} \geq \mathbf{0}} \min_{\mathbf{w}, b, \boldsymbol{\xi}} \left\{ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{n=1}^{N} \xi_n + \sum_{n=1}^{N} \alpha_n \left\{ 1 - \xi_n - y_n \left(\mathbf{w}^{\mathrm{T}} \mathbf{x}_n + b \right) \right\} - \sum_{n=1}^{N} \mu_n \xi_n \right\}.$$

We can simplify the dual problem by noting that once $\boldsymbol{\alpha}$ and $\boldsymbol{\mu}$ are fixed, the optimization problem with respect to $\mathbf{w}$ is unconstrained and the objective is differentiable; thus, at the optimum, the gradient equals zero:

$$\mathbf{w} - \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n = 0 \Rightarrow \mathbf{w} = \sum_{n=1}^{N} \alpha_n y_n \mathbf{x}_n.$$

Dual Form for Soft-SVM

$\left\{\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^N \xi_n + \sum_{n=1}^N \alpha_n \left\{1 - \xi_n - y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} - \sum_{n=1}^N \mu_n \xi_n\right\}$ is a linear function of b with coefficient $-\sum_{n=1}^N \alpha_n y_n$, whose minimum value would be $-\infty$ for all $\boldsymbol{\alpha}$ except for the $\boldsymbol{\alpha}$ satisfying

$$\sum_{n=1}^N \alpha_n y_n = 0.$$

$\left\{\frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{n=1}^N \xi_n + \sum_{n=1}^N \alpha_n \left\{1 - \xi_n - y_n\left(\mathbf{w}^{\mathrm{T}}\mathbf{x}_n + b\right)\right\} - \sum_{n=1}^N \mu_n \xi_n\right\}$ is also a linear function of ξ_n with coefficient $C - \alpha_n - \mu_n$, whose minimum value would be $-\infty$ for all choices of α_n and μ_n except for the choice of α_n and μ_n satisfying

$$C - \alpha_n - \mu_n = 0.$$

Dual Form for Soft-SVM

The Soft-SVM's dual optimization problem can be finally formulated as

$$\max_{\boldsymbol{\alpha},\boldsymbol{\mu}\in\mathbb{R}^N} \left\{ -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m + \sum_{n=1}^{N}\alpha_n \right\}$$

$$\text{s.t. } \alpha_n \geq 0 \text{ for } n = 1, 2, \cdots, N \text{ and } \sum_{n=1}^{N}\alpha_n y_n = 0,$$

$$\mu_n \geq 0 \text{ and } C - \alpha_n - \mu_n = 0 \text{ for } n = 1, 2, \cdots, N.$$

Equivalently, a concise formulation is

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \left\{ -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m + \sum_{n=1}^{N}\alpha_n \right\}$$

$$\text{s.t. } C \geq \alpha_n \geq 0 \text{ for } n = 1, 2, \cdots, N \text{ and } \sum_{n=1}^{N}\alpha_n y_n = 0.$$

Dual Form for Soft-SVM

Given the optimal solution $\boldsymbol{\alpha}^*$ to the Soft-SVM's dual optimization problem, we can recover the optimal solution $\mathbf{w}^*$ and b^* to the primal problem as

$$\mathbf{w}^* = \sum_{n=1}^{N} \alpha_n^* y_n \mathbf{x}_n,$$

$$b^* = \frac{\sum_{n=1}^{N} \mathbb{I}(C > \alpha_n^* > 0) \cdot \left(y_n - \sum_{m=1}^{N} \alpha_m^* y_m \mathbf{x}_m^{\mathrm{T}} \mathbf{x}_n \right)}{\sum_{n=1}^{N} \mathbb{I}(C > \alpha_n^* > 0)},$$

where $\mathbb{I}(C > \alpha_n^* > 0)$ is an indicator function, whose value is 1 if $C > \alpha_n^* > 0$, and 0 otherwise.

Dual Form for Soft-SVM

Given the feature vector of a new test sample $\mathbf{x}$, its label $\hat{y}$ can be predicted as

$$\hat{y} = \begin{cases} +1 & \text{if } \ {\mathbf{w}^*}^{\mathrm{T}}\mathbf{x} + b^* \geq 0 \\ -1 & \text{if } \ {\mathbf{w}^*}^{\mathrm{T}}\mathbf{x} + b^* < 0 \end{cases}.$$

Equivalently,

$$\hat{y} = \begin{cases} +1 & \text{if } \ \sum_{n=1}^{N} \alpha_n^* y_n \mathbf{x}_n^{\mathrm{T}}\mathbf{x} + b^* \geq 0 \\ -1 & \text{if } \ \sum_{n=1}^{N} \alpha_n^* y_n \mathbf{x}_n^{\mathrm{T}}\mathbf{x} + b^* < 0 \end{cases}.$$

Kernel Methods

Kernel SVM

Recall the Soft-SVM's dual optimization problem formulation

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \left\{ -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m \mathbf{x}_n^{\mathrm{T}}\mathbf{x}_m + \sum_{n=1}^{N}\alpha_n \right\}$$

$$\text{s.t.}\;\; C \geq \alpha_n \geq 0 \text{ for } n = 1, 2, \cdots, N \text{ and } \sum_{n=1}^{N}\alpha_n y_n = 0,$$

where the training samples' feature vectors $\mathbf{x}_n$ could be replaced by the transformed feature vectors $\phi(\mathbf{x}_n)$ through a non-linear basis function $\phi(\cdot)$:

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \left\{ -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m \phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m) + \sum_{n=1}^{N}\alpha_n \right\}$$

$$\text{s.t.}\;\; C \geq \alpha_n \geq 0 \text{ for } n = 1, 2, \cdots, N \text{ and } \sum_{n=1}^{N}\alpha_n y_n = 0.$$

Kernel Trick

The dot product $\phi(\mathbf{x}_n)^{\mathrm{T}}\phi(\mathbf{x}_m)$ can be extended to any kernel functions.

For all $\mathbf{x}$ and $\mathbf{x}'$ in the input space $\mathcal{X}$, the function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a kernel function, if there exists a feature map $\varphi : \mathcal{X} \to \mathcal{V}$ that satisfies

$$k\left(\mathbf{x}, \mathbf{x}'\right) = \left\langle \varphi(\mathbf{x}), \varphi\left(\mathbf{x}'\right)\right\rangle_{\mathcal{V}},$$

where $\mathcal{V}$ is a inner product space, and $\langle \cdot, \cdot \rangle_{\mathcal{V}}$ is the inner product operation defined by $\mathcal{V}$.

An alternative definition can be formulated by the *positive semidefinite (PSD)* property: For any points $(\mathbf{x}_1, \ldots, \mathbf{x}_N)$ in $\mathcal{X}$, and all choices of n real-valued coefficients $(c_1, \ldots, c_N)$, the function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is a kernel function, if

$$\sum_{n=1}^{N} \sum_{m=1}^{N} k\left(\mathbf{x}_n, \mathbf{x}_m\right) c_n c_m \geq 0.$$

The spanned matrix, $\boldsymbol{K} \in \mathbb{R}^N \times \mathbb{R}^N$ with its nm-th entry $K_{nm} = k(\mathbf{x}_n, \mathbf{x}_m)$, is called Gram matrix.

Kernel Trick

Many kernels can be chosen for various application scenarios

- Fisher kernel
- Polynomial kernel
- Radial basis function kernel (RBF)
- String kernels
- Graph kernels

The Radial basis function kernel (RBF) is also called squared-exp or Gaussian kernel, which is formulated as

$$k_{\mathrm{SE}}\left(\mathbf{x}_n, \mathbf{x}_m\right) = \sigma_f^2 \exp\left(-\frac{\|\mathbf{x}_n - \mathbf{x}_m\|_2^2}{2\ell^2}\right).$$

Kernel SVM

The kernel version of Soft-SVM's dual optimization problem is

$$\max_{\boldsymbol{\alpha}\in\mathbb{R}^N} \left\{ -\frac{1}{2}\sum_{n=1}^{N}\sum_{m=1}^{N} \alpha_n\alpha_m y_n y_m k(\mathbf{x}_n, \mathbf{x}_m) + \sum_{n=1}^{N}\alpha_n \right\}$$

$$\text{s.t. } C \geq \alpha_n \geq 0 \text{ for } n = 1, 2, \cdots, N \text{ and } \sum_{n=1}^{N}\alpha_n y_n = 0.$$

Kernel SVM

With the optimal solution $\boldsymbol{\alpha}^*$, for a new test sample with feature vector $\mathbf{x}$, it labeled can be predicted as

$$\hat{y} = \begin{cases} +1 & \text{if } \sum_{n=1}^{N} \alpha_n^* y_n k(\mathbf{x}, \mathbf{x}_n) + b^* \geq 0 \\ -1 & \text{if } \sum_{n=1}^{N} \alpha_n^* y_n k(\mathbf{x}, \mathbf{x}_n) + b^* < 0 \end{cases},$$

where

$$b^* = \frac{\sum_{n=1}^{N} \mathbb{I}(C > \alpha_n^* > 0) \cdot \left\{ y_n - \sum_{m=1}^{N} \alpha_m^* y_m k(\mathbf{x}_m, \mathbf{x}_n) \right\}}{\sum_{n=1}^{N} \mathbb{I}(C > \alpha_n^* > 0)}.$$

References

- Christopher M. Bishop. **Pattern Recognition and Machine Learning**. New York: Springer; 2006 Aug 17. (Chapter 7.1 Maximum Margin Classifiers)
- Shai Shalev-Shwartz, and Shai Ben-David. **Understanding Machine Learning: From Theory to Algorithms**. Cambridge university press, 2014. (Chapter 15 Support Vector Machines)
- `https://en.wikipedia.org/wiki/Support_vector_machine`

The End