

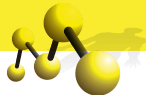
# Homework/Review

Luis Pedro Coelho

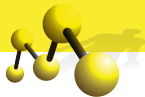
Programming for Scientists

October 22, 2012





- Download file
- Open/parse it
- Analyse it

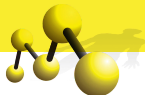


According to Wikipedia, we get a sequence every four lines:

- 1 Header
- 2 DNA sequence
- 3 + line
- 4 Qualities

```
ifile = open( 'hw-HeLa.fq' )
while True:
    header = ifile.readline().strip()
    seq = ifile.readline().strip()
    _ = ifile.readline().strip()
    qualities = ifile.readline().strip()
    if not len(qualities):
        break
```

(This is not the only way to structure this)



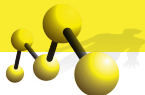
- We have read it as a string
- We have a sequence of characters
- Now, we want to get the numeric value

We will look it up.



```
import numpy as np
...
while True:
    ...
    qualities = [(ord(q) - 64) for q in qualities]
    qualities = np.array(qualities)
```

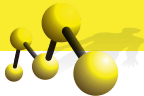
# Let us compute the averages



```
import numpy as np
from matplotlib import pyplot as plt

allqs = []
ifile = open( 'hw-HeLa.fq ' )
while True:
    header = ifile.readline().strip()
    seq = ifile.readline().strip()
    _ = ifile.readline().strip()
    qualities = ifile.readline().strip()
    if not len(qualities):
        break
    qualities = [(ord(q) - 64) for q in qualities]
    qualities = np.array(qualities)
    allqs.append(qualities)
allqs = np.array(allqs)
```

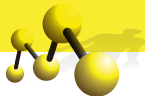
# Continuing...



Now we have an array named `allqs` with all the qualities

```
mean = allqs.mean(0)
```

```
std = allqs.std(0)
```



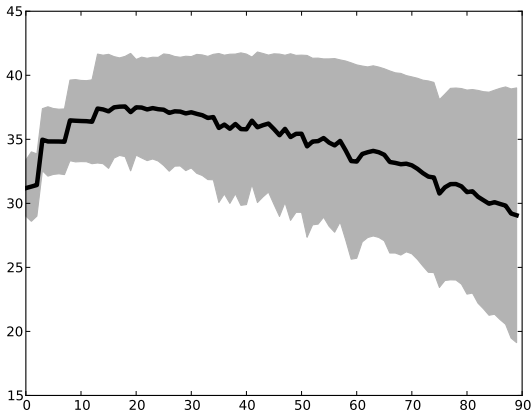
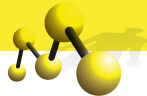
Now we have an array named `allqs` with all the qualities

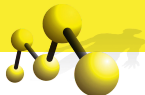
```
mean = allqs.mean(0)
std = allqs.std(0)

gray = (.7, .7, .7)
plt.fill_between(
    np.arange(allqs.shape[1]),
    mean-std,
    mean+std,
    color=gray)
plt.plot(mean, color='k', lw=4)
plt.show()
```

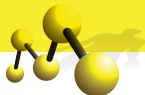


# Output





- So far, as I said, I used the file after unzipping
- Can we use the file **as is**?
- For that, we need to open a gzipped file.
- Let us look it up...



Replace

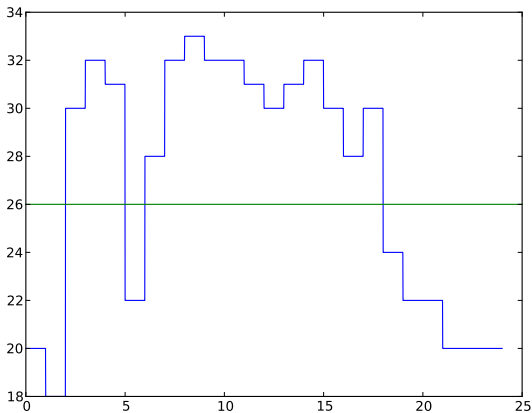
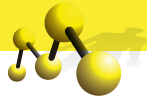
```
ifile = open( 'hw-HeLa.fq ' )
```

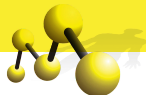
by

```
import gzip
```

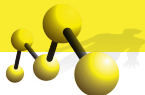
```
ifile = gzip.open( 'hw-HeLa.fq.gz ' )
```

# Trimming



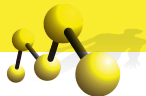


- ① We look from left to right.
- ② Remember the longest substring we have seen.
- ③ When we are done, the longest substring will be it.

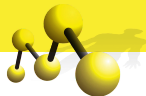


```
def trim(qs, thresh):
    bestlen = 0
    startcur = 0
    for i in xrange(len(qs)+1):
        if i >= len(qs) or qs[i] < thresh:
            curlen = i - startcur
            if curlen > bestlen:
                bestlen = curlen
                best = (startcur, i)
            startcur = (i+1)
    s, e = best
    return s, e
```

# Putting it Together



Let us look at the source code...



- Do it in one pass