ABIN | End Semester | Monsoon 2023
Answer any FOUR questions.
Time: 90 minutes

Q1:

a: Name an assay that is used to track transcription factor activity. (1)

Sol: ChIP based assay or ELISA

b: Provide a formal definition to the Motif Finding problem. (3)

Sol:

Note: Full marks given only if formal definition with Input and output written

The Motif Finding Problem: Formulation

- Goal: Given a set of DNA sequences, find a set of *l*-mers, one from each sequence, that maximizes the consensus score
- Input: A ***t*** x ***n*** matrix of ***DNA***, and *l*, the length of the pattern to find
- Output: An array of ***t*** starting positions **s** = (s_1, s_2, … s_t) maximizing *Score*(**s**,***DNA***)

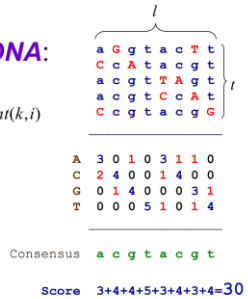
c: Describe a suitable motif scoring strategy, and provide an example. (3)

Sol-

Scoring Motifs

- Given **s** = (s_1, … s_t) and ***DNA***:

$$Score(\mathbf{s},\mathit{DNA}) = \sum_{i=1}^{l} \max_{k \in \{A,T,C,G\}} count(k,i)$$



(*Any valid strategy is acceptable*)

d: Establish the relationship between Motif Finding and Median String Search problems. (3)

Sol:

Motif Finding Problem == Median String Problem

- The *Motif Finding* is a maximization problem while *Median String* is a minimization problem
- However, the *Motif Finding* problem and *Median String* problem are computationally equivalent
- Need to show that minimizing *TotalDistance* is equivalent to maximizing *Score*

Q2:

a. What is memoization, in the context of dynamic programming? Explain with an example. (2)

Solution –

Memoization is a top-down approach for improving the performance of recursive algorithms, where we cache the results of function calls and return the cached result if the function is called again with the same inputs. The purpose of caching is to improve the performance of our programs and keep data accessible that can be used later. This removes the extra effort to calculate again and again for the same problem. And we already know that if the same problem occurs again and again, then that problem is recursive in nature. It is well-suited for problems that have a relatively small set of inputs.

Example – Fibonacci

b. State two application for each of the global and local alignments. (2)

Solution –

Local Alignment – BLAST, Pair-wise alignment, Needleman-Wunsch.

Global Alignment – Smith-Watermann, SNP identification, Pair-wise alignment.

c. What are the algorithms (do not need to write the algorithm, just provide a sketch of the algorithmic strategy) and their complexities associated with finding hamming Hamming distance and Edit distance between two strings. Clearly explain.

Solution –

- Any algorithm using Hamming Distance with proper explanation of the mathematical notations along with Time Complexity.

- Any algorithm using Edit Distance with proper explanation of the mathematical notations along with Time Complexity.

Note : Full marks awarded only if the answer covers all the required components i.e. Clear explanation of the algorithm and Time complexity.

d. For the below case of local alignment, provide the alignment of the two strings with gaps where necessary.
(2)

Solution -

```
G T T _ A C
|  |  \   |  |
G T T G A C
```

Q3:

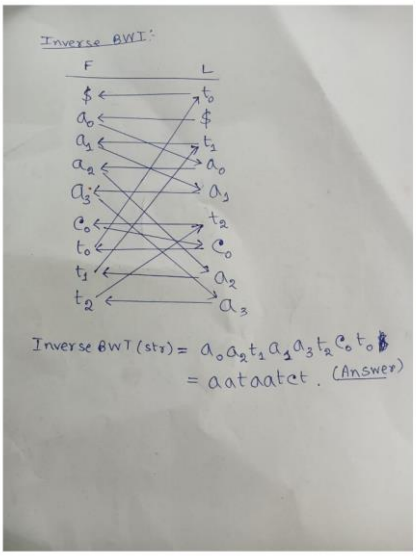
a: Burrows' Wheeler Transformation is lossless and can accelerate search. Explain with an example sequence of nucleotides. (5)

str = **aataatct**

	rotation	sorted rotation
0	aataatct$	$aataatct
1	$aataatct	aataatct$
2	t$aataatc	aatct$aat
3	ct$aataat	ataatct$a
4	tct$aataa	atct$aata
5	atct$aata	ct$aataat
6	aatct$aat	t$aataatc
7	taatct$aa	taatct$aa
8	ataatct$a	tct$aataa

BWT(str) = **t$taatcaa**

Inverse BWT:



Lossless: The original sequence can be recovered from the encoded sequence

Accelerate search: by indexing/compression

b: Write the steps of the GSEA algorithm with adequate mathematical notations. (3)

Note: *If mathematical notations are not written, then only partial marks are awarded*

Steps are:
1: Calculation of an Enrichment Score
2: Estimation of Significance Level of ES
3: Adjustment for Multiple Hypothesis Testing

- Enrichment score ES=0
- Screen from top to bottom
  - If hit a gene in the pathway
    - $ES = ES + S_{hit}$
  - If hit a gene not in the pathway
    - $ES = ES - S_{unhit}$
  - $S_{hit} = \frac{|Correlation_g|^a}{\sum_{g \in pathway} |Correlation_g|^a}$
  - $S_{unhit} = \frac{1}{Number\ of\ gene\ not\ in\ the\ pathway}$

Sol-

c: What is the procedure for conducting a randomization test to determine the statistical significance of the enrichment gene set enrichment score? (2)

Estimating Significance. We assess the significance of an observed ES by comparing it with the set of scores ES_{NULL} computed with randomly assigned phenotypes.

1. Randomly assign the original phenotype labels to samples, reorder genes, and re-compute $ES(S)$.
2. Repeat step 1 for 1,000 permutations, and create a histogram of the corresponding enrichment scores ES_{NULL}.
3. Estimate nominal P value for S from ES_{NULL} by using the positive or negative portion of the distribution corresponding to the sign of the observed $ES(S)$.

Sol-

Q4:

a: For ONE of the below problem statements, define a Hidden Markov Model by clearly defining the starting probability, hidden and observed states, the transition matrix, and emission probabilities. Fictitious probability values are acceptable. (5)

Option 1: Gene finding Option

2: AMR gene finding Option

3: Fining viral motifs/proteins potent for host interaction

This is an open question with choice, marks will be given if everything is defined as mentioned in question.

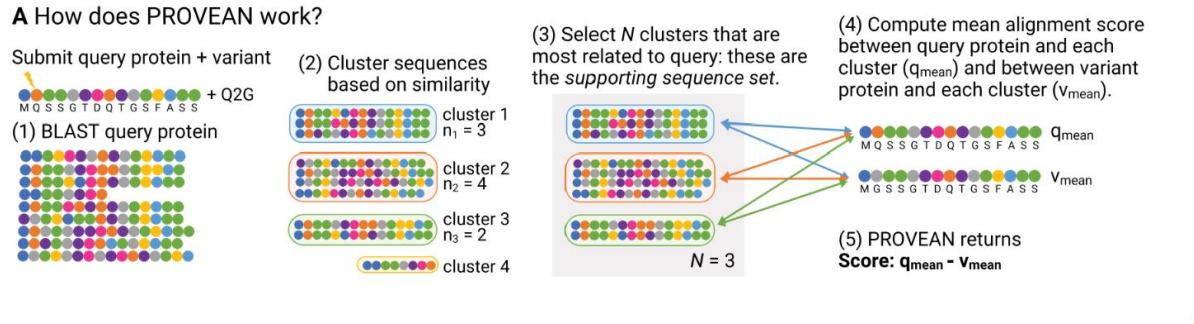
b: Clearly articulate how the above HMM will be trained and used for real life applications

Sol-

HMMs and their variants have been used in gene prediction, base-calling, modeling DNA sequencing errors, protein secondary structure prediction, ncRNA identification, annotation.

Q5:

a: Write the steps of the PROVEAN algorithm with adequate details



a. Write the steps of the PROVEAN algorithm with adequate details. (4)

b. You want to design a motif classifier. After dividing your dataset into training and test sets, you observe the classes are balanced. You also observe that the labels in the training and test set follow a pattern where positive classes is followed by negative classes. A friend advises you "**Shuffle both the training and test sets before the training process. Otherwise, the approach will be incorrect.**"

Justify if your friend is right for each of the below cases.

- You plan to use mini-batch gradient descent (1)
- You plan to use gradient descent (1)

Answer:

Test set: The test set does not need any shuffle as once the decision boundary is fixed after training, the prediction does not depend on order of test data.

Train data: In gradient descent the order of input does not matter as all the sample go in single batch. However in min-batch gradient descent if one batch have sample of one classes only, and the other batch have sample from opposite class, the model will not be trained properly. So in mini-batch gradient descent shuffling is required.

Note: If you don't define the correct choice and justification individually for both train and test, you will be awarded zero marks. No marks for writing yes/No.

c. You have a dataset containing 10000 gene expression profiles, with only 4000 labeled as 1, and the rest labeled as 0. In order to address the class imbalance, you augment the positive examples to 6000 and subsequently divide the data into training, validation, and test sets. The model demonstrates excellent performance in the test set, achieving metrics such as F1 score >95%, precision >95%, recall >95%, and accuracy >95%. Do you think the model can maintain this high performance when deployed in a production environment (assuming no domain shift takes place)? Provide a justification for your answer (2)

Answer:

Data augmentation should be performed only in the train data. But in the above question, data augmentation is performed before splitting of data.

So test data may contain augmented data. So the model may not be properly generalized on unseen data. It might lead to poor performance in production deployment.

Note: No marks for writing yes/no.

d. You are solving a binary image classification problem, and you've designed a model as described below. Do you find any possible issues in the provided code? If there are no errors, simply state that the code functions correctly. However, if you identify any bugs, kindly propose solutions of the bug. (1)

Note:

```
model = tf.keras.Sequential()
model.add(tf.keras.layers.Conv2D(filters = 16,
                                 kernel_size = (5,5),
                                 activation ='relu',
                                 input_shape = (28,28, 1)))
model.add(tf.keras.layers.Flatten())
model.add(tf.keras.layers.Dense(256, activation = "relu"))
model.add(tf.keras.layers.Dense(1, activation = "softmax"))
model.compile(
    loss='categorical_crossentropy',
    optimizer=keras.optimizers.Adam(learning_rate=0.0001),
    metrics=['accuracy', 'Precision', 'Recall']
)
```

Assume all library has been correctly imported and the image size is (28,28,1)

Answer: There can be two possible answer.

Case 1 : For binary classification if output layer has single node, sigmoid activation must be used.

Case2: For binary classification if output layer has softmax activation, there must be two output nodes.

e. You want to design a model for a binary classification task, and you've been provided with the entire dataset features in X and corresponding labels in y. The following code represents your pre-processing steps for classification.

```
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)
X_train, X_test, y_train, y_test = train_test_split(
    X_scaled, y, test_size=0.2, random_state=0)
```

Explain the purpose of the three lines of code provided above. Are there any potential issues in the code? If yes, please describe. (1)

Note: Assume that all libraries have been appropriately imported, and variable names adhere to conventional usage

Answer: The fit_transform should be applied on training data only, not on test data. But in the above question fit_transform has been applied on entire data, before data splitting. So it is incorrect.