22BIO211: Intelligence of Biological Systems - 2

Lab Sheet 2

1. Write a python program to create a directed overlap graph for the given K-mer set .

   {CAT,ATG,TTC,TCC,TGC,GCA,ATT,CCA}

   Also visualize the graph using python libraries.

2. Construct a weighted and directed overlap graph of all 6-mers from "GTACGTACGAT" where edges are overlaps of length >= 4. Visualize the graph using 'Networkx'.

   Steps:

   - Construct all k-mers
   - Represent each k-mer as a node
   - Connect two nodes with an edge if they overlap
   - Assign overlap length as the weight

3. Solve the following Rosalind Challenge : https://rosalind.info/problems/ba3c/ and upload the 'strand' page.

Overlap Graph Problem

Construct the overlap graph of a collection of k-mers.

Given an arbitrary collection of k-mers Patterns, we form a graph having a node for each k-mer in Patterns and connect k-mers Pattern and Pattern' by a directed edge if Suffix(Pattern) is equal to Prefix(Pattern'). The resulting graph is called the overlap graph on these k-mers, denoted Overlap(Patterns).

we use the terms prefix and suffix to refer to the first $k - 1$ nucleotides and last $k - 1$ nucleotides of a k-mer, respectively.

Given: A collection Patterns of k-mers.

Return: The overlap graph Overlap(Patterns), in the form of an adjacency list.

Sample Dataset

ATGCG

GCATG

CATGC

AGGCA

GGCAT

Sample Output

AGGCA -> GGCAT

CATGC -> ATGCG

GCATG -> CATGC

GGCAT -> GCATG

4. A Hamiltonian Path in a graph having N vertices is nothing but a permutation of the vertices of the graph [v_1, v_2, v_3, ...v_{N-1}, v_N] , such that there is an edge between v_i and v_{i+1} where $1 \leq i \leq N-1$. So it can be checked for all permutations of the vertices whether any of them represents a Hamiltonian Path or not.Write a program to check whether any of these permutations gives rise to Hamiltonian path.

Pseudocode

```
function HamiltonianPath(dict graph)
        #create all permutations of the keys of the dictionary
    Allp = get_all_permutation(graph.keys);
    foreach p in Allp
            if(isHamiltonian(p))
             print ("Hamiltonian path found and the path is: " , p)
             return true
    print ("No Hamiltonian path found")
    return false

function isHamiltonian(path)
      # Check if every pair of adjacent vertices in the path is connected
        for i in range(graph.keys-1):
          if path[i+1] not in graph[path[i]]:
            return false
    return true
```

5. Find Hamiltonian paths for the overlap graphs constructed in Q1. Also apply string reconstruction problem to reconstruct the source string(genome) from the k-mers in the Hamiltonian path.
6. Find Hamiltonian paths for the overlap graphs constructed in Q2. Also apply string reconstruction problem to reconstruct the source string(genome) from the k-mers in the Hamiltonian path.