Class 10: Structural Bioinformatics

Erica Sanchez (A15787505)

What is in the PDB database?

The main repository of biomolecular structure info is in the PDB <www.rcsb.org>

Let"s see what this database contains:

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

	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

We have to get rid of the commas. Can you find a function to get rid of the commas?

```
#gsub(",","", x)
```

```
x <- stats$X.ray
as.numeric(gsub(",","", x))
```

```
[1] 161663   9348   8404   2758    164     11
```

```
x <- stats$Total
sum(as.numeric(gsub(",","", x)))
```

```
[1] 215684
```

I am going to turn this into a function and then use `apply()` to work on the entire table of data.

```
sumcomma <- function(x){
  sum(as.numeric(gsub(",","", x)))
}

sumcomma(stats$X.ray)
```

```
[1] 182348
```

```
apply(stats, 2, sumcomma) / sumcomma(stats$Total)
```

```
        X.ray                EM               NMR Multiple.methods
0.8454405519     0.0872433746     0.0657118748     0.0010663749
     Neutron             Other             Total
0.0003662766     0.0001715473     1.0000000000
```

```
n.total <- sumcomma(stats$Total)
n.total
```

```
[1] 215684
```

```
sumcomma(stats$EM)
```

```
[1] 18817
```

```
apply(stats, 2, sumcomma)
```

```
      X.ray         EM        NMR Multiple.methods
     182348      18817      14173              230
    Neutron      Other      Total
         79         37     215684
```

```
apply(stats, 2, sumcomma) / sumcomma(stats$Total)
```

```
      X.ray         EM        NMR Multiple.methods
0.8454405519 0.0872433746 0.0657118748 0.0010663749
    Neutron      Other      Total
0.0003662766 0.0001715473 1.0000000000
```

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

X.ray = 85% EM = 9%

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

```
(186898/248805733)*100
```

```
[1] 0.07511804
```

visualizing the HIV-1 protease structure

Mol* ("mol-star") viewer is now everywhere.The homepage is here: https://molstar.org/viewer/

I want to insert my image from Mol* here.

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

A Hydrogen molecule has a resolution that is too small, so it can't be seen.

Figure 1: My first molecular image

Figure 2: My second molecular image

Working with bio3d

```
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
```

```
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)[25]
```

```
 25
"D"
```

Predicting functional motions of a single structure

We can do a bioinformatics prediction of functional motions (i.e, flexibility/dynamics):

```
pdb <- read.pdb("6s36")
```

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

```
pdb
```

```

 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) ]
```

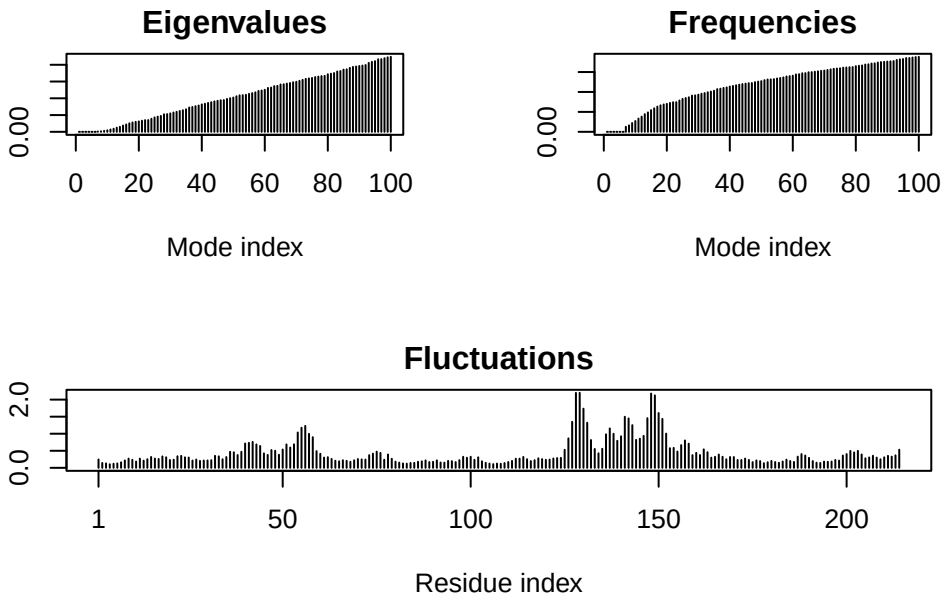
```
   Protein sequence:
      MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
      DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDKI
      VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
      YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG

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

```
m <- nma(pdb)
```

```
 Building Hessian...        Done in 0.013 seconds.
 Diagonalizing Hessian...   Done in 0.257 seconds.
```

```
plot(m)
```




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