Class 12 Structural Bioinformatics II

Cody

11/4/2021

#comparative analysis of protein structures

using the add on bio3d package

library(bio3d)  
  
pdb <- read.pdb("1hel")

## Note: Accessing on-line PDB file

pdb

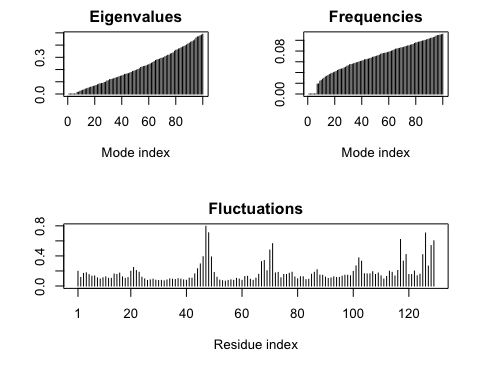
##   
## Call: read.pdb(file = "1hel")  
##   
## Total Models#: 1  
## Total Atoms#: 1186, XYZs#: 3558 Chains#: 1 (values: A)  
##   
## Protein Atoms#: 1001 (residues/Calpha atoms#: 129)  
## Nucleic acid Atoms#: 0 (residues/phosphate atoms#: 0)  
##   
## Non-protein/nucleic Atoms#: 185 (residues: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

lets use a bioinformatics method called NMA (Normal Mode Analysis) to predict the dynamics (flexibility) of this enzyme

modes <- nma(pdb)

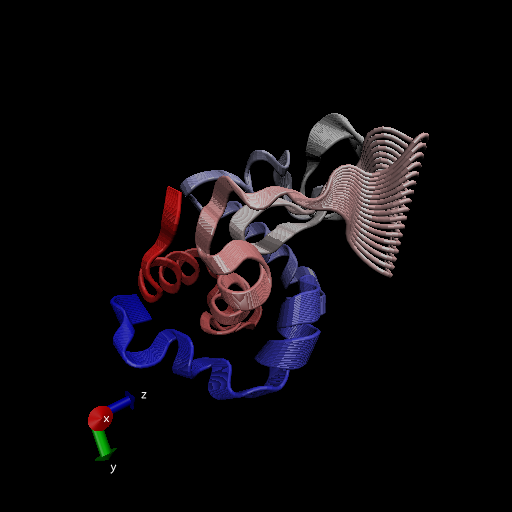
## Building Hessian... Done in 0.037 seconds.  
## Diagonalizing Hessian... Done in 0.061 seconds.

plot(modes)



Make a “move” of its predicted motion. We often call this a “trajectory”

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



aa <- get.seq("1ake\_A")

## Warning in get.seq("1ake\_A"): Removing existing file: seqs.fasta

## Fetching... Please wait. Done.

aa

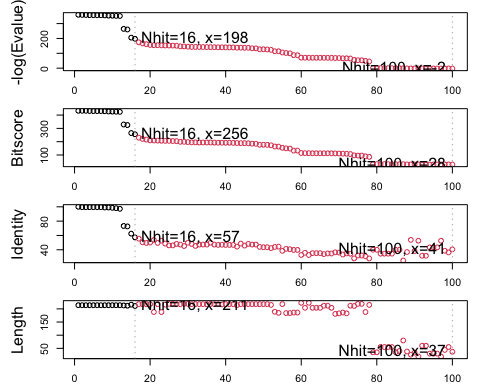
## 1 . . . . . 60   
## pdb|1AKE|A MRIILLGAPGAGKGTQAQFIMEKYGIPQISTGDMLRAAVKSGSELGKQAKDIMDAGKLVT  
## 1 . . . . . 60   
##   
## 61 . . . . . 120   
## pdb|1AKE|A DELVIALVKERIAQEDCRNGFLLDGFPRTIPQADAMKEAGINVDYVLEFDVPDELIVDRI  
## 61 . . . . . 120   
##   
## 121 . . . . . 180   
## pdb|1AKE|A VGRRVHAPSGRVYHVKFNPPKVEGKDDVTGEELTTRKDDQEETVRKRLVEYHQMTAPLIG  
## 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 BLAST  
blast <- blast.pdb(aa)

## Searching ... please wait (updates every 5 seconds) RID = SJY4DSUG013   
## ....  
## Reporting 100 hits

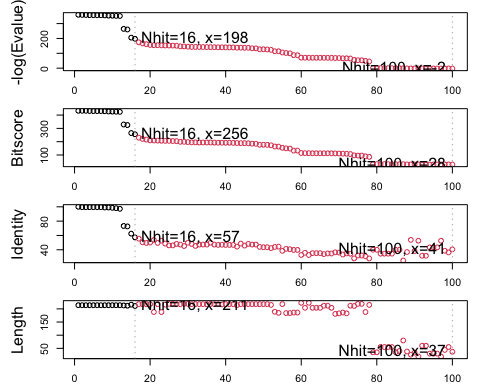
plot(blast)

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



hits <- plot(blast)

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



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

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

#Align and superpose structures multiple structure alignment

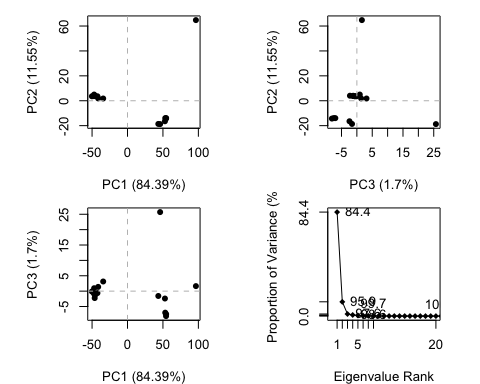
pdbs<- pdbaln(files, fit=TRUE)

## 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

#PCA

we will use the bio3d pca() function which is designed for protein structure data

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



Make a trajectory visualization of the motion captured by the first Princial Component

# Visualize first principal component  
pc1 <- mktrj(pc.xray, pc=1, file="pc\_1.pdb")

