CM146, Winter 2019
Problem Set 3: SVM and Kernels
Due Mar 2, 2019

Jingjing Nie

1 Problem 1

(a) Problem 1a

Solution:

Assume there are a unique words in x, b unique words in z, and c unique words in both x and z. Thus, the matrix representing this example would be:

$$\begin{bmatrix} a & c \\ c & b \end{bmatrix}$$

Then, to find the eigenvalues of this matrix,

$$det \begin{bmatrix} a-\lambda & c \\ c & b-\lambda \end{bmatrix} = (a-\lambda)(b-\lambda) = 0$$

Thus, $\lambda_1 = a$, $\lambda_2 = b$.
Since the number of unique words in x and z are both non-negative, λ is also non-negative.
So k(x,z) is a kernel.

We can also prove this by showing that this case can be mapped by $\phi(x)^T\phi(z)$ such that using a kernel is valid.

Suppose that there are N total words that might appear in any documents. Then, we can give each element in the list of words $(x_1, x_2, x_3, ...x_N)^T$ a label of either 1, suggesting that this word is in document x, and 0, suggesting that this word is not in x. Similarly, we can do the same thing for a list of words $(z_1, z_2, z_3, ...z_N)^T$ with labels 1 or 0.

Then, to find out the number of unique words that appear in both x and z, we can find the dot product of these two lists.

$\phi(x)^T\phi(z) = (x_1, x_2, x_3, ...x_N)(z_1, z_2, z_3, ...z_N)^T = x_1z_1 + x_2z_2 + ... + x_Nz_N$

Each of the x_iz_j is either 1, suggesting that the word is in both documents, or 0, suggesting that it is not present in both. So the sume will be the number of unique words in both documents.

Therefore, we can then use a kernel for this $\phi(x)^T\phi(z)$ and the kernel is valid.

(b) Problem 1b

Solution:

Let $f(x) = \frac{1}{||x||}$, and $f(z) = \frac{1}{||z||}$,
Since $k(x, z) = x \cdot z$ is a kernel, we can apply the scaling rule:

$$k(x, z) = f(x)k_1(x, z)f(z)$$

is a kernel.
Thus,

$$\left(\left(\frac{x}{||x||}\right)\left(\frac{z}{||z||}\right)\right) = f(x)k(x, z)f(z)$$

is a kernel.
Also, since 1 is both symmetric and positive semi-definite, we can apply the sum rule:

$$k(x, z) = k_1(x, z) + k_2(x, z)$$

is a kernel.
Thus,

$$1 + \left(\left(\frac{x}{||x||}\right)\left(\frac{z}{||z||}\right)\right)$$

is a kernel.
Finally, as the power of three can be regarded as three same items multiplying with each other, we can use the product rule:

$$k(x, z) = k_1(x, z)k_2(x, z)$$

is a kernel.
Therefore,

$$k(x,z) = \left(1 + \left(\left(\frac{x}{||x||}\right)\left(\frac{z}{||z||}\right)\right)\right)^3$$

is a kernel.

(c) Problem 1c

Solution:

Since x and z are in R^2,

$$\begin{aligned}
k_\beta(x,z) &= (1+\beta x\cdot z)^3 \\
&= \Big(1+\beta(x_1z_1+x_2z_2)\Big)^3 \\
&= 1+3\beta(x_1z_1+x_2z_2)+3\beta^2(x_1z_1+x_2z_2)^2+\beta^3(x_1z_1+x_2z_2)^3 \\
&= 1+3\beta(x_1z_1+x_2z_2)+3\beta^2(x_1^2z_1^2+2x_1z_1x_2z_2+x_2^2z_2^2)+ \\
&\beta^3(x_1^3z_1^3+3x_1^2z_1^2x_2z_2+3x_1z_1x_2^2z_2^2+x_2^3z_2^3) \\
&= 1+3\beta x_1z_1+3\beta x_2z_2+3\beta^2x_1^2z_1^2+6\beta^2x_1z_1x_2z_2+3\beta^2x_2^2z_2^2+ \\
&\beta^3x_1^3z_1^3+3\beta^3x_1^2z_1^2x_2z_2+3\beta^3x_1z_1x_2^2z_2^2+\beta^3x_2^3z_2^3
\end{aligned}$$

Therefore, the feature map would be:

$$\phi_\beta(\cdot) = (1, \sqrt{3\beta}x_1, \sqrt{3\beta}x_2, \sqrt{3}\beta x_1^2, \sqrt{6}\beta x_1x_2, \sqrt{3}\beta x_2^2,$$
$$\beta\sqrt{\beta}x_1^3, \beta\sqrt{3\beta}x_1^2x_2, \beta\sqrt{3\beta}x_1x_2^2, \beta\sqrt{\beta}x_2^3)^T$$

Through the comparisons between k(x,z) and $k_\beta(x,z)$, which differs by the constant term β, we can see that the addition of the constant β has slightly changed the weight of the terms with higher powers and the terms with lower powers. For instance, these two kernels will be identical to each other if the value of β is 1, since 1 to the power of any value is still 1. Then, if $\beta < 1$, then $\beta\sqrt{\beta}$ will be greater than β, and β will be greater than $\sqrt{\beta}$. This suggests that the terms with higher orders now have higher weights. On the other hand, is $0 < \beta < 1$, then $\beta\sqrt{\beta}$ will be smaller than β, and β will be smaller than $\sqrt{\beta}$. This suggests that the terms with higher orders now have lower weights.

2 Problem 2

(a) Problem 2a

Solution:

Since we want to minimize $\frac{1}{2}||\theta||^2$ subject to $y_n\theta^T x_n \geq 1, n = 1, ..., N$, we want to solve

$$max\frac{1}{||\theta||}, s.t.min[y_n\theta^T x_n] = 1$$

Since $y = -1, x = (a, e)^T$, Thus,

$$-\theta^T(a, e)^T = 1$$
$$\theta^T(a, e)^T = -1$$

Therefore, θ^* is:

$$\theta^* = -\frac{1}{a^2 + e^2}(a, e)^T$$

(b) Problem 2b

Solution:

Similarly as above, we want to solve:

$$max\frac{1}{||\theta||}, s.t.min[y_n\theta^T x_n] = 1$$

Since $x_1 = (1, 1)^T$, $x_2 = (1, 0)^T$, $y_1 = 1$, and $y_2 = -1$,

$$y_1\theta^T x_1 = 1$$
$$y_2\theta^T x_2 = 1$$
$$\theta^T(1, 1)^T = 1$$
$$-\theta^T(1, 0)^T = 1$$

Thus, $\theta_1 + \theta_2 = 1$, $-\theta_1 = 1$

$$\theta_1 = -1,$$
$$\theta_2 = 2$$
$$\theta^* = (-1, 2)^T$$

The margin is:

$$\gamma = \frac{1}{||\theta||} = \frac{1}{\sqrt{(-1)^2 + 2^2}} = \frac{1}{\sqrt{5}}$$

(c) Problem 2c

Solution:

If b is not zero, then what we want to solve now becomes:

$$y_1(\theta^T x_1 + b) = 1$$
$$y_2(\theta^T x_2 + b) = 1$$

Since $x_1 = (1, 1)^T$, $x_2 = (1, 0)^T$, $y_1 = 1$, and $y_2 = -1$,

$$\theta^T (1, 1)^T + b = 1$$
$$-\theta^T (1, 0)^T - b = 1$$

Thus, $\theta_1 + \theta_2 = 1 - b$, $-\theta_1 = 1 + b$

$$\theta_2 = 2$$

So θ^* can be found as:

$$\theta^* = (0, 2)^T$$
$$b = -1$$

The margin is:

$$\gamma = \frac{1}{||\theta||} = \frac{1}{\sqrt{2^2}} = \frac{1}{2}$$

As a result, since $\frac{1}{2} > \frac{1}{\sqrt{5}}$, the margin we found with offset is larger than the one we found without offset.

3 Problem 3

3.1 Problem 3.1

3.2 Problem 3.2

(a)

(b) **Solution:**

It is beneficial to maintain the class proportions across the folds because if the proportion of positive and negative classes in the test set is pretty different from the one in the training set, then the training set will not be able to properly represent the model, which will then negatively affect the results of the machine learning algorithm.

(c)

(d) **Solution:**

C	Accuracy	F1-score	AUROC	Precision	Sensitivity	Specificity
10^{-3}	0.7089	0.8297	0.8105	0.7089	1.0	0.0
10^{-2}	0.7107	0.8306	0.8111	0.7102	1.0	0.0063
10^{-1}	0.8060	0.8755	0.8576	0.8357	0.9294	0.5081
10^{0}	0.8146	0.8749	0.8712	0.8562	0.9017	0.6045
10^{1}	0.8182	0.8766	0.8696	0.8595	0.9017	0.6167
10^{2}	0.8182	0.8766	0.8696	0.8595	0.9017	0.6167
best C	10, 100	10, 100	1	10, 100	10^{-3}, 10^{-2}	10, 100

As C increases, while accuracy, F1-score, precision, specificity increase, sensitivity decreases. Then, when C reaches 10, the values of all of the parameters except AUROC reach stable values. Thus, the best score for AUROC occurs when C is 1, and the best score for sensitivity occurs when C is 0.001.

3.3 Problem 3.3

(a) **Solution:**

Since the additional hyperparameter γ for an RBF kernel SVM is the inverse of the radius of influence of a training data, when this value is big, then the radius of influence is small, so the influence of the support vector is small. This implies that there might be overfitting problems as the influence might only include itself and thus reguularization will not be able to prevent overfitting. On the other hand, if γ is small, then the influence of the support vector is big, and might include the whole training set. In this case, the model will be too constrained and lose its complexity. Therefore, the behavior of the model is largely dependent on the γ value.

(b) **Solution:**

For the values of C and γ, I choose to set the range for both of them as 0.001, 0.01, 0.1, 1, 10, 100, and then tested on all combinations of these two values. The reason I choose this grid is that it can cover a wide range of hyperparameter settings, which has made it more likely for the best cases to be found.

(c) **Solution:**

metric	score	C	γ
Accuracy	0.8165	100	0.01
F1-score	0.8763	100	0.01
AUROC	0.8746	100	0.01
Precision	0.8583	100	0.01
Sensitivity	1.0	0.001	0.001
Specificity	0.6047	100	0.01

For most of the hyperparameters, the best performance scores are achieved when C is 100 and γ is 0.01. For sensitivity, however, the highest score is achieved when C is 1.0, and the value then dropped as C further increases.

3.4 Problem 3.4

(a) **Solution:**

Based on the results obtained, the hyperparameter settings for the linear-kernel SVM are:

metric	C_linear
Accuracy	10
F1-score	10
AUROC	1
Precision	10
Sensitivity	0.001
Specificity	10

And the hyperparameter settings for the RBF-kernel SVM are:

metric	C_rbf	gamma_rbf
Accuracy	100	0.01
F1-score	100	0.01
AUROC	100	0.01
Precision	100	0.01
Sensitivity	0.001	0.001
Specificity	100	0.01

(b)

(c) **Solution:**

metric	C_linear	score_linear	C_rbf	gamma_rbf	score_rbf
Accuracy	10	0.7429	100	0.01	0.7571
F1-score	10	0.4375	100	0.01	0.4516
AUROC	1	0.7405	100	0.01	0.7328
Precision	10	0.6364	100	0.01	0.7
Sensitivity	0.001	1.0	0.001	0.001	1.0
Specificity	10	0.9184	100	0.01	0.9388

Based on the comparisons between the scores achieved when using linear-kernel SVM and RBF-kernel SVM, the RBF-kernel SVM is a better classifier, since its scores are higher for all hyperparameters except AUROC. For AUROC, the two scores are similar to each other.