Danning Yu, 305087992
CS M146 Discussion 1A
PS3
2/23/2020

1. (a) $k(\mathbf{x}, \mathbf{z})$ is a kernel function because we can represent it with the following matrix, which we will show is positive semi-definite, so by Mercer's theorem, k is a kernel function.

$$\mathbf{K} = \begin{bmatrix} \|\mathbf{x} \cap \mathbf{x}\| & \|\mathbf{x} \cap \mathbf{z}\| \\ \|\mathbf{z} \cap \mathbf{x}\| & \|\mathbf{z} \cap \mathbf{z}\| \end{bmatrix}$$

Note that $\|\mathbf{a} \cap \mathbf{a}\| = \|\mathbf{a}\|$ for any $\mathbf{a}$ and $\|\mathbf{x} \cap \mathbf{z}\| = \|\mathbf{z} \cap \mathbf{x}\|$, so $\mathbf{K}$ reduces to:

$$\mathbf{K} = \begin{bmatrix} \|\mathbf{x}\| & \|\mathbf{x} \cap \mathbf{z}\| \\ \|\mathbf{x} \cap \mathbf{z}\| & \|\mathbf{z}\| \end{bmatrix}$$

The eigenvalues λ of $\mathbf{K}$ are then given by:

$$0 = (\|\mathbf{x}\| - \lambda)(\|\mathbf{z}\| - \lambda) - \|\mathbf{x} \cap \mathbf{z}\|^2$$

$$\|\mathbf{x}\|\|\mathbf{z}\| - \lambda\|\mathbf{z}\| - \lambda\|\mathbf{x}\| + \lambda^2 - \|\mathbf{x} \cap \mathbf{z}\|^2 = 0$$

$$\lambda^2 - \lambda(\|\mathbf{x}\| + \|\mathbf{z}\|) + \|\mathbf{x}\|\|\mathbf{z}\| - \|\mathbf{x} \cap \mathbf{z}\|^2 = 0$$

By property of intersections, we have $\|\mathbf{x} \cap \mathbf{z}\| \le \|\mathbf{x}\|$ and $\|\mathbf{x} \cap \mathbf{z}\| \le \|\mathbf{z}\|$, so thus,

$$\|\mathbf{x} \cap \mathbf{z}\|^2 \le \|\mathbf{x}\|\|\mathbf{z}\| \tag{1}$$

Now use the quadratic formula to find the values of λ:

$$\lambda = \frac{-\left(-(\|\mathbf{x}\| + \|\mathbf{z}\|)\right) \pm \sqrt{\left(-(\|\mathbf{x}\| + \|\mathbf{z}\|)\right)^2 - 4(1)\left(\|\mathbf{x}\|\|\mathbf{z}\| - \|\mathbf{x} \cap \mathbf{z}\|^2\right)}}{2(1)}$$

$$\lambda = \frac{(\|\mathbf{x}\| + \|\mathbf{z}\|) \pm \sqrt{(\|\mathbf{x}\| + \|\mathbf{z}\|)^2 - 4\|\mathbf{x}\|\|\mathbf{z}\| + 4\|\mathbf{x} \cap \mathbf{z}\|^2}}{2(1)}$$

For the positive root case, it is obvious that λ is non-negative. For the negative case, we must show that

$$(\|\mathbf{x}\| + \|\mathbf{z}\|) - \sqrt{(\|\mathbf{x}\| + \|\mathbf{z}\|)^2 - 4\|\mathbf{x}\|\|\mathbf{z}\| + 4\|\mathbf{x} \cap \mathbf{z}\|^2} \ge 0$$

Rearranging and simplifying, we get:

$$(\|\mathbf{x}\| + \|\mathbf{z}\|)^2 \ge (\|\mathbf{x}\| + \|\mathbf{z}\|)^2 - 4\|\mathbf{x}\|\|\mathbf{z}\| + 4\|\mathbf{x} \cap \mathbf{z}\|^2$$

$$4\|\mathbf{x}\|\|\mathbf{z}\| \ge 4\|\mathbf{x} \cap \mathbf{z}\|^2$$

$$\|\mathbf{x}\|\|\mathbf{z}\| \ge \|\mathbf{x} \cap \mathbf{z}\|^2$$

This last inequality is true, by equation (1) above. Thus, all eigenvalues of $\mathbf{K}$ are nonnegative, meaning that $\mathbf{K}$ is positive semi-definite, and thus by the Mercer theorem, k is a kernel function.

1. (b) Given that **x**•**z** is a kernel, by the scaling property, k' is also a kernel, where

$$k'(\mathbf{x}, \mathbf{z}) = \frac{\mathbf{x}\bullet\mathbf{z}}{\|\mathbf{x}\|\|\mathbf{z}\|}$$

In this case, the scaling constants are $\|\mathbf{x}\|^{-1}$ and $\|\mathbf{z}\|^{-1}$, both of which are nonnegative.

Then, because 1 is a kernel function (a possible generating function would be $\phi(\mathbf{x}) = 1$), by the sum property, $1+k'$ is also a kernel.

Then, by repeated application of the product property,

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

is also kernel, as desired.

1. (c) Define the kernel $k_\beta(\mathbf{x},\mathbf{z}) = (1+\beta\mathbf{x}\bullet\mathbf{z})^3$.

Expanding this gives:

$$1+3\beta\mathbf{x}\bullet\mathbf{z}+3\beta^2(\mathbf{x}\bullet\mathbf{z})^2+\beta^3(\mathbf{x}\bullet\mathbf{z})^3$$

$$1+3\beta x_1z_1+3\beta x_2z_2+3\beta^2\left(x_1^2z_1^2+2x_1x_2z_1z_2+x_2^2z_2^2\right)+\beta^3\left(x_1^3x_2^3+3x_1^2x_2^2z_1z_2+3x_1x_2z_1^2z_2^2+z_1^3z_2^3\right)$$

We wish to find a function ϕ such that $\phi(\mathbf{x})^{\mathrm{T}}\phi(\mathbf{z})$ equals the expression above. Taking inspiration from the case for 2nd degree polynomials, we get:

$$\phi = \begin{bmatrix} 1 \\ \sqrt{3\beta}x_1 \\ \sqrt{3\beta}x_2 \\ \sqrt{3}\beta x_1^2 \\ \sqrt{3}\beta x_2^2 \\ \sqrt{6}\beta x_1x_2 \\ \sqrt{\beta^3}x_1^3 \\ \sqrt{\beta^3}x_2^3 \\ \sqrt{3\beta^3}x_1^2x_2 \\ \sqrt{3\beta^3}x_1x_2^2 \end{bmatrix}$$

The similarities are that both involve the dot product of two vectors, **x** and **z**, but the kernel $k(\mathbf{x},\mathbf{z}) = (1+\mathbf{x}\bullet\mathbf{z})^3$ is for the case of $\beta = 1$, in which case the transformation function no longer has the scaling factor β present for each term. Note that as the degree of the term in the transformation vector increases, β increases as well, so it could potentially be a regularization parameter.

2. (a) We use the method of Lagrange multipliers, where:

$$f(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{\theta}\|^2 \text{ and } g(\boldsymbol{\theta}) = y_n \boldsymbol{\theta}^T \mathbf{x}_n \geq 1$$

For a training vector $x = \begin{bmatrix} a & e \end{bmatrix}^T$ with label $y = -1$, the functions above reduce to:

$$f(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{\theta}\|^2 \text{ and } g(\boldsymbol{\theta}) = -\boldsymbol{\theta}^T \mathbf{x}_n \geq 1$$

The Lagrangian is

$$L = \frac{1}{2}\|\boldsymbol{\theta}\|^2 - \lambda\left(-\boldsymbol{\theta}^T \mathbf{x}_n - 1\right) = \frac{1}{2}\|\boldsymbol{\theta}\|^2 + \lambda\left(\boldsymbol{\theta}^T \mathbf{x}_n + 1\right)$$

We wish to minimize this over $\boldsymbol{\theta}$, so take the derivative with respect to $\boldsymbol{\theta}$ and set to 0:

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \boldsymbol{\theta} + \lambda \mathbf{x}_n = 0$$

$$\boldsymbol{\theta}^* = -\lambda \mathbf{x}_n$$

We wish to maximize λ, so substitute in the previous result, take the derivative with respect to λ and set to 0:

$$L = \frac{1}{2}\|-\lambda \mathbf{x}_n\|^2 + \lambda\left(\left(-\lambda \mathbf{x}_n\right)^T \mathbf{x}_n + 1\right)$$

$$L = \frac{1}{2}\lambda^2 \|\mathbf{x}_n\|^2 - \lambda^2 \|\mathbf{x}_n\|^2 + \lambda = -\frac{1}{2}\lambda^2 \|\mathbf{x}_n\|^2 + \lambda$$

$$\frac{\partial L}{\partial \lambda} = -\lambda \|\mathbf{x}_n\|^2 + 1 = 0$$

$$\lambda^* = \frac{1}{\|\mathbf{x}_n\|^2}$$

Substituting this value of λ^* into the expression for θ^* above gives the answer:

$$\boldsymbol{\theta}^* = -\frac{1}{\|\mathbf{x}_n\|}\mathbf{x}_n$$

$$\boldsymbol{\theta}^* = -\frac{1}{a^2 + e^2}\begin{bmatrix} a \\ e \end{bmatrix}$$

2. (b) Now, the optimization problem is as follows:

$$f(\boldsymbol{\theta}) = \frac{1}{2}\|\boldsymbol{\theta}\|^2$$

$$g_1(\boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{x}_1 \geq 1$$

$$g_2(\boldsymbol{\theta}) = \boldsymbol{\theta}^T \mathbf{x}_2 \leq -1$$

Assuming the data has 2 dimensions, the two constraint equations give

$$\theta_1 + \theta_2 \geq 1 \text{ and } \theta_1 \leq -1$$

We wish to minimize the magnitude of $\boldsymbol{\theta}$, so we pick the value $\theta_2 = -1$, which gives $\theta_1 = 2$. The margin in this case is $\frac{1}{\|\boldsymbol{\theta}\|_2} = \frac{\sqrt{5}}{5}$. Thus, the answer is:

$$\theta^* = \begin{bmatrix} -1 \\ 2 \end{bmatrix} \text{ and } \gamma = \frac{\sqrt{5}}{5}$$

2. (c) If b is allowed to be nonzero, then the constraint equations become:

$$\theta_1 + \theta_2 + b \geq 1 \text{ and } \theta_1 + b \leq -1$$

Then, to minimize the magnitude of $\boldsymbol{\theta}$, we can let $b = -1$ so that $\theta_2 = 0$, and then $\theta_1 + 0 - 1 \geq 1$, giving $\theta_1 = 2$. Thus, the new values are

$$(\boldsymbol{\theta}^*, b^*) = (\,[\,1 \;\; 0\,]^T, -1), \text{ and } \gamma = 0.5$$

The magnitude of $\boldsymbol{\theta}$ has decreased, while the margin has increased, which makes sense because we are now allowing for a greater set of hyperplanes to pick from, so the model is able to find a better one with smaller magnitude of $\boldsymbol{\theta}$ and larger margin.

3.1: Feature Extraction
(a) See code.

(b) See code.

(c) See code.

3.2: Hyperparameter Selection for Linear Kernel SVM
(a) See code

(b) It would be beneficial to maintain class proportions across folds because this way, we avoid the possibility of the proportion of a particular label in a particular fold affecting the result. If we did not maintain the same class proportions, we could potentially have a particular fold perform unusually well in testing, giving an error rate that is too low in comparison to the actual error rate of the model. As an extreme example, consider the case where all the labels in the training set are positive and all the labels in the testing set are negative: then the model will learn to always predict positive and have 0% error, but then during testing it will have 100% error due to all the testing labels being negative. This kind of fold division is definitely not representative of the actual data, and thus can be prevented by maintaining the class proportions to be roughly the same across folds.

(c) See code.

(d) The results are shown in the table below.

C	Accuracy	F1-score	AUROC	Precision	Sensitivity	Specificity
10^{-3}	0.7089	0.8297	0.5000	0.7089	1.0000	0.0000
10^{-2}	0.7107	0.8306	0.5031	0.7102	1.0000	0.0063
10^{-1}	0.8060	0.8755	0.7188	0.8357	0.9294	0.5081
10^{0}	0.8146	0.8749	0.7531	0.8562	0.9017	0.6045
10^{1}	0.8182	0.8766	0.7592	0.8595	0.9017	0.6167
10^{2}	0.8182	0.8766	0.7592	0.8595	0.9017	0.6167
Best C	10^{2}	10^{2}	10^{2}	10^{2}	10^{-2}	10^{2}

For all metrics except sensitivity, as C increases, the accuracy according to that metric, so the best value is $C = 100$. With sensitivity, the best C value is 10^{-2} instead, so perhaps the sensitivity metric may not be the best for measuring the accuracy of this data set, as it gives different results compared to all the other metrics. Also, for all values, the performance for $C = 10$ and $C = 100$ is the same (up to 4 decimal places), so $C = 10$ may be good enough for this particular problem.

3.3: Hyperparameter Selection for RBF Kernel SVM

(a) The role of the γ hyperparameter is to act as a kernel coefficient (from sklearn.svm.SVC documentation) that determines far the influence from a single training data point reaches. If γ is low, then the influence extends far, while if γ is high, then the influence does not go very far. Thus, if γ is low, then every point has a "say" in creating the hyperplane, while if γ is high, then each support vector will only contain itself, resulting in a overfitted model that does not generalize well.

If γ is high, it can be thought to be analogous to 1-nearest neighbors, while if γ is low, it can be thought of as *n*-nearest neighbors, where *n* is all the training data points.

(b) I did a grid search that enumerated across all combinations of C and γ, where elements of C were chosen from the set $\{10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$ and elements of γ were chosen from the set $\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^{0}, 10^{1}, 10^{2}\}$. An instructor on Piazza post suggested a range of 10^{-5} to 10^{0} for γ, and I increased the upper value to 10^{2} so that range of values would also match the original values of C that were searched. I thought this would work well because the range represents a broad range of numbers (over 6 orders of magnitude for C and 8 for γ), thus increasing the likelihood of finding the best combination of parameters.

(c) See table below. When different parameters gave the same result, I picked the lowest value (or combination of values) for that parameter.

Metric	Score	C	γ
Accuracy	0.8165	10^{2}	10^{-2}
F1-score	0.8763	10^{2}	10^{-2}
AUROC	0.7545	10^{2}	10^{-2}
Precision	0.8583	10^{2}	10^{-2}
Sensitivity	1.0000	10^{-3}	10^{-5}
Specificity	0.6047	10^{2}	10^{-2}

The optimal values for C and γ are very similar for the RBF kernel when compared to that of the linear kernel: in the vast majority of metrics, a value of $C = 100$ is the best hyperparameter, and it is always accompanied by a value of $\gamma = 0.01$. The high value of C and low value of γ indicates a high regularization parameter and each training data point having a relatively far influence. The only exception is sensitivity, where the optimal value of C is 10^{-3} and the optimal value of γ is 10^{-5}. The same occurred with the linear kernel, which seems to hint at the fact that sensitivity may not be a good metric of performance for this particular problem.

3.4: Test Set Performance

(a) Based on the results from 3.2(d) and 3.3(c), I picked $C = 100$ and $\gamma = 0.01$ for both linear and RBF kernels. This is because those were the hyperparameter values that resulted in the best metric scores for almost all the metrics except for sensitivity, which seems to be the odd one out.

(b) See code.

(c) See table below.

Metric	Linear Kernel SVM Score	RBF Kernel SVM Score
Accuracy	0.7429	0.7571
F1-score	0.4375	0.4516
AUROC	0.6529	0.6361
Precision	0.6364	0.7000
Sensitivity	0.3333	0.3333
Specificity	0.9184	0.9388

From the table, we can see in the table that on all metrics except for AUROC, the RBF kernel performs as good (sensitivity metric) or slightly better than the linear kernel (all other metrics except for AUROC and sensitivity). Thus, for this particular set of data, the RBF kernel seems to be a better choice for classifying the data.