Problem Set 3. Zhixuan Yang

1. (a) (Outlook) sunny: 3 (Temperture) hot: 3 (Humidity). high: 2

overcast: 2 mild: 2 normal: 1

rain: 1 cool: 1

(Wind) strong: 2 Target: PlayTennis: No = -1

weak: 1 Yes = 1.

Example	outlook	Tempeture	Humidity	Wind	Play Tennis
1	3	3	2	1	-1
2	3	3	2	2	-1
3	2	3	2	1	1
4	1	2	2	1	1
5	1	1	1	1	1
6	1	1	1	2	-1
7	2	1	1	2	1
8	3	2	2	1	-1
9	3	1	1	1	1
10	1	2	1	1	1
11	3	2	1	2	1
12	2	2	2	2	1
13	2	3	1	1	1
14	1	2	2	2	-1

Since a support vector machine classifier requires the data set's values are numeric, I convert the values into numbers based on the above rules, and this data replesentation that makes it possible to data samples for a SVM classifier. I used number to present the levels of attribute values.

(b) $\lambda = 0.1$, $C = 2$, and weights and bias initalized to 0.
Since weights and b initalized to 0, $y_i(w, x_i)+b < 1$ all the time, and then Indicator function I always equal to one.

$$w_1 = (-0.1)(0) + (2)(0.9)\underbrace{\sum_{i=1}^{n}(y_i)(x_{i1})}_{(6)}(1) = 10.8$$

$$w_2 = (-0.1)(0) + (2)(0.9)\underbrace{\sum_{i=1}^{n}(y_i)(x_{i2})}_{(6)}(1) = 10.8$$

$$w_3 = (-0.1)(0) + (2)(0.9)\underbrace{\sum_{i=1}^{n}(y_i)(x_{i3})}_{(3)}(1) = 5.4$$

$$w_4 = (-0.1)(0) + (2)(0.9)\underbrace{\sum_{i=1}^{n}(y_i)(x_{i4})}_{(4)}(1) = 7.2$$

∴ the final weights after updated through the entire training dataset is (10.8, 10.8, 5.4, 7.2)

$$b = 0 + (2)(0.9)\underbrace{\sum_{i=1}^{n} y_i}_{(4)}(1) = 7.2$$

∴ the final bias is 7.2

$w = (10.8, 10.8, 5.4, 7.2)$

$b = 7.2$

2. (a) DNA strings of arbitrary length (recall that the DNA alphabet has 4 letters - A, C, G, T)

For this situation, string kernel function is the best choice because this kind of kernel function can be intuitively understood as functions measuring the similiarity of pairs of strings., and without having to translate these to fixed-length. Thus, string kernel function is good for this situation.

String kernel

(b) Unlabeled, undirected graphs

For this situation, random walk kernel is the best choice. Random walk kernels quantify the similarity between a pair of graphs based on the number of common walks in the two graphs. Since our graphs are unlabeled and unlabled, we kan detect the similiarities between graphs. And the formula of random walk kernel is.

$$V_x = \{(v_i, v_j) : v_i \in V_i \wedge u_j \in V_j \wedge \ell(v_i) = \ell(v_j)\} \quad \text{Vertex.}$$

$$E_x = \{\{(v_i, v_j), (u_i, v_j)\} : \{v_i, u_i\} \in E_i \wedge \{v_j, u_j \in E_j\}$$

Thus, random walk is good for this situation.

Random walk kernel

(c) Node labeled undirected graphs where the nodes take labels from the set $\{A, B\}$.

Fisher kernel is the best for this problem. This kind of function measures the similarity of two objects on the basis of sets of measurements for each object and a statistical model. Since our graphs have Node labeled, we have the basis of sets of measurments, so we can use these nodes to classify different graphs. The function of fisher kernel is

$$K(x_i, x_j) = U^T_{x_i} I^{-1} U_{x_j}$$

Therefore, Fisher Kernel is a very good design for this situation because labeled nodes.

Fisher Kernel

3. The original logistic Regression function is

$$P(Y=1|X) = \frac{1}{1+e^{-\langle W, G(X)\rangle}} = \frac{1}{1+e^{-h(X,W)}} = \mu(X,W)$$

Where

$$h(X,W) = W^T G(X) = \langle W, G(X)\rangle$$

And the posterior probability of $Y=1$ is same as the conditional expectation of y given X:

$$E(Y|X) = 1\cdot P(Y=1|X) + 0\,P(Y=0|X) = P(Y=1|X) = \mu(X,W)$$

$$= (\mu(X,W))^{Y}(1-\mu(X,W))^{1-Y}$$

where $\mu(X,W) = \frac{1}{1+e^{-h(X,W)}} = \frac{1}{1+e^{-w^T x}}$

Therefore, estimating $P(Y=1|X)$ is equivalent to performing logistic regression.

We can write $h(X,W)$ as $h(X,W) = \sum_{i=0}^{K} w_i \varphi_i(X)$,

For Kernel regression, $\varphi_i(X) = \phi(X_i)\cdot\phi(X) = K(X, X_i)$

So, we have to replace Kernel regression into $h(X,W)$, and that would be

$$h(X,W) = W_0 + \sum_{i=1}^{N} W_i K(X, X_i)$$

$$\therefore P(Y=1|X) = \frac{1}{1+e^{-(W_0 + \sum_{i=1}^{N} W_i K(X,X_i))}} = \mu(X,W)$$

That is how logistic regression incorporate a Kernel function.

$$h(X,W) = W_0 + \sum_{i=1}^{N} W_i K(X, X_i)$$