Class 09 Structural Bioinformatics Pt 1

Sindy Chavez

1: Introduction to the RCSB Protein Data Bank (PDB)

Skipped section 1 because the PDB site was too slow

```
pdb_database <- read.csv('Data Export Summary.csv')
pdb_database
```

```
           Molecular.Type   X.ray    NMR    EM Multiple.methods Neutron Other
1          Protein (only) 150,417 12,056 8,586              188      72    32
2 Protein/Oligosaccharide   8,869     32 1,552                6       0     0
3              Protein/NA   7,943    280 2,690                6       0     0
4     Nucleic acid (only)   2,522  1,425    74               13       2     1
5                   Other     154     31     6                0       0     0
6  Oligosaccharide (only)      11      6     0                1       0     4
    Total
1 171,351
2  10,459
3  10,919
4   4,037
5     191
6      22
```

2: Visualizing the HIV-1 protease structure

Using Mol* (pronounced molstar) to view PBD structures

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

The resolution of the image is 2A, but the the hydrogen atoms in the water molecules are smaller than the 2A resolution.

> 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

Yes, it interacts with residue ILE 50 on the B chain of the protein

> 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. Discussion Topic: Can you think of a way in which indinavir, or even larger ligands and substrates, could enter the binding site?

While the image shows the protein in a single configuration, proteins are actually moving all the time. The movement of the protein can cause configurations where molecules could enter the binding site.

> Q7: [Optional] As you have hopefully observed HIV protease is a homodimer (i.e. it is composed of two identical chains). With the aid of the graphic display can you

identify secondary structure elements that are likely to only form in the dimer rather than the monomer?

The dimer is able to form a pocket between the two chains where a molecule can bind.

3. Introduction to Bio3D in R

The 'bio3d' package for structural bioinformatics has lots of features for reading and working with biomolecular sequences and 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
```

```
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>
```

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

198

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

MK1

Q9: How many protein chains are in this structure?

2

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
```

```
adk
```

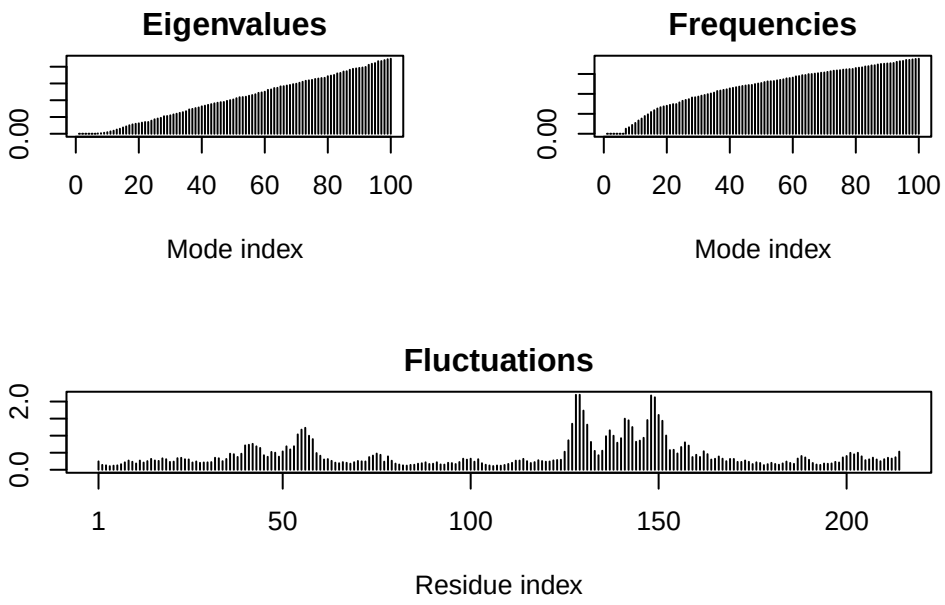
```
 Call:  read.pdb(file = "6s36")

   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) it is a bioinformatics method for predicting functional motions. It will show us the parts of the proteins that are "flexible" (i.e. most dynamic).

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

```
 Building Hessian...        Done in 0.03 seconds.
 Diagonalizing Hessian...   Done in 0.56 seconds.
```

```
plot(m)
```



Make a "movie" of this thing moving.

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

4. Comparative analysis of all ADK structures

Setup: Install and load the following packages in the console

install.packages("bio3d") install.packages("devtools") install.packages("BiocManager")

BiocManager::install("msa") devtools::install_bitbucket("Grantlab/bio3d-view")

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

msa

Q11. Which of the above packages is not found on BioConductor or CRAN?:

None, they are all found on either BioConductor or CRAN

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

True

Search and retrieve ADK structures

First we get the sequence of ADK and use this to search the PDB database.

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

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

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

```
read.pdb("adk_nma.pdb")
```

```
 Call:  read.pdb(file = "adk_nma.pdb")

   Total Models#: 1
     Total Atoms#: 214,  XYZs#: 642  Chains#: 1  (values: NA)

     Protein Atoms#: 214  (residues/Calpha atoms#: 214)
     Nucleic acid Atoms#: 0  (residues/phosphate atoms#: 0)

     Non-protein/nucleic Atoms#: 0  (residues: 0)
     Non-protein/nucleic resid values: [ none ]

   Protein sequence:
      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
      AAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

+ attr: atom, xyz, calpha, call
```

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

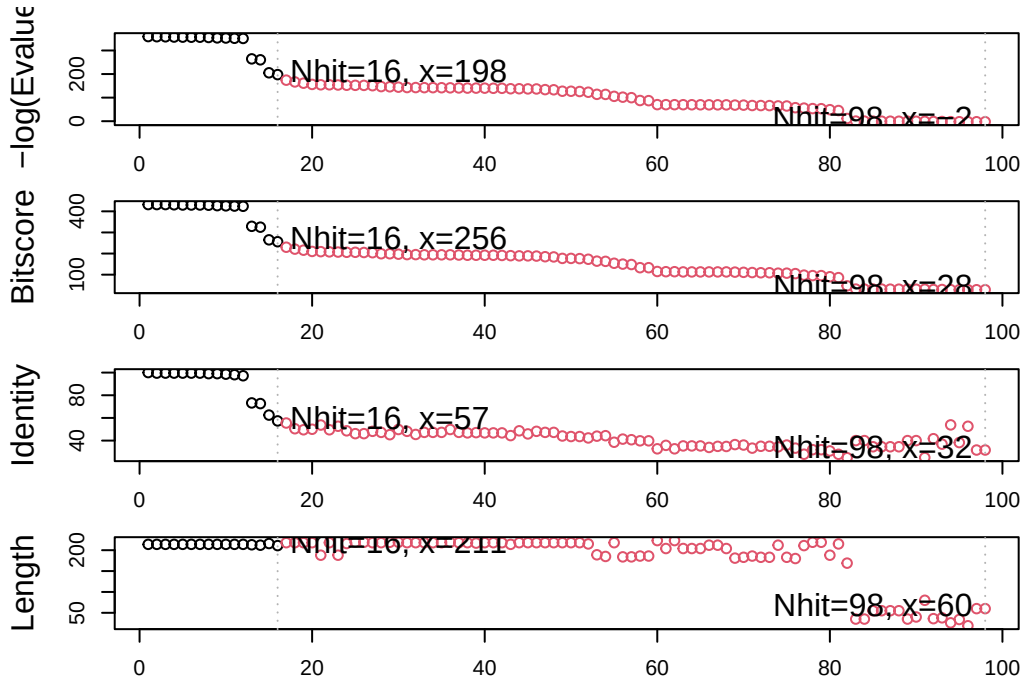
214

```
blast <- blast.pdb(aa)
```

```
 Searching ... please wait (updates every 5 seconds) RID = NNXK5KCK016
 ..........
 Reporting 98 hits
```

```
hits <- plot(blast)
```

```
* Possible cutoff values:    197 -3
          Yielding Nhits:    16 98

* Chosen cutoff value of:    197
          Yielding Nhits:    16
```



nhit - 15 hits that have a really good e-value

```
hits$pdb.id
```

```
[1] "1AKE_A" "4X8M_A" "6S36_A" "6RZE_A" "4X8H_A" "3HPR_A" "1E4V_A" "5EJE_A"
[9] "1E4Y_A" "3X2S_A" "6HAP_A" "6HAM_A" "4K46_A" "4NP6_A" "3GMT_A" "4PZL_A"
```

What are these structures?

```
head(pdb.annotate(hits$pdb.id))
```

```
       structureId chainId macromoleculeType chainLength experimentalTechnique
1AKE_A        1AKE       A           Protein         214                 X-ray
4X8M_A        4X8M       A           Protein         214                 X-ray
6S36_A        6S36       A           Protein         214                 X-ray
6RZE_A        6RZE       A           Protein         214                 X-ray
4X8H_A        4X8H       A           Protein         214                 X-ray
3HPR_A        3HPR       A           Protein         214                 X-ray
       resolution       scopDomain                                      pfam
1AKE_A       2.00 Adenylate kinase Adenylate kinase, active site lid (ADK_lid)
4X8M_A       2.60             <NA> Adenylate kinase, active site lid (ADK_lid)
6S36_A       1.60             <NA> Adenylate kinase, active site lid (ADK_lid)
6RZE_A       1.69             <NA> Adenylate kinase, active site lid (ADK_lid)
4X8H_A       2.50             <NA> Adenylate kinase, active site lid (ADK_lid)
3HPR_A       2.00             <NA> Adenylate kinase, active site lid (ADK_lid)
              ligandId                                    ligandName
1AKE_A              AP5             BIS(ADENOSINE)-5'-PENTAPHOSPHATE
4X8M_A             <NA>                                         <NA>
6S36_A CL (3),NA,MG (2) CHLORIDE ION (3),SODIUM ION,MAGNESIUM ION (2)
6RZE_A    NA (3),CL (2)              SODIUM ION (3),CHLORIDE ION (2)
4X8H_A             <NA>                                         <NA>
3HPR_A              AP5             BIS(ADENOSINE)-5'-PENTAPHOSPHATE
                    source
1AKE_A      Escherichia coli
4X8M_A      Escherichia coli
6S36_A      Escherichia coli
6RZE_A      Escherichia coli
4X8H_A      Escherichia coli
3HPR_A Escherichia coli K-12

1AKE_A STRUCTURE OF THE COMPLEX BETWEEN ADENYLATE KINASE FROM ESCHERICHIA COLI AND THE INHIBI
4X8M_A
6S36_A
6RZE_A
4X8H_A
3HPR_A
                                                citation rObserved  rFree
1AKE_A               Muller, C.W., et al. J Mol Biol (1992)    0.1960     NA
4X8M_A              Kovermann, M., et al. Nat Commun (2015)    0.2491 0.3089
6S36_A                Rogne, P., et al. Biochemistry (2019)    0.1632 0.2356
6RZE_A                Rogne, P., et al. Biochemistry (2019)    0.1865 0.2350
4X8H_A              Kovermann, M., et al. Nat Commun (2015)    0.1961 0.2895
3HPR_A Schrank, T.P., et al. Proc Natl Acad Sci U S A (2009)    0.2100 0.2432
        rWork spaceGroup
```

```
1AKE_A 0.1960  P 21 2 21
4X8M_A 0.2463    C 1 2 1
6S36_A 0.1594    C 1 2 1
6RZE_A 0.1819    C 1 2 1
4X8H_A 0.1914    C 1 2 1
3HPR_A 0.2062  P 21 21 2
```

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

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
1AKE.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
4X8M.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
6S36.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
6RZE.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
4X8H.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
3HPR.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
1E4V.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
5EJE.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
1E4Y.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
3X2S.pdb exists. Skipping download
```

```
Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
6HAP.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
6HAM.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
4K46.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
4NP6.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
3GMT.pdb exists. Skipping download

Warning in get.pdb(hits$pdb.id, path = "pdbs", split = TRUE, gzip = TRUE): pdbs/
4PZL.pdb exists. Skipping download


  |
  |                                                                      |   0%
  |
  |====                                                                  |   6%
  |
  |=========                                                             |  12%
  |
  |=============                                                         |  19%
  |
  |==================                                                    |  25%
  |
  |======================                                                |  31%
  |
  |==========================                                            |  38%
  |
  |==============================                                        |  44%
  |
  |==================================                                    |  50%
  |
  |======================================                                |  56%
  |
  |==========================================                            |  62%
```

```
  |
  |=================================================                     |  69%
  |
  |=====================================================                 |  75%
  |
  |=========================================================             |  81%
  |
  |=============================================================         |  88%
  |
  |=================================================================     |  94%
  |
  |======================================================================| 100%
```

The arguments make this quicker

We will align and superimpose these structures.

```
pdbs <- pdbaln(files, fit = TRUE, exefile="msa")
```

```
Reading PDB files:
pdbs/split_chain/1AKE_A.pdb
pdbs/split_chain/4X8M_A.pdb
pdbs/split_chain/6S36_A.pdb
pdbs/split_chain/6RZE_A.pdb
pdbs/split_chain/4X8H_A.pdb
pdbs/split_chain/3HPR_A.pdb
pdbs/split_chain/1E4V_A.pdb
pdbs/split_chain/5EJE_A.pdb
pdbs/split_chain/1E4Y_A.pdb
pdbs/split_chain/3X2S_A.pdb
pdbs/split_chain/6HAP_A.pdb
pdbs/split_chain/6HAM_A.pdb
pdbs/split_chain/4K46_A.pdb
pdbs/split_chain/4NP6_A.pdb
pdbs/split_chain/3GMT_A.pdb
pdbs/split_chain/4PZL_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
..   PDB has ALT records, taking A only, rm.alt=TRUE
.   PDB has ALT records, taking A only, rm.alt=TRUE
..   PDB has ALT records, taking A only, rm.alt=TRUE
..   PDB has ALT records, taking A only, rm.alt=TRUE
....   PDB has ALT records, taking A only, rm.alt=TRUE
```

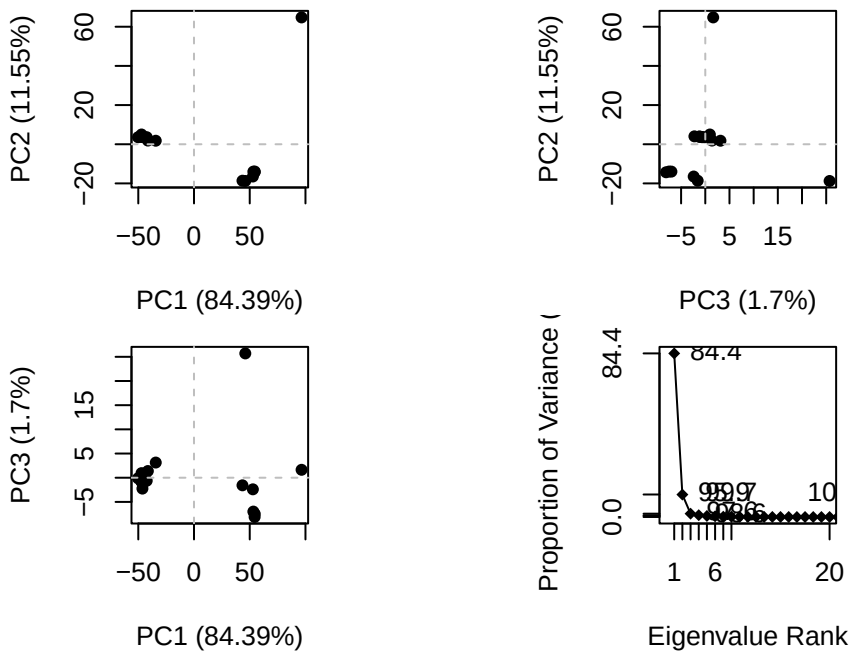
```
.   PDB has ALT records, taking A only, rm.alt=TRUE
....

Extracting sequences

pdb/seq: 1   name: pdbs/split_chain/1AKE_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 2   name: pdbs/split_chain/4X8M_A.pdb
pdb/seq: 3   name: pdbs/split_chain/6S36_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 4   name: pdbs/split_chain/6RZE_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 5   name: pdbs/split_chain/4X8H_A.pdb
pdb/seq: 6   name: pdbs/split_chain/3HPR_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 7   name: pdbs/split_chain/1E4V_A.pdb
pdb/seq: 8   name: pdbs/split_chain/5EJE_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 9   name: pdbs/split_chain/1E4Y_A.pdb
pdb/seq: 10   name: pdbs/split_chain/3X2S_A.pdb
pdb/seq: 11   name: pdbs/split_chain/6HAP_A.pdb
pdb/seq: 12   name: pdbs/split_chain/6HAM_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 13   name: pdbs/split_chain/4K46_A.pdb
   PDB has ALT records, taking A only, rm.alt=TRUE
pdb/seq: 14   name: pdbs/split_chain/4NP6_A.pdb
pdb/seq: 15   name: pdbs/split_chain/3GMT_A.pdb
pdb/seq: 16   name: pdbs/split_chain/4PZL_A.pdb
```

PCA to the rescue...

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

And plot my results

```
plot(pc.xray)
```

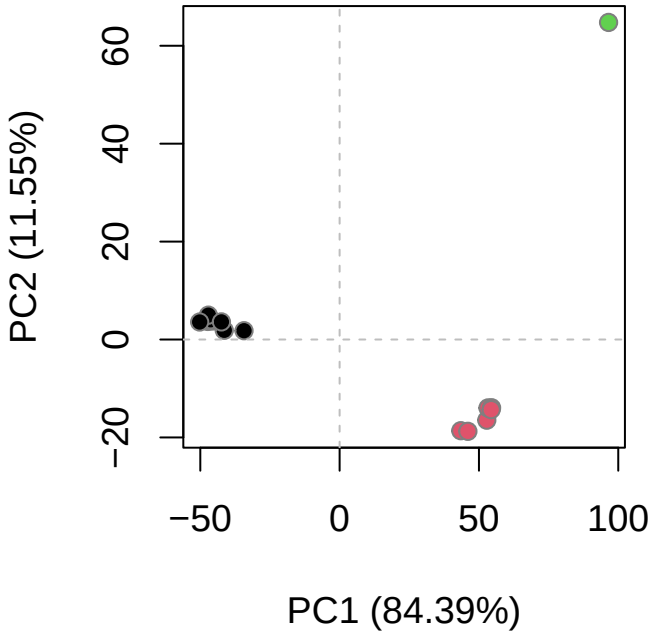


Pairwise clustering

```
rd <- rmsd(pdbs)
```

```
Warning in rmsd(pdbs): No indices provided, using the 204 non NA positions
```

```
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)
```



5. Optional further visualization

Let's make a movie

```
mktrj(pc.xray, pc=1, file="pc_1.pdb")
```

6. Normal mode analysis [optional]

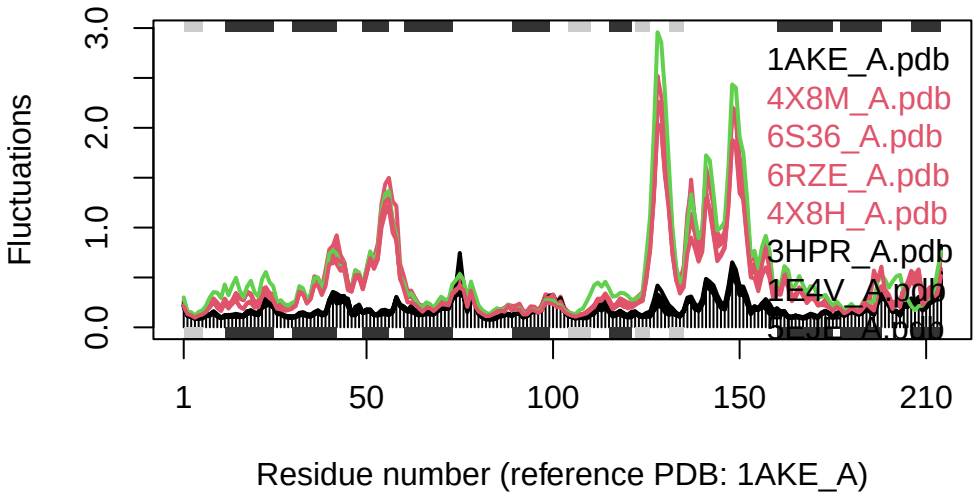
```
modes <- nma(pdbs)
```

```
Details of Scheduled Calculation:
  ... 16 input structures
  ... storing 606 eigenvectors for each structure
  ... dimension of x$U.subspace: ( 612x606x16 )
  ... coordinate superposition prior to NM calculation
  ... aligned eigenvectors (gap containing positions removed)
  ... estimated memory usage of final 'eNMA' object: 45.4 Mb
```

```
  |
  |                                                                      |   0%
  |
  |====                                                                  |   6%
  |
  |=========                                                             |  12%
  |
  |==============                                                        |  19%
  |
  |==================                                                    |  25%
  |
  |======================                                                |  31%
  |
  |==========================                                            |  38%
  |
  |================================                                      |  44%
  |
  |====================================                                  |  50%
  |
  |========================================                              |  56%
  |
  |============================================                          |  62%
  |
  |================================================                      |  69%
  |
  |====================================================                  |  75%
  |
  |========================================================              |  81%
  |
  |============================================================          |  88%
  |
  |================================================================      |  94%
  |
  |======================================================================| 100%
```

```
plot(modes, pdbs, col=grps.rd)
```

```
Extracting SSE from pdbs$sse attribute
```



Q14. What do you note about this plot? Are the black and colored lines similar or different? Where do you think they differ most and why?

The black and colored lines are different, though the colored lines are more similar to each other. The places where the colored lines differ most from the black line look like areas near the pocket of the protein. These residues could be important for binding a substrate.