HACETTEPE UNIVERSITY

BBM 411: FUNDAMENTALS OF BIOINFORMATICS - 2022 FALL

Assignment 2

TUNCA DOĞAN

Student name:
Ataberk ASAR

Student Number:
2210356135

1 Question 1

(a) 2 axes represent torsional angles of amino acids contained in a protein molecule. They are important since the used the describe the 3d structure of the proteins, and in Ramachandran plot, they show what Ψ and Φ angles are possible for said protein structure.

(b) Protein structures are made out of amino acids forming peptide bonds. The secondary structure is determined by the dihedral angles of the peptide bonds. The tertiary structure is determined by the folding of protein chains in 3d space.

(c) The term is used to describe similarities in protein sequences. They are important to show shared biological functions and evolutionary connections between protein families.

(d) DNA sequencing is such a procedure, that is used determine order of nucleotides in a DNA molecule. Process fragments DNA molecules into smaller pieces and sequences each fragment separately. It is generally less expensive since it requires fewer steps than protein sequencing.

2 Question 2

Note that python used to obtain these results.
Hits:

```
          0           1           2           3           4
seq:    MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDI
alpha:  *               *     ************     *   ***         ***
beta:   *             ******   ***                   *         ***
seq:    EQWFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQ
alpha:  **              ** **    **          *
beta:   ***                                                    **
seq:    KTYQGSYGFRLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDST
alpha:            *                     **********     ****
beta:    *        *             ****       *******   *******
seq:    PPPGTRVRAMAIYKQSQHMTEVVRRCPHHERCSDSDGLAPPQHLIRVEGN
alpha:        ******* *   ***                           *********
beta:              * *   *******                         ***       *
seq:    LRVEYLDDRNTFRHSVVVPYEPPEVGSDCTTIHYNYMCNSSCMGGMNRRP
alpha:  *                 *                                      *
beta:   ***             ***   *       ********                  **
seq:    ILTIITLEDSSGNLLGRNSFEVRVCACPGRDRRTEEENLRKKGEPHHELP
alpha:  ****          *    *****             *******
beta:   *****               ****           ****
seq:    PGSTKRALPNNTSSSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALEL
alpha:      ***         ****       ***       ***************
beta:                           *******      ******* ****
seq:    KDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKKLMFKTEGPDSD
alpha:  ****              ***         ******
beta:                                  ***
```

After extending regions:

```
          0          1          2          3          4
seq:    MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDI
alpha:  *****      ****************************************
beta:   *****      ****************************************
turn:     ** *     *                *      * *    **      **
seq:    EQWFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQ
alpha:  ******       ************************************
beta:   ******                                   *********
turn:         * *                    *         *   * **   *
seq:    KTYQGSYGFRLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDST
alpha:  ****     ******             ********************
beta:   **************************************************
turn:     ****         *         *   *                    *
seq:    PPPGTRVRAMAIYKQSQHMTEVVRRCPHHERCSDSDGLAPPQHLIRVEGN
alpha:        ******************                 ************
beta:         ************************       ************
turn:   **                          *      ****
seq:    LRVEYLDDRNTFRHSVVVPYEPPEVGSDCTTIHYNYMCNSSCMGGMNRRP
alpha:  *************************        ******************
beta:   ******      ************************************
turn:         ***              *    **      *     *** *   * *
seq:    ILTIITLEDSSGNLLGRNSFEVRVCACPGRDRRTEEENLRKKGEPHHELP
alpha:  ********     **************       *****************
beta:   ********       *************     ********
turn:            ***     *          * *** *          **   *     **
seq:    PGSTKRALPNNTSSSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALEL
alpha:       *******     ***********************************
beta:                          ****************************
turn:   ***     ****** * **      *
seq:    KDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKKLMFKTEGPDSD
alpha:  ********       ************************
beta:   ********                          *******
turn:           ****    **     ** * * *           **
```

After overlap treatments:

```
          0          1          2          3          4
seq:    MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDI
alpha:             ************** *****          *****   ***
beta:
turn:     ** *     *                 *      * *    **      **
seq:    EQWFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQ
alpha:  ******       ***************** ********           ***
beta:
turn:         * *                         *          *   * **   *
seq:    KTYQGSYGFRLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDST
alpha:  **
beta:         ******** ********     ********************
turn:     ****         *         *   *                     *
seq:    PPPGTRVRAMAIYKQSQHMTEVVRRCPHHERCSDSDGLAPPQHLIRVEGN
alpha:
beta:     ********************** *****        ************
turn:   **                              *      ****
seq:    LRVEYLDDRNTFRHSVVVPYEPPEVGSDCTTIHYNYMCNSSCMGGMNRRP
alpha:
beta:   *****       **********       *****                 ***
turn:         ***               *    **      *     *** *   * *
seq:    ILTIITLEDSSGNLLGRNSFEVRVCACPGRDRRTEEENLRKKGEPHHELP
alpha:
beta:   ********
turn:           ***     *          * *** *          **   *     **
seq:    PGSTKRALPNNTSSSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALEL
alpha:
beta:
turn:   ***     ****** * **       *
seq:    KDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKKLMFKTEGPDSD
alpha:
beta:
turn:           ****    **     ** * * *            **
```

Prediction:

```
___TT_T____THHHHHHHHHHHHHHTHHHHHT_T___TTHHHHHTTHHH
HHHHHHT_T___HHHHHHHHHHHHHHHHTHHHHHHHHT__T_TT__THHH
HHTTTTEEEEEEEETEEEEEEEET__TEEEEEEEEEEEEEEEEEEET__
TTEEEEEEEEEEEEEEEEEEEEEETEEEEETTTT___EEEEEEEEEEE
EEEEETTT___EEEEEEEEET___TTEEEEET____TTT_T__T_TEEE
EEEEEEETTT____T_________T_TTT_T________TT__T____TT
TTT____TTTTTT_T_TT_____T_____________________
_______TTTT___TT____TT_T_T_T_________TT___
```

True Labels:

```
___HHHH_TTT________HHHHH___EEEHHHEEEHHH__HHHH__HHH
HHHHHH_____________________________________________
_____TTTT_EEE_____EEETTTEEEETTTTEEEE_____EEEEEE_EE
E_____EEEEEEEEEEHHH_________HHHHEEE___EEE____EEEEEE
____EEEE_TTT__EEEEEE_____TTTEEEEEEEEE___HHHTTTTTT_
_EEEEEEEE_EEE_EEEEEEEEEE___HHHHHHHHHHHHH_________
_____________________HHH__EEEEEEEEHHHHHHHHHHHHHHH
HHHHH_____________________HHHHHH_____________
```

Confusion Matrix:

```
|20   7  13|
| 3  53  14|
| 0  13   6|
```

	precision	recall	f1-score
H	0.870	0.500	0.635
E	0.726	0.757	0.741
T	0.182	0.316	0.231

accuracy : 0.612

3 Question 3

(a) Transition Matrix:

```
[0.905212   0.008710        0.086078]
[0.036419   0.810262        0.153319]
[0.138269   0.195079        0.666652]
```

Emission Matrix:
Note that transpose of emission matrix given in order to fit it into the page.

```
[0.086532   0.046444        0.047183]
[0.047693   0.036164        0.036065]
[0.025190   0.019383        0.050506]
[0.039027   0.024453        0.062136]
[0.007488   0.010399        0.006947]
[0.072084   0.285287        0.051891]
[0.037294   0.021803        0.026560]
[0.050490   0.035652        0.124552]
[0.215385   0.017750        0.019925]
[0.046701   0.075349        0.019691]
[0.094096   0.079416        0.042847]
[0.051509   0.035461        0.047318]
[0.019591   0.015803        0.010401]
[0.031447   0.043282        0.022296]
[0.019336   0.015731        0.048101]
[0.037402   0.037108        0.155632]
[0.032365   0.048548        0.174939]
[0.011798   0.013276        0.007234]
[0.048256   0.101823        0.025771]
[0.026316   0.036867        0.020006]
```

Pi:

```
[0.47154485, 0.27856788, 0.24988727]
```

Forward algorithm used to compute the probability of the sequence being emitted from this HMM.
Prob: 7.56e-225
Viterbi algorithm used to find most probable path, ie. predictions.

Prediction:

```
HHEEEHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHHEEEEHHEHHHHH
HTTTHHHEEHHETETTTTTTEEHEEEEHHTEETEEEEEETEEEHTTTTEE
EEETETETEEHHHHHEHEETTETEEEEHHHTHHTEETTTTHHEEETETET
EHHHHHHHHHTTHHEEETT
```

True Labels:

```
HHHHTTTHHHHHEEEHHHEEEHHHHHHHHHHHHHHHHHTTTTEEEEEETTT
EEEETTTTEEEEEEEEEEEEEEEEEEEEEEEHHHHHHHHEEEEEEEEEEE
EEEETTTEEEEEETTTEEEEEEEEEHHHTTTTTTEEEEEEEEEEEEEEE
EEEEEEHHHHHHHHHHHHH
```

(b) Confusion Matrix:

```
|33  13   5|
|28  36  28|
|15   8   3|
```

	precision	recall	f1-score
H	0.434	0.647	0.520
E	0.632	0.391	0.483
T	0.083	0.115	0.097

accuracy : 0.426

(c) As one can see Chou-Fasman method outperforms our HMM. One reason is that HMM model tends to predict alpha helix much more than other secondary structures, thus prediction has large alpha helix regions. Performance can be improved by using better datasets suitable for this task.