Homework for Class 6 - Writing R Functions

Jessica Le (PID: A17321021)

2025-01-24

Can you improve this code analysis?

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

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

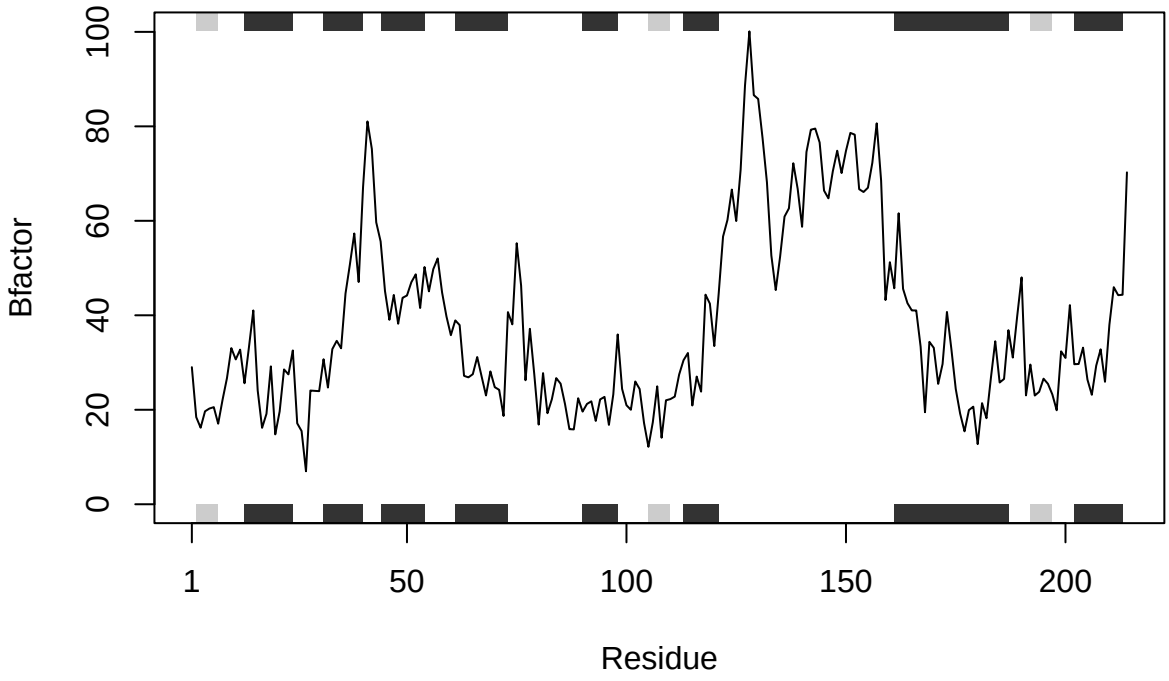
```
s2 <- read.pdb("1AKE") # kinase no drug
```

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

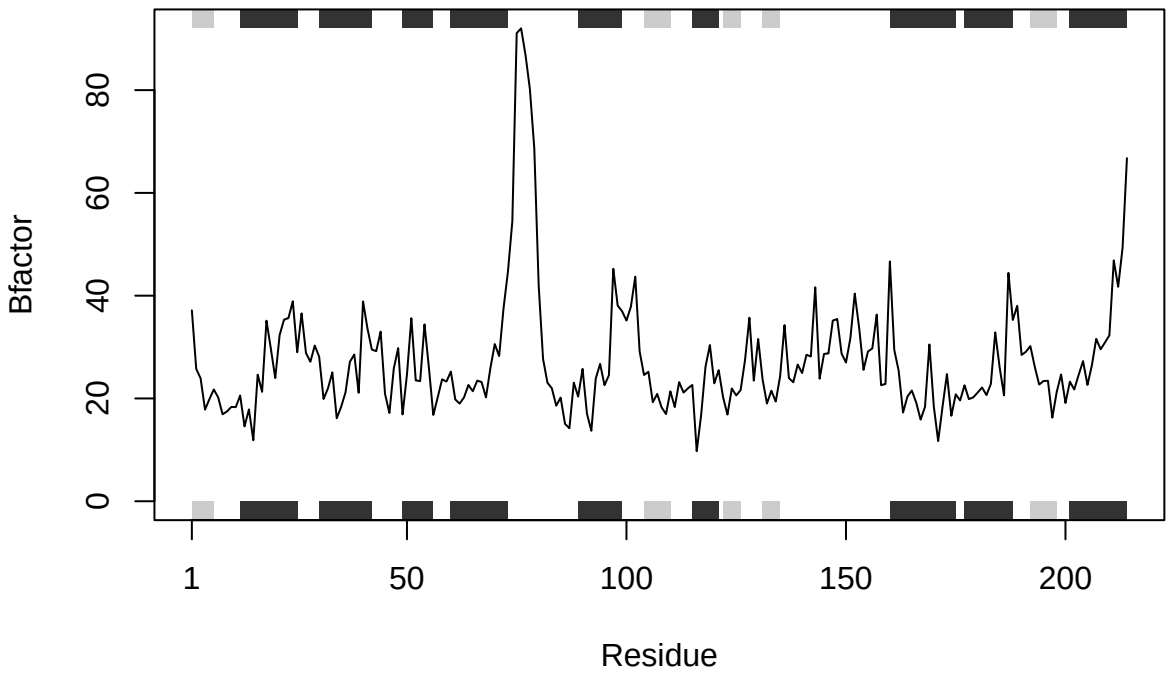
```
s3 <- read.pdb("1E4Y") # kinase with drug
```

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

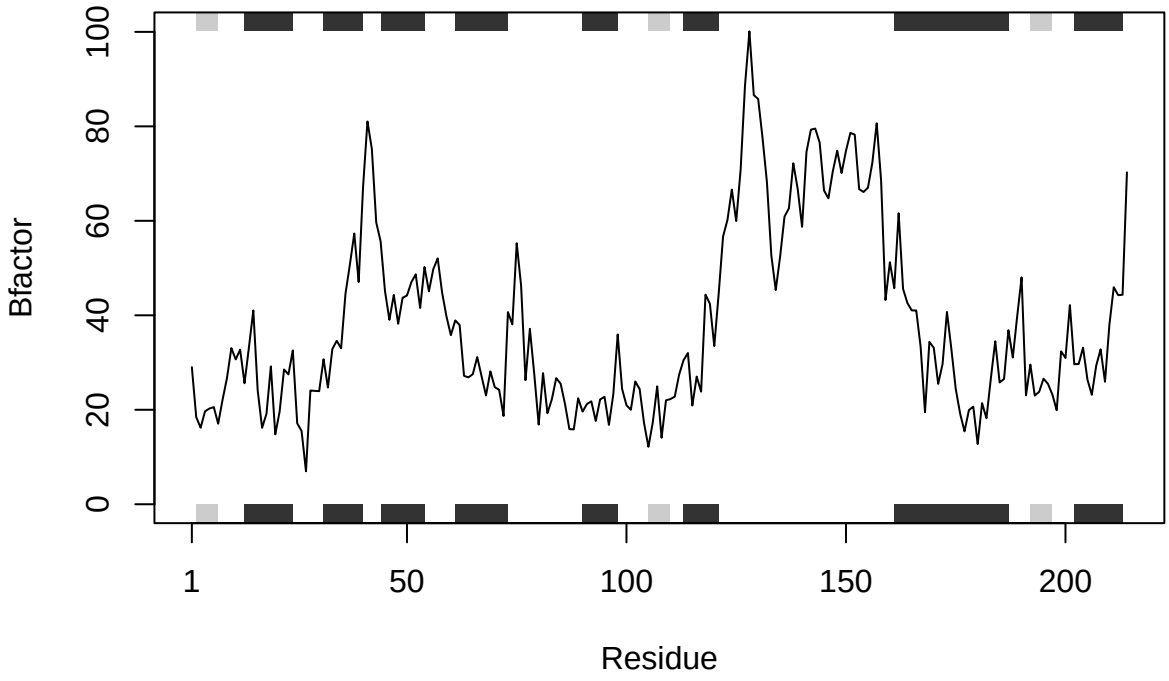
```
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s1, chain="A", elety="CA")
s1.b <- s1.chainA$atom$b
s2.b <- s2.chainA$atom$b
s3.b <- s3.chainA$atom$b
plotb3(s1.b, sse=s1.chainA, typ="l", ylab="Bfactor")
```




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




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



The generated function takes the input of a protein code from the Protein Data Bank to generate a plot of the beta factor of the atoms in chain A of the specified protein. It will work with any set of input protein structures if `plot_pdb` is called.

```
library(bio3d)
plot_pdb <- function(pdb_code) {
  s <- read.pdb(pdb_code)
  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")
}
```

This is a breakdown of the generated function.

```
#Call installed bio3d database
library(bio3d)

#Takes the input of a pdb code and creates pdb file that is read using `read.pdb`
plot_pdb <- function(pdb_code) {
  s <- read.pdb(pdb_code)

#Reduces the PDB structures to a smaller subset of atoms using function `trim.pdb` and the input is "s"
  s.chainA <- trim.pdb(s, chain="A", elety="CA")

#Selecting beta factor from atoms and the input is "s.chainA"
  s.b <- s.chainA$atom$b
```

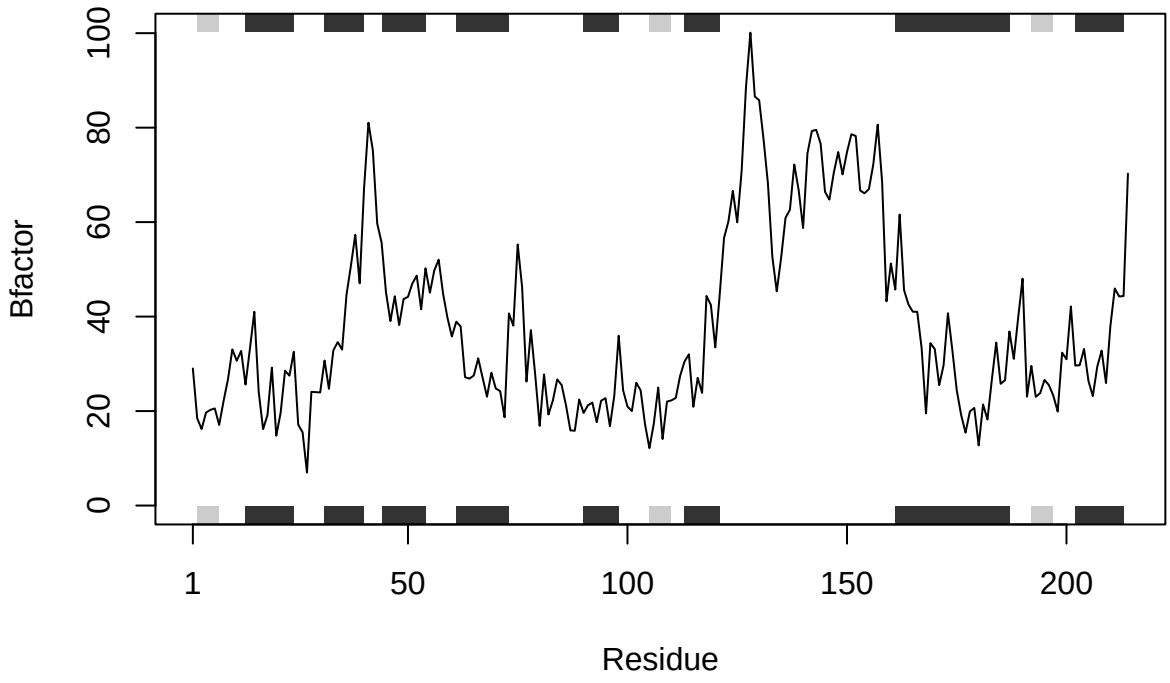
```
#Plotting the beta factor vs. residues for the specified atom where the input is "s.b"
  plotb3(s.b, sse=s.chainA, typ="l", ylab="Bfactor")
}
```

Running the generated function by calling `plot_pdb()`, the output plots for the three specified proteins are.

```
sapply(c("4AKE", "1AKE", "1E4Y"), FUN=plot_pdb)
```

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

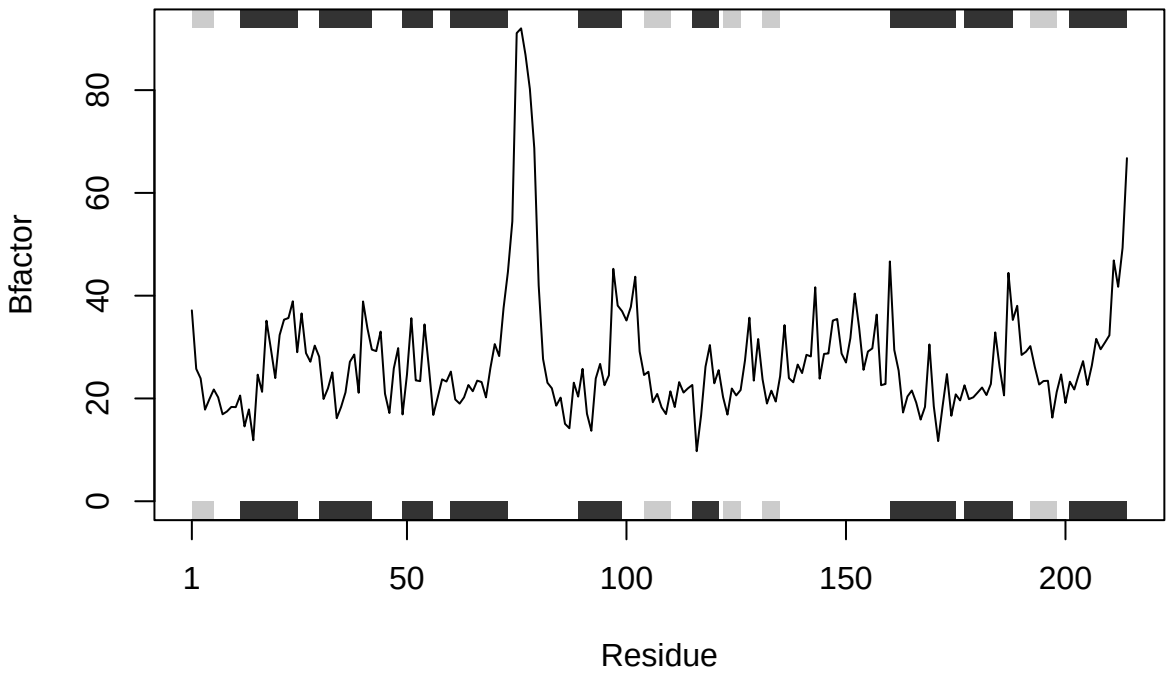
```
## Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
## /var/folders/s8/d27zs9y55db6qp39x9h69v9r0000gn/T//RtmpDBjn2w/4AKE.pdb exists.
## Skipping download
```



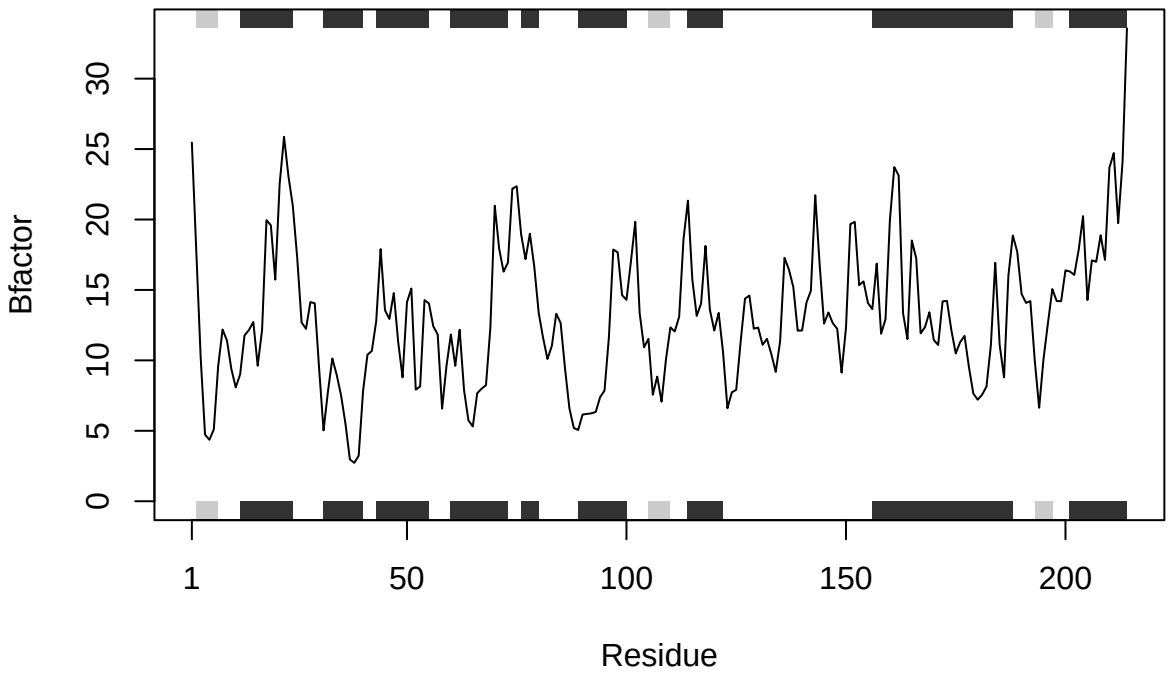

```
##   Note: Accessing on-line PDB file
```

```
## Warning in get.pdb(file, path = tempdir(), verbose = FALSE):
## /var/folders/s8/d27zs9y55db6qp39x9h69v9r0000gn/T//RtmpDBjn2w/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/s8/d27zs9y55db6qp39x9h69v9r0000gn/T//RtmpDBjn2w/1E4Y.pdb exists.
## Skipping download
```




```
## $`4AKE`
## NULL
##
## $`1AKE`
## NULL
##
## $`1E4Y`
## NULL
```

‘‘