Class 06 HW

Suraj Sidhu (A18512793)

Can you improve this analysis code?

```
library(bio3d)

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

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

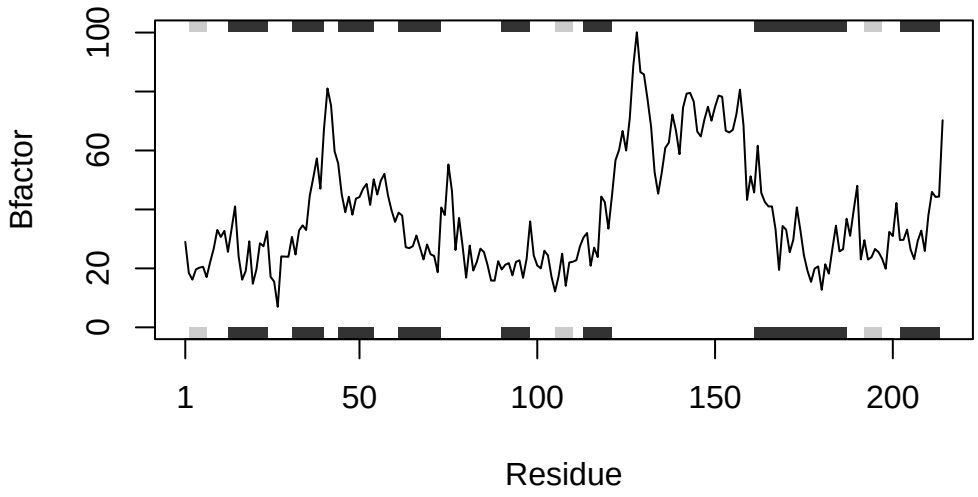
```
s2 <- read.pdb("1AKE")  # kinase no drug
```

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

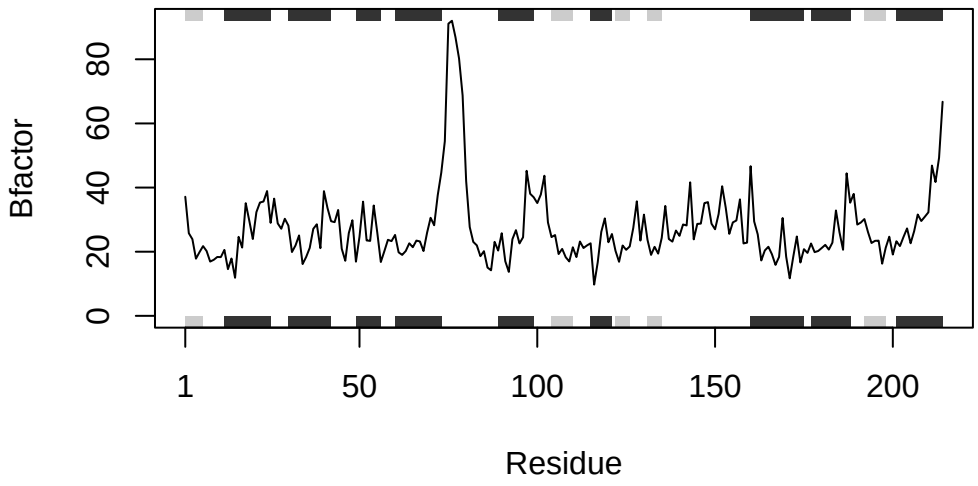
```
s3 <- read.pdb("1E4Y")  # kinase with drug
```

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

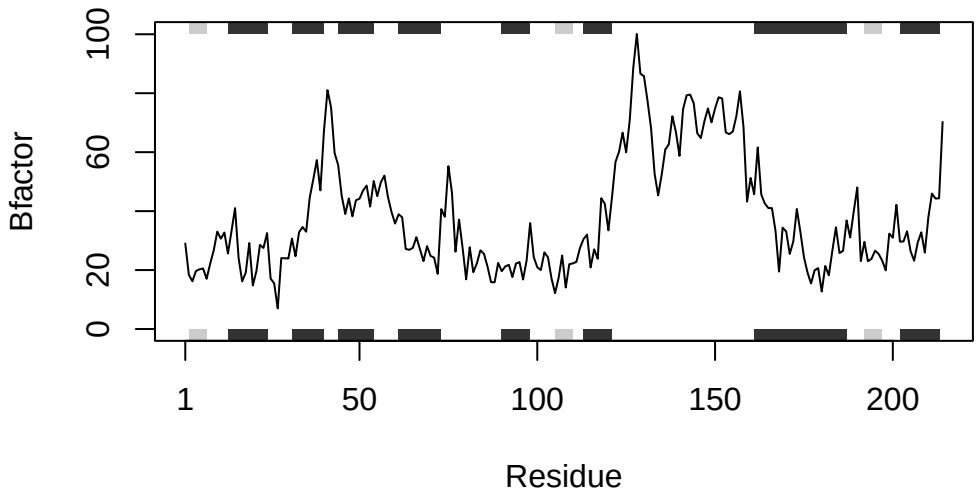
```
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s1, chain="A", elety="CA")

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")
```



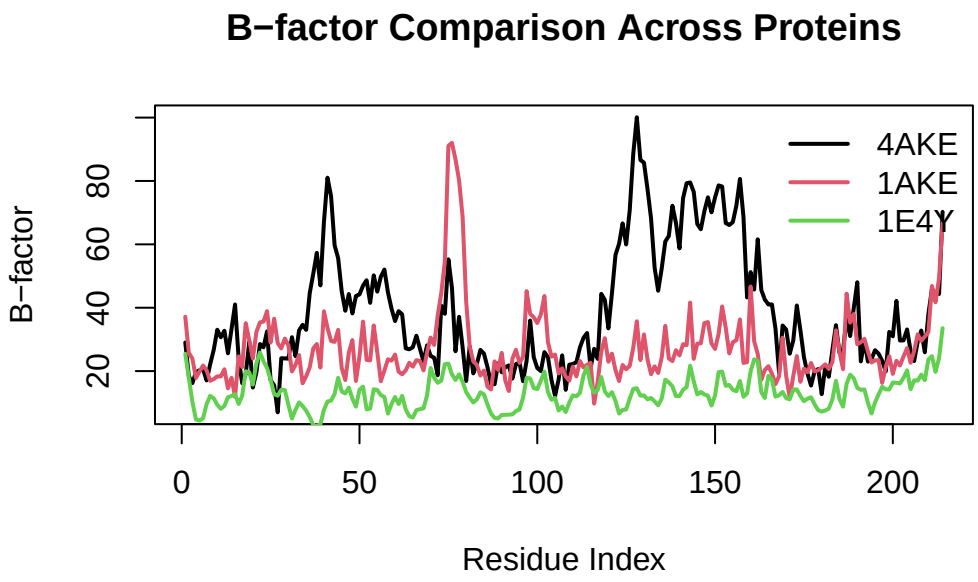
Improved version

```
# -------------------------------------------------------------
# Function: analyze_protein_bfactors()
# -------------------------------------------------------------
# INPUTS:
#   - pdb_ids: A character vector of PDB structure IDs (e.g., c("4AKE", "1AKE", "1E4Y"))
#
# WHAT IT DOES:
#   - Downloads and reads the specified protein structures.
#   - Extracts chain A and alpha carbon (CA) atoms.
#   - Calculates B-factors (temperature factors).
#   - Plots all B-factors together for easy comparison.
#
# HOW TO USE:
#   Call the function with a vector of PDB IDs, e.g.:
#       analyze_protein_bfactors(c("4AKE", "1AKE", "1E4Y"))
#
# OUTPUT:
#   - A line plot showing B-factor trends for all proteins.
```

```
#   - Returns a named list of B-factors for further analysis.
# ------------------------------------------------------------

# Load required package
library(bio3d)

# Define the function
analyze_protein_bfactors <- function(pdb_ids) {

  # 1  Initialize a list to store B-factors
  b_factors <- list()

  # 2  Loop through each PDB structure
  for (pdb in pdb_ids) {

    # 2a: Read the PDB file
    pdb_data <- read.pdb(pdb)

    # 2b: Extract chain A and C-alpha atoms
    pdb_chainA <- trim.pdb(pdb_data, chain = "A", elety = "CA")

    # 2c: Extract B-factors
    b_factors[[pdb]] <- pdb_chainA$atom$b
  }

  # 3  Plot the first protein's B-factors
  plot(
    b_factors[[1]],
    type = "l",
    lwd = 2,
    col = 1,
    ylab = "B-factor",
    xlab = "Residue Index",
    main = "B-factor Comparison Across Proteins"
  )

  # 4  Overlay the remaining proteins
  for (i in 2:length(b_factors)) {
    lines(
      b_factors[[i]],
      col = i,
      lwd = 2
```

```
    )
  }

  # 5  Add a legend for clarity
  legend(
    "topright",
    legend = pdb_ids,
    col = seq_along(b_factors),
    lwd = 2,
    bty = "n"
  )

  # 6  Return all B-factors as a named list
  return(b_factors)
}

# ------------------------------------------------------------
# Example call (demonstration of working function)
# ------------------------------------------------------------

pdb_ids <- c("4AKE", "1AKE", "1E4Y")

bf_results <- analyze_protein_bfactors(pdb_ids)
```

```
  Note: Accessing on-line PDB file

Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
C:\Users\sidhu\AppData\Local\Temp\Rtmp2jmNRY/4AKE.pdb exists. Skipping download

  Note: Accessing on-line PDB file

Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
C:\Users\sidhu\AppData\Local\Temp\Rtmp2jmNRY/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):
C:\Users\sidhu\AppData\Local\Temp\Rtmp2jmNRY/1E4Y.pdb exists. Skipping download
```


B−factor Comparison Across Proteins
B−factor
20
40
60
80
0
50
100
150
200
Residue Index
4AKE
1AKE
1E4Y