Class12: Structural Bioinformatics II

Tom Quach (A15549142)

11/4/2021

Comparative analysis of protein structures

##Using Bio3D

I need to load the library

```
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?

There are 198 amino acid residues

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

HOH

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

There are 2 chains

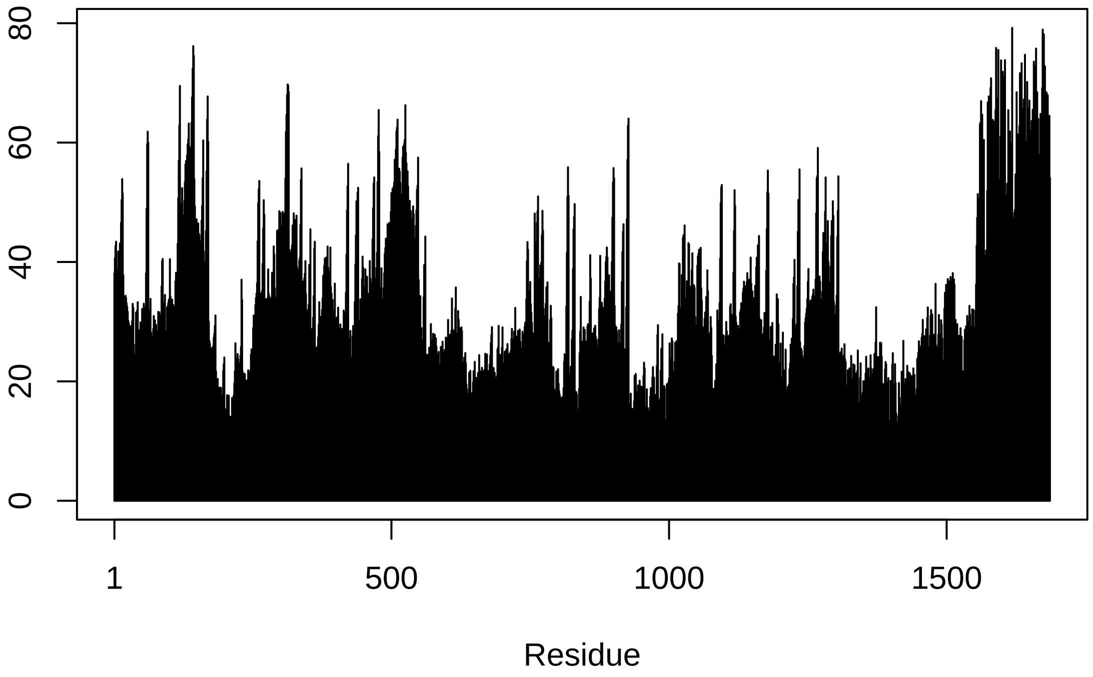
```
aa123(pdbseq(pdb))
```

```
##   [1] "PRO" "GLN" "ILE" "THR" "LEU" "TRP" "GLN" "ARG" "PRO" "LEU" "VAL" "THR"
##  [13] "ILE" "LYS" "ILE" "GLY" "GLY" "GLN" "LEU" "LYS" "GLU" "ALA" "LEU" "LEU"
##  [25] "ASP" "THR" "GLY" "ALA" "ASP" "ASP" "THR" "VAL" "LEU" "GLU" "GLU" "MET"
##  [37] "SER" "LEU" "PRO" "GLY" "ARG" "TRP" "LYS" "PRO" "LYS" "MET" "ILE" "GLY"
##  [49] "GLY" "ILE" "GLY" "GLY" "PHE" "ILE" "LYS" "VAL" "ARG" "GLN" "TYR" "ASP"
##  [61] "GLN" "ILE" "LEU" "ILE" "GLU" "ILE" "CYS" "GLY" "HIS" "LYS" "ALA" "ILE"
##  [73] "GLY" "THR" "VAL" "LEU" "VAL" "GLY" "PRO" "THR" "PRO" "VAL" "ASN" "ILE"
##  [85] "ILE" "GLY" "ARG" "ASN" "LEU" "LEU" "THR" "GLN" "ILE" "GLY" "CYS" "THR"
##  [97] "LEU" "ASN" "PHE" "PRO" "GLN" "ILE" "THR" "LEU" "TRP" "GLN" "ARG" "PRO"
## [109] "LEU" "VAL" "THR" "ILE" "LYS" "ILE" "GLY" "GLY" "GLN" "LEU" "LYS" "GLU"
## [121] "ALA" "LEU" "LEU" "ASP" "THR" "GLY" "ALA" "ASP" "ASP" "THR" "VAL" "LEU"
## [133] "GLU" "GLU" "MET" "SER" "LEU" "PRO" "GLY" "ARG" "TRP" "LYS" "PRO" "LYS"
## [145] "MET" "ILE" "GLY" "GLY" "ILE" "GLY" "GLY" "PHE" "ILE" "LYS" "VAL" "ARG"
## [157] "GLN" "TYR" "ASP" "GLN" "ILE" "LEU" "ILE" "GLU" "ILE" "CYS" "GLY" "HIS"
## [169] "LYS" "ALA" "ILE" "GLY" "THR" "VAL" "LEU" "VAL" "GLY" "PRO" "THR" "PRO"
## [181] "VAL" "ASN" "ILE" "ILE" "GLY" "ARG" "ASN" "LEU" "LEU" "THR" "GLN" "ILE"
## [193] "GLY" "CYS" "THR" "LEU" "ASN" "PHE"
```

Plot of B-factor

```
plot.bio3d(pdb$atom$b, sse=pdb)
```

```
## Warning in plotb3(...): Length of input ’sse’ does not equal the length of input
## ’x’; Ignoring ’sse’
```



The ATOM records

```
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>
```

> 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?

bio3d-view

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

True

Using bio3d package

```
library(bio3d)

pbd <- read.pdb("1hel")
```

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

```
pbd
```

```
##
##  Call:  read.pdb(file = "1hel")
##
##    Total Models#: 1
##      Total Atoms#: 1186,  XYZs#: 3558  Chains#: 1  (values: A)
##
##      Protein Atoms#: 1001  (residues/Calpha atoms#: 129)
##      Nucleic acid Atoms#: 0  (residues/phosphate atoms#: 0)
##
##      Non-protein/nucleic Atoms#: 185  (residues: 185)
##      Non-protein/nucleic resid values: [ HOH (185) ]
##
##    Protein sequence:
##       KVFGRCELAAAMKRHGLDNYRGYSLGNWVCAAKFESNFNTQATNRNTDGSTDYGILQINS
##       RWWCNDGRTPGSRNLCNIPCSALLSSDITASVNCAKKIVSDGNGMNAWVAWRNRCKGTDV
##       QAWIRGCRL
##
## + attr: atom, xyz, seqres, helix, sheet,
##         calpha, remark, call
```

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

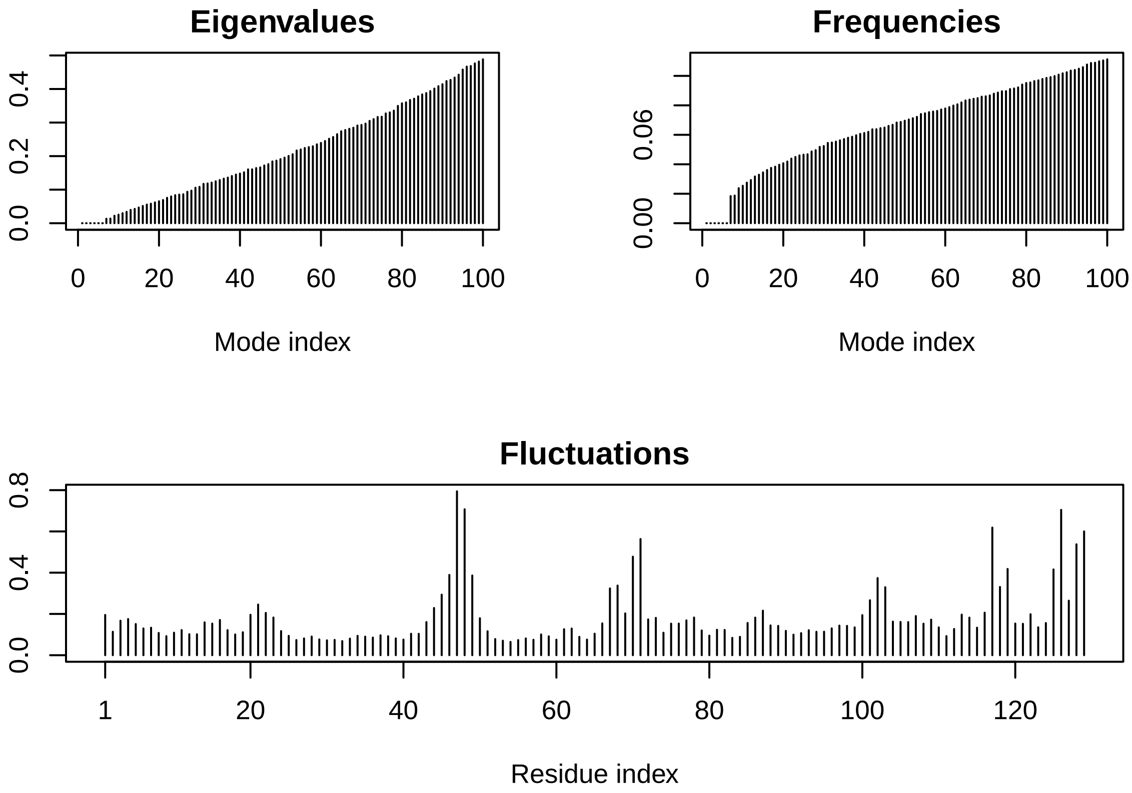
214

Let’s use a bioinformatics method called NMA (Normal Mode Analysis) to predict the dynamics (flexibility) of this enzyme.

```
modes <- nma(pbd)
```

```
##  Building Hessian...     Done in 0.02 seconds.
##  Diagonalizing Hessian...    Done in 0.16 seconds.
```

```
plot(modes)
```



Make a "movie" of its predicted motion. We often call this a "trajectory." Making a file that can be read in VMD.

```
mktrj(modes, file = "nma.pdb")
```

1hel[ng.png, not knitting into PDF enough though it is .png file

#Analysis of ADK

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

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

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

```
aa
```

```
##                1        .         .         .         .         .         60
## pdb|1AKE|A     MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT
##                1        .         .         .         .         .         60
##
##               61        .         .         .         .         .         120
## pdb|1AKE|A     DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDRI
##               61        .         .         .         .         .         120
##
```

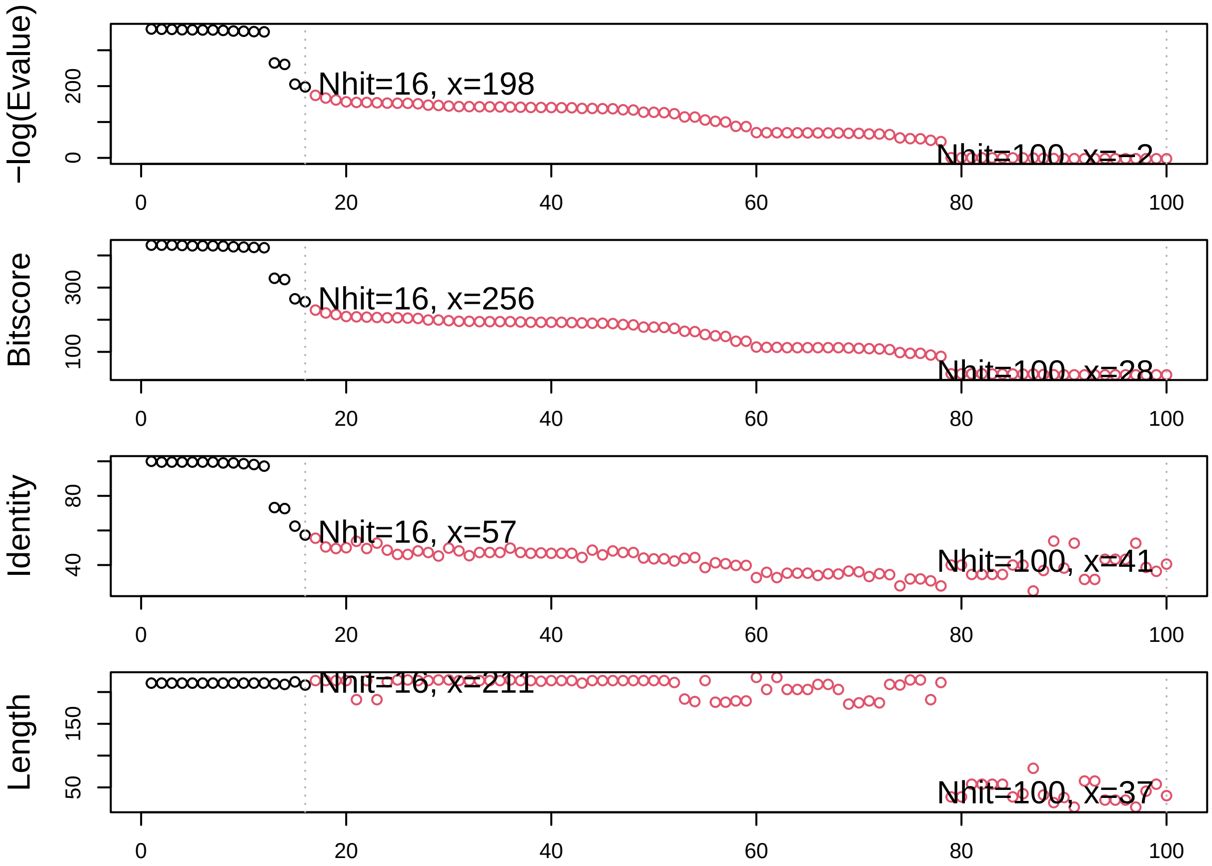
```
##            121        .         .         .         .         .         180
## pdb|1AKE|A   VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG
##            121        .         .         .         .         .         180
##
##            181        .         .         .   214
## pdb|1AKE|A   YYSKEAEAGNTKYAKVDGTKPVAEVRADLEKILG
##            181        .         .         .   214
##
## Call:
##   read.fasta(file = outfile)
##
## Class:
##   fasta
##
## Alignment dimensions:
##   1 sequence rows; 214 position columns (214 non-gap, 0 gap)
##
## + attr: id, ali, call
```

```
#Run BLAST from R
blast <- blast.pdb(aa)
```

```
##  Searching ... please wait (updates every 5 seconds) RID = SF024MBM016
##  .
##  Reporting 100 hits
```

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

```
##   * Possible cutoff values:    197 -3
##             Yielding Nhits:    16 100
##
##   * Chosen cutoff value of:    197
##             Yielding Nhits:    16
```



```r
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"
```

```r
# Download releated PDB files
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
```

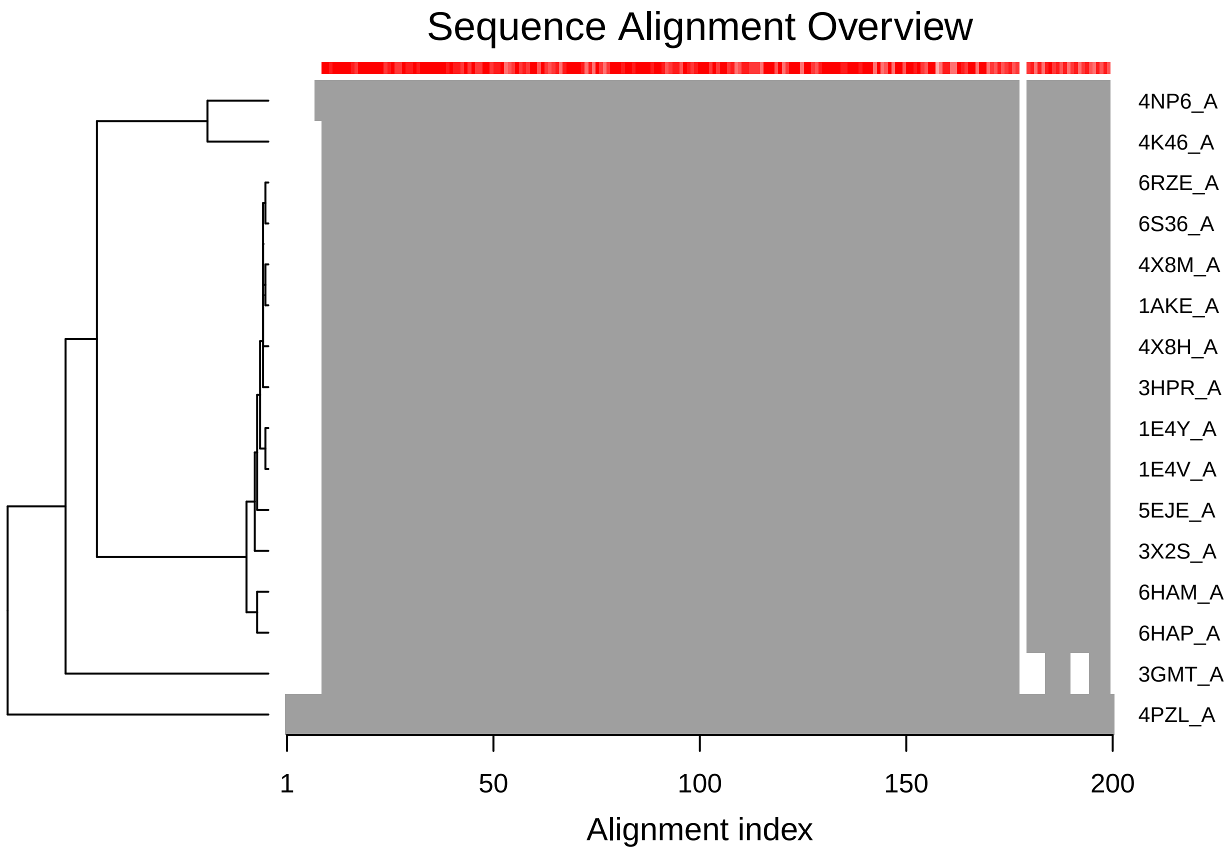
```
##   |                                                                              |
```

Multiple structure alignment

```r
# Align releated PDBs
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
```

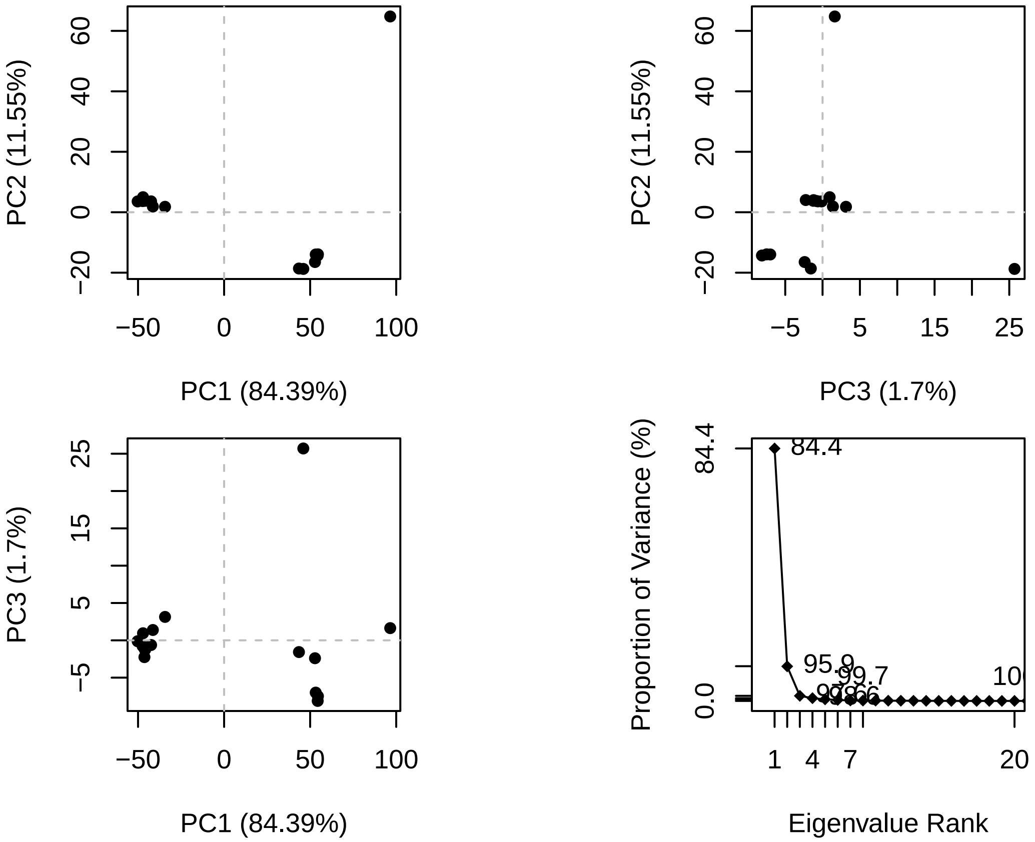
```r
# Vector containing PDB codes for figure axis
ids <- basename.pdb(pdbs$id)

# Draw schematic alignment
plot(pdbs, labels=ids)
```



PCA

We will use the bio3d pca() function which is deesigned for protein structure data .

```
# Perform PCA
pc.xray <- pca(pdbs)
plot(pc.xray)
```



Make a trajectory visualization of the motion captured by the first Principal Componet

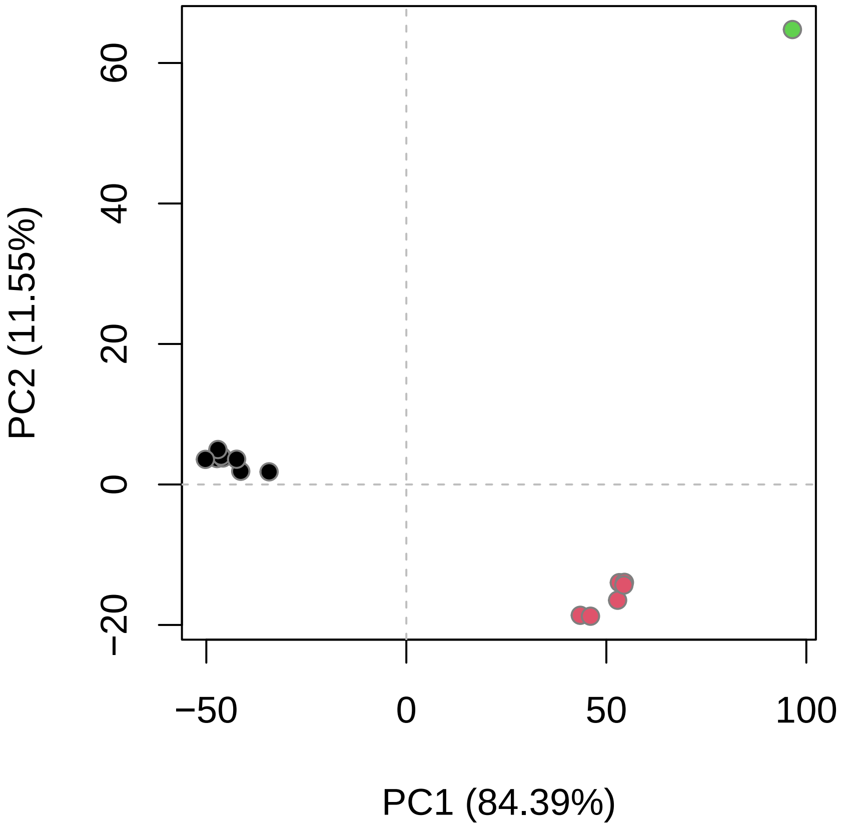
```
# Visualize first principal component
pc1 <- mktrj(pc.xray, pc=1, file="pc_1.pdb")
```

pc_1p.png not knitting into PDF enough though it is .png file

```
# Calculate RMSD
rd <- rmsd(pdbs)
```

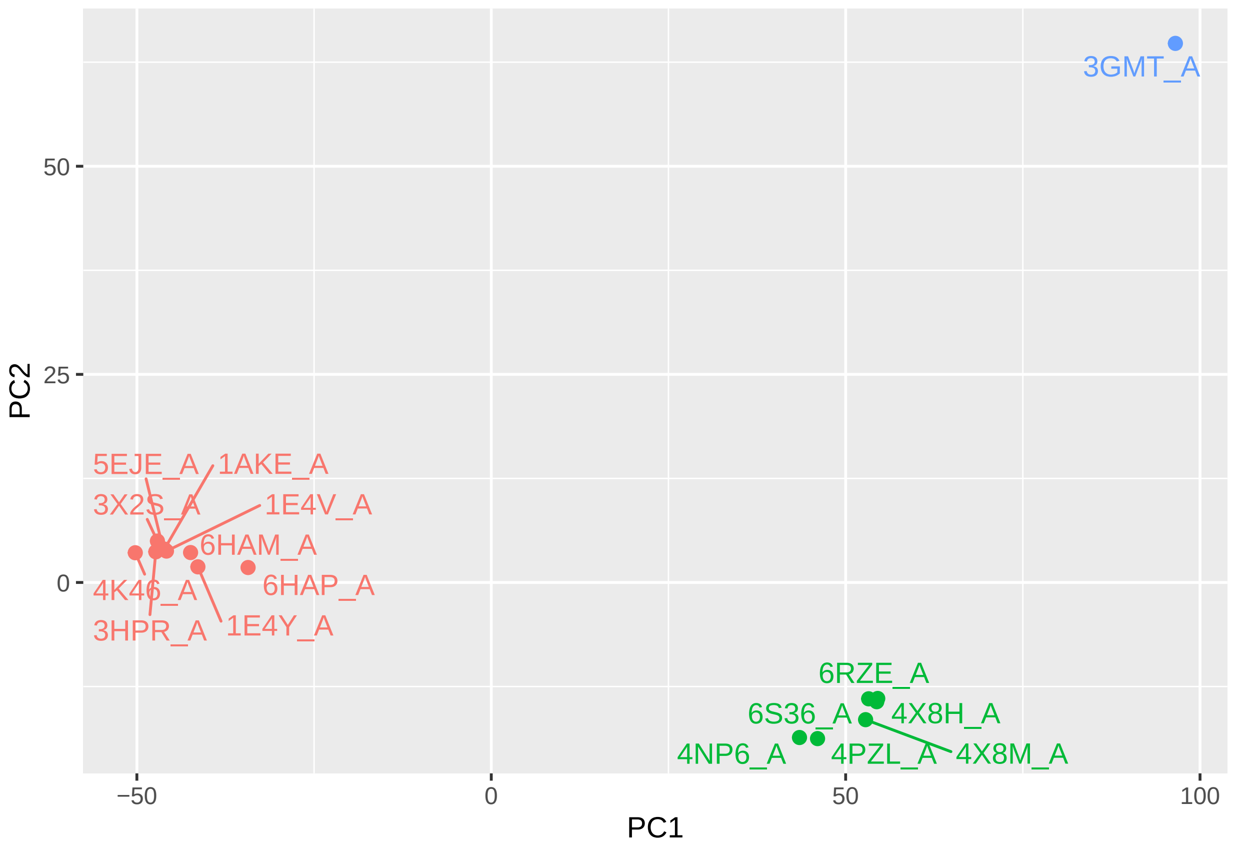
```
## Warning in rmsd(pdbs): No indices provided, using the 204 non NA positions
```

```
# Structure-based clustering
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)
```



```
#view.xyz(pc1)
```

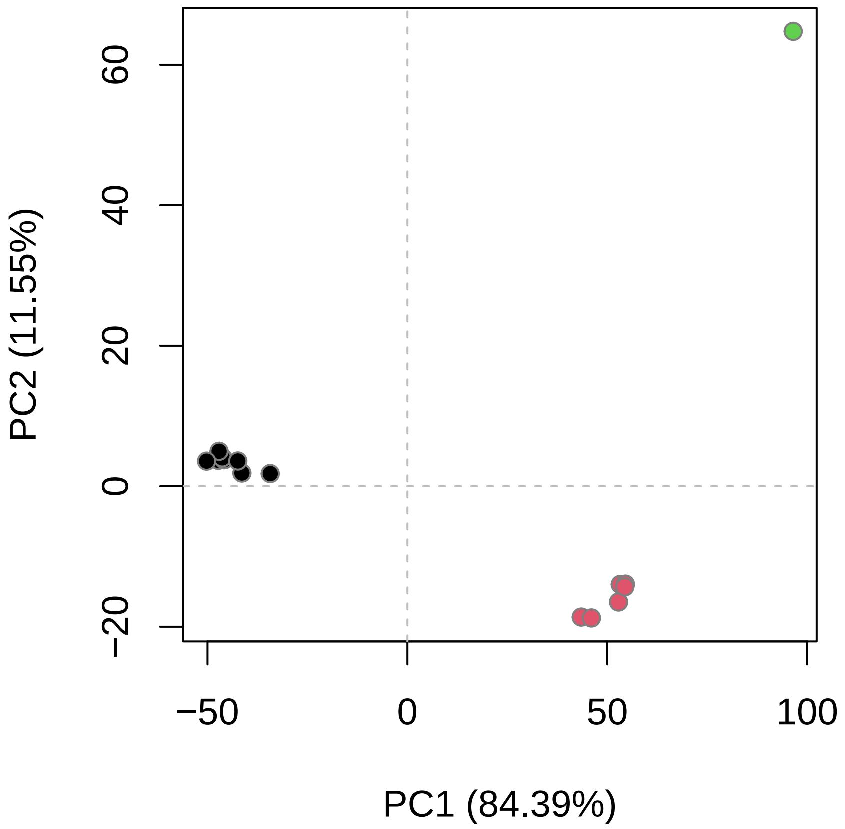
```
#Plotting results with ggplot2
library(ggplot2)
library(ggrepel)

df <- data.frame(PC1=pc.xray$z[,1],
                 PC2=pc.xray$z[,2],
                 col=as.factor(grps.rd),
                 ids=ids)

p <- ggplot(df) +
  aes(PC1, PC2, col=col, label=ids) +
  geom_point(size=2) +
  geom_text_repel(max.overlaps = 20) +
  theme(legend.position = "none")
p
```



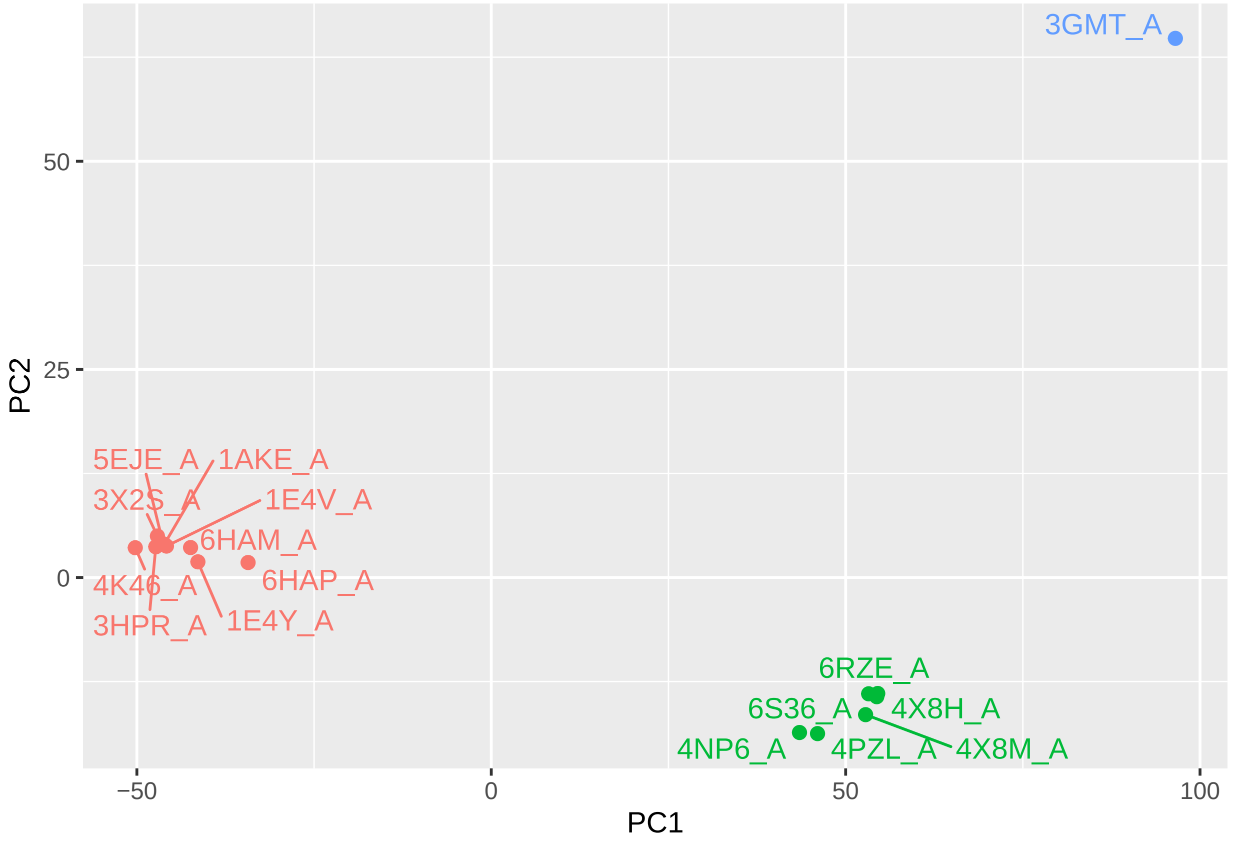
```
# Calculate RMSD
rd <- rmsd(pdbs)
```

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

```
# Structure-based clustering
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)
```



```
#Plotting results with ggplot2
library(ggplot2)
library(ggrepel)

df <- data.frame(PC1=pc.xray$z[,1],
                 PC2=pc.xray$z[,2],
                 col=as.factor(grps.rd),
                 ids=ids)

p <- ggplot(df) +
  aes(PC1, PC2, col=col, label=ids) +
  geom_point(size=2) +
  geom_text_repel(max.overlaps = 20) +
  theme(legend.position = "none")

p
```

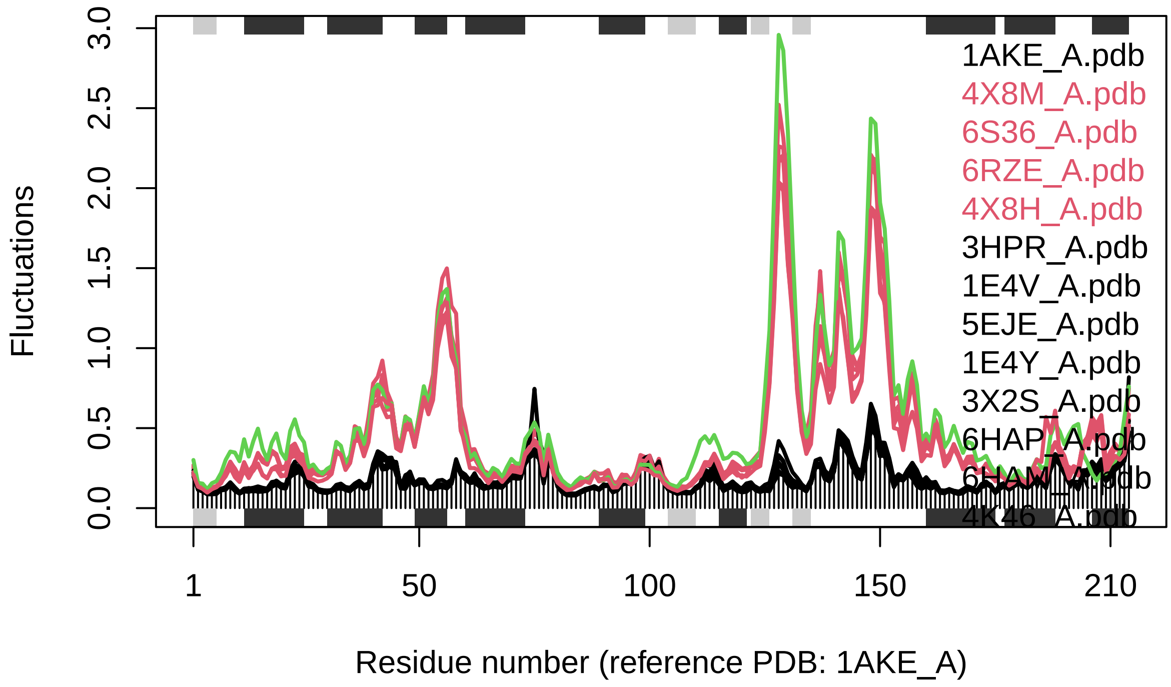


```r
# NMA of all structures
modes <- nma(pdbs)
```

```
##
## 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
##
##   |                                                                      |
```

```r
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?

I note that there is a big fluctuation difference between the colored lines and the black reference line in a couple regions within the above plot. This difference between the the black and colored lines occur around in the residue 30-70 and again at around 130 to 150. I think this difference in fluctuation indicate a potential binding site. The potential reason why the colored lines have a rise in fluctuation in those two regions of the plot may be because of an activation of a binding site as a result of a ligand binding. The reference did not have a change in fluctuation may be because it does not activate due to a binding of a ligand. Nevertheless, this is just a prediction meaning that it is not concrete evidence that it is a binding site. However, now we have a specific target to research to confirm our findings of a potential binding site.