In The Name of God, The Compassionate, The Merciful

Name: ....

Student ID#: ....

Statistical Pattern Recognition (CE-725)
Department of Computer Engineering
Mini Exam #1 (SVM & Kernel) - Spring 2012

1. (15 points) Prove that $K(x,x') = \exp\left(-\frac{\|x-x'\|^2}{c}\right)$ is a valid kernel. You can only use the following given identities:

If $k_1(x,x')$ and $k_2(x,x')$ are valid kernels, then the following $k(x,x')$s are also valid kernels:

1. $k(x,x') = ck_1(x,x')$
2. $k(x,x') = k_1(x,x') + k_2(x,x')$
3. $k(x,x') = k_1(x,x').\ k_2(x,x')$
4. $k(x,x') = exp(\ k_1(x,x')\ )$
5. $k(x,x') = f(x)\ k_1(x,x')\ f(x')$

Sol:

$$k(x,x') = \exp\left(-\frac{\|x-x'\|^2}{c}\right)$$

$$= \exp\left(\frac{-\|x\|^2}{c} + \frac{-\|x'\|^2}{c} + \frac{2x^T x'}{c}\right)$$

$$= \exp\left(\frac{-\|x\|^2}{c}\right)\exp\left(\frac{2x^T x'}{c}\right)\exp\left(\frac{-\|x'\|^2}{c}\right)$$

$$= f(x)\exp\left(\frac{2}{c}x^T x'\right)f(x')$$

$x^T x'$ is a valid kernel. Then, using rules 1, 4 and 5, the given kernel will also be valid.

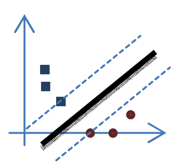
2. (15 points) Consider the training points as below:

$$C_1 = \{ (1,3)^T, (1,4)^T, (2,2)^T \}$$

$$C_2 = \{ (4,0)^T, (5,0)^T, (6,1)^T \}$$

Plot data points and draw the large margin classifier that SVM finds visually. Then without explicitly solving the optimization problem, determine support vectors and λ_is for each training points.

Sol:



SVs: $(4,0)^T$, $(2,2)^T$.

$$\sum \lambda_i y_i = 0 \Rightarrow \lambda_1 = \lambda_2$$

$$w = \sum \lambda_i y_i x_i = \lambda_1(2,2) - \lambda_2(4,0) = (-2\lambda_1, 2\lambda_2)$$

$$SVs: \begin{cases} w(4,0) + w_0 = -1 \\ w(2,2) + w_0 = 1 \end{cases} \Rightarrow \begin{cases} -8\lambda_1 + w_0 = -1 \\ w_0 = 1 \end{cases} \Rightarrow \lambda_1 = \frac{1}{4}, \quad w = (-\frac{1}{2}, \frac{1}{2}), \quad w_0 = 1$$

3. (15 points) Suppose that you are given n-dimensional real-valued feature vectors. You are thinking of using as kernel a function that counts in how many places both x and x' are greater than some constant threshold θ. In other words, let:

$$\delta_\theta(a,b) = \begin{cases} 1 & \text{if } a > \theta \;\&\; b > \theta \\ 0 & \text{otherwise} \end{cases}$$

Then the function defined by $k_\theta(x,x') = \sum_{j=1}^{n} \delta_\theta(x_j, x'_j)$ and x_j denotes j^{th} component of x. Show that this function is a valid kernel.

Sol:

To showvalidity of given kernel, it is enough to find its corresponding *φ(x)*. We have,

$$\varphi(x) = I(x > \theta), or\ \varphi_i(x) = \begin{cases} 1 & x_i > \theta \\ 0 & otherwise \end{cases} \Rightarrow k(x,x') = \varphi^T(x)\varphi(x')$$

4. (20 points) After mapping into higher dimensional feature space, through a radial basis (Gaussian) kernel function, is 1-NN using unweighted Euclidean distance able to achieve better classification performance than in the original space?

Sol:

Suppose that x_i and x_j are two neighbors for the test instance such that $\|x - x_i\| < \|x - x_j\|$. After mapped to feature space, we have,

$$\|\phi(x) - \phi(x_i)\|^2 = 2 - 2\exp\left(-\frac{1}{2}\|x - x_i\|^2\right) < 2 - 2\exp\left(-\frac{1}{2}\|x - x_j\|^2\right) = \|\phi(x) - \phi(x_j)\|^2$$

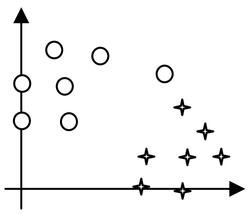
So, if x_i is the nearest neighbor of x in the original space, it will also be the nearest neighbor of x in the original space. Therefore, the result of 1-NN in the new space is the same as the result in the original space.

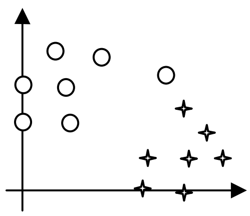
5. (15 points) Suppose we run SVM on sampleswith features $x_1,\ldots,x_n$ i.e. $x = (x_1,\ldots,x_n)^T$, and obtain the boundary. Then, we add a random feature x_{n+1} to feature vectors. How does the boundary changes by adding this feature?

Sol:

SVM will automatically ignore this feature because it cannot possibly increase the margin.

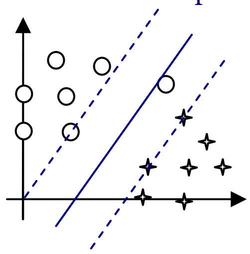
6. (20 points) Consider the following classification problem. We will use soft margin linear SVM for this problem. Use the following figures to show where the decision boundaries are likely to be for c=0.05 (in the left figure) and c=10000(in the right figure) , respectively.Don't solve the problem explicitly; just give a brief justification for each case.

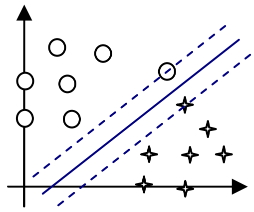




Sol:

Small c results in small penalties. Then, focus will be on maximizing the margin. Large





c leads to large penalties. Then, focus will be on getting all points outside the margins.