Turning biology into mathematics

Niranga Udumulla

September 13, 2020

Abstract

All life on the Earth is composed of cells, which are themselves composed of molecules. A cross section through a single bacterial cell is shown here. It is surrounded by a multi-layered cell wall, colored green. The long corkscrew-shaped flagella are turned by motors in the cell wall, propelling the cell through its environment. The interior of the cell is filled with molecular machines for building and repairing molecules, for harnessing different sources of energy, and for sensing and protecting against environmental dangers

1 Introduction

In this report we will explore this common birthright of molecular machines. We will start with a look at the machines themselves and the unusual molecular world in which they operate. Then, we will explore how they are combined in living cells. Finally, we will look at a few special topics related to our own molecules and cells.

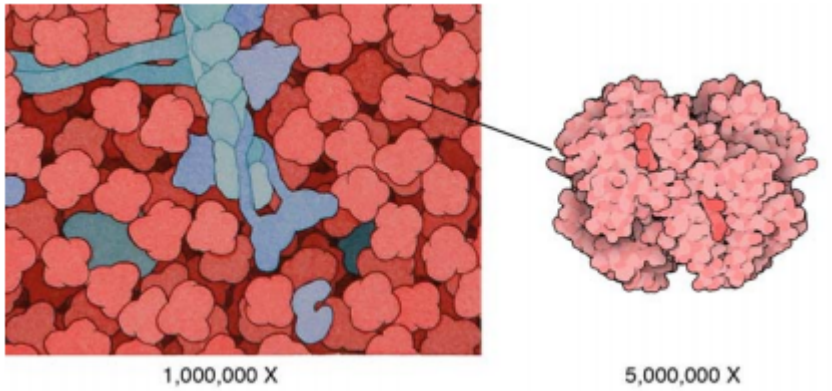
2 Theory

In this lab demonstration we will access the uniprot database and create a new dataset. Our dataset will consist of four thousand proteins, half associated with the keyword antibody and the other half not related to the keyword. The proteins are represented by their primary structure sequence of amino acids, in other words each protein is a string of letters representing each amino acid in the sequence.

2.1 A Matter of Scale

Cells are small but not unimaginably small, and molecules are really,really small. Cells are about 1000 times smaller in length than objects in our everyday world. The largest cells, such as protozoa, can be seen with a magnifying glass, but a microscope is needed to see most of the cells in your body. Typical human cells are about 10 mm in length. This is roughly 1000 times smaller than the last joint in your finger. A 1000-fold difference in size is not difficult to visualize: a grain of rice is about 1000 times smaller in length than the room

you are sitting in. Imagine your room filled with grains of rice. That will give you an idea of the billion or so cells that make up your fingertip.



3 Procedure

3.1 Create a new data set using uniprot database

There is a database name uniprot and it has primary structure sequence of amino acids.Following names and short names gives a indication of each variable.

A Ala Alanine

R Arg Arginine

N Asn Asparagine

D Asp Aspartic acid

C Cys Cysteine

Q Gln Glutamine

E Glu Glutamic acid

G Gly Glycine

H His Histidine

I Ile Isoleucine

L Leu Leucine

K Lys Lysine

M Met Methionine

F Phe Phenylalanine

P Pro Proline

S Ser Serine

T Thr Threonine

W Trp Tryptophan

Y Tyr Tyrosine

V Val Valine

O Pyl Pyrrolysine

U Sec Selenocysteine

B Asx Aspartic acid or Asparagine

Z Glx Glutamic acid or Glutamine

X Xaa Any amino acid

```
!pip install git+https://github.com/williamedwardhahn/mpcr
from mpcr import *
```

4 Analysis

This Part will show the different types of statistical analysis.There are 2000 protein samples have been loaded for our analysis.

4.1 Loading data set to google colab

```
# This code will create a dataset from the uniprot database
X, Y = get_uniprot_data('=antibody', '!antibody', 2000)
# create dataset with 2000 samples
```

```
number_X = len(X)
number_Y = len(Y)

print(number_X)
print(number_Y)

X[0]
```

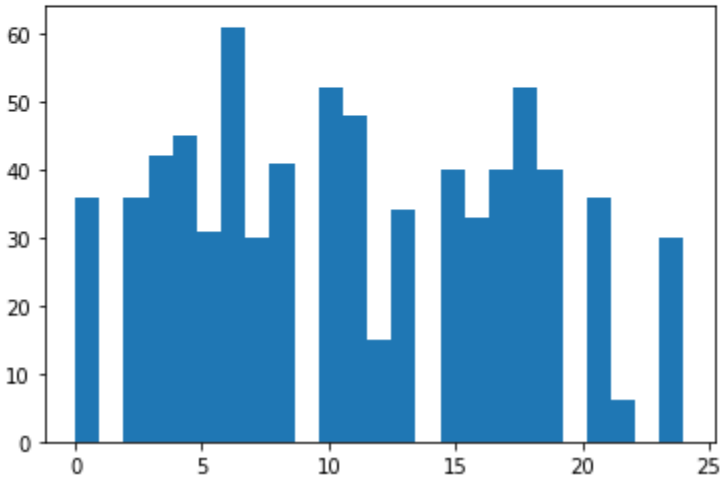
This output shoes the Amino acid sequence of the first protein on the list of proteins associated with 'antibody'. The given output gives us a lengthy string set.Following function was helped to resolve that problem and got the exact values for amino acid sequence.

```
def process_strings(c):
    '''Takes in a list of sequences 'c' and turns each one
       into a list of numbers.'''

    X = []

    for  m, seq in enumerate(c):
        x = []
        for letter in seq:
            x.append(max(ord(letter)-97, 0))

        X.append(x)

    return X


X = process_strings(X)
Y = process_strings(Y)


print(X[0])
# shows the exact values for first protein
```

4.2 Histogram for first 25 observations of the first protein

```
[10]  1  plt.hist(X[0],25)
```

```
(array([36.,  0., 36., 42., 45., 31., 61., 30., 41.,  0., 52., 48., 15.,
        34.,  0., 40., 33., 40., 52., 40.,  0., 36.,  6.,  0., 30.]),
 array([ 0.  ,  0.96,  1.92,  2.88,  3.84,  4.8 ,  5.76,  6.72,  7.68,
         8.64,  9.6 , 10.56, 11.52, 12.48, 13.44, 14.4 , 15.36, 16.32,
        17.28, 18.24, 19.2 , 20.16, 21.12, 22.08, 23.04, 24.  ]),
 <a list of 25 Patch objects>)
```



This histogram shows the distribution of first 25 observations of the first sequence of first protein.Moreover,we can find the length of each and every protein using following syntax.

```
X_lengths = [len(s) for s in X]
Y_lengths = [len(s) for s in Y]
```

The maximum and minimum length would be as follows.

```
1  np.max(X_lengths)
```

```
5654
```

```
[ ]  1  np.max(Y_lengths)
```

```
11103
```

```
[ ]  1  np.min(X_lengths)
```

```
5
```

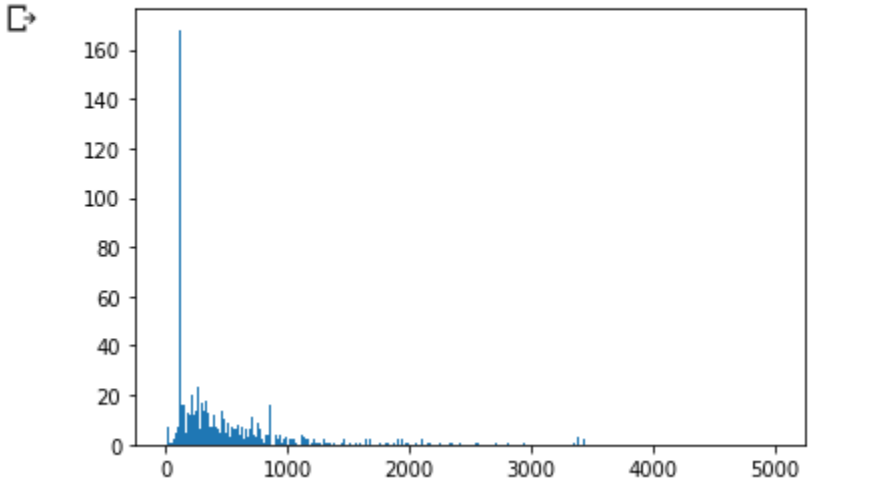
```
[ ]  1  np.min(Y_lengths)
```

```
6
```

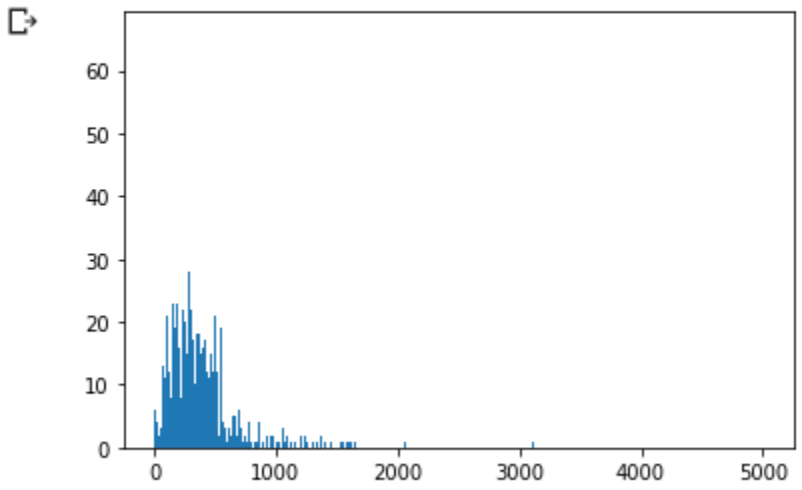
4.3 Histogram for first observations of the all protein for 2 sequences

```
plt.hist(X_lengths,bins=1000,range=(0,5000));
```

```
[ ]  1  plt.hist(X_lengths,bins=1000,range=(0,5000));
```



```
[ ]  1  plt.hist(Y_lengths,bins=1000,range=(0,5000));
```



5 Conclusions

Unprot data set has 2 data sets and each data set contains thousands of proteins and each protein category has their own values.

According to the first sequence, loaded has 2000 proteins and each protein contains different types of values and most of proteins have size below 1000. There are very rare case occur in

X and Y data sets which is the values in a protein is greater than 3000.

References

[1] *Uniprot database*, available at `https://www.uniprot.org/`.