Sekwencjonowanie łańcuchów DNA z błędami negatywnymi i pozytywnymi

Autor: Mateusz Oleszek, nr. 144608

Ogólny opis

Algorytm bazuje na koncepcji Profesora Jacka Błażewicza

Wygeneruj graf skierowany pełny gdzie wierzchołki to oligonukleotydów. A każda krawędź pomiędzy nimi ma koszt zależny od stopnia pokrycia pomiędzy nimi, równy *l-k*. Gdzie *l* to długość oligonukleotydów a k ilość pokrywających się aminokwasów. Krawedzię będą więc miały koszty 1 do *L*.

Sekwencja genomu powstanie ze znalezionej ścieżki we grafie. Poszukiwana jest ścieżka próbująca balansować dwa parametry, jak najmniejszy koszt wynikający z przejścia ścieżki i największy zysk pochodzący z odwiedzenia jak największą ilość wierzchołków, przy jednoczesnym ograniczeniu kosztu sekwencji nie większym niż *N-I* (w przeciwnym razie długość sekwencji byłaby większa niż *N*)

Jest ona znajdowana przez rozwiązanie selektywnego problemu komiwojażera, przy wykorzystaniu algorytmu genetycznego. Algorytm genetyczny bierze pewną losową pólę rozwiązań, wybiera z nich te najlepsze według pewnej funkcji zysku (w naszym przypadku biorącej pod uwagę koszt ścieżki i zysk z odwiedzonych wierzchołków), a następnie na ich podstawie poprzez różne modyfikacje (rozmnażanie i mutacje) tworzy nowy zestaw potencjalnych rozwiązań. Nadzieją jest, że z kolejnymi iteracjami uda się znajdować coraz lepsze rozwiązania.

Terminologia:

- rodzic/dziecko: potencjalne rozwiązanie problemu komiwojażera, reprezentowany jako uszeregowana lista kolejnych wierzchołków ścieżki.
- populacja: zbiór potencjalnych rozwiązań.

Kroki algorytmu genetycznego

- 1. Wygenerowanie inicjalnej populacji o wielkości S
- 2. Przez *i* iteracji:
 - 1. Wybór najlepszego odsetka populacji jako rodziców
 - 2. Rozmnażanie rodziców poprzez krzyżówki w celu wygenerowaniu zbioru nowych dzieci o wielkości S
 - 3. Mutacja dzieci
 - 4. Stworzenie nowej populacji na podstawie zmutowanych dzieci

Generowanie inicjalnej populacji

- W pętli S razy:
 - 1. Wylosuj długość śnieżki n od 2 do N
 - 2. Wylosuj kolejne *n* wierzchołki ścieżki
 - 3. Zapisz ścieżkę i jej koszt. Jeśli długość sekwencji przekracza N, wróć do kroku 1. i wygeneruj ją ponownie.

Wybór najlepszego odsetka populacji

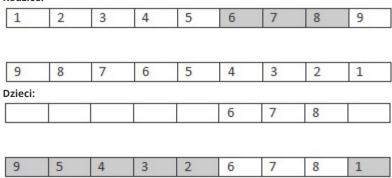
Każda sekwencjia w populacji dostaje ocenę w postaci wzoru $\left|\frac{1}{2*w-k-(W+2)}\right|$ który będzie dawał wyniki z przedziału (0;1>. w to liczba wierzchołków w ścieżce, k to koszt ścieżki, a W to liczba wszystkich wierzchołków w grafie. Przy tym wzorze idealny wynik, taki który przechodzi przez wszystkie wierzchołki w grafie i w każdym przejściu wykorzystujący krawędzie o minimalnym koszcie 1, będzie miał najlepszą ocenę 1, a gorsze ścieżki będą miały oceny ułamkowe.

Te oceny są wykorzystywane jako wagi przy wybieraniu losowych *S*sel* rodziców z populacji. Gdzie *sel* to selektywność z wartością w przedziale (0;1))

Sposób krzyżowania populacji

- Dopóki nie zostanie stworzona nowa populacja o wielkości S:
 - Z pewnym prawdopodobieństwem zajdzie jedno z dwóch wydarzeń:
 - 1. Losowy rodzic zostanie bezpośrednio przeniesiony do dzieci
 - 2. nowe dziecko zostanie stworzone przez krzyżówkę 2 losowo wybranych rodziców. Krzyżówka działa na zasadzie wybrania pewnej ciągłej podścieżki w jednym dziecku, zostawieniu tylko jej i usunięciu reszty, i wypełnieniem tej reszty przed i po podścieżką elementami z drugiego dziecka, w tej kolejności w jakiej tam występują. Jeśli po krzyżówce koszt ścieżki jest wyższy niż akceptowalny będzie ona powtarzana do skutku lub ustalonej ilości powtórzeń.

Rodzice:



Sposób mutacji

Każde dziecko ma pewne prawdopodobieństwo przejścia mutacji. Jeśli tak się stanie może się wydarzyć jedna z 3 rzeczy.

- 1. 2 losowe wierzchołki zostaną zamienione ze sobą miejscami
- 2. Zostanie dodany losowy wierzchołek w losowym miejscu w ścieżce.
- 3. Zostanie usunięty losowy wierzchołek ze ścieżki.

Jeśli po mutacji koszt ścieżki wzrósłby ponad limit inne mutacje będą po kolei powtarzane w pętli aż nie powstanie akceptowalna (albo do ustalonej granicy powtórzeń)

Źródła i inspiracje

- 1. https://jaketae.github.io/study/genetic-algorithm/
- $\textbf{2.}\ \underline{https://towardsdatascience.com/evolution-of-a-salesman-a-complete-genetic-algorithm-tutorial-for-python-6fe5d2b3ca35}$
- 3. https://www.theprojectspot.com/tutorial-post/applying-a-genetic-algorithm-to-the-travelling-salesman-problem/5