# Class 9: Structural Bioinformatics (pt.1)

Sofia Lanaspa, A17105313

The main database for strucutal data on proteins is PDB(protein data book), let see what it contain:

Data from: https://www.rcsb.org/stats

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

```
pdbdb$total
```

```
NULL
```

```
#issue because numbers are in quotes due to the coma
as.numeric(sub(",","",pdbdb$Total)) #convert values to numeric to do math
```

```
[1] 195610  12318  13720   4531    213     22
```

Could turn the code above into a function

```
comma2numeric <- function(x){
  as.numeric(sub(",","",x))
}
comma2numeric((pdbdb$X.ray)) #test function
```

```
[1] 167192   9639   8730   2869    170     11
```

```
apply(pdbdb,2,comma2numeric)
```

```
Warning in FUN(newX[, i], ...): NAs introduced by coercion
```

```
     Molecular.Type  X.ray    EM   NMR Multiple.methods Neutron Other  Total
[1,]             NA 167192 15572 12529              208      77    32 195610
[2,]             NA   9639  2635    34                8       2     0  12318
[3,]             NA   8730  4697   286                7       0     0  13720
[4,]             NA   2869   137  1507               14       3     1   4531
[5,]             NA    170    10    33                0       0     0    213
[6,]             NA     11     0     6                1       0     4     22
```

OR TRY DIFFERENT FUNCTION

```
library(readr)
pdbdb <- 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.
```

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

```
#Sum of all x-rays divided the Total
(sum(pdbdb$"X-ray")/sum(pdbdb$Total)) * 100
```

```
[1] 83.30359
```

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

```
#First value of column 'Total' is only protein, divided by all the values in 'Total'
(pdbdb$Total[1]/ sum(pdbdb$Total)) * 100
```

```
[1] 86.39483
```

**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?**

```
#Look up HIV on website, number of results = 4,553
```

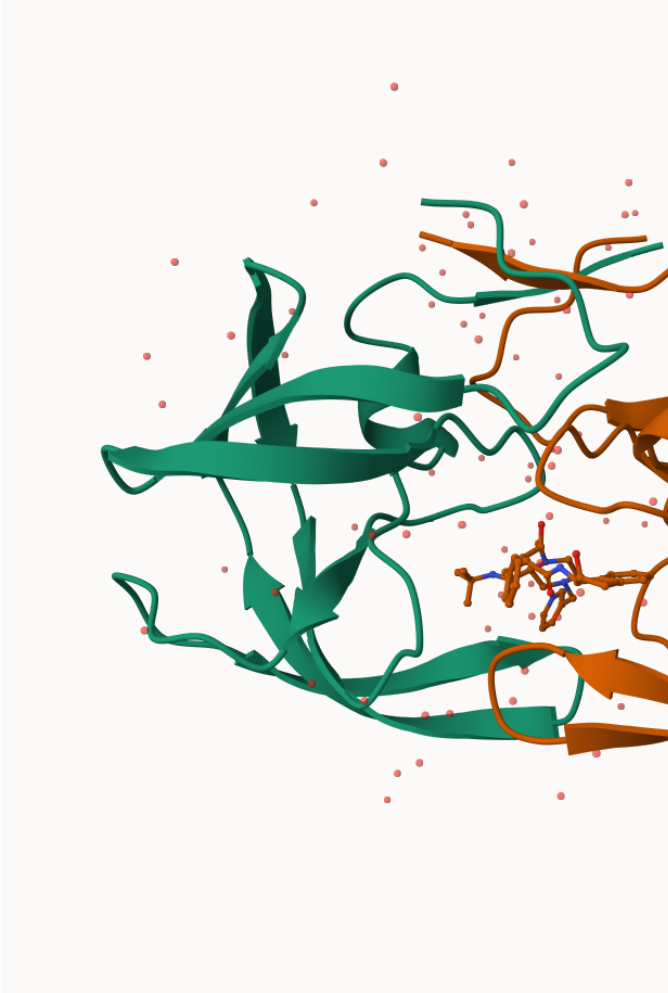
**Q4: Water molecules normally have 3 atoms. Why do we see just one atom per water molecule in this structure?**

```
# because the diagram focuses on the central oxygen atom
```

**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?**

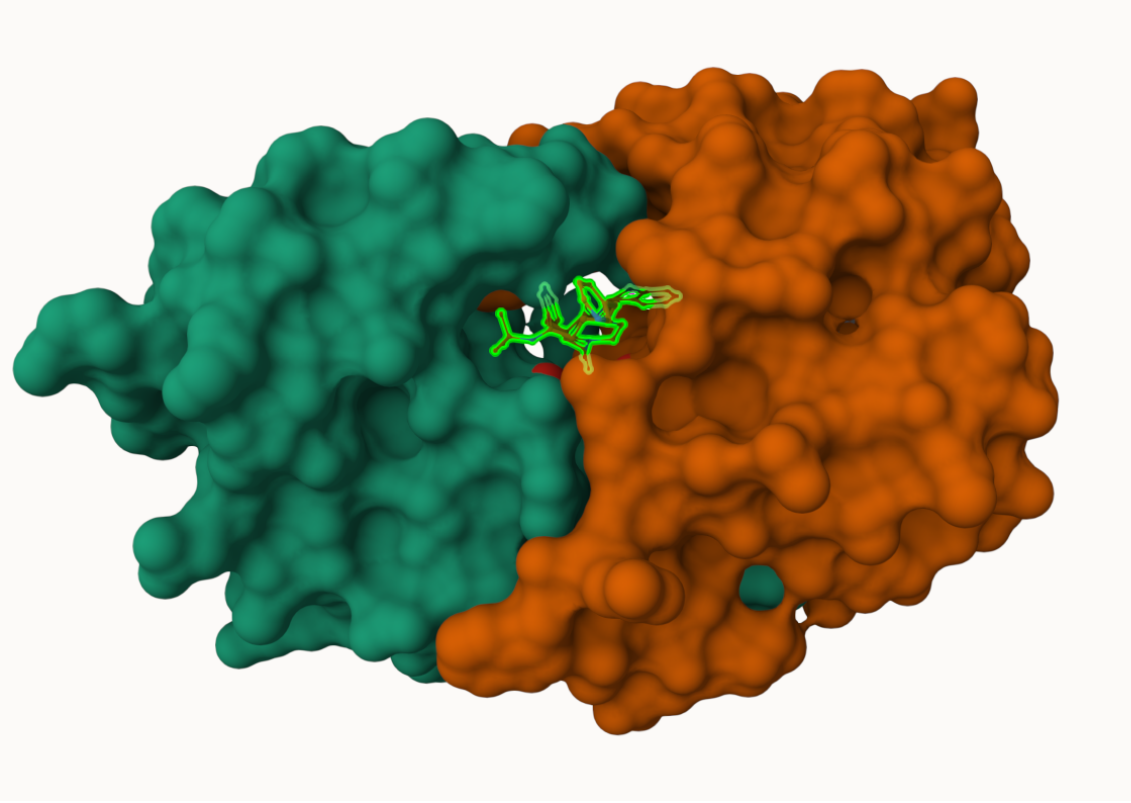
**Q7: [Optional] As you have hopefully observed HIV protease is a homodimer (i.e. it is composed of two identical chains). With the aid of the graphic display can you identify secondary structure elements that are likely to only form in the dimer rather than the monomer?**



Import image from molstar (https://molstar.org/viewer/):

Steps to edit this protein structure:

click arrow on right side - select D25 (aspartate) - 3D box - set representation to ‘spacefill’ - do the same for chain B to visualize it in a more realistic way, and see where the ligand binds: on right hand side click components - add - selection = ‘polymer’ - representation = ‘molecular sur-



face' - add component

## Bio3D

The Bio3D pachage allows us to do all sorts of structural bioinformatics work in R.

Let's start with how it can read PDB files:

```
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
```

```
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>
```

```
pdbseq(pdb)
```

```
  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"
```

```
pdbseq(pdb)[25] #number gives you amino acid at that position
```

```
 25
"D"
```

**Q7: How many amino acid residues are there in this pdb object?**

```
sum(pdb$calpha) #there is 1 calpha per AA so count calpha's
```

```
[1] 198
```

```
length(pdb)
```

```
[1] 8
```

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

```
#HOH, MK1
```

**Q9: How many protein chains are in this structure?**

```
# 2 chains
```

```
adk <- read.pdb("3s36")
```

```
  Note: Accessing on-line PDB file
```

```
adk
```

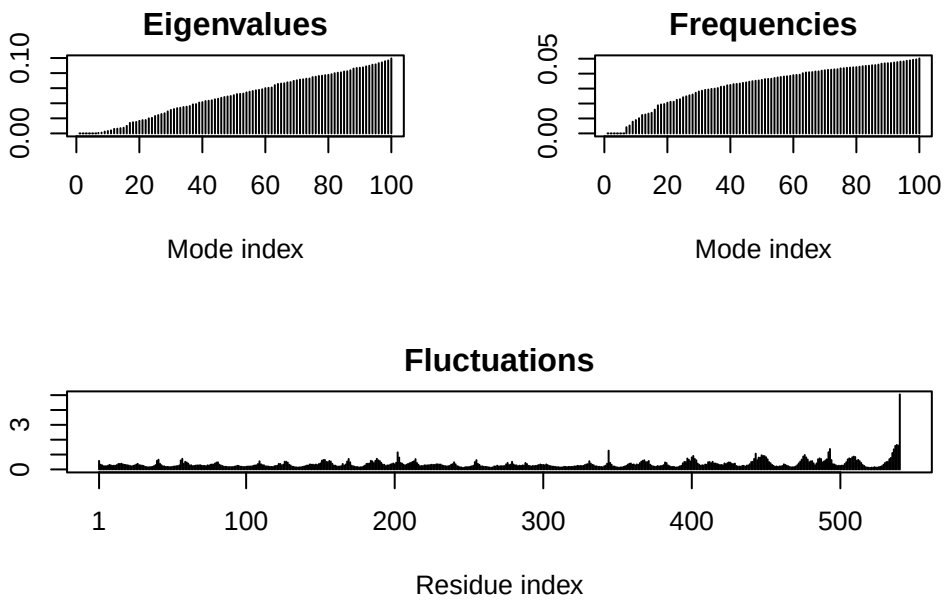
```
 Call:  read.pdb(file = "3s36")

   Total Models#: 1
     Total Atoms#: 4104,  XYZs#: 12312  Chains#: 3  (values: L H X)

     Protein Atoms#: 4104  (residues/Calpha atoms#: 540)
     Nucleic acid Atoms#: 0  (residues/phosphate atoms#: 0)

     Non-protein/nucleic Atoms#: 0  (residues: 0)
     Non-protein/nucleic resid values: [ none ]

   Protein sequence:
      DIQMTQSPSSVSASIGDRVTITCRASQGIDNWLGWYQQKPGKAPKLLIYDASNLDTGVPS
      RFSGSGSGTYFTLTISSLQAEDFAVYFCQQAKAFPPTFGGGTKVDIKRTVAAPSVFIFPP
      SDEQLKSGTASVVCLLNNFYPREAKVQWKVDNALQSGNSQESVTEQDSKDSTYSLSSTLT
      LSKADYEKHKVYACEVTHQGLSSPVTKSFNRGECVQLVQSGGGLV...<cut>...KPFV

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

```
# Perform fleibility prediction
m <- nma(adk)
```

```
Warning in nma.pdb(adk): Possible multi-chain structure or missing in-structure residue(s) pr
  Fluctuations at neighboring positions may be affected.
```

```
 Building Hessian...        Done in 0.058 seconds.
 Diagonalizing Hessian...   Done in 3.573 seconds.
```

```
plot(m)
```




Write out multi-model PDB file (trajectory) that we can use to make an animation or the predicted motions

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

I can open this in Mol* to play the trajectory...