SVM for Classification of Spam Email Messages

EE5904/ME5404 Neural Network

Yansong Jia
A0263119H
jiayansong@u.nus.edu

Project I for EE5904/ME5404 Neural Network Part II

Department of Mechanical Engineering
National University of Singapore
2023.4.2

1 Data Pre-processing: Normalization

The SVM model treats all the input variables equally as simple numbers, so it is important to pre-process the input features of emails. The normalization method is used to normalize all the input features of emails to the same range such that the mean is close to zero.

First, calculate the mean of the input data:

$$\bar{x}_i = \frac{\sum_{n=1}^{N} x_i(n)}{N} \tag{1}$$

Second, calculate the standard deviation of the input data:

$$\sigma = \sqrt{\frac{\sum_{n=1}^{N} (x_i(n) - \bar{x}_i)^2}{N}} \tag{2}$$

Finally, use mean and standard deviation to normalize the input and generate the normalized data:

$$x_i'(n) = \frac{(x_i(n) - \bar{x}_i)}{\sigma} \tag{3}$$

The resulting input features now have zero mean and unit variance.

2 Admissibility of the Kernels

Mercer's condition is used to judge whether the kernel is admissible. Mercer's condition is that for training set $S = (x_i, d_i), i = 1, 2, \ldots, N$ the Gram matrix:

$$K = \begin{bmatrix} K(x_1, x_1) & \cdots & K(x_1, x_N) \\ \vdots & \ddots & \vdots \\ K(x_N, x_1) & \cdots & K(x_N, x_N) \end{bmatrix} \in R^{N \times N} \tag{4}$$

K, is positive semi-definite.

In task 2, there are two kernels, linear kernel:

$$K(x_1, x_2) = x_1^T x_2 \tag{5}$$

and polynomial kernel:

$$K(x_1, x_2) = (x_1^T x_2 + 1)^p, p \in [1, 2, 3, 4, 5] \tag{6}$$

After using the *eig* built-in function in MATLAB to calculate the eigenvalues of the Gram matrix, the admissibility of the kernels is shown in Table 1.

Table 1: The Admissibility of Kernels

Type of Kernel	Admissibility
Linear Kernel	Yes
Polynomial Kernel p=1	Yes
Polynomial Kernel p=2	Yes
Polynomial Kernel p=3	Yes
Polynomial Kernel p=4	No
Polynomial Kernel p=5	No

From Table 1, Linear kernel, Polynomial kernels with $p = 1, 2, 3$ are admissible, but Polynomial kernels with $p = 4, 5$ are not admissible.

3 Finding Optimal Hyperplane: Dual Problem

Task 1 is to solve the dual problem and find the optimal hyperplane for different types of SVM models by MATLAB built-in function *quadprog*. The three different dual problem models are subsections as follows.

3.1 Hard-margin Linear Kernel

Kernel:

$$K(x_i, x_j) = x_i^T x_j \tag{7}$$

Given:

$$\mathrm{S}\{(x_i, d_i)\} \tag{8}$$

Find:

$$\text{Lagrange multipliers}\{\alpha_i\} \tag{9}$$

Maximizing:

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) \tag{10}$$

Minimizing:

$$Q(\alpha) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) - \sum_{i=1}^{N} \alpha_i \tag{11}$$

Subject to:

$$\sum_{i=1}^{N} \alpha_i d_i = 0, \alpha_i \geq 0 \tag{12}$$

3.2 Hard-margin Polynomial Kernel

Kernel:

$$K(x_i, x_j) = (x_i^T x_j + 1)^p, p \in [1, 2, 3, 4, 5] \tag{13}$$

Given:

$$\mathrm{S}\{(x_i, d_i)\} \tag{14}$$

Find:

$$\text{Lagrange multipliers}\{\alpha_i\} \tag{15}$$

Maximizing:

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) \tag{16}$$

Minimizing:

$$Q(\alpha) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) - \sum_{i=1}^{N} \alpha_i \tag{17}$$

Subject to:

$$\sum_{i=1}^{N} \alpha_i d_i = 0, \alpha_i \geq 0 \tag{18}$$

3.3 Soft-margin Polynomial Kernel

Kernel:

$$K(x_i, x_j) = (x_i^T x_j + 1)^p, p \in [1, 2, 3, 4, 5] \tag{19}$$

Given:

$$\mathrm{S}\{(x_i, d_i)\} \tag{20}$$

Find:

$$\text{Lagrange multipliers}\{\alpha_i\} \tag{21}$$

Maximizing:

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) \tag{22}$$

Minimizing:

$$Q(\alpha) = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j d_i d_j K(x_i, x_j) - \sum_{i=1}^{N}\alpha_i \tag{23}$$

Subject to:

$$\sum_{i=1}^{N}\alpha_i d_i = 0, 0 \leq \alpha_i \leq C, C \in [0.1, 0.6, 1.1, 2.1] \tag{24}$$

4 Existence of Optimal Hyperplanes

MATLAB built-in function *quadprog* is used to solve the three dual problems in section 3. When the *quadprog* function finds the optimization result that satisfies the constraints, it shows *Minimum found that satisfies the constraints. Optimization completed because the objective function is non-decreasing in feasible directions, to within the value of the optimality tolerance, and constraints are satisfied within the value of the constraint tolerance.* in the terminal.

By MATLAB built-in function *quadprog*'s output, the existence of optimal hyperplanes is shown in Table 2. Hard margin with linear kernel, hard and soft margin with polynomial kernels with $p = 1, 2, 3$ have optimal hyperplanes, but hard or soft margin with polynomial kernels with $p = 4, 5$ do not have optimal hyperplanes.

Table 2: Existence of Optimal Hyperplanes

Type of SVM	Existence of Optimal Hyperplanes			
Hard Margin with Linear Kernel	Yes			
Hard Margin with Polynomial Kernel	$p = 2$	$p = 3$	$p = 4$	$p = 5$
	Yes	Yes	No	No
Soft Margin with Polynomial Kernel	$C = 0.1$	$C = 0.6$	$C = 1.1$	$C = 2.1$
$p = 1$	Yes	Yes	Yes	Yes
$p = 2$	Yes	Yes	Yes	Yes
$p = 3$	Yes	Yes	Yes	Yes
$p = 4$	No	No	No	No
$p = 5$	No	No	No	No

5 Comments on the Results

After getting Lagrange multipliers $\{\alpha_i\}$ by *quadprog* function, determine b_0 in

$$g(x) = \sum_{i=1}^{N}\alpha_i d_i K(x_i, x_j) + b_0 \tag{25}$$

using the fact that for a support vector $x^{(s)}$,

$$g(x^{(s)}) = \pm 1 = d^{(s)} \tag{26}$$

Then we get the discriminant function $g(x)$.

To test the prediction of the SVM model by a test set $S_{test} = \{(x_i, d_i)\}$, first calculate the kernel data $K(x_i, x_{test})$ with the SVM training set, then use:

$$g(x_{test}) = \sum_{i=1}^{N}\alpha_{0,i} d_i K(x_i, x_{test}) + b_0 \tag{27}$$

and

$$d_{test} = sgn[g(x_{new})] \tag{28}$$

Table 3: Results of SVM Classification

Type of SVM	Training Accuracy				Test Accuracy			
Hard Margin with Linear Kernel	93.7%				92.513%			
Hard Margin with Polynomial Kernel	$p=2$	$p=3$	$p=4$	$p=5$	$p=2$	$p=3$	$p=4$	$p=5$
	99.2%	90.25%	None	None	90.365%	86.133%	None	None
Soft Margin with Polynomial Kernel	$C=0.1$	$C=0.6$	$C=1.1$	$C=2.1$	$C=0.1$	$C=0.6$	$C=1.1$	$C=2.1$
$p=1$	93.35%	93.85%	93.7%	93.9%	92.253%	92.578%	92.513%	92.448%
$p=2$	97.6%	98.7%	98.75%	98.75%	92.448%	92.448%	92.318%	92.448%
$p=3$	90.55%	88.65%	88.2%	86.65%	84.766%	83.594%	83.073%	81.836%
$p=4$	None	None	None	None	None	None	None	None
$p=5$	None	None	None	None	None	None	None	None

to get the prediction result of the test set.

Final results of SVM classification on training and test sets are shown in Table 3.

From Table 3, it is obvious that the classification accuracy on the training set is higher than that of the test set. Hard margin with polynomial kernel with $p=2$ performs best on the training test with 99.2% classification accuracy. Soft margin with polynomial kernel with $p=1$ and $C=0.6$ performs best on the test set with 92.578% test accuracy.

Overall, Hard margin with linear kernel and soft margin with polynomial kernels with $p=1,2$ are generally the best choice for classifying spam emails because they perform better than other types of SVM on the test set.

6 Discussion on Design Decisions

Radial basis function (RBF) kernel is also one of the most popular kernels for SVM, and it is also the default type of kernel in the $scikit-learn$ machine learning toolbox.

The RBF kernel is

$$K(x_i, x_j) = e^{-\gamma||x_i - x_j||^2}, \gamma \in [0.01, 0.05, 0.5, 1.0] \tag{29}$$

where γ is the hyper-parameter.

Then the dual problem now becomes:

Kernel:

$$K(x_i, x_j) = e^{-\gamma||x_i - x_j||^2}, \gamma \in [0.01, 0.05, 0.5, 1.0] \tag{30}$$

Given:

$$\mathrm{S}\{(x_i, d_i)\} \tag{31}$$

Find:

$$\text{Lagrange multipliers}\{\alpha_i\} \tag{32}$$

Maximizing:

$$Q(\alpha) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) \tag{33}$$

Minimizing:

$$Q(\alpha) = \frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_i \alpha_j d_i d_j K(x_i, x_j) - \sum_{i=1}^{N} \alpha_i \tag{34}$$

Subject to:

$$\sum_{i=1}^{N} \alpha_i d_i = 0, 0 \leq \alpha_i \leq C, C \in [1.0, 2.0] \tag{35}$$

The classification results of the soft margin RBF kernel with $C \in [1.0, 2.0]$, and $\gamma \in [0.01, 0.05, 0.5, 1.0]$ on the training and test set are shown in

Table 4: Results of RBF Kernel SVM Classification

Type of SVM	Training Accuracy		Test Accuracy	
Soft Margin with RBF Kernel	$C = 1.0$	$C = 2.0$	$C = 1.0$	$C = 2.0$
$\gamma = 0.01$	94.45%	94.9%	93.164%	93.49%
$\gamma = 0.05$	96.4%	97.1%	93.359%	93.68%
$\gamma = 0.5$	99.2%	99.3%	87.174%	87.305%
$\gamma = 1.0$	99.4%	99.75%	85.938%	86.263%

From Table 4, it is obvious that when the $\gamma = 0.05$, the test accuracy is the best, and if γ increases, the training accuracy increases but test accuracy decreases dramatically, which means there is over-fitting. If γ decreases, both training and test accuracy decrease means that it does not reach the best.

Overall, Soft margin RBF kernel with $C = 2.0$ and $\gamma = 0.05$ SVM is chosen to be the designed SVM, and its classification accuracy on the test set, 93.68% is also higher than that of the best soft margin with polynomial kernel SVM in section 5, which is 92.578%.