

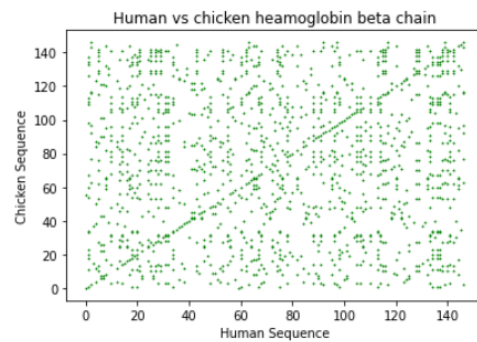
Bioinformatics Practical 4

Kudari Pavani
BE19B023

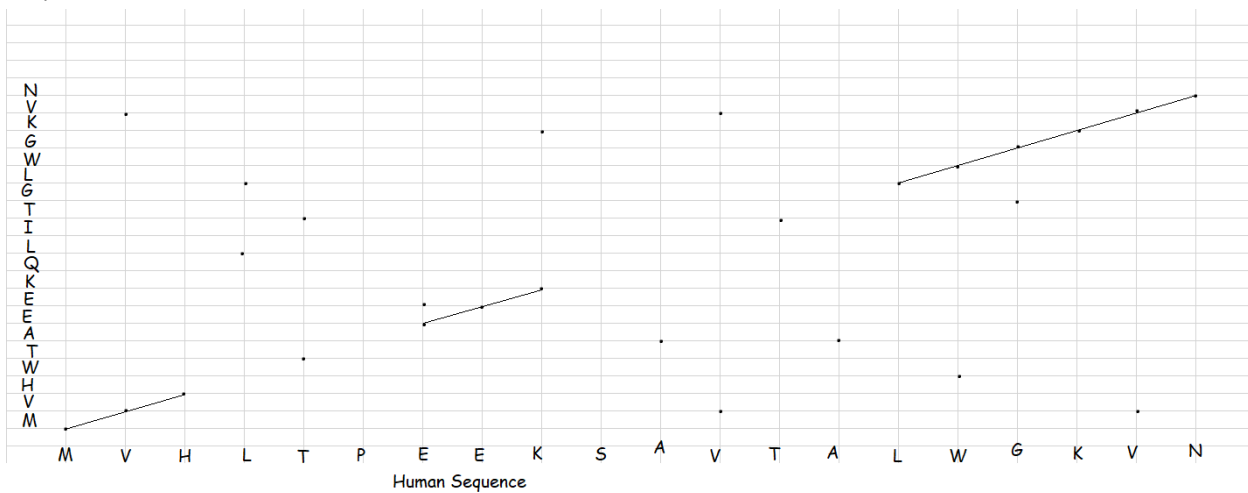
1.a)

```
import matplotlib.pyplot as plt
import numpy as np
c_sequence = 'MVHWTAEKQLITGLWGKVNVAECGAELARLLIVYPWTQRRFFASFGNLSPTAILGNPMVRAHGKKVLTSGDAVKNLNDIKNTFSQLSELHCDKLHVDPENFRLGDIILIVL
h_sequence = 'MVHLTPEEKSAVTALWGKVNVDDEVGGEALGRLLVYPWTQRRFFESFGDLSTPDVAMGNPKVKAHGKKVLGAFSDGLAHLNLLKGTFTLSELHCDKLHVDPENFRLGIVLVLCV
def dot_plot(seq1, seq2):
    x_axis = []
    y_axis = []
    for i in range(len(seq1)):
        for j in range(len(seq2)):
            if seq1[i] == seq2[j]:
                x_axis = x_axis + [i]
                y_axis = y_axis + [j]
    plt.scatter(x_axis, y_axis, marker = "*", s = 1, color = 'green')
    plt.title('Human vs chicken hemoglobin beta chain')
    plt.xlabel('Human Sequence')
    plt.ylabel('Chicken Sequence')
    plt.show()
    return 0

dot_plot(c_sequence, h_sequence)
```



1.b)



2.

```
sequence1='AATCTATA'
sequence2='AAG--ATA'
match=1
match_count=0
mismatch=0
mismatch_count=0
origination_penalty=-2
o_count=0
length_penalty=-1
l_count=0
i=0
for i in range(len(sequence1)):
    if sequence1[i]==sequence2[i]:
        match_count+=1
    elif sequence1[i]=='-' or sequence2[i]=='-':
        l_count+=1
        if (sequence1[i]=='-' and sequence1[i+1]!='-') or (sequence2[i]=='-' and sequence2[i+1]!='-'):
            o_count+=1
mismatch_count=len(sequence1)-match_count
print(match_count*match+mismatch_count*mismatch+o_count*origination_penalty+l_count*length_penalty)
```

1

3.

Question-3:-

AATCTATA
AAG--ATA

Given sequence

Total Origination = 1

Total matches = 5

Total no. of mismatches = 1

$$\begin{aligned} \text{score} &= (\text{Matches} \times \text{match score}) + (\text{mismatch} \times \text{mismatch score}) + \\ & (\text{Origination} \times \text{origination penalty}) + (\text{length} \times \text{length penalty}) \\ &= (5 \times 1) + (1 \times 0) + (1 \times -2) + (2 \times -1) \\ &= -1 \end{aligned}$$

4.

```
string1='ACAGTCGAACG'
string2='ACCGTCCG'
x=len(string1)
y=len(string2)
def NWdp(string1,string2,x,y):
    if x==0:
        return y-2
    elif y==0:
        return x-2
    elif string1[x-1]==string2[y-1]:
        return 2+ NWdp(string1,string2,x-1,y-1)
    else:
        return max(NWdp(string1,string2,x-1,y)-2,NWdp(string1,string2,x,y-1)-2,NWdp(string1,string2,x-1,y)-1)
print(NWdp(string1,string2,x,y))
```

7

5.

		A	C	A	G	T	C	G	A	A	C	G
	0	-2	-4	-6	-8	-10	-12	-14	-16	-18	-20	-22
A	-2	2	0	-2	-4	-6	-8	-10	-12	-14	-16	-18
C	-4	0	4	2	0	-2	-4	-6	-8	-10	-12	-14
C	-6	-2	2	3	1	-1	0	-2	-4	-6	-8	-10
G	-8	-4	0	1	5	3	1	2	0	-2	-4	-6
T	-10	-6	-2	-1	3	7	5	3	1	-1	-3	-5
C	-12	-8	-4	-3	1	5	9	7	5	3	1	-1
C	-14	-10	-6	-5	-1	3	7	8	6	4	5	3
G	-16	-12	-8	-7	-3	1	5	9	7	5	3	7