Homework

Emily Hendrickson (PID: A69034780)

To improve analysis code:

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

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

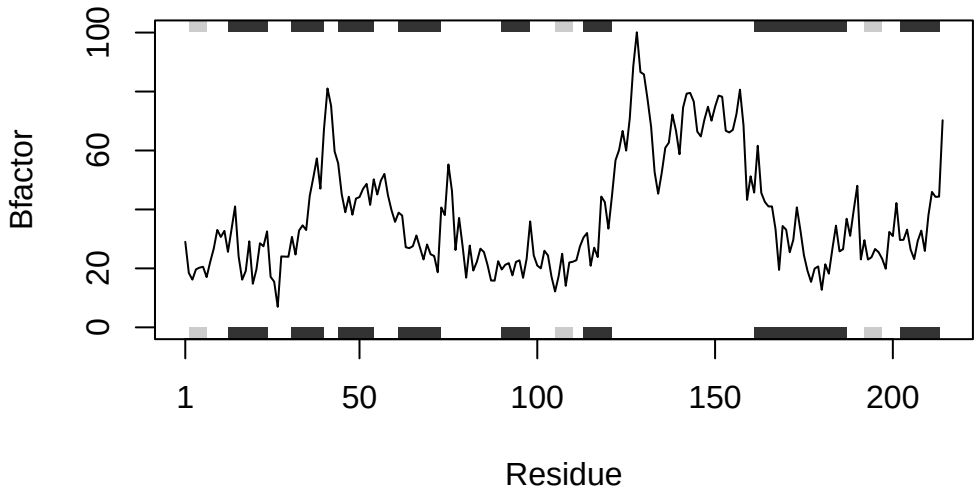
```
s2 <- read.pdb("1AKE")  # kinase no drug
```

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

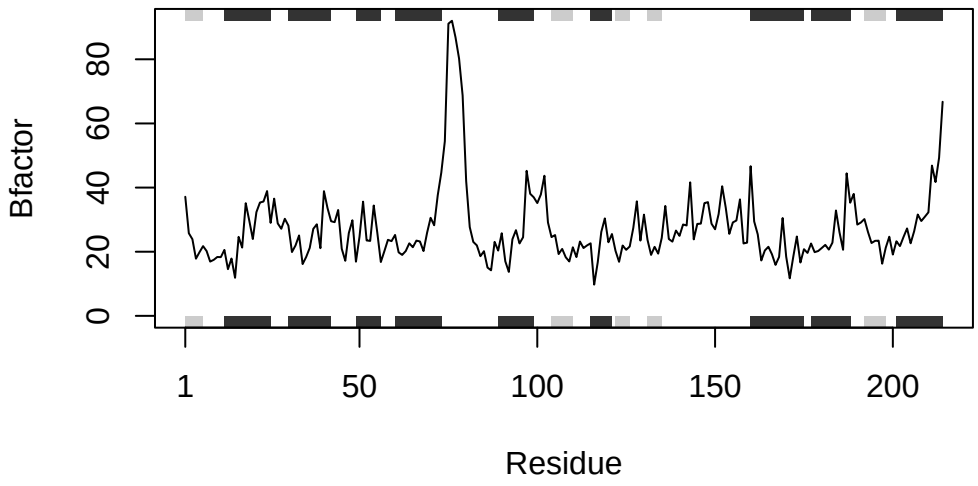
```
s3 <- read.pdb("1E4Y")  # kinase with drug
```

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

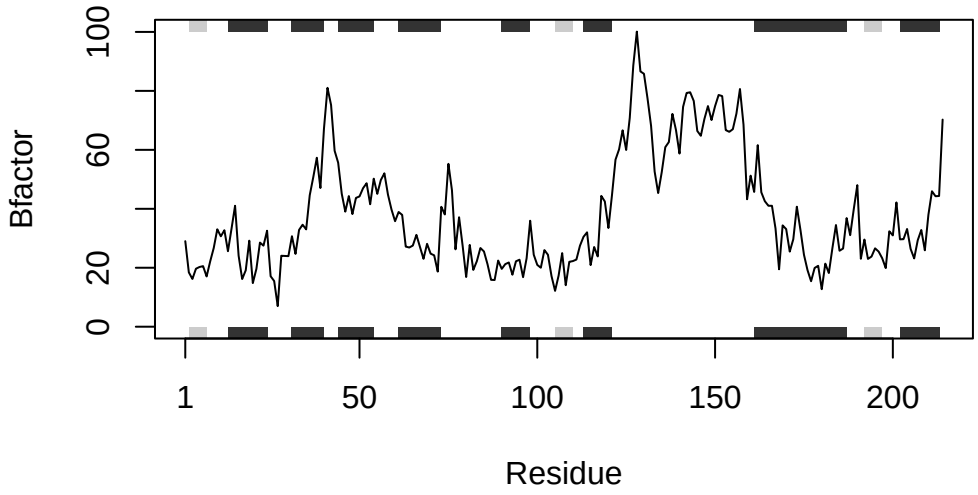
```
s1.chainA <- trim.pdb(s1, chain="A", elety="CA")
s2.chainA <- trim.pdb(s2, chain="A", elety="CA")
s3.chainA <- trim.pdb(s1, chain="A", elety="CA")

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")
```

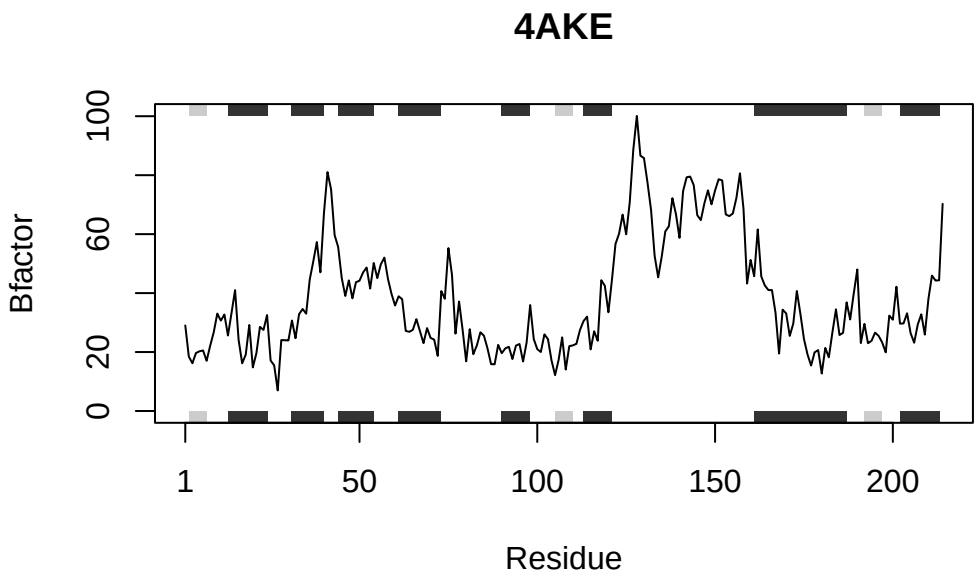


Improved code with input of a csv file and output of a figure with title of PDB ID to differentiate between proteins:

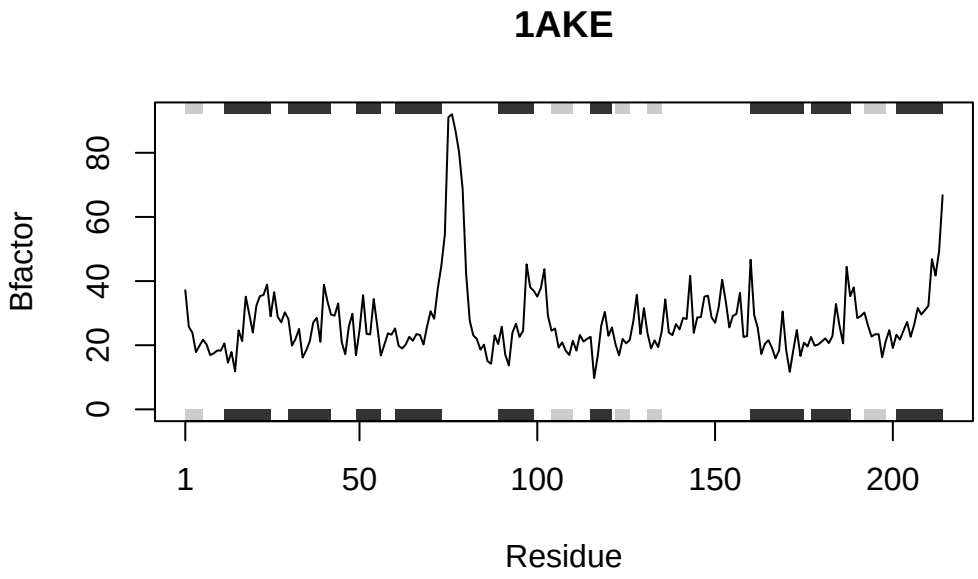
```
# Change file name to your list of proteins where column title = PDB_ID

protein_list <- "proteins.csv"


# Make function to generate plot of y = B factor and x = Chain A residue with PDB
# ID as plot title

plot_bfactor <- function(protein_code) {
  seq <- read.pdb(protein_code)
  chainA <- trim.pdb(seq, chain="A", elety="CA")
  seq.b <- chainA$atom$b
  plotb3(seq.b, sse=chainA, typ="l", ylab="Bfactor", main = protein_code)
}


# Generate Figures
```

```
proteins <- read.csv(protein_list, header = TRUE, stringsAsFactors = FALSE)
pf <- as.vector(proteins$PDB_ID)
sapply(pf, plot_bfactor)
```

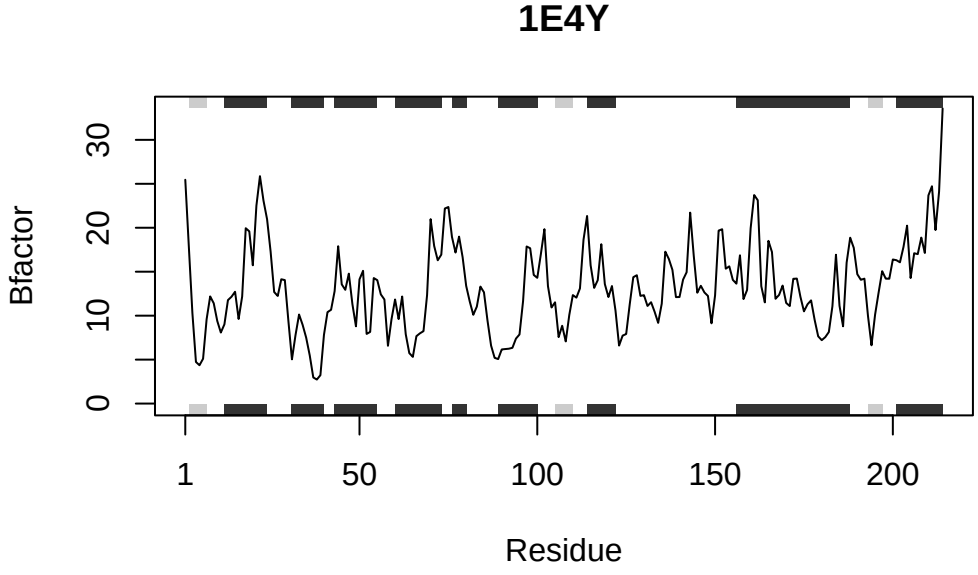
```
Note: Accessing on-line PDB file
```




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




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




```
$`4AKE`
```

```
NULL

$`1AKE`
NULL

$`1E4Y`
NULL
```