Class09

Parnaz Boroon PID:A13557370

The PDB Database

The main respository for biomolecular structure data is the Protein Data Bank (PDB).

Let's have a quick look at the composition of this database:

QUESTION 1: What percentage of structures in the PDB are solved by X-Ray and Electron Microscopy?

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

stats$X.ray <- as.numeric(gsub(",", "", stats$X.ray))
stats$EM <- as.numeric(gsub(",", "", stats$EM))
stats$Total <- as.numeric(gsub(",", "", stats$Total))

total_xray <- sum(stats$X.ray, na.rm = TRUE)
total_em <- sum(stats$EM, na.rm = TRUE)
total_all <- sum(stats$Total, na.rm = TRUE)

percent_xray <- (total_xray / total_all) * 100
percent_em <- (total_em / total_all) * 100

cat("Percentage solved by X-Ray:", round(percent_xray, 2), "%\n")
```

```
Percentage solved by X-Ray: 81.43 %
```

```
cat("Percentage solved by Electron Microscopy:", round(percent_em, 2), "%\n")
```

```
Percentage solved by Electron Microscopy: 12.27 %
```

This is annoying lets try a different import function from the **readr** package.

```
library("readr")

stats <- read_csv("Data Export Summary.csv")
```

```
Rows: 6 Columns: 9
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): Molecular Type
dbl (4): Integrative, 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.
```

```
n.total <- sum(stats$Total)
n.xray <- sum(stats$`X-ray`)
n.em <- sum(stats$EM)

round(n.xray/n.total * 100,3)
```

```
[1] 81.432
```

```
round(n.em/n.total * 100,3)
```

```
[1] 12.271
```

QUESTION 2: What proportion of structures in the PDB are protein?

```
stats <- read_csv("Data Export Summary.csv")
```

```
Rows: 6 Columns: 9
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (1): Molecular Type
dbl (4): Integrative, 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.
```

```
n.prot <- sum(stats$Total[1])
row.total <- sum(stats$Total)
round((n.prot/row.total)*100, digits = 2)
```

```
[1] 86.05
```

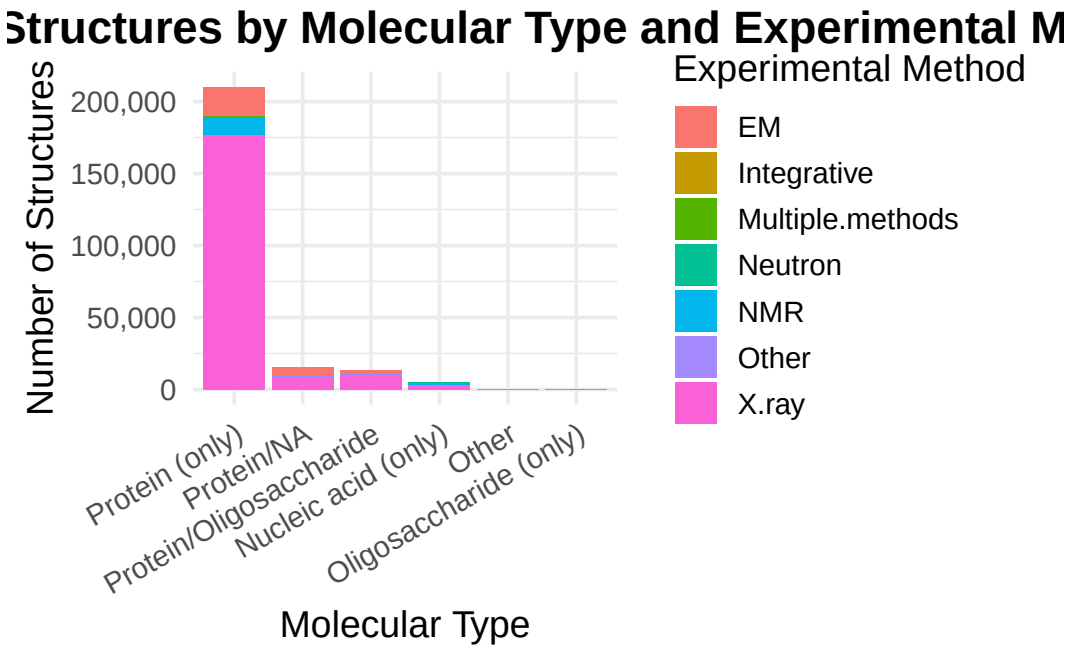
QUESTION 3: Make a bar plot overview

```
library(ggplot2)
library(tidyr)
library(dplyr)
```

```
Attaching package: 'dplyr'

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

    filter, lag

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

    intersect, setdiff, setequal, union
```

```
# Read CSV
my_data <- read.csv("Data Export Summary.csv")

# Clean numeric columns (matching your actual names)
cols_to_fix <- c("X.ray", "EM", "NMR", "Integrative", "Multiple.methods", "Neutron", "Other"
my_data[cols_to_fix] <- lapply(my_data[cols_to_fix], function(x) as.numeric(gsub(",", "", x))

# Sort Molecular.Type by total number of structures (descending)
my_data <- my_data %>%
  arrange(desc(Total)) %>%
  mutate(Molecular.Type = factor(Molecular.Type, levels = Molecular.Type))
```

```r
# Reshape from wide → long
long_data <- my_data %>%
  pivot_longer(
    cols = c("X.ray", "EM", "NMR", "Integrative", "Multiple.methods", "Neutron", "Other"),
    names_to = "Method",
    values_to = "Count"
  )

# Create stacked bar plot
ggplot(long_data, aes(x = Molecular.Type, y = Count, fill = Method)) +
  geom_bar(stat = "identity") +
  scale_y_continuous(labels = scales::comma) +
  labs(
    title = "Overview of PDB Structures by Molecular Type and Experimental Method",
    x = "Molecular Type",
    y = "Number of Structures",
    fill = "Experimental Method"
  ) +
  theme_minimal(base_size = 14) +
  theme(
    plot.title = element_text(face = "bold", hjust = 0.5),
    axis.text.x = element_text(angle = 30, hjust = 1),
    legend.position = "right"
  )
```



Visualizing structure data

The Mol* viewer is embedded in many bioinformatics websites.

I can insert any figure or image file using markdown format

Figure 1: HIV-Pr dimer with bound inhibitor

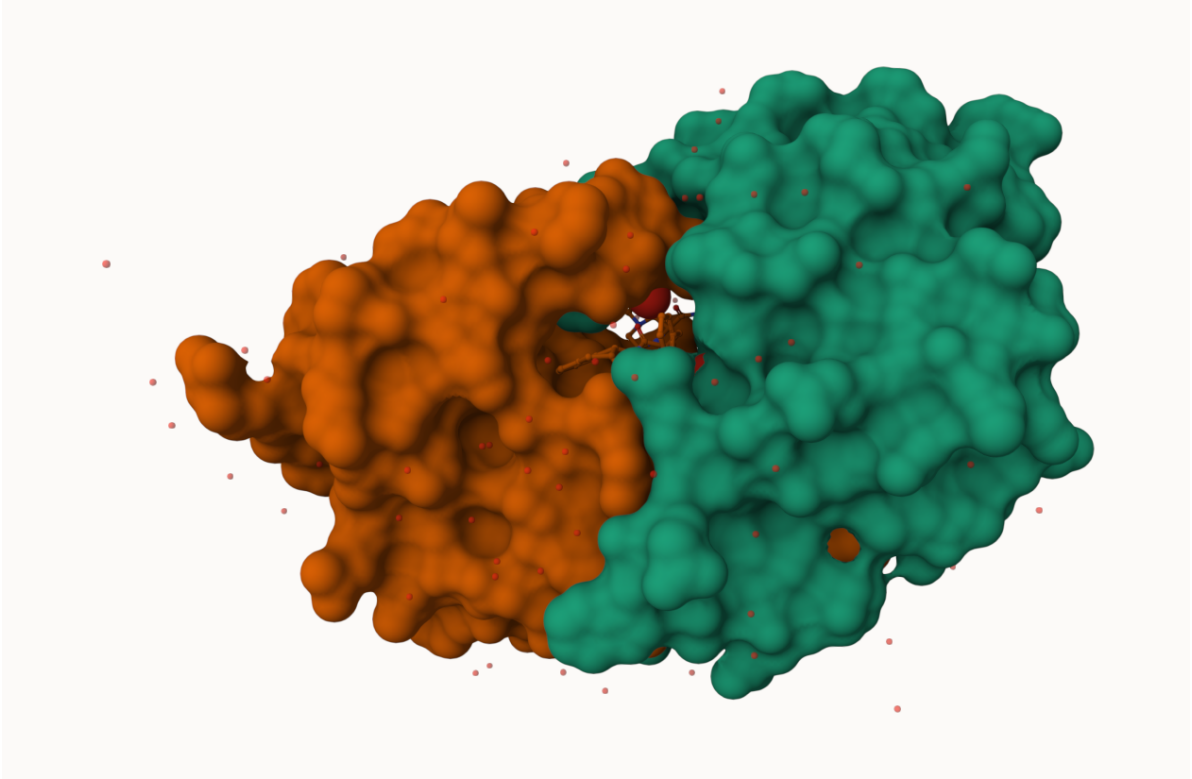
QUESTION 4: Water molecules normally have 3 atoms. Why do we see just one atom per water molecule in this structure?

If there were more than just one atom per water molecule in this structure, we wouldn't be able to see

QUESTION 5: 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?

H308

QUESTION 6: 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.



Bio3D package for structural bioinformatics

We can use the bio3d package to read and analyze biomolecular data in R:

```
library(bio3d)

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

```
  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
```

```
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"
```

Let's trim to chain A and get just it's sequence:

```
chainA <- trim.pdb(hiv, chain="A")
chainA.seq <- pdbseq(chainA)
```

QUESTION 7 How many amino acid residues are there in this pdb object?

2 ## QUESTION 8 Name one of the two non-protein residues? HOH (127) ## QUESTION 9 How many protein chains are in this structure? 2 [A, B]

Prediction of functional motions

We can run a Normal mode analysis (NMA) to predict large scale motions/flexibility/dynamics of any biomolecule that can read into R.

Let's look at ADK and chain A only!

```
adk <- read.pdb("1ake")
```

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

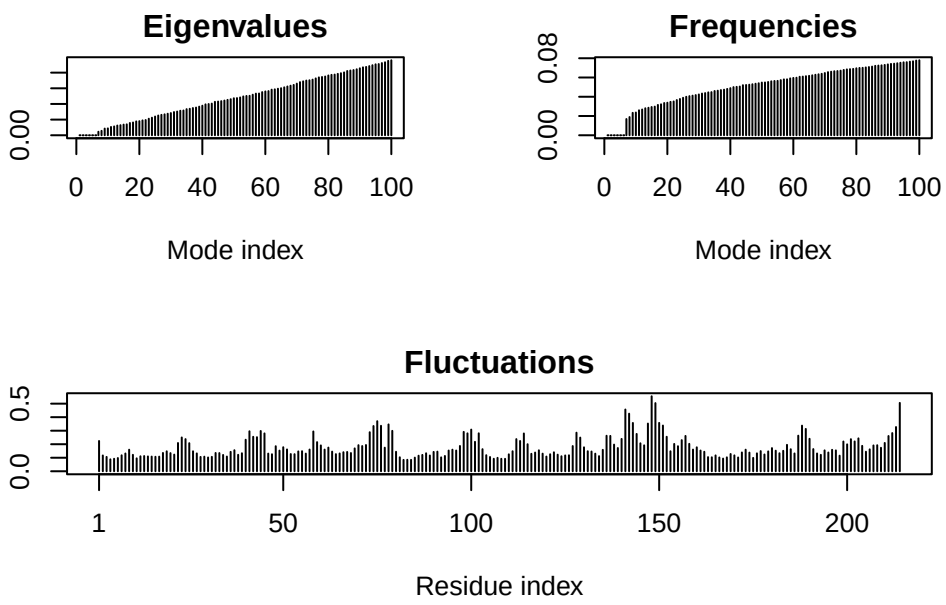
```
adk_A <- trim.pdb (adk, chain="A")
adk_A
```

```
 Call:  trim.pdb(pdb = adk, chain = "A")

   Total Models#: 1
     Total Atoms#: 1954,  XYZs#: 5862  Chains#: 1  (values: A)

     Protein Atoms#: 1656  (residues/Calpha atoms#: 214)
     Nucleic acid Atoms#: 0  (residues/phosphate atoms#: 0)

     Non-protein/nucleic Atoms#: 298  (residues: 242)
     Non-protein/nucleic resid values: [ AP5 (1), HOH (241) ]

   Protein sequence:
      MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
      DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDRI
      VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
      YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG

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

```
m <- nma(adk_A)
```

```
 Building Hessian...        Done in 0.012 seconds.
 Diagonalizing Hessian...   Done in 0.271 seconds.
```

```
plot(m)
```



Let's write out a "trajectory" of prediction

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

Play with 3D viewing in R

We can use the new **bio3dview** package, which is not yet on CRAN, to render interactive 3D views in R and HTML quarto output reports.

To install from GitHub we can use the **pak** package.

QUESTION 10: Which of the packages above is found only on BioConductor and not CRAN?

Biocmanager ## QUESTION 11: Which of the above packages is not found on BioConductor or CRAN? devtools ## QUESTION 12: True or False? Functions from the devtools package can be used to install packages from GitHub and BitBucket? True ## QUESTION 13: How many amino acids are in this sequence, i.e. how long is this sequence? 214