

class09: structural bioinformatics

Kaitlyn Madriaga, A17217752

PDB Statistics

The PDB is the main database for structural information on biomolecules. Let's see what it contains:

```
db <- read.csv("pdb_stats.csv")
#db
```

```
knitr::kable(db)
```

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

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

```
#1) use gsub() to remove commas
#2) use as.numeric() to convert chr to numbers
#3) take the sum
sum(as.numeric(gsub(",", "", db$X.ray)))
```

```
[1] 174797
```

```
#write a function:
sum_comma <- function(x) {
  #Substitute the comma and convert to numeric
  sum(as.numeric(gsub(",", "", x)))
}
```

For Xray:

```
sum_comma(db$X.ray) / sum_comma(db$Total)
```

```
[1] 0.8553721
```

For EM:

```
round(sum_comma(db$EM) / sum_comma(db$Total), 2)
```

```
[1] 0.07
```

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

```
round(sum_comma(db$Total[1]) / sum_comma(db$Total), 2)
```

```
[1] 0.87
```

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!

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

The structure is too low to see H atoms. You need a sub 1 Angstrom resolution to see Hydrogen.

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?

HOH308

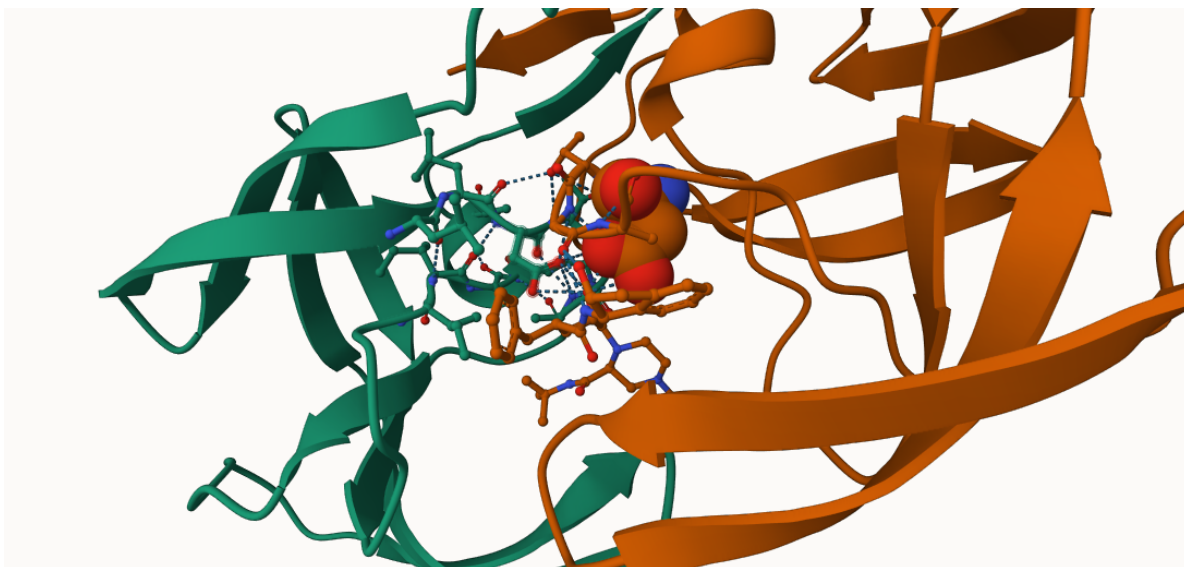


Figure 1: HIV-PR structure

Working with Structures in R

We can use the `bio3d` package to read and perform bioinformatics calculations on PDB structures.

```
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 >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>

```

Read an ADK structure

```
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:

```

MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLV
TDELVIALVKERIAQEDCRNGFLLDGFPRTPQADAMKEAGINVDYVLEFDVPDELIVDKI
VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
YYSKEAEAGNTKYAKVDGTPVAEVRADLEKILG

```

```

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

```

Perform a prediction of flexibility with a technique called normal mode analysis (NMA)

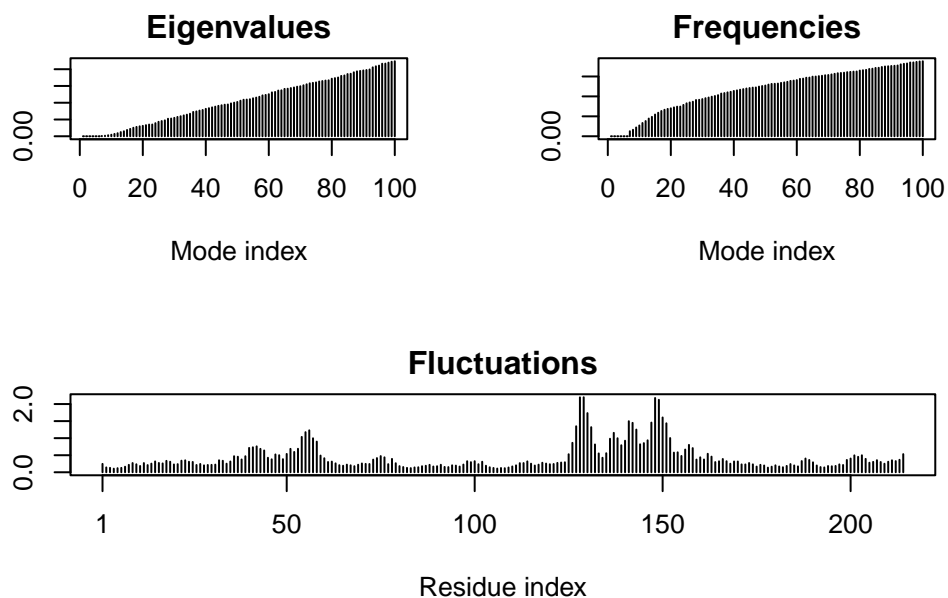
```

#Perform flexibility prediction
m <- nma(adk)

```

Building Hessian... Done in 0.049 seconds.
Diagonalizing Hessian... Done in 0.448 seconds.

```
plot(m)
```



Write out a “movie” (aka trajectory) of the motion for viewing in Molstar

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