Class 10: Structural Bioinformatics Analysis Pt. 1

Snehita Vallumchetla (PID: A16853399)

Table of contents

1. The PDB database

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

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

First let's clean the data:

```
pdbstats <- read.csv('data.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
```

```
pdbstats$X.ray
```

```
[1] "169,563" "9,939"   "8,801"   "2,890"   "170"     "11"
```

The comma in these numbers is causing them to be read as characters rather than numeric.

I can fix this by replacing "," with the `sub()` function.

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

```
[1] 191374
```

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

```
library(readr)

pdbstats <- read_csv('data.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 the column names so they are all lower case

```
library(janitor)
```

```
Attaching package: 'janitor'

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

    chisq.test, fisher.test
```

```
df <- clean_names(pdbstats)

df
```

```
# 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/Oligosacchar~   9939  2839    34                8       2     0  12822
3 Protein/NA              8801  5062   286                7       0     0  14156
4 Nucleic acid (only)     2890   151  1521               14       3     1   4580
5 Other                    170    10    33                0       0     0    213
6 Oligosaccharide (onl~     11     0     6                1       0     4     22
```

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

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

```
[1] 191374
```

total number of structures

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

```
[1] 231029
```

Percentage of the X-ray Structures:

```
xray_percent <- (sum(df$x_ray)/sum(df$total))*100

xray_percent
```

```
[1] 82.83549
```

Percentage of Electron Microscopy

```
em_percent <- (sum(df$em)/sum(df$total))*100

em_percent
```

```
[1] 10.75017
```

82.8% of the structures are solved through X-ray while 10.75% are solved by Electron Microscopy.

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

```
sum(df[1,'total'])/sum(df$total)
```

```
[1] 0.8623852
```

> 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?

There are 2,298 HIV 1 protease structures in the current PDB. https://molstar.org/viewer/

2. Using Mol*

The main Mol* homepage at: We can input our own PDB files or just give it a PDB database accession code (4 letter PDB code)

Figure 1: Molecular View of 1HSG

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

We see one atom per water molecule because it allows for better visualization of the protein target of interest, and simplifies the overall structure.

> 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 conserved water molecule in this binding site is water-308

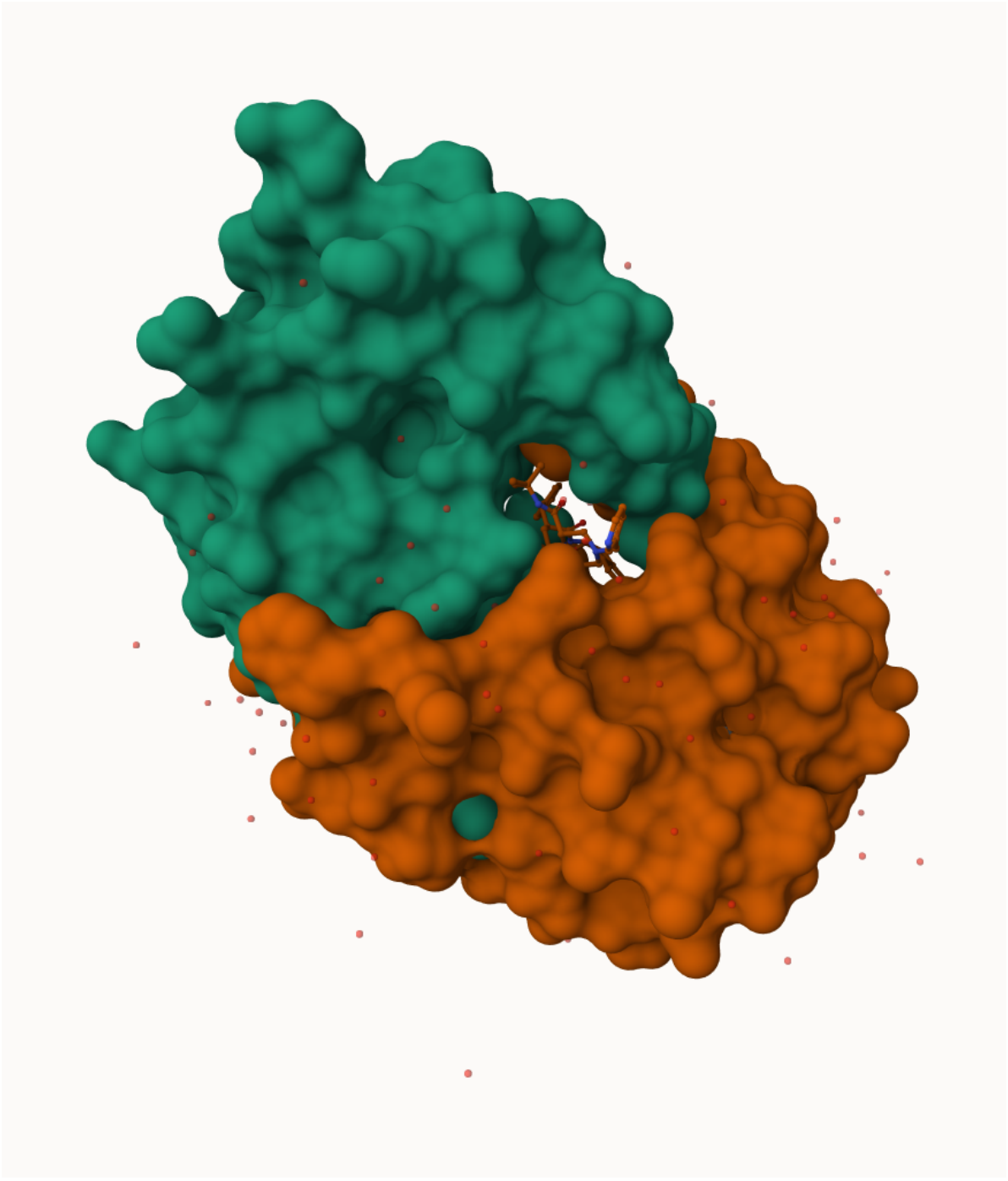


Figure 2: View of the the binding pocket with the Ligand

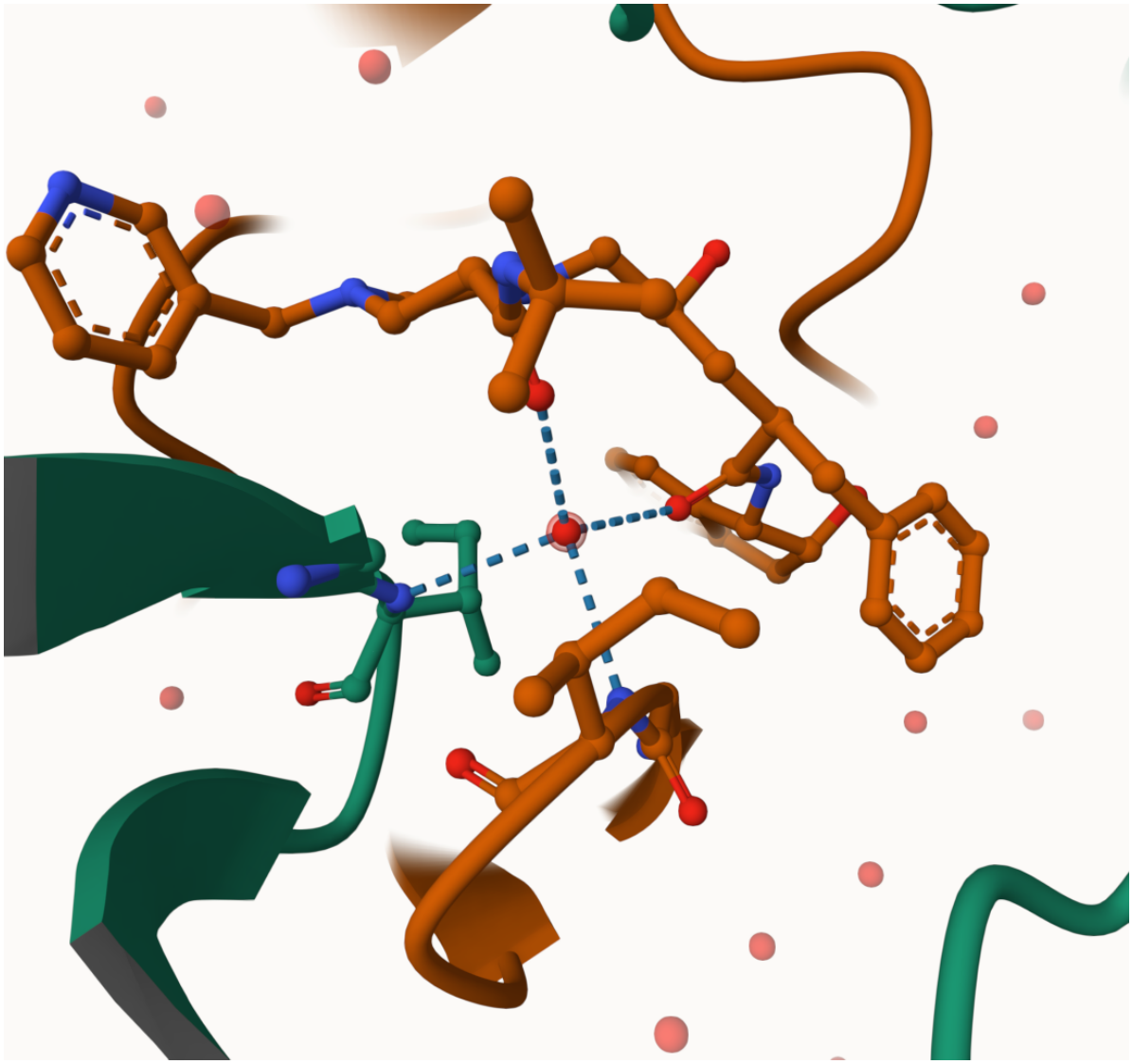


Figure 3: Interaction of water 308 and the Ligand

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.

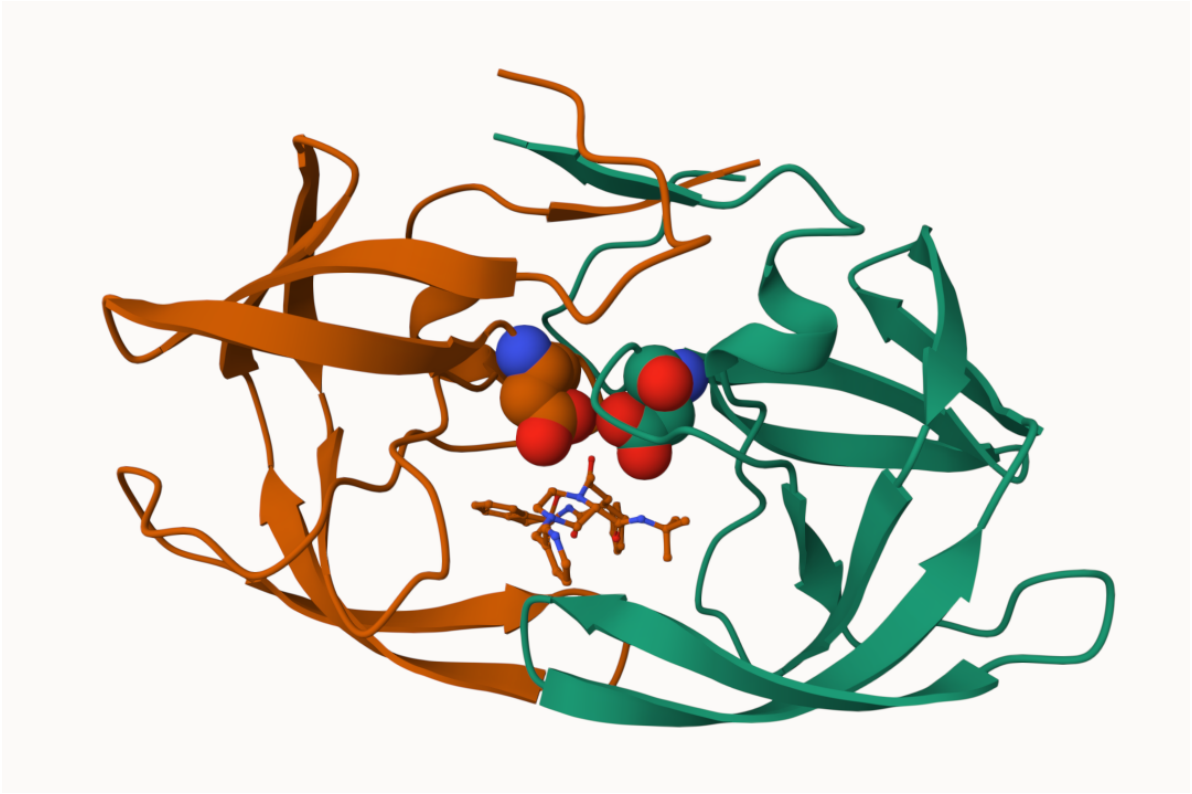


Figure 4: The important ASP 25 labeled on chain A and chain B

3. Introduction to Bio 3D 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?

There are 198 amino acid residues in this pdb object

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

```
[1] 198
```

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

MK1

Q9: How many protein chains are in this structure?

There are two chains (A and B)

```
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. It requires the **r3dmol** package that we need to install along with **shiny**

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

4. Predicting functional 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
```

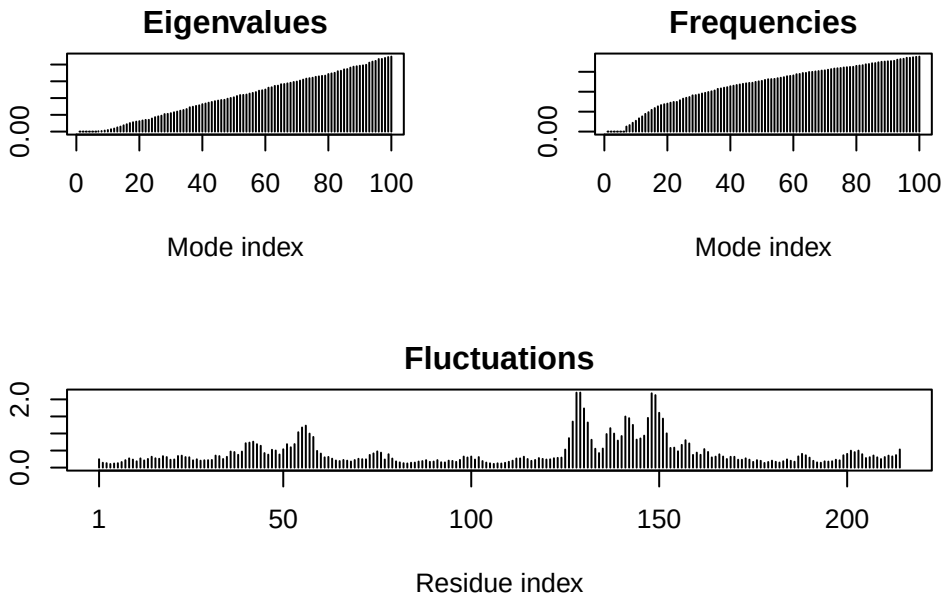
```
      VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
      YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG

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

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

```
 Building Hessian...        Done in 0.015 seconds.
 Diagonalizing Hessian...   Done in 0.299 seconds.
```

```
plot(m)
```



Write out a trajectory of the predicted molecular motion:

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

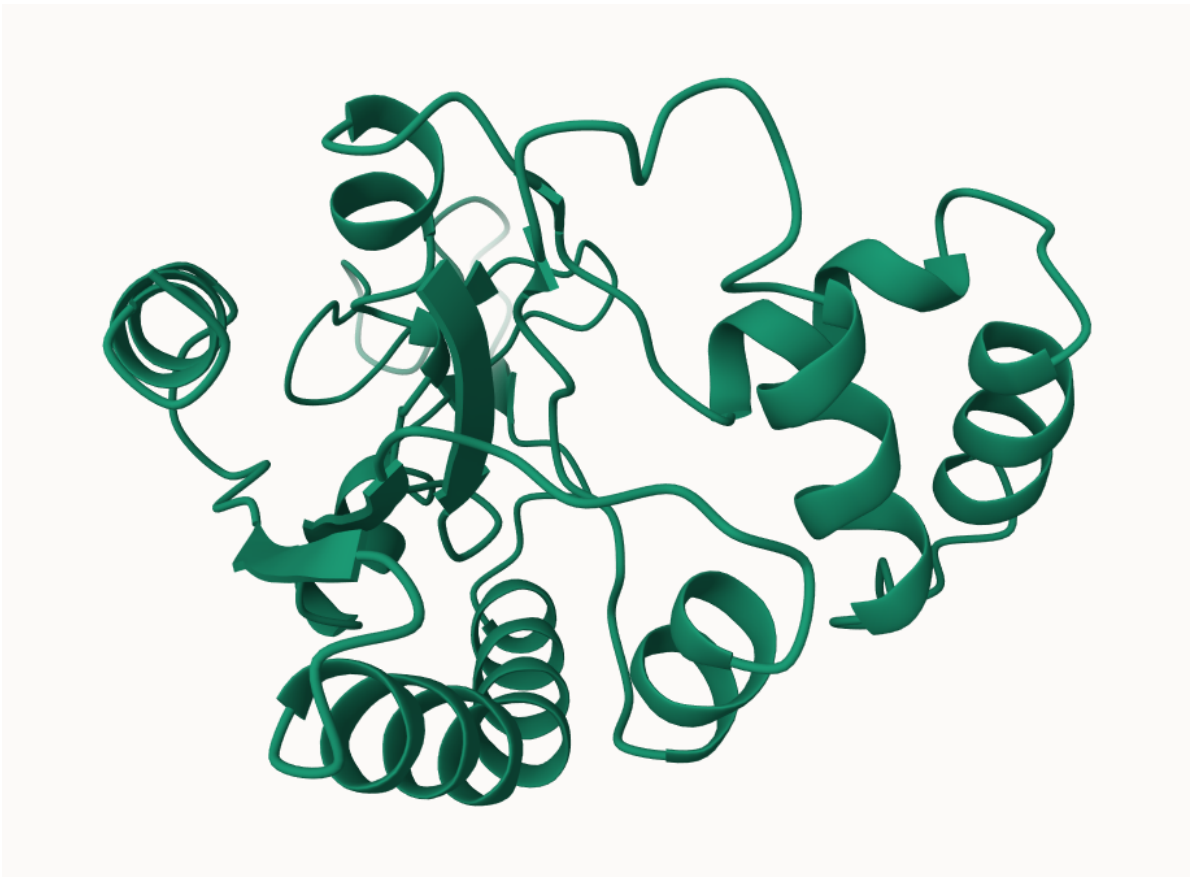


Figure 5: predicted motion of protein