Component Specification for simulate_xna_signal.py

Jade Minzlaff

1. **Software Components:**

Functions:

- import_kmer_library
    - Generates a pandas dataframe with 2 columns and 4352 rows containing all possible 4mers for the model with their associated experimental current signal.
    - Inputs: 8 .csv files
    - Outputs: KXmer_signal (Pandas dataframe, Columns: ‘Kmer’, ‘Mean level’, 4352 rows)
- input_to_4mers
    - Converts an inputted nucleotide sequence into a list of 4mers where each 4mer in the list corresponds to a base in the sequence which can be simulated.
    - Inputs: user_input (string) Input given as a string with no spaces or other symbols, comprised of the bases from the Artificially Expanded Genetic System (AEGIS): A,T,G,C, J, K, P, S, V, X, and Z.
    - Outputs: KXmers (list where each entry is a 4-character string)
- generate_xplot
    - Generates the sequence of bases from the input which can be simulated and uses its length to generate the x-axis for the step function plot
    - Inputs: KXmers (list where each entry is a 4-character string)
    - Outputs: x (list of integers the same length as list KXmers)
- generate_yplot
    - Generates a list of floats, where each value in the list is a current value from the dataframe KXmer_signal, which corresponds to a 4mer in the list KXmers.
    - Inputs: KXmers (list where each entry is a 4-character string), KXmer_signal (Pandas dataframe)
    - Outputs: y (a list of float values of the same length as KXmers)
- plot_signal
    - Generates a step function plot where the height of each step corresponds to the experimental unitless normalized current value for each base in the sequence being simulated, and the x-position of each step corresponds to the position of each base in the sequence.
    - Inputs: x (list of integers), y (list of floats)

- Outputs: Step-function plot generated by Matplotlib.

Interactions to accomplish use-case:
Use case: The user has experimentally run a known sequence on Nanopore and is seeking a visual aid to confirm that the experimental signal is what is expected before decoding. In this use case, the user will first run the function import_kmer_library to generate the pandas dataframe KXmer_signal. The user will then input their genetic sequence of interest as user_input (str) when prompted by the function input_to_4mers, which will then generate a list of corresponding 4mers called KXmers. KXmers and KXmer_signal will be used as inputs to the function generate_yplot, which will generate y, a list of current signals as floats which correspond to each base in the sequence. generate_xplot uses the list KXmers as an input to generate the x-data for the step function plot in the form of a list of integers, named x. Running the function plot_signal uses the lists x and y as inputs to generate a step function plot using Matplotlib.

Required packages:
Pandas is used to read data from a .csv file containing a library of possible 4mers and their associated experimental nanopore signal into Python.
NumPy is used for numerical operations involved in signal processing and in the generation of synthetic noise. This uses data imported by Pandas.
MatPlotLib is used to generate the step function plot which is output by the package, dependent on numerical operations and data accomplished by Pandas and NumPy.