Class6 HW Q6, R Functions

Xinyu Wen (PID:A17115443)

Table of contents

Original code:

```
library(bio3d)
s1 <- read.pdb("4AKE") # kinase with drug
```

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

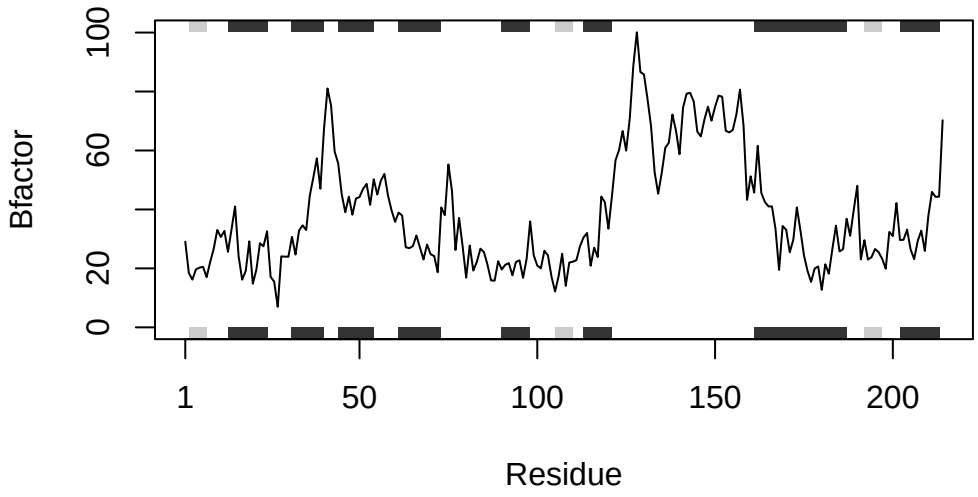
```
s2 <- read.pdb("1AKE") # kinase no drug
```

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

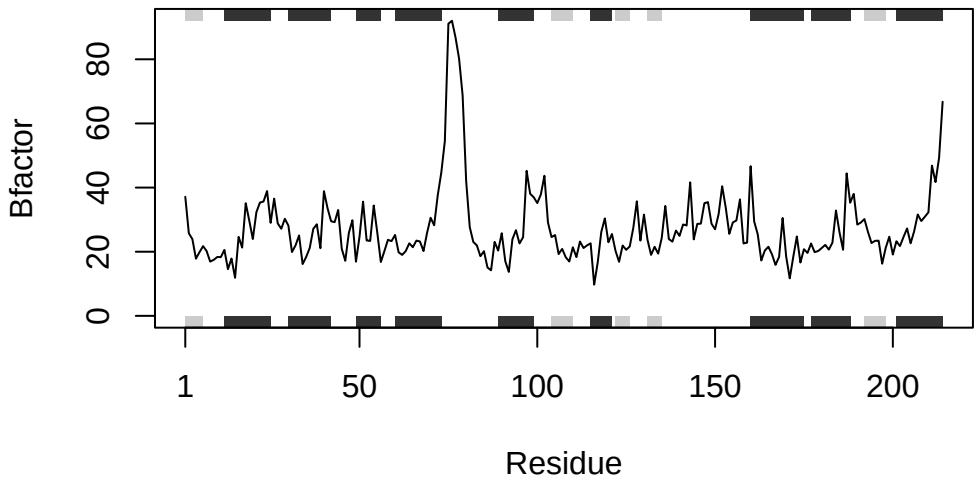
```
s3 <- read.pdb("1E4Y") # kinase with drug
```

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

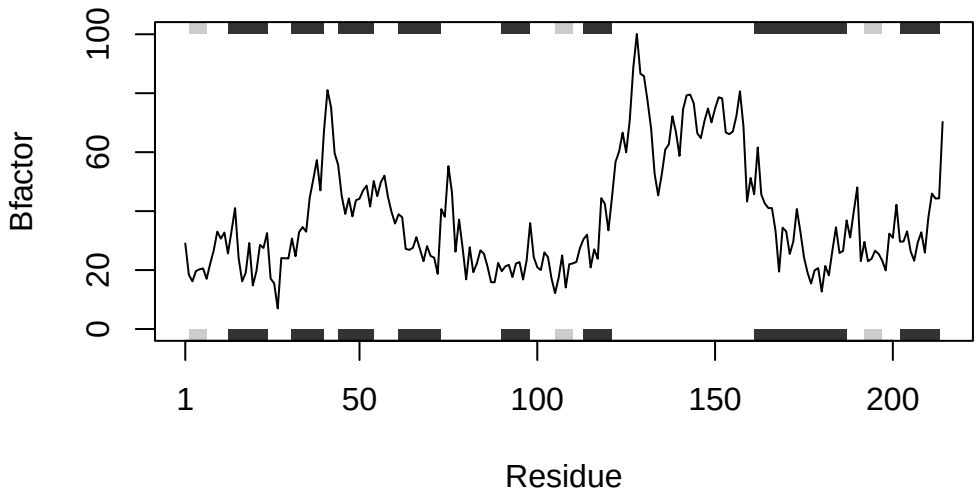
```
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s1, chain="A", elety="CA")
s1.b <- s1.chainA$atom$b
s2.b <- s2.chainA$atom$b
s3.b <- s3.chainA$atom$b
plotb3(s1.b, sse=s1.chainA, typ="l", ylab="Bfactor")
```




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



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



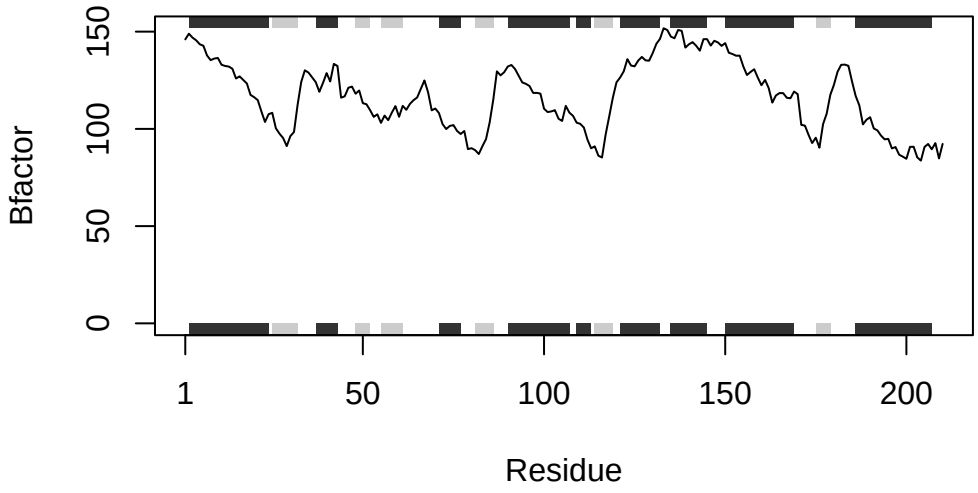
Q6.

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

```
library(bio3d)
proteinname <- c("9BHM")
anyproteinplot <- function(proteinname) {
  s <- read.pdb(proteinname)
  s.chainA <- trim.pdb(s, chain="A", elety="CA")
  s.b <- s.chainA$atom$b
  plotb3(s.b, sse=s.chainA, typ="l", ylab="Bfactor")
}
```

```
anyproteinplot("9BHM")
```

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



The input of the function is the PDB code of the protein of interest.

The function first reads the protein file from the bio3d library. Then it trims the PDB object into a new smaller PDB object with a subset of atoms. Then it plots the trimmed PDB object as a standard scatter plot.

To use the function, input/replace the PDB code after "proteinname".

The output of the function is the scatter plot of the protein of interest, with the B factor vs Residue.