Name: Brian Thibodeau

|  |  |  |
| --- | --- | --- |
| Problem | Points | Score |
| 1(a) | 30 |  |
| 1(b) | 20 |  |
| 2(a) | 10 |  |
| 2(b) | 20 |  |
| 2(c) | 10 |  |
| 2(d) | 10 |  |
| Total | 100 |  |

Notes:

1. The exam is closed books and notes except for one double-sided sheet of notes.
2. Please indicate clearly your answer to the problem.
3. If I can’t read or follow your solution, it is wrong and no partial credit will be awarded.

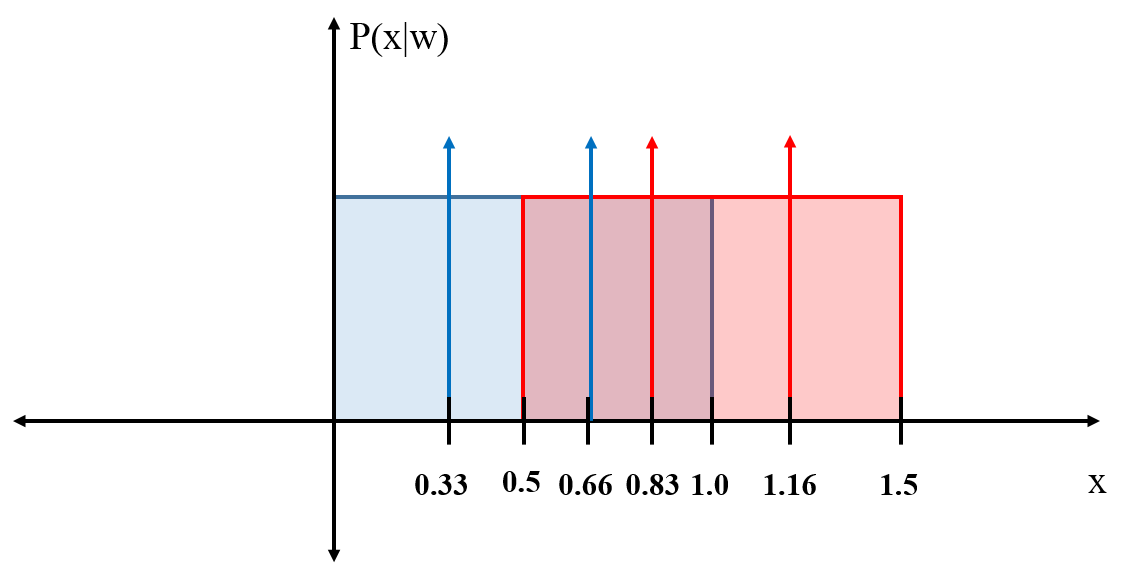
**Problem No. 1: Consider two probability distributions defined by:**

** and **

**and assume equal priors.**

**(a) Draw two points at random from each class. Design a nearest-neighbor classifier based on these points. Compute the probability of error.**

*Solution* – Figure 1depicts the two class conditional densities under consideration. The test points for each class are represented with vertical arrows. The test points for class w1 are 0.33 and 0.66 while the test points for class w2 are 0.83 and 1.16.



**Figure 1. Class conditional densities with 4 test points used for nearest neighbor classification**

The nearest neighbor classification rule says to assign the class of the test point to the class of the neighbor nearest to the test point. Using the nearest neighbor rule, Table I summarizes the class assignments for each test point and the probability of error.

**Table I. Nearest Neighbor Classification with 4 test points**

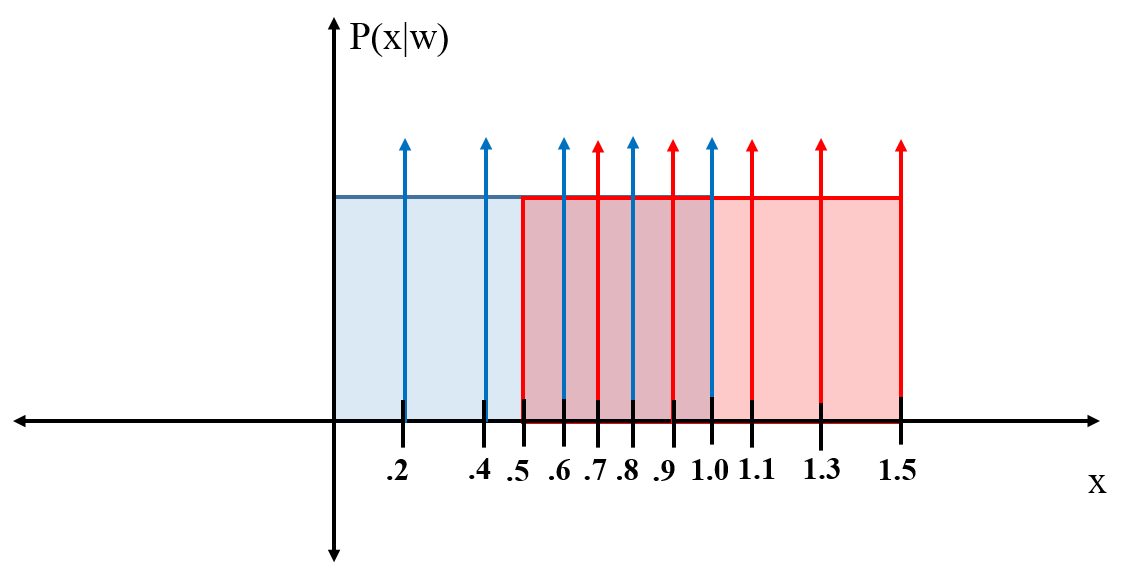
|  |  |  |
| --- | --- | --- |
| **Test Point** | **Nearest Neighbor** | **Class Assignment** |
| 0.33 | 0.66 | w1 |
| 0.66 | 0.83 | w2 |
| 0.83 | 0.66 | w1 |
| 1.16 | 0.83 | w2 |
| **P(error) : 0.5** | | |

**(b) Explain what happens as you allow the number of points drawn to increase. Show that your result in (a) converges to the correct result.**

*Solution* – For a nearest neighbor classifier, it can be shown that the error is never worse than twice the Bayes rate. As the number of test points drawn from the class conditional densities increases, the nearest neighbor classifier error approaches the Bayes rate. Equation (1) illustrates the limits of the nearest neighbor error bounds.

Note that is the Bayes error rate, *P* is the nearest neighbor error rate, and *c* is the number of classes. Thus for our two class example, *c =* 2. is computed by finding the area of overlap between the two densities in Figure 1. The resulting Bayes error is and the bounds on the nearest neighbor classifier are:

We then demonstrate that equation (1) hold by choosing 5 test points from each class and repeating the nearest neighbor classification done in part (a).



**Figure 2. Class conditional densities with 5 test points chosen from each class.**

Figure 2 illustrates the 5 test points that are drawn from each class. The test points for class w1 are 0.2, 0.4, 0.6, 0.8, and 1.0 while the points from class w2 are 0.7, 0.9, 1.1, 1.3 and 1.6. Like part (a), Table II summarizes the results of the nearest neighbor classification with the additional test points.

**Table I. Nearest Neighbor Classification with 10 test points**

|  |  |  |
| --- | --- | --- |
| **Test Point** | **Nearest Neighbor** | **Class Assignment** |
| 0.2 | 0.4 | w1 |
| 0.4 | 0.2/0.6 | w1 |
| 0.6 | 0.7 | w2 |
| 0.8 | 0.7/0.9 | w2 |
| 1.0 | 0.9/1.1 | w2 |
| 0.7 | 0.6/0.8 | w1 |
| 0.9 | 0.8/1.0 | w1 |
| 1.1 | 1.0 | w1 |
| 1.3 | 1.1/1.5 | w2 |
| 1.5 | 1.3 | w2 |
| **P(error) : 0.60** | | |

Table II shows that increasing the total number of test points from 4 to 10 resulted in an increased error rate from 0.5 to 0.6, but that the limits of equation (1) still hold.

**Problem No. 2: Consider the following models for a system that outputs sequences of the characters “$” and “%”. For these models, you must start in state 1 and end in state 2.**

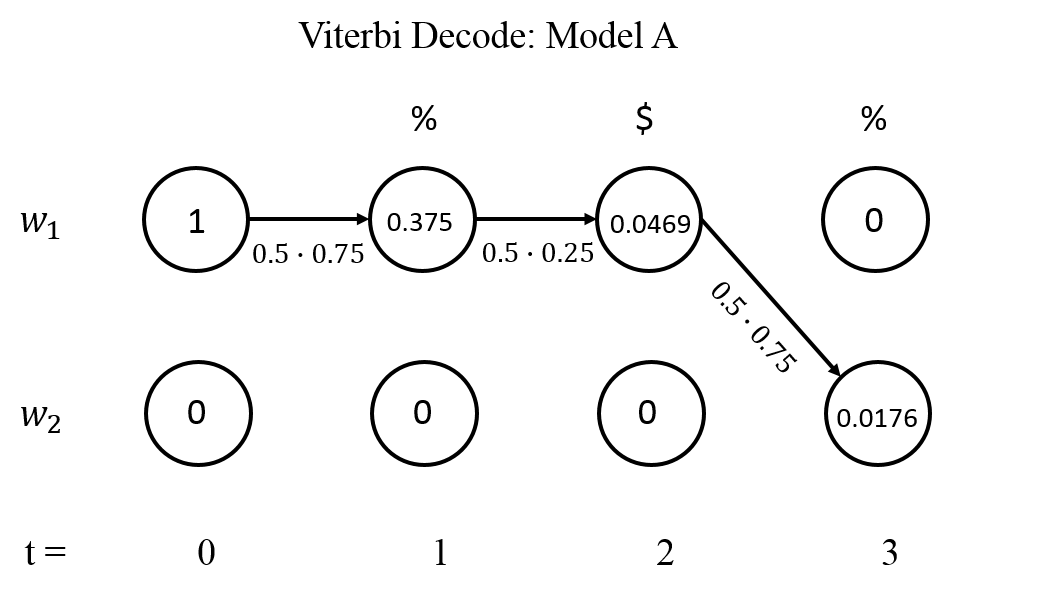
1. **Compute the probability that model A produced the sequence “%$%”.**



*Solution* – The probability of the sequence “%$%” given model A, P(“%$%”|A) is computed by taking the product of the corresponding transition probabilities and the output probabilities at each time step. Considering the state diagram for Model A, we first assume that the state transition matrix *A* with elements is,

And the emission matrix *B* with elements is,

Then considering that we must start in state w1 and end in state w2, there is only 1 possible state sequence that can generate the output sequence “%$%.”. This is illustrated by the trellis diagram for Model A.



**Figure 1. Trellis diagram for decoding the output sequence “%$%” in Model A**

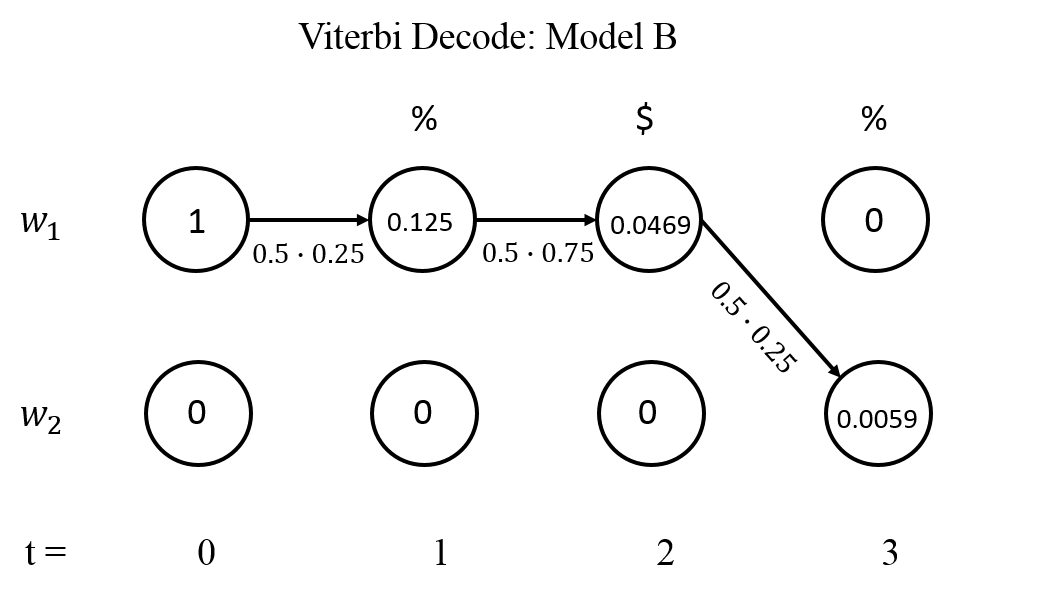
At t=0 the HMM is initialized to state w1 and at t=1 the first symbol (%) is output with probability,

Then at t=2 the symbol is output is “$.” However to end in state w2, the HMM must stay in state w1 or else it would be stuck in state w2. The probability for outputting “$” in state w1 is computed similarly to the above computation but we must also now consider the previous time steps probability.

Repeating the above procedure for t=3 , the resulting probability of outputting the sequence “%$%” is **P(“%$%”|A)=0.0176.**

1. **Which model most likely produced the sequence “%$%”. Explain.**

*Solution* – The model that maximizes the probability of outputting “%$%” is determined by comparing P(“%$%”|A) and P(“%$%”|B) and determining which is larger. Applying the same procedure for Model B that was used in part (a) we can determine P(“%$%”|B).



**Figure 2. Trellis diagram for decoding the output sequence “%$%” in Model B.**

From the trellis diagram for model B in Figure 2, its determined that **P(“%$%”|B)=0.0059**. Thus comparing P(“%$%”|A) and P(“%$%”|B) we find that model most likely produced the sequence “%$%”.

1. **Which state sequence most likely produced the sequence “%$%”. What was the probability of that state sequence?**

*Solution* – It part (a) it was explained that there is only one possible state sequence that can result in the output “%$%.” The reason was because there is only two states and the constraint that the HMM must begin in state w1 and end in state w2. Since there is only 1 possible sequence of states, the probability is 1.0. However if there were *r* different state sequences then we choose the state sequence that maximizes P(“%$%”|), where *r* indexes a particular sequence and *T* is the number of hidden states.

1. **Give at least two reasons why the probabilities in (a) and (c) differ.**

*Solution* – In part (a) we were interested in the probability that a particular sequence of *visible* states were generated by models A and B. In part (c) we wanted to know the most likely sequence of *hidden* states that led to those *visible* states. Part (a) is an **Evaluation** problem whereas part (c) is a **Decoding** problem.