Przedmiot obieralny Bioinformatyka, wykładowca prof. Marta Kasprzak

Materiały uzupełniające do wykładu 6: asemblacja

Na części slajdów podane są namiary na artykuł źródłowy opisujący daną metodę. Zachęcam do dalszej lektury osoby, które chciałyby pogłębić swoją wiedzę nt. danej metody.

SLAJD 2

Wcześniejsze techniki sekwencjonowania, laboratoryjne na żelu i SBH w różnych odmianach, zostały wyparte przez sekwencjonowanie nowej generacji, zautomatyzowane, wysokoprzepustowe i niewymagające etapu algorytmicznego. Pirosekwencjonowanie zaproponowane w 1996 r. stało się podstawą pierwszej technologii sekwencjonowania wysokoprzepustowego, tzw. sekwencjonowania 454 (od nazwy firmy 454 Life Sciences) uruchomionego w 2005 r. Obecnie już ta technologia nie jest wspierana. Na rysunku jasność znaku ma obrazować mniejszą bądź większą emisję światła.

SLAJD 3

Na podstawie obserwowanej w eksperymencie jasności można w przybliżeniu wywnioskować, ile nukleotydów danego rodzaju pod rząd przyłączyło się w danym kroku.

SLAJD 5

Odczytem jest nazywana pojedyncza sekwencja odczytana przez sekwenator. Odczyty z sekwenatora wchodzą na wejście algorytmów asemblacji wraz z wiarygodnościami, które mogą posłużyć do odsiania odczytów gorszej jakości lub do ich korekcji w słabszych miejscach. Obecnie najczęściej stosuje się protokoły sekwencjonowania z odczytami sparowanymi, gdzie na wyjściu sekwenatora podawane są pary odczytów, o których można powiedzieć, że są od siebie oddalone w badanym fragmencie DNA o odstęp określony w przybliżeniu jako pewien zakres wartości (np. dwa odczyty o długości 100 nukleotydów każdy oddalone od siebie o 50–150 nukleotydów), od czego są odstępstwa na skutek błędów eksperymentalnych.

SLAJD 6

Asemblacja *de novo* łańcuchów DNA to składanie sekwencji wynikowej na podstawie zbioru sekwencji wejściowych DNA, bez znajomości genomu referencyjnego, do którego można by dopasowywać rozwiązanie (łac. „*de novo*" znaczy „na nowo"). Proces ten nazywany jest też sekwencjonowaniem *de novo*, jednak tu pozostaniemy przy nazwie asemblacja dla odróżnienia od sekwencjonowania niższego poziomu (laboratoryjnego na żelu, SBH czy maszynowego sekwencjonowania nowej generacji). Składanie ze znajomością genomu referencyjnego nazywane jest resekwencjonowaniem i służy do wykrywania różnic w genomie pomiędzy sekwencjonowanym osobnikiem a wzorcowym. Na wejściu procesu asemblacji mamy zbiór sekwencji wykrytych wcześniej na etapie sekwencjonowania, w ogólności nie posiadamy nic więcej, ale w zależności od sposobu pozyskania sekwencji możemy dysponować dodatkową informacją. W przypadku sekwencjonowania nowej generacji będą to wiarygodności poszczególnych odczytanych nukleotydów, a w razie zastosowania protokołu dla odczytów sparowanych, także parametry z tym związane, np. średnia odległość pomiędzy odczytami. Do złożenia rozwiązania należy użyć wszystkich lub niemal wszystkich sekwencji wejściowych, w praktyce na wstępie odsiewa się część sekwencji zawierających nukleotydy o niskiej wiarygodności, na koniec pomija się w raportowanym rozwiązaniu pojedyncze nieużyte sekwencje lub bardzo krótkie złożone ścieżki. Asemblacja przypomina nieco sekwencjonowanie przez hybrydyzację, tam też mieliśmy na wejściu krótsze sekwencje i składana była z nich dłuższa, teraz jednak mamy do czynienia z bardziej uciążliwymi błędami i ze znacznie większą instancją. Rozwiązywane w praktyce instancje problemu asemblacji mogą zawierać kilkadziesiąt/kilkaset milionów odczytów o długości min. 100 nukleotydów każdy i składane są naraz całe chromosomy lub nawet całe genomy organizmów o długości setek milionów nukleotydów i więcej.

SLAJD 7

Sekwencje wejściowe dopasowywane są do siebie jak w dopasowaniu semiglobalnym (wykład 4), zwykle sufiks jednej nakłada się na prefiks drugiej, ale może też zachodzić zawieranie sekwencji. Na wyjściu procesu asemblacji mamy, w praktyce, nie jedną spójną sekwencję wynikową, a wiele takich rozłącznych sekwencji pokrywających (z przerwami) badany wycinek genomu, których nie dało się na podstawie posiadanej informacji połączyć w całość. Sekwencje takie nazywane są kontigami (ang. *contigs*, od *contiguous*). Czasami możliwe jest ich częściowe uszeregowanie na podstawie informacji pochodzącej ze sparowanych odczytów, gdy niektóre pary mają po jednym odczycie zawartym w różnych kontigach; takie struktury nazywane są z ang. *scaffolds*. Niemożność uzyskania jednej spójnej sekwencji na wyjściu dla spójnego badanego wycinka genomu może być spowodowana zarówno brakiem informacji na temat niektórych miejsc (nie zsekwencjonowano ich lub zsekwencjonowano, lecz pokrywające je odczyty/sekwencje okazały się za mało wiarygodne), jak i niedoskonałością programów do asemblacji, które z konieczności muszą być heurystykami o bardzo małej złożoności czasowej. Wiarygodność danych wejściowych jest powiązana z wiarygodnością nukleotydów zwróconą przez sekwenator, ale także z liczbą odczytów (sekwencji) pokrywających daną pozycję w genomie. Na rysunku wyszczególniono miejsca o pokryciu jedną i czterema sekwencjami. Ponieważ zakładamy teraz obecność w sekwencjach błędów pochodzących z etapu sekwencjonowania (insercje, delecje, zamiany nukleotydów), informacja pochodząca z jednej sekwencji wejściowej może być błędna i nie ma możliwości jej skorygowania. Gdy daną pozycję w genomie pokrywa większa liczba sekwencji, można oczekiwać, że większość sekwencji na danej pozycji będzie poprawnie odczytana i na tej podstawie będzie można skorygować błędy. Dużą uciążliwością instancji problemu asemblacji jest pochodzenie sekwencji/odczytów z obu komplementarnych nici badanego wycinka genomu, czyli inaczej niż w sekwencjonowaniu przez hybrydyzację, gdzie założone było pochodzenie oligonukleotydów z tej samej nici. Przy dopasowywaniu dwóch sekwencji do siebie trzeba zatem uwzględnić, że mogą być one czytane wprost, tak jak są podane na wejściu, albo jedną z nich trzeba przetłumaczyć na jej odwrotnie komplementarny odpowiednik. Jeśli jeden z dwóch wariantów odczytu sekwencji (wprost, odwrotnie komplementarnie) pasuje do reszty zbioru, a drugi nie, sytuacja jest jasna, najczęściej jednak przy dopuszczeniu błędów dopasowania oba warianty nakładają się lepiej lub gorzej z różnymi sekwencjami. W praktyce programy do asemblacji radzą sobie tak, że duplikują zbiór sekwencji wejściowych (dla każdej z nich dopisują jej odwrotnie komplementarny odpowiednik, co zwiększa i tak olbrzymi zbiór sekwencji do przetworzenia) i przy składaniu rozwiązania pilnują, aby użyć po jednym wariancie z każdej pary. Kontigi na wyjściu oczywiście także mają różną orientację.

SLAJD 8

Przykład przedstawia użycie w rozwiązaniu problemu asemblacji czterech sekwencji wejściowych czytanych wprost i jednej (fioletowej) w orientacji odwrotnie komplementarnej (czytanej od prawej do lewej i z przetłumaczeniem na komplementarne nukleotydy). Jest to jedno z możliwych rozwiązań, które można generować dla różnych orientacji sekwencji wejściowych z założeniem różnych kryteriów optymalizacji rozwiązania i różnych dopuszczonych progów jakości dopasowania par sekwencji (bierze się tu pod uwagę długość nakładających się odcinków sekwencji i odsetek błędów dopasowania na tym odcinku).

SLAJD 9

W literaturze nie funkcjonuje ścisłe sformułowanie problemu asemblacji, ze względu na złożoność obliczeniową oraz złożone uwarunkowania biologiczne jest to problem traktowany w bardzo rozmyty sposób, kryterium optymalizacji zwykle nie jest jedno ani jasno postawione. W ogólności za lepsze rozwiązanie problemu uważa się takie, które: ma mniejszą liczbę kontigów/scaffoldów, ma dłuższe najdłuższe kontigi, jest bardziej zwarte (sekwencje wejściowe są bardziej upakowane, nakładają się dłuższymi odcinkami), ma mniejszą liczbę błędów dopasowań par sekwencji, zawiera większą liczbę sekwencji wejściowych, jest bardziej zbliżone długością do oczekiwanego rozwiązania (o ile docelowa

długość jest możliwa do oszacowania); kryteria te są częściowo sprzeczne. Ostatecznie rozwiązanie oceniane jest najczęściej przez porównanie do genomu referencyjnego, o ile istnieje dla danego organizmu, chociaż genom nie jest obecny w procesie składania rozwiązania.

SLAJD 10

Wczesne algorytmy asemblacji mogły sprawdzać zgodność wszystkich par sekwencji poprzez zastosowanie programowania dynamicznego dla dokładnego dopasowania semiglobalnego. Wraz ze wzrostem rozmiaru danych wejściowych stało się to jednak niewykonalne. Obecnie podobieństwo sekwencji określane jest wstępnie zgrubnymi heurystykami, np. poprzez porównanie profili zawartości k-merów w sekwencjach, a programowanie dynamiczne (lub prostszy algorytm) stosowane jest do dokładniejszego określenia nałożenia sekwencji dla wybranych tylko par. Etap drugi, realizowany zazwyczaj podejściem grafowym i kończący się znalezieniem zbioru ścieżek (w miarę) pokrywających graf, również ze względu na rozmiar grafu nie może być zrealizowany optymalnie. Graf ten przypomina graf z metody Lysova i in. dla sekwencjonowania przez hybrydyzację, gdzie w wierzchołkach są sekwencje wejściowe, a łuki oddają ich relację nakładania się (tutaj mają wagi oddające jakość/długość dopasowania). Nawet ostatni etap sprawia problem, gdyż sprowadza się do wielokrotnego rozwiązywania trudnego obliczeniowo problemu dopasowania wielu sekwencji. Jak mogliśmy zaobserwować przy okazji wykładu 4, obliczone niezależnie optymalne dopasowanie par sekwencji nie musi złożyć się w optymalne dopasowanie naraz całego zbioru. Na tej samej zasadzie złożenie w całość dobrze dopasowanych par poprzednik-następnik może rodzić kolizje w dopasowaniu znaków ciągów niebędących bezpośrednimi sąsiadami.

SLAJD 11

Każdej sekwencji wejściowej odpowiada w grafie para wierzchołków, jeden dla sekwencji czytanej wprost, drugi dla odwrotnie komplementarnej. Najlepsze dopuszczalne nałożenia sekwencji przekładają się na łuki, które także są zdublowane: jeśli dwie sekwencje mogą łączyć się z dopuszczonym błędem, ich odwrotnie komplementarne odpowiedniki mogą być w analogiczny sposób połączone.

SLAJD 12

Sekwencje wejściowe identyczne jak na slajdzie 8. Dopuszczalne nałożenie sekwencji przyjęte w tym przykładzie zostało opisane poniżej rysunku (nie jest to założenie z oryginalnej metody). Łuki oznaczone pojedynczą linią reprezentują nałożenie typu prefiks-sufiks i poprowadzone są od sekwencji poprzedzającej. Łuki oznaczone linią podwójną reprezentują zawieranie i poprowadzone są od sekwencji dłuższej. Łuki nie są prowadzone pomiędzy wariantami tej samej sekwencji.

SLAJD 13

Las, do którego graf ma zostać zredukowany, ma mieć taką właściwość, że z żadnego wierzchołka nie wychodzą dwa łuki reprezentujące zazębianie się sekwencji oraz że z wierzchołka, do którego dochodzi łuk reprezentujący zawieranie, nie może wychodzić łuk reprezentujący zazębianie. Wagi łuków nie są naniesione na rysunki ze względu na czytelność.

SLAJD 14

Pierwszy etap redukcji realizowany jest algorytmem zachłannym wspomnianym na slajdzie 13 i kończy się pozostawieniem po jednym wierzchołku z każdej pary i wszystkich łuków pomiędzy tymi wierzchołkami.

SLAJD 15

Graf po redukcji odpowiada rozwiązaniu problemu asemblacji podanemu pod rysunkiem. Jest ono zupełnie inne niż to ze slajdu 8, głównie ze względu na dopuszczone bardzo krótkie nałożenia sekwencji wynikające z rozmiaru instancji. Dla danych pochodzących z rzeczywistych eksperymentów

biologicznych przyjmuje się oczywiście bardziej restrykcyjne warunki, w szczególności długość nakładających się podciągów musi być znacznie większa, gdyż i sekwencje są znacznie dłuższe.

SLAJD 16

Jak wygląda dekompozycja dłuższych sekwencji na krótsze l-mery i budowa z nich grafu Pevznera mogliśmy wcześniej zobaczyć na przykładzie algorytmu Zhanga i Watermana dla problemu dopasowania wielu sekwencji. Graf konstruowany w problemie asemblacji jest podobny (nie jest transformowany potem do postaci acyklicznej), inne mogą być wagi w łukach, np. oddające całkowitą liczbę wystąpień danego l-meru w instancji. W idealnej sytuacji poszukiwanym rozwiązaniem byłaby ścieżka Eulera, w rzeczywistości stosowane kryteria optymalizacji, obecność rozmaitych błędów i pochodzenie sekwencji wejściowych z obu nici DNA sprawiają, że problem jest trudny obliczeniowo. W podejściu tym zarzucono schemat dopasowanie-uszeregowanie-konsensus. Z tych trzech etapów pozostało tutaj uszeregowanie poprzedzone konstrukcją grafu. Nie ma dopasowania par sekwencji, po dekompozycji sekwencje posiadające wspólne podciągi będą reprezentowane w grafie przez przeplatające się ścieżki, sprawi to obecność w nich tych samych l-merów. Nie tworzy się sekwencji konsensusowej w zwyczajowym rozumieniu, rozwiązanie odczytywane jest wprost ze ścieżki, w której nie ma niezgodności etykiet sąsiednich wierzchołków. W pewnym sensie rozwiązanie może być efektem konsensusu w tych algorytmach, w których przeprowadza się korekcję błędów na zasadzie punktowej mutacji w l-merach, co zapoczątkowano w algorytmie EULER (slajdy 19-21).

SLAJD 18

Ponownie ta sama instancja jak we wcześniejszych przykładach. Dekompozycja przeprowadzona na słowa o długości 4, łuki ważone są liczbą wystąpień danego l-meru w instancji. Liczba ta jest tym większa, im większe pokrycie danego miejsca genomu dają sekwencje wejściowe (wtedy oddaje wiarygodność danego l-meru), ale także im więcej razy dany l-mer występuje w analizowanym wycinku genomu (i wtedy taki łuk powinien zostać włączony do rozwiązania więcej niż raz). Można to częściowo rozstrzygnąć, sprawdzając, czy sekwencje zawierające ten sam l-mer nakładają się na siebie czy nie. Jakkolwiek by nie zinterpretować wagi 2 w tym przykładzie, czyli przejść ten odcinek jeden lub dwa razy, rozwiązanie problemu asemblacji jest tu inne niż we wcześniejszych przykładach, dla przejścia jednokrotnego otrzymalibyśmy np. takie kontigi: ATCAGCAAC, GCATA, GACTC, GTAGA.

SLAJD 19

Graf dekompozycji nie zawiera błędnych połączeń, ale błędy na wejściu problemu są obecne. W grafie objawiają się one jako dodatkowe struktury: krótkie równoległe ścieżki, krótkie ślepe odgałęzienia. Rozróżnić, które tego typu struktury są poprawne, a które błędne, można na podstawie liczności składających się na nie l-merów (błędne powinny być mniej liczne). Błędy korygować można ingerując w strukturę grafu, ale można też przed jego utworzeniem, przez wykrycie l-merów o małej liczności, które są bardzo podobne do innych o dużej liczności (zatem mogą być rezultatem błędnego odczytu danego miejsca w genomie i występują w małej liczbie sekwencji vs. duża liczba innych sekwencji pokrywających to samo miejsce w genomie i odczytanych poprawnie). W razie wykrycia takich l-merów można je skorygować punktowymi mutacjami (wstawieniem, usunięciem, zamianą znaku) do postaci l-merów o dużej liczności, przez co odpowiednie niepożądane struktury w grafie nie pojawią się. Mutacja w l-merze pochodzącym z pewnej sekwencji pociąga za sobą taką samą zmianę we wszystkich l-merach obejmujących w tej sekwencji zmienianą pozycję. Przykładowo, zmiana A na T w sekwencji wejściowej CGTACTG zdekomponowanej początkowo na {CGT, GTA, TAC, ACT, CTG} daje skorygowane l-mery {CGT, GTT, TTC, TCT, CTG}, analogiczną zmianę należy wprowadzić w odwrotnie komplementarnym odpowiedniku tej sekwencji. Stąd jedna mutacja może zmniejszyć liczebność zbioru wszystkich l-merów o nawet kilkanaście/kilkadziesiąt (w zależności od ich długości) i niektóre mutacje są bardziej cenne, te są wykonywane w pierwszej kolejności.

SLAJD 20

Przykład z konieczności został uproszczony względem propozycji z artykułu, gdzie l-mery słabe (rzadko występujące) mogły mieć większą liczność niż 1 i więcej mutacji przypadających na jedną sekwencję wejściową było dopuszczonych. W ogólności dopuszczone są mutacje typu insercja, delecja, substytucja, tutaj tylko ta ostatnia. Na pierwszy ogień poszła najdłuższa sekwencja, w której zamiana jednego znaku daje w efekcie AGCAGCAAC i zmniejsza liczebność spektrum (zbioru l-merów) o 4 (ATCA zmienia się w obecny już element AGCA, TCAG w GCAG, plus ich odwrotnie komplementarne odpowiedniki). W tym momencie dwie najdłuższe sekwencje nie mogą być dalej zmieniane, najdłuższa ze względu na wprowadzoną właśnie jedną dopuszczalną mutację, a GCAGCA dlatego, że składa się teraz wyłącznie z mocnych l-merów. Kolejnym kandydatem do zmiany może być GTAGA, po zmianie GCAGA z zyskiem w postaci dwóch ubytków w spektrum. Trzecia mutacja zostaje wprowadzona do TATGC (TCTGC) i odejmuje kolejne cztery elementy spektrum. Pozostał jeden kandydat do ewentualnej mutacji, sekwencja GACTC, i jej l-mery można dopasować do innych z różnicą na jednej pozycji: GACT z GAGT i ACTC z AGTC. Problem w tym, że sparowane l-mery pokrywają ten sam odcinek tej samej sekwencji (w komplementarnych niciach) i mutacja C w G (lub odwrotnie) pociągałaby za sobą analogiczną zmianę w drugim elemencie z pary. Ponieważ zmiana taka nie spowodowałaby zmniejszenia liczebności spektrum, nie została wprowadzona. Korekta tu dokonana jest optymalna pod względem przyjętych kryteriów, w ogólności jednak tak nie będzie, gdyż realizowana jest podejściem heurystycznym. Dylemat, który z dwóch znaków ze spornej pozycji w l-merach uznać za obowiązujący, był tutaj rozstrzygany w jedyny możliwy sposób, w parze był zawsze jeden l-mer mocny i zmianie ulegał ten drugi. W sytuacji, gdy oba l-mery są słabe, należałoby wziąć pod uwagę ich liczność (w bieżącym przykładzie byłaby taka sama).

SLAJD 21

Graf po korekcji wyraźnie się zmniejszył względem grafu ze slajdu 18. Także dopasowanie skorygowanych sekwencji (po prawej u góry) jest bardziej oczywiste.

SLAJD 22

Etap korekcji błędów jest tu przeprowadzany poprzez usuwanie struktur w grafie.

SLAJD 24

Dekompozycja sprawia, że sekwencje wejściowe reprezentowane są w grafie przez ścieżki, które łączą się nie tylko końcami, czyli na oczekiwanych odcinkach ich pokrywania się, ale także przeplatają się w przypadkowych miejscach, gdy ten sam l-mer (lub jego prefiks lub sufiks stanowiący wierzchołek) wystąpi w dwóch niepowiązanych w rzeczywistości sekwencjach. Odwiedzanie wierzchołków bez wykorzystania dodatkowej informacji może odbyć się w zupełnie dowolnej kolejności. Wartość l jest parametrem takich metod, większa daje większą szansę na unikalne l-mery i na uniknięcie przypadkowego przeplatania się ścieżek, z kolei wtedy trudniej skorygować błędy i traci się korzyści płynące z tego podejścia.

SLAJD 26

Graf idealny ma znaczenie wyłącznie teoretyczne, gdyż nigdy nie zajdzie w rzeczywistej instancji taki sam odstęp pomiędzy odczytami ze wszystkich par. Graf przybliżony też nie do końca sprawdzi się w praktyce, ponieważ uwzględnienie odchylenia Δ staje się kłopotliwe ze wzrostem tej wartości już powyżej $½l$, a spodziewać się należy dużo większych odchyleń. W obu wariantach założona jest przynależność wszystkich odczytów do tej samej nici DNA. Za to graf jest interesujący jako koncepcja dedykowana odczytom sparowanym, pokazująca, jak uwzględnienie tej informacji może uprościć graf.

SLAJD 28

Przykład jest bardzo mały, dlatego też tak mały założony odstęp pomiędzy odczytami. Pary odczytów wygenerowane zostały z sekwencji podanej poniżej jako rozwiązanie, a następnie zdekomponowane

na pary l-merów, z których został utworzony graf. Każdy łuk reprezentuje parę l-merów i jest zaczepiony w wierzchołkach odpowiadających ich prefiksom/sufiksom o długości l-1 (analogicznie jak u Pevznera). Jeśli któraś para etykiet dla wierzchołka powtórzyłaby się, wierzchołek nie jest dodawany drugi raz do grafu, wykorzystywany jest istniejący. Rozwiązaniem w grafie jest ścieżka Eulera, która tłumaczona jest na sekwencję wynikową na podstawie etykiet wierzchołków i założonego pomiędzy nimi odstępu (czyli różnicy pomiędzy długością odczytu i l-1 powiększonej o odstęp pomiędzy odczytami). Zachęcam do samodzielnej konstrukcji tradycyjnego grafu Pevznera dla tej sekwencji i l=4, bez wykorzystania informacji o sparowaniu, i porównania go z grafem z rysunku.

SLAJD 29

Graf przybliżony powstaje z idealnego przez sklejenie wierzchołków, których lewe etykiety pokrywają się, a prawe są względem siebie przesunięte w ramach dopuszczonego zakresu, tu ±1. W grafie na slajdzie 28 były dwa takie wierzchołki, tu sklejone i opisane podwójną etykietą. To wystarczyło, żeby zaistniały w grafie dwie ścieżki Eulera i w efekcie dwa rozwiązania.