CLass10_q

AUTHOR
Dhruv Khanduja

Complete question 1 to 3

1. Read CSV

```r
pdbstats <- read.csv("Class10Q1_3.csv")
```

The table is in characters (commas), we need it to be numerical. We can change this to numeric using `AS.NUMERIC()`

```r
x <- pdbstats$Total
#as.numeric(x)
gsub(',', '',x) #g stands for global
```

```
[1] "195866" "12328"  "13746"  "4532"   "213"    "22"
```

We see an error. We now know that the comma is the problem. How do we get rid of commas.

```r
convert_comma_numbers <- function(x) {
  x <- gsub(',', '',x)
  x<- as.numeric(x)
  return(x)
}
```

How to use this function. We can cbind new column, or apply the function too.

```r
convert_comma_numbers(pdbstats$Total)
```

```
[1] 195866  12328  13746   4532    213     22
```

Lets apply this now

```r
apply(pdbstats, 2, convert_comma_numbers)
```

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

```
     Molecular.Type  X.ray    EM   NMR Multiple.methods Neutron
Other  Total
[1,]             NA 167317 15698 12534              208      77
32 195866
[2,]             NA   9645  2639    34                8       2
0  12328
[3,]             NA   8735  4718   286                7       0
0  13746
[4,]             NA   2869   138  1507               14       3
1   4532
[5,]             NA    170    10    33                0       0
0    213
[6,]             NA     11     0     6                1       0
4     22
```

Total structures in the database

```
n.tot <- sum(convert_comma_numbers(pdbstats$Total))
```

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

```
(colSums(apply(pdbstats, 2, convert_comma_numbers))/ n.
```

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

```
  Molecular.Type            X.ray               EM
NMR
              NA      83.25592064      10.23479646
6.35181093
Multiple.methods          Neutron            Other
Total
      0.10498132       0.03617003       0.01632063
100.00000000
```

A.1: Xray = 83.25%; EM = 10.234%

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

A2. 86.3% of the structures are protein only. 97.8% include protein.

Can also read csv differently

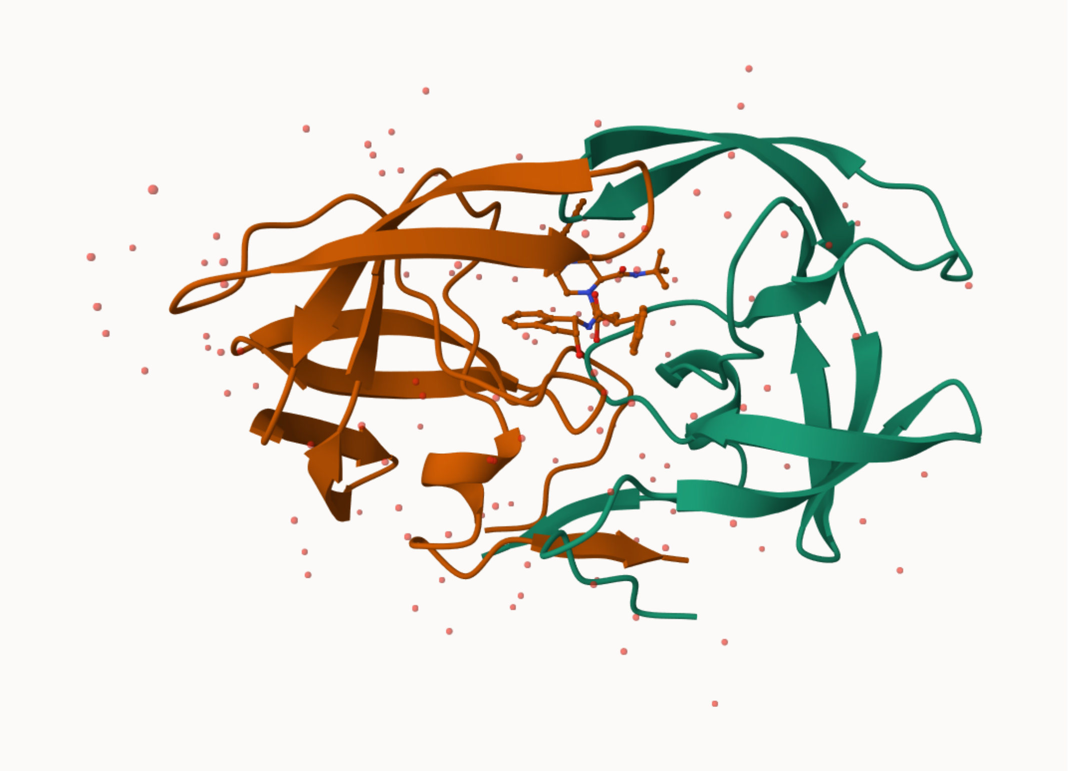
```
library(readr)
pdb_stats <- read_csv("Class10Q1_3.csv")
```

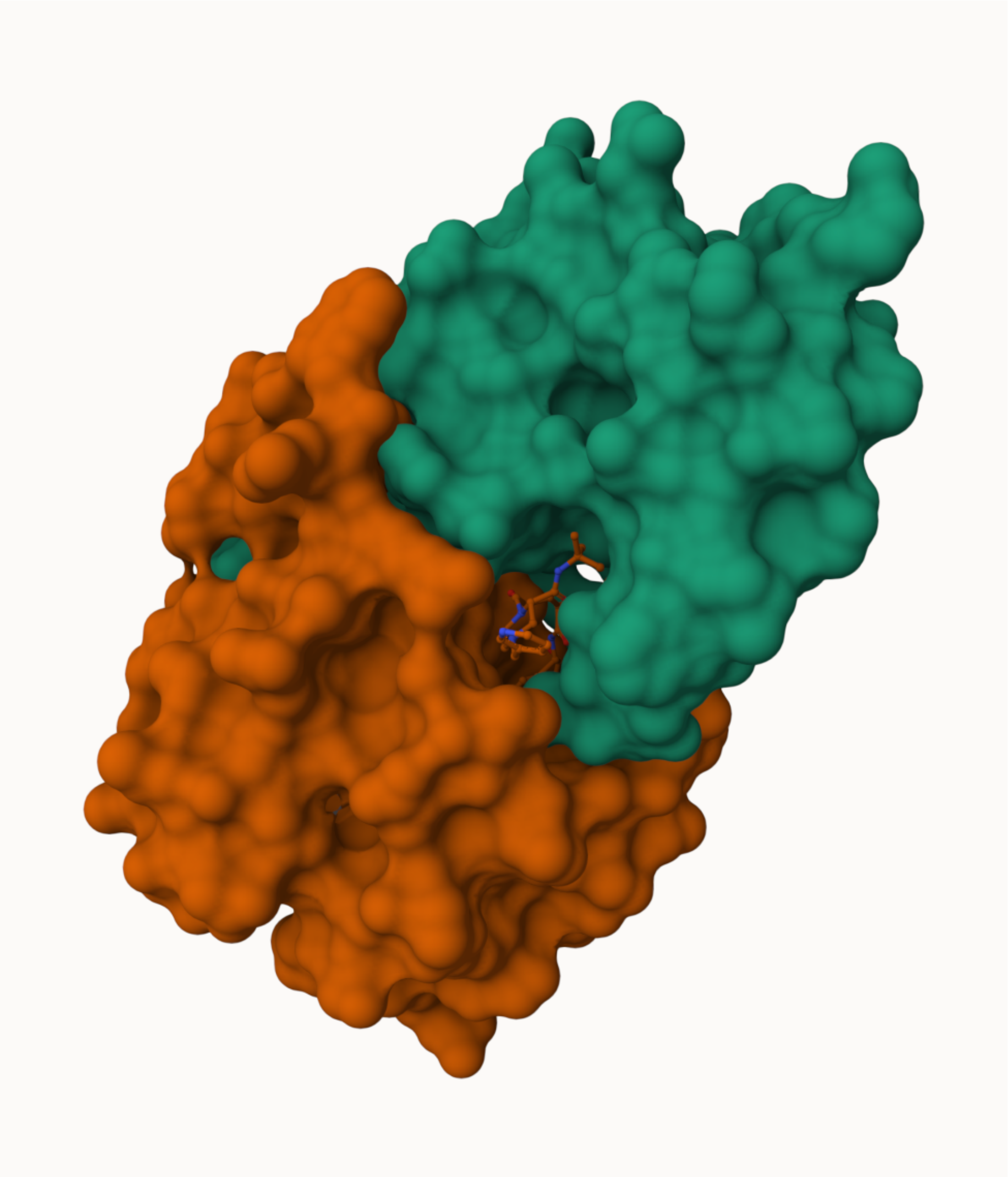
```
Rows: 6 Columns: 8
── Column specification ────────────────────────────────────────────────────────
Delimiter: ","
chr (1): Molecular Type
dbl (3): Multiple methods, Neutron, Other
num (4): X-ray, EM, NMR, Total

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.
```

##Using Mol*

How to add picture to my document in Markdown





Showing A25 and water residue near ligand

Bio3D package for structural bioinformatics

```
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
```

Look at this in summary

```
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>
```

```
#can also look at the sequence
pdbseq(pdb)[25]
```

```
 25
"D"
```

> Q. How many amino acids are in this structure?

```
length(pdbseq(pdb))
```

```
[1] 198
```

Functional dynamics prediction

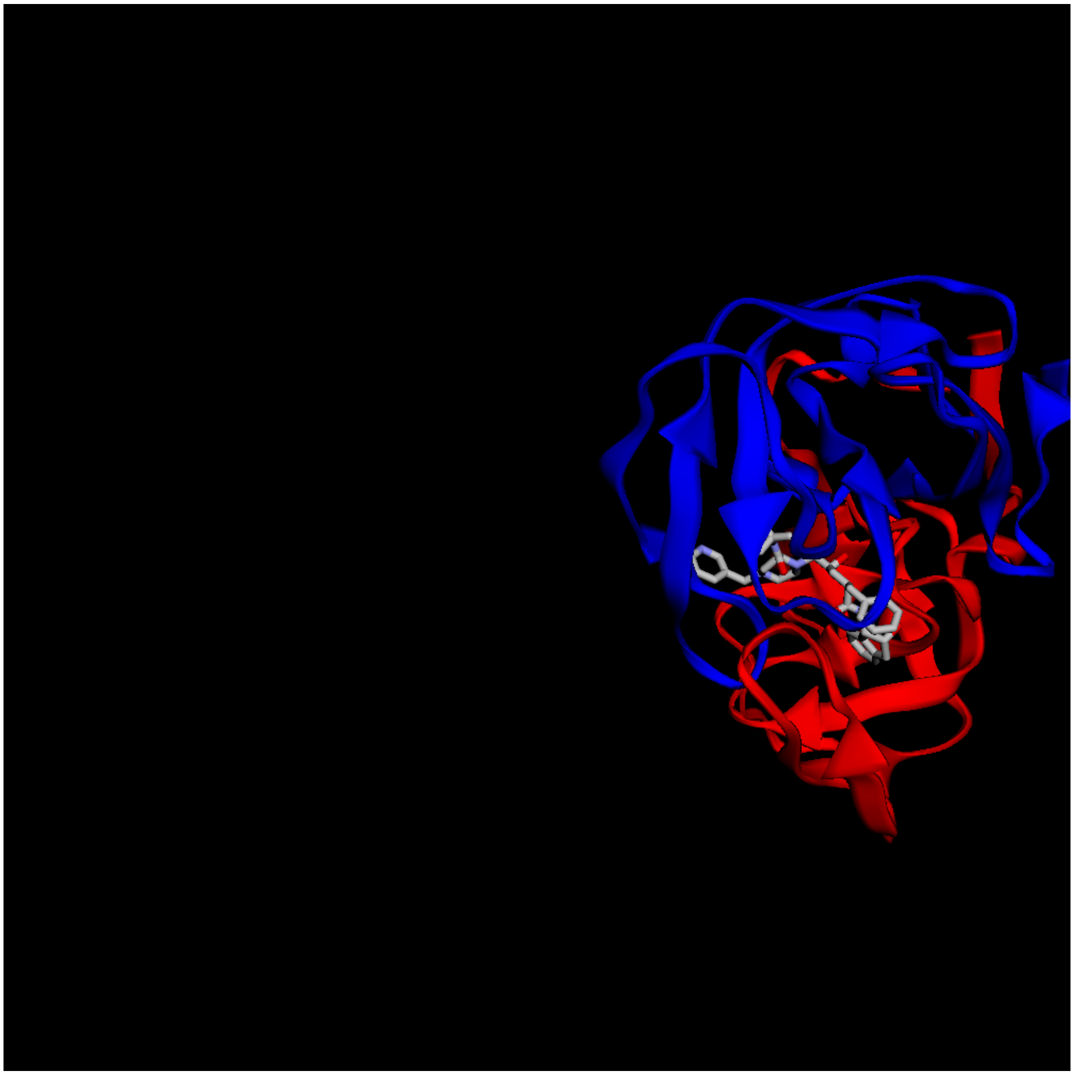
Predicting functional motions of a single structure

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

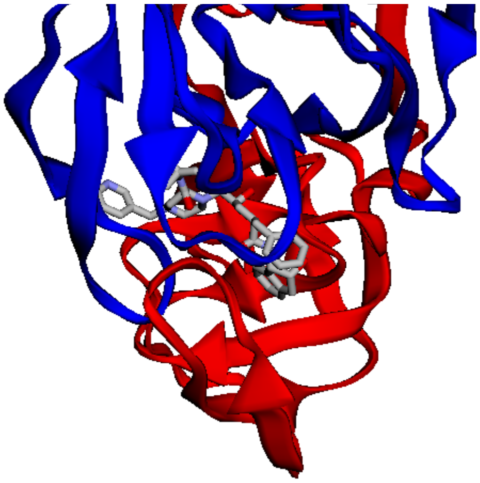
```
  Note: Accessing on-line PDB file
   PDB has ALT records, taking A only, rm.alt=TRUE
```

From slides from Barry

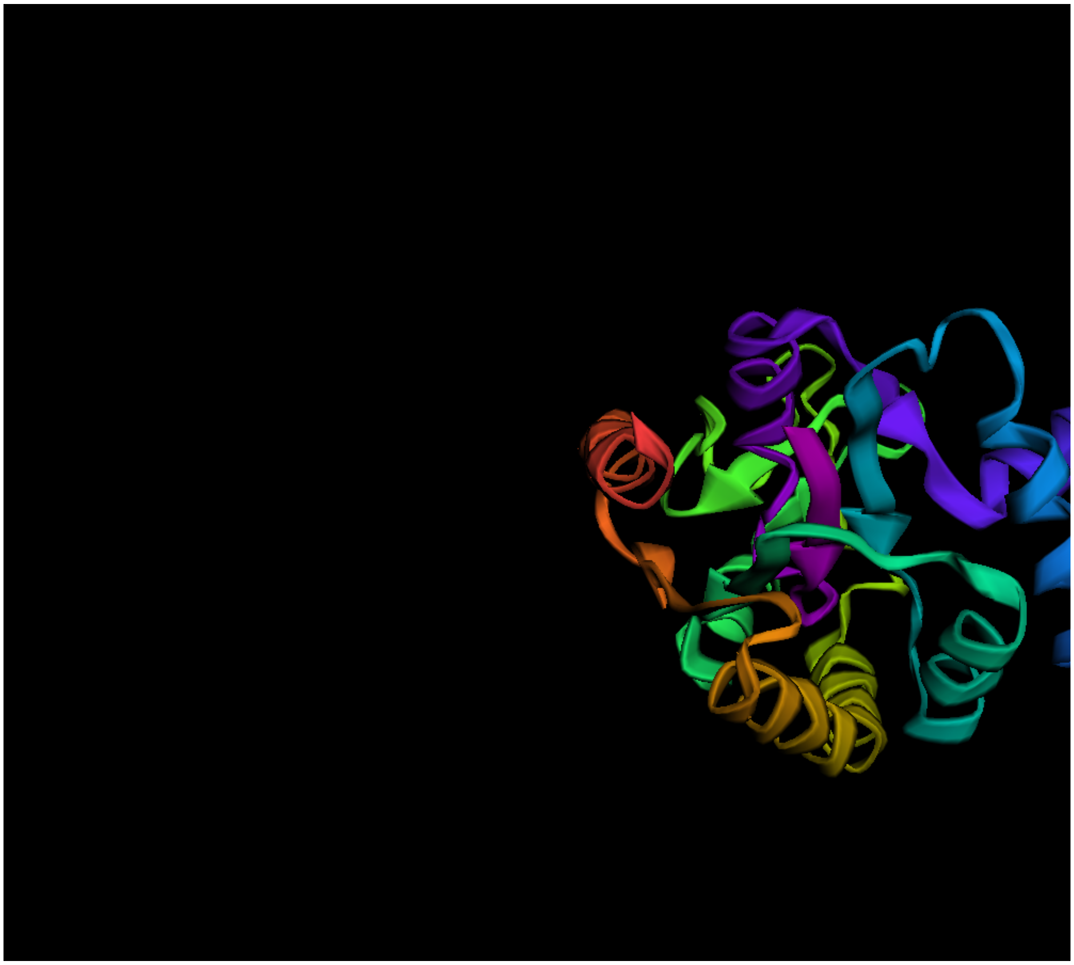
```r
source("https://tinyurl.com/viewpdb")
library(r3dmol)
# shiny
view.pdb(pdb)
```



```r
view.pdb(pdb, backgroundColor = "white")
```



```
view.pdb(adk)
```

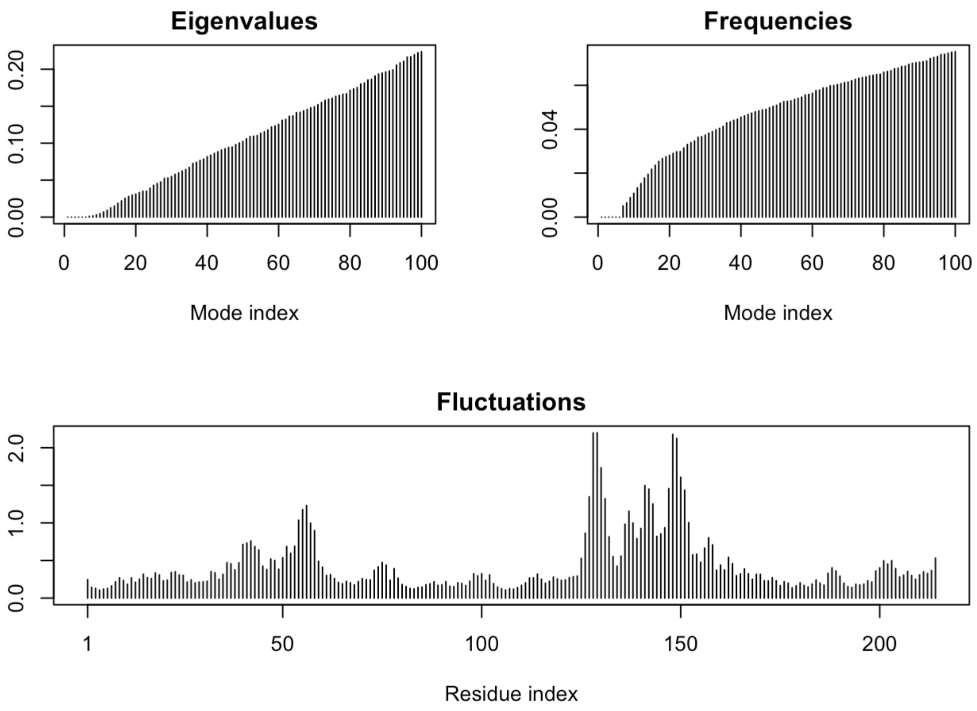


[redacted]

```r
modes <- nma(adk)
```

```
 Building Hessian...        Done in 0.016 seconds.
 Diagonalizing Hessian...   Done in 0.363 seconds.
```

```r
plot(modes)
```



All together

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

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

```
Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
/var/folders/lc/sgvmgnjn49q1m8n1ggfj8gz00000gn/T//RtmpE7VWz7/6s
36.pdb exists.
Skipping download
```

```
   PDB has ALT records, taking A only, rm.alt=TRUE
```

```
modes <- nma(adk)
```

```
Building Hessian...        Done in 0.016 seconds.
Diagonalizing Hessian...   Done in 0.36 seconds.
```

```
mktrj(modes, file="adk.pdb")
mktrj(modes, pdb = adk, file="adk.pdb") #can give an ad
```