Wykład VIII: Kompatybilność zbioru drzew, puzzle drzewowe

Anna Gambin
21.11.2002

1 Różne definicje drzewa

Zastanowimy się nad alternatywnymi acz równoważnymi sobie sposobami na zdefiniowanie drzewa filogenetycznego. Omawiane przez nas definicje, własności i fakty były zaprezentowane w [3].

Niech T będzie nieukorzenionym drzewem, a e dowolną jego krawędzią. Przez $\mathcal{L}(T)$ będziemy oznaczać zbiór liści drzewa T. Usunięcie krawędzi e podzieli drzewo na dwie składowe oraz wyznaczy **podział** zbioru $\mathcal{L}$ na pewne dwa zbiory A oraz B (oznaczamy ten podział $A|B$ i nazywamy **dwupodziałem**). Jeśli $|A| = 1$ lub $|B| = 1$ nazywamy dwupodział trywialnym — odpowiada on krawędziom incydentnym liściom drzewa. Zbiór wszystkich dwupodziałów drzewa (odpowiadających wszystkim krawędziom) oznaczamy przez $\beta(T)$. Istnieje liniowy algorytm rekonstruujący drzewo na podstawie zbioru jego dwupodziałow $\beta(T)$.

Rozważmy teraz drzewo ukorzenione. Każdy wierzchołek wewnętrzny drzewa odpowiada podzbiorowi liści, które są jego potomkami (nazywamy ten podzbiór **klastrem**).

Zbiór klastrów odpowiadających wszystkim wierzchołkom drzewa T oznaczamy przez $\delta(T)$. Klaster $A \in \delta(T)$ jest trywialny jeśli $|A| = 1$ lub $A = \mathcal{L}(T)$.

Podobnie jak dla zbioru podziałów, zbiór klastrów pozwala jednoznacznie w czasie liniowym zrekonstruować drzewo.

Dla każdej trójki liści a, b, c istnieje tylko jedno nieukorzenione drzewo o tych trzech liściach, natomiast dla czterech liści istnieją trzy nieukorzenione drzewa binarne (nazywamy je **czwórkami** i oznaczmy $ab|cd$, $ac|bd$, oraz $ad|bc$).

Powiemy, że czwórka $ab|cd$ **pasuje** do drzewa T jeśli ścieżka łącząca a i b w drzewie T nie ma wspólnych wierzchołków ze ścieżką $c \rightsquigarrow d$.

Przez $q(T)$ oznaczamy zbiór czwórek pasujących do drzewa T. Każde drzewo mające chociaż jedną krawędź wewnętrzną (czyli nieincydentną do liścia) można zrekonstruować ze zbioru jego czwórek.

Dla każdych trzech liści a, b, c istnieją trzy ukorzenione drzewa binarne o tychże liściach (zwane **ukorzenionymi trójkami**): $ab|c$, $ac|b$ oraz $bc|a$. Ukorzeniona trójka $ab|c$ pasuje do drzewa T (jest kompatybilna z drzewem) jeżeli scieżka $a \rightsquigarrow b$ jest rozłączna ze ścieżką łączącą c z korzeniem. Zbiór trójek dla drzewa T będziemy oznaczać przez $r(T)$.

2 Operacja kontrakcji i indukowane poddrzewa

Jest jasne, że drzewa ewolucyjne zawierają pewną informację i wszyscy zgadzają się, że pewne drzewa zawierają jej więcej niż inne.

Podstawowe pytanie jakie się tutaj nasuwa to: *jak zdefiniwać zawartość informacji w drzewie?*

Zdefiniujemy za chwilę pojęcie rozszerzenia i kompatybilności drzewa i zajmiemy się następującym zagadnieniem: *dla danego zbioru drzew $\mathcal{T}$ rozstrzygnij czy istnieje drzewo, które rozszerza wszystkiem drzewa w $\mathcal{T}$, czyli drzewo zawierające łączną informację z $\mathcal{T}$.*

Okazuje się, że w przypadku drzew nieukorzenionych ten problem jest NP-zupełny. Jedynie w pewnych szczególnych przypadkach istnieją wielomianowe algorytmy.

Dla drzew ukorzenionych istnieje wielomianowy algorytm zaproponowany przez Aho w roku 1981 w kontekście relacyjnych baz danych [2].

Mimo tego, że dla drzew ukorzenionych umiemy rozwiązać problem kompatybilności, pewne problemy pozostają NP trudne, jak np znalezienie maksymalnego kompatybilnego pozbioru ukorzenionych trójek.

Pojęcia rozszerzenia i kompatybilności opierają się na operacji **kontrakcji** krawędzi, która usuwa wybraną krawędź z drzewa i utożsamia jej końce.

Rozważmy zbiór drzew nieukorzenionych o tym samym zbiorze liści. Operacja kontrakcji definiuje porządek częściowy na tym zbiorze: $S \leq_* T$ wttw S może być otrzymane z T przez kontrakcje pewnej liczby krawędzi. Można pokazać, że ten porządek jest półkratą dolną oraz kres dolny dwóch drzew jest drzewem ich **dokładnego konsensusu** (ta tajemnicza nazwa wyjaśni się wkrótce).

Dla ukorzenionego, lub nie, drzewa T oraz podzbioru jego liści $A \subseteq \mathcal{L}(T)$ rozważmy minimalny podgraf T który łączy elementy z A. Jeżeli T jest nieukorzenionym drzewem usuwamy wszystkie wierzchołki stopnia 2 (identyfikując incydentne do nich krawędzie) otrzymane w ten sposób nieukorzenione drzewo odnaczamy $T_{|A}$. Jeśli T jest drzewem ukorzenionym usuwamy wszystkie wierzchołki stopnia 2 z wyjątkiem jednego, który znajduje się najbliżej korzenia w oryginalnym drzewie. Staje się on korzeniem drzewa $T_{|A}$. W obydwu przypadkach nazywamy drzewo $T_{|A}$ **poddrzewem T indukowanym przez A** (inna stosowana nazwa to **homeomorficzne poddrzewo** T).

Znowu możemy uporządkować częściowo zbiór drzew przyjmując, że $S \leq_{\bullet} T$ jeśli S jest poddrzewem indukowanym T. Ten porządek nie jest półkratą (dlaczego ?).

Rozważmy kolejny porządek na zbiorze drzew: $S \trianglelefteq T$ jeżeli S może być otrzymane przez kontrakcje krawędzi z pewnego indukowanego poddrzewa T. Mówimy w takim przypadku, że drzewo T rozszerza drzewo S. Porządek $\trianglelefteq$ nazywamy **porządkiem kompatybilności**.

Powiemy, że zbiór drzew $\mathcal{T}$ jest **kompatybilny** jeżeli istniej drzewo T, które rozszerza wszystkie drzewa ze zbiou $\mathcal{T}$. Dwa drzewa T_1 oraz T_2 są kompatybilne jeżeli istniej drzewo, które rozszerza obydwa.

Dla zbioru drzew $\mathcal{T}$ niech $L = \mathcal{L}(T)$ będzie sumą zbiorów liści drzew z $\mathcal{T}$; przez $< \mathcal{T} >$ oznaczymy zbiór drzew o zbiorze liści L, które rozszerzają $\mathcal{T}$. Czyli $\mathcal{T}$ jest kompatybilny jeśli $< \mathcal{T} > \neq \emptyset$.

3 Charakteryzacje drzewa i kompatybilność

Dla drzewa T zdefiniowaliśmy zbiór trójek ukorzenionych $r(T)$ oraz zbior czworek $q(T)$ odpowiadających drzewu T (odpowiednio ukorzenionemu lub nie). Wykorzystując porządek kompatybilności możemy napisać:

$$r(T) = \{ab|c : ab|c \trianglelefteq T\}$$

$$q(T) = \{ab|cd : ab|cd \trianglelefteq T\}$$

Zachodzi następujący fakt (dla drzew nieukorzenionych, dla ukorzenionych analogiczny z trójkami):

Twierdzenie 1

$$S \trianglelefteq T \Leftrightarrow q(S) \subseteq q(T) \quad \textit{oraz} \quad \mathcal{L}(S) \subseteq \mathcal{L}(T)$$

Z powyższego twierdzenia wyciągamy wniosek że badając zbiór $< \mathcal{T} >$ możemy założyć bez straty ogólności że składa się on z czterolistnych drzew.

4 Zagnieżdżenia i n-taksony

Relacja zagnieżdżania $<_T$ została zaproponowana przez Adamsa [1] na potrzeby drzewa konsensusu. Niech A i B będą podzbiorami $\mathcal{L}(T)$; definiujemy $A <_T B$ jeśli $A \subset B$ oraz najbliższy wspólny przodek elementów ze zbioru A jest właściwym potomkiem najbliższego wspólnego przodka elementów z B. Mówimy wtedy, że A zagnieżdża się w B, oraz $A <_T B$ jest **zagnieżdżeniem** drzewa T. Porządek zagnieżdżeń definiuje jednoznacznie drzewo T. Można pokazać, że kompatybilność oznacza zachowanie całej informacji o zagnieżdżeniach.

Fakt 1 *Dla dwóch ukorzenionych drzew filogenetycznych S i T zachodzi:* $S \trianglelefteq T$ *wttw* $A <_S B \implies A <_T B$.

Inne podejście do informacji w drzewie oparte jest na pojęciu n-taksonów. Niech A i B będą podzbiorami liści pewnego ukorzenionego drzewa. Powiemy, że $(A)B$ jest **n-taksonem** dla drzewa T jeśli $n = |A \cup B|$ oraz liście należące do A są ze sobą bardziej spokrewnione niż z jakimkolwiek liściem ze zbioru B. Oznacza to, że istnieje pewien klaster $X \in \delta(T)$ taki że $A \subseteq X$ oraz $B \cap X = \emptyset$.

Ukorzeniona trójka $ab|c$ odpowiada 3-taksonowi $(a,b)c$, natomiast klaster $A \in \delta(T)$ odpowiada n-taksonowi $(A)B$, gdzie $B = \mathcal{L}(T) \setminus A$.

Zauważono, że $(A)B$ jest n-taksonem dla T wttw $ab|c \in r(T)$ dla każdych $a, b \in A$ oraz $c \in B$. Zachodzi następujący fakt:

Fakt 2 *Dla dwóch ukorzenionych drzew filogenetycznych S i T zachodzi:* $S \trianglelefteq T$ *wttw $(A)B$ jest n-taksonem dla T o ile $(A)B$ jest n-taksonem dla S.*

5 Czwórkowe puzzle

Chcielibyśmy poznać odpowiedź na pytanie: *czy zbiór drzew jest kompatybilny?* Kiedy myślimy o drzewach nieukorzenionych możemy zapytać: czy dany zbiór czwórek jest kompatybilny. Niestety znalezienie odpowiedzi na to pytanie jest problemem NP-trudnym. Czasami odpowiedź jest prosta, np jeśli rozważany zbiór czwórek zawiera dwie czwórki nad tym samym zbiorem liści (czyli sprzeczne ze sobą) to jest oczywiście niekompatybilny. Podobnie, dla zbioru drzew nieukorzenionych o tym samym zbiorze liści istnieje efektywny algorytm rozstrzygający kompatybilność (naszkicujemy go za chwilę).

Z kolei analogiczny problem dla zbioru drzew (trójek) ukorzenionych jest wielomianowo rozstrzygalny algorytmem opisanym w [2], którego pierwotne zastosowania dotyczyły zagadnień związanych z relacyjnymi bazami danych.

5.1 Kompatybilność dwupodziałów

Zastanowimy się na problemem kompatybilności dla zbioru dwupodziałów i zbadamy jego powiązania z problemem kompatybilności zbioru czwórek. Pozwoli nam to dla danego zbioru czwórek Q skonstruować drzewo T maksymalne (w sensie porządku kompatybilności $\trianglelefteq$) w zbiorze $\{T : q(T) \subset Q, \mathcal{L}(T) = \mathcal{L}(Q)\}$.

Oczywiście nie rozwiąże to problemu kompatybilności Q, gdyż w tym celu potrzebujemy drzewa T^*, dla którego $Q \subset q(T^*)$.

Zauważmy, że dwupodział jest dokładnie równoważny cesze binarnej, a jak pamiętamy kompatybilność zbioru cech binarnych jest wielomianowo rozstrzygalna.

Przypomnijmy kilka własności, które poznaliśmy już w kontekscie binarnych cech.

Twierdzenie 2

1. *Dwa dwupodziały $A|B$ oraz $C|D$ są kompatybilne wttw co najmniej jedno z przecięć: $A \cap C$, $A \cap D$, $B \cap C$ lub $B \cap D$ jest puste.*

2. *Zbiór dwupodziałów jest kompatybilny wttw jest parami kompatybilny.*

Powiedzieliśmy, że dla zbioru drzew $\mathcal{T}$ nad tym samym zbiorze liści łatwo potrafimy rozstrzygnąć jego kompatybilność. Dzieje się tak dlatego, że zbiór drzew $\mathcal{T}$ jest kompatybilny wttw zbiór wszystkich dwupodziałów drzew z $\mathcal{T}$ czyli $\cup_{T\in\mathcal{T}}\beta(T)$ jest kompatybilny.

Przejdziemy teraz do zagadnienia kompatybilności zbioru czwórek. Dla danego dwupodziału $A|B$ definiujemy **zbiór czwórek tego dwupodziału** następująco:

$$q(A|B) := \{ab|cd : a, b \in A, c, d \in B\}.$$

Łatwo widać, że $A|B \in \beta(T)$ wttw $q(A|B) \subseteq q(T)$ oraz $q(T) = \cup_{A|B\in\beta(T)}q(A|B)$.

Poniższy lemat wiąże zagadnienie kompatybilności dwupodziałów z istnieniem sprzecznych czwórek:

Lemat 1

1. *Dwa dwupodziały $A|B$ oraz $C|D$ są kompatybilne wttw $q(A|B)\cup q(C|D)$ nie zawiera sprzecznej czwórki.*

2. *Zbiór dwupodziałów $\mathcal{S}$ jest kompatybilny wttw $\cup_{A|B\in\mathcal{S}}q(A|B)$ nie zawiera sprzecznej czwórki.*

Powiemy, że czwórka $ab|cd$ jest sprzeczna z dwupodziałem $A|B$ jeżeli $ac|bd \in q(A|B)$ lub $ad|bc \in q(A|B)$. Zdefiniujmy zbiór czwórek niesprzecznych z danym dwupodziałem następująco:

$$\hat{q}(A|B) := \{ab|cd : ac|bd \notin q(A|B) \text{ oraz } ab|bc \notin q(A|B)\}.$$

Zachodzi następujący fakt:

Fakt 3 *Dwa dwupodziały $A|B$ oraz $C|D$ są kompatybilne wttw $q(A|B) \subseteq \hat{q}(C|D)$.*

Możemy rozszerzyć ten rezultat dla zbioru dwupodziałów:

Twierdzenie 3 *Zbiór dwupodziałów $\mathcal{S}$ jest kompatybilny wttw*

$$\cup_{A|B\in\mathcal{S}}q(A|B) \subseteq \cap_{A|B\in\mathcal{S}}\hat{q}(A|B).$$

Rowność zachodzi jeśli $\mathcal{S} = \beta(T)$ dla pewnego drzewa binarnego T.

Niech Q będzie niesprzecznym zbiorem czwórek. Rozważmy zbiór dwupodziałów $S = \{A|B : q(A|B) \subseteq Q\}$. Z powyższych własności wynika, że ten zbiór jest kompatybilny.

Zbudujmy teraz drzewo T, które zawiera wszystkie dwupodziały z S razem z trywialnymi podziałami (odpowiadającymi liściom). Drzewo T jest wyznaczone jednoznacznie dla danego Q i jest maksymalnym w sensie porządku kompatybilności drzewem w zbiorze drzew, dla których $q(T) \subseteq Q$.

Niestety drzewo T nie rozstrzyga, czy zbiór Q jest kompatybilny. Z Twierdzenia 3 wiemy, że drzewo binarne jest kompatybilne ze zbiorem czwórek wttw wszystkie jego dwupodziały są kompatybilne za zbiorem czwórek. Możemy problem kompatybilności Q przedstawić następująco:

1. Znajdź zbiór $\mathcal{S}(Q)$ dwupodziałów $A|B$, które są kompatybilne ze zbiorem czwórek Q, czyli $Q \subseteq \hat{q}(A|B)$.
2. Rozstrzygnij czy istnieje binarne drzewo T, takie że $\beta(T) \subseteq \mathcal{S}(Q)$.

I znowu, o ile drugi podproblem daje się rozwiązać wielomianowo to pierwszy pozostaje NP trudny (zauważmy, że liczba dwupodziałow kompatybilnych ze zbiorem Q może być wykładnicza).

6 Drzewa z trójek

Algorytm zaproponowany przez Aho [2] bierze na wejściu zbiór warunków w postaci $(a, b) < (c, d)$ oznaczających, że najbliższy wspólny przodek a i b jest potomkiem najbliższego wspólnego przodka c i d. Algorytm Aho był wielokrotnie modyfikowany i przyspieszany.

Rozważmy jego pochodną wersję, która bierze na wejściu zbiór trójek R oraz zbiór liści $S = \{x_1, x_2, \ldots x_n\}$ oraz zwraca jedno drzewo należące do $< R >$. Następnie zastanowimy się jakie własności ma to właśnie drzewo.

Algorytm jest rekurencyjny: w każdej fazie buduje graf $[R, S]$, którego wierzchołkami są liście drzewa, a każda krawędz odpowiada pewnej trójce ze zbioru R. Dla trójki $ab|c$ tworzymy krawędź (a, b) i etykietujemy ją c. Zauważmy, że graf może mieć wielokrotnie etykietowane krawędzie. Dla tak skonstruowanego grafu znajdujemy podział na spójne składowe i uruchamiamy algorytm rekurencyjnie dla każdej ze składowych. Złożoność algorytmu wynosi $\mathcal{O}(mn)$ gdzie n jest liczbą liści a $m = |R|$.

Algorithm 1 Aho (R, S)

jeśli $n = 1$ lub $n = 2$ zwróć drzewo jedno/dwu-listne.
wpp skonstruuj graf $G = [R, S]$
jeśli G jest spójny to NIE MA DRZEWA
dla każdej składowej S_i grafu $[R, S]$
  jeśli rekurencyjne wywołanie Aho(R, S_i) zwróci drzewo nazwij je T_i.
  wpp NIE MA DRZEWA
skonstruuj drzewo T podłączając drzewa T_i do wspólnego korzenia.

Opisany algorytm dla zbioru ukorzenionych trójek R buduje drzewo wttw zbiór R jest kompatybilny. Jedynym przypadkiem kiedy algorytm nie zbuduje drzewa jest sytuacja, że graf $[R, S]$ jest spójny. Zbiory R i S oznaczają tutaj zarówno pierwotne zbiory liści i ukorzenionych trójek, jak również ich podzbiory rozważane w kolejnych fazach algorytmu.

To spostrzeżenie pozwala nam na podanie następującej charakteryzacji kompatybilności zbioru trójek:

Twierdzenie 4 *Zbiór ukorzenionych trójek R nad zbiorem liści L jest kompatybilny wttw dla każdego podzbioru $S \subseteq L$ zawierającego co najmniej trzy elementy, graf $[R, S]$ nie jest spójny.*

Dowód: Jeżeli zbiór R jest kompatybilny to istnieje dla niego drzewo T, takie że $R \subseteq r(T)$. Wybierzmy podzbiór liści $S \subseteq L$, $|S| \geq 3$, i rozważmy poddrzewo indukowane przez ten zbiór $T_{|S}$. Niech M będzie korzeniem $T_{|S}$ (czyli najbliższym wspólnym przodkiem elementów z S). Bezpośredni potomkowie M indukują podział S na bloki. Zauważmy, że jeśli a i b należa do różnych bloków w tym podziale to nie istnieje element $c \in S$ taki że trójka $ab|c \in R$. Czyli w grafie $[R, S]$ nie istnieją krawędzie pomiędzy elementami z różnych bloków, a więc graf nie jest spójny.

Aby udowodnić odwrotną implikację załóżmy, że zbiór R nie jest kompatybilny, wtedy algorytm Aho nie zwróci drzewa, czyli w pewnej fazie działania graf $[R, S]$ jest spójny.

6.1 Własności drzewa Aho

Mówiliśmy już, że algorytm Aho konstruuje jedno szczególne drzewo ze zbioru $< R >$. Zastanowimy się teraz jakie własności ma to drzewo. Okazuje się, że drzewo to nie jest minimalne ani pod względem liczby wierzchołków wewnętrznych ani pod względem zawartych w nim ukorzenionych trójek. Np, dla zbioru $R = \{bc|a, bd|a, ef|a, eg|a\}$ drzewo Aho jest następujące: $(a, (b, c, d), (e, f, g))$, istnieje też inne drzewo rozszerzające zbiór R: $(a, (b, c, d, e, f, g))$, które ma mniej wierzchołków wewnętrznych i mniej trójek.

Bardzo ciekawą własnością jest związek drzewa Aho z drzewem konsensusu Adamsa, które zdefiniujemy niebawem, co pozwoli nam udowodnić następujące twierdzenie:

Twierdzenie 5 *Dla zbioru ukorzenionych trójek R nad zbiorem liści L, drzewo Aho dla R jest drzewem konsensusu Adamsa dla zbioru drzew $< R >$.*

Można pokazać, że drzewo Aho jest minimalnym drzewem w porządku kompatybilności $\trianglelefteq$, czyli jeśli T jest drzewem skonstruowaym przez algorytm Aho dla zbioru trójek R, to żadna kontrakcja T nie będzie kompatybilna z R. W szczególności dla danego drzewa T wynik algorytmu Aho dla zbioru $r(T)$ jest znów drzewem T.

Literatura

[1] E.N. Adams. N-trees as nestings: complexity, similarity, and consensus. *Journal of Classification*, 3:299–317, 1986.

[2] A.V. Aho, T.G. Sagiv, T. Szymanski, and J.D. Ullman. Inferring a tree from lowest common ancestors with an application to the optimization of relational expressions. *SIAM Journal of Computing*, 10(3):405–421, 1981.

[3] D. Bryant. Hunting for trees, buiding trees and comparing trees: theory and method in phylogenetic analysis. *phd thesis, Department of Mathematics, University of Canterbury*, 1997.

7 Zadania

1. Udowodnij, że dla ustalonego drzewa T porządek częściowy $\leq_*$ na zbiorze drzew, które są jego kontrakcjami oraz porządek $\leq_\bullet$ na zbiorze jego indukowanych poddrzew są kratami Boola.

2. Udowodnij Twierdzenie 1.

3. Algorytm Aho w zaprezentowanej przez nas wersji zakłada, że wszystkie trójki w zbiorze R są "rozwiązane" (czyli są drzewami binarnymi). Zmodyfikuj algorytm, ażeby dopuszczał w zbiorze R trójki będące gwiazdami.