Artificial Intelligence Lab Report 6

1st Dipean Dasgupta
202151188
BTech CSE
IIIT,Vadodara

2nd Shobhit Gupta
202151149
BTech CSE
IIIT,Vadodara

3rd Rahul Rathore
202151126
BTech CSE
IIIT,Vadodara

4th Rohan Deshpande
202151133
BTech CSE
IIIT,Vadodara

***Abstract*—To implement Expectation Maximization routine for learning parameters of a Hidden Markov Model, to be able to use the EM framework for deriving algorithms for problems with hidden or partial information.**

I. Problem Statement

1) Read through the reference carefully. Implement routines for learning the parameters of HMM given in section 7. In section 8, "A not-so-simple example", an interesting exercise is carried out. Perform a similar experiment on "War and Peace" by Leo Tolstoy.
2) Ten bent (biased) coins are placed in a box with unknown bias values. A coin is randomly picked from the box and tossed 100 times. A file containing results of five hundred such instances is presented in tabular form with 1 indicating head and 0 indicating tail. Find out the unknown bias values. (2020 ten bent coins.csv) To help you, a sample code for two bent coin problem along with data is made available in the work folder: two bent coins.csv and embentcoinsol.m
3) A point set with real values is given in 2020 em clustering.csv. Considering that there are two clusters, use EM to group together points belonging to the same cluster. Try and argue that k-means is an EM algorithm.

II. Explanation

A. Problem 1

A Markov model serves as a tool for understanding the dynamics of systems that exhibit random transitions, adhering to the Markov property. This property signifies that the future state solely depends on the present state and is unaffected by the past. Hidden Markov Models (HMMs) are particularly useful for modeling systems where transitions between states occur over time and are indirectly observed through outputs. The objective of an HMM is to infer the underlying sequence of states given a series of observed outputs. Common algorithms employed for this purpose include the Viterbi Algorithm, Forward-Backward Algorithm, and Baum-Welch Algorithm.

In this context, referencing "A Revealing Introduction to Hidden Markov Models," we conducted calculations involving the α-pass, β-pass, and di-gammas to update parameters such as the state transition probability matrix (A), observation probability matrix (B), and initial state distribution (π) using Leo Tolstoy's "War and Peace" text. These updates were based on the observed sequence of characters from the text, which were converted to lowercase and stripped of punctuation. Initializing each element of π and A to around 1/2, we iteratively refined these parameters until convergence. The initial values were as follows:

$$A : \begin{bmatrix} 0.47468 & 0.52532 \\ 0.51656 & 0.48344 \end{bmatrix}$$

$$\pi : \begin{bmatrix} 0.51316 & 0.48684 \end{bmatrix}$$

After the initial iteration, the log likelihood of the observed sequence given the model (λ) was calculated to be 142533.41283009356, which improved to 138404.497179602 after 100 iterations.

B. Problem 2

1) Expectation Maximization Algorithm:: The Expectation Maximization (EM) algorithm is a powerful iterative optimization technique used for estimating latent parameters (denoted as Θ) given observed data (denoted as U).

In this problem scenario, we were tasked with determining the biases of ten biased coins using the EM algorithm. The process involves an Expectation step where initial biases are hypothesized, followed by an evaluation of the likelihood of each coin generating the observed outcomes. This step is then followed by a Maximization step where the inferred biases are updated. These Expectation and Maximization steps are iteratively repeated until convergence is achieved.

C. Problem 3

Similarly, in the third problem, we applied the EM algorithm to cluster a set of points into two distinct clusters. Beginning with initial guesses for the cluster parameters, we iteratively refined these parameters to improve the fit to the data. The Expectation step involves calculating the likelihood of each point belonging to each cluster, while the Maximization step updates the cluster parameters based on these probabilities. This iterative process continues until convergence is reached, and points are assigned to the cluster with the highest probability.

III. RESULTS

1) First Task Outcome:
   Upon completing 100 iterations for the first task, all alphabets were effectively categorized into two distinct

groups: vowels and consonants. Referencing Figure 1, it is evident that vowels were assigned to the first group, while consonants belonged to the second group.

```
Observation probability matrix:
[[1.40667453e-001 7.98836398e-050 6.71850005e-008 1.67031156e-027
  2.05350308e-001 6.25839882e-060 3.41810088e-004 3.78152237e-010
  1.19814376e-001 9.72334826e-077 1.80675392e-016 5.24143415e-003
  7.80453181e-058 1.46029671e-022 1.20855544e-001 1.00706383e-003
  8.31306205e-065 1.15658398e-035 1.66075398e-022 8.23826393e-010
  3.82508778e-002 2.70512825e-068 3.03172534e-069 5.57386523e-019
  2.38157375e-009 2.55777401e-104 3.68471062e-001]
 [2.50030317e-004 2.29742335e-002 4.07142703e-002 7.12401015e-002
  8.42416514e-027 2.98864812e-002 3.16630619e-002 9.90888677e-002
  1.08751246e-028 1.71807312e-003 1.21064222e-002 6.19749371e-002
  3.90761746e-002 1.19665790e-001 1.81584034e-026 3.59537421e-002
  1.39843161e-003 9.39346487e-002 1.03563849e-001 1.37086252e-001
  4.10741541e-003 2.29342783e-002 3.52005213e-002 2.51717689e-003
  3.18842382e-002 1.03883491e-003 2.21676562e-005]]
```

Fig. 1. Observation Probability Matrix

2) Second Task Outcome: Initial bias values for all ten coins are depicted in Figure 2, while Figure 3 showcases the final converged bias values after multiple iterations. The last row in Figure 3 represents the conclusive bias values for each coin.

```
th =

    0.8147    0.9058    0.1270    0.9134    0.6324    0.0975    0.2785    0.5469    0.9575    0.9649
    0.7915    0.8920    0.1338    0.8972    0.6559    0.0473    0.3094    0.5007    0.9361    0.9427
    0.7790    0.8814    0.1417    0.8855    0.6571    0.0267    0.3146    0.4890    0.9188    0.9258
    0.7699    0.8724    0.1378    0.8768    0.6523    0.0184    0.3163    0.4848    0.9088    0.9160
    0.7622    0.8640    0.1331    0.8695    0.6463    0.0139    0.3151    0.4820    0.9043    0.9116
    0.7555    0.8548    0.1302    0.8626    0.6403    0.0122    0.3128    0.4791    0.9027    0.9101
    0.7490    0.8440    0.1286    0.8556    0.6343    0.0116    0.3103    0.4759    0.9024    0.9096
    0.7421    0.8312    0.1276    0.8484    0.6282    0.0114    0.3078    0.4725    0.9023    0.9090
    0.7344    0.8170    0.1270    0.8416    0.6217    0.0114    0.3055    0.4690    0.9020    0.9079
```

Fig. 2. Initial iterations of biases

```
0.5109    0.6243    0.0965    0.7187    0.4256    0.0102    0.1954    0.3191    0.7976    0.8994
0.5098    0.6226    0.0964    0.7176    0.4248    0.0102    0.1950    0.3184    0.7969    0.8993
0.5087    0.6210    0.0964    0.7165    0.4240    0.0102    0.1947    0.3178    0.7962    0.8991
0.5077    0.6193    0.0963    0.7154    0.4232    0.0102    0.1943    0.3171    0.7956    0.8990
0.5066    0.6178    0.0962    0.7144    0.4224    0.0101    0.1939    0.3164    0.7950    0.8989
0.5056    0.6162    0.0961    0.7135    0.4215    0.0101    0.1936    0.3157    0.7945    0.8988
0.5047    0.6147    0.0960    0.7126    0.4207    0.0101    0.1932    0.3151    0.7941    0.8988
0.5037    0.6132    0.0959    0.7117    0.4199    0.0101    0.1928    0.3144    0.7936    0.8987
```

Fig. 3. Final converged values of biases

3) Third Task Outcome: Upon completion of the iterations, two distinct clusters were formed, enabling accurate prediction of each point's cluster assignment. Additionally, it's noteworthy to mention that K-means algorithm shares similarities with the EM algorithm. In K-means, similar to EM, initial parameters are guessed, and data points are assigned to the nearest cluster center (E step), followed by an update of the cluster centers (M step). Therefore, K-means can be viewed as an EM algorithm with latent variables representing the cluster assignments of points.

IV. References

The following references were consulted for this assignment:

1) "A Revealing Introduction to Hidden Markov Models" by Mark Stamp, 2018
2) "What is the expectation maximization algorithm?" by Chuong B Do and Serafim Batzoglou, published in Nature Biotechnology, Volume 26, Number 8, August 2008