Class09: Structural Bioinformatics pt1

Mahsa Naeimi

1: Introduction to the RCSB Protein Data Bank

To read the file we are going to use command `read.csv`:

```
pdb_stats <- read.csv('Data Export Summary.csv',
                      row.names=1)
head(pdb_stats)
```

```
                         X.ray     EM    NMR Multiple.methods Neutron Other
Protein (only)         154,766 10,155 12,187               191      72    32
Protein/Oligosaccharide  9,083  1,802     32                 7       1     0
Protein/NA               8,110  3,176    283                 6       0     0
Nucleic acid (only)      2,664     94  1,450                12       2     1
Other                      163      9     32                 0       0     0
Oligosaccharide (only)      11      0      6                 1       0     4
                         Total
Protein (only)         177,403
Protein/Oligosaccharide  10,925
Protein/NA               11,575
Nucleic acid (only)       4,223
Other                       204
Oligosaccharide (only)       22
```

we need to sum all the elements of the X.ray column:

```
(pdb_stats$X.ray)
```

```
[1] "154,766" "9,083"   "8,110"   "2,664"   "163"     "11"
```

we are going to use `gsub` to remove the commas

```
gsub(',','',pdb_stats$X.ray)
```

```
[1] "154766" "9083"   "8110"   "2664"   "163"    "11"
```

```
as.numeric(gsub(',','',pdb_stats$X.ray))
```

```
[1] 154766   9083   8110   2664    163     11
```

We use the sum command to get the sum

```
n_xray <- sum(as.numeric(gsub(',','',pdb_stats$X.ray)))
n_em <- sum(as.numeric(gsub(',','',pdb_stats$EM)))
n_Total <-sum(as.numeric(gsub(',','',pdb_stats$Total)))
```

- **Q1:** What percentage of structures in the PDB are solved by X-Ray and Electron Microscopy.

```
p_xray <- (n_xray /n_Total)
p_em <- (n_em) / n_Total

p_xray
```

```
[1] 0.8553721
```

```
p_em
```

```
[1] 0.07455763
```

```
p_Total = p_xray + p_em
p_Total
```

```
[1] 0.9299297
```

- **Q2:** What proportion of structures in the PDB are protein?

```
total_protein <- as.numeric( gsub(',', '', pdb_stats[1, 7]) )
 total_protein
```

```
[1] 177403
```

```
total_protein/ n_Total
```

```
[1] 0.8681246
```

- **Q3:** Type HIV in the PDB website search box on the home page and determine how many HIV-1 protease structures are in the current PDB? 2064 stucture

2. Visualizing the HIV-1 protease structure

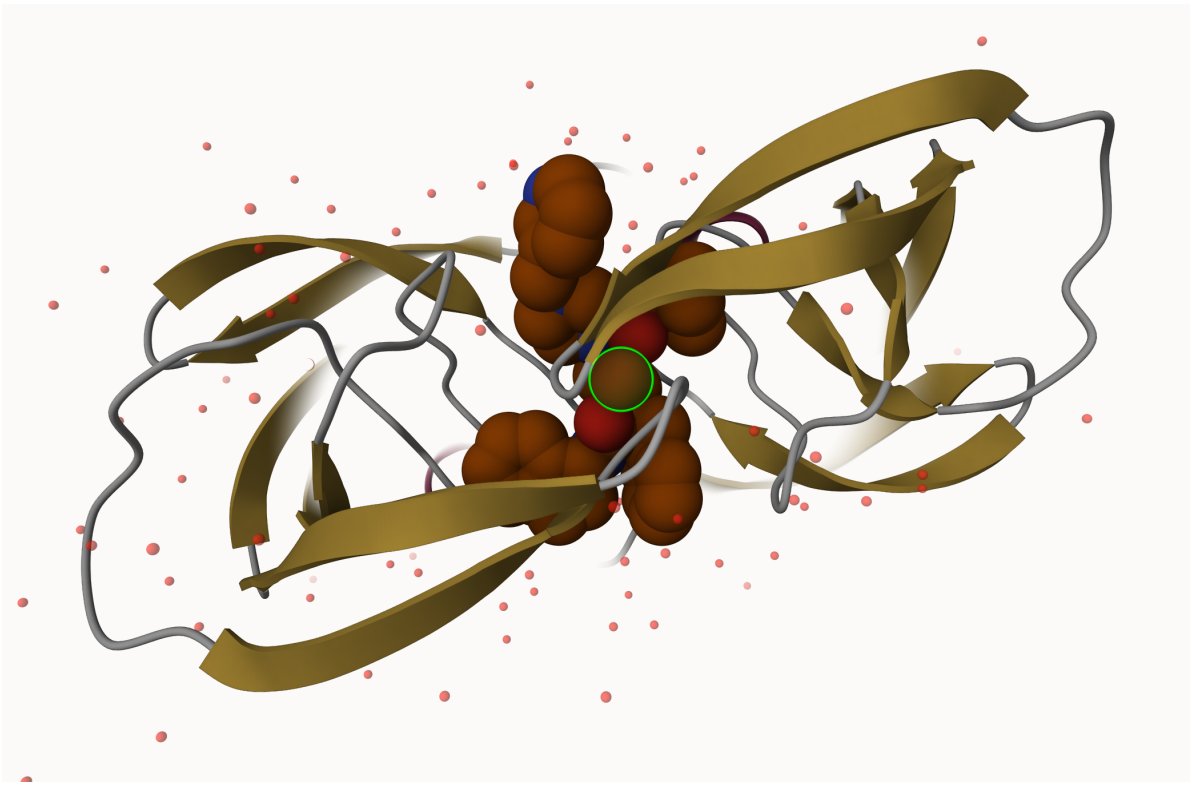
Q4: Water molecules normally have 3 atoms. Why do we see just one atom per water molecule in this structure? **to see the structure more clear and get better visualization**

Q5: There is a critical "conserved" water molecule in the binding site. Can you identify this water molecule? What residue number does this water molecule have? **HOH 308**

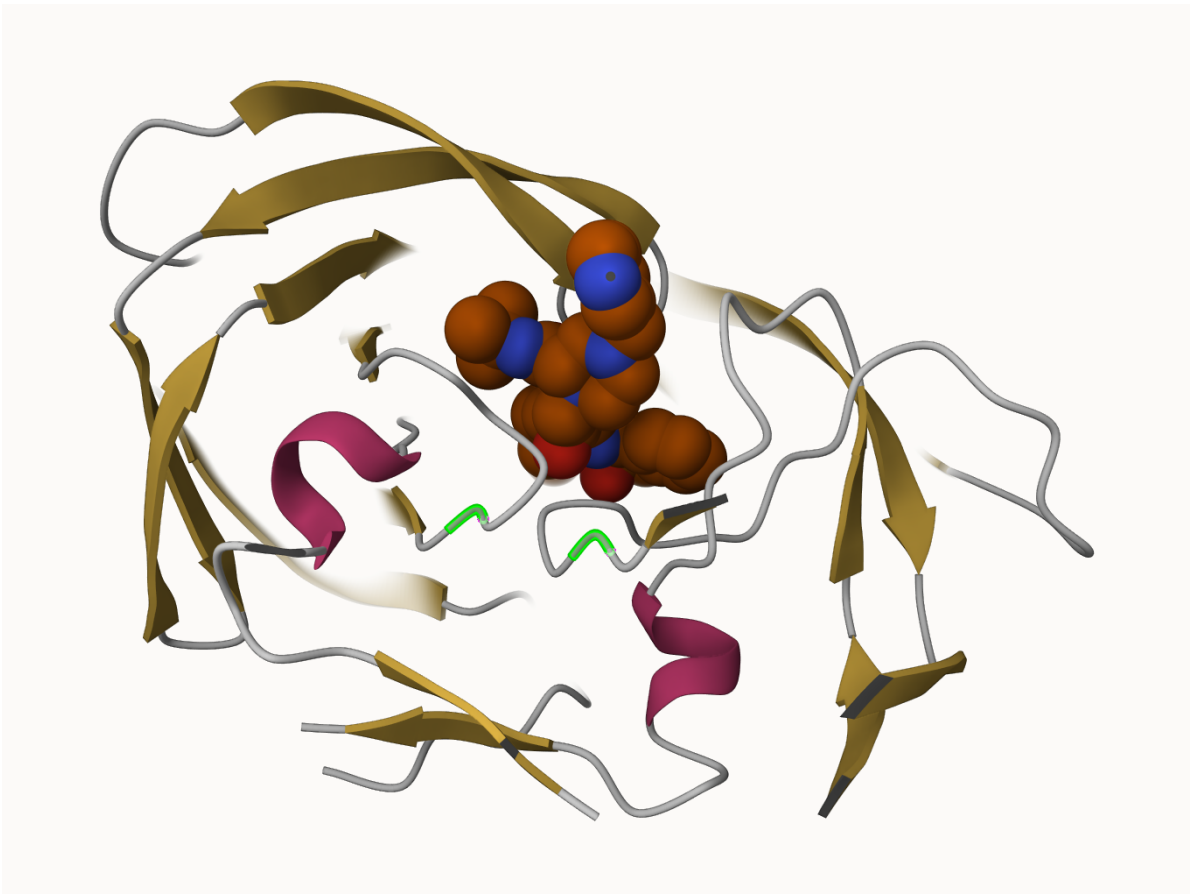
Q6: Generate and save a figure clearly showing the two distinct chains of HIV-protease along with the ligand. You might also consider showing the catalytic residues ASP 25 in each chain and the critical water (we recommend *"Ball & Stick"* for these side-chains). Add this figure to your Quarto document.

Showing ligand in spacefilled presentation (1HSG model) :

Showing ater molecules surrounding the model with critical water molecule selected (1HSG model):



Showing residue ASP 25 in two chains (1HSG model):



3. Introduction to Bio3D in R

```
#install.packages("bio3d")
library(bio3d)
pdb <-  read.pdb("1hsg")
```

```
Note: Accessing on-line PDB file
```

```
pdb
```

```

Call:  read.pdb(file = "1hsg")
```

```
   Total Models#: 1
     Total Atoms#: 1686,  XYZs#: 5058  Chains#: 2  (values: A B)

     Protein Atoms#: 1514  (residues/Calpha atoms#: 198)
     Nucleic acid Atoms#: 0  (residues/phosphate atoms#: 0)

     Non-protein/nucleic Atoms#: 172  (residues: 128)
     Non-protein/nucleic resid values: [ HOH (127), MK1 (1) ]

   Protein sequence:
      PQITLWQRPLVTIKIGGQLKEALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYD
      QILIEICGHKAIGTVLVGPTPVNIIGRNLLTQIGCTLNFPQITLWQRPLVTIKIGGQLKE
      ALLDTGADDTVLEEMSLPGRWKPKMIGGIGGFIKVRQYDQILIEICGHKAIGTVLVGPTP
      VNIIGRNLLTQIGCTLNF

+ attr: atom, xyz, seqres, helix, sheet,
        calpha, remark, call
```

Q7: How many amino acid residues are there in this pdb object?

198

Q8: Name one of the two non-protein residues?

HOH and KM1

Q9: How many protein chains are in this structure?

2 chains

```
attributes(pdb)
```

```
$names
[1] "atom"   "xyz"    "seqres" "helix"  "sheet"  "calpha" "remark" "call"

$class
[1] "pdb" "sse"
```

```
head(pdb$atom)
```

```
  type eleno elety  alt resid chain resno insert      x      y     z o     b
1 ATOM     1     N <NA>   PRO     A     1   <NA> 29.361 39.686 5.862 1 38.10
```

```
2 ATOM     2    CA <NA>   PRO     A     1   <NA> 30.307 38.663 5.319 1 40.62
3 ATOM     3     C <NA>   PRO     A     1   <NA> 29.760 38.071 4.022 1 42.64
4 ATOM     4     O <NA>   PRO     A     1   <NA> 28.600 38.302 3.676 1 43.40
5 ATOM     5    CB <NA>   PRO     A     1   <NA> 30.508 37.541 6.342 1 37.87
6 ATOM     6    CG <NA>   PRO     A     1   <NA> 29.296 37.591 7.162 1 38.40
  segid elesy charge
1  <NA>     N   <NA>
2  <NA>     C   <NA>
3  <NA>     C   <NA>
4  <NA>     O   <NA>
5  <NA>     C   <NA>
6  <NA>     C   <NA>
```

Predicting functional motions of a single structure by NMA

for thhis part we are going to read a new PDB structure of Adenylate Kinase and perform Normal mode analysis:

```
adk <- read.pdb('6s36')
```

```
Note: Accessing on-line PDB file
 PDB has ALT records, taking A only, rm.alt=TRUE
```

```
adk
```

```
Call:  read.pdb(file = "6s36")

  Total Models#: 1
    Total Atoms#: 1898,  XYZs#: 5694  Chains#: 1  (values: A)

    Protein Atoms#: 1654  (residues/Calpha atoms#: 214)
    Nucleic acid Atoms#: 0  (residues/phosphate atoms#: 0)

    Non-protein/nucleic Atoms#: 244  (residues: 244)
    Non-protein/nucleic resid values: [ CL (3), HOH (238), MG (2), NA (1) ]

  Protein sequence:
     MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
     DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDKI
```

```
      VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
      YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG

+ attr: atom, xyz, seqres, helix, sheet,
        calpha, remark, call
```

with nma or normal mode analysis we can predict protein flexibility and potential functional motions:

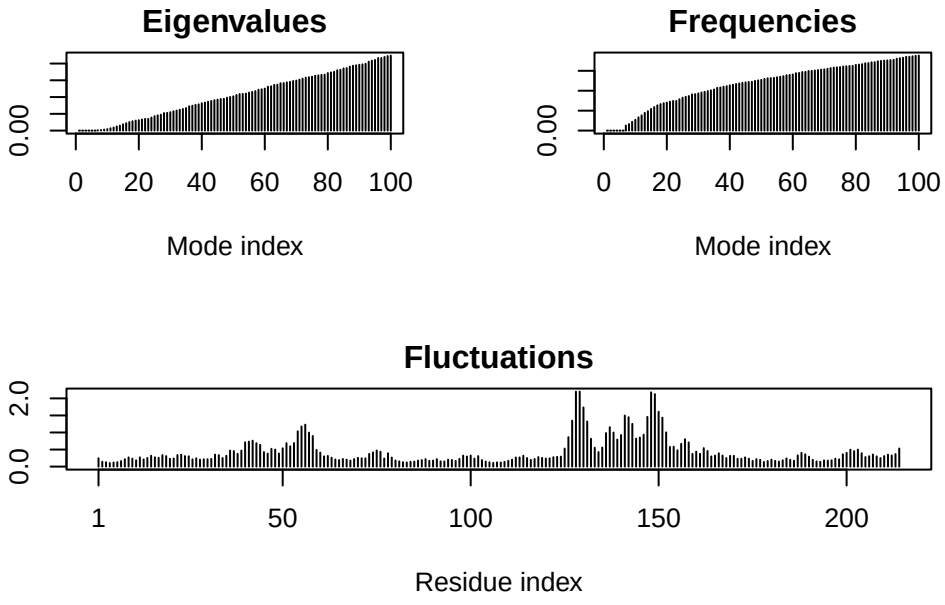
```
m <- nma(adk)
```

```
 Building Hessian...        Done in 0.081 seconds.
 Diagonalizing Hessian...   Done in 0.418 seconds.
```

```
class(m)
```

```
[1] "VibrationalModes" "nma"
```

```
plot(m)
```



To view a “movie” of these predicted motions we generate a molecular “trajectory” with the `mktrj()` function and load it in the Mol* :

```
mktrj(m, file="adk_m7.pdb")
```

Then we were able to generate the animation in Mol*.