Algorytmy kombinatoryczne w bioinformatyce

wykład 4: dopasowanie sekwencji, poszukiwanie motywów

prof. dr hab. inż. **Marta Kasprzak**
Instytut Informatyki, Politechnika Poznańska

Dopasowanie sekwencji

- Badanie podobieństwa sekwencji stanowi podstawę wielu gałęzi bioinformatyki: asemblacji, konstrukcji drzew filogenetycznych, czy też znajdowania rodzin białek odpowiedzialnych za tę samą funkcję w różnych organizmach
- Ze względu na mutacje w sekwencjach lub błędy eksperymentalne popełnione na etapie ich odczytywania, dopasowanie dokładne sekwencji nie znajduje tu zastosowania. Należy założyć występowanie insercji, delecji oraz substytucji pozycji w sekwencjach (nukleotydowych, aminokwasowych), co skutkuje dopasowaniem z pewnym odsetkiem niezgodności oraz wstawionych spacji

Dopasowanie dwóch sekwencji

- Odległość pomiędzy sekwencjami można mierzyć minimalną liczbą operacji (wstawienia, usunięcia, zamiany znaku) potrzebnych do przetransformowania jednej sekwencji w drugą (*odległość Levenshteina*)
- Taki schemat obliczania odległości nie premiuje dopasowanych znaków, a więc w efekcie długości dopasowania. Nie nadaje się do znajdowania dopasowania o nieustalonej długości
- Częściej stosowane podejście polega na maksymalizacji funkcji oddającej podobieństwo sekwencji, w którym przyznaje się punkty za zgodność znaków i odejmuje punkty za ich niezgodność lub spację

Dopasowanie dwóch sekwencji

- Istnieje dokładny algorytm wielomianowy oparty na programowaniu dynamicznym, obliczający optymalne (przy założeniu pewnej przyjętej punktacji za każdą z operacji) dopasowanie dwóch sekwencji
- Sposób punktacji zależy od zastosowania dopasowania:
  — sekwencje nukleotydowe vs. aminokwasowe,
  — sekwencje homologiczne vs. niehomologiczne,
  — sekwencje krótkie vs. długie,
  a także od rodzaju eksperymentu, w którym sekwencje zostały pozyskane (spodziewane błędy)
- Przykładowa punktacja: +1 za zgodność pary znaków, –1 za niezgodność, –1 za każdą wprowadzoną spację

Dopasowanie dwóch sekwencji

- Globalne dopasowanie dwóch sekwencji

[S.B. Needleman i C.D. Wunsch,
J. Molecular Biology 48 (1970)]

$$M[0,0] = 0$$

$$M[i,j] = \max \begin{cases} M[i-1, j-1] + s(i,j) \\ M[i-1, j] + g \\ M[i, j-1] + g \end{cases}$$

$$i = 0..n, j = 0..m$$

$$s(i,j) = \pm 1, \quad g = -1$$

ATCAC-AGTA
AG-ACTACT-

		A	G	A	C	T	A	C	T
	0	-1	-2	-3	-4	-5	-6	-7	-8
A	-1	**1**	0	-1	-2	-3	-4	-5	-6
T	-2	0	**0**	-1	-2	-1	-2	-3	-4
C	-3	-1	**-1**	-1	0	-1	-2	-1	-2
A	-4	-2	-2	**0**	-1	-1	0	-1	-2
C	-5	-3	-3	-1	**1**	**0**	-1	1	0
A	-6	-4	-4	-2	0	0	**1**	0	0
G	-7	-5	-3	-3	-1	-1	0	**0**	-1
T	-8	-6	-4	-4	-2	0	-1	-1	**1**
A	-9	-7	-5	-3	-3	-1	1	0	**0**

Dopasowanie dwóch sekwencji

- Lokalne dopasowanie dwóch sekwencji

[T.F. Smith i M.S. Waterman,
J. Molecular Biology 147 (1981)]

$$M[i,j] = \max \begin{cases} M[i-1, j-1] + s(i,j) \\ M[i-1, j] + g \\ M[i, j-1] + g \\ 0 \end{cases}$$

$$i = 0..n, j = 0..m$$

$$s(i,j) = \pm 1, \quad g = -1$$

TACATAGTA
AGAC-TACT

		A	G	A	C	T	A	C	T
	0	0	0	0	0	0	0	0	0
T	0	0	**0**	0	0	1	0	0	1
A	0	1	0	**1**	0	0	2	1	0
C	0	0	0	0	**2**	1	1	3	2
A	0	1	0	1	**1**	1	2	2	2
T	0	0	0	0	0	**2**	1	1	3
A	0	1	0	1	0	1	**3**	2	2
G	0	0	2	1	0	0	2	2	1
T	0	0	1	1	0	1	1	1	3
A	0	1	0	2	1	0	2	1	2

Dopasowanie dwóch sekwencji

- Semiglobalne dopasowanie dwóch sekwencji

$$M[0,0] = M[i,0] = M[0,j] = 0$$

$$M[i,j] = \max \begin{cases} M[i-1,j-1] + s(i,j) \\ M[i-1,j] + g \\ M[i,j-1] + g \end{cases}$$

$$i = 1..n,\ j = 1..m$$

$$s(i,j) = \pm 1, \quad g = -2$$

```
ATCAAG-CAAC
    AGACTACTC
```

		A	G	A	C	T	A	C	T	C
	0	0	0	0	0	0	0	0	0	0
A	0	1	-1	1	-1	-1	1	-1	-1	-1
T	0	-1	0	-1	0	0	-1	0	0	-2
C	0	-1	-2	-1	0	-1	-1	0	-1	1
A	0	1	-1	-1	-2	-1	0	-2	-1	-1
A	0	1	0	0	-2	-3	0	-1	-3	-2
G	0	-1	2	0	-1	-3	-2	-1	-2	-4
C	0	-1	0	1	1	-1	-3	-1	-2	-1
A	0	1	-1	1	0	0	0	-2	-2	-3
A	0	1	0	0	0	-1	1	-1	-3	-3
C	0	-1	0	-1	1	-1	-1	2	0	-2

Dopasowanie dwóch sekwencji

- Dokładny algorytm programowania dynamicznego ma złożoność czasową i pamięciową O($n \cdot m$). W praktycznych zastosowaniach, gdy wymagane jest obliczenie dopasowania dla wszystkich par sekwencji z dużego zbioru, lub też dopasowania dwóch długich sekwencji, czas obliczeń staje się problemem. Dlatego podejmuje się próby ograniczenia liczby kroków algorytmu, co skutkuje podejściem heurystycznym
- Jednym z rozpowszechnionych podejść jest wypełnianie tablicy programowania dynamicznego jedynie w okolicy przekątnej. W wielu przypadkach wziętych z życia takie podejście daje wystarczająco satysfakcjonujące wyniki

[W.J. Wilbur i D.J. Lipman, *Proc. Natl. Acad. Sci. USA* 80 (1983)]

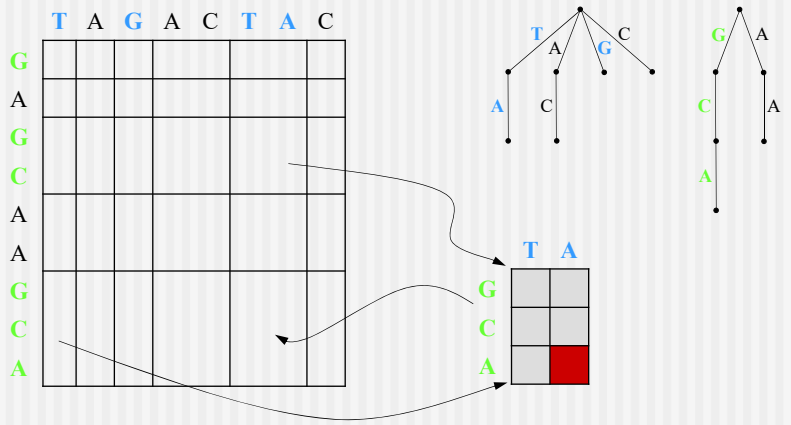
Dopasowanie dwóch sekwencji

[M. Crochemore i in., *SIAM J. Comput.* 32 (2003)]

- Algorytm wyliczający optymalne dopasowanie (globalne lub lokalne) dwóch sekwencji w czasie $O(h \cdot n^2/\log n)$, gdzie n jest długością sekwencji, a $0 < h \leq 1$ jest entropią sekwencji
- Entropia mierzy stopień losowości ciągu znaków. Entropia jest mała, jeśli porządek w ciągu jest duży
- Obie dopasowywane sekwencje są na wstępie poddawane faktoryzacji LZ (Lempel-Ziv)
- Tablica programowania dynamicznego dzielona jest na bloki o zmiennej wielkości, w których porównywane jest słowo z pierwszej sekwencji ze słowem z drugiej sekwencji, oba słowa będące rezultatem faktoryzacji

Dopasowanie dwóch sekwencji

[M. Crochemore i in., *SIAM J. Comput.* 32 (2003)] – cd.



Dopasowanie dwóch sekwencji

[M. Crochemore i in., *SIAM J. Comput.* 32 (2003)] − cd.

- Fragmenty sekwencji są identyfikowane i przechowywane w słowniku i jeśli dane słowo powtórzy się w sekwencji, jest kodowane przez odpowiednie wskaźniki
- Każde następne identyfikowane słowo jest jednym ze słów zidentyfikowanych wcześniej plus jeden znak (wyjątkiem może być koniec sekwencji). Przykładowo, sekwencja AACGACGA jest dzielona w faktoryzacji LZ na pięć słów: A, AC, G, ACG, A
- Przyspieszenie jest osiągane przez wykorzystanie w trakcie wypełniania macierzy części obliczeń dokonanych wcześniej dla podobnego słowa. Bloki są tak konstruowane, że w każdym z nich porównywana jest tylko jedna nowa para znaków

Dopasowanie dwóch sekwencji

- Poza opisanym wcześniej podejściem opartym na stałej punktacji za każdą zgodność, niezgodność czy spację, stosuje się w dopasowaniu sekwencji nukleotydowych i aminokwasowych tzw. afiniczne kary za spacje oraz macierze substytucji
- Model kar afinicznych polega na obarczeniu wysoką karą pierwszej spacji z rzędu i niższymi karami kolejnych spacji. Uzasadnieniem biologicznym jest większe prawdopodobieństwo wystąpienia na skutek ewoluowania sekwencji takiej „blokowej" różnicy niż wielu pojedynczych insercji. Podobnie jak w modelu kar stałych, wysokość kary ustalana jest w zależności od rodzaju problemu i zamierzonego efektu

Dopasowanie dwóch sekwencji

- Dopasowanie „literalne” częściej stosowane jest w przypadku sekwencji nukleotydowych. W sekwencjach aminokwasowych trudniej o idealne dopasowanie pary znaków z uwagi na większy alfabet oraz podobieństwo różnych aminokwasów (substytucja w trakcie ewolucji). Zastosowanie punktacji z macierzy substytucji przybliża rezultat do stanu naturalnego
- Macierze substytucji dla aminokwasów zawierają wartości liczbowe odzwierciedlające prawdopodobieństwo zamiany w trakcie ewolucji jednego aminokwasu w drugi. Najpopularniejsze macierze to serie BLOSUM i PAM. Przykładem macierzy substytucji dla nukleotydów jest EDNAFULL/NUC4.4 [ftp.ncbi.nih.gov/blast/matrices/]

Dopasowanie wielu sekwencji

- Problem dopasowania wielu sekwencji jest silnie NP-trudny i może zostać rozwiązany metodą programowania dynamicznego w sposób dokładny w czasie $O(2^k n^k)$, gdzie k jest liczbą sekwencji a n ich długością
- Naturalną heurystyką jest obliczenie optymalnych dopasowań wszystkich par sekwencji i połączenie ich w całość, co nie jest łatwe w przypadku znacznych rozbieżności w dopasowaniach
- Wynikowe dopasowanie przełożyć można na postać *sekwencji konsensusowej* reprezentującej w jak najbardziej zbliżony sposób cały zbiór. Zwykle stosuje się w tym celu regułę względnej większości, rzadziej regułę bezwzględnej większości

Dopasowanie wielu sekwencji

- Progresywna strategia dopasowania wielu sekwencji
  [M.S. Waterman i M.D. Perlwitz, *Bull. Math. Biol.* 46, 1984]

  Dopasowanie wielu sekwencji jest konstruowane począwszy od dopasowania najbliższej sobie pary sekwencji, a następnie przez dodawanie kolejnych bliskich sekwencji.

  Strategia wynika z założenia, że najbliższe sobie sekwencje prawdopodobnie pochodzą z bliskich ewolucyjnie organizmów i że ich dopasowanie odpowiada najbardziej wiarygodnej informacji. Informacja ta jest następnie propagowana na kolejne etapy tworzenia dopasowania.

  Łączyć można zarówno sekwencję z podzbiorem dopasowanych już sekwencji, jak i dwa podzbiory niezależnie dopasowanych sekwencji.

Dopasowanie wielu sekwencji

- Strategia iteracyjnego poprawiania dopasowania
  [D. Sankoff i in., *J. Mol. Evol.* 7, 1976]

  Początkowe dopasowanie, utworzone heurystyką, poprawiane jest w serii kroków optymalizacji lokalnej.

- Strategia gwiazdy
  [D. Gusfield, *Bull. Math. Biol.* 55, 1993]

  Obliczane jest dopasowanie wszystkich par sekwencji, a następnie wskazywana jest sekwencja najbliższa wszystkim innym ze zbioru, na której konstruowane jest dopasowanie wynikowe. Metoda nie sprawdza się na zbiorze mocno różniących się sekwencji.

Dopasowanie wielu sekwencji

[Y. Zhang i M.S. Waterman, *J. Comput. Biol.* 10 (2003)]

- Metoda *EulerAlign* − zamodelowanie problemu za pomocą problemu poszukiwania ścieżki w grafie. Wykorzystuje graf Pevznera zastosowany wcześniej do rozwiązania problemu sekwencjonowania DNA
- Nowość tego podejścia polega na pominięciu standardowo używanego schematu tworzenia sekwencji konsensusowej na podstawie dopasowania sekwencji. Tutaj generowana jest sekwencja konsensusowa przed poznaniem samego dopasowania. Co więcej, pomija się w ogóle etap obliczania dopasowania wszystkich par sekwencji

Dopasowanie wielu sekwencji

[Y. Zhang i M.S. Waterman, *J. Comput. Biol.* 10 (2003)] − cd.

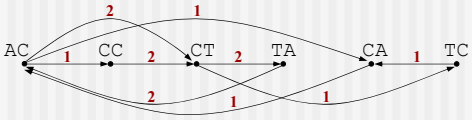
Algorytm *EulerAlign*

1. Budowa grafu skierowanego z *k*-merami pochodzącymi z sekwencji umieszczonymi w łukach grafu
2. Transformacja grafu do postaci acyklicznej
3. Uzyskanie sekwencji konsensusowej z grafu jako tej, która odpowiada ścieżce o największej wadze
4. Dopasowanie wszystkich sekwencji wejściowych do uzyskanego rozwiązania

Dopasowanie wielu sekwencji

[Y. Zhang i M.S. Waterman, *J. Comput. Biol.* 10 (2003)] – cd.

`sekwencje: ACCTACA, CCTCACT, ACTACT`
`k-mery (k=3): ACC, CCT, CTA, TAC, ACA, CTC, TCA, CAC, ACT`



Waga w łukach jest liczbą sekwencji zawierających dany *k*-mer, nie licznością *k*-merów w sekwencjach (patrz `ACT`).

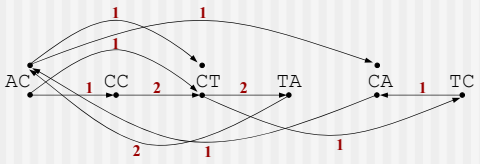
Rozwiązaniem jest acykliczna ścieżka w grafie o maksymalnej sumarycznej wadze.

Transformacja grafu do postaci acyklicznej ma uprościć poszukiwanie ścieżki (co będzie możliwe w liniowym czasie) przy zachowaniu maksimum informacji o podobieństwie sekwencji wejściowych. Informacja ta jest reprezentowana przez łuki o dużej wadze. Transformacja realizowana jest procedurą heurystyczną.

Dopasowanie wielu sekwencji

[Y. Zhang i M.S. Waterman, *J. Comput. Biol.* 10 (2003)] – cd.

`sekwencje: ACCTACA, CCTCACT, ACTACT`
`k-mery (k=3): ACC, CCT, CTA, TAC, ACA, CTC, TCA, CAC, ACT`



Dwa optymalne rozwiązania: **ACCTACA**, **ACCTACT**

```
ACCT-ACA        ACCT-ACT
ACCT-ACA        ACCT-ACA
-CCTCACT        -CCTCACT
AC-T-ACT        AC-T-ACT
```

Poszukiwanie motywów

- Poszukiwanie charakterystycznych ciągów w sekwencjach nukleotydowych lub aminokwasowych jest częstym problemem biologicznym. Dotyczy np. regionów promotorowych w sekwencjach nukleotydowych oraz fragmentów odpowiedzialnych za strukturę białka w sekwencjach aminokwasowych
- Poszukiwanie może dotyczyć motywów o stałej strukturze, identycznej w badanych sekwencjach (np. najdłuższe powtórzenia), ale najczęściej wystąpienia tego samego motywu w różnych organizmach różnią się w szczegółach (np. wskutek ewolucji), za to posiadają charakterystyczny wspólny „szkielet" odpowiedzialny za ich funkcję

Poszukiwanie motywów

- Przedmiotem poszukiwań może być pojedyncze wystąpienie motywu w sekwencji lub też cała seria różnych motywów. W tym drugim przypadku kolejne wystąpienia motywów mogą następować (bądź nie) w zadanym porządku
- Dodatkowo można nakładać ograniczenia na odległości pomiędzy wystąpieniami motywów. Przykładowo, region promotorowy ma pojawić się w przedziale zdefiniowanym w pewnej odległości przed kodonem startu, kluczowe fragmenty sekwencji białkowej mają zawierać się w oknach o zadanym położeniu wewnątrz sekwencji
- Można poszukiwać motywy znane lub nieznane

Poszukiwanie motywów

- Wystąpienia tego samego motywu w różnych sekwencjach mogą różnić się sekwencją znaków, długością (istotnie), pozycją (znacznie), może się także zdarzyć brak wystąpienia motywu w którejś z sekwencji ze zbioru
- W grupie problemów związanych z poszukiwaniem motywów można wyróżnić poszukiwanie najdłuższego motywu w zbiorze sekwencji, poszukiwanie największej liczby motywów, poszukiwanie najbardziej zakonserwowanego zbioru motywów
- Problem poszukiwania motywów staje się trudny obliczeniowo przy wielu sekwencjach i zmiennych motywach

Poszukiwanie motywów

[C. Boucher i in., *Lect. Notes Comput. Sci.* 4645 (2007)]

- Metoda poszukiwania w zbiorze sekwencji motywów „słabych", zdegenerowanych, w których na stosunkowo wielu pozycjach dopuszcza się wystąpienie niezgodności
- Wykorzystywany jest tu model grafu nieskierowanego ważonego i w podejściu heurystycznym poszukiwane są gęste podgrafy o dużej wadze
- Zastosowanie wag daje nowej metodzie przewagę nad wcześniejszymi algorytmami, w których poszukiwano struktury zbliżone do klik w grafach bez wag. Autorzy wykryli widoczną różnicę w wagach klik odpowiadających i nieodpowiadających rzeczywistym motywom

Poszukiwanie motywów

[C. Boucher i in., *Lect. Notes Comput. Sci.* 4645 (2007)] − cd.

- W pierwszym etapie algorytm *MarkovCluster* poszukuje gęste i wysoko punktowane podgrafy w zdefiniowanym grafie, w drugim etapie dokładny algorytm znajduje w nich motywy
- Wierzchołki grafu odpowiadają wszystkim podciągom o długości l, krawędzie łączą wierzchołki z różnych sekwencji, jeśli odległość Hamminga pomiędzy podciągami jest nie większa niż $2d$, gdzie d jest liczbą dopuszczalnych zdegenerowanych pozycji w motywie. Waga krawędzi jest równa $l-k$ dla odległości Hamminga $d < k \leq 2d$, lub $10(l-k)$ dla $k \leq d$
- *Odległość Hamminga* dla dwóch sekwencji o tej samej długości jest liczbą pozycji, na których te sekwencje różnią się

Poszukiwanie motywów

- Zadanie

Należy skonstruować graf ważony metodą Boucher i in. dla następującego zbioru sekwencji: {ACCTACA, CCTCACT, ACTACT}. Wartości parametrów: długość podciągów l=4, liczba dopuszczalnych zdegenerowanych pozycji w motywie d=1.

- każdy podciąg o długości l to osobny wierzchołek
- wierzchołki są łączone krawędziami, jeśli reprezentujące je ciągi znaków pochodzą z różnych sekwencji oraz dzieli je odległość Hamminga k nie większa niż $2d$
- waga krawędzi jest równa $l-k$, jeżeli $d < k \leq 2d$, lub $10(l-k)$, jeżeli $k \leq d$

- W grafie tym można wyróżnić wiele klik, ale wagi pozwalają łatwo wybrać preferowane czteroliterowe podciągi: CCTA, CTAC, TACT, i końcowy motyw: CCTACT