CS181 Homework 3

RUITAO (TOBY) DU

03/27/2015

1 Composing Kernel Functions

First, we can expand the square,

$$||\boldsymbol{x} - \boldsymbol{x}'||_2^2 = \boldsymbol{x}^T\boldsymbol{x} + (\boldsymbol{x}')^T\boldsymbol{x}' - 2x^T x'$$

Now, we have

$$\begin{aligned} K(\boldsymbol{x}, \boldsymbol{x}') &= \exp\{-\boldsymbol{x}^T\boldsymbol{x} - (\boldsymbol{x}')^T\boldsymbol{x}' + 2x^T x'\} \\ &= \exp\{-\boldsymbol{x}^T\boldsymbol{x}\}exp\{2x^T x'\}exp\{-(\boldsymbol{x}')^T\boldsymbol{x}'\} \end{aligned}$$

We already know that linear kernel $K_1(\boldsymbol{x}, \boldsymbol{x}') = \boldsymbol{x}^T\boldsymbol{x}'$ and according to the properties in the question, $K_2(\boldsymbol{x}, \boldsymbol{x}') = 2\boldsymbol{x}^T\boldsymbol{x}'$ is a valid kernel as well. And then we define $f(\boldsymbol{x})$ as

$$f(\boldsymbol{x}) = \exp\{-\boldsymbol{x}^T\boldsymbol{x}\}$$

Now we can rewrite the kernel as

$$K(\boldsymbol{x}, \boldsymbol{x}') = f(\boldsymbol{x})K_2(\boldsymbol{x}, \boldsymbol{x}')f(\boldsymbol{x}')$$

And according to the property of $K(\boldsymbol{x}, \boldsymbol{x}') = f(\boldsymbol{x})\,K_1(\boldsymbol{x}, \boldsymbol{x}')\,f(\boldsymbol{x}')$, we know that our kernel $K(\boldsymbol{x}, \boldsymbol{x}') = \exp\{-||\boldsymbol{x} - \boldsymbol{x}'||_2^2\}$ is a valid kernel.

2 Slack Variables and Importances

For soft margin case of the support vector machine, our best parameters are

$$
\begin{aligned}
\boldsymbol{w}^*, b^* &= \arg\min_{\boldsymbol{w},b}\{\frac{1}{2}||\boldsymbol{w}||_2^2 + c\sum_{n=1}^{N} r_n\xi_n\} \\
&= \arg\min_{\boldsymbol{w},b}\{\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + c\sum_{n=1}^{N} r_n\xi_n\} \\
&\quad \text{such that } t_n(\boldsymbol{w}^T\boldsymbol{x}_n + b) \geq 1 - \xi_n,\ \xi_n \geq 0 \\
&\quad \text{where } \xi_n = 1 - (\boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x}_n) + b)
\end{aligned}
$$

Therefore, we can use Langrange Duality to rewrite it to

$$
L(\boldsymbol{w}, b, \alpha) = \frac{1}{2}||\boldsymbol{w}||_2^2 + c\sum_{n=1}^{N} r_n\xi_n - \sum_{n=1}^{N}\alpha_n(t_n(\boldsymbol{w}^T\boldsymbol{x}_n + b) + \xi_n - 1) - \sum_{n=1}^{N}\mu_n\xi_n
$$

The dual problem is

$$
\begin{aligned}
\alpha^* &= \arg\min_{\alpha}\max_{\boldsymbol{w},b} L(\boldsymbol{w}, b, \alpha) \\
&= \arg\max_{\alpha}\min_{\boldsymbol{w},b} \frac{1}{2}||\boldsymbol{w}||_2^2 + c\sum_{n=1}^{N} r_n\xi_n - \sum_{n=1}^{N}\alpha_n(t_n(\boldsymbol{w}^T\boldsymbol{x}_n + b) + \xi_n - 1) - \sum_{n=1}^{N}\mu_n\xi_n
\end{aligned}
$$

We take the partial derivative of $\boldsymbol{w}, b, \xi_n$ and set them to zero

$$
\begin{aligned}
\frac{\partial L}{\partial \boldsymbol{w}} = 0 \qquad &\Rightarrow \qquad \boldsymbol{w} = \sum_{n=1}^{N}\alpha_n t_n\boldsymbol{\phi}(\boldsymbol{x}_n) \\
\frac{\partial L}{\partial b} = 0 \qquad &\Rightarrow \qquad \sum_{n=1}^{N}\alpha_n t_n = 0 \\
\frac{\partial L}{\partial \xi_n} = 0 \qquad &\Rightarrow \qquad a_n = c \cdot r_n - \mu_n
\end{aligned}
$$

Using these results to eliminate $\boldsymbol{w}, b, and\xi_n$ from the Lagrangian, we obtain the dual Lagrangian in the form

$$
\begin{aligned}
L(\alpha) = \sum_{n=1}^{N}\alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N}\alpha_n\alpha_{n'}t_n t_{n'}\boldsymbol{\phi}(\boldsymbol{x}_n)\boldsymbol{\phi}(\boldsymbol{x}_{n'}) \\
\text{such that } 0 \leq \alpha_n \leq c \cdot r_n, \sum_{n=1}^{N}\alpha_n t_n = 0
\end{aligned}
$$

3 SVM as Quadratic Program

The origin optimization problem in SVM is

$$
\begin{aligned}
\boldsymbol{w}^*, b^* &= \arg\min_{w,b}\{\frac{1}{2}||\boldsymbol{w}||_2^2\} \\
&= \arg\min_{w,b}\{\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w}\} \\
&\quad \text{such that } t_n(\boldsymbol{w}^T\boldsymbol{\phi}(\boldsymbol{x}_n) + b) \geq 1
\end{aligned}
$$

We take $\boldsymbol{z}^T = [\boldsymbol{w}^T, b]$, so we can rewrite as follow

$$
\begin{aligned}
\boldsymbol{z}^\star &= \arg\min_{z} \frac{1}{2}\boldsymbol{z}^\mathsf{T}\boldsymbol{P}\boldsymbol{z} + \boldsymbol{q}^\mathsf{T}\boldsymbol{z} \\
&\quad \text{such that } \boldsymbol{G}\boldsymbol{z} \geq \boldsymbol{h} \text{ and } \boldsymbol{A}\boldsymbol{z} = \boldsymbol{b}
\end{aligned}
$$

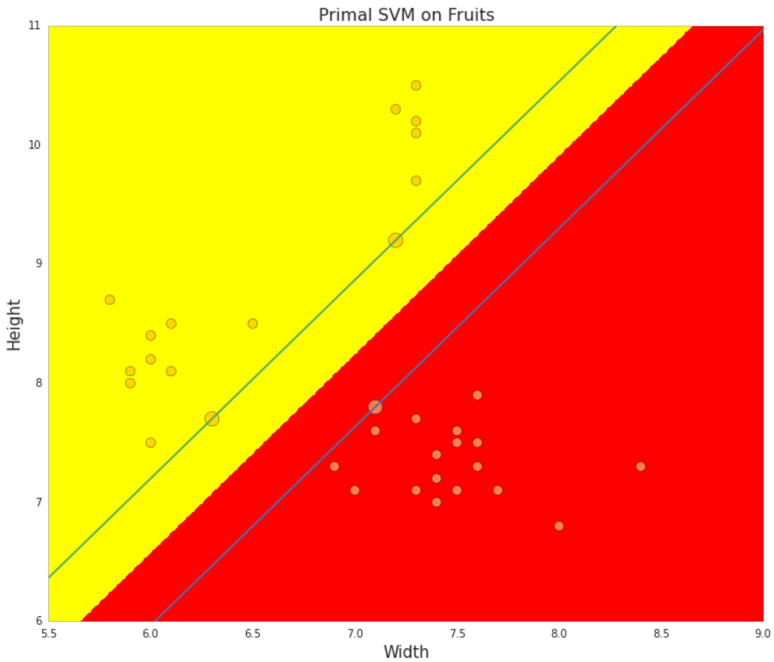
where $\boldsymbol{P} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & \ddots & 0 & \vdots \\ \vdots & 0 & 1 & 0 \\ 0 & \cdots & 0 & 0 \end{bmatrix}$, $\boldsymbol{q} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$, $\boldsymbol{G} = \begin{bmatrix} t_1\boldsymbol{\phi}(\boldsymbol{x}_1)^T & t_1 \\ t_2\boldsymbol{\phi}(\boldsymbol{x}_2)^T & t_2 \\ \vdots & \vdots \\ t_N\boldsymbol{\phi}(\boldsymbol{x}_N)^T & t_N \end{bmatrix}$, $\boldsymbol{h} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}$, $\boldsymbol{A} = \begin{bmatrix} 0 & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & 0 \\ 0 & \cdots & 0 & 0 \\ 0 & \cdots & 0 & 0 \end{bmatrix}$,
$\boldsymbol{b} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$

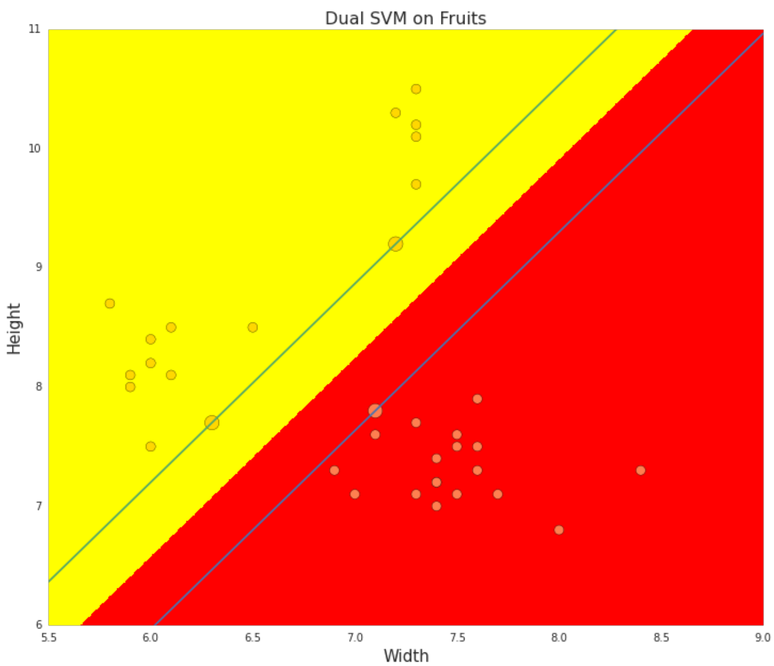
The dual optimization problem in SVM is

$$
\begin{aligned}
\alpha^* &= \arg\max_{\alpha} \sum_{n=1}^{N}\alpha_n - \frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N}\alpha_n\alpha_{n'}t_nt_{n'}\boldsymbol{\phi}(\boldsymbol{x}_n)^T\boldsymbol{\phi}(\boldsymbol{x}_{n'}) \\
&= \arg\min_{\alpha} \frac{1}{2}\sum_{n=1}^{N}\sum_{n'=1}^{N}\alpha_n\alpha_{n'}t_nt_{n'}\boldsymbol{\phi}(\boldsymbol{x}_n)^T\boldsymbol{\phi}(\boldsymbol{x}_{n'}) - \sum_{n=1}^{N}\alpha_n \\
&\quad \text{such that } \alpha_n \geq 0, \sum_{n=1}^{N} t_n\alpha_n = 0 \\
&= \arg\min_{\alpha} \frac{1}{2}\alpha^\mathsf{T}\boldsymbol{P}\alpha + \boldsymbol{q}^\mathsf{T}\alpha \\
&\quad \text{such that } \boldsymbol{G}\alpha \geq \boldsymbol{h} \text{ and } \boldsymbol{A}\alpha = \boldsymbol{b}
\end{aligned}
$$

where $\boldsymbol{P} = \begin{bmatrix} t_0t_0\boldsymbol{\phi}(\boldsymbol{x}_0)^T\boldsymbol{\phi}(\boldsymbol{x}_0) & \cdots & t_0t_N\boldsymbol{\phi}(\boldsymbol{x}_0)^T\boldsymbol{\phi}(\boldsymbol{x}_N) \\ \vdots & \ddots & \vdots \\ t_Nt_0\boldsymbol{\phi}(\boldsymbol{x}_N)^T\boldsymbol{\phi}(\boldsymbol{x}_0) & \cdots & t_Nt_N\boldsymbol{\phi}(\boldsymbol{x}_N)^T\boldsymbol{\phi}(\boldsymbol{x}_N) \end{bmatrix}$, $\boldsymbol{q} = \begin{bmatrix} -1 \\ \vdots \\ -1 \end{bmatrix}$, $\boldsymbol{G} = \begin{bmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{bmatrix}$, $\boldsymbol{h} = \begin{bmatrix} 0 \\ \vdots \\ 0 \end{bmatrix}$,
$\boldsymbol{A} = \begin{bmatrix} t_0 & \cdots & t_N \end{bmatrix}$, $\boldsymbol{b} = \begin{bmatrix} 0 \end{bmatrix}$

4 Classifying Fruit





After implementing the dual version of the problem, from the plots we can see that both decision boundary and margin are the same. We have same solution with primal and dual version of the problem.The support vectors are three points on the margin and are nearest to the decision boundary with larger circles.

5 Calibration

8 hours