Class 10: Structural Bioinformatics Pt. 1

Aileen Andrade (PID A17033749)

2025-02-06

Table of contents

The PDB Database

The main repository of biomolecular structure data is called the PDB https://www.rcsb.org

Let's see what this database contains. I went to PDB > Analyze > PDB Statistics > By Exp method and molecular type.

```r
pdbstats <- read.csv("Data Export Summary.csv", row.names = 1)
pdbstats
```

```
                          X.ray     EM    NMR Multiple.methods Neutron Other
Protein (only)          169,563 16,774 12,578              208      81    32
Protein/Oligosaccharide   9,939  2,839     34                8       2     0
Protein/NA                8,801  5,062    286                7       0     0
Nucleic acid (only)       2,890    151  1,521               14       3     1
Other                       170     10     33                0       0     0
Oligosaccharide (only)       11      0      6                1       0     4
                          Total
Protein (only)          199,236
Protein/Oligosaccharide  12,822
Protein/NA               14,156
Nucleic acid (only)       4,580
```

```
Other                          213
Oligosaccharide (only)          22
```

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

```r
pdbstats$X.ray
```

```
[1] "169,563" "9,939"   "8,801"   "2,890"   "170"     "11"
```

The comma in these numbers is causing them to be read as characters rather than numeric.

I can fix this by replacing "," for nothing " " with the `sub()` function

```r
x <- pdbstats$X.ray
sum(as.numeric( sub(",", "", x) ))
```

```
[1] 191374
```

Or, I can use the **readr** package and the `read_csv()` function.

```r
library(readr)
pdbstats <- read_csv("Data Export Summary.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.
```

```r
pdbstats
```

```
# A tibble: 6 x 8
  `Molecular Type`   `X-ray`    EM   NMR `Multiple methods` Neutron Other  Total
  <chr>                <dbl> <dbl> <dbl>              <dbl>   <dbl> <dbl>  <dbl>
1 Protein (only)      169563 16774 12578                208      81    32 199236
2 Protein/Oligosacc~    9939  2839    34                  8       2     0  12822
3 Protein/NA            8801  5062   286                  7       0     0  14156
4 Nucleic acid (onl~    2890   151  1521                 14       3     1   4580
5 Other                  170    10    33                  0       0     0    213
6 Oligosaccharide (~       11     0     6                  1       0     4     22
```

I want to clean up the column names so that they are all lower case and don't have spaces in them

```r
library(janitor)
```

```

Attaching package: 'janitor'

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

    chisq.test, fisher.test
```

```r
pdbstats <- clean_names(pdbstats)
pdbstats
```

```
# A tibble: 6 x 8
  molecular_type         x_ray    em   nmr multiple_methods neutron other  total
  <chr>                  <dbl> <dbl> <dbl>            <dbl>   <dbl> <dbl>  <dbl>
1 Protein (only)        169563 16774 12578              208      81    32 199236
2 Protein/Oligosacchar~   9939  2839    34                8       2     0  12822
3 Protein/NA              8801  5062   286                7       0     0  14156
4 Nucleic acid (only)     2890   151  1521               14       3     1   4580
5 Other                    170    10    33                0       0     0    213
6 Oligosaccharide (onl~     11     0     6                1       0     4     22
```

Total number of X-ray structures

```r
xraysum <- sum(pdbstats$x_ray)
```

Total number of EM structures

```r
emsum <- sum(pdbstats$em)
```

Total number of structures

```r
totalstruc <- sum(pdbstats$total)
```

Percentage of X-ray structures

```r
xraysum/totalstruc *100
```

```
[1] 82.83549
```

Percentage of EM structures

```r
emsum/totalstruc *100
```

```
[1] 10.75017
```

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

Total number of protein structures

```r
pdbstats[1,]$total / sum(pdbstats$total) *100
```

```
[1] 86.23852
```

2. Using Mol*

The main Mol* homepage at: https://molstar.org/viewer/ We can input our own PDB files or just give it a PDB accession code (4 letter PDB code)

> 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?

There are 231,029 HIV-1 protease structures currently in PDB

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

This is a simplified view.

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

HOH 308

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.

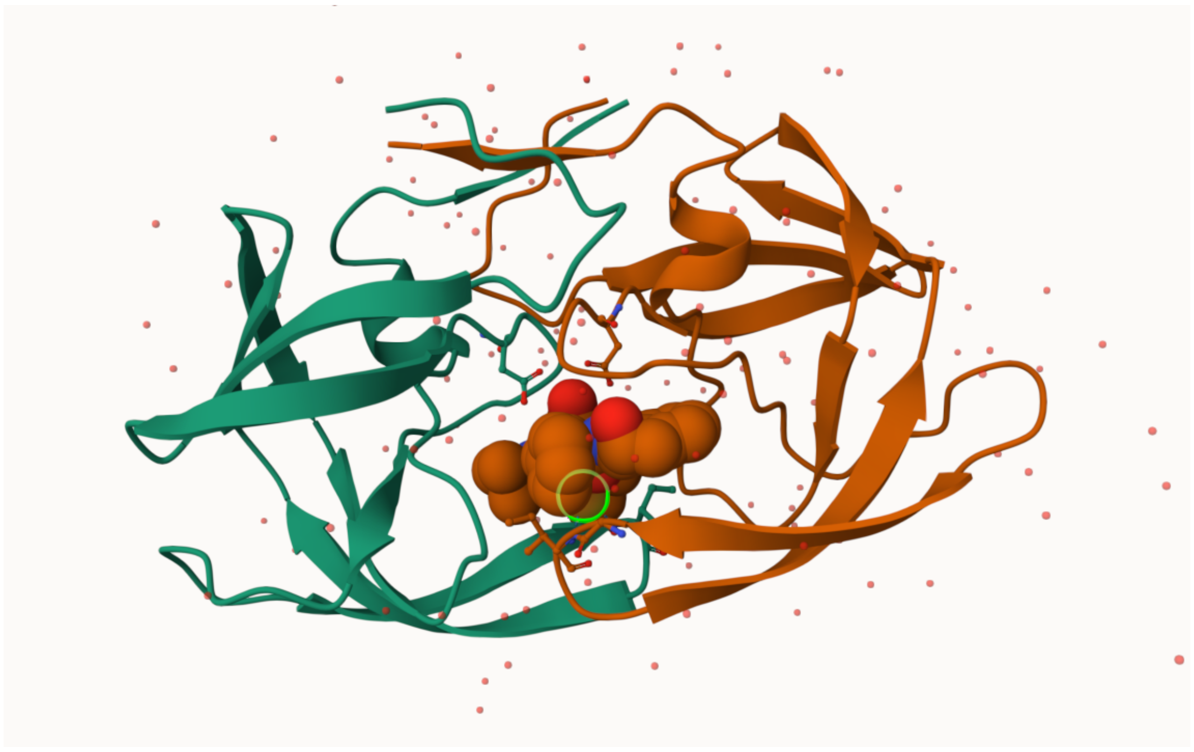


Figure 1: Molecular view of 1HSG

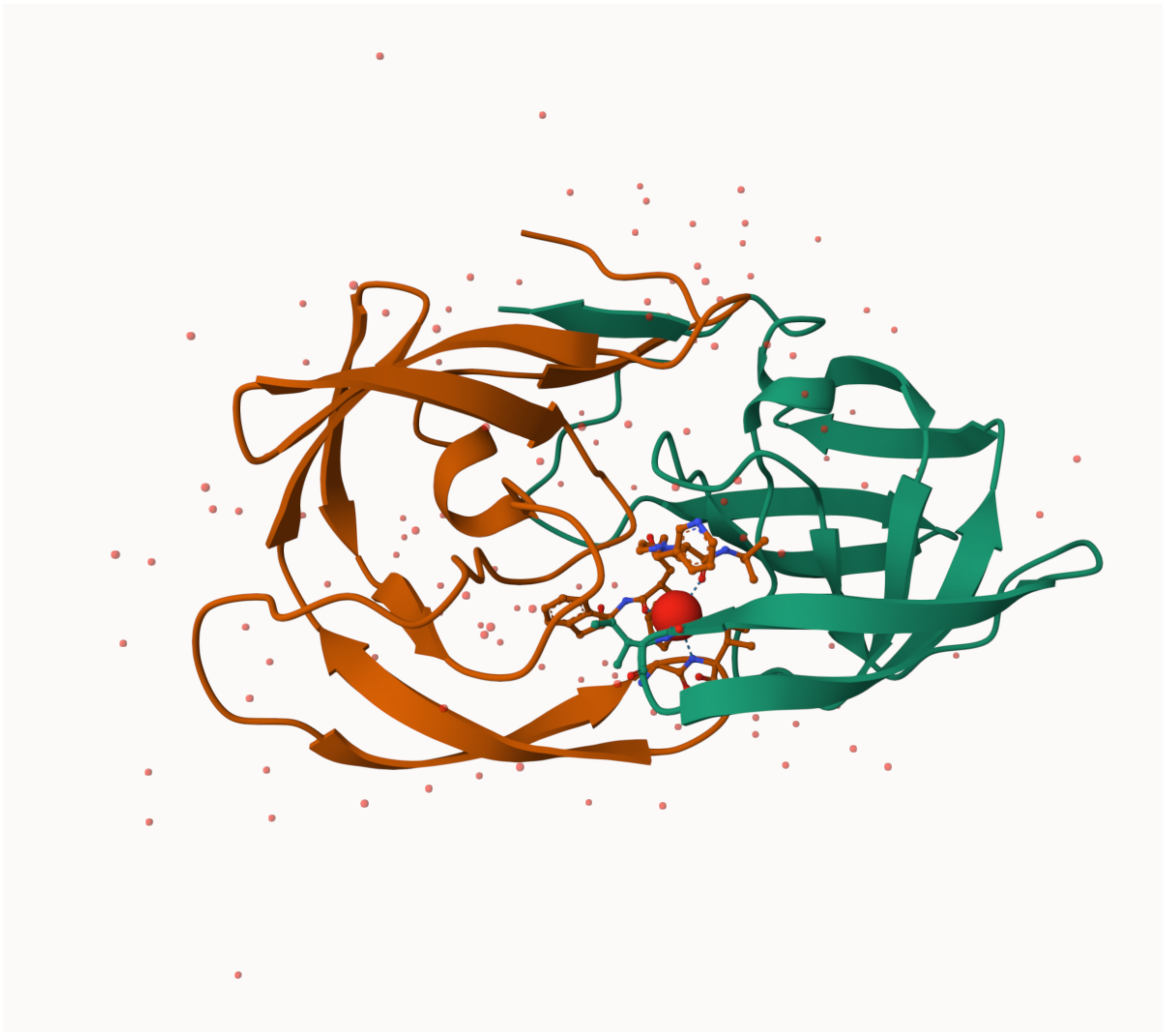


Figure 2: Water 308 in the binding site

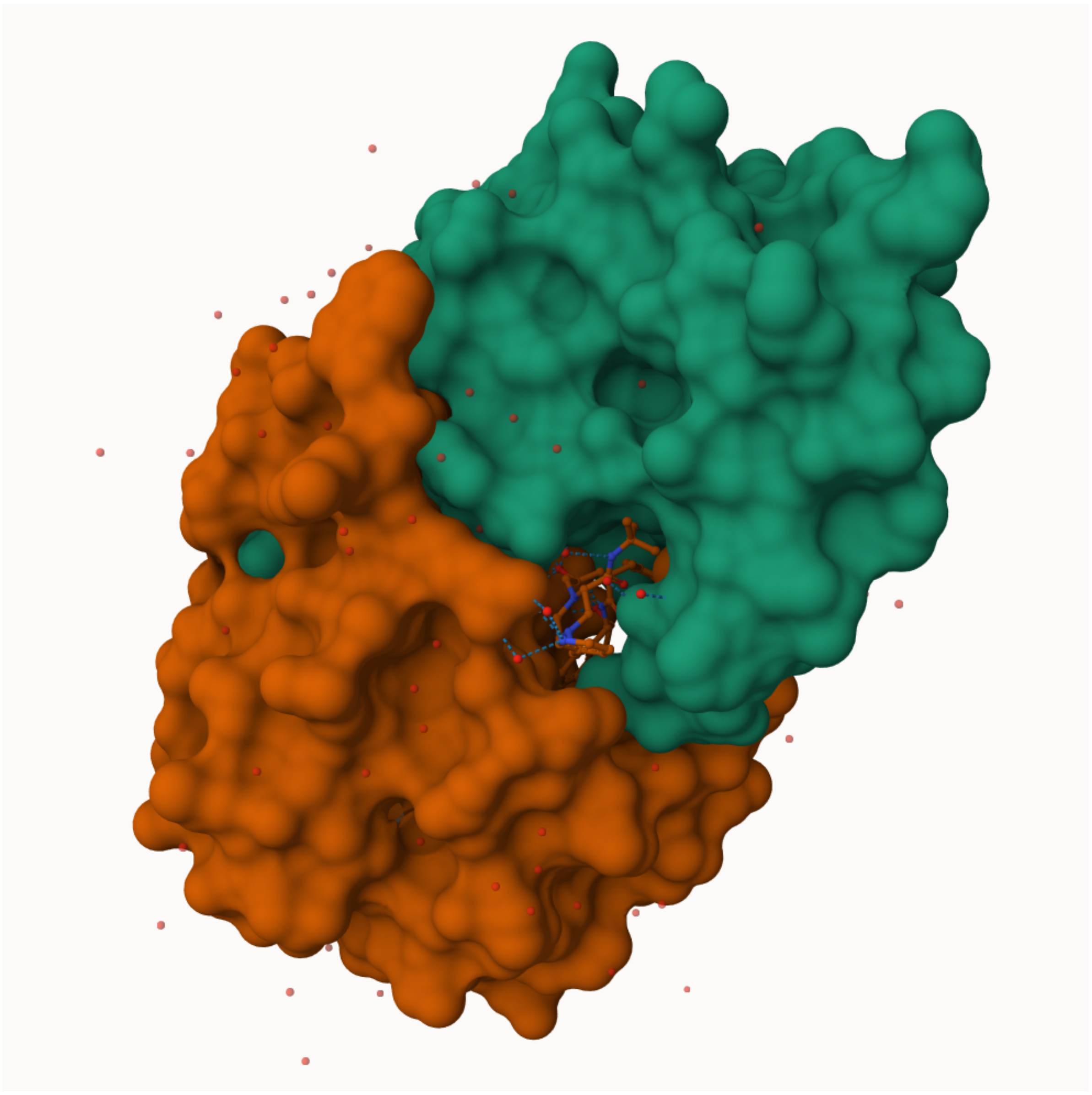


Figure 3: Ligand in the binding site

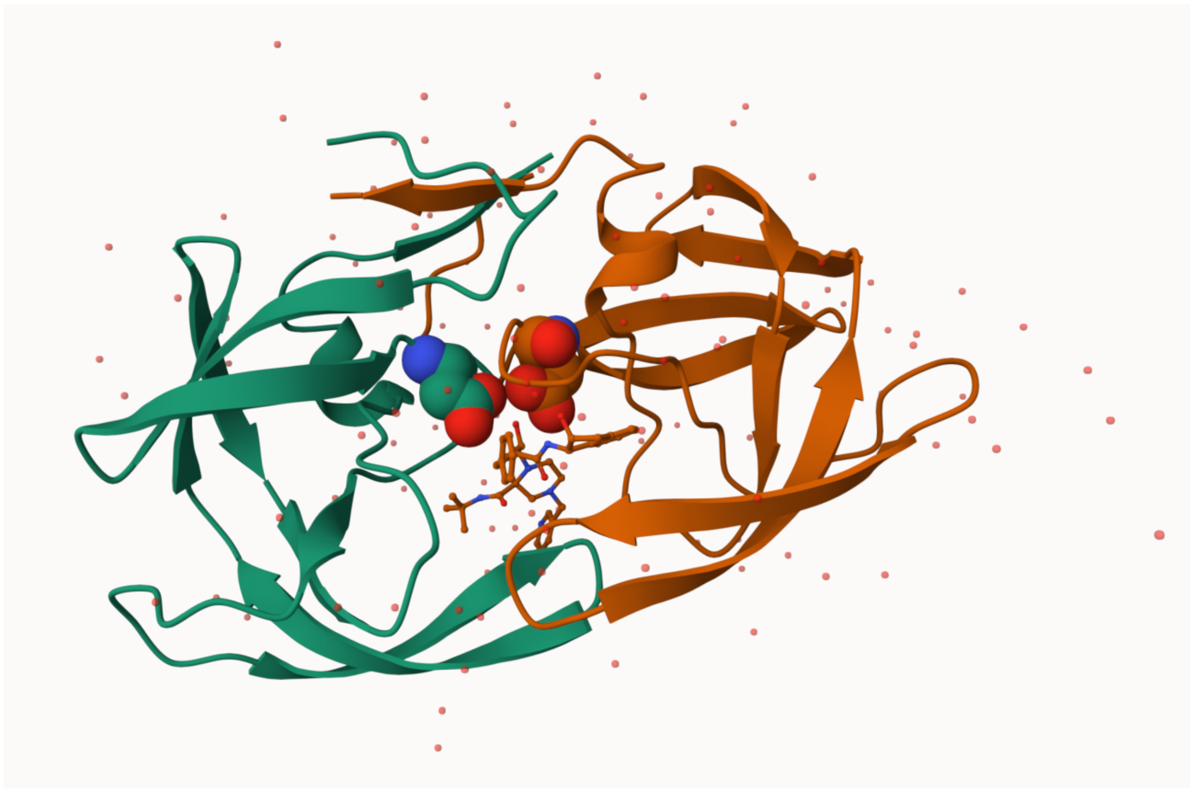


Figure 4: Chain A and B Asp25 Spacefill

3. Introduction to Bio3D in R

We can use the **bio3d** package for structural bioinformatics to read PDB data into R

```
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?

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

```
[1] 198
```

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

MK1

> Q9: How many protein chains are in this structure?

2 chains, A and B

Looking at the `pdb` object in more detail

```r
attributes(pdb)
```

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

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

```r
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>
```

Let's try a new function not yet in the bio3d package. It requires the **r3dmol** and **shiny** package that we need to install with `install.packages("r3dmol")` and `install.packages("shiny")`

```r
source("https://tinyurl.com/viewpdb")
#view.pdb(pdb, backgroundColor = "peachpuff")
```

4. Predicting functional dynamics

We can use the `nma()` function in bio3d to predict the large-scale functional motions of biomolecules.

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

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

```r
adk
```

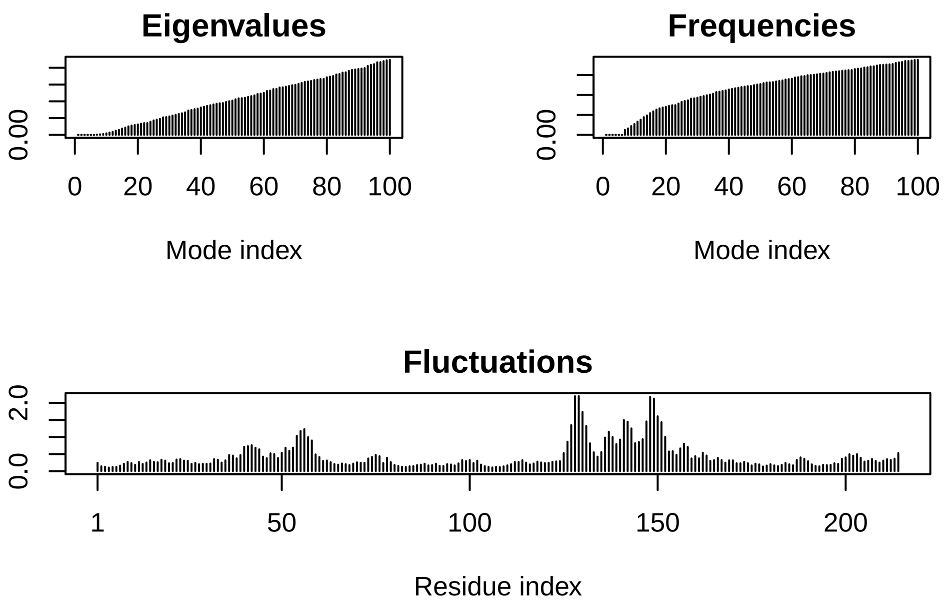
```
 Call:  read.pdb(file = "6s36")
```

```
   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
```

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

```
 Building Hessian...        Done in 0.015 seconds.
 Diagonalizing Hessian...   Done in 0.286 seconds.
```

```
plot(m)
```



Write out a trajectory of the predicted molecular motion:

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