6.867 Machine learning

Final exam (Fall 2003)

December 14, 2003

Problem 1: your information

1.1. Your name and MIT ID:

1.2. The grade you would give to yourself + brief justification (if you feel that there's no question your grade should be an A, then just say A):

A ... why not?

Problem 2

2.1. (3 points) Let $\mathcal{F}$ be a set of classifiers whose VC-dimension is 5. Suppose we have four training examples and labels, $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_4, y_4)\}$, and select a classifier $\hat{f}$ from $\mathcal{F}$ by minimizing classification error on the training set. In the absence of any other information about the set of classifiers $\mathcal{F}$, can we say that the prediction $\hat{f}(\mathbf{x}_5)$ for a new example $\mathbf{x}_5$ has any relation to the training set? Briefly justify your answer.

Since VC-dimension is 5, F can shatter (some) five points. These points could be $x_1, \ldots, x_5$. Thus we can find $f_1 \in \mathcal{F}$ consistent with the four training examples and $f_1(\mathbf{x}_5) = 1$, as well as another classifier $f_{-1} \in \mathcal{F}$ also consistent with the training examples for which $f_{-1}(\mathbf{x}_5) = -1$. The training set therefore doesn't constrain the prediction at $\mathbf{x}_5$.

2.2. (T/F – 2 points) Consider a set of classifiers that includes all linear classifiers that use different choices of strict subsets of the components of the input vectors $\mathbf{x} \in \mathcal{R}^d$. Claim: the VC-dimension of this combined set cannot be more than $d + 1$.

T

A linear classifier based on a subset of features can be represented as a linear classifier based on all the features but we have simply set some of the parameters to zero. The set of classifiers here is therefore simply linear classifiers in $\mathcal{R}^d$.

2.3. (T/F – 2 points) Structural risk minimization is based on comparing upper bounds on the generalization error, where the bounds hold with probability $1 - \delta$ over the choice of the training set. Claim: the value of the confidence parameter δ cannot affect model selection decisions.

F

The δ parameter changes the complexity penalty in a manner that depends on the VC-dimension and the number of training examples. It can therefore affect the model selection results.

2.4. (6 points) Suppose we use class-conditional Gaussians to solve a binary classification task. The covariance matrices of the two Gaussians are constrained to be $\sigma^2 I$, where the value of σ^2 is fixed and I is the identity matrix. The only adjustable parameters are therefore the means of the class conditional Gaussians, and the prior class frequencies. We use the maximum likelihood criterion to train the model. Check all that apply.

( ) For any three distinct training points and sufficiently small σ^2, the classifier would have zero classification error on the training set

( ) For any three training points and sufficiently large σ^2, the classifier would always make one classification error on the training set

( ) The classification error of this classifier on the training set is always at least that of a linear SVM, whether the points are linearly separable or not

None is the correct answer. The classifier is trained as a generative model, thus it places each Gaussian at the sample mean of the points in the class. The prior class frequencies will reflect the number of points in each class.

- *Since points in one class can be further away from their mean than from points in the other class, the first option cannot be correct.*
- *For the second option we can assume that the variance σ^2 is much larger than the distances between the points. In this case the Gaussians from each class assign roughly the same probability to any training point. The prior class frequencies favor the dominant class, therefore resulting in one error. This does not hold, however, when all the training points come from the same class. Thus you got 2pts for either answer to the second.*
- *The last option is incorrect since SVM does not minimize the number of misclassified points when the points are not linearly separable; the accuracy could go either way, albeit typically in favor of SVMs.*

Problem 3

3.1. (T/F – 2 points) In the AdaBoost algorithm, the weights on all the misclassified points will go up by the same multiplicative factor.

T

You can verify this by inspecting the weight update. Note that the adaboost algorithm was formulated for weak classifiers that predict ± 1 *labels. The weights of all misclassified points will be multiplied by* $\exp(-\hat{\alpha} y_i h(\mathbf{x}_i; \hat{\theta}_k)) = \exp(\hat{\alpha})$ *before normalization.*

3.2. (3 points) Provide a brief rationale for the following observation about AdaBoost. The weighted error of the k^{th} weak classifier (measured relative to the weights at the beginning of the k^{th} iteration) tends to increase as a function of the iteration k.

The weighting on the training examples focuses on examples that are hard to classify correctly (few weak classifiers have classified such examples correctly). After a few iterations most of the weight will be on these hard examples and the weighted error on the next weak classifier will be closer to chance.

Consider a text classification problem, where documents are represented by binary (0/1) feature vectors $\phi = [\phi_1, \ldots, \phi_m]^T$; here ϕ_i indicates whether word i appears in the document. We define a set of weak classifiers, $h(\phi; \theta) = y\phi_i$, parameterized by $\theta = \{i, y\}$ (the choice of the component, $i \in \{1, \ldots, m\}$, and the class label, $y \in \{-1, 1\}$, that the component should be associated with). There are exactly $2m$ possible weak learners of this type.

We use this boosting algorithm for feature selection. The idea is to simply run the boosting algorithm and select the features or components in the order in which they were identified by the weak learners. We assume that the boosting algorithm finds the best available weak classifier at each iteration.

3.3. (T/F – 2 points) The boosting algorithm described here can select the exact same weak classifier more than once.

T

The boosting algorithm optimizes each new α *by assuming that all the previous votes remain fixed. It therefore does not optimize these coefficients jointly. The only way to correct the votes assigned to a weak learner later on is to introduce the same weak learner again. Since we only have a discrete set of possible weak learners here, it also makes sense to talk about selecting the exact same weak learner again.*

3.4. (4 points) Is the ranking of features generated by the boosting algorithm likely to be more useful for a linear classifier than the ranking from simple mutual information calculations (estimates $\hat{I}(y; \phi_i)$). Briefly justify your answer.

The boosting algorithm generates a linear combination of weak classifiers (here features). The algorithm therefore evaluates each new weak classifier (feature) relative to a linear prediction based on those already included. The mutual information criterion considers each feature individually and is therefore unable to recognize how multiple features might interact to benefit linear prediction.

Problem 4

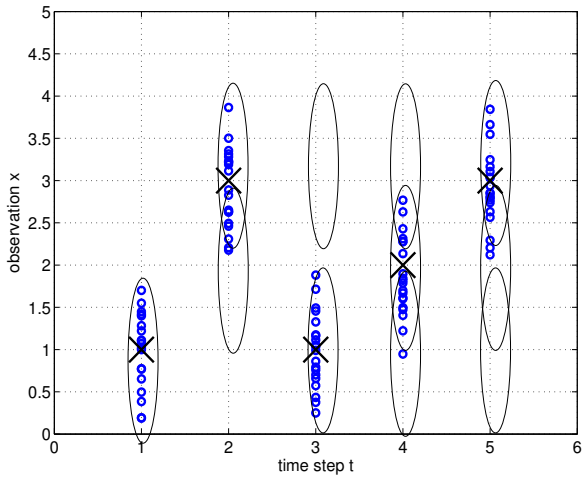


Figure 1a)

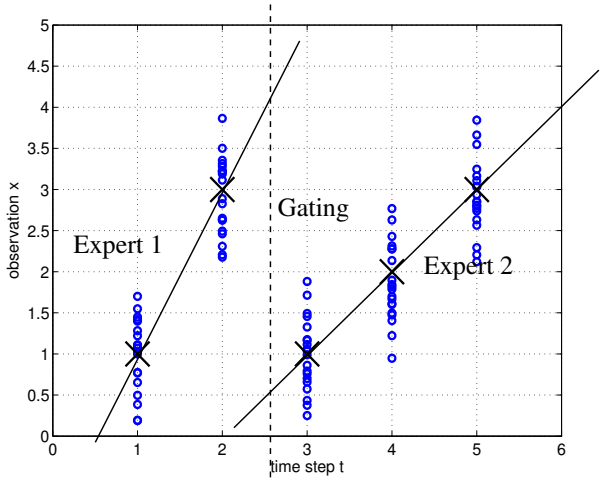
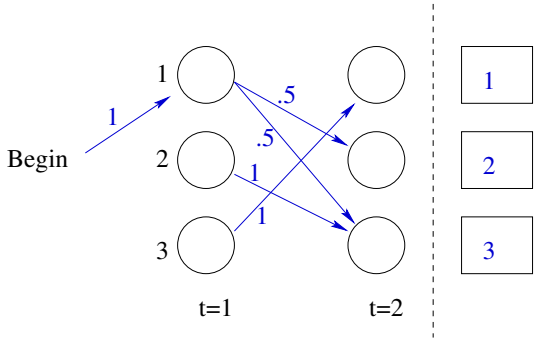


Figure 1b)

Figure 1: Time dependent observations. The data points in the figure are generated as sets of five consecutive time dependent observations, $x_1, \ldots, x_5$. The clusters come from repeatedly generating five consecutive samples. Each visible cluster consists of 20 points, and has approximately the same variance. The mean of each cluster is shown with a large X.

Consider the data in Figure 1 (see the caption for details). We begin by modeling this data with a three state HMM, where each state has a Gaussian output distribution with some mean and variance (means and variances can be set independently for each state).

4.1. (4 points) Draw the state transition diagram and the initial state distribution for a three state HMM that models the data in Figure 1 in the maximum likelihood sense. Indicate the possible transitions and their probabilities in the figure below (whether or not the state is reachable after the first two steps). In order words, your drawing should characterize the 1st order homogeneous Markov chain govering the evolution of the states. Also indicate the means of the corresponding Gaussian output distributions (please use the boxes).



4.2. (4 points) In Figure 1a draw as ovals the clusters of outputs that would form if we repeatedly generated samples from your HMM over time steps $t = 1, \ldots, 5$. The height of the ovals should reflect the variance of the clusters.

4.3. (4 points) Suppose at time $t = 2$ we observe $x_2 = 1.5$ but don't see the observations for other time points. What is the most likely state at $t = 2$ according to the marginal posterior probability $\gamma_2(s)$ defined as $P(s_2 = s|x_2 = 1.5)$.

2

We have only two possible paths for the first three states, 1,2,3, or 1,3,1. The marginal posterior probability comes from averaging the state occupancies across these possible paths, weighted by the corresponding probabilities. Given the observation at $t = 2$ (mean of the output distribution from state 2), the first path is more likely.

4.4. (2 points) What would be the most likely state at $t = 2$ if we also saw $x_3 = 0$ at $t = 3$? In this case $\gamma_2(s) = P(s_2 = s|x_2 = 1.5, x_3 = 0)$.

3

The new observation at $t = 3$ is very unlikely to have come from state 3, thus we switch to state sequence 1,3,1.

4.5. (4 points) We can also try to model the data with conditional mixtures (mixtures of experts), where the conditioning is based on the time step. Suppose we only use two experts which are linear regression models with additive Gaussian noise, i.e.,

$$P(x|t, \theta_i) = \frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\Big\{ -\frac{1}{2\sigma_i^2}(x - \theta_{i0} - \theta_{i1}t)^2 \Big\}$$

for $i = 1, 2$. The gating network is a logistic regression model from t to binary selection of the experts. Assuming your estimation of the conditional mixture model is successfully in the maximum likelihood sense, draw the resulting mean predictions of the two linear regression models as a function of time t in Figure 1b). Also, with a vertical line, indicate where the gating network would change it's preference from one expert to the other.

4.6. (T/F – 2 points) Claim: by repeatedly sampling from your conditional mixture model at successive time points $t = 1, 2, 3, 4, 5$, the resulting samples would resemble the data in Figure 1

T

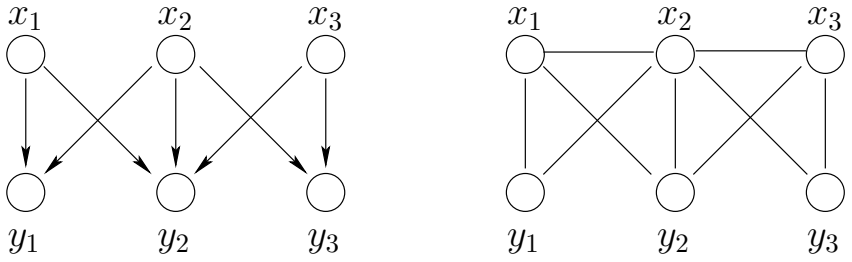
.

See the figure.

4.7. (4 points) Having two competing models for the same data, the HMM and the mixture of experts model, we'd like to select the better one. We think that any reasonable model selection criterion would be able to select the better model in this case. Which model would we choose? Provide a brief justification.

The mixture of experts model assigns a higher probability to the available data since the HMM puts some of the probability mass where there are no points. The HMM also has more parameters so any reasonable model selection criterion should select the mixture of experts model.

Problem 5



a) Bayesian network (directed) b) Markov random field (undirected)

Figure 2: Graphical models

5.1. (2 points) List two different types of independence properties satisfied by the Bayesian network model in Figure 2a.

1) x_1 and x_2 are marginally independent.
2) y_1 and y_2 are conditionally independent given x_1 and x_2.
Lots of other possibilities.

5.2. (2 points) Write the factorization of the joint distribution implied by the directed graph in Figure 2a.

$P(x_1)P(x_2)P(x_3)P(y_1|x_1, x_2)P(y_2|x_1, x_2, x_3)P(y_3|x_2, x_3)$.

5.3. (2 points) Provide an alternative factorization of the joint distribution, different from the previous one. Your factorization should be consistent with all the properties of the directed graph in Figure 2a. Consistency here means: whatever is implied by the graph should hold for the associated distribution.

Any factorization that incorporates all the independencies from the graph and a few more would be possible. For example, $P(x_1)P(x_2)P(x_3)P(y_1)P(y_2)P(y_3)$. In this case all the variables are independent, so any independence statement that we can derive from the graph clearly holds for this distribution as well.

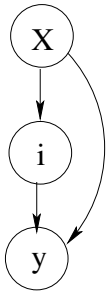
5.4. (4 points) Provide an independence statement that holds for the undirected model in Figure 2b but does NOT hold for the Bayesian network. Which edge(s) should we add to the undirected model so that it would be consistent with (wouldn't imply anything that is not true for) the Bayesian network?

x_1 is independent of x_3 given x_2 and y_2. In the bayesian network any knowledge of y_2 would make x_1 and x_3 dependent. Adding an edge between x_1 and x_3 would suffice (cf. moralization).

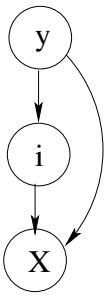
5.5. (2 points) Is your resulting undirected graph triangulated (Y/N)?

Y

5.6. (4 points) Provide two directed graphs representing 1) a mixture of two experts model for classification, and 2) a mixture of Gaussians classifiers with two mixture components per class. Please use the following notation: $\mathbf{x}$ for the input observation, y for the class, and i for any selection of components.



A mixture of experts classifier, where $i = 1, 2$ selects the expert.



A mixture of Gaussians model, where $i = 1, 2$ selects the Gaussian component within each class.