class 06: homework function

jack olmstead

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

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

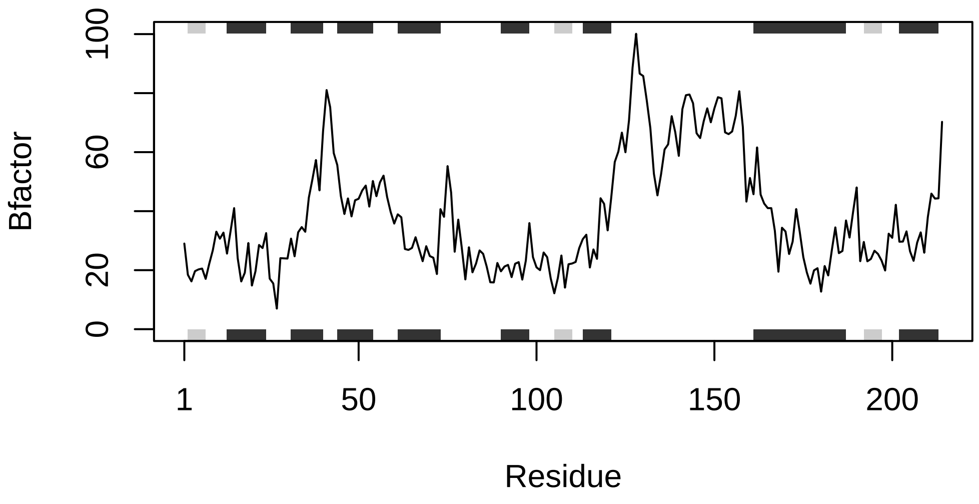
```r
s2 <- read.pdb("1AKE") # kinase no drug
```

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

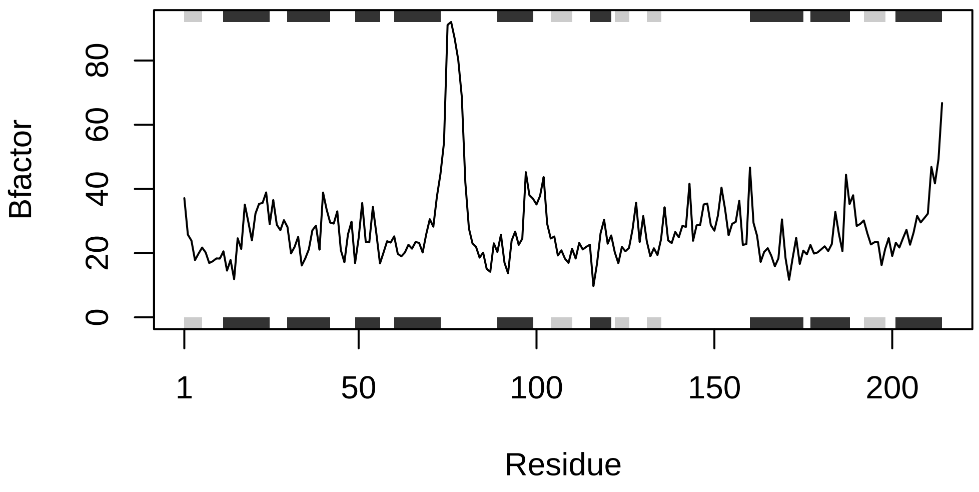
```r
s3 <- read.pdb("1E4Y") # kinase with drug
```

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

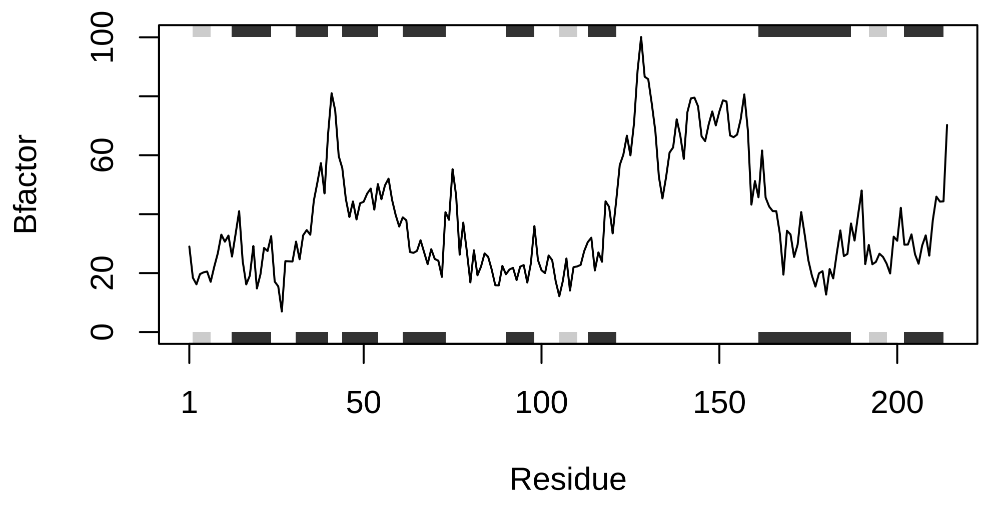
```r
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s1, chain="A", elety="CA")
s1.b <- s1.chainA$atom$b
s2.b <- s2.chainA$atom$b
s3.b <- s3.chainA$atom$b
plotb3(s1.b, sse=s1.chainA, typ="l", ylab="Bfactor")
```



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



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



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

```r
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 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"
```

s1 is a "Large pdb" object, which is a list of 8 elements. It is the PDB structure.

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

trim.pdb() creates a smaller pdb object from the subset of atoms in a larger pdb object.

> 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 second, "sse" parameter turns off the marginal black/grey rectangles. They represent secondary structures that span contiguous residues.

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

To compare B-factor trends, it would be most helpful to have all the lines on one plot

> Q5. Which proteins are more similar to each other in their B-factor trends. How could you quantify this? HINT: try the rbind(), dist() and hclust() functions together with a resulting dendrogram plot. Look up the documentation to see what each of these functions does.

Let's make this into a function!

```r
library(bio3d)

#' Title
#'
#' @param id The PDB code for the structure of interest.
#'
#' @return A plot of B-factor values
#' @export
#'
#' @examples
plotPdb <- function(id) {

  # get structure from PDB using identifier
  struc <- read.pdb(id)

  # trim and plot structure
  struc.chainA <- trim.pdb(struc, chain="A", elety="CA")

  plotb3(struc.chainA$atom$b, sse=struc.chainA, typ="l", ylab="Bfactor")
}
```

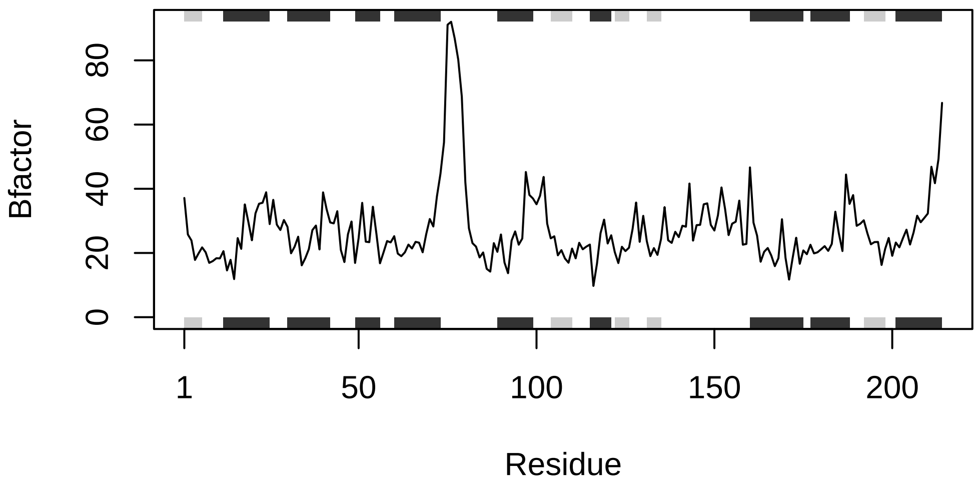
Try using the function

```r
plotPdb("1AKE")
```

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

```
Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
C:\Users\User\AppData\Local\Temp\RtmpqOLUzj/1AKE.pdb exists. Skipping download
```

```
   PDB has ALT records, taking A only, rm.alt=TRUE
```


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