class09

Eric Garcia Ordunez (A16316409)

PDB Statistics

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:

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

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

Molecular.Type	X.ray	EM	NMR	Multiple.methods	Neutron	Other	Total
Protein (only)	152,809	9,421	12,117	191	72	32	174,642
Protein/Oligosaccharide	9,008	1,654	32	7	1	0	10,702
Protein/NA	8,061	2,944	281	6	0	0	11,292
Nucleic acid (only)	2,602	77	1,433	12	2	1	4,127
Other	163	9	31	0	0	0	203
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.

```
(db$X.ray)
```

```
[1] "152,809" "9,008"   "8,061"   "2,602"   "163"     "11"
```

```
gsub(",", "", db$X.ray)
```

```
[1] "152809" "9008"   "8061"   "2602"   "163"    "11"
```

```
xray.total <- sum(as.numeric(gsub(",", "", db$X.ray)))
em.total <- sum(as.numeric(gsub(",", "", db$EM)))
```

Can i write a function instead of repeating the same line of code?

```
  # I will work with `x` as input.
sum_comma <- function(x) {
    # Substitute the comma and convert to numeric
    sum(as.numeric(gsub(",","", x)))
}
```

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

```
[1] 172654
```

```
sum_comma(db$EM)
```

```
[1] 14105
```

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

```
[1] 0.8590264
```

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!

Figure 1: HIV-PR structure from MERK with a bound drug

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 a resolution 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

Working With a Structure 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
```

```
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)
```

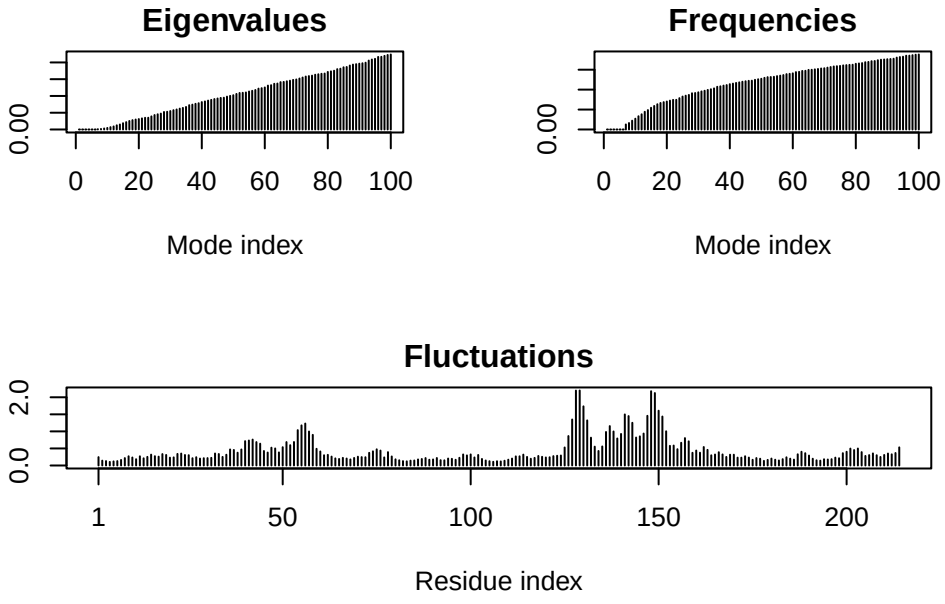
```
     Non-protein/nucleic Atoms#: 244  (residues: 244)
     Non-protein/nucleic resid values: [ CL (3), HOH (238), MG (2), NA (1) ]

   Protein sequence:
      MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
      DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDKI
      VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
      YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG

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

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

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

```
Building Hessian...        Done in 0.043 seconds.
Diagonalizing Hessian...   Done in 0.4 seconds.
```

```
plot(m)
```



Write out a “movie”(a.k.a. trajectory) of the motion for a viewing in MOlstar

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

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

Answer: 198

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

Answer: MK1

Q9: How many protein chains are in this structure?

Answer: 2