EE5904/ME5404 Neural Networks Part II Project Report

AY2022/2023, Semester 2

Department of Mechanical Engineering

SVM for Classification of Spam Email Messages

Sun Shaowei

A0263124N

April 2023

1 Data pre-processing

The data pre-processing strategy employed in this work is the standardization of the data. The standardization method is also called Z-score Normalization method, which makes the values of each feature in the data have zero-mean (when subtracting the mean in the numerator) and unit-variance. This policy could avoid the poor performances caused by the individual features deviated from the standard normally distributed data.

$$\text{standardization}: x^* = \frac{x - \bar{x}}{\sigma}$$

$$\text{mean}: \bar{x} = \frac{1}{n}\sum_i^n x_i$$

$$\text{standard deviation}: \sigma = \sqrt{\frac{1}{n}\sum_i^n (\bar{x} - x_i)^2}$$

2 Admissibility of the kernels

To evaluate the admissibility of a kernel after choosing an expression for $K(\cdot,\cdot)$, the benchmark is the Mercer's Condition. If the kernel expression satisfies the condition, this kernel could be treated as a proper strategy to project the original dataset to a high dimensional space.

Mercer's Condition: For training set, $S = (\mathbf{x}_i, d_i), i = 1,2,\cdots,N$, the Gram matrix shown in Figure 1 is positive semi-definite (i.e., its eigenvalues are non-negative).

$$\mathbf{K} = \begin{bmatrix} K(\mathbf{x}_1,\mathbf{x}_1) & \dots & K(\mathbf{x}_1,\mathbf{x}_N) \\ \vdots & \ddots & \vdots \\ K(\mathbf{x}_N,\mathbf{x}_1) & \dots & K(\mathbf{x}_N,\mathbf{x}_N) \end{bmatrix} \in R^{N\times N}$$

Figure 1: Mercer's Condition

In this project, two required kernels are the linear kernel and the polynomial kernel:

(i) A SVM with the linear kernel: $\mathbf{K}(\mathbf{x}_1,\mathbf{x}_2) = \mathbf{x}_1^T\mathbf{x}_2$

(ii) A SVM with a polynomial kernel: $\mathbf{K}(\mathbf{x}_1,\mathbf{x}_2) = (\mathbf{x}_1^T\mathbf{x}_2 + 1)^p$ Where the values of p and the admissibility are both listed in Table 1.

Type of SVM	Admissibility of the kernels			
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

3 Existence of optimal hyperplanes

The typical primal problem for the SVM with hard margin should satisfy the following equations.

Given data set: $S = \{(\mathbf{x}_i, d_i)\}$

Find: $\mathbf{w}$ and b

Minimizing: $f(\mathbf{w}) = \frac{1}{2}\mathbf{w}^T\mathbf{w}$

Subject to: $d_i(\mathbf{w}^T\mathbf{x}_i + b) \geq 1$

The soft margin SVM follows a similar optimization procedure with a couple of differences. First, in this scenario, this strategy allows misclassifications to happen. The corresponding misclassification error should be minimized by defining a loss function. A common loss function

used for soft margin is the hinge loss.

$$\max\{0,1 - y_i(\boldsymbol{w}^T \boldsymbol{x}_i + b)\}$$

The loss of a misclassified point is called a slack variable and is added to the primal problem that we had for hard margin SVM. For the primal problem, optimal hyperplane with soft margin should satisfy the following restrictions.

$$\begin{aligned} \text{Minimize:} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{N} \xi_i \\ \text{Subject to:} \quad & d_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i \geq 0 \\ & \xi_i \geq 0 \end{aligned}$$

Apply the KKT conditions to reduce the unknows in the primal problem to just the Lagrange multipliers α_i to form the dual problem. Both the soft margin and hard margin share the same formula $Q(\boldsymbol{\alpha})$. But they have one difference: for hard margin the range of Lagrange multipliers is 0 to an infinite number; for soft margin the range is 0 to a user-decided number C.

$$\text{Maximize :} \quad Q(\boldsymbol{\alpha}) = \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j d_i d_j K(\mathbf{x}_i, \mathbf{x}_j)$$

$$\text{Subject to:} \quad \sum_{i=1}^{N}\alpha_i d_i = 0, 0 \leq \alpha_i \leq C$$

After solving of these dual problems, according to the built-in function *quadprog*() of MATLAB, optimization completed sign satisfies the following requirement: "*the objective function is non-decreasing in feasible directions, to within the value of the optimality tolerance, and constraints are satisfied to within the value of the constraint tolerance.*"

Type of SVM	Existence of optimal hyperplanes			
Hard margin with linear kernel	No			
Hard margin with polynomial kernel	P=2	P=3	P=4	P=5
	Yes	Yes	None	None
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	None	None	None	None
P=5	None	None	None	None

Only the SVM with hard margin using linear kernel could not find the optimal hyperplanes. The response from the *quadprog*() of Matlab is "*quadprog stopped because it exceeded the iteration limit*".

4 Comments on results

After the implementation of codes, the corresponding $\boldsymbol{\alpha}$ value and bias item can be obtained. The the predicted label of training set and testing set is acquired accordingly. The results under each condition are collected in the Table 2 below.

Type of SVM	Training accuracy				Testing accuracy			
Hard margin with linear kernel	93.9%				92.77%			
Hard margin with polynomial kernel	P=2	P=3	P=4	P=5	P=2	P=3	P=4	P=5
	99.8%	81.75%	None	None	91.01%	77.87%	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%	92.25%	92.58%	92.51%	92.45%
P=2	97.6%	98.7%	98.75%	98.75%	92.45%	92.45%	92.32%	92.45%
P=3	90.55%	88.65%	88.2%	86.65%	84.77%	83.59%	83.07%	81.84%

P=4	None	None	None	None	None	None	None	None
P=5	None	None	None	None	None	None	None	None

(1) For the training dataset, the accuracy is always higher than that of the testing dataset. This could be easily understood because the weights are directly adapted by the training dataset. The possible over-fitting training would make the SVM lack the predictability.

(2) The SVM of hard margin with linear kernel has quite good performances beyond expectations. Although it could not find the optimal hyperplane, it avoids the over-fitting problem at the same time.

(3) The SVM of hard margin with polynomial kernel performs well at p=2 and the performances deteriorate at p=3. This contrast could verify that at higher-dimensional space, polynomial kernel with p=2 is more likely to make dataset separatable.

(4) The SVM of soft margin with polynomial kernel performs well at p=1 and p=2. Actually, the soft margin doesn't contribute a lot in the cases of these two kernels because the results from the SVM of hard margin with polynomial kernel have indicated that training data is separable. The contributions of soft margin play the role at p=3. Soft margin allows the misclassifications and decrease the inefficiencies brought by the kernel.

5 Discussion on design decisions

Gaussian RBF (Radial Basis Function) is another popular Kernel method used in SVM models for more. RBF kernel is a function whose value depends on the distance from the origin or from some point. Gaussian Kernel is of the following format:

$$K(X_1, X_2) = \text{ exponent } \left(-\gamma \| X_1 - X_2 \|^2\right)$$

$\| X1 - X2 \|$ = Euclidean distance between $X1$&$X2$

γ is decided by σ according to the formula:

$$\gamma = \frac{1}{2\sigma^2}$$

If γ is set to be a large number, it will easily lead to the over-fitting problem. Because σ will be very small for a large γ and the Gaussian distribution curve will become tall and thin. In other words, for a single data point, it would only be imposed by the nearest neighbors rather than the whole training dataset. Conversely, the small value of γ would lead to the under-fitting problem. Every data point would be imposed by the whole dataset and lose the features held by itself.

The best couple is found to be $C = 50$ and $\gamma = 0.1$. The regularization parameter and the RBF ensure the high accuracies in both training and testing dataset.

$C = 50$ and $\gamma = 0.1$	Training	Testing
Accuracy	99.55%	93.22%