CSCI E-181 Midterm 2
Spring 2014

Name: DAVID WIHL

Start Time: 2:15

End Time: 5:15

1	/ 15
2	/ 10
3	/ 10
4	/ 15
5	/ 15
6	/ 15
7	/ 20
Total	/ 100

1. **Combining Kernel Functions** {15pts}

Suppose that $K_1(x, x')$ and $K_2(x, x')$ are both valid kernel functions. Recall that a valid kernel is one that corresponds to an inner product in some (possibly infinite-dimensional) feature space and produces a matrix $K_{ij} = K(x_i, x_j)$ that is a positive semi-definite for any finite set of examples $x_1, x_2, \ldots, x_N$. Show that

$$K(x, x') = K_1(x, x') + K_2(x, x')$$

is a valid kernel if $K_1(x, x')$ and $K_2(x, x')$ are both valid kernels. [Hint: It may be useful to recall that a matrix $\boldsymbol{K}$ is positive semi-definite if $\boldsymbol{y}^{\mathsf{T}}\boldsymbol{K}\boldsymbol{y} \geq 0, \forall \boldsymbol{y}$.]

$$K = K_1 + K_2$$

adding matrices by component

$$\forall \underline{y},\ \underline{y}^T K \underline{y} = \underline{y}^T K_1 \underline{y} + \underline{y}^T K_2 \underline{y} \geq 0$$

Alternative:

$$\phi^1(x) = (\phi^1_1(x), \ldots, \phi^1_{N_1}(x))$$

$$\phi^2(x) = (\phi^2_1(x), \ldots \phi^2_{N_2}(x))$$

as feature maps of K_1 and K_2, combining

$$\phi(x) = (\phi^1_1(x), \ldots \phi^1_{N_1}(x), \phi^2_1(x), \ldots \phi^2_{N_2}(x))$$

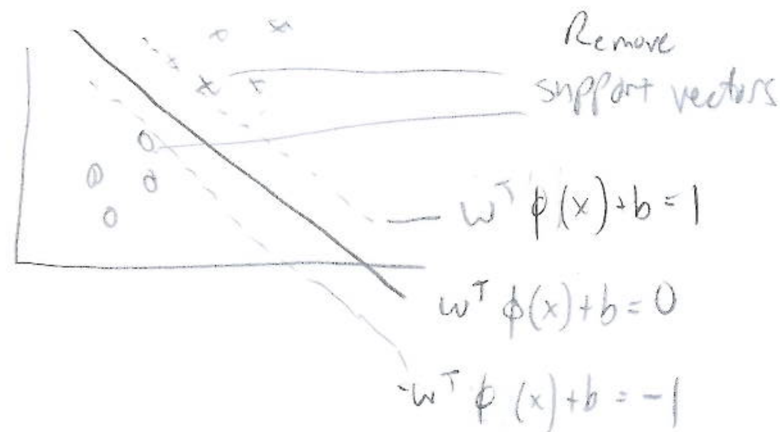
which satisifyes

$$\phi(x) \cdot \phi(y) = \phi^1(x) \cdot \phi^1(x) + \phi^2(x)\phi^2(x)$$

2. **Support Vector Machines** {10pts}

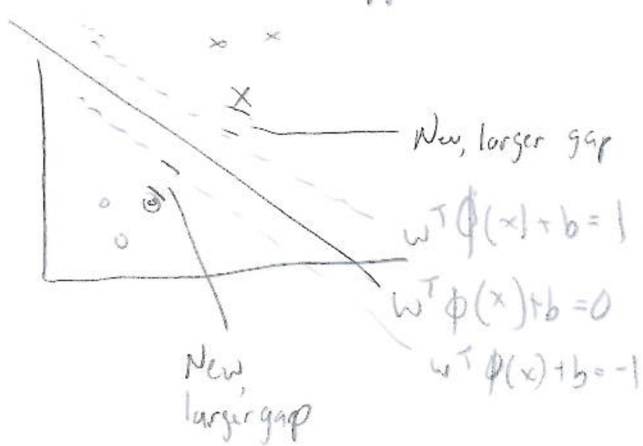
Suppose that we have a data set and we train two support vector machines as follows. We train the first SVM on all of the data. We then remove the SVM's support vectors from the data set and train a second support vector machine on the remaining data. How does the size of the optimal margin change from the first to the second SVM? Provide an explanation and/or diagrams to make your case.

The optimal margin of the second svm will be larger in dwidth than first.



$\min_n \left(t_n \frac{1}{\|w\|}(w^T\phi(x)+b)\right)$

We then remove the support vectors



New margin $= \frac{t_1 w^T \phi(x_1)+b}{\|w\|} - \frac{t_0 w^T \phi(x_0)+b}{\|w\|}$

3. **Utility Functions** {10pts}

Assume you have a cookie-eating utility function $U_1(c)$ and your friend has a utility function $U_2(c)$. If $U_1(c) = 2 \times U_2(c) - 100$. Explain whether or not you and your friend have the same cookie-eating preferences.

My friend and I will have same preferences

$$u_1(c) = a \cdot U_2(c) + b$$

since $a > 0$

if $u_2(c') > u_2(c)$,

then $u_1(c') > u_1(c)$

Utilities determine relative preferences. The actual utility value is not meaningful.

4. **Multi-State Rewards for MDPs** {15pts}

In CS181, we've been assuming that the reward function has the form $R(s,a)$, i.e., it only depends on the current state and action. In the discounted infinite-horizon case, we use this when computing the Q-function via:

$$Q(s,a) = R(s,a) + \gamma \sum_{s'} P(s' \mid s,a)\, V(s')\,.$$

Now, imagine that we instead have a more complicated reward function that depends on both the current state **and** the next state, i.e., $R(s,a,s')$. Write an expression for $Q(s,a)$ that incorporates this alternative type of reward.

Applying the chain rule of probability

$$Q(s,a) = R(s,a,s') + \gamma \sum_{s'} \left[P(s' \mid s,a) V(s') \cdot P(s'' \mid s', s, a) \right]$$

$$+ \gamma^2 \sum_{s''} P(s'' \mid s', s, a) V(s'')$$

5. **Reinforcement Learning and Planning** {15pts}

The update rule for Q-learning is:

$$Q(s,a) \leftarrow Q(s,a) + \alpha \left[R(s,a) + \gamma \max_{a'} Q(s',a') - Q(s,a) \right]$$

Consider two states $S = \{s_1, s_2\}$ and actions $A = \{a_1, a_2\}$, and current Q-values

	a_1	a_2
s_1	6	4
s_2	2	3

(a) {3pts} Suppose the agent is in state s_1. Using ϵ-greedy, how would it decide to act?

$$(1-\varepsilon) \max_a Q(s,a) + \frac{\varepsilon}{A} \sum_a Q(s,a)$$

(b) {4pts} Suppose the agent exploits in s_1 and lands in s_2. Which Q-value would be updated, and what is the value for $\max_{a'} Q(s', a')$ used in the update?

$$Q'(s,a) = 2 + \alpha[2 + \gamma \cdot 6 - 2]$$
$$= 2 + 2\alpha + 6\alpha\gamma - 2\alpha$$
$$= 6\alpha\gamma + 2$$

$$\max_{a'} Q(s',a') = 6$$

(c) {4pts} What is an advantage of model-based learning approaches over Q-learning?

EVENTUALLY KNOW ENTIRE STATE OF THE WORLD
SO EXPLOITATION IS MAXIMIZED OVER TIME.

(d) {4pts} State one advantage of policy iteration over value iteration for planning.

- MAY TAKE LESS ITERATIONS THAN VALUE ITERATION
- DOES NOT REQUIRE PARAMETER ε
- DOES NOT NEED TO WAIT FOR CONVERGENCE

6. **Mean of a Mixture Model** {15pts}

Show that if we are given a mixture model of the form

$$p(x \mid \{\theta_k\}_{k=1}^K) = \sum_{k=1}^{K} \pi_k \, p(x \mid \theta_k)$$

where the elements of x could be discrete or continuous or a combination of these, the mean of $p(x \mid \{\theta_k\}_{k=1}^K)$ of the mixture distribution is given by

$$\mathbb{E}[x] = \sum_{k=1}^{K} \pi_k \boldsymbol{\mu}_k$$

where the $\boldsymbol{\mu}_k$ are the means of the component distributions $p(x \mid \theta_k)$.

$$p(x \mid \underline{\mu}, \underline{\Pi}) = \sum_{k=1}^{K} \Pi_k \, p(\underline{x} \mid \mu_k)$$

$$p(\underline{x} \mid \mu_k) = \prod_{i=1}^{D} \mu_{ki}^{x_i} (1-\mu_{ki})^{(1-x_i)}$$

$$\mathbb{E}[\underline{x}] = \sum_{k=1}^{K} \Pi_k \underline{\mu}_k$$

$$= \sum_{k=1}^{K} \Pi_k \mathbb{E}_k[\underline{x}]$$

Since expectation of a sum is the sum of expectations

Assumes distribution is normalized.

$\mathbb{E}_k[x] = \mu_k$ because $\sum_{x \in \{0,1\}} x \, p(x \mid \mu) = 0 \, p(x=0 \mid \mu) + 1 \, p(x=1 \mid \mu)$

$= \mu$

7. **Expectation Maximization** {20pts}

Imagine that we have a collection of N binary images, each of which is 25×25, and we wish to perform model-based clustering using a mixture. We treat each image as a 625-dimensional binary vector and use a product of Bernoulli distributions for each component in the mixture:

$$p(x \mid \mu_k) = \prod_{d=1}^{625} \mu_{k,d}^{x_{n,d}} (1 - \mu_{k,d})^{1 - x_{n,d}} \qquad x_n \in \{0,1\}^{625} \qquad \mu_k \in (0,1)^{625}.$$

Here, image n is denoted as a vector x_n and the dth pixel is $x_{n,d}$. Each of the K mixture components has a parameter μ_k and each dimension $\mu_{k,d}$ is a Bernoulli distribution parameter that specifies the probability of pixel d being black. The (known) mixture weights are $\{\pi_k\}_{k=1}^K$. You want to learn the parameters $\{\mu_k\}_{k=1}^K$ and find a clustering for the data, so you decide to use the expectation maximization algorithm. (There are four parts to this, make sure you do parts (c) and (d) on the next page.)

(a) {2pts} Write the probability mass function for the mixture, i.e., $p(x \mid \{\mu_k\}_{k=1}^K)$.

$$p(x \mid \underline{\pi}, \{\mu_k\}_{k=1}^K) = \sum_{k=1}^K \pi_k \, p(x \mid \mu_k)$$

(b) {2pts} This is an example of a latent variable model. What kind of latent variables are they? What distribution would you use to estimate them within EM?

Since the underlying pixel is either on or off, the latent

$X_{n,d}$ variable would be a single bit. A Bernoulli distribution

would be appropriate

(c) {8pts} In the E-step, you improve the estimate of the latent variables, fixing the parameters $\{\mu_k\}_{k=1}^K$. Derive the update for this approximation.

$$\gamma_{kd} = p\left(z_{kd} = 1 \mid \underline{x}_n, \{\pi_k, \mu_k, \Sigma_k\}\right)$$

$$= \frac{p(z_{kd}=1)\, p(\underline{x}_n \mid z_{kd}=1, \{\mu_k, \Sigma_k\})}{p(x_n \mid \{\pi_k, \mu_k, \Sigma_k)}$$

$$= \frac{\pi_k N(x_n \mid \mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k N(\underline{x}_n \mid \mu_{k'} \Sigma_{k'})}$$

(d) {8pts} In the M-step, you use the approximation to update the parameters $\{\mu_k\}_{k=1}^K$. Derive the update for the parameters.

$$\mu_k = \frac{1}{\hat{N}_k} \sum_{n=1}^{N} \gamma_{kd} x_n$$

$$\frac{\partial}{\partial \mu_{ki}} \mathbb{E}_z\left[\ln p(\underline{x}, \underline{z} \mid \underline{\mu}, \underline{\pi})\right] = 0$$

$$= \sum_{n=1}^{N} \gamma(z_{nk}) \left(\frac{x_{ni}}{\mu_{ki}} - \frac{1-x_{ni}}{1-\mu_{ki}}\right)$$

$$= \frac{\sum_n \gamma(z_{nk}) x_{ni} - \sum_n \gamma(z_{nk}) \mu_{ki}}{\mu_{ki}(1-\mu_{ki})}$$

$$\mu_k = \frac{\sum r_k x_n}{\sum r_{kd}}$$