Laplacian SVM and semi-supervised kernels

Zalán Bodó

1 SVM and LapSVM

Consider the following data setting: $X = X_L \cup X_U = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_\ell\} \cup \{\boldsymbol{x}_{\ell+1}, \ldots, \boldsymbol{x}_{\ell+u=:N}\}$, $Y = \{-1, 1\}$, where X_L is the labeled, X_U is the unlabeled dataset.

For Laplacian SVMs the objective function can be written as

$$\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{l}\xi_i + \gamma_I \boldsymbol{f}'\boldsymbol{L}\boldsymbol{f}$$

such that

$$\begin{aligned} \boldsymbol{f} &= \boldsymbol{\Phi}'\boldsymbol{w} + b \\ y_i(\boldsymbol{w}'\phi(\boldsymbol{x}_i) + b) &\geq 1 - \xi_i, i = 1, \ldots, \ell \end{aligned}$$

where $\boldsymbol{\Phi}$ contains the mapped points (i.e. $\Phi_{\cdot,i} = \phi(\boldsymbol{x}_i)$), $y_i, i = 1, \ldots, \ell$ are the known labels and the parameters $(\boldsymbol{w}, b)$ to be found determine the hyperplane. The above optimization problem can be rewritten as the following minimization (i.e. it can be reduced to standard SVM optimization)

$$\frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{l}\xi_i$$

such that

$$y_i(\boldsymbol{w}'\phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i, i = 1, \ldots, \ell$$

using the following "semi-supervised" kernel ($k(\boldsymbol{x}, \boldsymbol{z}) = \phi(\boldsymbol{x})'\phi(\boldsymbol{z})$):

$$\hat{k}(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{k}_{\boldsymbol{x}}' \left(\frac{1}{2\gamma_I}\boldsymbol{I} + \boldsymbol{L}\boldsymbol{K}\right)^{-1} \boldsymbol{L}\boldsymbol{k}_{\boldsymbol{z}}$$

where

$$\begin{aligned} \boldsymbol{L} &= \boldsymbol{D} - \boldsymbol{W} \text{ (graph Laplacian)} \\ \boldsymbol{k}_x &= [k(\boldsymbol{x}, \boldsymbol{x}_1) \ldots k(\boldsymbol{x}, \boldsymbol{x}_N)]' \\ \boldsymbol{K}_{ij} &= k(\boldsymbol{x}_i, \boldsymbol{x}_j) \end{aligned}$$

2 Proof

To solve the optimization involved in SVMs, we form the Wolfe dual of problem:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & -\frac{1}{2}\boldsymbol{\alpha}'\boldsymbol{B}\boldsymbol{\alpha} + \boldsymbol{\alpha}'\mathbf{1} \\ \text{s.t.} \quad & \boldsymbol{\alpha}'\boldsymbol{y} = 0 \\ & 0 \leq \alpha_i \leq C, i = 1, \ldots, \ell \end{aligned}$$

where $B_{ij} = y_i y_j k(\boldsymbol{x}_i, \boldsymbol{x}_j)$. This is the problem we solve. We show that the objective function of Laplacian SVM ca be written in the same way, only the points have to be transformed using a "semi-supervised" kernel. We start with

$$\begin{aligned} \min_{\boldsymbol{w},b} \quad & \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{\ell}\xi_i + \gamma_I \boldsymbol{w}\boldsymbol{\Phi}'\boldsymbol{L}\boldsymbol{\Phi}\boldsymbol{w} \\ \text{s.t.} \quad & y_i(\boldsymbol{w}'\phi(\boldsymbol{x}_i) + b) \geq 1 - \xi_i \end{aligned}$$

The objective function can be rewritten as

$$\frac{1}{2}\boldsymbol{w}'\underbrace{(\boldsymbol{I} + 2\gamma_I\boldsymbol{\Phi}'\boldsymbol{L}\boldsymbol{\Phi})}_{\boldsymbol{P}}\boldsymbol{w} + C\sum_{i=1}^{\ell}\xi_i$$

Introducing the Lagrange multipliers we rewrite the problem as

$$\begin{aligned} \min_{\boldsymbol{w},b}\max_{\boldsymbol{\alpha},\boldsymbol{\beta}} \quad & \frac{1}{2}\boldsymbol{w}'\boldsymbol{P}\boldsymbol{w} + C\sum_{i=1}^{\ell}\xi_i - \sum_{i=1}^{\ell}\alpha_i(y_i(\boldsymbol{w}'\boldsymbol{x}_i + b) - 1 + \xi_i) - \sum_{i=1}^{\ell}\beta_i\xi_i \\ \text{s.t.} \quad & \alpha_i \geq 0 \\ & \beta_i \geq 0 \end{aligned}$$

We calculate the derivatives and set them equal to zero:

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{w}} &= \boldsymbol{P}\boldsymbol{w} - \sum_{i=1}^{\ell}\alpha_i y_i \boldsymbol{x}_i = 0 \Rightarrow \boldsymbol{w} = \boldsymbol{P}^{-1}\sum_{i=1}^{\ell}\alpha_i y_i \boldsymbol{x}_i \\ \frac{\partial}{\partial \xi_i} &= C - \alpha_i - \beta_i = 0 \Rightarrow 0 \leq \alpha_i \leq C, i = 1, \ldots, \ell \\ \frac{\partial}{\partial b} &= -\sum_{i=1}^{\ell}\alpha_i y_i = 0 \end{aligned}$$

Substituting these back we obtain

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & -\frac{1}{2}\sum_{i,j=1}^{\ell}\alpha_i\alpha_j y_i y_j \boldsymbol{x}_i'\boldsymbol{P}^{-1}\boldsymbol{x}_j + \boldsymbol{\alpha}'\mathbf{1} \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq C \\ & \boldsymbol{\alpha}'\boldsymbol{y} = 0 \end{aligned}$$

Now we decompose $\boldsymbol{P}$ in the following way:

$$\boldsymbol{P} = \boldsymbol{T} \cdot \boldsymbol{T}'$$

where $\boldsymbol{T}$ is obtained by Cholesky decomposition (for positive definite $\boldsymbol{P}$) or SVD ($\boldsymbol{P} = \boldsymbol{T}\boldsymbol{T}' = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{U}' \Rightarrow \boldsymbol{T} = \boldsymbol{U}\boldsymbol{\Sigma}^{1/2}$). Denote $\boldsymbol{v}_i = \boldsymbol{T}^{-1}\phi(\boldsymbol{x}_i)$ and we obtain

$$\begin{array}{ll} \max\limits_{\boldsymbol{\alpha}} & -\frac{1}{2}\boldsymbol{\alpha}'\boldsymbol{S}\boldsymbol{\alpha} + \boldsymbol{\alpha}'\boldsymbol{1} \\ \text{s.t.} & 0 \le \alpha_i \le C \\ & \boldsymbol{\alpha}'\boldsymbol{y} = 0 \end{array}$$

where $S_{ij} = y_i y_j \boldsymbol{v}_i'\boldsymbol{v}_j$. The solution is given by

$$\begin{aligned} \boldsymbol{w}^* &= \boldsymbol{P}^{-1}\sum_{i=1}^{\ell}\alpha_i^* y_i \phi(\boldsymbol{x}_i) \\ b^* &= \frac{1}{\ell}\sum_{j=1}^{\ell}\left(y_j - \sum_{i=1}^{\ell}\alpha_i^* y_i \boldsymbol{v}_i'\boldsymbol{v}_j\right) \\ f(\boldsymbol{x}) &= sign\left(\sum_{i=1}^{\ell}\alpha_i^* y_i \boldsymbol{v}_i'(\boldsymbol{T}^{-1}\phi(\boldsymbol{x})) + b^*\right) \end{aligned}$$

Thus the optimization problem remains the same as for standard SVM, but the points have to be transformed using

$$\boxed{\boldsymbol{v}_i = \boldsymbol{T}^{-1}\phi(\boldsymbol{x}_i)}$$

We apply the Woodbury formula for $\boldsymbol{P}$,

$$(\boldsymbol{A} + \boldsymbol{C}\boldsymbol{B}\boldsymbol{C}')^{-1} = \boldsymbol{A}^{-1} - \boldsymbol{A}^{-1}\boldsymbol{C}(\boldsymbol{B}^{-1} + \boldsymbol{C}'\boldsymbol{A}^{-1}\boldsymbol{C})^{-1}\boldsymbol{C}'\boldsymbol{A}^{-1}$$

obtaining

$$\begin{aligned} \boldsymbol{P}^{-1} &= \boldsymbol{I} - \boldsymbol{\Phi}'\left(\frac{1}{2\gamma_I}\boldsymbol{L}^{-1} + \boldsymbol{\Phi}\boldsymbol{\Phi}'\right)^{-1}\boldsymbol{\Phi} \\ &= \boldsymbol{I} - \boldsymbol{\Phi}'\left(\frac{1}{2\gamma I}\boldsymbol{I} + \boldsymbol{L}\boldsymbol{K}_{NN}\right)^{-1}\boldsymbol{L}\boldsymbol{\Phi} \end{aligned}$$

Here $\boldsymbol{K}_{NN}$ denotes the $N \times N$ kernel matrix, that is for example by the above formula the kernel matrix of the labeled data is

$$\hat{\boldsymbol{K}}_{\ell\ell} = \boldsymbol{K}_{\ell\ell} - \boldsymbol{K}_{\ell N}\left(\frac{1}{2\gamma I}\boldsymbol{I} + \boldsymbol{L}\boldsymbol{K}_{NN}\right)^{-1}\boldsymbol{L}\boldsymbol{K}_{N\ell}$$

From this the "semi-supervised" kernel function for two points becomes

$$\hat{k}(\boldsymbol{x}, \boldsymbol{z}) = k(\boldsymbol{x}, \boldsymbol{z}) - \boldsymbol{k}_{\boldsymbol{x}}'\left(\frac{1}{2\gamma I}\boldsymbol{I} + \boldsymbol{L}\boldsymbol{K}_{NN}\right)^{-1}\boldsymbol{L}\boldsymbol{k}_{\boldsymbol{z}}$$