Class 10: Structural Bioinformatics

Dylan Mullaney (A16869792)

Table of contents

1. Introduction to the RCSB Protein Data Bank (PDB)

The main repository of biomolecular structure data is called the PDB found at: https://www.rcsb.org/

Let's see what this database contains. I went to PDB > Analyze > PDB Statistics > By Experimental method and molecular type

```
pdbstats <- read.csv ("Data Export Summary.csv")
pdbstats
```

```
           Molecular.Type   X.ray     EM    NMR Multiple.methods Neutron Other
1          Protein (only) 169,563 16,774 12,578              208      81    32
2 Protein/Oligosaccharide   9,939  2,839     34                8       2     0
3              Protein/NA   8,801  5,062    286                7       0     0
4     Nucleic acid (only)   2,890    151  1,521               14       3     1
5                   Other     170     10     33                0       0     0
6  Oligosaccharide (only)      11      0      6                1       0     4
    Total
1 199,236
2  12,822
3  14,156
4   4,580
```

```
5     213
6      22
```

Looks like columns with numbers over 999 have become chr due to the commas. We cannot do math with these values - they must be a numeric type. I can fix this by replacing "," for nothings "" with the `sub()` function:

```
#x<- pdbstats$X.ray
#sum (as.numeric(sub(",", "", x)))
```

Or I can use the **readr** package and the `read_csv()` function.

```
library (readr)
pdbstats <- read_csv ("Data Export Summary.csv")
```

```
Rows: 6 Columns: 8
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): Molecular Type
dbl (3): Multiple methods, Neutron, Other
num (4): X-ray, EM, NMR, Total

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
pdbstats
```

```
# A tibble: 6 x 8
  `Molecular Type`   `X-ray`    EM   NMR `Multiple methods` Neutron Other  Total
  <chr>                <dbl> <dbl> <dbl>              <dbl>   <dbl> <dbl>  <dbl>
1 Protein (only)      169563 16774 12578                208      81    32 199236
2 Protein/Oligosacc~    9939  2839    34                  8       2     0  12822
3 Protein/NA            8801  5062   286                  7       0     0  14156
4 Nucleic acid (onl~    2890   151  1521                 14       3     1   4580
5 Other                  170    10    33                  0       0     0    213
6 Oligosaccharide (~      11     0     6                  1       0     4     22
```

I want to clean up the column names so they're more consistently capitalized.

```
library (janitor)
```

```
Attaching package: 'janitor'

The following objects are masked from 'package:stats':

    chisq.test, fisher.test
```

```
df <- clean_names (pdbstats)
```

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

83%, 11%

Total # of xray, em

```
sum(df$x_ray)
```

```
[1] 191374
```

```
sum(df$em)
```

```
[1] 24836
```

Total number of structures

```
sum(df$total)
```

```
[1] 231029
```

Percentage of X ray

```
(sum(df$x_ray) / sum(df$total)) *100
```

```
[1] 82.83549
```

Percentage of electron microscopy

```
(sum(df$em) / sum(df$total)) *100
```

```
[1] 10.75017
```

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

86% of structures are protein

```
totalp <- df [1,8]
totalp
```

```
# A tibble: 1 x 1
   total
   <dbl>
1 199236
```

```
protein.structures <- (totalp/sum(df$total))*100
protein.structures
```

```
     total
1 86.23852
```

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?

Skipped

2. Using Mol*

The main Mol* homepage is at: https://molstar.org/viewer/ We can input our own PDB files or just give it a PDB accession code (the 4 letter PDB code).

I took a screenshot on Mol* and downloaded it as a png. Let's use markdown code to import it: ! [] () –> caption in square, location/title in smooth, must have dragged image to folder

Figure 1: Molecular View of 1HSG

More images with more specificity

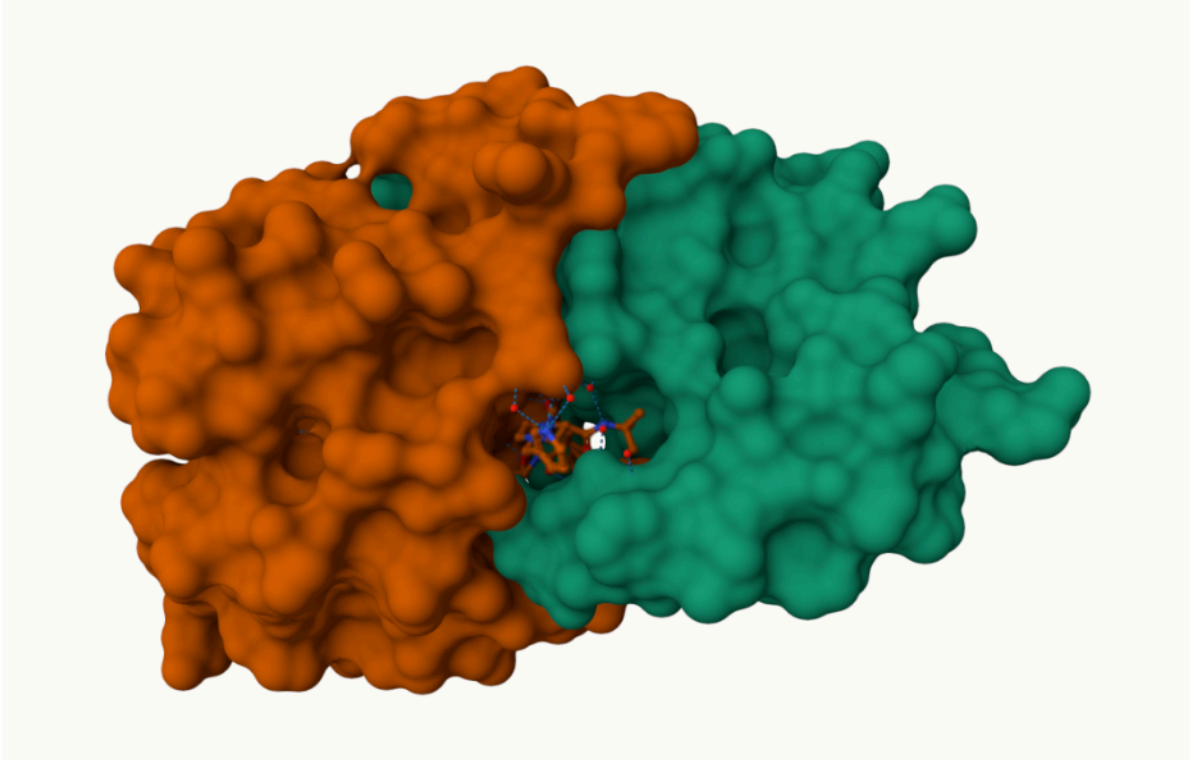


Figure 2: Molecular Binding Site

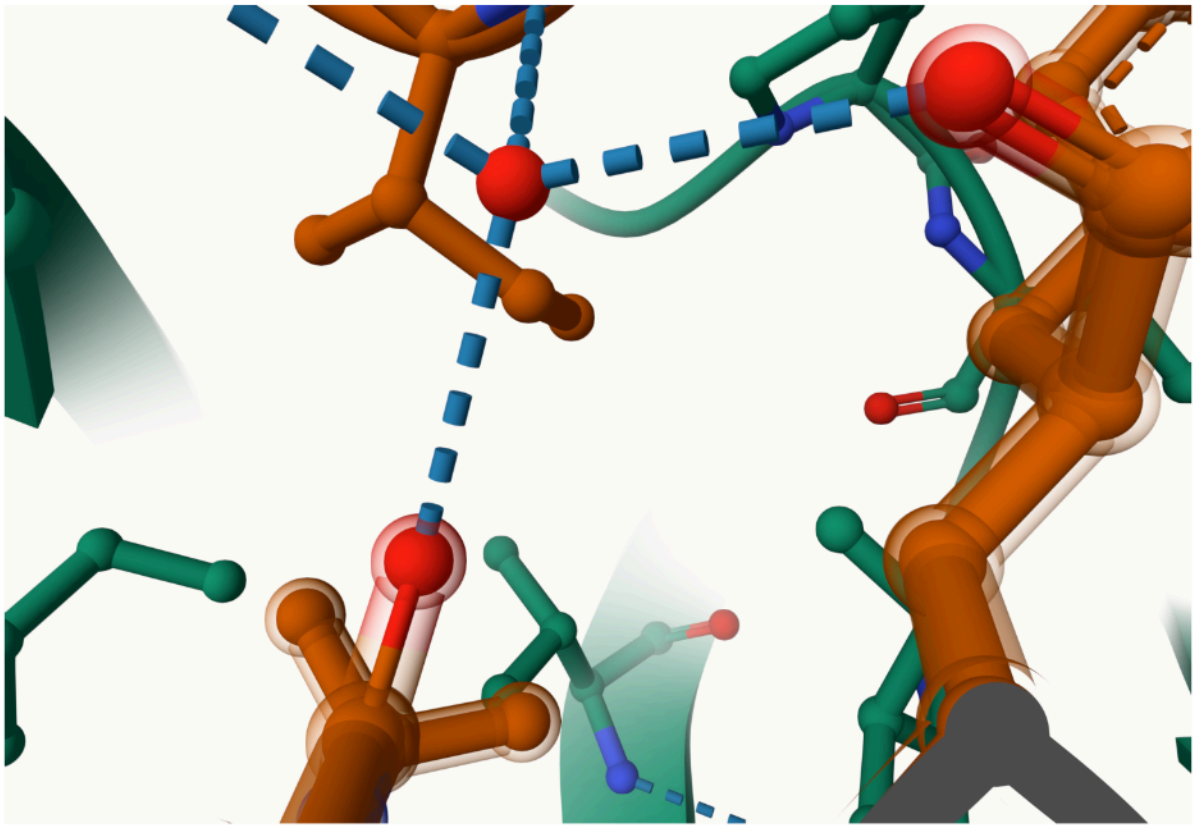


Figure 3: Conserved water in binding site

> Q4: Water molecules normally have 3 atoms. Why do we see just one atom per water molecule in this structure?

The hydrogen bonds are not shown. The presence of hydrogen is generally assumed and not explicitly added. This simplifies the image and makes it more digestible without losing vital information/

> 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?

This water molecule is shown in the "Conserved water in binding site" image above. This water is residue number 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.

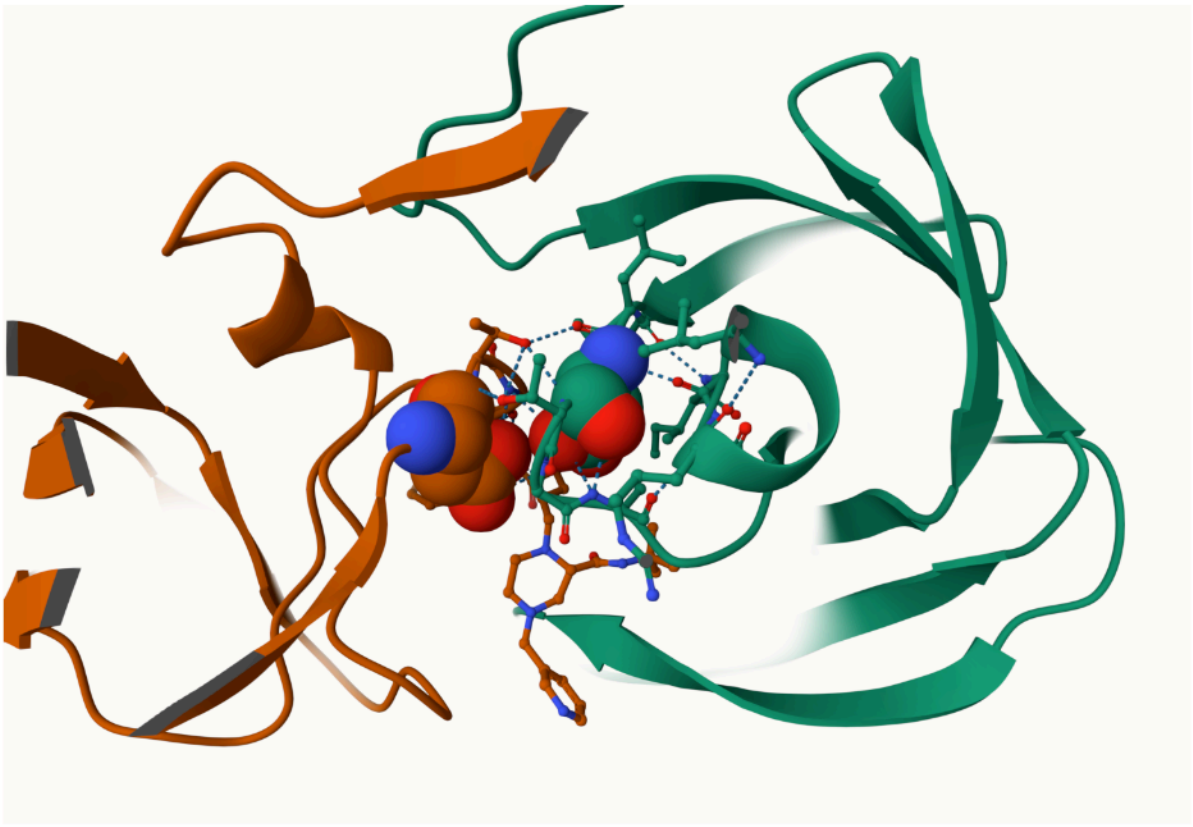


Figure 4: Two ASP shown in the binding sites

3. Intro to Bio3D in R

We can use the **bio3d** package for structural bioinformatics to read PDB data into R

```
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

```
length(pdbseq(pdb))
```

```
[1] 198
```

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

Water, MK1 (ligand)

Q9: How many protein chains are in this structure?

Two chains (A, B)

Looking at `pdb` in more details

```
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>
```

Let's try a new function not yet in the bio3d package and install "r3dmol" and "shiny" in R console

```
source("https://tinyurl.com/viewpdb")
#view.pdb(pdb, backgroundColor = "lightyellow")
```

The purpose of the bio3d package is to be able to make predictions and analyses on the data imported.

4. Predicting function dynamics

We can use the `nma()` function in bio3d to predict the large-scale functional motions of biomolecules.

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

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

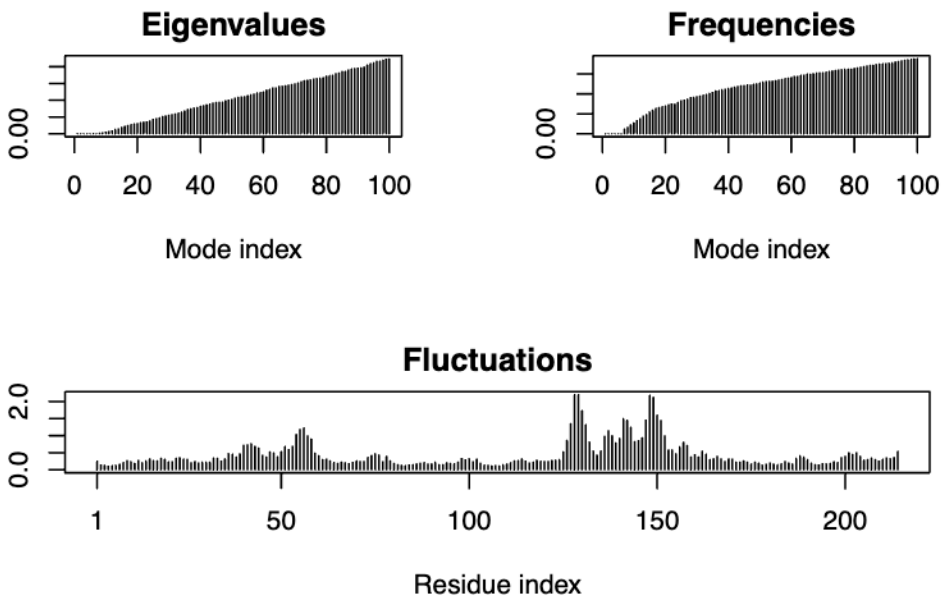
```
adk
```

```
 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
```

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

```
 Building Hessian...        Done in 0.015 seconds.
 Diagonalizing Hessian...   Done in 0.278 seconds.
```

```
plot(m)
```



Make a forward trajectory of the predicted molecular motion

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