Homework 6

Mia Fava

2024-10-20

hc <- hclust(dist(rbind(s1.b, s2.b, s3.b))) plot(hc)

> Q6. How would you generalize the original code above to work with any set of input protein structures?

-First need to download the following packages:

```
library(bio3d)
library(cluster)
```

Inputs:

-pdb_files: character vector for names of PDB files (4AKE: Kinase with drug bound,1AKE: Kinase without drug, 1E4Y: Another kinase with a drug)

-chain : chain identifier

-elety : type of atom

Function works like this: it reads multiple of the PDB files to then extract the B-factors from a specified chain and atom type. Then, hierarchial clustering will occur to produce a dendrogram plot, which will give the output where the proteins are clustered based on their similiarity in B-factor data.

```
analyze_and_cluster_bfactors<- function(pdb_files, chain = "A", elety = "CA") {
  #step 1: extract B-factors from all PDB files
  b_factors <- lapply(pdb_files, function(file) {
    pdb_data <- read.pdb(file)
    chain_data <- trim.pdb(pdb_data, chain = chain, elety = elety)
    return(chain_data$atom$b)  # return the B-factor values
    })

  #step 2: combine the B-factors and calculate distance matrix
  b_matrix <- do.call(rbind, b_factors)
  dist_matrix <- dist(b_matrix)

  #step 3: perform hierarchical clustering
  hc <- hclust(dist_matrix)

  #step 4: plot the dendrogram
  plot(hc, main = "Hierarchical Clustering of Protein B-factors",
       xlab = "Proteins", sub = "", ylab = "Distance")
}
```

-lapply() is used to avoid calculations that are redundant through stacking the B-factor matrix

-do.call() stacks B-factor vectors with no redundant loops for clustering of data. both allow for less recalculations.

```
# Define the PDB files to compare
pdb_files <- c("4AKE", "1AKE", "1E4Y")
result <- analyze_and_cluster_bfactors(pdb_files)
```

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