HW6

Hanhee Jo

Can you improve this analysis code?

```
library(bio3d)
```

```
Warning: package 'bio3d' was built under R version 4.3.3
```

```
s1 <- read.pdb("4AKE") # kinase with drug
```

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

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

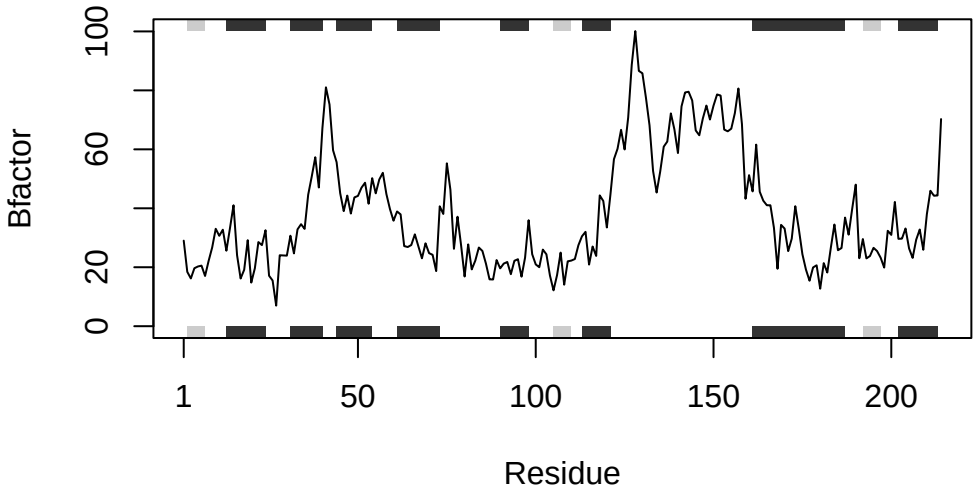
```
  Note: Accessing on-line PDB file
   PDB has ALT records, taking A only, rm.alt=TRUE
```

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

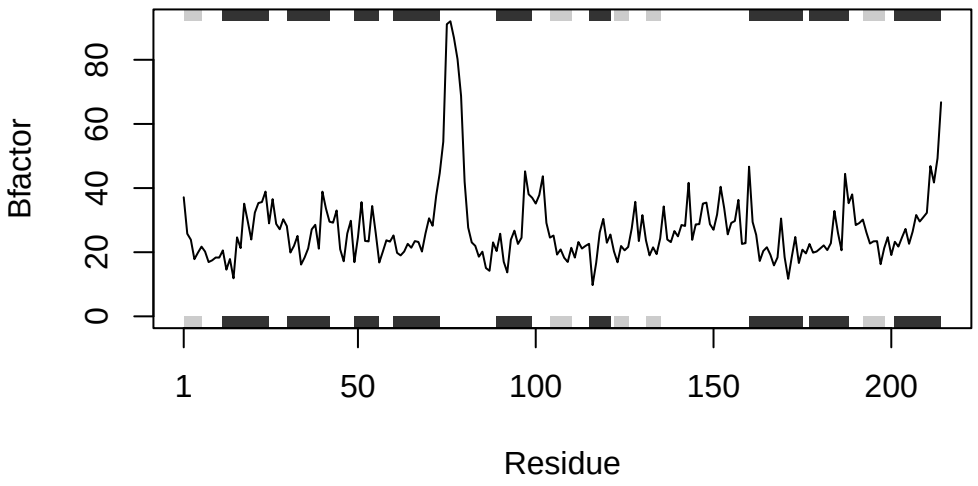
```
  Note: Accessing on-line PDB file
```

```
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
```

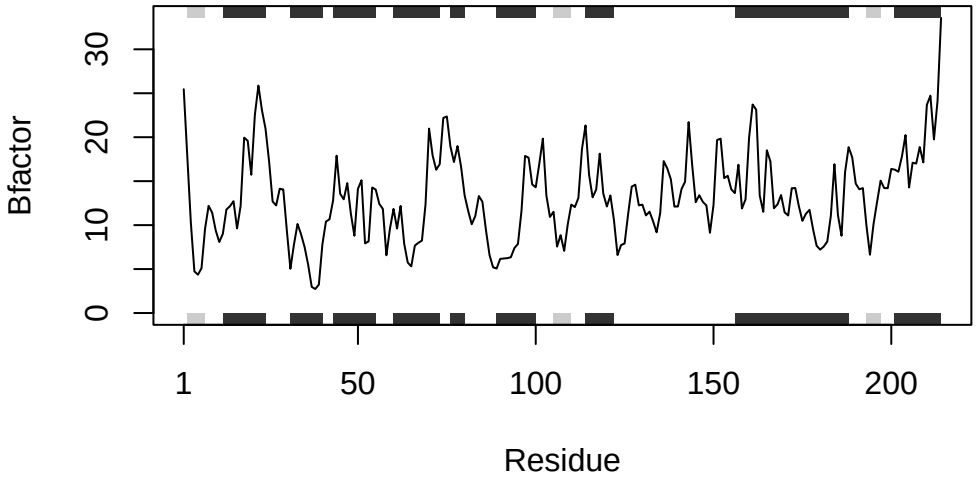
```
s3.b <- s3.chainA$atom$b
plotb3(s1.b, sse=s1.chainA, typ="l", ylab="Bfactor")
```




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




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



Q1. What type of object is returned from the read.pdb() function?

The `read.pdb()` function in the Bio3D R package returns an object of class **"pdb"**. This object is essentially a list that contains several components extracted from the PDB file, which represent different aspects of the structure.

Q2. What does the trim.pdb() function do?

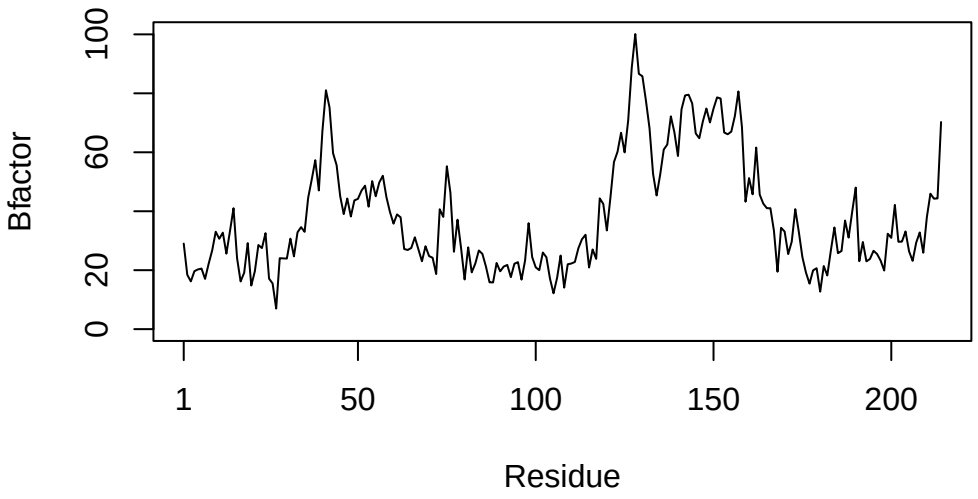
The `trim.pdb()` function is used to extract specific subsets of information from a PDB structure object. This is particularly useful when you only want to work with a specific chain, atom type, residue range, or other structural elements, instead of the entire structure.

Q3. What input parameter would turn off the marginal black and grey rectangles in the plots and what do they represent in this case?

The marginal black and grey rectangles in the plots represent *secondary structure elements (SSEs)* such as helices and beta strands along the protein sequence.

To turn off these rectangles, you can use the parameter sse and set it to **NULL** in the `plotb3()` function:

```
plotb3(s1.b, sse = NULL, typ = "l", ylab = "Bfactor")
```



Q4. What would be a better plot to compare across the different proteins?

Q5. Which proteins are more similar to each other in their B-factor trends. How could you quantify this?

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

```
# Define PDB files and their descriptions
pdb_files <- list(
  list(id="4AKE", desc="kinase with drug"),
  list(id="1AKE", desc="kinase no drug"),
  list(id="1E4Y", desc="kinase with drug")
)

# Function to process a single PDB structure
process_pdb <- function(pdb_id) {
  # Read PDB file
  structure <- read.pdb(pdb_id)
  # Trim to chain A and CA atoms
  chain_a <- trim.pdb(structure, chain="A", elety="CA")
  # Extract B-factors
  b_factors <- chain_a$atom$b

  return(list(
    chain=chain_a,
    bfac=b_factors
  ))
}

# Process all structures
structures <- lapply(pdb_files, function(file) {
  result <- process_pdb(file$id)
  result$desc <- file$desc
  return(result)
})
```

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

```
Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
/var/folders/nh/cp0mpgtn4r332hmc98mh5h280000gn/T//Rtmp7M97b1/4AKE.pdb exists.
Skipping download
```

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

```
Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
/var/folders/nh/cp0mpgtn4r332hmc98mh5h280000gn/T//Rtmp7M97b1/1AKE.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/nh/cp0mpgtn4r332hmc98mh5h280000gn/T//Rtmp7M97b1/1E4Y.pdb exists.
Skipping download
```

```r
# Plot B-factors for all structures
par(mar=c(4,4,2,2))
par(mfrow=c(3,1))  # Set up 3x1 plot layout
for(i in seq_along(structures)) {
  plotb3(structures[[i]]$bfac,
         sse=structures[[i]]$chain,
         typ="l",
         ylab="Bfactor",
         main=paste(pdb_files[[i]]$id, "-", pdb_files[[i]]$desc))
}
```