

Midterm

*Professor: Eric Xing**Date: October 26, 2011*

- . There are 6 questions in this exam (13 pages including this cover sheet)
- . Questions are not equally difficult.
- . This exam is open book and open notes. Computers, PDAs, Cell phones are not allowed.
- . You have one hour and twenty minutes.
- . Good luck!

Last Name:**First Name:****Andrew Id:**

Q	Topic	Max. Score	Score
1	Assorted	26	
2	K-means and EM	18	
3	Classification methods	18	
4	Neural networks	11	
5	Decision trees	13	
6	HMM	14	
Total		100	

1 Assorted questions [26 points]

1.1 True or False

Please answer whether the following statement is true or not. Briefly explain why or give a counterexample.

If your answer is correct but explanation is wrong, you only get 1 point.

- [2 points]** Suppose that rain is the only cause of thunder and grass getting wet. Then the probability of hearing a thunder given it is raining is the same as the probability of hearing a thunder given it is raining and the grass is wet.

True. The question says that thunder and grass getting wet are independent given rain. The second line is an equivalent way of expressing that idea.
- [2 points]** Given infinite data, the Naive Bayes classifier will be able to learn the rules used by the Bayes optimal classifier.

False. If the conditional independence of the NB classifier does not hold, it will never be able to learn the rule used by the Bayes optimal classifier.
- [2 points]** Linear regression means the model should be linear to the input variables as well as the parameters.

False. Linear regression means the model should be linear to the parameters. Basis function linear regression allows non-linear models with respect to input variables.
- [2 points]** Regardless of the size of the neural network, the back-propagation learning algorithm always finds the globally optimal weights for the neural network.

False. If the error is nonconvex (due to non-linear units such as the sigmoid), it can get stuck in local optima.
- [2 points]** The more hidden layers a neural network has, the better it can predict desired outputs for new inputs that it was not trained with.

False. With more hidden layers, the NN is likely to overfit.
- [2 points]** For any model with latent variables z and parameters θ , the EM algorithm alternates between the following two steps (1) replacing each occurrence of z in the likelihood function with its expectation $\langle z \rangle$, and (2) maximizing this likelihood function with respect to the model parameters θ .

False. Step 1 replaces the sufficient statistics of the hidden variables by their expected values, as we see in the HMM Baum-Welch algorithm.
- [2 points]** If X is Gaussian with mean 0 and variance 1, $\text{Probability}(X=0) > \text{Probability}(X=1)$.

False. Since X is continuous, $\text{Probability}(X=i) = 0$ for any i . Therefore both probabilities are zero and hence equal. (Note: Some people interpreted this wrongly as the pdf. However, due to some confusion in the explanation, we decided to give everyone 2 points for this question.)
- [2 points]** Decision trees learn linear decision boundaries.

False. A decision tree can learn the XOR boundary.

9. [2 points] With K -Gaussians EM, we can select the best K as follows: compute the log-likelihood $P(x|\theta)$ for each K , and pick the K that gives the highest value.

False. Adding more gaussians only increases likelihood. In the extreme case, one data point per gaussian gives near-perfect likelihood.

10. [2 points] The Hidden Markov Model directly models the dependency of each hidden state on all previous hidden states.

False. It only models dependence on one previous hidden state.

1.2 Multiple-choice questions

[3 points] Which of these properties do the K -means algorithm and K -Gaussians EM algorithm have in common? Tick the correct options (there may be more than one correct answer).

1. They optimize a lower bound on the complete log-likelihood of the data
2. They optimize a lower bound on the incomplete log-likelihood of the data
3. They perform gradient descent (or ascent) for their objective function
4. They perform alternating minimization (or maximization) for their objective function
5. They assume the clusters are Gaussian

Only option 4 is correct. If you marked 4, you get +3 points, For every other option marked, you lose 1 point if your score for this question is greater than 0.

[3 points] Suppose you observe the following sequence of real numbers:

1.1, 3.1, 2.5, 3.6, 5.0, 9.4, 8.1, 7.2, 9.9, 6.3, 6.1, 2.2, 2.3, 1.7, 3.2, 9.5, 8.2

Which of the following unsupervised techniques would you choose to model this data?

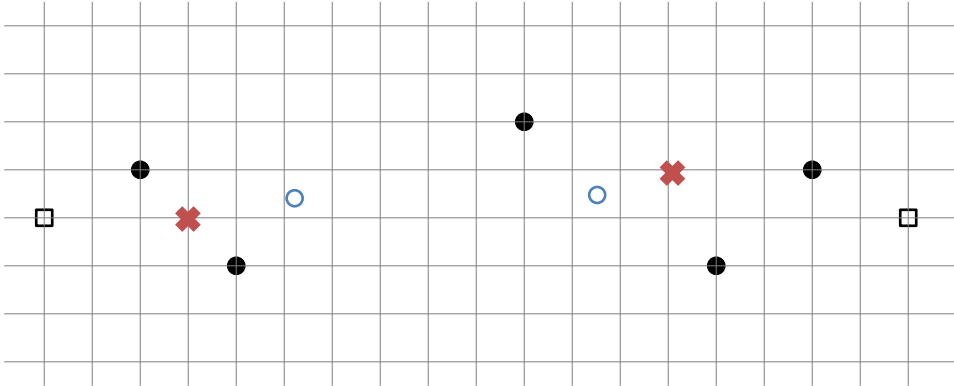
1. K-Multinomials Mixture Model
2. Hidden Markov Model
3. Spectral Clustering
4. K-Means

Since the data is real numbers, option 1 is incorrect and you lose 1.5 points if you mark that. You get 1.5 points for marking 2 (HMM) as correct. The other two options can also be marked correct since they can be used for modeling the data.

2 K -means and EM [18 points]

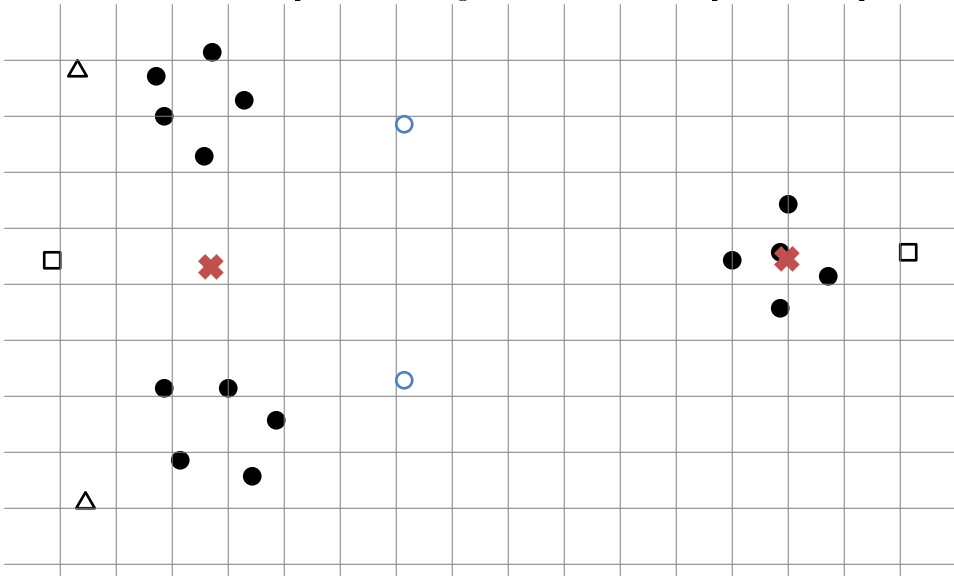
2.1 Short questions

Consider the dataset depicted on the grid below. The data points are represented by filled black circles. Hollow shapes do not represent data points.



- 1. [3 points] Suppose we initialize the K -means algorithm (with $K = 2$) to the two hollow squares, after which we run the algorithm. On the diagram, mark the final positions of the two centers with an 'x'. You only need to draw the positions qualitatively; there is no need to compute the exact locations.
- 2. [3 points] Suppose we initialize a K -Gaussian mixture model (with $K = 2$) to the two hollow squares, after which we run the EM algorithm. On the diagram, mark the final centers of the two Gaussians with an 'o'. Again, do this qualitatively.

Now consider the dataset depicted on the grid below. The data points are represented by filled black circles.



- 3. [3 points] Suppose we initialize a K -means algorithm (with $K = 2$) to the two hollow squares, after which we run the algorithm. On the diagram, mark the final positions of the two centers with an 'x'. Again, do this qualitatively.
- 4. [3 points] Suppose we initialize a K -means algorithm (with $K = 2$) to the two hollow triangles, after which we run the algorithm. On the diagram, mark the final positions of the two centers with an 'o'. Again, do this qualitatively.

Note: For part 4, we accepted several other answers.

2.2 EM for a mixture of K multinomials

In class, we applied the EM algorithm to the K -Gaussians mixture model. Now, let's explore what happens when we change the Gaussians into multinomials.

Let x_1, \dots, x_n be D -dimensional indicator vectors (i.e. exactly one element equal to '1', and the rest '0'). For example, $[0, 0, 1, 0]$ is such a vector. Let $\theta_1, \dots, \theta_K$ be the D -dimensional parameter vectors for the K multinomials. Finally, let z_1, \dots, z_n be the K -dimensional latent class indicator vectors for each datapoint, and let π be the D -dimensional prior probability for the latent classes. For simplicity, assume that $\pi_1, \dots, \pi_K = K^{-1}$, i.e. a uniform prior distribution.

Thus, our mixture of K multinomials has the following probability distributions:

$$p(z_i | \pi) = \prod_{k=1}^K (\pi_k)^{z_i^k} = \frac{1}{K}$$

$$p(x_i | z_i^k = 1, \theta) = \prod_{d=1}^D (\theta_k^d)^{x_i^d},$$

where the notation z_i^k denotes the k -th element of z_i , and similarly for x_i^d and θ_k^d .

- **1. [3 points]** When we apply the EM algorithm to this model, we are optimizing some function. Explicitly write out this function in terms of data $\{x_i\}$ and parameters θ and π . (Hint: z is a unobserved random variable here and therefore can not be in the objective. How can you overcome this problem?)

$$\langle \ell_c(\theta, \pi; x, z) \rangle = \sum_{i=1}^N \sum_{k=1}^K \langle z_i^k \rangle \sum_{d=1}^D x_i^d \log(\theta_k^d) + \sum_{i=1}^N \sum_{k=1}^K \langle z_i^k \rangle \log(\pi_k)$$

- **2. [3 points]** Under the Gaussian mixture model, recall that

$$\langle z_i^k \rangle_q = \frac{N(x_i | \mu_k, \Sigma_k)}{\sum_{\ell=1}^K N(x_i | \mu_\ell, \Sigma_\ell)},$$

where N is the Gaussian probability density function, and where we have assumed $\pi_k = \frac{1}{K}$ so that the prior probabilities of z_i don't show up. Write down the expression for $\langle z_i^k \rangle_q$ under our K multinomial model.

$$\langle z_i^k \rangle_q = \frac{\prod_{d=1}^D (\theta_k^d)^{x_i^d}}{\sum_{\ell=1}^K \prod_{d=1}^D (\theta_\ell^d)^{x_i^d}}$$

3 Classification [18 points]

Let's consider a classification problem. Assume we have m continuous variables, x^1, x^2, \dots, x^m and a boolean variable, y . Assume $P(y = k) = \pi_k$, $P(x_1, x_2, \dots, x_m | y = k)$ follows a multivariate normal distribution, $\mathcal{N}(\vec{\mu}_k, \Sigma_k)$. We would like to predict y based on x^1, x^2, \dots, x^m . We have a set of training data, $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, where $x_i = (x_i^1, x_i^2, \dots, x_i^m)$.

3.1 Naive Bayes Classifier

Alice proposes to use naive Bayes classifier.

- (1) [2 points] Please write down the expression used in naive Bayes to predict $P(y = 1 | x^1, x^2, \dots, x^m)$.

$$P(y = 1 | x^1, x^2, \dots, x^m) = \frac{\pi_1 \prod_{i=1}^m P(x^i | y = 1)}{\sum_{k=0}^1 \pi_k \prod_{i=1}^m P(x^i | y = k)} \quad (1)$$

where $P(x^i | y = 1) = \frac{1}{\sqrt{2\pi\Sigma_{1,ii}}} e^{-\frac{(x^i - \mu_{1,i})^2}{2\Sigma_{1,ii}}}$

- (2) [2 points] How many parameters do we need in this naive Bayes classifier?

Prior parameter: 1

2m parameters for each class

Total: $1 + 2 \times 2m = 1 + 4m$

- (3) [2 points] However, the naive Bayes classifier does not perform well. Alice checks the data and finds out that the predicted value of $P(y = 1 | x^1, x^2, \dots, x^m)$ calculated from the expression in problem (1) is not equivalent to the actual distribution. Explain why in one sentence.

Conditional independence assumption does not hold.

3.2 Logistic Regression

Since the naive Bayes classifier does not work well, Bob proposes to use logistic regression instead.

1. [2 points] To train logistic regression, Bob needs to maximize the conditional likelihood of training data, $l(\theta)$. Please write down the expression for $l(\theta)$.

$$l(\theta) = \sum_{n=1}^N \ln P(y_n | x_n; \theta) \quad (2)$$

$$= \sum_{n=1}^N (y_n - 1)\theta^T x_n + \ln(1 + e^{-\theta^T x_n}) \quad (3)$$

2. [2 points] Is naive Bayes also maximizing the conditional likelihood of training data during training? If yes, please briefly explain why. If not, please write down the expression used in Naive Bayes, and briefly describe the difference between the objective function of logistic regression and that of naive Bayes in one sentence.

No, naive Bayes is maximizing the likelihood of training data.

$$\ln \prod_{n=1}^N P(x_n, y_n | \theta) \quad (4)$$

3. [2 points] Bob decides to use gradient ascent to train the parameters. Please give at least one advantage and one limitation of the proposed algorithm.

Advantage: Since the objective function for logistic regression is concave, we will reach global optimum.

Limitation: Slow to converge.

Note: If you say gradient ascent may stuck in local optimum. You are still given partial credit, since it is true in general for gradient ascent. But it is not true for logistic regression.

3.3 Feature Selection

Later, Bob finds out that only a small number of features are actually relevant to the learning task. He wants to know which are the relevant features, and only uses the relevant features to train the classifier.

1. **[3 points]** Hence, Bob changes the objective function from problem 3.2.1 a little to prefer sparsity. Please write down the modified objective function. Note that if you would like to introduce new parameters in the objective function, please explicitly specify how the value of the parameter could affect sparsity (e.g. the larger the parameter is, the more sparsity the modified objective function prefers).

$$l(\theta) - \lambda \sum_i |\theta_i| \quad (5)$$

Add L1 norm, the larger the λ , the more sparsity the modified objective function prefers.

Note: Since we are maximizing the likelihood, the penalty term should be subtracted from the original objective function.

2. **[3 points]** Another way to find the relevant features is forward selection. In this approach, one adds variables to the model one at a time. At each step, each variable that is not already in the model is tested for inclusion in the model. The variable that improves the model performance the most is added to the model. Suppose you are given a series of models obtained using forward selection, each model having one more feature than the previous model. Suggest a criterion that could be used to decide which of these models is best for the learning problem. Briefly explain your choice.

Acceptable answers: cross validation, AIC, BIC, TIC and any measurement that is sensitive to the model complexity.

Answers such as the likelihood function, or information gain are not accepted. By using the likelihood function, the value will keep increase as you add more features, which will lead to overfitting. Information gain is measuring the feature quality without considering the model performance.

4 Neural Network Representation [11 points]

Consider a neural network with one hidden layer (as shown in figure 1), where X_1 , X_2 and X_3 are 3 input attributes to the network. Suppose you have two types of activation functions at hand (notice the slight change of notation from HW1):

- Signed sigmoid function: $S(a) = \text{sign}[\sigma(a) - 0.5] = \text{sign}\left[\frac{1}{1+e^{-a}} - 0.5\right] = \begin{cases} 1 & \text{if } \frac{1}{1+e^{-a}} \geq 0.5 \\ -1 & \text{otherwise} \end{cases}$
- Linear function: $L(a) = ca$, where c is a constant

where in both cases $a = \sum_i w_i X_i$. In other words, each node in the network first computes a weighted sum a using its inputs and corresponding weights, then apply its activation function L or S on the weighted sum a .

1. **(4 points)** Assign proper activation functions (S or L) to each unit in the following neural network (figure 1), such that it simulates a binary logistic regression classifier: $Y = \arg \max_y P(Y = y|X)$, where $P(Y = 1|X) = \frac{\exp(\beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3)}{1 + \exp(\beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3)}$ and $P(Y = -1|X) = \frac{1}{1 + \exp(\beta_1 X_1 + \beta_2 X_2 + \beta_3 X_3)}$

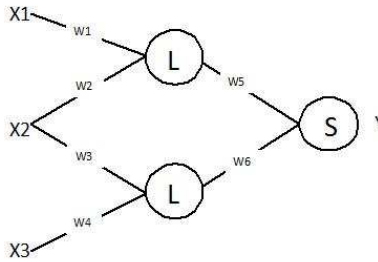


Figure 1: question 1

2. **(3 points)** For question 1, derive β_1 , β_2 and β_3 in terms of $\{w_1, w_2, w_3, w_4, w_5, w_6\}$.

Answer:

$$\begin{aligned} \beta_1 &= cw_1w_5 \\ \beta_2 &= c(w_2w_5 + w_3w_6) \\ \beta_3 &= cw_4w_6 \end{aligned}$$

3. Consider a training set composed of N data points, $\{(X_1, y_1), \dots, (X_N, y_N)\}$, where $X_i \in \mathbb{R}^d$ and $y_i \in \{+1, -1\}$. We have tried logistic regression on this training set, but failed to get satisfactory results. Thus, we decide to use more powerful basis functions $\{\phi_j(X)\}$, where $\phi_j(X)$ is the product of a subset of elements in X and $\{\phi_j(X)\}$ is the set of all such $\phi_j(X)$. For example, assume $X = (X^{(1)}, X^{(2)}) \in \mathbb{R}^2$, then $\{\phi(X)\}$ is $\{1, X^{(1)}, X^{(2)}, X^{(1)}X^{(2)}\}$ (where 1 corresponds to the case where no element of X is selected in the product), and the decision function in logistic regression becomes $\sum_j w_j \phi_j(X)$.

For $X \in \mathbb{R}^d$, how many features would we end up with?

Answer: 2^d

Assume $N = 1000$ and $d = 10$, what is an obvious problem with the above “non-linear” formulation?

Answer: The number of features will be $2^d = 1024 > 1000$. Therefore, we will have over-fitting problem.

5 Decision trees [13 points]

5.1 VC dimension

[3 points] Consider a decision tree of depth 2 with binary splits. What is the VC dimension of such a tree? A decision tree of depth 2 with binary splits has 4 leaves, so it can correctly classify 4 points for any labeling by putting a point in each leaf. If there are 5 points, at least 2 points must be in the same leaf and therefore cannot have opposite labels. So a set of size 4 can be shattered but no set of 5 points can be shattered, so VC dimension = 4

5.2 Constructing decision trees

Consider the problem of predicting whether the university will be closed on a particular day. We will assume that the factors which decide this are whether there is a snowstorm, whether it is a weekend or an official holiday. Suppose we have the training examples described in the Table 5.2.

Snowstorm	Holiday	Weekend	Closed
T	T	F	F
T	T	F	T
F	T	F	F
T	T	F	F
F	F	F	F
F	F	F	T
T	F	F	T
F	F	F	T

Table 1: Training examples for decision tree

- [2 points] What would be the effect of the Weekend attribute on the decision tree if it were made the root? Explain in terms of information gain.

If we split on Weekend, all the data goes to the same subtree, therefore $H(\text{subtree}_1) = H(\text{root})$ and $H(\text{subtree}_2) = 0$ and as a result $IG(\text{Weekend}) = 0$. So there is no improvement by making Weekend the root

- [8 points] If we cannot make Weekend the root node, which attribute should be made the root node of the decision tree? Explain your reasoning and show your calculations. (You may use $\log_2 0.75 = -0.4$ and $\log_2 0.25 = -2$)

We will use Information Gain to decide which attribute to make root.

$$H(\text{root}) = H(4+, 4-) = H(0.5, 0.5) = -0.5 \log_2 0.5 - 0.5 \log_2 0.5 = 1$$

Splitting on Snowstorm produces subtrees (2+, 2-) and (2+, 2-) for the two values.

$$H(2+, 2-) = 1 \text{ as before.}$$

$$IG(\text{Snowstorm}) = 1 - (4/8 * 1 + 4/8 * 1) = 0$$

Splitting on Holiday produces subtrees (3+, 1-) and (1+, 3-) for the two values.

$$H(3+, 1-) = H(0.75, 0.25) = -0.75 \log_2 0.75 - 0.25 \log_2 0.25 \approx 0.8$$

$$H(1+, 3-) = H(0.25, 0.75) = -0.25 \log_2 0.25 - 0.75 \log_2 0.75 \approx 0.8$$

$$IG(\text{Holiday}) = 1 - (4/8 * 0.8 + 4/8 * 0.8) = 0.2$$

So $IG(\text{Holiday}) > IG(\text{Snowstorm})$, therefore we should make Holiday the root.

6 Hidden Markov Models [14 points]

6.1 Posterior decoding and Viterbi algorithm

Consider a Hidden Markov model with states $Y \in \{S1, S2, S3\}$ and observations $X \in \{1, 2, 3\}$. The starting probabilities, transition probabilities and emission probabilities are specified by Table 2. In the emission probabilities, the second subscript indexes the observations.

$\pi_1 = 0.90$	$a_{11} = 0.50, a_{12} = 0.25, a_{13} = 0.25$	$b_{11} = 0.475, b_{12} = 0.475, b_{13} = 0.050$
$\pi_2 = 0.05$	$a_{21} = 0.25, a_{22} = 0.50, a_{23} = 0.25$	$b_{21} = 0.050, b_{22} = 0.475, b_{23} = 0.475$
$\pi_3 = 0.05$	$a_{31} = 0.05, a_{32} = 0.05, a_{33} = 0.90$	$b_{31} = 0.475, b_{32} = 0.050, b_{33} = 0.475$

Table 2: HMM parameters

Suppose we observe the sequence "1223313". Table 3 shows the Viterbi decoding matrix and the posterior decoding matrix for this sequence. Both matrices are in logarithm form, i.e, the Viterbi decoding matrix stores $\log(V_t^k)$ and the posterior decoding matrix stores $\log(p(y_t^k = 1|x))$ instead of V_t^k and $p(y_t^k = 1|x)$ respectively. Here the subscript t indexes observations and k indexes states.

(a) Viterbi decoding matrix				(b) Posterior decoding matrix			
Obs.	S1	S2	S3	Obs.	S1	S2	S3
1	-0.85	-5.99	-3.74	1	-0.02	-5.13	-4.63
2	-2.29	-2.98	-5.23	2	-0.45	-1.05	-4.20
2	-3.72	-4.42	-6.67	2	-0.80	-0.75	-2.60
3	-7.41	-5.86	-5.86	3	-3.34	-0.94	-0.56
3	-10.24	-7.29	-6.70	3	-4.06	-1.57	-0.26
1	-9.42	-10.98	-7.56	1	-2.28	-3.70	-0.14
3	-13.11	-11.30	-8.40	3	-4.18	-2.23	-0.13

Table 3: Viterbi and Posterior decoding matrices in log form

- [4 points] From the table, what is the most likely set of states according to Viterbi decoding? You may use $\log(0.9) \approx -0.1$, $\log(0.5) \approx -0.7$, $\log(0.475) \approx -0.75$, $\log(0.25) \approx -1.4$ and $\log(0.05) \approx -3.0$. Also write the equation used to compute a single entry in the Viterbi decoding matrix. (You do not need to show your work)

State sequence = "S1" "S1" "S1" "S3" "S3" "S3" "S3"

Equation:

$$\log(V_t^k) = \log(b_{x_t, k}) + \max_j \left(\log(a_{jk}) + \log(V_{t-1}^j) \right) \quad (6)$$

Using this equation means that the Viterbi decoding chooses state S3 over S2 at step 4, even though the Viterbi decoding probabilities are the same due to the differing transition probabilities.

- [4 points] From the table, what is the most likely set of states according to posterior decoding? Explain briefly.

State sequence: "S1" "S1" "S2" "S3" "S3" "S3" "S3"

This can be obtained by just reading off the maximum for each row of the table since we are only computing $\max_k p(y_t = k|x)$.

- **[2 points]** Are the two decodings identical? Explain why or why not.

The decodings are not identical. This happens because the Viterbi decoding is interested in the posterior probability of the entire state sequence while the posterior decoding only takes into account the posterior probability of states at a single time.

6.2 Learning in HMMs

Suppose you were trying to learn the parameters of the HMM in the previous question in a supervised setting, i.e, you are given an observation sequence $x = x_1, \dots, x_N$ and the true state path $y = y_1, \dots, y_N$ which generated the sequence.

1. **[2 points]** If you had few such training sequences, what problem might you encounter in learning the parameters of the HMM?

Some emissions and transitions may not be seen and as a result their probabilities may get set to zero.

2. **[2 points]** How can you correct this problem (if you cannot get more training sequences)? Explain briefly.

This problem can be corrected by adding pseudocounts, i.e, by assuming all transitions and emissions to have taken place a small number of times.

(Note: Cross-validation is not an acceptable answer, the problem here is not one of model selection. Also, using cross-validation reduces the amount of data available for learning even further)