Class 10: Structural Bioinformatics Pt 1

Julia Di Silvestri (A16950824)

Introduction to the RCSB Protein Data Bank

First, I will the data set, save it in my current project, and make sure that I can read it.

This may look slightly different from in class because I was not able to make it to class because of illness, so I am trying this on my own

```
# Read the CSV file
 data <- read.csv("pdbdata.csv", row.names = 1)

 # Remove commas and convert to numeric for each column
 data[] <- lapply(data, function(x) as.numeric(gsub(",", "", x)))

data
```

	X.ray	EM	NMR	Multiple.methods	Neutron	Other
Protein (only)	161663	12592	12337	200	74	32
Oligosaccharide (only)	11	0	6	1	0	4
Nucleic acid (only)	2758	125	1477	14	3	1
Protein/Oligosaccharide	9348	2167	34	8	2	0
Protein/NA	8404	3924	286	7	0	0
Other	164	9	33	0	0	0

	Total
Protein (only)	186898
Oligosaccharide (only)	22
Nucleic acid (only)	4378
Protein/Oligosaccharide	11559
Protein/NA	12621
Other	206

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

```
(sum(data$X.ray, data$EM)) / colSums(data, na.rm = T)
```

```
          X.ray              EM             NMR Multiple.methods
      1.1031928      10.6905989      14.1935370      874.6304348
        Neutron           Other           Total
   2546.3924051    5436.8918919       0.9326839
```

```
#another way to do it

(sum(data$X.ray, data$EM)) / sum(data$Total)
```

```
[1] 0.9326839
```

93.27% of all structures in the dataset are solved by X-Ray and Electron Microscopy

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

```
#First figure out how to specifiy the data we want to sum
ptot <- data[c(1, 4, 5), 7]
ptot
```

```
[1] 186898  11559  12621
```

```
#The sum it and divide it by the total data
sum(ptot)
```

```
[1] 211078
```

```
sum(ptot) / sum(data$Total)
```

```
[1] 0.9786447
```

97.86% of the structures are proteins

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?

The search returns 4,410 results

#PDB Format

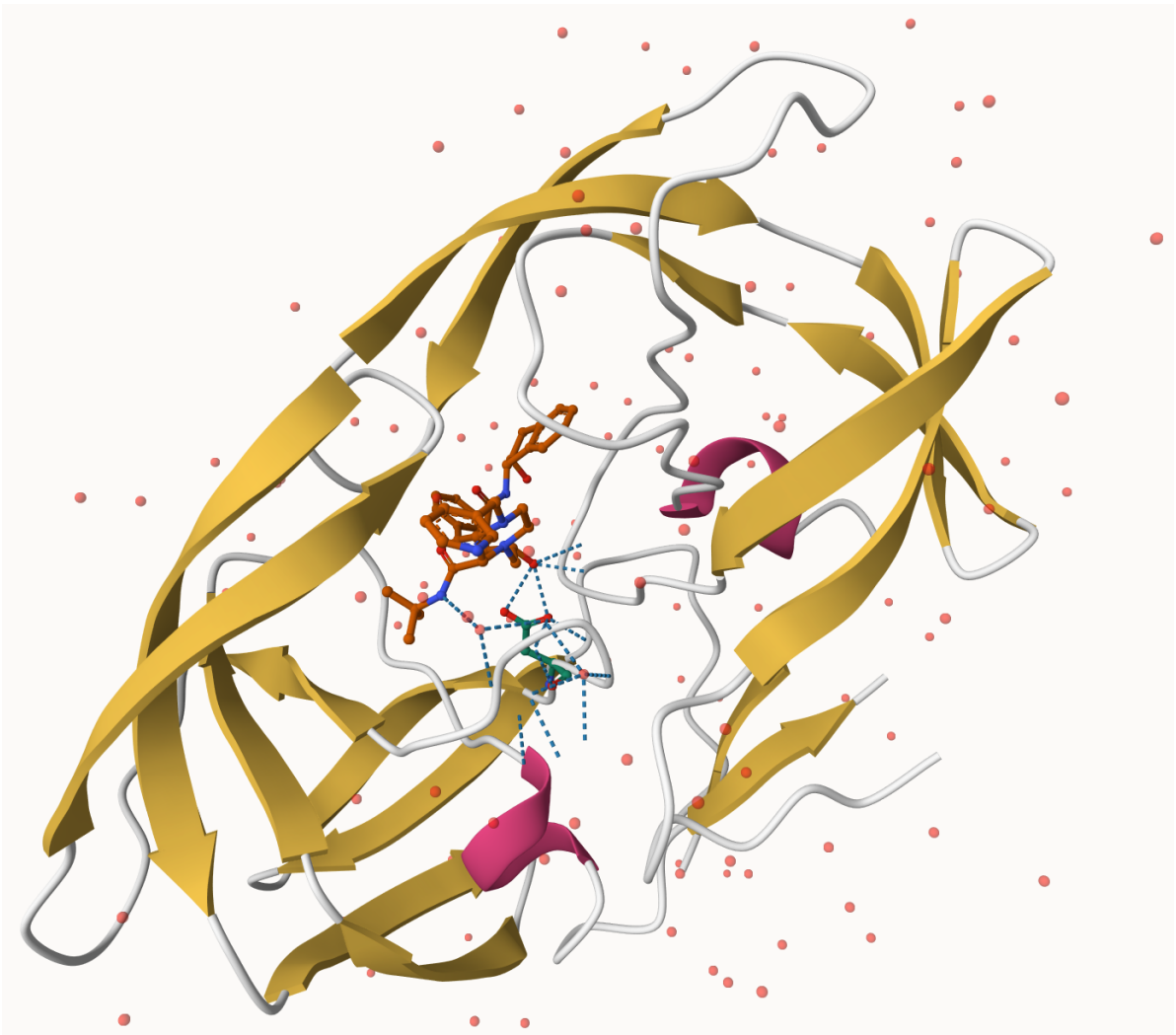
> Q4: Water molecules normally have 3 atoms. Why do we see just one atom per water molecule in this structure?

Hydrogen atoms aren't shown in the ball and stick format, so only the oxygen is displayed

> 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

The residue number of this water molecule is HOH 332

> 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.



#Introduction to Bio3D

```
#Loading in the package
library(bio3d)

#Reading file
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 (read from file)

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

HOH (read from file)

Q9: How many protein chains are in this structure?

2 (read from file)

To view the attributes of this object:

```
attributes(pdb)
```

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

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

#Predicting Functional Motions of a Single Structure

We will read a new PDB structure and perfrom Normal Mode analysis

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

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

```
adk
```

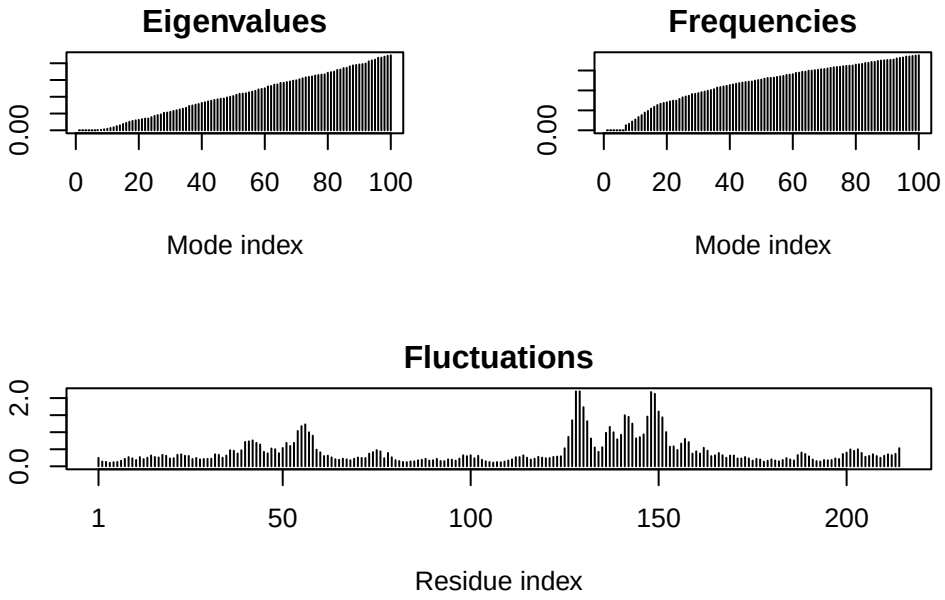
```
 Call:  read.pdb(file = "6s36")

   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
```

To perform a flexibility / conformational change prediction, use `nma()`

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

```
 Building Hessian...        Done in 0.013 seconds.
 Diagonalizing Hessian...   Done in 0.258 seconds.
```

```
plot(m)
```



We can generate a "movie" of these predicted motions using the `mktrj()` function and load the results into Mol*

#Comparative structure analysis of Adenylate Kinase

First, all of the packages need to be installed

```
# Install packages in the R console NOT your Rmd/Quarto file

install.packages("bio3d")
install.packages("devtools")
install.packages("BiocManager")

BiocManager::install("msa")
devtools::install_bitbucket("Grantlab/bio3d-view")
```

Q10. Which of the packages above is found only on BioConductor and not CRAN?

msa

Q11. Which of the above packages is not found on BioConductor or CRAN?

bio3d-view

> Q12. True or False? Functions from the devtools package can be used to install packages from GitHub and BitBucket?

True

To perform a BLAST search of our structure, we first need to get the sequence:

```
aa <- get.seq("1ake_A")
```

```
Warning in get.seq("1ake_A"): Removing existing file: seqs.fasta

Fetching... Please wait. Done.
```

```
aa
```

```
             1        .         .         .         .         .         60
pdb|1AKE|A   MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
             1        .         .         .         .         .         60

            61        .         .         .         .         .        120
pdb|1AKE|A   DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDRI
            61        .         .         .         .         .        120

           121        .         .         .         .         .        180
pdb|1AKE|A   VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
           121        .         .         .         .         .        180

           181        .         .         .   214
pdb|1AKE|A   YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG
           181        .         .         .   214

Call:
  read.fasta(file = outfile)

Class:
  fasta

Alignment dimensions:
  1 sequence rows; 214 position columns (214 non-gap, 0 gap)

+ attr: id, ali, call
```

Q13. How many amino acids are in this sequence, i.e. how long is this sequence?

214

Now we can run the BLAST:

```
# Blast or hmmer search
b <- blast.pdb(aa)
```

^ could not get the request to go through – could not do much with this