

class10: Structural Bioinformatics (pt. 1)

Sawyer (PID: A16335277)

The PDB database

Here we examine the size and composition of the main database of biomolecular structures - the PDB.

Get a CSV file

```
pdbstats <- read.csv("pdb_stats.csv", row.names = 1)
head(pdbstats)
```

	X.ray	EM	NMR	Multiple.methods	Neutron	Other
Protein (only)	161,663	12,592	12,337	200	74	32
Protein/Oligosaccharide	9,348	2,167	34	8	2	0
Protein/NA	8,404	3,924	286	7	0	0
Nucleic acid (only)	2,758	125	1,477	14	3	1
Other	164	9	33	0	0	0
Oligosaccharide (only)	11	0	6	1	0	4
Total						
Protein (only)	186,898					
Protein/Oligosaccharide	11,559					
Protein/NA	12,621					
Nucleic acid (only)	4,378					
Other	206					
Oligosaccharide (only)	22					

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

My pdbstats data frame has numbers with commas in them. This may cause us problems. Let's see:

```
pdbstats$X.ray
```

```
[1] "161,663" "9,348"  "8,404"  "2,758"  "164"    "11"
```

```
as.numeric(pdbstats$X.ray)
```

Warning: NAs introduced by coercion

```
[1] NA NA NA NA 164 11
```

```
x <- "22,200"  
as.numeric(x) + 1
```

Warning: NAs introduced by coercion

```
[1] NA
```

We found a function called `gsub()` now we can figure out how it works

```
x <- "22,200"  
as.numeric(gsub(",", "", x))
```

```
[1] 22200
```

I can turn this snippet into a function that I can use for every column in the table.

```
commasum <- function(x) {  
  sum(as.numeric(gsub(",", "", x)))  
}  
  
commasum(pdbstats$X.ray)
```

```
[1] 182348
```

Apply across all columns

```
totals <- apply(pdbstats, 2, commasum)
```

```
round(totals/totals["Total"] * 100, 2)
```

X-ray	EM	NMR	Multiple.methods
84.54	8.72	6.57	0.11
Neutron	Other	Total	
0.04	0.02	100.00	

84.54% of the structures are solved using X-ray and 8.72% of the structures are solved using Electron Microscopy.

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

```
as.numeric(gsub(",", "", pdbstats[1, 'Total']))/commasum(pdbstats[, 'Total'])
```

```
[1] 0.8665362
```

0.87 of all structures are protein.

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?

5 structures showed up on the database under the HIV-1 search keyword.

Q. What fraction of all molecule structures are stored in the PDB database?

```
(215684/249751891 * 100)
```

```
[1] 0.08635931
```

2. Visualizing Protein Structure

We will learn the basics of Mol* (mol-star) homepage: <https://molstar.org/viewer/>

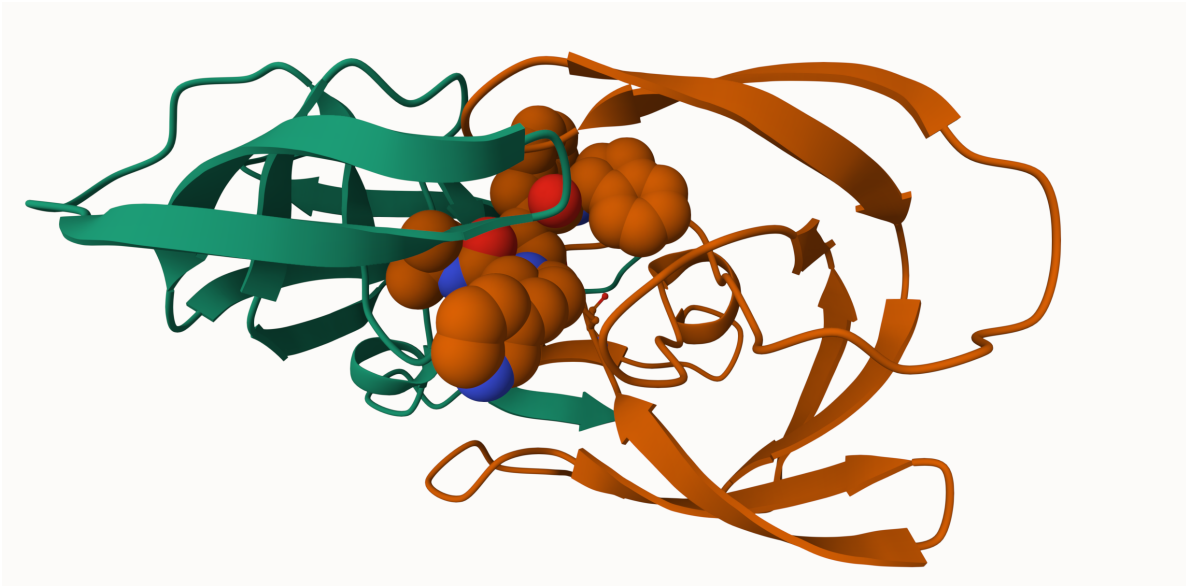
We will play with PDB code 1HSG

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

Hydrogen atoms are too small to be picked up by the limited resolution of the measurement.

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 has 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.

Discussion Topic: Can you think of a way in which indinavir, or even larger ligands and substrates, could enter the binding site?

It's possible that if the flaps surrounding the binding site open then larger ligands or substrates could enter the site.

Show the ASP 25 amino acids:

Back to R and working with PDB structures

Predict the dynamics (flexibility) of an important protein:

```
library(bio3d)

hiv <- read.pdb("1hsg")
```

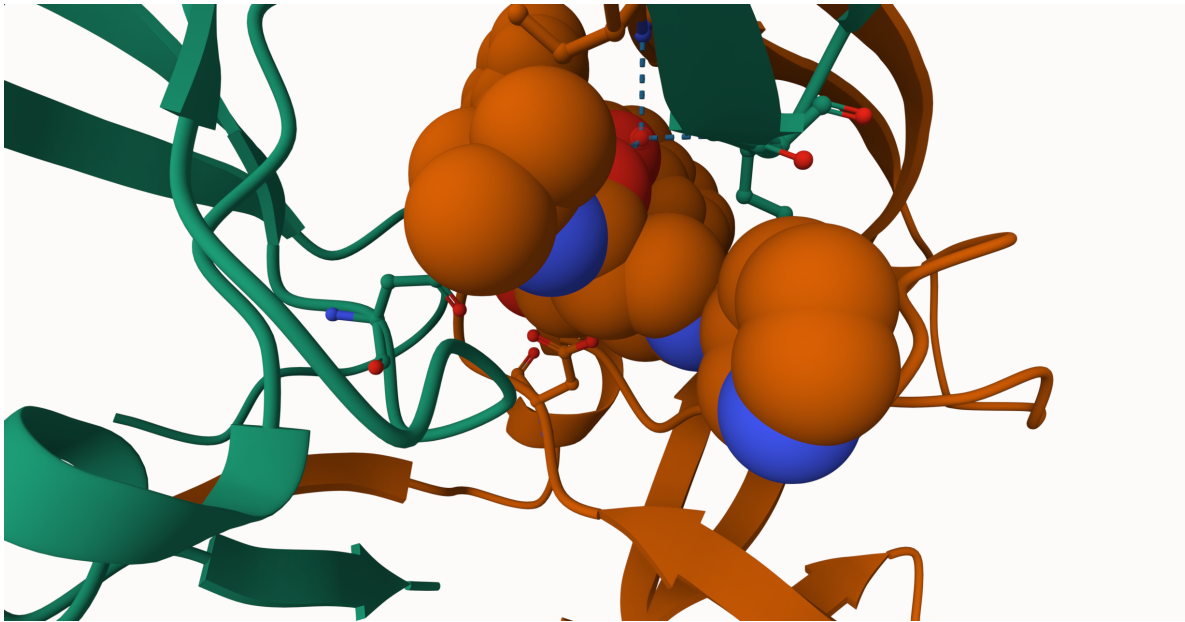


Figure 1: HIV-Pr with a bound inhibitor showing the two important ASP 25 amino acids

Note: Accessing on-line PDB file

```
hiv
```

```
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.

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

HOH is one of the two non-protein residues.

Q9: How many protein chains are in this structure?

There are 2 protein chains in this structure.

```
head(hiv$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>

```
pdbseq(hiv)
```

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
"P"	"Q"	"I"	"T"	"L"	"W"	"Q"	"R"	"P"	"L"	"V"	"T"	"I"	"K"	"I"	"G"	"G"	"Q"	"L"	"K"
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
"E"	"A"	"L"	"L"	"D"	"T"	"G"	"A"	"D"	"D"	"T"	"V"	"L"	"E"	"E"	"M"	"S"	"L"	"P"	"G"
41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60
"R"	"W"	"K"	"P"	"K"	"M"	"I"	"G"	"G"	"I"	"G"	"G"	"F"	"I"	"K"	"V"	"R"	"Q"	"Y"	"D"
61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80

```

"Q" "I" "L" "I" "E" "I" "C" "G" "H" "K" "A" "I" "G" "T" "V" "L" "V" "G" "P" "T"
81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 1
"P" "V" "N" "I" "I" "G" "R" "N" "L" "L" "T" "Q" "I" "G" "C" "T" "L" "N" "F" "P"
2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21
"Q" "I" "T" "L" "W" "Q" "R" "P" "L" "V" "T" "I" "K" "I" "G" "G" "Q" "L" "K" "E"
22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
"A" "L" "L" "D" "T" "G" "A" "D" "D" "T" "V" "L" "E" "E" "M" "S" "L" "P" "G" "R"
42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61
"W" "K" "P" "K" "M" "I" "G" "G" "I" "G" "G" "F" "I" "K" "V" "R" "Q" "Y" "D" "Q"
62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81
"I" "L" "I" "E" "I" "C" "G" "H" "K" "A" "I" "G" "T" "V" "L" "V" "G" "P" "T" "P"
82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99
"V" "N" "I" "I" "G" "R" "N" "L" "L" "T" "Q" "I" "G" "C" "T" "L" "N" "F"

```

Here we will do a Normal Mode Analysis (NMA) to predict functional motions of a kinase protein.

```
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:
```

```

MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMRLRAAVKSGSELGKQAKDIMDAGKLV
TDELVIALVKERIAQEDCRNGFLLDGFPRTPQADAMKEAGINVDYVLEFDVPDELIVDKI
VGRRVHAPSGRVYHVKFNPKEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG

```

YYSKEAEAGNTKYAKVDGTPVAEVRADLEKILG

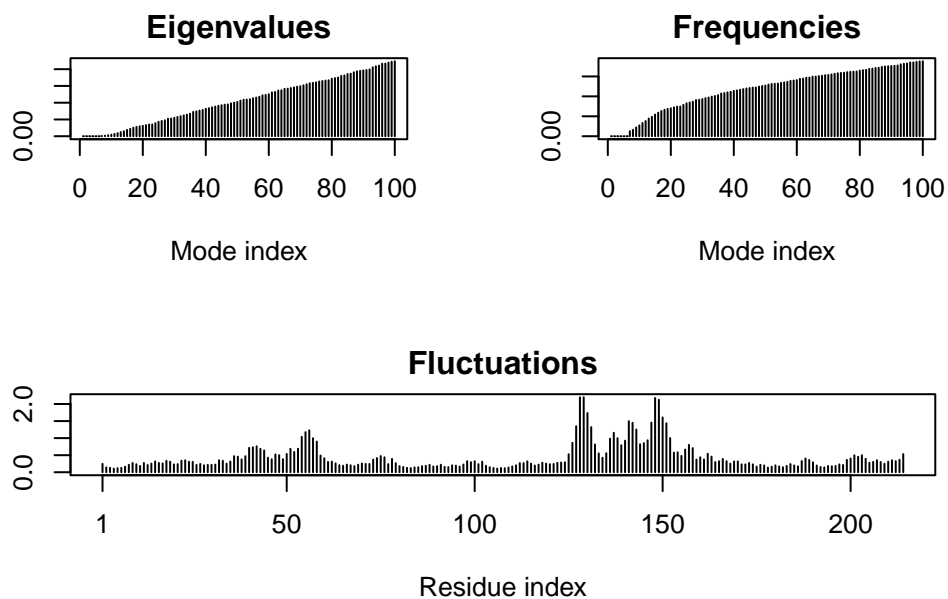
```
+ attr: atom, xyz, seqres, helix, sheet,  
      calpha, remark, call
```

```
modes <- nma(adk)
```

Building Hessian... Done in 0.03 seconds.

Diagonalizing Hessian... Done in 0.33 seconds.

```
plot(modes)
```



Make a “movie” called a trajectory of the predicted motions:

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

Then I can open this file in Mol*....