class09

Pham Vo

2/17/2022

```
pdb <- "Data Export Summary.csv"
pdb.df <- read.csv(pdb)
pdb.df
```

```
##            Molecular.Type  X.ray   NMR   EM Multiple.methods Neutron Other
## 1          Protein (only) 144433 11881 6732              182      70    32
## 2 Protein/Oligosaccharide   8543    31 1125                5       0     0
## 3              Protein/NA   7621   274 2165                3       0     0
## 4     Nucleic acid (only)   2396  1399   61                8       2     1
## 5                   Other    150    31    3                0       0     0
## 6  Oligosaccharide (only)     11     6    0                1       0     4
##    Total
## 1 163330
## 2   9704
## 3  10063
## 4   3867
## 5    184
## 6     22
```

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

```
solved.Xray <- sum(pdb.df$X.ray)
solved.EM <- sum(pdb.df$EM)
solved.total <- sum(pdb.df[,2:8])

percent.solved.Xray.EM <- ((solved.Xray + solved.EM)/solved.total)*100

percent.solved.Xray.EM
```

```
## [1] 46.27878
```

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

```
mol.type.protein <- pdb.df[1,8]
mol.type.total <- sum(pdb.df[1:6,8])

prop.mol.type.protein <- (mol.type.protein/mol.type.total)*100
prop.mol.type.protein
```

```
## [1] 87.26292
```

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?

1868

#Introductoion to Bio3D

```
#Load Bio3D package to R
library(bio3d)

#Reading PDB file data into R

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? 198

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

Q9: How many protein chains are in this structure? 2

```
#To find the attributes of any such object:

attributes(pdb)
```

```
## $names
## [1] "atom"   "xyz"    "seqres" "helix"  "sheet"  "calpha" "remark" "call"
##
## $class
## [1] "pdb" "sse"
```

```
# To access these individual attributes we use the dollar-attribute name convention that is common with

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>
```

Comparative structure analysis of Adenylate Kinase

```
# we analyze all currently available Adk structures in the PDB to reveal detailed features and mechanis

# In terms of protein structures PCA can be used to capture major structural variations within a set of
```

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? Grantlab/bio3d-view

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
## fetch the query sequence for chain A of the PDB ID 1AKE
library(bio3d)
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
```

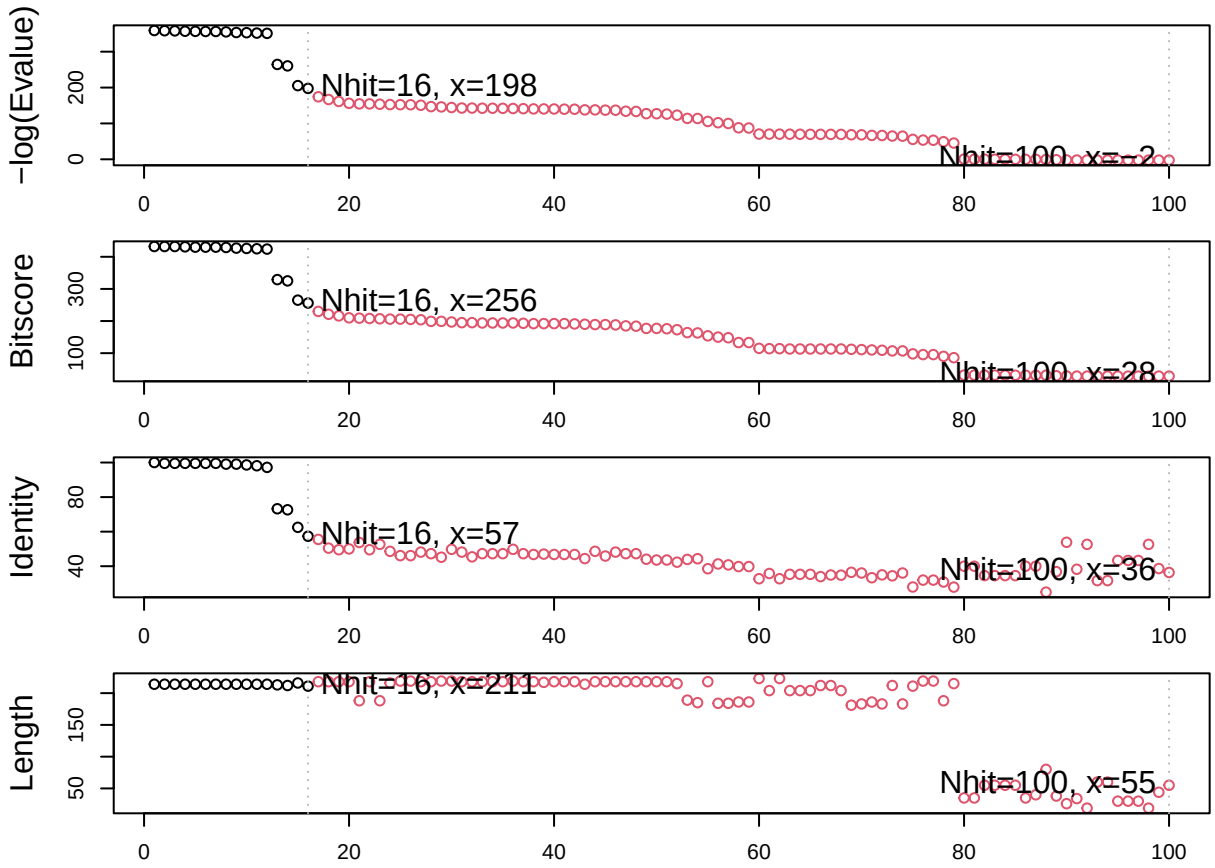
Q13. How many amino acids are in this sequence, i.e. how long is this sequence? 214

```
# use this sequence as a query to BLAST search the PDB to find similar sequences and structures

# Blast or hmmer search
b <- blast.pdb(aa)
```

```
##  Searching ... please wait (updates every 5 seconds) RID = 0Y7GMHHS016
##  .
##  Reporting 100 hits
```

```
# Plot a summary of search results (adjusting the cutoff argument (to plot.blast()) will result in a de
hits <- plot(b)
```

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




```
# List out some 'top hits'
head(hits$pdb.id)
```

```
## [1] "1AKE_A" "4X8M_A" "6S36_A" "6RZE_A" "4X8H_A" "3HPR_A"
```

```
# 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.gz exists. Skipping download
```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

##   |                                                                              |
```

Align and superpose structures

```
# Align releated PDBs
pdbs <- pdbaln(files, fit = TRUE)
```

```
## 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
```

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

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

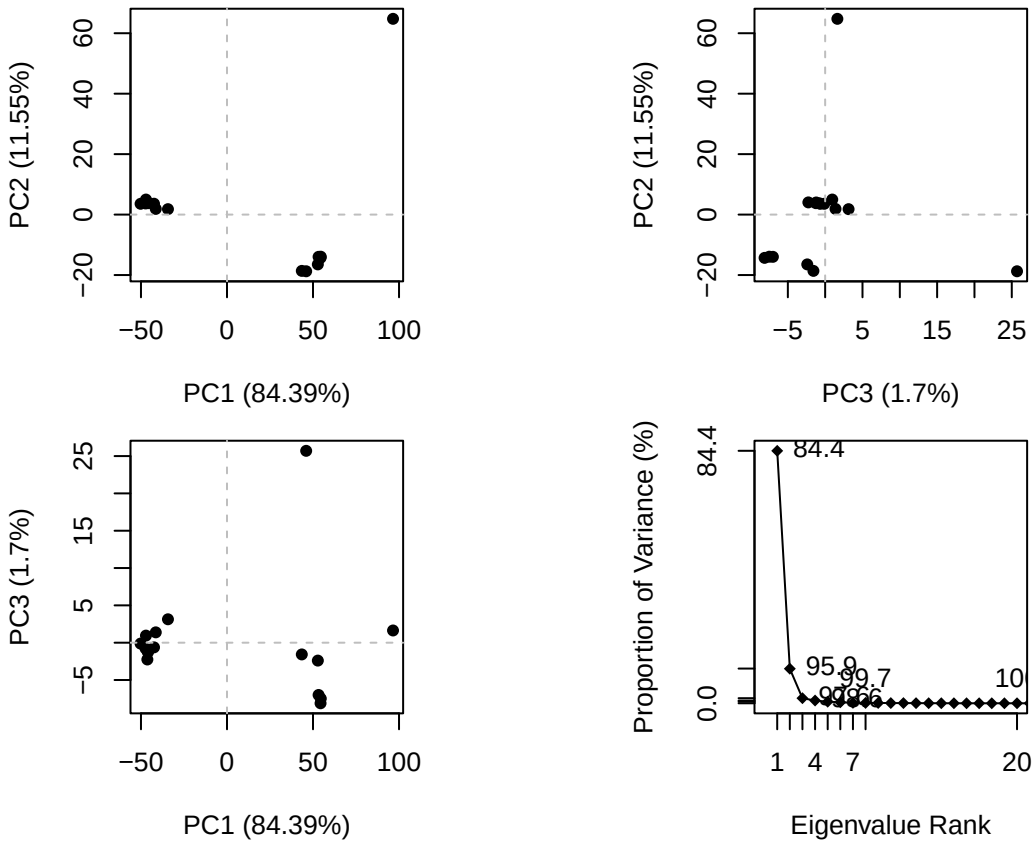
```
#Viewing our superposed structures
#library(bio3d.view)
#library(rgl)
```

```
#view.pdbs(pdbs)
```

```
#Annotate collected PDB structures
#ids <- basename.pdb(pdbs$id)
#anno <- pdb.annotate(as.vector(ids))
#unique(anno$source)

# View all available annotation data
#anno
```

Principal component analysis

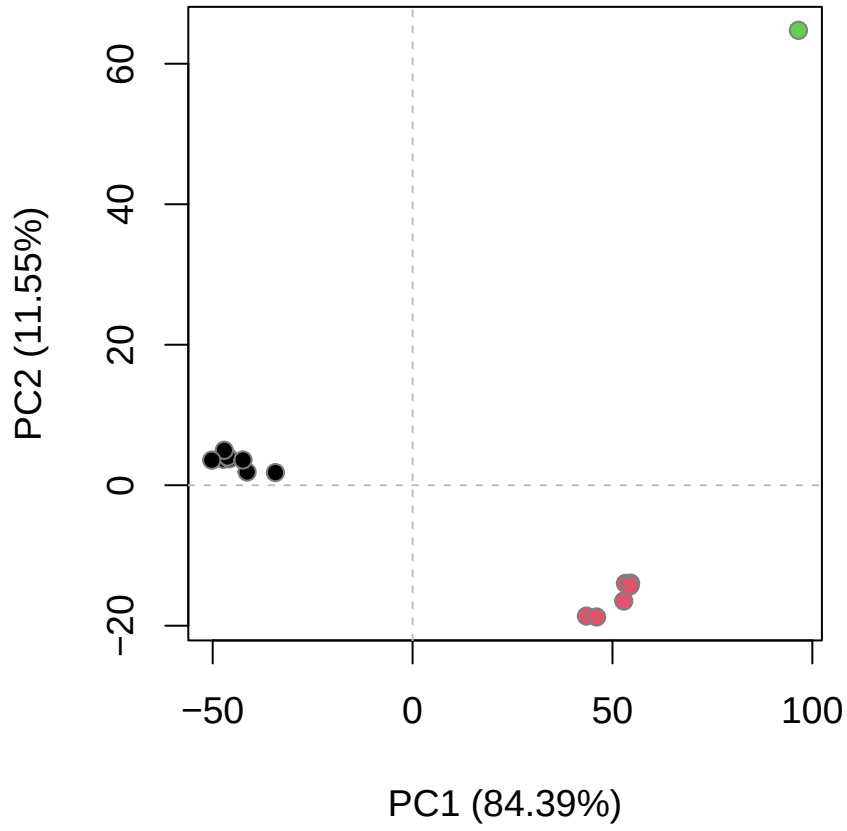
```
# Perform PCA
pc.xray <- pca(pdbs)
plot(pc.xray)
```



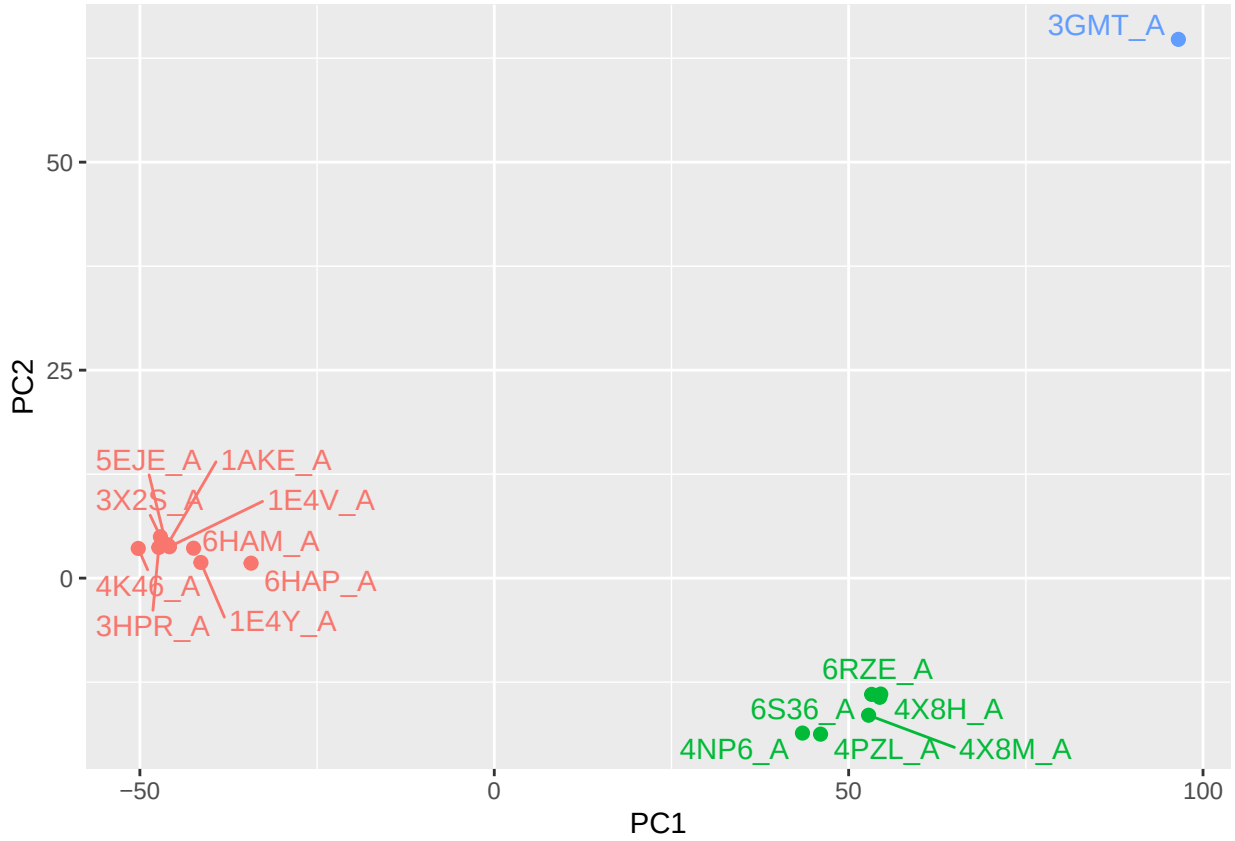

```
# Calculate RMSD
rd <- rmsd(pdbs)
```

```
## 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
```



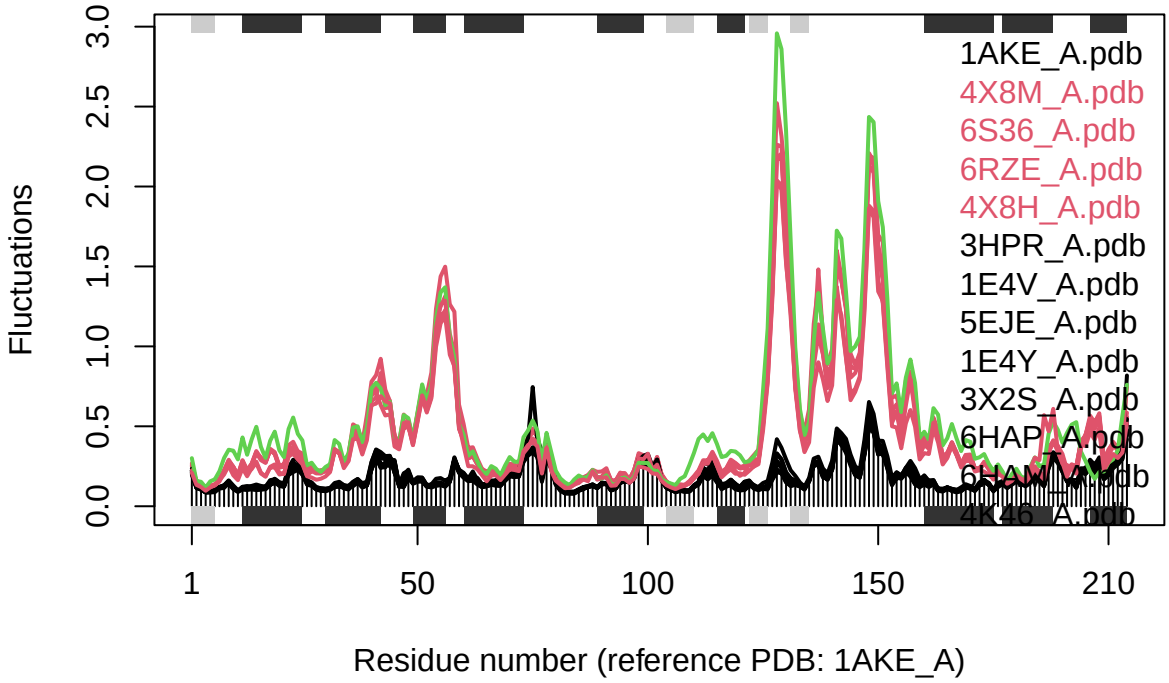
Norrmal mode analysis

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

```
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?

Black and colored lines are different. They are different the most in 2 regions: aa 25-75 and aa 125-175. These could be nucleotide-binding regions, which is essential to be flexible.