HW6

Chan-yu Kuo

2023-01-30

```
## Question: Can you improve this code?
df <- data.frame(a=1:10, b=seq(200,400,length=10),c=11:20,d=NA)
df$a <- (df$a - min(df$a)) / (max(df$a) - min(df$a))
df$b <- (df$b - min(df$a)) / (max(df$b) - min(df$b))
df$c <- (df$c - min(df$c)) / (max(df$c) - min(df$c))
df$d <- (df$d - min(df$d)) / (max(df$a) - min(df$d))
df$d
```

```
##  [1] NA NA NA NA NA NA NA NA NA NA
```

```
first_question<- function(a,b,c,d) {
  return((a - min(b)) / (max(c) - min(d)))
}
df <- data.frame(a=1:10, b=seq(200,400,length=10),c=11:20,d=NA)
df$a <-first_question(df$a,df$a,df$a,df$a)
df$b <-first_question(df$b,df$a,df$b,df$b)
df$c <-first_question(df$c,df$c,df$c,df$c)
df$d <-first_question(df$d,df$d,df$a,df$d)
```

```
#install.packages("bio3d")
```

```
# Can you improve this analysis code?
library(bio3d)
s1 <- read.pdb("4AKE") # kinase with drug
```

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

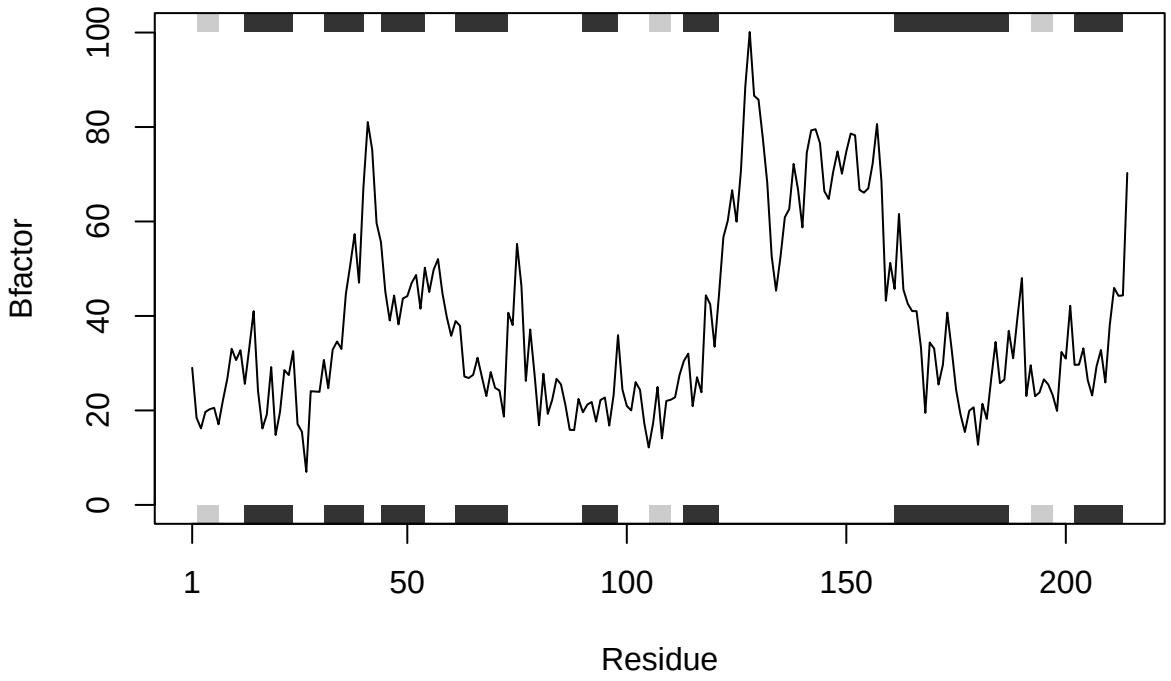
```
s2 <- read.pdb("1AKE") # kinase no drug
```

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

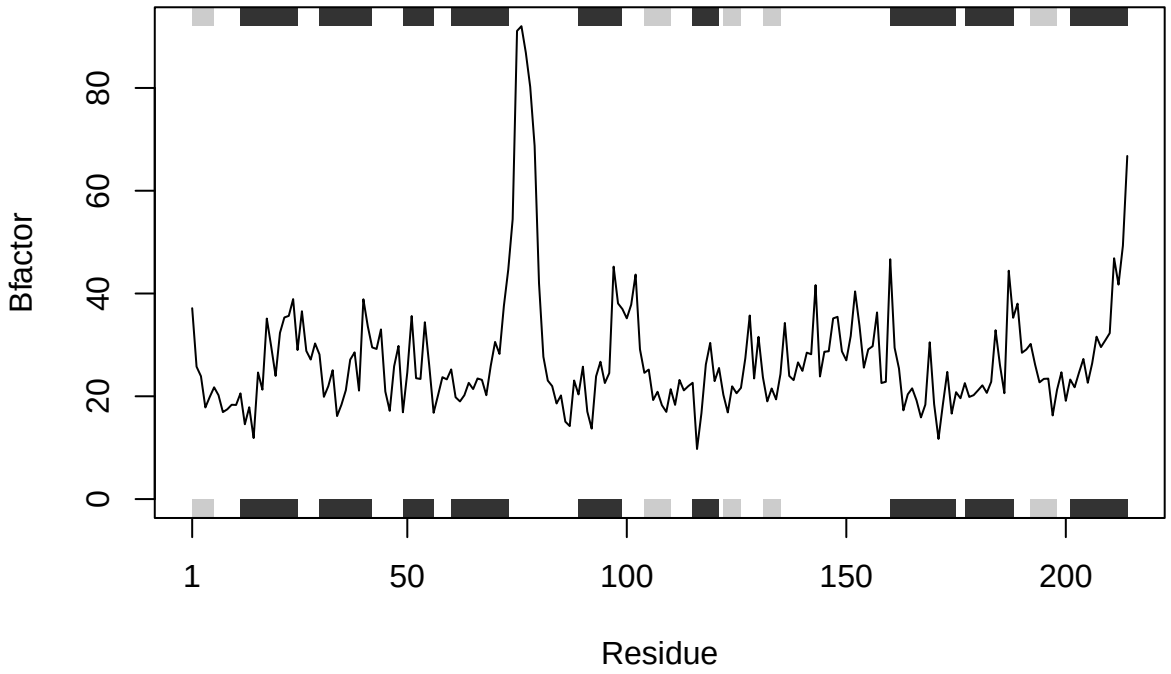
```
s3 <- read.pdb("1E4Y") # kinase with drug
```

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

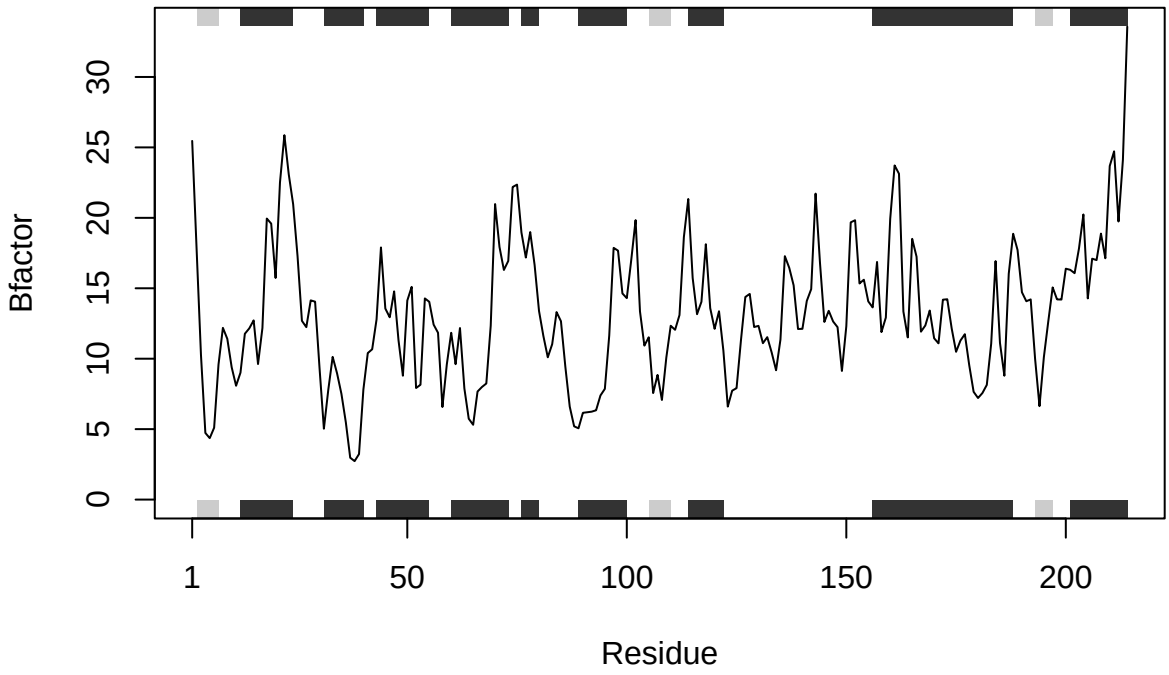
```
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s3, chain="A", elety="CA")
s1.b <- s1.chainA$atom$b
s2.b <- s2.chainA$atom$b
s3.b <- s3.chainA$atom$b
plotb3(s1.b, sse=s1.chainA, typ="l", ylab="Bfactor")
```




```
plotb3(s2.b, sse=s2.chainA, typ="l", ylab="Bfactor")
```




```
plotb3(s3.b, sse=s3.chainA, typ="l", ylab="Bfactor")
```



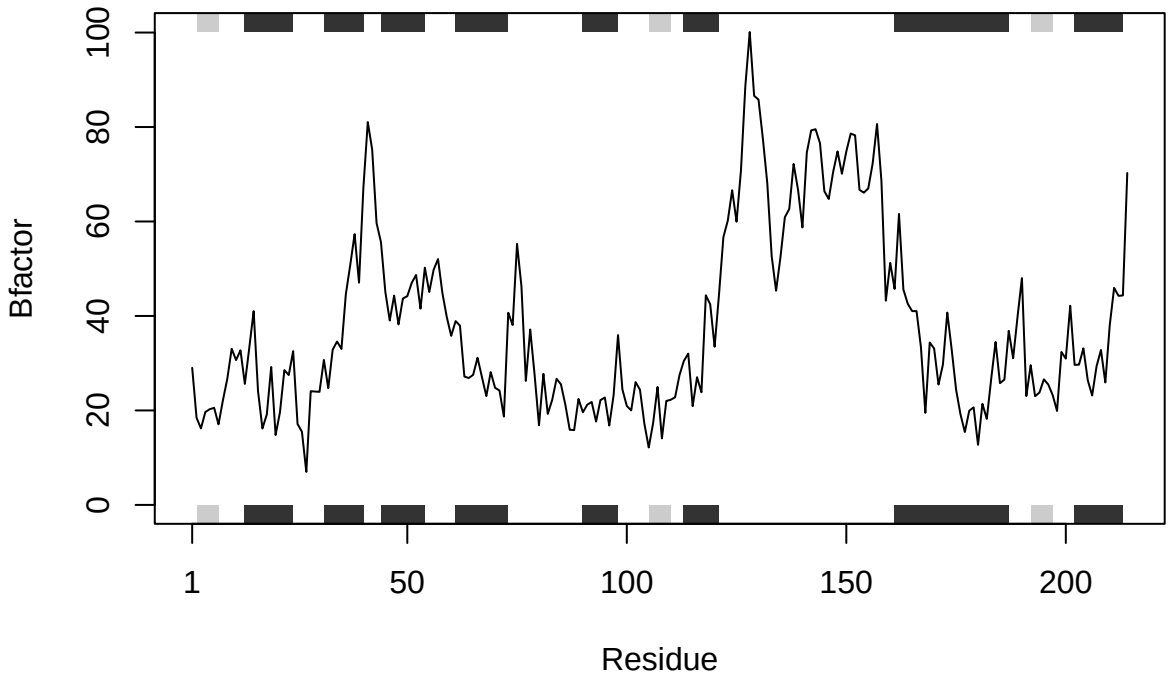

```
second_question<- function(str1,str2,str3){
  vector = c(str1,str2,str3)
  for (individual_string in vector){
    s1 <- read.pdb(individual_string)
    s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
    s1.b <- s1.chainA$atom$b
    plotb3(s1.b, sse=s1.chainA, typ="l", ylab="Bfactor")
  }

}

second_question("4AKE","1AKE","1E4Y")
```

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

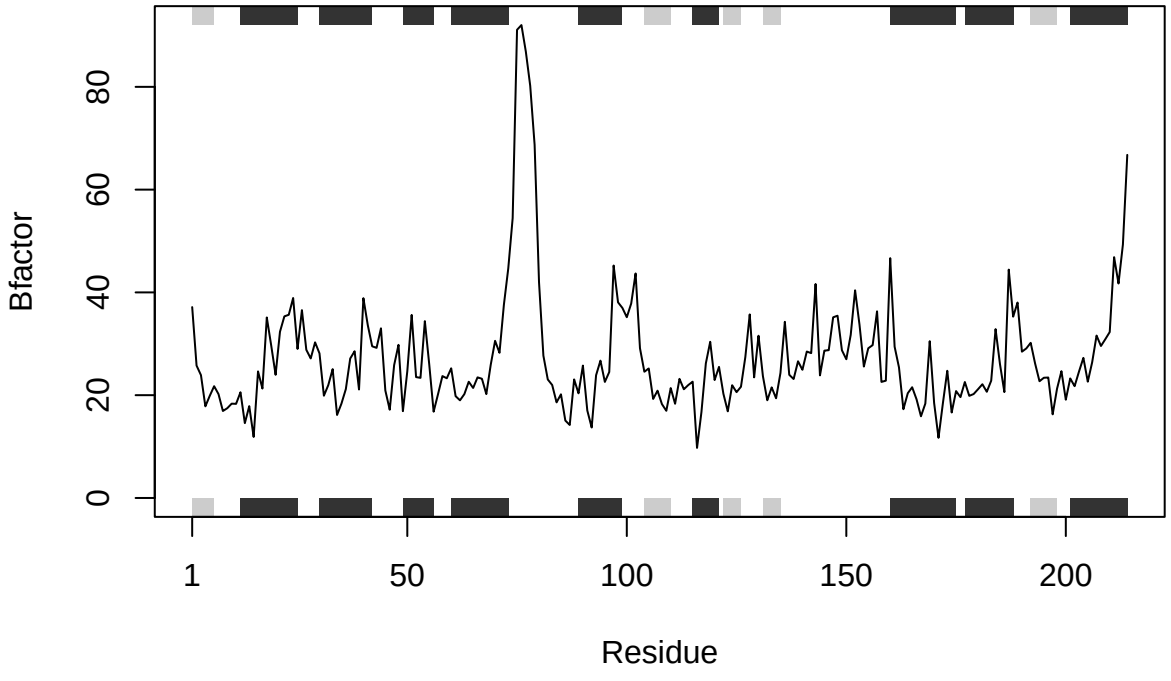
```
## Warning in get.pdb(file, path = tempdir(), verbose =
## FALSE): /var/folders/b3/c05kjh0d3q70tnfmlv9gpy800000gn/T//Rtmp67fWEy/4AKE.pdb
## exists. Skipping download
```

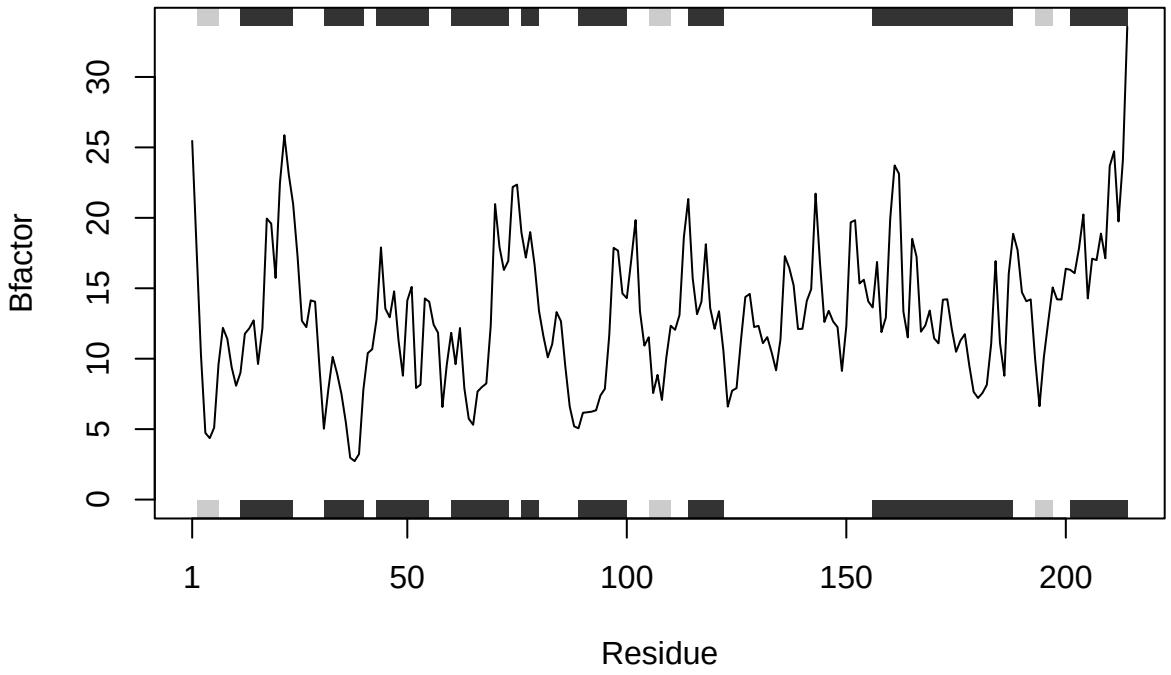
```
##   Note: Accessing on-line PDB file
```

```
## Warning in get.pdb(file, path = tempdir(), verbose =
## FALSE): /var/folders/b3/c05kjh0d3q70tnfmlv9gpy800000gn/T//Rtmp67fWEy/1AKE.pdb
## exists. Skipping download
```




```
##    PDB has ALT records, taking A only, rm.alt=TRUE
##   Note: Accessing on-line PDB file
## Warning in get.pdb(file, path = tempdir(), verbose =
## FALSE): /var/folders/b3/c05kjh0d3q70tnfmlv9gpy800000gn/T//Rtmp67fWEy/1E4Y.pdb
## exists. Skipping download
```






```
s1 <- read.pdb("4AKE")
```

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

```
## Warning in get.pdb(file, path = tempdir(), verbose =
## FALSE): /var/folders/b3/c05kjh0d3q70tnfmlv9gpy800000gn/T//Rtmp67fWEy/4AKE.pdb
## exists. Skipping download
```

```
str(s1)
```

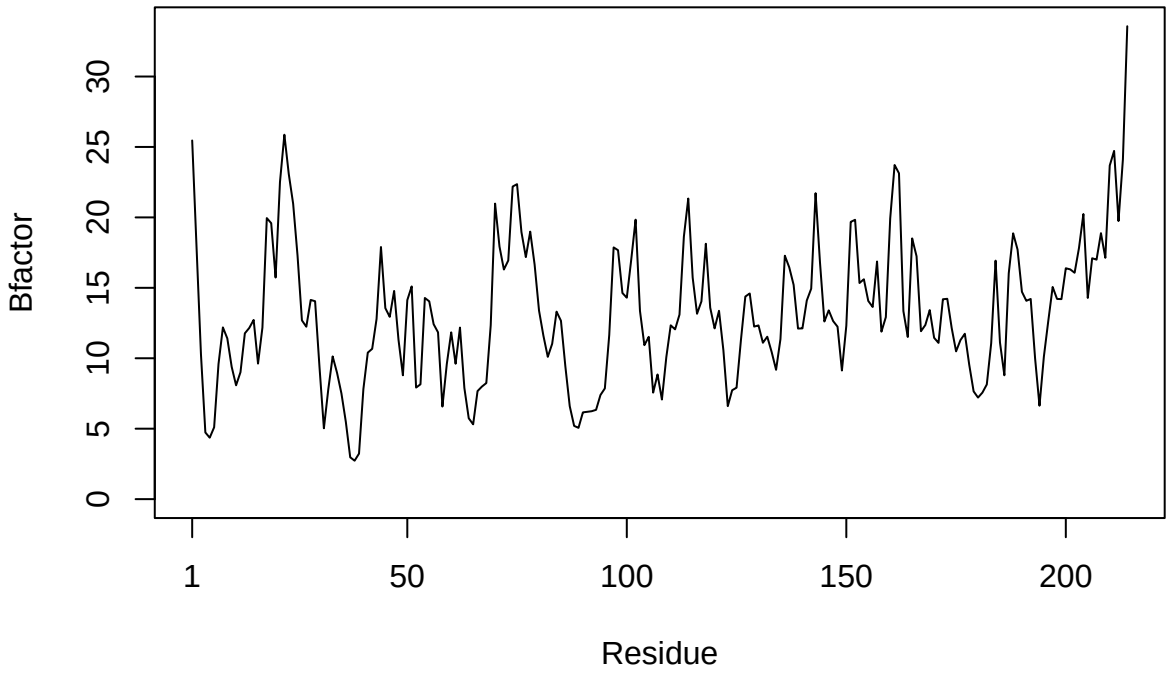
```
## List of 8
##  $ atom  :'data.frame':  3459 obs. of  16 variables:
##   ..$ type  : chr [1:3459] "ATOM" "ATOM" "ATOM" "ATOM" ...
##   ..$ eleno : int [1:3459] 1 2 3 4 5 6 7 8 9 10 ...
##   ..$ elety : chr [1:3459] "N" "CA" "C" "O" ...
##   ..$ alt   : chr [1:3459] NA NA NA NA ...
##   ..$ resid : chr [1:3459] "MET" "MET" "MET" "MET" ...
##   ..$ chain : chr [1:3459] "A" "A" "A" "A" ...
##   ..$ resno : int [1:3459] 1 1 1 1 1 1 1 2 2 ...
##   ..$ insert: chr [1:3459] NA NA NA NA ...
##   ..$ x     : num [1:3459] -10.93 -9.9 -9.17 -9.8 -10.59 ...
##   ..$ y     : num [1:3459] -24.9 -24.4 -23.3 -22.3 -24 ...
##   ..$ z     : num [1:3459] -9.52 -10.48 -9.81 -9.35 -11.77 ...
##   ..$ o     : num [1:3459] 1 1 1 1 1 1 1 1 1 1 ...
##   ..$ b     : num [1:3459] 41.5 29 27.9 26.4 34.2 ...
##   ..$ segid : chr [1:3459] NA NA NA NA ...
##   ..$ elesy : chr [1:3459] "N" "C" "C" "O" ...
##   ..$ charge: chr [1:3459] NA NA NA NA ...
##  $ xyz   : 'xyz' num [1, 1:10377] -10.93 -24.89 -9.52 -9.9 -24.42 ...
##  $ seqres: Named chr [1:428] "MET" "ARG" "ILE" "ILE" ...
##   ..- attr(*, "names")= chr [1:428] "A" "A" "A" "A" ...
##  $ helix :List of 4
##   ..$ start: Named num [1:19] 13 31 44 61 75 90 113 161 202 13 ...
##   .. ..- attr(*, "names")= chr [1:19] "" "" "" "" ...
```

```
##   ..$ end  : Named num [1:19] 24 40 54 73 77 98 121 187 213 24 ...
##   .. ..- attr(*, "names")= chr [1:19] "" "" "" "" ...
##   ..$ chain: chr [1:19] "A" "A" "A" "A" ...
##   ..$ type : chr [1:19] "5" "1" "1" "1" ...
##  $ sheet :List of 4
##   ..$ start: Named num [1:14] 192 105 2 81 27 123 131 192 105 2 ...
##   .. ..- attr(*, "names")= chr [1:14] "" "" "" "" ...
##   ..$ end  : Named num [1:14] 197 110 7 84 29 126 134 197 110 7 ...
##   .. ..- attr(*, "names")= chr [1:14] "" "" "" "" ...
##   ..$ chain: chr [1:14] "A" "A" "A" "A" ...
##   ..$ sense: chr [1:14] "0" "1" "1" "1" ...
##  $ calpha: logi [1:3459] FALSE TRUE FALSE FALSE FALSE FALSE ...
##  $ remark:List of 1
##   ..$ biomat:List of 4
##   .. ..$ num   : int 1
##   .. ..$ chain :List of 1
##   .. .. ..$ : chr [1:2] "A" "B"
##   .. ..$ mat   :List of 1
##   .. .. ..$ :List of 1
##   .. .. .. ..$ A B: num [1:3, 1:4] 1 0 0 0 1 0 0 0 1 0 ...
##   .. ..$ method: chr "AUTHOR"
##  $ call  : language read.pdb(file = "4AKE")
##  - attr(*, "class")= chr [1:2] "pdb" "sse"
```

```
## Q1 based on the output, the object is data.frame called atom (or PDB structure object)
```

```
?trim.pdb()
## Q2 based on the output, trim.pdb() select a subset from the large PDB obejct, with specified paramet
```

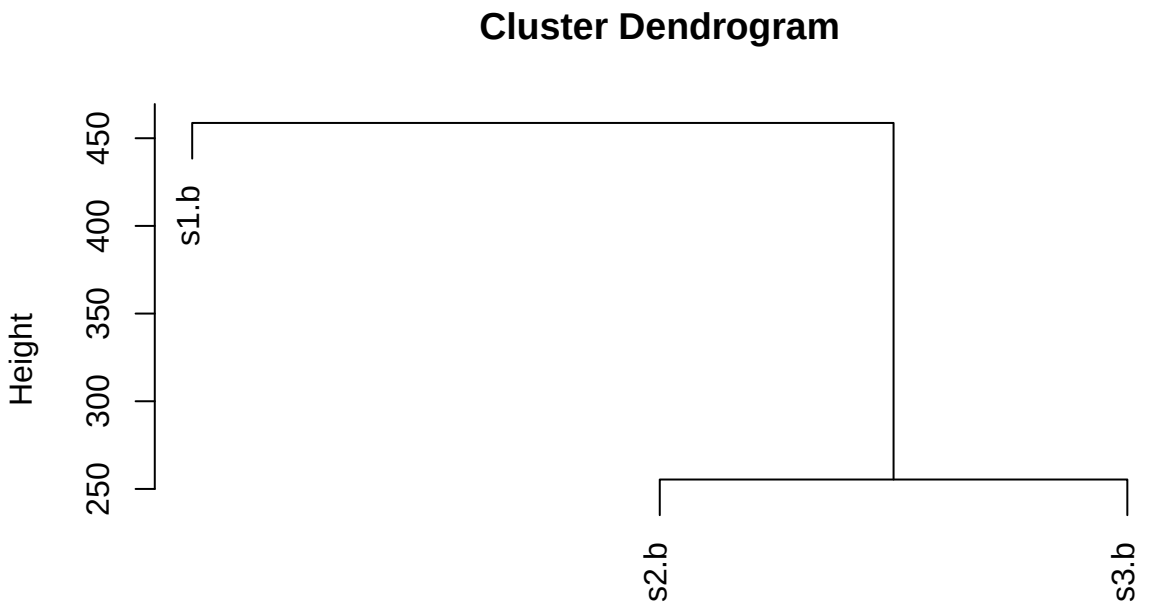
```
plotb3(s3.b, typ="l", ylab="Bfactor")
```




```
?plotb3
## Q3: sse=Null would turn off the marginal black and grey rectangles. sse represent the secondary strc
```

```
### Q4 use facet in ggplot2 to combine several plots into one. This way we can compare multiple protein
```

```
hc <- hclust( dist( rbind(s1.b, s2.b, s3.b) ) )
## rbind (combine rows into one)
## dist compute the distance between any two rows of the matrix
## hclust input a dissimilarity structure produced by dist. (Meaning, the clustering is based on the di
plot(hc)
```



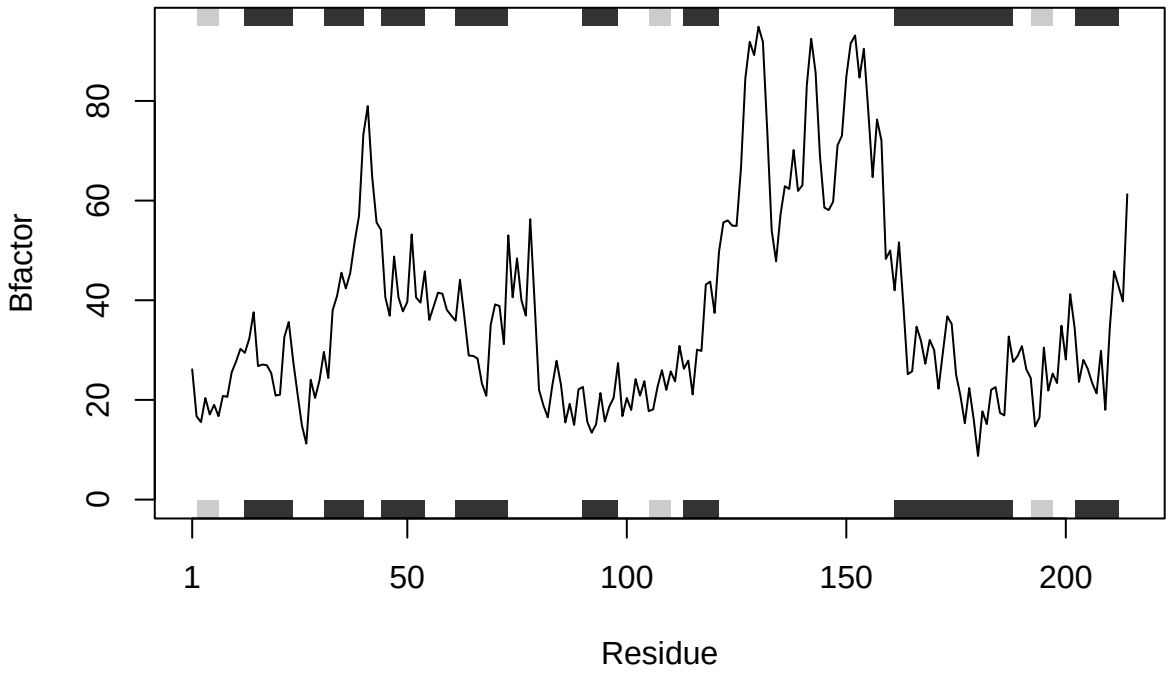

```
## Based on Cluster Dendrogram, s2.b and s3.b have highest similarity with least distance in between.
```

```
## Question 6

sixth_question<- function(protein_name,chain_name){
  kinase <- read.pdb(protein_name)
  kinase.chainA <- trim.pdb(kinase, chain=chain_name, elety="CA")
  kinase.b <- kinase.chainA$atom$b
  plotb3(kinase.b, sse=kinase.chainA, typ="l", ylab="Bfactor")
}
### The input of the function sixth_question are protein_name and chain_name.
### The function will read the data base based on the given protein_name, trim the data based on the ch
### The output of a function is to plot out a graph that shows the Bfactor across the residue of the pr
```

```
## Example
sixth_question("4AKE","B")
```

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

```
## Warning in get.pdb(file, path = tempdir(), verbose =
## FALSE): /var/folders/b3/c05kjh0d3q70tnfmlv9gpy800000gn/T//Rtmp67fWEy/4AKE.pdb
## exists. Skipping download
```


Bfactor
0
20
40
60
80
1
50
100
150
200
Residue