BT3040 – BIOINFORMATICS – Assignment 12

Submitted by Sahana (BE17B038)

Question 1

Get the PDB files, 1ALC, 4 LYZ and 1TIM

Done

Question 2

Construct contact maps (A chain; C_α atoms; distance cutoff: 8 Å) for the three proteins. Comment on the results.

Code –

To parse the .pdb file in python and extract only positions

```
all_pos = []
sequence = ''
for line in open('D:/Textbook-Sem6/Bioinformatics/PDB/4lyz.pdb'):
    list = line.split()
    id = list[0]
    if id == 'ATOM':
        type = list[2]
        if type == 'CA':
            residue = list[3]
            type_of_chain = list[4]
            atom_count = int(list[5])
            position = list[6:9]
            if atom_count >= 0:
                if type_of_chain == 'A':
                    sequence+=residue
                    x = float(position[0])
                    y = float(position[1])
                    z = float(position[2])
                    all_pos.append([x,y,z])
```

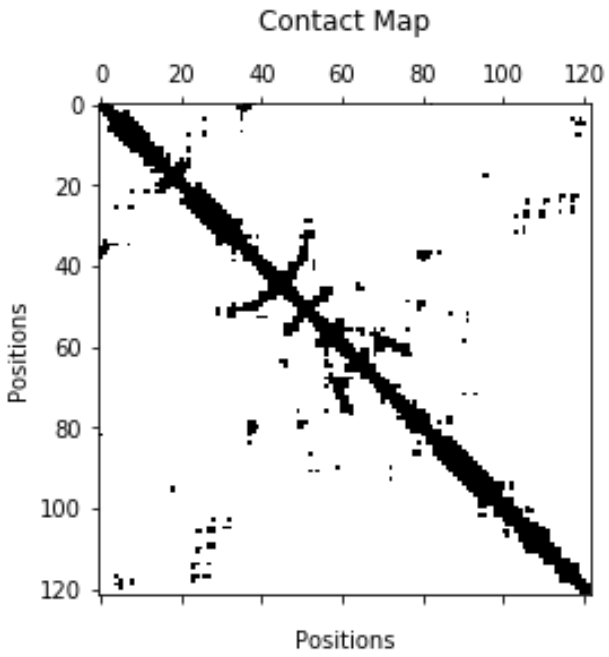
To check positions and construct contact maps

```
#Contact map

#all_pos is the list of coordinates of all CA residues of the A chain

import matplotlib.pylab as plt
import math
n = len(all_pos)
#contact map
CM = [[0 for i in range(n)] for i in range(n)]
```

```
for i in range(n):
    pos1 = all_pos[i]

    for j in range(n):
        pos2 = all_pos[j]
        #Calculate distance between residues i and j
        dist = math.sqrt(sum([(a - b) ** 2 for a, b in zip(pos1,
pos2)]))
        if dist< 8:
            CM[i][j]=1
plt.spy(CM)

plt.title('Contact Map\n')
plt.xlabel('\nPositions')
plt.ylabel('Positions')
```

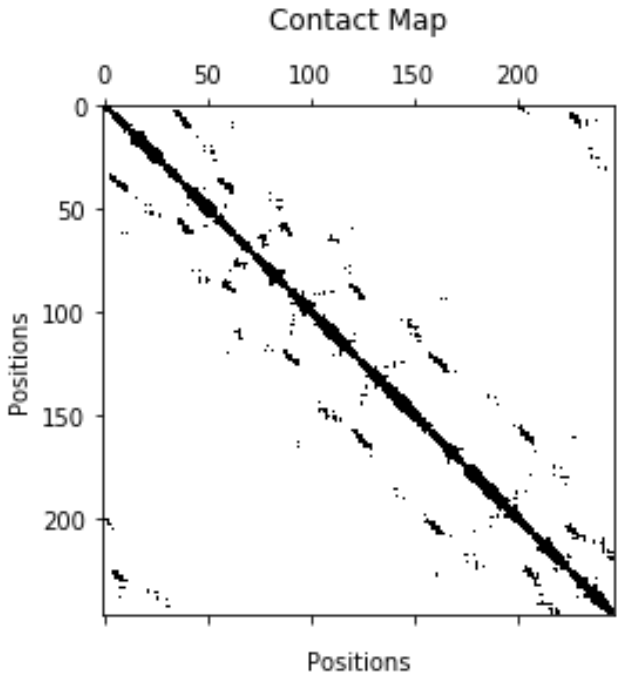
Results –

1ALC protein -



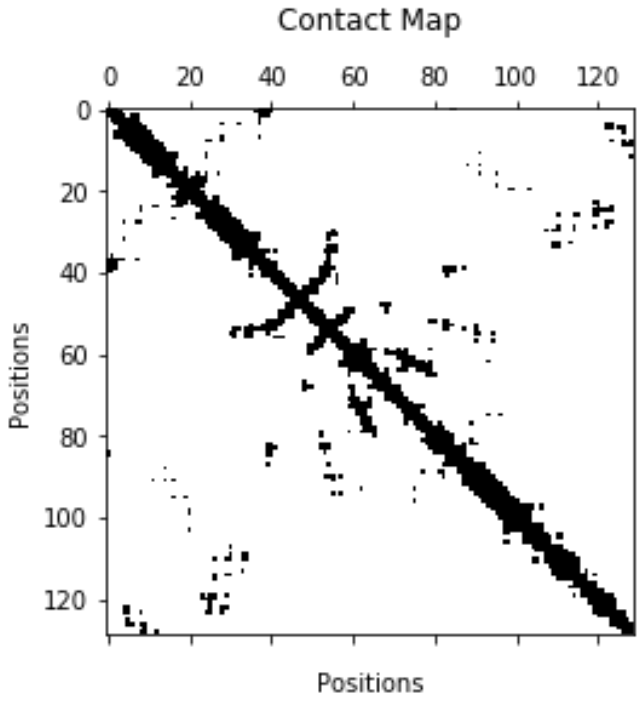
Reference – There are lesser dots that are really far away from the main diagonal. This suggests that this protein has very little long-range contacts. There is a visible cloud around the main diagonal and this represents medium range contacts, i.e., the ones formed in a helical structure. Hence this protein should have more helices than sheets.

1TIM protein -



Reference – This protein has a lot of points that are scattered away from the main diagonal and separated by distance of more than 4-5 residues. This means that the protein has a lot of long-range contacts, and therefore it has a higher percentage of beta sheets than alpha helices.

4LYZ Protein –



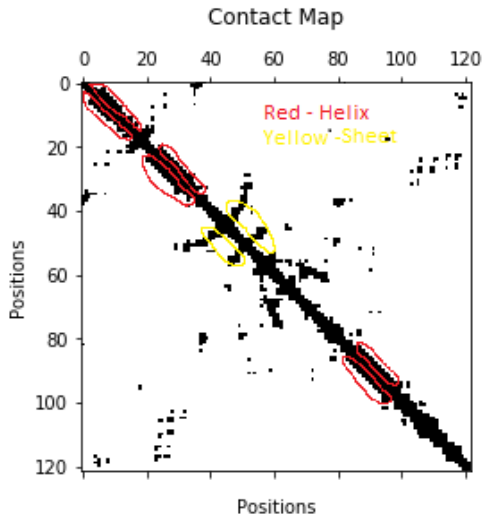
Reference – This map shows a lot of scatters around the main diagonal both at distances less than 4 and greater than 4. Hence it has a good proportion of both the secondary structures – helices and sheets.

Question 3

Identify two helical and strand segments from contact maps.

- Medium range contacts are a part of helix.
- Lower range of long-range contacts are a part of beta sheets
- Higher range of long-range contacts are a part of turns

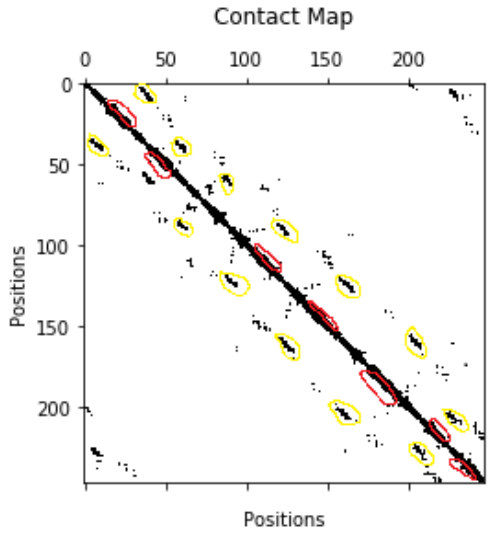
(The mentioned regions next to each contact maps are approximate estimates as finding exact residue from the figure is a bit difficult.)



1ALC

Helix (approx.) = 5 – 20 residues and 22-35 residues

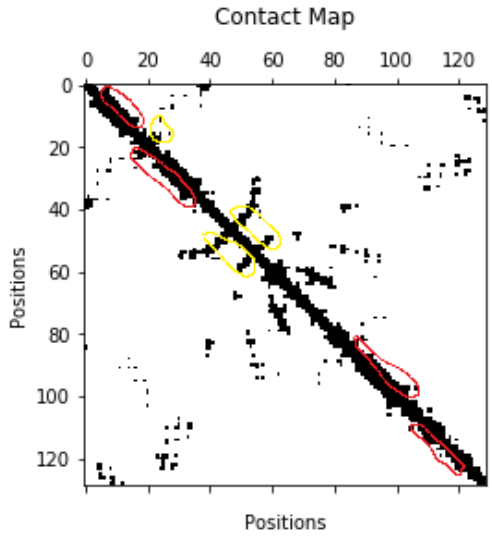
Sheet (approx.) = 40 - 43 residues and 45-50 residues



1TIM

Helix (approx.) = 15 – 30 residues and 42 – 55 residues

Sheet (approx.) = 5 – 10 residues and 60 – 65 residues



4LYZ

Helix (approx.) = 5 – 17 residues and 23 – 36 residues

Sheet (approx.) = 40 – 25 residues and 55 – 60 residues

Question 4

List two pairs of short-range, medium-range and long-range contacts in 4LYZ

Code –

```
short = []
medium = []
long = []
import numpy as np

for i in range(n):
    contacts = CM[i]
    res = [idx for idx, val in enumerate(contacts) if val != 0]
    l = len(res)
    for j in range(l):
        for k in range(j+1,l):
            pair = data[j][0]+str(data[res[j]][1])+'-
'+data[k][0]+str(data[res[k]][1])
            dif = res[k]-res[j]
            if dif<3:
                short.append(pair)
            if dif==3 or dif==4:
                medium.append(pair)
            if dif>4:
                long.append(pair)

print('Two pair of short range contact are - %s and %s\n'
%(short[0],short[1]))
print('Two pair of medium range contact are - %s and %s\n'
%(medium[0],medium[1]))
print('Two pair of long range contact are - %s and %s\n'
%(long[0],long[1]))
```

Result –

Two pair of short range contact are - LYS1-VAL2 and LYS1-PHE3

Two pair of medium range contact are - GLY38-GLU41 and LYS1-GLY4

Two pair of long range contact are - LYS1-GLY38 and LYS1-ARG39

Question 5

Compute contact order and long-range order for all the three proteins. Comment on the results.

Code –

```
#To calculate contact order

separation_dist = 0 #Sij
n_contacts = 0 #N - number of contacts
l = n #L - number of residues

for i in range(n):
    pos1 = all_pos[i]

    for j in range(i+1,n):
        pos2 = all_pos[j]
        dist = math.sqrt(sum([(a - b) ** 2 for a, b in zip(pos1,
pos2)]))
        if i!=j:
            # In a radius of 6 A
            if dist<6:
                separation_dist+=abs(j-i)
                n_contacts+=1
contact_order = (separation_dist)/(n_contacts*l)

print(contact_order)

#To calculate long-range order

l_separation_dist = 0 #nij
N = n #N - total number of residues

for i in range(n):
    pos1 = all_pos[i]

    for j in range(i+1,n):
        pos2 = all_pos[j]
        dist = math.sqrt(sum([(a - b) ** 2 for a, b in zip(pos1,
pos2)]))
        #in a radius of 8 A
        if dist<8:
            #if the 2 residues are atleast 12 residues apart in
sequence
```

```
            if abs(i-j)>12:
                l_separation_dist+=1

long_contact_order = (l_separation_dist)/(N)

print(long_contact_order)
```

Answer -

Protein Name	Separation residues (sigma S_{ij})	Total number of contacts (N)	Total number of residues (L)	Contact Order	n_{ij} in long range order (sigma nij)	Long range order
1ALC	2062	331	122	0.0510623545143876	109	0.8934426229508197
4LYZ	2390	362	129	0.05117992205233629	156	1.2093023255813953
1TIM	6156	654	247	0.038108680310515175	395	1.5991902834008098

Comments –

The long-range order of 1ALC is lesser than that of 4LYZ and 1TIM. This could be due to the absence of multiple beta sheets that might otherwise contribute to increasing long range contact. 1TIM seems to have the highest number of beta sheets (which is obvious because it is a membrane barrel protein) than 4LYZ.

The overall contact order of 1ALC and 4LYZ are comparable but not that of 1TIM. This suggests that 1TIM makes lesser non-local contacts when compared to the other 2 proteins.

Question 6

Compute the long-range order for the residues, Leu108 and Leu162 in 1TIM.

Code –

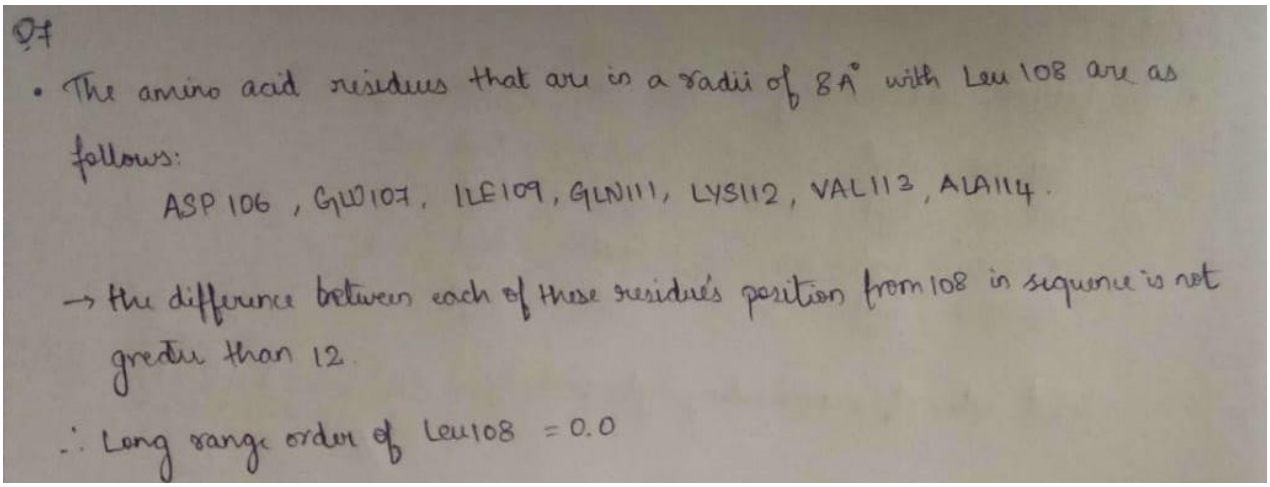
```
def long_range(position):
    #position = position of the residue for which we would find long-
range order
    l_separation_dist = 0

    for i in range(n):
        if position ==data[i][1]:
            pos1 = data[i][2]
```

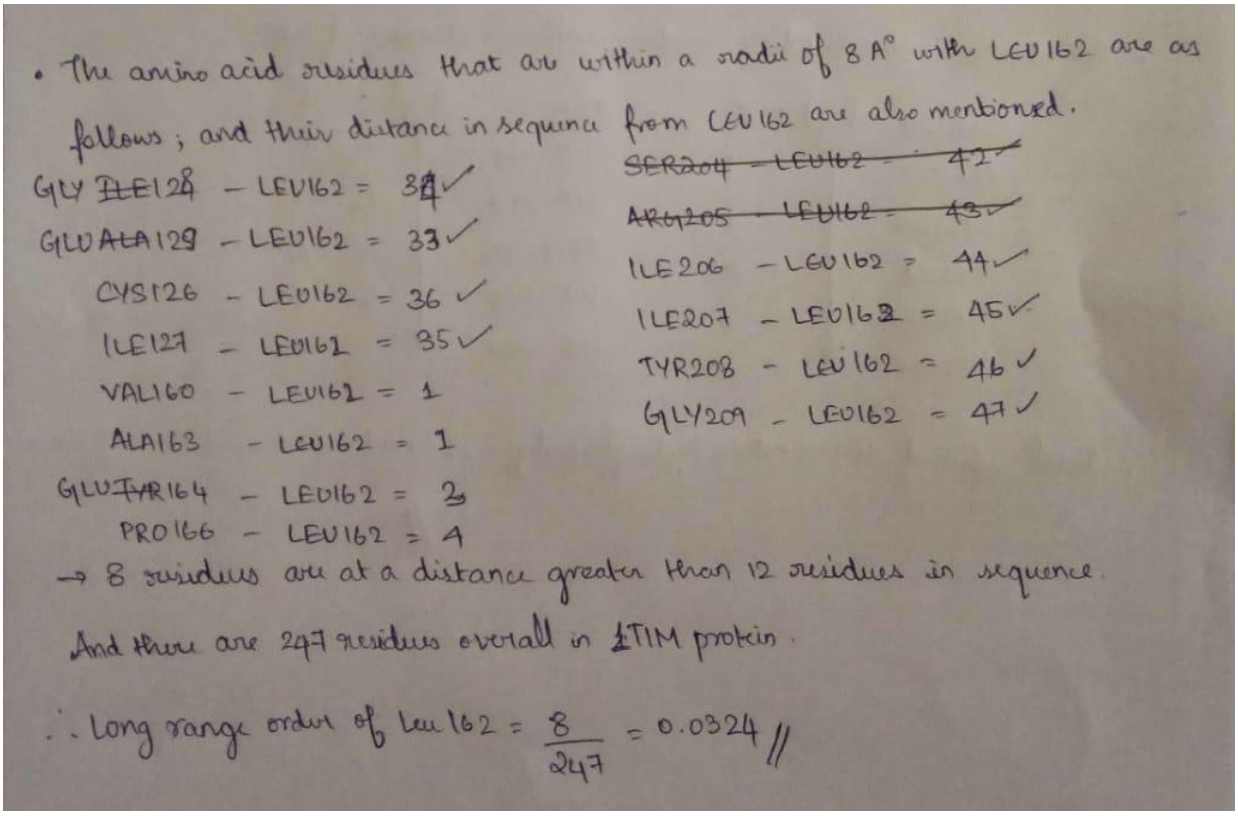
```
    for i in range(n):
        j = data[i][1]
        pos2 = data[i][2]
        dist = math.sqrt(sum([(a - b) ** 2 for a, b in zip(pos1,
pos2)]))

        #in a radius of 8 A
        if dist<=8:
            if j!=position:
                if abs(j-position)>12:
                    l_separation_dist+=1
    long_contact_order = (l_separation_dist)/n
    print(long_contact_order)

long_range(162)
```

Result –

```
Long range order =

For residue Leu108 = 0.0
For residue Leu162 = 0.032388663967611336
```

Question 7 - Check the values manually





Question 8

Compute the surrounding hydrophobicity for the residues, Leu108 and Leu162 in 1TIM.

Code -

```
def surrounding_hyd(position):
    nSH = {'ALA':1, 'CYS':1, 'GLY':1, 'MET':1, 'TYR':1,
           'PHE':2, 'ILE':2, 'LEU':2, 'VAL':2, 'TRP':2,
           'ASP':-2, 'GLU':-2, 'HIS':-2, 'LYS':-2, 'ARG':-2,
           'ASN':-1, 'PRO':-1, 'GLN':-1, 'SER':-1, 'THR':-1,}
    hyd = 0

    for i in range(n):
        if position ==data[i][1]:
            pos1 = data[i][2]

    for i in range(n):
        j = data[i][1]
        pos2 = data[i][2]
        dist = math.sqrt(sum([(a - b) ** 2 for a, b in zip(pos1,
pos2)]))
        #in a radius of 8 A
        if dist<=8:
            if j!=position:
                residue = nseq[i][0]
```

```
                print(residue, j)
                hyd+=nSH[residue]
    print('The surrounding hydrophobicity value for %dth residues =
%d'%(position,hyd))
surrounding_hyd(162)
```

Result –

For position 108 –

```
GLU 104
SER 105
ASP 106
GLU 107
ILE 109
GLY 110
GLN 111
LYS 112
The surrounding hydrophobicity value for 108th residues = -7
```

For position 162 –

```
ILE 124
ALA 125
CYS 126
ILE 127
VAL 160
VAL 161
ALA 163
TYR 164
SER 204
ARG 205
ILE 206
ILE 207
The surrounding hydrophobicity value for 162th residues = 13
```

Question 9

Check the values manually