hw6

Courtney Cameron PID:A69028599

Q6

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

The original code does every step individually for each pdb code but by making a function graphs can be made for each desired pdb code with one line of code rather than the entire process. Copying long chuncks of code is time consuming and can be error prone.

Calling bio3d package

```
library(bio3d)
```

```
Warning: package 'bio3d' was built under R version 4.2.3
```

orginal code:

```
# Can you improve this analysis code? library(bio3d)
#s1 <- read.pdb("4AKE")  # kinase with drug
#s2 <- read.pdb("1AKE")  # kinase no drug
#s3 <- read.pdb("1E4Y")  # kinase with drug

#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")
```

created function to create graph based on imputted pdb code:

- reads the pdb coordinate file
- trims to a smaller file only including chain A
- Selects for atom b
- plots Bfactor across of the selected residue

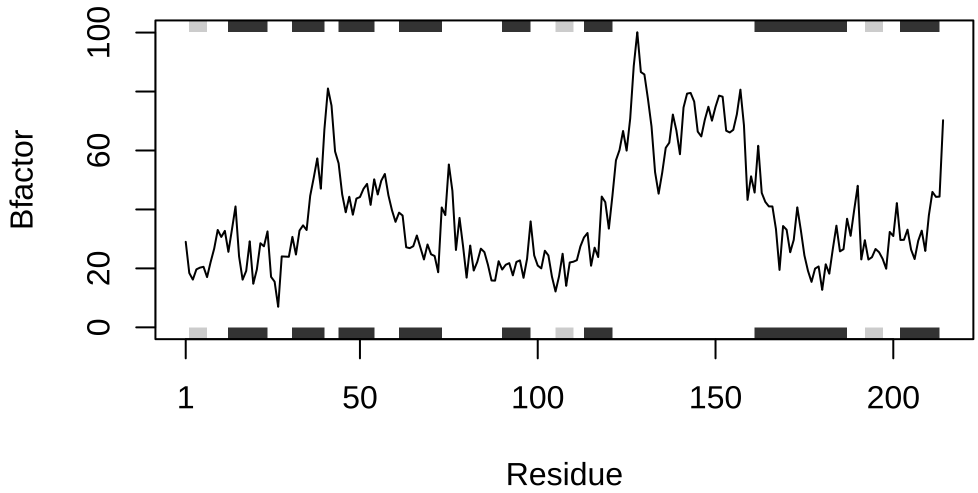
The original code is condensed and renamed with clearer names into the function "pdb_graph" to generate a function that does the same actions as the longer, repetitive original code.

```
pdb_graph <- function(x) {
  pdb <- read.pdb(x)
  chain <- trim.pdb(pdb, chain="A", elety="CA")
  plotb3(chain$atom$b, sse=chain, typ="l", ylab="Bfactor")
}
```

testing function with multiple pdb codes to ensure the function is working properly

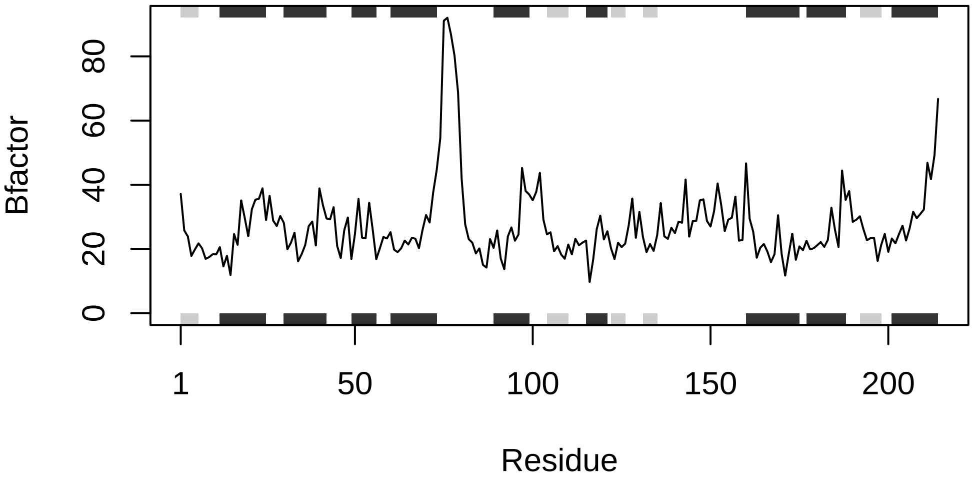
```
pdb_graph('4AKE')
```

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



```r
pdb_graph('1AKE')
```

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


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