Susan Rui Ling
2425170
BIOEN537 Software Project
11/08/24

Component Specifications

Software components

Data Manager:

The Data Manager is responsible for loading and managing protein structures from PDB files. This component provides an interface for accessing key properties of the protein structure, such as residues, atoms, chains, and side chains, in a format that the other components can easily use. It enables the identification of specific atoms for user highlighting and colouring by residue type.

Functions:

1. Load Protein Structure: loads and parses a PDB file to extract protein data, organizing info on chains, residues and atoms.
    - Input: path to PDB file
    - Output: structured protein data object using BioPython package containing chains, residues, and atoms.
2. Generate Residue Colour Map: creates a mapping of residue types to specific colours for visualizations.
    - Input: list of residue types
    - Output: a dictionary mapping each residue type to a colour

2D Visualization Manager:

The 2D Visualization Manager provides 2D representations of protein structures. This component creates simplified diagrams of the protein's primary and secondary structures. It visualizes the amino acid sequence as a linear string and uses different shapes to represent helices, sheets, and coils for secondary structures.

Functions:

1. Plot Primary Structure: displays the amino acid sequence of the protein in a linear format
    - Input: protein structure data and colour map from the Data Manager
    - Output: a figure showing the linear sequence of amino acids using matplotlib
2. Plot Secondary Structure: displays a 2D schematic of the protein's secondary structure using shapes to represent helices, sheets, and coils.
    - Input: protein structure data, residue colour map, and highlighting configuration
    - Output: a figure displaying the 2D schematic of secondary structures using matplotlib

Susan Rui Ling
2425170
BIOEN537 Software Project
11/08/24

3D Visualization Manager:

The 3D Visualization Manager provides a 3D interactive view of the protein, showing atoms, bonds, and spatial relationships. This component enables colouring by residue type and highlighting specific atoms.

Functions:

1. Render 3D Model: renders the 3D structure of the protein, displaying atoms and bonds with appropriate spatial relationships.
    - Input: protein structure data, colour map, highlighting configuration
    - Output: a 3D plot displaying the protein structure with optional colouring and highlighting using nglview package.

Interactions to accomplish use cases

a) The user would load a PDB file which will then parse the protein data into a data object.

To accomplish this use case, the user will go to RCSB Protein Data Bank (RCSB PDB), search and click into their protein of interest. The user will then download the PDB Format file. Using the "Load Protein Structure" function, the user will load in their PDB file.

b) The user would choose to view the protein's primary structure which will then display the linear sequence of amino acids in a 2D diagram.

To accomplish this use case, the "Plot Secondary Structure" function can be called from the protein structure object from above. This will then display a schematic of the protein's secondary structures in 2D.

c) The user would choose to view the protein's 3D structure which will then render the full 3D structure, displaying all atoms and bonds in 3D space.

To accomplish this use case, the "Plot Render 3D Model" function can be called from the protein structure object from above. This will then display a schematic of the protein's 3D structure.

Susan Rui Ling
2425170
BIOEN537 Software Project
11/08/24

Preliminary plan

1. Project Initialization: set up a Git repository and define a clear project structure including: /data, /visualization, /utils, and /tests
2. For the Data Manager component, implement a function using BioPython package to load and parse PDB files and extract primary and secondary structure information. Build a class for accessing protein data including chains, residues, and atomic coordinates.
3. For the 2D Visualization Manager, implement a function that generates a linear representation of the primary structure using matplotlib. Add function for basic colouring based on residue type.
4. For the 2D Visualization Manager, incorporate a function to represent secondary structures with different shapes or colours. Add function for highlighting specific residues.
5. For the 3D Visualization Manager, implement a function that displays a 3D rendering of the structure using nglview. Add function for basic interactive capabilities and colouring by residues.
6. Write unit tests for each function and test the package with various PDB files. Debug errors.
7. Finalize documentation, setup.py and upload to PyPI for user installation.