

Cel projektu:

Pierwszym celem projektu było napisanie skryptu, który na podstawie danych z PDB znajduje redundantne łańcuchy (tj. takich, gdzie przynajmniej 2 różne łańcuchy mają identyczną sekwencję). Drugim celem było stworzenie algorytmów do porównania struktury drugorzędowej tych łańcuchów.

Struktury drugorzędowej:

Dla każdego łańcucha z danego klastra należało otrzymać strukturę drugorzędową. W tym celu skorzystano ze strony <http://rnapdbee.cs.put.poznan.pl/>, która zwraca strukturę drugorzędową dla wybranej cząsteczki RNA na podstawie pliku pdb. Ponieważ nie jest możliwe otrzymanie struktury drugorzędowej w sposób zautomatyzowany, interesujące wyniki należało pobrać ręcznie.

Otworzyłem funkcję konwertować pdb file na fasta string (ale nie działa).

Szczegóły implementacyjne

Funkcja NeedlemanWunsch- funkcja do pasowania

Funkcja przyjmuje 2 parametry. Punktacja podstawowa ustawiona została na dopasowywanie = 1, niedopasowanie i przerwa = -1. Wyświetlane są sekwencje po dopasowaniu, procent ich identyczności oraz punktacja

Funkcja metrykaGóry

Funkcja przyjmuje 2 argumenty będące sekwencjami w notacji kropkowo-nawiasowej. W implementacji również obecna jest struktura stosu. Funkcja zwraca wartość odległości pomiędzy sekwencjami.

Například:

$S1 = ..(((.....)))..$ $vS1 = (0; 0; 1; 2; 3; 3; 3; 3; 3; 3; 2; 1; 0; 0; 0) = 24$
 $S2 = (((.....)))..$ $VS1 = (0; 1; 2; 3; 4; 4; 4; 4; 4; 4; 3; 2; 1; 0; 0) = 36$
 $S3 = ..(((.....)))..$ $vS3 = (0; 0; 1; 2; 3; 4; 4; 4; 4; 3; 2; 1; 0; 0; 0) = 28$
 $dm(S1; S2) = |24 - 36| = 12$ $dm(S1; S3) = |24 - 28| = 4$

Funkcja metryka:

Zwraca procent liczby niezgodności w dot-bracketach dla dwóch sekwencji. Przyjmuje wartości od 0 (identyczne struktury 2-rzędowe) do 100.

Funkcja get_pdb_file:

Funkcja pobiera pdb z serwera do pliku.

Funkcja check_lancuchow

Analiza wyników

Na podstawie przeanalizowanych wyników można wysunąć hipotezę, że jednym z najczęstszych przypadków było zaliczenie do klastra dwóch (lub więcej) łańcuchów tej samej cząsteczki –

```
>strand_A
gGAAGUUCGGUCUUCGGACCGGCUUCC
.((((((((((((.....)))))))))).
>strand_A
gGAAGUUCGGUCUUCGGACCGGCUUCC
((((((((((((.....))))))))))
```

