ACCESS BIOINFORMATICS DATABASES WITH BIO-PYTHON

This project is aimed to deploy python-based programming pipelines and scripts to automate biological data retrieval and analysis.

4. EXPASY

In this section, using the ExPASy module, I fetched domain details of the proteins that I retrieved and processed from the section above. ExPASy is an online bioinformatics resource operated by SIB (Swiss Institute of Bioinformatics). Prosite is a protein database, which consists of entries describing the protein families, domains and functional sites, as well as amino acid patterns and profiles in them.

4.1. PROSITE

Import Modules

```
In [53]: from Bio import ExPASy
         from Bio.ExPASy import Prosite
```

In this step, I passed in the prosite ID to the ExPASy module to show the contents:

```
In [54]: handle = ExPASy.get_prosite_raw('PS51442')
         record = Prosite.read(handle)
print(record.description)
```

Coronavirus main protease (M-pro) domain profile.

Various PDB structures possessing this domain profile can be found:

```
In [55]: print(record.pdb_structs[:10])
```

```
['1LVO', '1P9S', '1P9U', '1Q2W', '1UJ1', '1UK2', '1UK3', '1UK4', '1WOF', '1Z1I']
```

These are the proteins containing the domain profile of coronavirus main protease (M-pro).

To find patterns in the domain, I created a similar variable handle, to which I passed in the ExPASy module followed by the prosite function, to which I passed in another ID PS00001. The prosite handle is then read.

```
In [56]: handle = ExPASy.get_prosite_raw('PS00001')
         record = Prosite.read(handle)
print(record.pattern)
```

```
N-{P}-[ST]-{P}.
```

The result shows that the common pattern within the domain is asparagine, proline, serine, and threonine.