15-781 Final Exam, Fall 2002

1. Write your name and your andrew email address below.

   Name:

   Andrew ID:

2. There should be 17 pages in this exam (excluding this cover sheet).

3. If you need more room to work out your answer to a question, use the back of the page and clearly mark on the front of the page if we are to look at what's on the back.

4. You should attempt to answer all of the questions.

5. You may use any and all notes, as well as the class textbook.

6. All questions are worth an equal amount. They are not all equally difficult.

7. You have 3 hours.

8. Good luck!

1 Computational Learning Theory

1.1 PAC learning for Decison Lists

A decision list is a list of if-then rules where each condition is a literal (a variable or its negation). It can be thought of as a decision tree with just one path. For example, say that I like to go for a walk if it's warm or if it's snowing and I have a jacket, as long as it's not raining. We could describe this as the following decision list:

```
if rainy then no
else if warm then yes
else if not(have-jacket) then no
else if snowy then yes
else no.
```

(a) Describe an algorithm to learn DLs given a data set, for example

a	b	c	class
1	0	0	+
0	1	1	-
1	1	1	+
0	0	0	-
1	1	0	+

Your algorithm should have the characteristic that it should always classify examples that it has already seen correctly (ie, it should be consistent with the data).If it's not possible to continue to produce a decision list that's consistent with the data, your algorithm should terminate and announce that it has failed.

(b) Find the size of the hypothesis space, $|H|$, for decisions lists of k attributes.

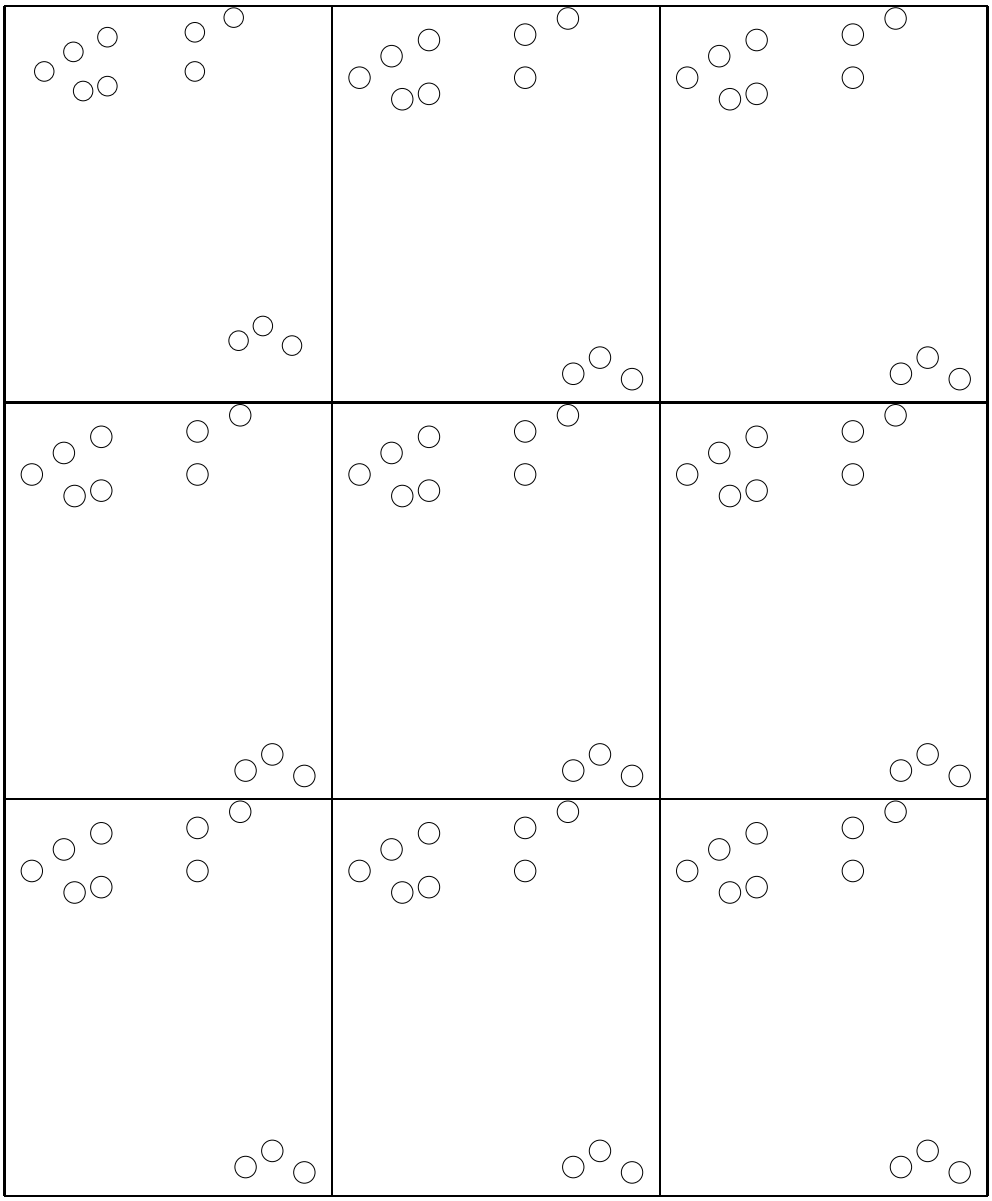
(c) Find an expression for the number of examples needed to learn a decision list of k attributes with error at most .10 with probability 90%.

(d) What if the learner is trying to learn a decision list, but the representation that it is using is a conjunction of k literals? Find the expression for the number of examples needed to learn the decision list with error at most .10 with 90% probability.

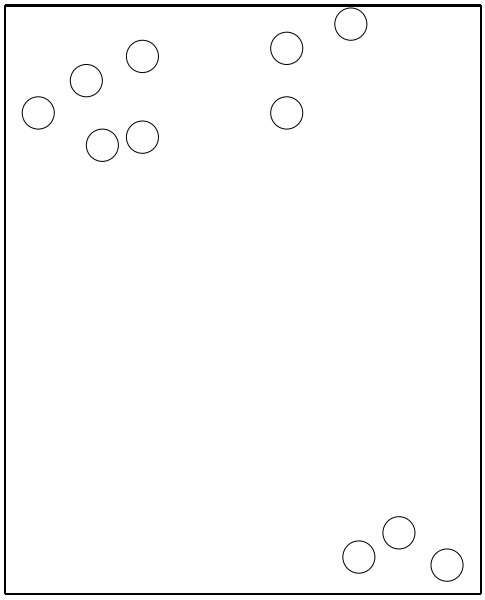
2 K-means and Gaussian Mixture Models

(a) What is the effect on the means found by k-means (as opposed to the true means) of overlapping clusters?

(b) Run k-means manually for the following dataset. Circles are data points and squares are the initial cluster centers. Draw the cluster centers and the decision boundaries that define each cluster. Use as many pictures as you need until convergence.
Note: Execute the algorithm such that if a mean has no points assigned to it, it stays where it is for that iteration.



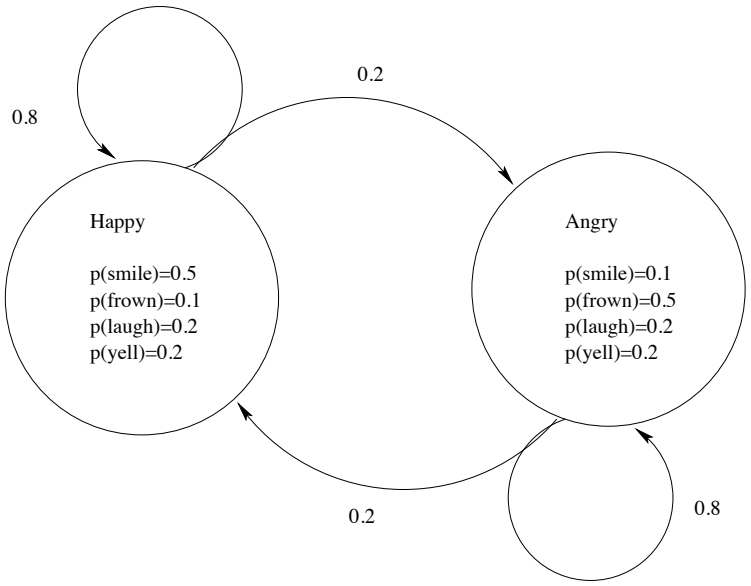
(c) Now draw (approximately) what a Gaussian mixture model of three gaussians with the same initial centers as for the k-means problem would converge to. Assume that the model puts no restrictions on the form of the covariance matrices and that EM updates both the means and covariance matrices.



(d) Is the classification given by the mixture model the same as the classification given by k-means? Why or why not?

3 HMMs

Andrew lives a simple life. Some days he's Angry and some days he's Happy. But he hides his emotional state, and so all you can observe is whether he smiles, frowns, laughs, or yells. We start on day 1 in the Happy state, and there's one transition per day.



Definitions:
q_t = state on day t.
O_t = observation on day t.

(a) What is $P(q_2 = Happy)$?

(b) What is $P(O_2 = frown)$?

(c) What is $P(q_2 = Happy | O_2 = frown)$?

(d) What is $P(O_{100} = yell)$?

(e) Assume that $O_1 = frown$, $O_2 = frown$, $O_3 = frown$, $O_4 = frown$, and $O_5 = frown$. What is the most likely sequence of states?

4 Bayesian Inference

(a) Consider a dataset over 3 boolean attributes, X, Y, and Z.

Of these sets of information, which are sufficient to specify the joint distribution? Circle all that apply.

A. $P(\sim X|Z)\ P(\sim X|\sim Z)\ P(\sim Y|X\wedge Z)\ P(\sim Y|X\wedge\sim Z)$

$P(\sim Y|\sim X\wedge Z)\ P(\sim Y|\sim X\wedge\sim Z)\ P(Z)$

B. $P(\sim X|\sim Z)\ P(X|\sim Z)\ P(Y|X\wedge Z)\ P(Y|X\wedge\sim Z)$

$P(Y|\sim X\wedge Z)\ P(Y|\sim X\wedge\sim Z)\ P(Z)$

C. $P(X|Z)\ P(X|\sim Z)\ P(Y|X\wedge Z)\ P(Y|X\wedge\sim Z)$

$P(Y|\sim X\wedge Z)\ P(\sim Y|\sim X\wedge\sim Z)\ P(\sim Z)$

D. $P(X|Z)\ P(X|\sim Z)\ P(Y|X\wedge Z)\ P(Y|X\wedge\sim Z)$

$P(\sim Y|\sim X\wedge\sim Z)\ P(Y|\sim X\wedge\sim Z)\ P(Z)$

Given this dataset of 16 records:

A	B	C
0	0	1
0	0	1
0	0	1
0	1	0
0	1	1
0	1	1
0	1	1
0	1	1
1	0	0
1	0	0
1	0	0
1	0	0
1	0	0
1	1	0
1	1	0
1	1	1

(b) Write down the probabilities needed to make a joint density bayes classifier

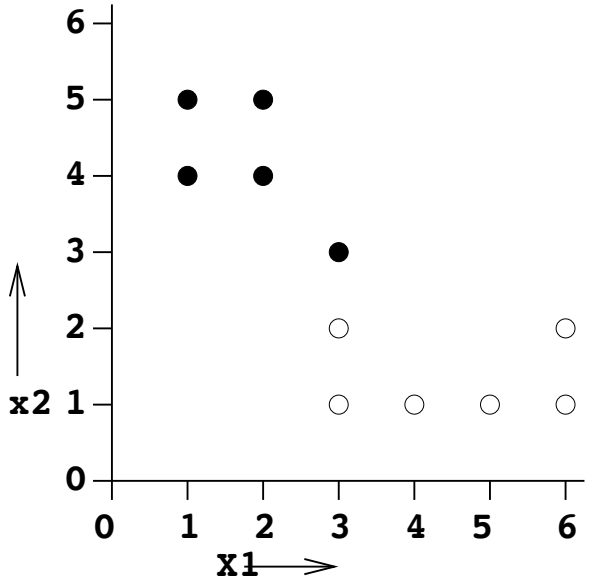
(c) Write down the probabilities needed to make a naive bayes classifier.

(d) Write the classification that the joint density bayes classifier would make for C given A=0,B=1.

(e) Write the classification that the naive bayes classifier would make for C given A=0,B=1.

5 Support Vector Machines

This picture shows a dataset with two real-valued inputs (x_1 and x_2) and one categorical output class. The positive points are shown as solid dots and the negative points are small circles.



(a) Suppose you are using a linear SVM with no provision for noise (i.e. a Linear SVM that is trying to maximize its margin while ensuring all datapoints are on their correct sides of the margin). Draw three lines on the above diagram, showing the classification boundary and the two sides of the margin. Circle the support vector(s).

(b) Using the familiar LSVM classifier notation of class = sign($\mathbf{w}.\mathbf{x} + b$), calculate the values of $\mathbf{w}$ and b learned for part (a)

(c) Assume you are using a noise-tolerant LSVM which tries to minimize

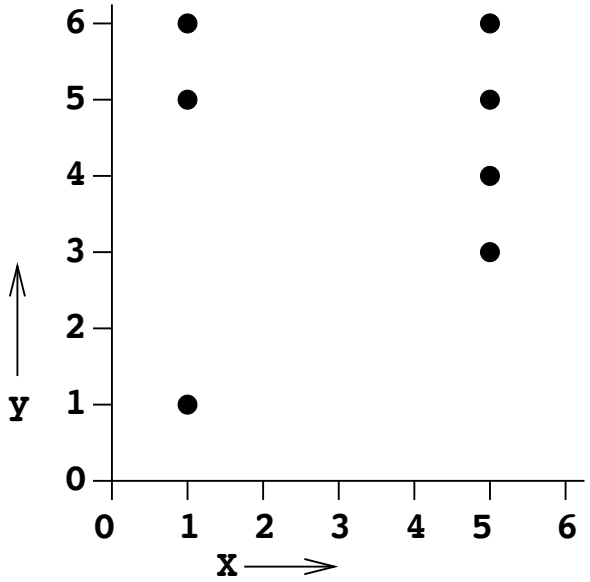
$$\frac{1}{2}\mathbf{w}.\mathbf{w} + C\sum_{k=1}^{R}\epsilon_k \tag{1}$$

using the notation of your notes and the Burges paper.

Question: is it possible to invent a dataset and a positive value of C in which (a) the dataset is linearly separable but (b) the LSVM would nevertheless misclassify at least one training point? If it is possible to invent such an example, please sketch the example and suggest a value for C. If it is not possible, explain why not.

6 Instance-based learning

This picture shows a dataset with one real-valued input x and one real-valued output y. There are seven training points.



Suppose you are training using kernel regression using some unspecified kernel function. The only thing you know about the kernel function is that it is a monotonically decreasing function of distance that decays to zero at a distance of 3 units (and is strictly greater than zero at a distance of less than 3 units).

(a) What is the predicted value of y when $x = 1$?

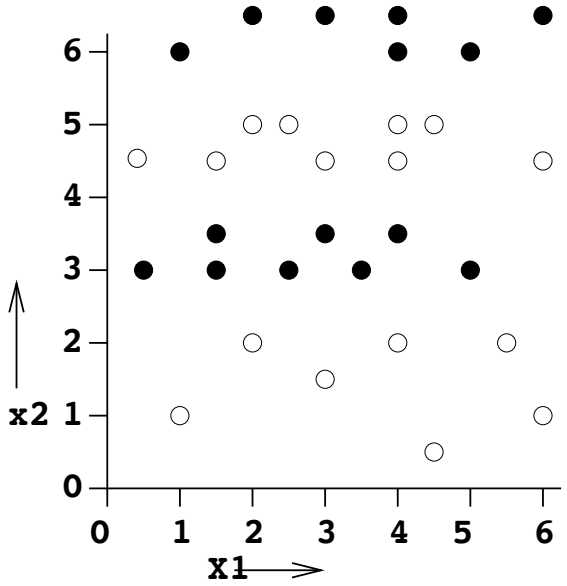
(b) What is the predicted value of y when $x = 3$?

(c) What is the predicted value of y when $x = 5$?

(d) What is the predicted value of y when $x = 6$?

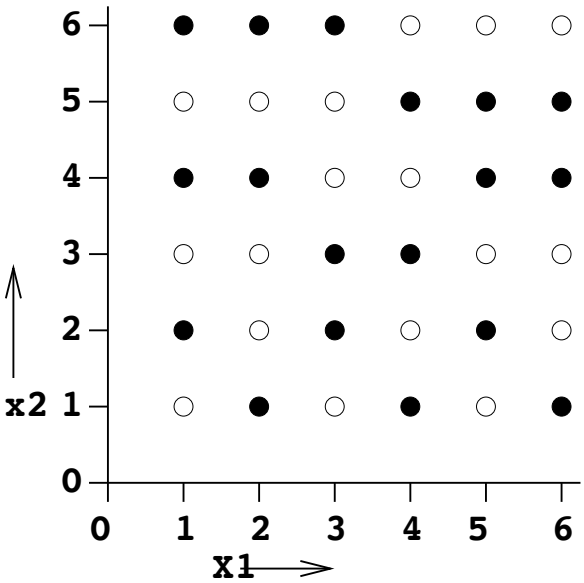
The final two parts of this question concern 1-nearest neighbor used as a classifier.

The following dataset has two real valued inputs and one binary categorical output. The class is denoted by the color of the datapoint.



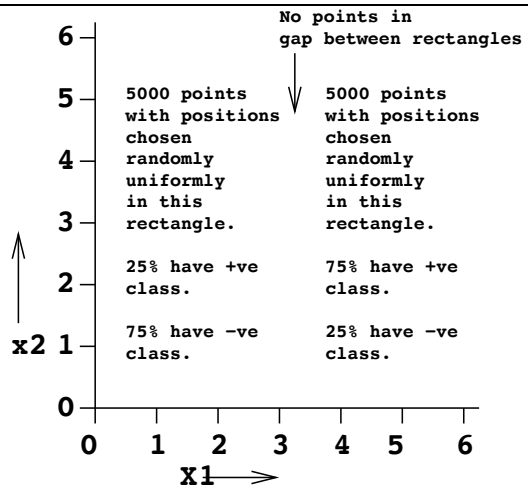
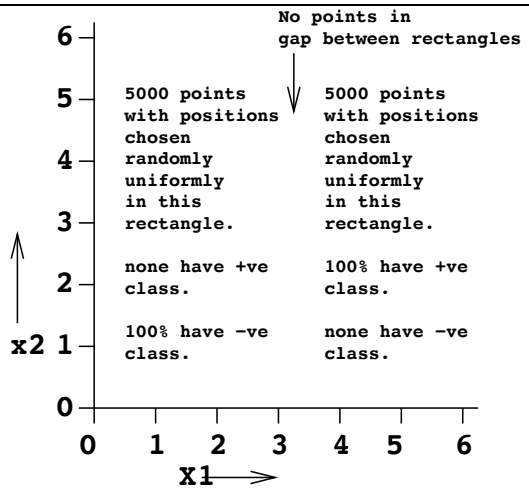
(e) Does there exist a choice of Euclidian distance metric for which 1-nearest-neighbor would achieve zero training set error on the above dataset?

Now let's consider a different dataset:



(f) Does there exist a choice of Euclidian distance metric for which 1-nearest-neighbor would achieve zero training set error on the above dataset?

7 Nearest Neighbor and Cross-Validation

Recipe for making training set of 10,000 datapoints with two real-valued inputs and one binary output class:	Recipe for making test set of 10,000 datapoints with two real-valued inputs and one binary output class:
	

Using the above recipes for making training and test sets you will see that the training set is noisy: in either region, 25% of the data comes from the minority class. The test set is noise-free.

In each of the following questions, circle the answer that most closely defines the expected error rate, expressed as a fraction.

(a) What is the expected training set error using one-nearest-neighbor?

0 1/8 1/4 3/8 1/3 1/2 5/8 2/3 3/4 7/8 1

(b) What is the expected leave-one-out cross-validation error on the training set using one-nearest-neighbor?

0 1/8 1/4 3/8 1/3 1/2 5/8 2/3 3/4 7/8 1

(c) What is the expected test set error if we train on the training set, test on the test set, and use one-nearest-neighbor?

0 1/8 1/4 3/8 1/3 1/2 5/8 2/3 3/4 7/8 1

(d) What is the expected training set error using 21-nearest-neighbor?

0 1/8 1/4 3/8 1/3 1/2 5/8 2/3 3/4 7/8 1

(e) What is the expected leave-one-out cross-validation error on the training set using 21-nearest-neighbor?

0 1/8 1/4 3/8 1/3 1/2 5/8 2/3 3/4 7/8 1

(f) What is the expected test set error if we train on the training set, test on the test set, and use 21-nearest-neighbor?

0 1/8 1/4 3/8 1/3 1/2 5/8 2/3 3/4 7/8 1

8 Learning Bayes Net Structure

For each of the following training sets, draw the structure and CPTs that a Bayes Net Structure learner should learn, assuming that it tries to account for all the dependencies in the data as well as possible while minimizing the number of unnecessary links. In each case, your Bayes Net will have three nodes, called A B and C. Some or all of these questions have multiple correct answers...you need only supply one answer to each question.

(a)

A	B	C
0	0	0
0	0	0
0	0	1
0	0	1
0	1	0
0	1	0
0	1	1
0	1	1
1	0	0
1	0	0
1	0	1
1	0	1
1	1	0
1	1	0
1	1	1
1	1	1

(b)

A	B	C
0	0	0
0	0	0
0	0	0
0	1	1
0	1	1
0	1	1
1	1	0
1	1	0
1	1	0
1	0	1
1	0	1
1	0	1

(c)

A	B	C
0	0	0
0	0	0
0	0	0
1	0	1
1	0	1
1	0	1
1	1	0
1	1	0
1	1	0

9 Markov Decision Processes

Consider the following MDP, assuming a discount factor of $\gamma = 0.5$. Note that the action "Party" carries an immediate reward of +10. The action "Study" unfortunately carries no immediate reward, except during the senior year, when a reward of +100 is provided upon transition to the terminal state "Employed".

(a) What is the probability that a freshman will fail to graduate to the "Employed" state within four years, even if they study at every opportunity?

(b) Draw the diagram for the Markov Process (not the MDP, the MP) that corresponds to the policy "study whenever possible."

(c) What is the value associated with the state "Junior" under the "study whenever possible" policy?

(d) Exactly how rewarding would parties have to be during junior year in order to make it advisable for a junior to party rather than study (assuming, of course, that they wish to optimize their cumulative discounted reward)?

(e) Answer the following true or false. If true, give a one-sentence argument. If false, give a counterexample.

- **(True or False?)** If partying during junior year an optimal action when it is assigned reward r, then it will also be an optimal action for a freshman when assigned reward r.

- **(True or False?)** If partying during junior year is an optimal action when it is assigned reward r, then it will also be an optimal action for a freshman when assigned reward r.

10 Q Learning

Consider the robot grid world shown below, in which actions have deterministic outcomes, and for which the discount factor $\gamma = 0.5$. The robot receives zero immediate reward upon executing its actions, except for the few actions where an immediate reward has been written in on the diagram. Note the state in the upper corner allows an action in which the robot remains in that same state for one time tick.

IMPORTANT: Notice the immediate reward for the state-action pair $< C, South >$ is -100, not +100.

(a) Write in the Q value for each state-action pair, by writing it next to the corresponding arrow.

(b) Write in the $V^*(s)$ value for each state, by writing its value inside the grid cell representing that state.

(c) Write down an equation that relates the $V^*(s)$ for an arbitrary state s to the $Q(s, a)$ values associated with the same state.

(d) Describe one optimal policy, by circling only the actions recommended by this policy

(e) Hand execute the deterministic Q learning algorithm, assuming the robot follows the trajectory shown below. Show the sequence of Q estimates (describe which entry in the Q table is being updated at each step):

state	action	next-state	immediate-reward	updated-Q-estimates
A	East	B	0	
B	East	C	10	
C	Loop	C	0	
C	South	F	-100	
F	West	E	0	
E	North	B	0	
B	East	C	10	

(f) Propose a change to the immediate reward function that results in a change to the Q function, but not to the V function.

11 Short Questions

(a) Describe the difference between a *maximum likelihood* hypothesis and a *maximum a posteriori* hypothesis.

(b) Consider a learning problem defined over a set of instances X. Assume the space of possible hypotheses, H, consists of all possible disjunctions over instances in X. I.e., the hypothesis $x_1 \vee x_6$ labels these two instances positive, and no others. What is the VC dimension of H?

(c) Consider a naive Bayes classifier with 2 boolean input variables, X and Y, and one boolean output, Z.

- Draw the equivalent Bayesian network.

- How many parameters must be estimated to train such a naive Bayes classifier?

- How many parameters would have to be estimated if the naive Bayes assumption is not made, and we wish to learn the Bayes net for the joint distribution over X, Y, and Z?

True or False? If true, explain why in at most two sentences. If false, explain why or give a brief counterexample.

- **(True or False?)** The error of a hypothesis measured over the training set provides a pessimistically biased estimate of the true error of the hypothesis.

- **(True or False?)** Boosting and the Weighted Majority algorithm are both methods for combining the votes of multiple classifiers.

- **(True or False?)** Unlabeled data can be used to detect overfitting.

- **(True or False?)** Gradient descent has the problem of sometimes falling into local minima, whereas EM does not.

- **(True or False?)** HMM's are a special case of MDP's.

ANDREW ID (CAPITALS): _____

NAME (CAPITALS): _____

10-701/15-781 Final, Fall 2003

- You have 3 hours.
- There are 10 questions. If you get stuck on one question, move on to others and come back to the difficult question later.
- The maximum possible total score is 100.
- Unless otherwise stated there is no need to show your working.
- Good luck!

1 Short Questions (16 points)

(a) Traditionally, when we have a real-valued input attribute during decision-tree learning we consider a binary split according to whether the attribute is above or below some threshold. Pat suggests that instead we should just have a multiway split with one branch for each of the distinct values of the attribute. From the list below choose the single biggest problem with Pat's suggestion:

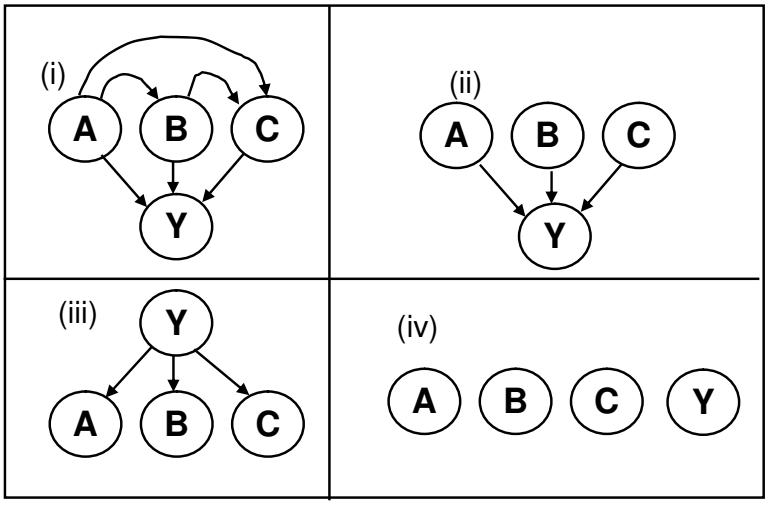
(i) It is too computationally expensive.

(ii) It would probably result in a decision tree that scores badly on the training set and a testset.

(iii) It would probably result in a decision tree that scores well on the training set but badly on a testset.

(iv) It would probably result in a decision tree that scores well on a testset but badly on a training set.

(b) You have a dataset with three categorical input attributes A, B and C. There is one categorical output attribute Y. You are trying to learn a Naive Bayes Classifier for predicting Y. Which of these Bayes Net diagrams represents the naive bayes classifier assumption?



(c) For a neural network, which one of these structural assumptions is the one that most affects the trade-off between underfitting (i.e. a high bias model) and overfitting (i.e. a high variance model):

(i) The number of hidden nodes

(ii) The learning rate

(iii) The initial choice of weights

(iv) The use of a constant-term unit input

(d) For polynomial regression, which one of these structural assumptions is the one that most affects the trade-off between underfitting and overfitting:

(i) The polynomial degree

(ii) Whether we learn the weights by matrix inversion or gradient descent

(iii) The assumed variance of the Gaussian noise

(iv) The use of a constant-term unit input

(e) For a Gaussian Bayes classifier, which one of these structural assumptions is the one that most affects the trade-off between underfitting and overfitting:

(i) Whether we learn the class centers by Maximum Likelihood or Gradient Descent

(ii) Whether we assume full class covariance matrices or diagonal class covariance matrices

(iii) Whether we have equal class priors or priors estimated from the data.

(iv) Whether we allow classes to have different mean vectors or we force them to share the same mean vector

(f) For Kernel Regression, which one of these structural assumptions is the one that most affects the trade-off between underfitting and overfitting:

(i) Whether kernel function is Gaussian versus triangular versus box-shaped

(ii) Whether we use Euclidian versus L_1 versus L_∞ metrics

(iii) The kernel width

(iv) The maximum height of the kernel function

(g) **(True or False)** Given two classifiers A and B, if A has a lower VC-dimension than B then A almost certainly will perform better on a testset.

(h) $P(\textit{Good Movie} \mid \textit{Includes Tom Cruise}) = 0.01$
$P(\textit{Good Movie} \mid \textit{Tom Cruise absent}) = 0.1$
$P(\textit{Tom Cruise in a randomly chosen movie}) = 0.01$

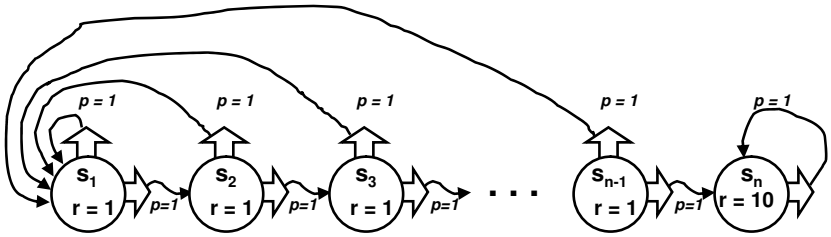
What is $P(\textit{Tom Cruise is in the movie} \mid \textit{Not a Good Movie})$?

2 Markov Decision Processes (13 points)

For this question it might be helpful to recall the following geometric identities, which assume $0 \leq \alpha < 1$.

$$\sum_{i=0}^{k} \alpha^i = \frac{1-\alpha^{k+1}}{1-\alpha} \qquad \sum_{i=0}^{\infty} \alpha^i = \frac{1}{1-\alpha}$$

The following figure shows an MDP with N states. All states have two actions (North and Right) except S_n, which can only self-loop. Unlike most MDPs, all state transitions are deterministic. Assume discount factor γ.



For questions (a)~~(e)~~, express your answer as a finite expression (no summation signs or ... 's) in terms of n and/or γ.

(a) What is $J^*(S_n)$?

(b) There is a unique optimal policy. What is it?

(c) What is $J^*(S_1)$?

(d) Suppose you try to solve this MDP using value iteration. What is $J^1(S_1)$?

(e) Suppose you try to solve this MDP using value iteration. What is $J^2(S_1)$?

(f) Suppose your computer has exact arithmetic (no rounding errors). How many iterations of value iteration will be needed before all states record their exact (correct to infinite decimal places) J^* value? Pick one:

(i) Less than $2n$

(ii) Between $2n$ and n^2

(iii) Between $n^2 + 1$ and 2^n

(iv) It will never happen

(g) Suppose you run policy iteration. During one step of policy iteration you compute the value of the current policy by computing the exact solution to the appropriate system of n equations in n unknowns. Suppose too that when choosing the action during the policy improvement step, ties are broken by choosing North.

Suppose policy iteration begins with all states choosing North.

How many steps of policy iteration will be needed before all states record their exact (correct to infinite decimal places) J^* value? Pick one:

(i) Less than $2n$

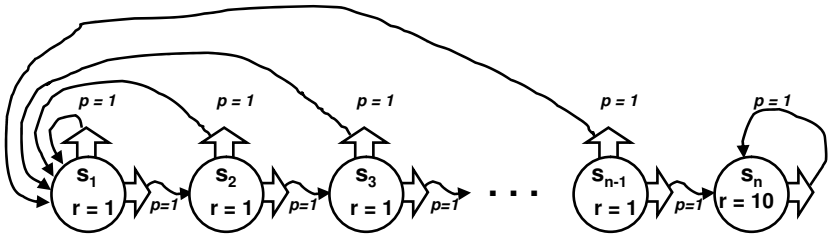
(ii) Between $2n$ and n^2

(iii) Between $n^2 + 1$ and 2^n

(iv) It will never happen

3 Reinforcement Learning (10 points)

This question uses the same MDP as the previous question, repeated here for your convenience. Again, assume $\gamma = \frac{1}{2}$.



Suppose we are discovering the optimal policy via Q-learning. We begin with a Q-table initialized with 0's everywhere:

$Q(S_i, North) = 0$ for all i
$Q(S_i, Right) = 0$ for all i

Because the MDP is determistic, we run Q-learning with a learning rate $\alpha = 1$. Assume we start Q-learning at state S_1.

(a) Suppose our exploration policy is to always choose a random action. How many steps do we expect to take before we first enter state S_n?

(i) $O(n)$ steps
(ii) $O(n^2)$ steps
(iii) $O(n^3)$ steps
(iv) $O(2^n)$ steps
(v) It will certainly never happen

(b) Suppose our exploration is greedy and we break ties by going North:

Choose North if $Q(S_i, North) \geq Q(S_i, Right)$
Choose Right if $Q(S_i, North) < Q(S_i, Right)$

How many steps do we expect to take before we first enter state S_n?

(i) $O(n)$ steps
(ii) $O(n^2)$ steps
(iii) $O(n^3)$ steps
(iv) $O(2^n)$ steps
(v) It will certainly never happen

(c) Suppose our exploration is greedy and we break ties by going Right:

Choose North if $Q(S_i, North) > Q(S_i, Right)$
Choose Right if $Q(S_i, North) \leq Q(S_i, Right)$

How many steps do we expect to take before we first enter state S_n?

(i) $O(n)$ steps

(ii) $O(n^2)$ steps

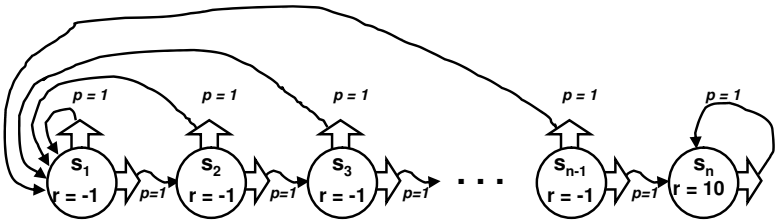
(iii) $O(n^3)$ steps

(iv) $O(2^n)$ steps

(v) It will certainly never happen

WARNING: Question (d) is only worth 1 point so you should probably just guess the answer unless you have plenty of time.

(d) In this question we work with a similar MDP except that each state other than S_n has a punishment (-1) instead of a reward (+1). S_n remains the same large reward (10). The new MDP is shown below:



Suppose our exploration is greedy and we break ties by going North:

Choose North if $Q(S_i, North) \geq Q(S_i, Right)$
Choose Right if $Q(S_i, North) < Q(S_i, Right)$

How many steps do we expect to take before we first enter state S_n?

(i) $O(n)$ steps

(ii) $O(n^2)$ steps

(iii) $O(n^3)$ steps

(iv) $O(2^n)$ steps

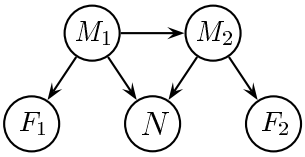
(v) It will certainly never happen

4 Bayesian Networks (11 points)

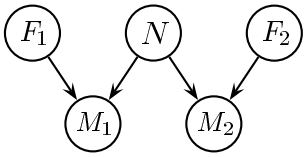
Construction. Two astronomers in two different parts of the world, make measurements M_1 and M_2 of the number of stars N in some small regions of the sky, using their telescopes. Normally, there is a small possibility of error by up to one star in each direction. Each telescope can be, with a much smaller probability, badly out of focus (events F_1 and F_2). In such a case the scientist will undercount by three or more stars or, if N is less than three, fail to detect any stars at all.

For questions (a) and (b), consider the four networks shown below.

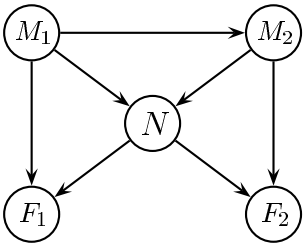
(i)



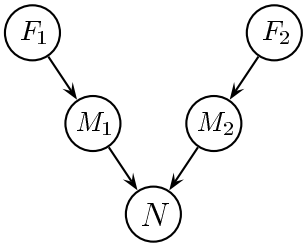
(ii)



(iii)



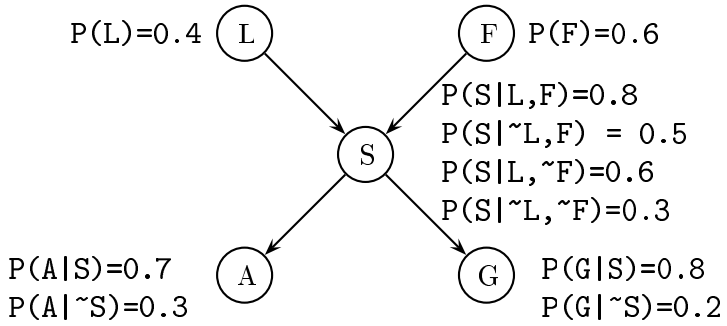
(iv)



(a) Which of them correctly, but not necessarily efficiently, represents the above information? **Note that there may be multiple answers.**

(b) Which is the best network?

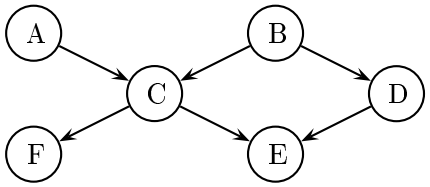
Inference. A student of the Machine Learning class notices that people driving SUVs (S) consume large amounts of gas (G) and are involved in more accidents than the national average (A). He also noticed that there are two types of people that drive SUVs: people from Pennsylvania (L) and people with large families (F). After collecting some statistics, he arrives at the following Bayesian network.



(c) What is $P(S)$?

(d) What is $P(S|A)$?

Consider the following Bayesian network. State whether the given conditional independences are implied by the net structure.



(f) **(True or False)** I<A,{},B>

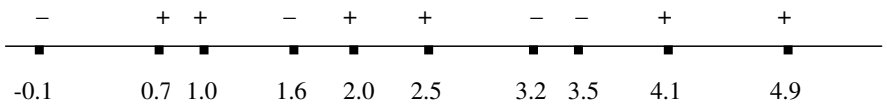
(g) **(True or False)** I<A,{E},D>

(h) **(True or False)** I<A,{F},D>

5 Instance Based Learning (8 points)

Consider the following dataset with one real-valued input x and one binary output y. We are going to use k-NN with unweighted Euclidean distance to predict y for x.

X	Y
-0.1	-
0.7	+
1.0	+
1.6	-
2.0	+
2.5	+
3.2	-
3.5	-
4.1	+
4.9	+



(a) What is the leave-one-out cross-validation error of 1-NN on this dataset? Give your answer as the number of misclassifications.

(b) What is the leave-one-out cross-validation error of 3-NN on this dataset? Give your answer as the number of misclassifications.

Consider a dataset with N examples: $\{(x_i, y_i)|1 \leq i \leq N\}$, where both x_i and y_i are real valued for all i. Examples are generated by $y_i = w_0 + w_1 x_i + e_i$ where e_i is a Gaussian random variable with mean 0 and standard deviation 1.

(c) We use least square linear regression to solve w_0 and w_1, that is $\{w_0^*, w_1^*\} = \arg\min_{\{w_0,w_1\}} \sum_{i=1}^{N} (y_i - w_0 - w_1 x_i)^2$.

We assume the solution is unique. Which one of the following statements is true?

(i) $\sum_{i=1}^{N} (y_i - w_0^* - w_1^* x_i) y_i = 0$

(ii) $\sum_{i=1}^{N} (y_i - w_0^* - w_1^* x_i) x_i^2 = 0$

(iii) $\sum_{i=1}^{N} (y_i - w_0^* - w_1^* x_i) x_i = 0$

(iv) $\sum_{i=1}^{N} (y_i - w_0^* - w_1^* x_i)^2 = 0$

(d) We change the optimization criterion to include local weights, that is $\{w_0^*, w_1^*\} = \arg\min_{\{w_0,w_1\}} \sum_{i=1}^{N} \alpha_i^2 (y_i - w_0 - w_1 x_i)^2$

where α_i is a local weight. Which one of the following statements is true?

(i) $\sum_{i=1}^{N} \alpha_i^2 (y_i - w_0^* - w_1^* x_i)(x_i + \alpha_i) = 0$

(ii) $\sum_{i=1}^{N} \alpha_i (y_i - w_0^* - w_1^* x_i) x_i = 0$

(iii) $\sum_{i=1}^{N} \alpha_i^2 (y_i - w_0^* - w_1^* x_i)(x_i y_i + w_1^*) = 0$

(iv) $\sum_{i=1}^{N} \alpha_i^2 (y_i - w_0^* - w_1^* x_i) x_i = 0$

6 VC-dimension (9 points)

Let H denote a hypothesis class, and $VC(H)$ denote its VC dimension.

(a) **(True or False)** If there exists a set of k instances that *cannot* be shattered by H, then $VC(H) < k$.

(b) **(True or False)** If two hypothesis classes H_1 and H_2 satisfy $H_1 \subseteq H_2$, then $VC(H_1) \leq VC(H_2)$.

(c) **(True or False)** If three hypothesis classes H_1, H_2 and H_3 satisfy $H_1 = H_2 \cup H_3$, then $VC(H_1) \leq VC(H_2) + VC(H_3)$.

For questions (d)–(f), give $VC(H)$. No explanation is required.

(d) $H = \{h_\alpha | 0 \leq \alpha \leq 1, h_\alpha(x) = 1$ iff $x \geq \alpha$ otherwise $h_\alpha(x) = 0\}$.

(e) H is the set of all perceptrons in 2D plane, i.e.
$H = \{h_{\mathbf{w}} | h_{\mathbf{w}} = \theta(w_0 + w_1 x_1 + w_2 x_2)$ where $\theta(z) = 1$ iff $z \geq 0$ otherwise $\theta_z = 0\}$.

(f) H is the set of all circles in 2D plane. Points inside the circles are classified as 1 otherwise 0.

7 SVM and Kernel Methods (8 points)

(a) Kernel functions implicitly define some mapping function $\phi(\cdot)$ that transforms an input instance $\mathbf{x} \in \mathbb{R}^d$ to a high dimensional feature space Q by giving the form of dot product in Q: $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j)$.

Assume we use radial basis kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{1}{2}||\mathbf{x}_i - \mathbf{x}_j||^2)$. Thus we assume that there's some implicit unknown function $\phi(\mathbf{x})$ such that

$$\phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) = K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\frac{1}{2}||\mathbf{x}_i - \mathbf{x}_j||^2)$$

Prove that for any two input instances $\mathbf{x}_i$ and $\mathbf{x}_j$, the squared Euclidean distance of their corresponding points in the feature space Q is less than 2, i.e. prove that $||\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)||^2 < 2$.

(b) With the help of a kernel function, SVM attempts to construct a hyper-plane in the feature space Q that maximizes the margin between two classes. The classification decision of any $\mathbf{x}$ is made on the basis of the sign of

$$\hat{\mathbf{w}}^T \phi(\mathbf{x}) + \hat{w}_0 = \sum_{i \in SV} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}) + \hat{w}_0 = f(\mathbf{x}; \alpha, \hat{w}_0),$$

where $\hat{\mathbf{w}}$ and $\hat{w}_0$ are parameters for the classification hyper-plane in the feature space Q, SV is the set of support vectors, and α_i is the coefficient for the support vector.

Again we use the radial basis kernel function. Assume that the training instances are linearly separable in the feature space Q, and assume that the SVM finds a margin that perfectly separates the points.

(True or False) If we choose a test point $\mathbf{x}_{far}$ which is far away from any training instance $\mathbf{x}_i$ (distance here is measured in the original space $\mathbb{R}^d$), we will observe that $f(\mathbf{x}_{far}; \alpha, \hat{w}_0) \approx \hat{w}_0$.

(c) **(True or False)** The SVM learning algorithm is guaranteed to find the globally optimal hypothesis with respect to its object function.

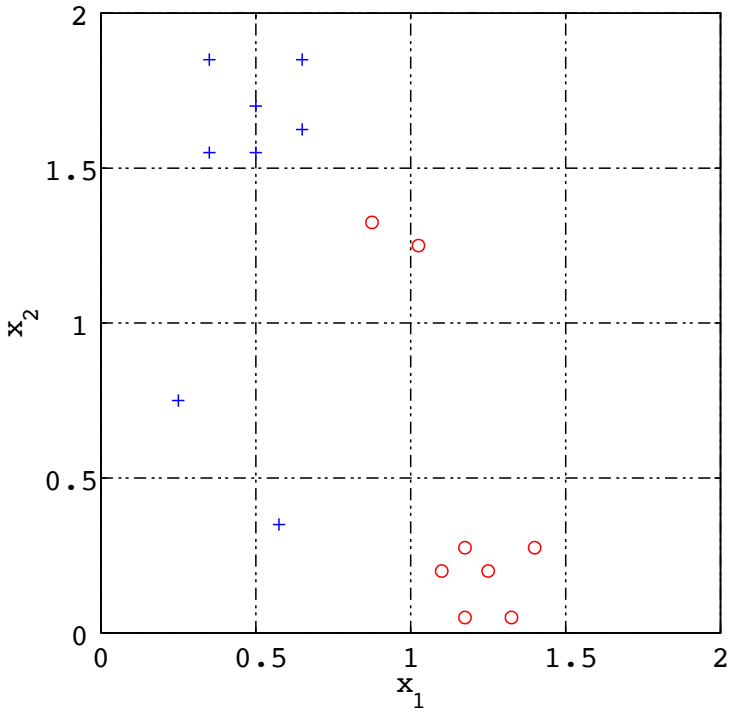
(d) **(True or False)** The VC dimension of a Perceptron is smaller than the VC dimension of a simple linear SVM.

(e) **(True or False)** After being mapped into feature space Q through a radial basis kernel function, a Perceptron may be able to achieve better classification performance than in its original space (though we can't guarantee this).

(f) **(True or False)** After mapped into feature space Q through a radial basis kernel function, 1-NN using unweighted Euclidean distance may be able to achieve better classification performance than in original space (though we can't guarantee this).

8 GMM (8 points)

Consider the classification problem illustrated in the following figure. The data points in the figure are labeled, where "o" corresponds to class 0 and "+" corresponds to class 1. We now estimate a GMM consisting of 2 Gaussians, one Gaussian per class, with the constraint that the covariance matrices are identity matrices. The mixing proportions (class frequencies) and the means of the two Gaussians are free parameters.



(a) Plot the maximum likelihood estimates of the means of the two Gaussians in the figure. Mark the means as points "x" and label them "0" and "1" according to the class.

(b) Based on the learned GMM, what is the probability of generating a new data point that belongs to class 0?

(c) How many data points are classified *incorrectly*?

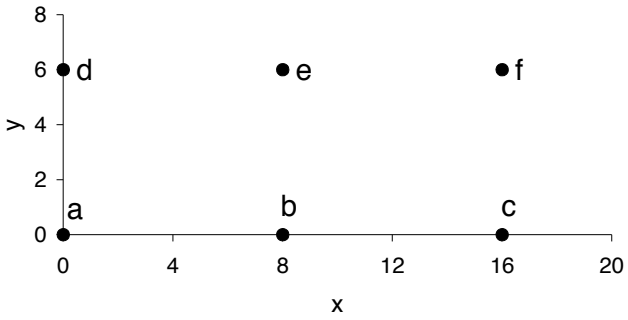
(d) Draw the decision boundary in the same figure.

9 K-means Clustering (9 points)

There is a set S consisting of 6 points in the plane shown as below, $a = (0,0)$, $b = (8,0)$, $c = (16,0)$, $d = (0,6)$, $e = (8,6)$, $f = (16,6)$. Now we run the k-means algorithm on those points with $k = 3$. The algorithm uses the Euclidean distance metric (i.e. the straight line distance between two points) to assign each point to its nearest centroid. Ties are broken in favor of the centroid to the left/down. Two definitions:

- A **k-starting configuration** is a subset of k starting points from S that form the initial centroids, e.g. $\{a, b, c\}$.
- A **k-partition** is a partition of S into k non-empty subsets, e.g. $\{a, b, e\}, \{c, d\}, \{f\}$ is a 3-partition.

Clearly any k-partition induces a set of k centroids in the natural manner. A k-partition is called *stable* if a repetition of the k-means iteration with the induced centroids leaves it unchanged.



(a) How many 3-starting configurations are there? (Remember, a 3-starting configuration is just a subset, of size 3, of the six datapoints).

(b) Fill in the following table:

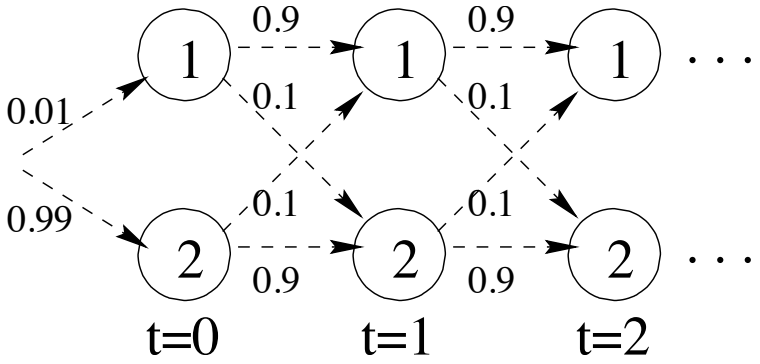
3-partition	Is it stable?	An example 3-starting configuration that can arrive at the 3-partition after 0 or more iterations of k-means (or write "none" if no such 3-starting configuration)	The number of unique starting configurations that can arrive at the 3-partition
$\{a,b,e\},\{c,d\},\{f\}$			
$\{a,b\},\{d,e\},\{c,f\}$			
$\{a,d\},\{b,e\},\{c,f\}$			
$\{a\},\{d\},\{b,c,e,f\}$			
$\{a,b\},\{d\},\{c,e,f\}$			
$\{a,b,d\},\{c\},\{e,f\}$			

10 Hidden Markov Models (8 points)

Consider a hidden Markov model illustrated as the figure shown below, which shows the hidden state transitions and the associated probabilities along with the initial state distribution. We assume that the state dependent outputs (coin flips) are governed by the following distributions

$P(x = heads|s = 1) = 0.51$
$P(x = heads|s = 2) = 0.49$
$P(x = tails|s = 1) = 0.49$
$P(x = tails|s = 2) = 0.51$

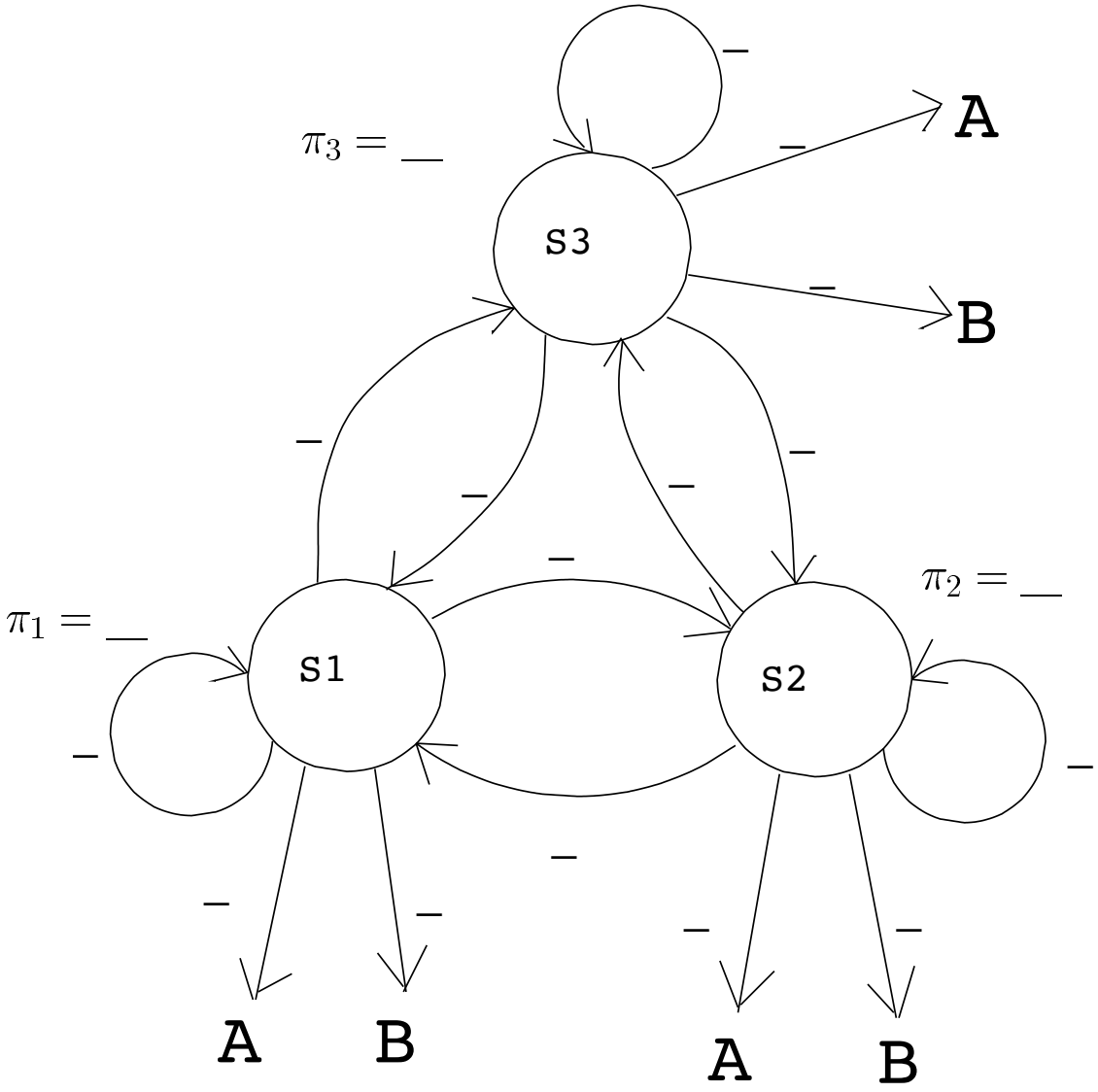
In other words, our coin is slightly biased towards *heads* in state 1 whereas in state 2 *tails* is a somewhat more probable outcome.



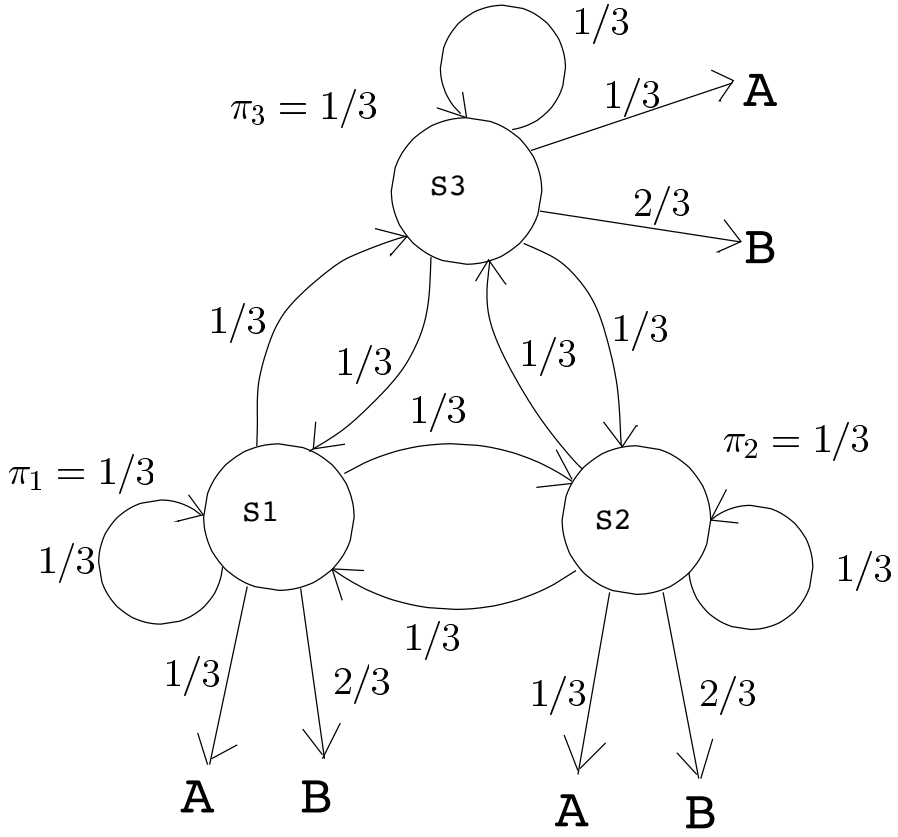
(a) Now, suppose we observe three coin flips all resulting in *heads*. The sequence of observations is therefore *heads*; *heads*; *heads*. What is the most likely state sequence given these three observations? (It is not necessary to use the Viterbi algorithm to deduce this, nor any subsequent questions).

(b) What happens to the most likely state sequence if we observe a long sequence of all heads (e.g., 10^6 heads in a row)?

(c) Consider the following 3-state HMM, π_1, π_2 and π_3 are the probabilities of starting from each state $S1$, $S2$ and $S3$. Give a set of values so that the resulting HMM maximizes the likelihood of the output sequence ABA.



(d) We're going to use EM to learn the parameters for the following HMM. Before the first iteration of EM we have initialized the parameters as shown in the following figure. **(True or False)** For these initial values, EM will successfully converge to the model that maximizes the likelihood of the training sequence ABA.



(e) **(True or False)** In general when are trying to learn an HMM with a small number of states from a large number of observations, we can almost always increase the training data likelihood by permitting more hidden states.

Full Name:_____

CS 15-681, 15-781 Fall 1998

Final Exam

December 1998

Instructions:

- Make sure that your exam is not missing any sheets, then write your name *on every page indicated*.
- This exam is open book, open notes.
- You have three hours to take this exam.
- Write your answers in the space provided. If you need extra space, use the back of the preceding sheet.
- Write clearly and be concise.

Full Name:_____

Problem 1. (points): Miscellaneous

A. Suppose H is a set of possible hypotheses and D is a set of training data. We would like our program to output the most probable hypothesis h from H, given the data D. Under what conditions does the following hold?

$$\underset{h \in H}{\text{argmax}}\, P(h|D) = \underset{h \in H}{\text{argmax}}\, P(D|h)$$

B. Name 2 similarities and 2 differences between Boosting (you may assume AdaBoost) and Bagging.

C. Explain in your own words why learning from examples is futile without some form of inductive bias.

D. For each of the following algorithms, (a) state the objective that the learning algorithm is trying to optimize, and (b) indicate whether the algorithm is guaranteed to find the global optimum hypothesis with respect to this objective.

- Backpropagation with multi-layer networks
  - Objective:
  - Global optimum?:

- The perceptron training rule applied to a single perceptron
  - Objective:
  - Global optimum?:

- The FindS algorithm from Chapter 2
  - Objective:
  - Global optimum?:

- Support Vector Machines.
  - Objective:
  - Global optimum?:

- The EM algorithm
  - Objective:
  - Global optimum?:

E. In one sentence each, give

- one advantage of ID3 over Backpropagation

- one advantage of Backpropagation over ID3

- one advantage of FOIL over ID3

- one advantage of Support Vector Machines over Perceptrons

F. Notice in the PAC learning result given by equation 7.2, the number m of required examples grows without bound as we set ϵ closer to zero. Explain in plain English (no equations please) why the learner cannot learn a zero error hypothesis in the PAC learning setting, even though it can if it gets to pose queries to a trainer.

Full Name:_____

Problem 2. (points): Naive Bayes

A. Consider a learning problem where each instance x is described by the boolean attributes $a_1 \ldots a_n$, and where the target function $f : X \rightarrow V$ has two possible values: v_1 and v_2. Write the decision rule learned by a Naive Bayes learner *in terms of an inequality.* Circle the parameters of this decision rule that are estimated from the training data.

B. What is the form of the hypothesis space H considered by a naive Bayes classifier for this problem? Show that H corresponds to a set of linear decision surfaces in a space with dimension $2n$. (Hint: start by taking the log of the above inequality.)

C. Given that the naive Bayes algorithm learns a linear decision surface in the Euclidean space $\Re^{2n}$, does this imply that the naive Bayes algorithm and the perceptron learning algorithm will learn the same hypothesis (assuming here that the perceptron has the same $2n$ inputs)? If so, explain how to construct the corresponding perceptron. If not, explain why the decision surfaces learned by naive Bayes and the perceptron learning algorithm can differ.

D. Can you use the PAC results for consistent learners to give a bound on the sample complexity for the above naive Bayes learner? If so, give it and explain the conditions under which it is correct. If not, explain the difficulty.

E. The PAC bounds we discussed are statements of the form

with probability $1 - \delta$, the (true) error of h is less than ...

.

This question asks you to determine analogous bounds on errors in the parameter estimates of the naive Bayes learning algorithm. Let $\hat{P}(v_j)$ represent the learner's estimate for $P(v_j)$, and $\hat{P}(a_i|v_j)$ represent its estimate for $P(a_i|v_j)$. Assume the learner is provided m training examples, including m_j examples with target value v_j, and m_{ij} examples with both target value v_j and $a_i = 1$. Derive statements of the form

with probability $1 - \delta$, the error in the estimate $\hat{P}(v_j)$ is less than ...

.

with probability $1 - \delta$, the error in the estimate $\hat{P}(a_i|v_j)$ is less than ...

.

Full Name:_____

Problem 3. (points): A Random Learner

Your friend asks your expert advice on her new learning algorithm, called LARCH (Learn A Random Consistent Hypothesis). The input to LARCH is any finite hypothesis space H and any consistent set of of training examples D. The LARCH algorithm is:

LARCH (training data D, hypothesis space H)

1. pick a hypothesis h at random from H
2. if h is consistent with all training examples in D, then output h and terminate, else
3. go to step 1.

A. She first wants to know whether it is possible to bound the error of the hypothesis output by LARCH, based on the number of training examples provided in D. What do you tell her? If yes, give the bound and justify why it applies to LARCH. If not, then explain why the PAC bounds on sample complexity do not apply.

B. The other algorithm your friend has tried is LACH (Learn All Consistent Hypotheses). LACH outputs the set of *all* hypotheses from H that are consistent with the training data D, then classifies each new instance by a vote of these consistent hypotheses (weighted equally). She has two questions about LACH and LARCH

   (a) Can you describe some condition under which one of these methods would be expected to produce a smaller true classification error than the other?

   (b) Can you bound the true classification error of LARCH in terms of the true classification error of LACH? If so, state precisely the assumptions that must be satisfied for this bound to hold.

Full Name:_____

Problem 4. (points): Miscellaneous

Answer each question in the space provided. Be concise and precise.

A. Amazing Web Technologies, Ltd. has hired you as a consultant for their latest genetic algorithm project: learning to distinguish which web home pages belong to Republicans versus Democrats. Due to the proprietary nature of their software product, they are unable to reveal to you the details of the hypothesis encoding used by the GA. However, they are willing to tell you that the GA encodes its hypothesis using a bitstring containing exactly 20 bits. They also reveal that this algorithm is allowed to run indefinitely until it outputs a hypothesis that classifies every training example correctly.

Their question to you is this: how many training examples of the boolean target function "Republican web pages" must they provide in order to assure that with 85% probability their GA will find a hypothesis whose true error is less than 15%? Please answer below.

They now run their GA and produce a hypothesis. When they test it on a set of 130 new instances they find that it commits 20 errors. What is the 90% confidence interval (two-sided) for the true error rate of this hypothesis? Give a one-sentence justification for your answer.

What is the 95% one-sided interval (i.e., what is the upper bound U such that $error_{\mathcal{D}}(h) \leq U$ with 95% confidence)? Give a one-sentence justification.

Full Name:_____

B. Instance-based learning methods such as nearest-neighbor simply store the training data when it is presented, and delay processing until a query instance is presented. For this reason they are called *lazy* learning methods, as opposed to *eager* methods such as ID3 and C4.5 that construct a general hypothesis at training time. Given a new query instance x, lazy methods have the advantage that they can construct a local approximation of the target function for the region of interest (i.e., the region near x).

Consider the eager Sequential Covering rule learning method, and the "Learn-one-Rule" algorithm summarized on pages 276–278 (Figure 10.1 and Tables 10.1 and 10.2). Suggest a lazy variant of this algorithm that delays learning until it is given a new query instance. Given the query instance x, this algorithm should use the training data to construct *a single rule that predicts the target value for* x. Describe your algorithm below, and show a trace analogous to the one in Figure 10.1. Use your trace to illustrate how your algorithm would operate when given the query instance

$$x = \langle Outlook = Sunny,\ Temperature = Hot,\ Humidity = High,\ Wind = Strong \rangle$$

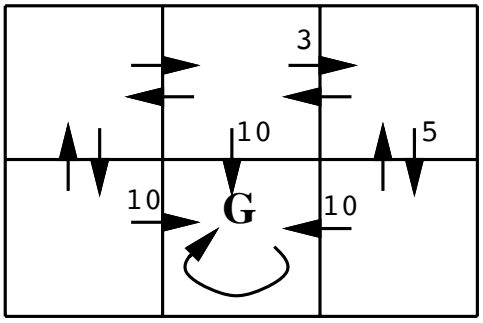
Does your lazy learning method ever classify instances differently than the eager algorithm described in the book? Explain your answer carefully.

Full Name:_____

Problem 5. (points): Reinforcement Learning

Consider the deterministic grid world shown below with the absorbing goal-state **G**. Here the immediate rewards $r(s, a)$ are 10, 5, or 3 for the labeled state-action transitions and 0 for all unlabeled transitions.

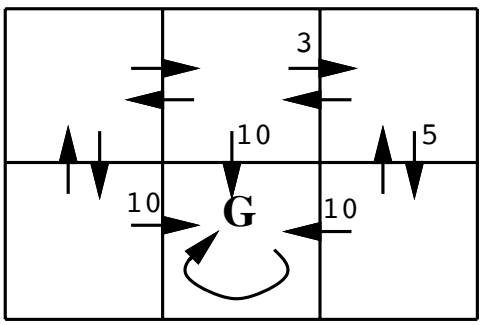
A. Write in the V^* values for each state in this grid world. Give the $Q(s, a)$ value for every transition. Finally, show an optimal policy. Use $\gamma = 0.8$.



B. Suggest a change to the reward function $r(s, a)$ that alters the $Q(s, a)$ values, but does not alter the optimal policy.

C. Suggest a change to $r(s, a)$ that alters $Q(s, a)$ but does not alter $V^*(s, a)$.

D. Now consider applying the Q learning algorithm to this grid world, assuming the table of $\hat{Q}$ values is initialized to zero. Assume the agent begins in the bottom left grid square and then travels clockwise around the perimeter of the grid until it reaches the absorbing goal state, completing the first training episode. Write in the new values of all $\hat{Q}$ values that are modified as a result of this episode. Answer the question again assuming the agent now performs a second identical episode (draw circles around the values you write in for this second episode, to distinguish them from the first values).



E. The task in reinforcement learning is to learn a policy $S \rightarrow A$ to choose an appropriate action a from the set A, given the current state s from the set S. The difficulty is that this is to be accomplished based only on indirect, delayed rewards. Q learning accomplishes this by instead learning an evaluation function over state-action pairs.

Suppose that you wish to learn the target function $S \rightarrow A$ directly, rather than learning an evaluation function. Consider Backpropagation, Genetic Algorithms, and Decision Tree learning. Which of these methods, if any, can be used to learn the function $S \rightarrow A$ from the kind of training experience that is provided to the Q learner in the above problem? For each of these three algorithms, explain why it is not possible, or sketch an approach.