Class 06 HW

Nicole Alfonso

2024-02-04

Section 1: Improving analysis code by writing functions

B.Can you improve this analysis code?

We are given a set of PBD files - protein data bases. The data bases are separated into three files - s1, s2, and s3.

```
library(bio3d)

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

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

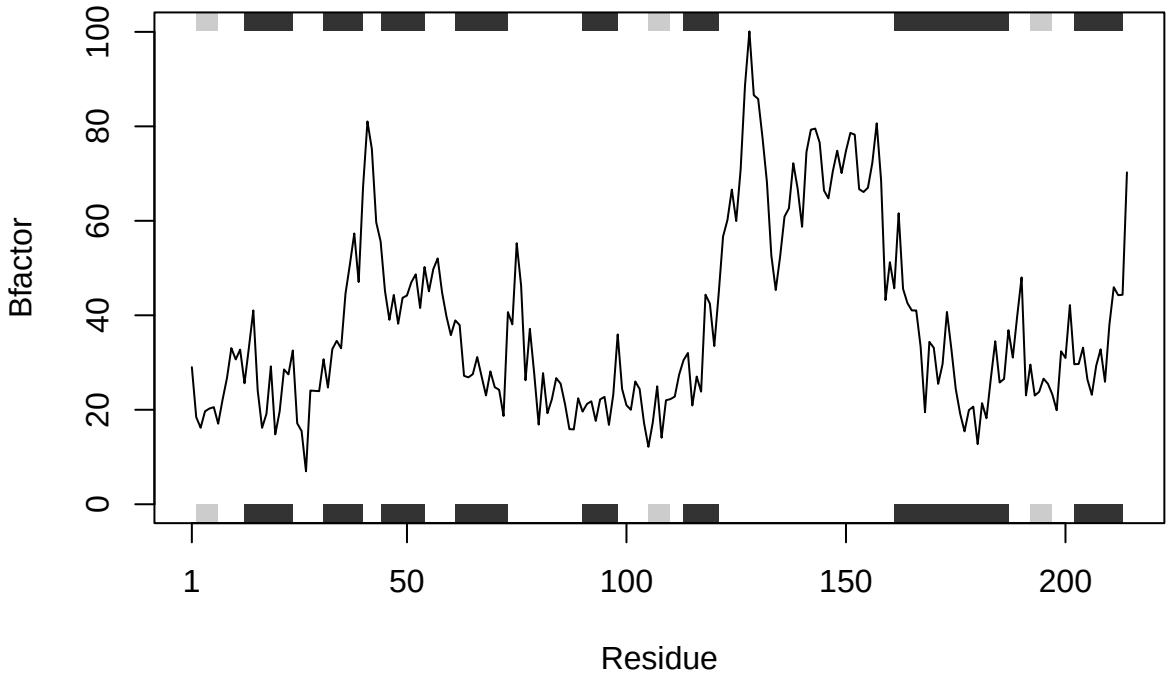
```
s2 <- read.pdb("1AKE") # kinase no drug
```

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

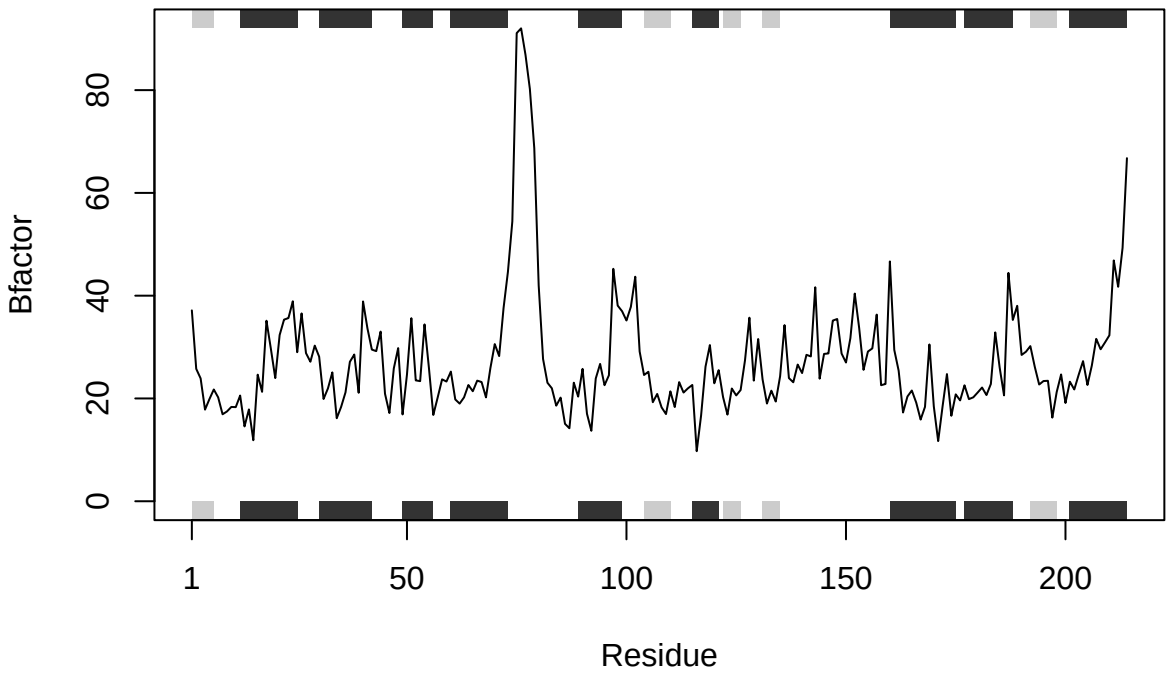
```
s3 <- read.pdb("1E4Y") # kinase with drug
```

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

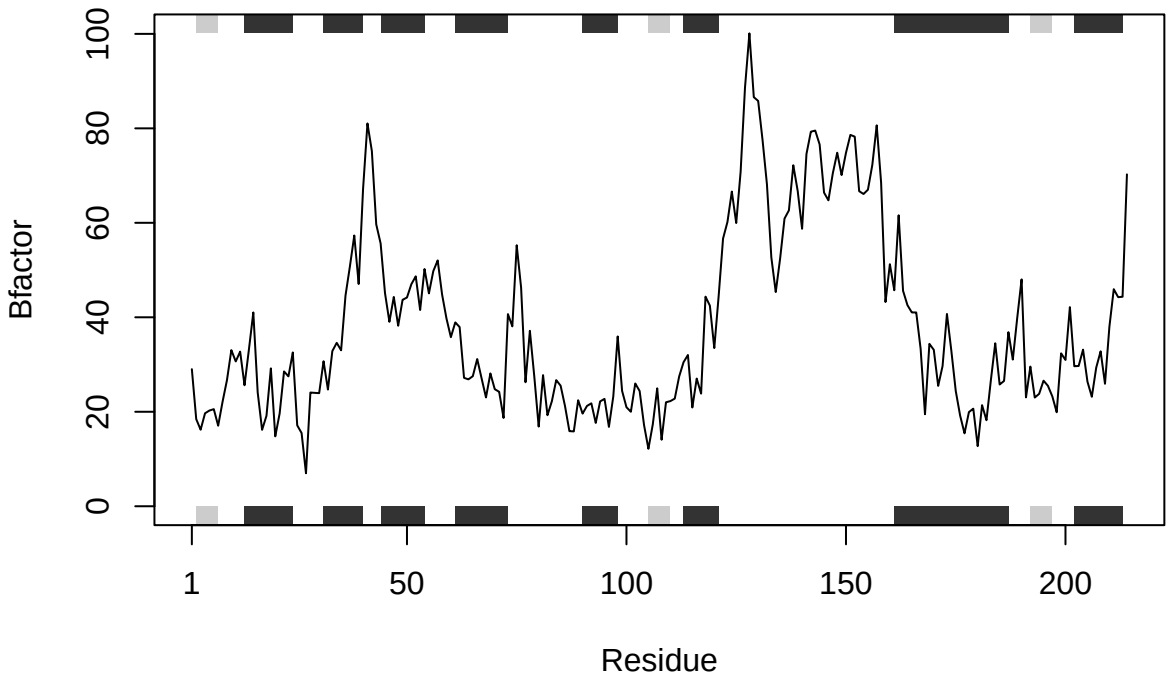
```
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s1, chain="A", elety="CA")

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")
```



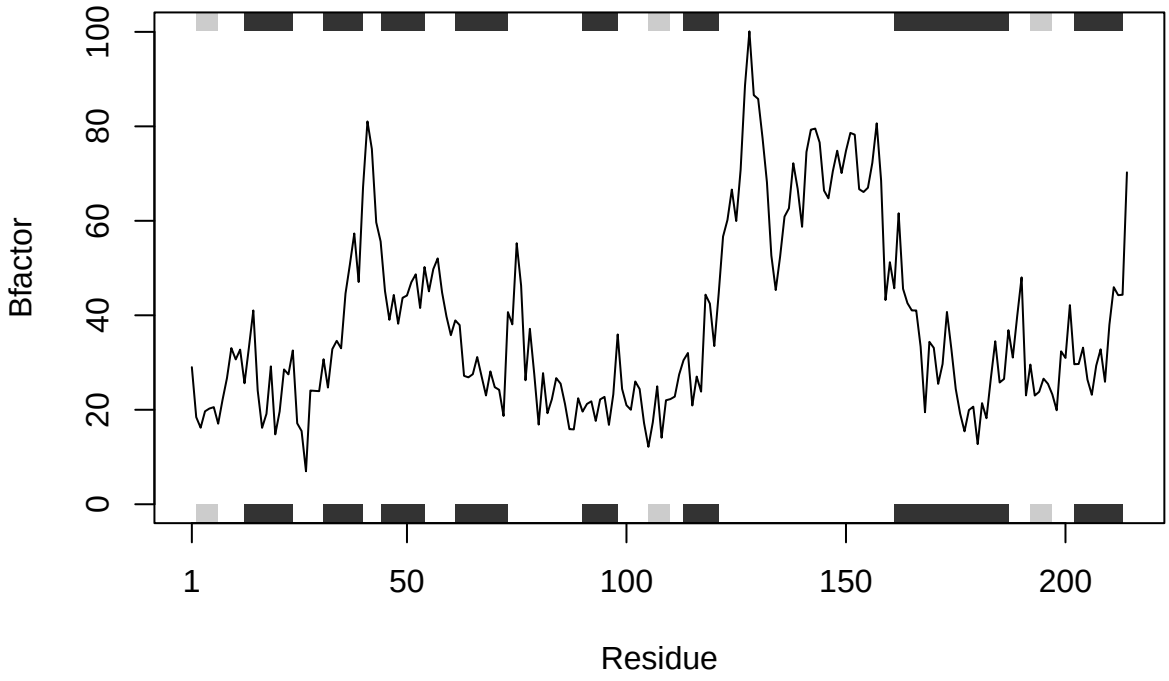
Q6. How would you generalize the original code above to work with any set of input protein structures?

Function: pdbPlot()

```
pdbPlot <- function(proteinFile, proteinChain, electrostaticMarker) {
  readFile <- read.pdb(proteinFile)
  storeChain <- trim.pdb(readFile, chain = proteinChain, elety = electrostaticMarker)
  storeBfactor <- storeChain$atom$b
  plotb3(storeBfactor, sse = storeChain, typ = "l", ylab = "Bfactor")
}
# EXAMPLE
pdbPlot("4AKE", "A", "CA")
```

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

```
## Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
## /var/folders/mg/vtnv_yzx5xj67yb2_2drgf2m0000gn/T//RtmpAQ5QYc/4AKE.pdb exists.
## Skipping download
```



Function with Comments

```
#' pdbPlot(): Protein Stability Analysis
#
#' @param proteinFile The PDB file that you would like to analyze
#' @param proteinChain The protein chain of interest
#' @param electrostaticMarker The electrostatic marker of interest
#'
#' @return The function pdbPlot() generates a plot that compares the B factors of desired protein struc
#'
#' @export
#'
#' @example
#' If a data analyst would like to analyze protein "4AKE", more specifically, its "A" side chain and it
#' pdbPlot("4AKE", "A", "CA")
#'
#'
pdbPlot <- function(proteinFile, proteinChain, electrostaticMarker) {
  # Loading and reading protein structure of interest
  readFile <- read.pdb(proteinFile)
  # Selecting and extracting the protein chain of interest
  storeChain <- trim.pdb(readFile, chain = proteinChain, elety = electrostaticMarker)
  # Selecting and extracting the B factors of interest
  storeBfactor <- storeChain$atom$b
```

```
  # Plotting the B factors against residues, with additional secondary structure elements (sse) which r
  plotb3(storeBfactor, sse = storeChain, typ = "l", ylab = "Bfactor")
}
```