Class10

Juan Gonzalez (PID: A69036681

#Introduction to the RCSB Protein

Download a CSV file from the PDB site (accessible from "Analyze" > "PDB Statistics" > "by Experimental Method and Molecular Type". Move this CSV file into your RStudio project and use it to answer the following questions:

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

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

```
xray_total <- sum(as.numeric(gsub(",", "", data$X.ray)))
em_total <- sum(as.numeric(gsub(",", "", data$EM)))
total_structures <- sum(as.numeric(gsub(",", "", data$Total)))

xray_percentage <- (xray_total / total_structures) * 100
em_percentage <- (em_total / total_structures) * 100

xray_percentage
```

```
[1] 83.25592
```

```
em_percentage
```

```
[1] 10.2348
```

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

```
protein_total <- as.numeric(gsub(",", "", data[data$Molecular.Type == "Protein (only)", "Tota

protein_proportion <- (protein_total / total_structures) * 100

protein_proportion
```

```
[1] 86.3961
```

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?

I receive 4,563 structures when I input HIV in the PDB website search box.

#Visualizing the HIV-1 protease structure Mol* (pronounced "molstar") is a new web-based molecular viewer that is rapidly gaining in popularity and utility. At the time of writing it is still a long way from having the full feature set of stand-alone molecular viewer programs like VMD, PyMol or Chimera. However, it is gaining new features all the time and does not require any download or complicated installation.

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

There could be x-ray limitations in detecting water molecules or unless it is implied the one atom means a water molecule.

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

I see HOH 308 inside the protein, which I assume is the binding site.

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.

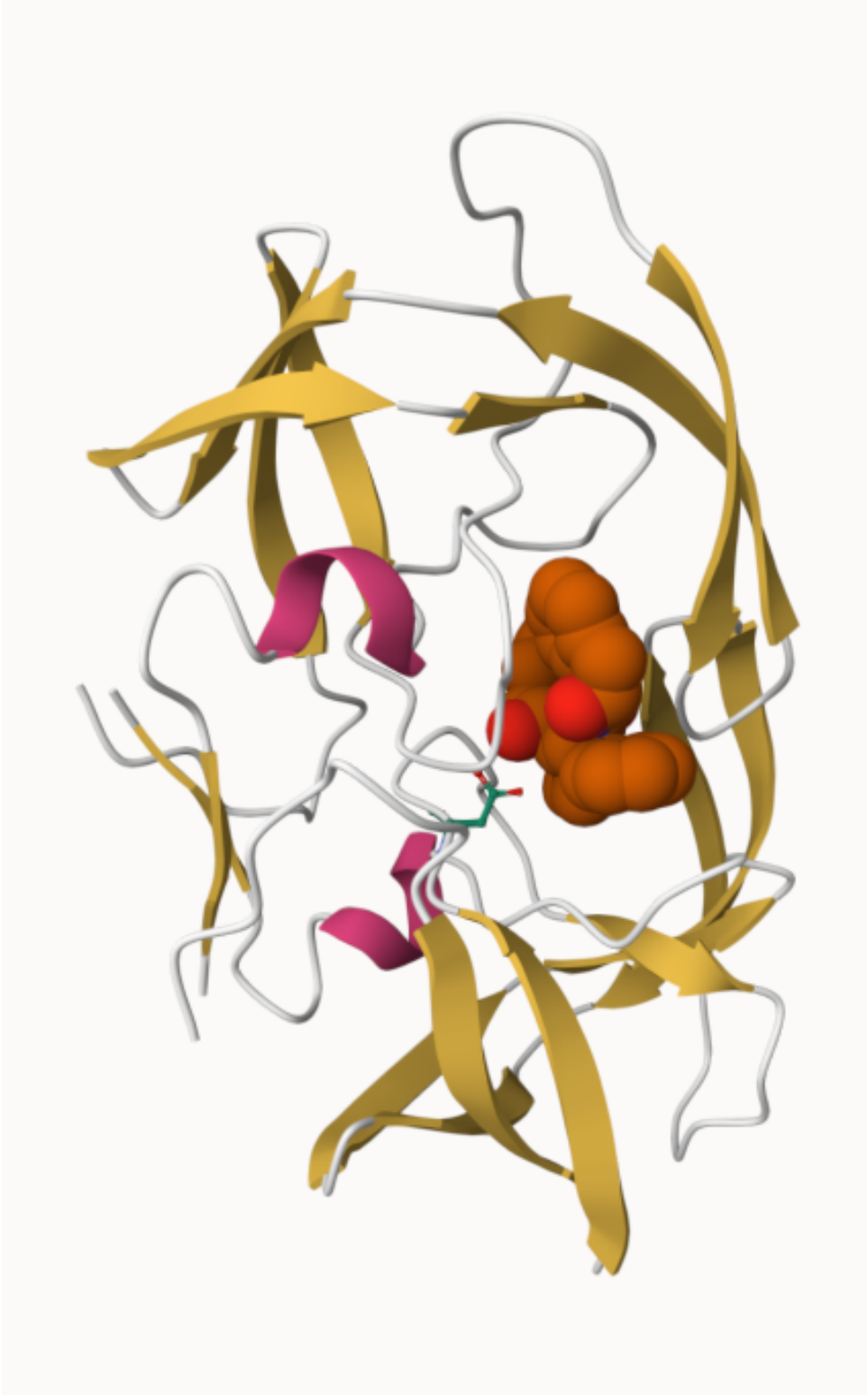

Figure 1: HIV-1 protease structure with ligand and catalytic residues

#Introduction to Bio3D in R

Bio3D is an R package for structural bioinformatics. Features include the ability to read, write and analyze biomolecular structure, sequence and dynamic trajectory data.

```
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 protein chains

```
attributes(pdb)
```

```
$names
[1] "atom"   "xyz"    "seqres" "helix"  "sheet"  "calpha" "remark" "call"

$class
[1] "pdb" "sse"
```

To access these individual attributes we use the dollar-attribute name convention that is common with R list objects. For example, to access the atom attribute or component use pdb$atom:

```
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>
```

Predicting functional motions of a single structure Let's read a new PDB structure of Adenylate Kinase and perform Normal mode analysis.

```
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
```

Normal mode analysis (NMA) is a structural bioinformatics method to predict protein flexibility and potential functional motions (a.k.a. conformational changes).

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

```
 Building Hessian...        Done in 0.01 seconds.
 Diagonalizing Hessian...   Done in 0.222 seconds.
```

```
#plot(m)
```

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

#Comparative structure analysis of Adenylate Kinase

The goal of this section is to perform principal component analysis (PCA) on the complete collection of Adenylate kinase structures in the protein data-bank (PDB).

Adenylate kinase (often called simply Adk) is a ubiquitous enzyme that functions to maintain the equilibrium between cytoplasmic nucleotides essential for many cellular processes. Adk operates by catalyzing the reversible transfer of a phosphoryl group from ATP to AMP. This reaction requires a rate limiting conformational transition (i.e. change in shape). Here we analyze all currently available Adk structures in the PDB to reveal detailed features and mechanistic principles of these essential shape changing transitions.

Q10. Which of the packages above is found only on BioConductor and not CRAN?

msa is only found in bioconductor

Q11. Which of the above packages is not found on BioConductor or CRAN? bio3d-view is not found in both, but being installed in Bitbucket.

Q12. True or False? Functions from the devtools package can be used to install packages from GitHub and BitBucket? True!

Below we perform a blast search of the PDB database to identify related structures to our query Adenylate kinase (ADK) sequence. In this particular example we use function get.seq() to fetch the query sequence for chain A of the PDB ID 1AKE and use this as input to blast.pdb(). Note that get.seq() would also allow the corresponding UniProt identifier.

```
library(bio3d)
aa <- get.seq("1ake_A")
```

```
Warning in get.seq("1ake_A"): Removing existing file: seqs.fasta
```

```
Fetching... Please wait. Done.
```

```
aa
```

```
             1        .         .         .         .         .         60
pdb|1AKE|A   MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
             1        .         .         .         .         .         60

            61        .         .         .         .         .         120
pdb|1AKE|A   DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDRI
            61        .         .         .         .         .         120

           121        .         .         .         .         .         180
pdb|1AKE|A   VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
           121        .         .         .         .         .         180

           181        .         .         .   214
pdb|1AKE|A   YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG
           181        .         .         .   214

Call:
  read.fasta(file = outfile)
```

```
Class:
  fasta

Alignment dimensions:
  1 sequence rows; 214 position columns (214 non-gap, 0 gap)

+ attr: id, ali, call
```

Q13. How many amino acids are in this sequence, i.e. how long is this sequence?
214

```
# Blast or hmmer search
#b <- blast.pdb(aa)
```

```
# Plot a summary of search results
#hits <- plot(b)
```

```
# List out some 'top hits'
#head(hits$pdb.id)
```

```
#hits <- NULL
#hits$pdb.id <- c('1AKE_A','6S36_A','6RZE_A','3HPR_A','1E4V_A','5EJE_A','1E4Y_A','3X2S_A','6
```

```
# Download releated PDB files
#files <- get.pdb(hits$pdb.id, path="pdbs", split=TRUE, gzip=TRUE)
```

```
# Align releated PDBs
#pdbs <- pdbaln(files, fit = TRUE, exefile="msa")
```

```
# Vector containing PDB codes for figure axis
#ids <- basename.pdb(pdbs$id)

# Draw schematic alignment
#plot(pdbs, labels=ids)
```

#Annotate collected PDB structures

```
#anno <- pdb.annotate(ids)
#unique(anno$source)
```

```
#pc.xray <- pca(pdbs)
#plot(pc.xray)
```

```
# Calculate RMSD
#rd <- rmsd(pdbs)

# Structure-based clustering
#hc.rd <- hclust(dist(rd))
#grps.rd <- cutree(hc.rd, k=3)

#plot(pc.xray, 1:2, col="grey50", bg=grps.rd, pch=21, cex=1)
```