Graphen und biologische Netze (WS 2016/17)

Inhaltsverzeichnis

1 Vorlesung 14.10.2016

1.1 Grundlagen der Graphen und biologische Netze

Graph: Knoten, Kanten (binäre Relationen)

Transitivität: implizite Verbindung (abhängig vom Kontext)
Labeled Graphs:

- Graph: (V, E)
- Labels: L_V (Knotenlabel), L_E (Kantenlabel)

$e \in E \Rightarrow \exists\ x, y \in V$: x und y sind die Endpunkte von e

Knoten-Labelfunktion α: $\alpha : V \to L_V : v \mapsto \alpha(v)$
Kanten-Labelfunktion β: $\beta : E \to L_E : e \mapsto \beta(e)$

ungerichtete Graphen

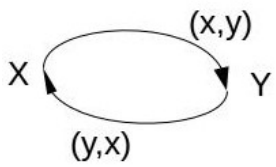
- Kante ist eine Menge von 2 (verschiedenen) Knoten
- $e = \{x, y\} = \{y, x\} \to$ Reihenfolge egal
- $E \subseteq V^{(2)} \to$ Kante ist Teilmenge von 2 Knoten

gerichtete Graphen

- Kante ist ein geordnetes Paar von 2 (verschiedenen) Knoten
- $e = (x, y)$ entspricht $x \to y$, (y, x) entspricht $y \to x$
- $E \subseteq V \times V$
- gerichtete Kante besteht aus head (in Pfeilrichtung) und tail

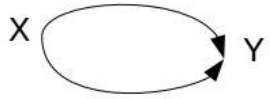
Funktionen gerichteter Graphen:
$h : E \to V : e \mapsto head(e)$
$t : E \to V : e \mapsto tail(e)$

Graphen in denen Kanten zwei verschiedenen Endpunkte haben **UND** zu jeden Paar von Kanten höchstens eine Kante gehört hießen EINFACH oder SIMPLE im gerichteten Fall:



trotzdem einfacher Graph!

erst:



ist Multigraph

Loops:

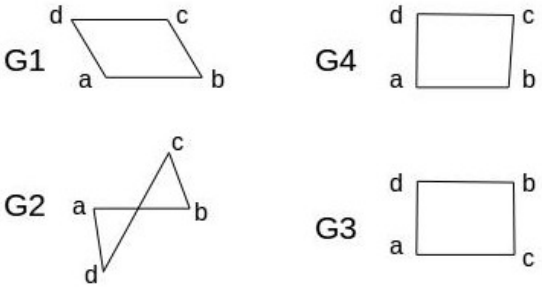
Abbildung 1: links: gerichtet; rechts: ungerichtet

⇒ einfacher Graph mit Loops

Durch Unterteilung der Kanten in Multigraphen kann eine Transformation in Graphen erzeugt werden:

- ungerichtet: zweifache Unterteilung mittels zweier Knoten
- gerichtet: einfache Unterteilung mittels Knoten

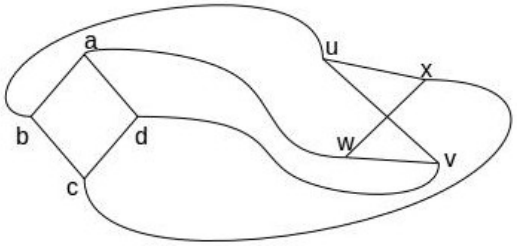
1.2 Gleichheit von Graphen



als labeled graphs: G_1=G_2=G_4 ≠G_3

⇒ 2 Graphen G_1=(V_1, E_1) und G_2=(V_2, E_2) sind isomorph wenn es einen bijektive Abbildung[1] $\pi : V_1 \to V_2$ gibt, sodass $\{x, y\} \in E_1 \Leftrightarrow \{\pi(x), \pi(y)\} \in E_1$

[1] `https://de.wikipedia.org/wiki/Bijektive_Funktion`



bijektive Abbildung: jedes Element von 1. wird zu genau einem Element von 2. zugeordnet

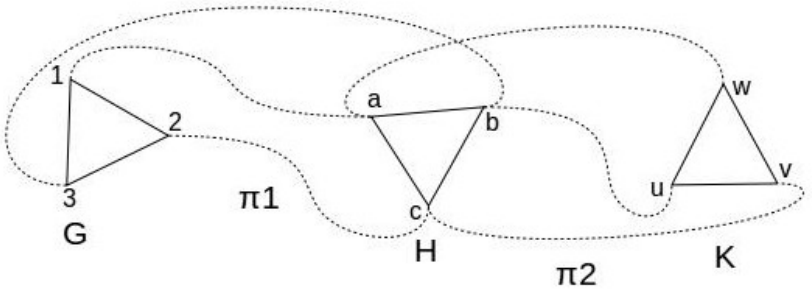
$\pi(a) = w, \pi(b) = u, \pi(c) = x, \pi(d) = v$
$\rightarrow$ hier ergibt bijektive Abbildung keinen Isomorpismus, da Bild(d) und Bild(c) Kante haben, jedoch v und x keine Kante haben

Durch folgende bijektive Abbildung wird aber Isomorphie erreicht:
$\pi(a) = w, \pi(b) = x, \pi(c) = u, \pi(d) = v$

Bezogen auf die Labels kann es mehrere mögliche Isomorphien geben.

Schreibweise: $G \simeq H$ (G ist isomorph zu H) mit $G \rightarrow^{\pi} H, G \leftarrow^{-\pi} H$ sodass π isomorph ist

Reflexivität: Ein Graph ist zu sich selbst immer isomorph: $G \simeq G$
Symmetrie: $G \simeq H \Leftrightarrow H \simeq G$ Transitivität: $G \simeq H, H \simeq K \Rightarrow G \simeq K$



$\simeq$ ist eine Äquivalenzrelation $\rightarrow$ Isomorphie teilt Graphen in Klassen ein (Isomorphieklassen)

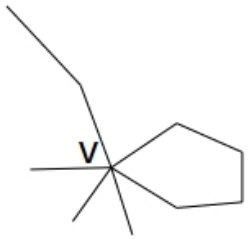
Nebenbemerkung: Labeled Graphen?
Zusätzliche Bedingung benötigt: $\lambda(\pi(x)) = \lambda(x) \rightarrow$ Labels müssen erhalten bleiben!

Testen auf Gleichheit
Gegeben: $G_1=(V_1, E_1)$, $G_2=(V_2, E_2)$
Frage: Sind die Graphen isomorph?

Grundbedingungen:

1. $|V_1| = |V_2| \rightarrow$ gleiche Anzahl von Knoten
2. $|E_1| = |E_2| \rightarrow$ gleiche Anzahl von Kanten

1.3 Eigenschaften von Graphen



Nachbarknoten von v: $N(v) := \{y \in V | \{v, y\} \in E\}$
$deg(v) := |N(v)|$
$\delta(G) := \min_{v \in V} deg(v)$
$\Delta(G) := \max_{v \in V} deg(v)$

Def: Ein Graph heißt **REGULÄR** wenn $\Delta(G) = \delta(G)$
(wenn alle Knoten gleichen Grad haben)

Gradfolge von G:

$\mathcal{F} = (n_0, n_1, n_2, \ldots, n_{|V|-1})$ mit $n_k := |\{x \in V | deg(x) = k\}|$

$\delta(G) \geq 0$
$\Delta(G) \leq |V| - 1$

Beispiel:

0 1 2 3 4
F= (0 4 0 0 1)

0 1 2 3
F= (0 4 0 0)

bei Isomorphie: $\mathcal{F}_1 = \mathcal{F}_2 \rightarrow$ Isomorphismus π erhält Grad der Knoten!

1.4 Graph-Invarianten

Eigenschaften, die unter Isomorphie erhalten bleiben
$\mathcal{G}$... Menge aller Graphen
F... ist ein Graph invariant wenn

$$F : \mathcal{G} \to X \tag{1}$$

die Eigenschaft hat, dass

$$G \simeq H \Rightarrow F(G) = F(H) \tag{2}$$

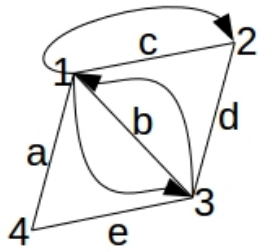
Invarianten bis jetzt: $|V|, |E|$, Gradfolge $\mathcal{F}$

Wenn F(G) $\neq$ F(H) für irgendeine Grapheninvariante $\Rightarrow$ G $\neg \simeq$ H

1.5 Pfade und Zusammenhänge

Kantenzug: Folge von Kanten in G"
$x_o, e_1, x_1, e_2, x_2, \ldots, e_l, x_l$ sodass $e_i := \{x_{i-1}, x_i\}$

Beispiel:



Weg: Kantenzug sodass $e_i \neq e_j$ für $i \neq j$ (keine Kante doppelt verwenden)

Pfad: Kantenzug sodass $x_i \neq x_j$ für $(i, j) \neq (0, l)$ mit 0=Startknoten und l=Endknoten des Pfades (keinen Knoten mehrfach bis auf x_0, x_l)

- offen: $x_o \neq x_e$ 
- geschlossen: $x_0 = x_e$ (nur hier 1 Knoten doppelt benutzt!) 

Definition: G ist zusammenhängend wenn es zwischen je zwei Knoten x,y $\in$ V einen Kantenzug gibt

Frage:

1. Ist Zusammenhang eine Grapheninvariante?
2. Kann man in der Definition Kantenzug durch Weg, Pfad oder Kreis ersetzt?

2 Vorlesung 21.10.2016

3 Vorlesung 28.10.2016

4 Vorlesung 11.11.2016

5 Vorlesung 18.11.2016

6 Vorlesung 25.11.2016

7 Vorlesung 02.12.2016

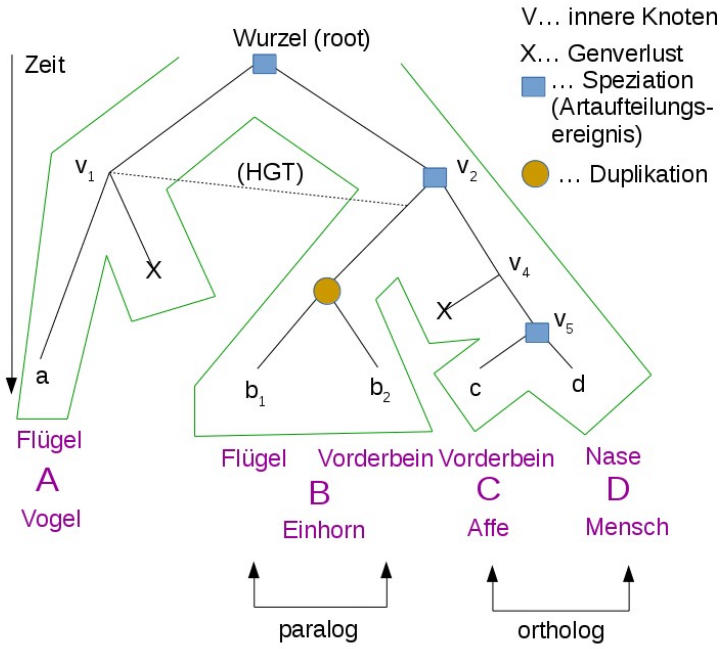
7.1 Cographen & Cotrees

Phylogenetik

- Erforschung von Abstammung
- Rekonstruktion von phylogenetischen Bäumen („Stammbäume“)
- Speziesbäume/Genbäume

Ergebnisse:

- Genverlust (loss)
- Aufspaltung zu einer neuen Spezies (speciation)
- Duplikation von Genen (ducplication)
- horizontaler Gentransfer (HGT)



Def.: Baum (tree)
Ein Baum T=(V,E) ist ein zusammenhängender Graph, der keine Kreise enthält (azyklisch).

Def.: Zusammenhang
Ein Graph G=(V,E) ist zusammenhängend, wenn es zwischen jedem möglichen Paar von Knoten einen Pfad gibt.

Theorem:
T=(V, E) ist ein Baum $\Leftrightarrow \exists!$ Pfad zwischen zwei zufällig gewählten Knoten existiert. ($\Leftrightarrow \ldots$ aus dem folgt; $\exists! \ldots$ genau einem)

Beweis:
$\Rightarrow$: Da T zusammenhängend ist, gibt es einen Pfad zwischen v, u $\in$ V(T), $\forall$ v, u $\in$ V(T). Angenommen es gäbe noch einen 2. Pfad, dann gibt es einen Kreis; Widerspruch zur Definition.
$\Leftarrow$: Wenn genau ein Pfad existiert, ist T zusammenhängend, Also gibt es auch keine Kreise; T ist ein azyklischer Graph = Baum.

Def.: Distanz
Die Distanz d(u,v) zwischen zwei Knoten u, v $\in$ V ist gleich der Anzahl der Kanten im kürzesten Pfad zwischen u und v.

Def.: Lowest Common Ancester (lca)
Seien x,y $\in$ V(T) Blätter im Baum T mit Wurzel r. Sei $P_x = \{x, x_1, x_2, \ldots, r\}$ der Pfad von x nach r und $P_y = \{y, y_1, y_2, \ldots, r\}$ der Pfad von y nach r. Dann $lca(x, y) = min(d(d, v_i), d(y, v_i))$ mit $v_i \in (P_x \cap P_y)$
$v_i \ldots$ mehrere v's (kann auch r sein)
r ... root (Wurzel)

- $P_{b_2r} = \{b_2, v_3, v_2, r\}$
- $P_{dr} = \{d, v_5, v_4, v_2, r\}$
- $P_{b_2r} \cap P_{dr} = \{v_2, r\}$
  - $d(b_2, v_2) = 2$
  - $d(b_2, r) = 3$

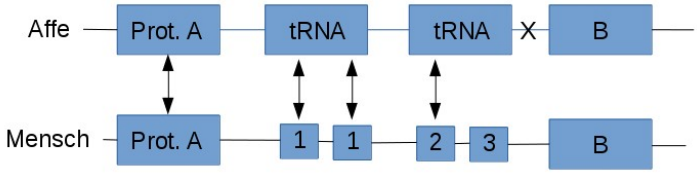
Def.:

- Homologie: 2 Gene sind homolog, wenn sie die selben Vorfahren haben
- Orthologie: 2 Gene sind ortholog, wenn ihr lca eine Speziation (Artaufteilungsereignis) ist
- Paralogie: 2 Gene sind paralog, wenn ihr lca eine Duplikation ist

Def.: Θ-Relation (Orthologie-Relation)
Seien x,y $\in$ H, H = Menge von Genen
(x,y) $\in \Theta \Leftrightarrow$ lca(x,y) ist eine Speziation.
Diese Relation ist reflexiv (rückbezüglich), symmetrisch, aber nicht transitiv (mit sich ziehend).
Bestimmung von Orthologie:

- Sequenzähnlichkeit

- Syntenie („Gemeinsamkeiten in der Reihenfolge von Genen oder Gensegmenten auf verschiedenen chromosomalen Abschnitten. [...] ist ein Maß für die genetische Verwandtschaft der beiden Arten.“[Wikipedia])



z.B. Tool: ProteinOrtho

Def.: $\sim$-Relation (fast-Orthologie)
(x,y) $\in \sim$, wenn x,y als ortholog eingestuft werden.
Ziel: Korrigieren $\sim$ sodass wir Θ erhalten. Dazu stellen wir $\sim$ und Θ als Graphen dar.

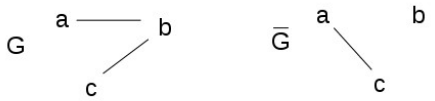
$G_\Theta = (V_\Theta, E_\Theta)$ $\qquad$ $G_\sim = (V_\sim, E_\sim)$

$V_\Theta = V_\sim = Gene$

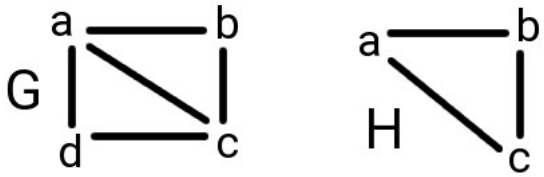
$E_\Theta = \{(x,y) \in \binom{V}{2} \mid x\Theta y\}$ $\qquad$ $E_\sim = \{(x,y) \in \binom{V}{2} \mid x \sim y, y \sim x\}$

$\binom{V}{2}$ ... alle möglichen Kombinationen von zwei Knoten

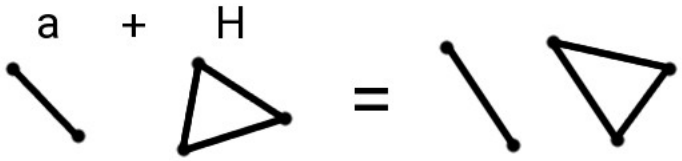
Def.: Komplementgraph (complement)
Sei G=(V,E) ein Graph. Das Komplement $\overline{G}$ von G ist der Graph $\overline{G} = (V, \overline{E})$ mit $\overline{E} = \{(u,v) \in \binom{V}{2} \mid (u,v) \notin E\}$



Def.: Teilgraph
Sei G=(V,E) ein Graph und H $\subseteq$ G. H ist Teilgraph von G, wenn V(H) $\subseteq$ V(G), E(H) $\subseteq$ E(G). Ein induzierter Teilgraph ist ein Teilgraph H von G bei dem alle Knoten die in G benachbart sind, auch in H benachbart sein müssen.
(v,u) $\in$ E(G) $\wedge$ u,v $\in$ V(H) $\Leftrightarrow$ (v,u) $\in$ E(H)
$\wedge$ ... Konjunktion



Def.: disjunkte Vereinigungen
Graphen G,H: G+H ist ein Graph mit V(G) $\cup$ V(H) und E(G) $\cup$ E(H).

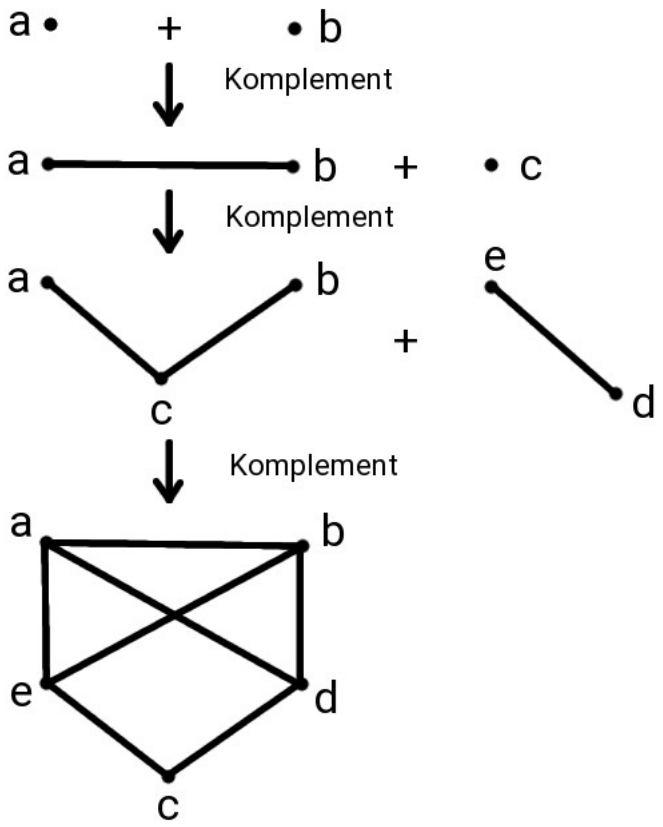


Def.: Cograph

1. K_1 ist ein Cograph 

2. G ist ein Cograph $\Leftrightarrow$ $\overline{G}$ ist ein Cograph
3. G, H sind Cographen $\Leftrightarrow$ G+H ist ein Cograph

Erstellung von Cographen:

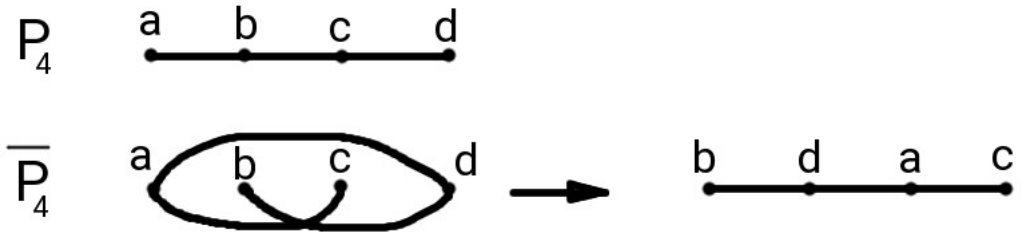


Eigenschaften von Cographen:
Sei G=(V,E) ein Cograph und H $\subseteq$ G,H Cograph

i) G enthält *keine* induzierten P_4's

ii) H ist zusammenhängend $\Leftrightarrow$ $\overline{H}$ ist nicht zusammenhängend

iii) G kann aus einzelnen Knoten (K_1) zusammengesetzt werden

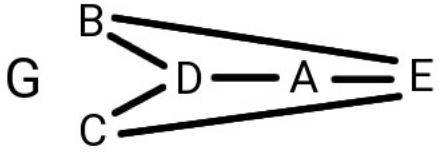


$\Rightarrow P_4 = \overline{P_4}$
Ein Cograph muss jedoch ein P_4-freier Graph sein.
Cograph = P_4-free graphs = complement reducible graphs

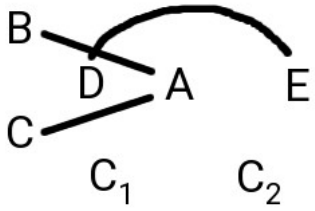
Test ob G=(V,E) ein Cograph ist:
Input: Graph G
i $\leq$ Cograph (G) {
if (| $V(G)$ |< 4) {return true;}
c = {Zusammenhangskomponenten von G}
if(| c | $= 1$) {c'={Komponenten von $\overline{G}$}}
if (| c' |$= 1$) {return false;}
else {
foreach (c $\in$ C)
{isCograph (c) }
}
}
Bei isCograph: je nachdem ob c oder c' rausgekommen ist, muss c oder c' geprüft werden.

Theorem:
$\sim = \Theta \Leftrightarrow G_\Theta = G_\sim$ und G_Θ ist ein Cograph
Damit können wir testen, ob $G_\sim$ ein Orthologiegraph ist.
Was passiert wenn $\sim \neq \Theta$ bzw. $G_\sim$ kein Graph?
$\Rightarrow$aktuelle Forschung $\Rightarrow$Es gibt Lösungen $G_\sim$ zu editieren mit optimalen Kriterien, sodass der editierte $G_\sim$ ein Cograph ist. Z.B. ILP (integer linear program), Cograph-editing. Alle Algorithmen, die exakte Möglichkeiten liefern, brauchen sehr lange und sind in der Praxis nicht nutzbar.
Weitere Literatur: Marc Hellmuth

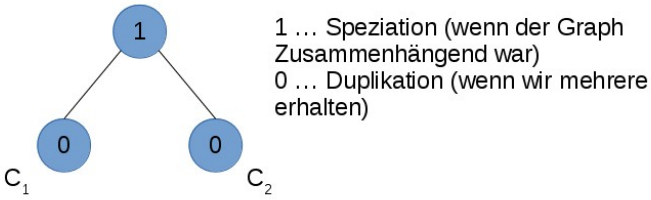
Theorem:
Für jeden Cographen gibt es einen eindeutigen Cotree (Cobaum)



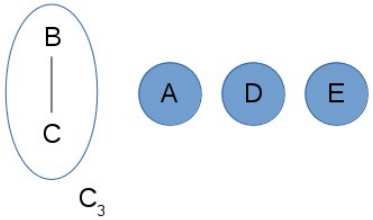
1. Schritt: Komplement

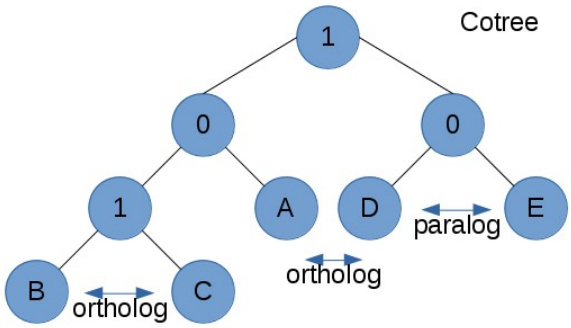


2. Schritt: umgekehrte disjunkte Vereinigung



3. Schritt: Komplement





8 Vorlesung 09.12.2016

9 Vorlesung 16.12.2016

Metrik:

1. $d_{uu} = 0$
2. $d_{uv} = 0 \Rightarrow u = v$
3. $d_{uv} = d_{vu}$
4. $d_{uv} + d_{vw} \geq d_{uw}$ (Dreiecksungleichung)

Pseudometrik: -,1,2,3
Metrik: 0,1,2,3
Distanzfunktion: 1,2

4-Punkte-Bedingung:
Eine Distanzfunktion d ist eine additive (Baum) Metrik wenn je vier Punkte so geordnet werden können, daß:
$d_{xy} + d_{uv} \leq d_{xu} + d_{yv} = d_{xv} + d_{yu} \Leftrightarrow \forall$x,y,u,v gilt:
$d_{xy} + d_{uv} \leq max\{d_{xu} + d_{yv}, d_{xv} + d_{yu}\}$

Isolationsindex:
$l(e) = \alpha(A|B) = max(0, \min\limits_{\substack{x,y \in A \\ u,v \in B}} \frac{1}{2}[max\{d_{xu} + d_{yv}, d_{xv} + d_{yu}\} - (d_{xy} + d_{uv})])$
=Länge der Baumkante, die A,B trennt oder ≤ 0 wenn $A|B$ keine Teilbäume bestimmt.

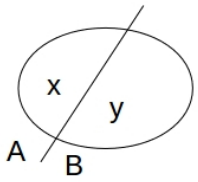
Wenn d eine additive Distanzfunktion:

- $\alpha(A|B) \geq 0$
- $A|B$ entspricht einer Kante im Baum $\Leftrightarrow \alpha(A|B) > 0$

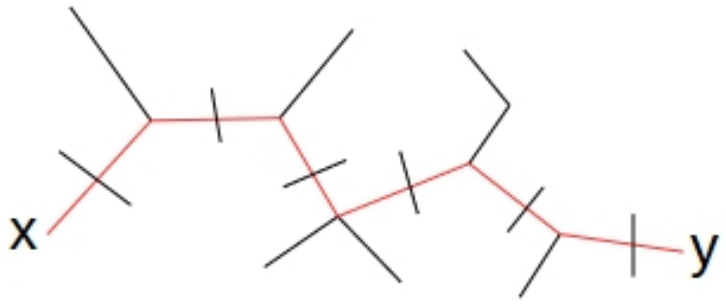
Splitpseudometrik:

$$\delta_{A|B}(x,y) = \begin{cases} 1 : x \in A, y \in B \\ 1 : x \in B, y \in A \\ 0 : x, y \in A \\ 0 : x, y \in B \end{cases} \tag{3}$$

x,y durch $A|B$ getrennt $\Leftrightarrow \delta_{A|B}(x,y) = 1$

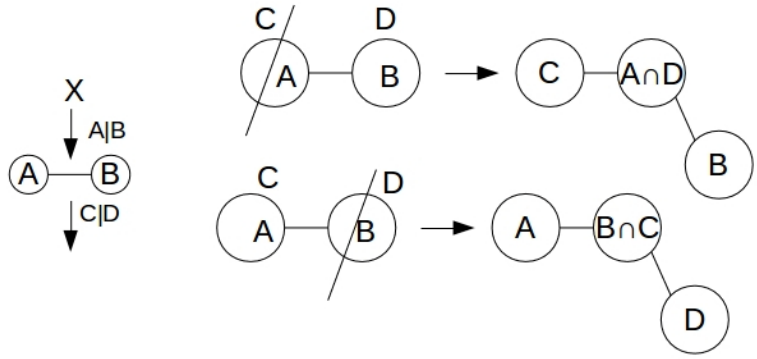


$$d_T(x,y) = \sum_{(A|B)\in\Sigma(T)} \alpha(A|B)\cdot\delta_{A|B}(x,y)$$



Genau die splits entlang des Pfades von x und y trennen x,y

Splits $\Sigma(T) \to$ Baum
wir wissen $\Sigma(T)$ ist kompatible
$A|B, C|D \in \Sigma(T)$ dann mindestens einer der vier Durchschnitte:
$A\cap C, A\cap D, B\cap C, B\cap D$ leer



jeder split-Teil <u>GENAU</u> eine der Mengen

Frage: Wie können Isolationsindizes, schnell und und alle Möglichkeiten durchzuprobieren, erzeugt werden?
Lösung: effiziente Berechnung von $\alpha(A|B) > 0$
Idee: erweitere X schrittweise
$|A|, |B| = 1$
$X' \leftarrow X \cup \{w\}$
$A \cup B = X$
in X' :
- $X|\{w\}$
- $A \cup \{w\}|B$
- $B \cup \{w\}|A$

$\beta_{xy|uv} := \frac{1}{2} max\{d_{xu} + d_{yv}, d_{xv} + d_{yu}\} - (d_{xy} + d_{uv})$
erster Fall:
$\alpha(\{x\}|X) = \min\limits_{u,v \in X} \beta_{ww|uv} = \min\limits_{u,v \in X} \frac{1}{2}(d_{wu} + d_{wv} - d_{uv})$
zweiter Fall:
$\alpha(A|B) = \min\limits_{\substack{x,y \in A \\ u,v \in B}} \beta_{xy|uv}$

$\alpha(A \cup \{w\}|B) = \min\{\min\limits_{\substack{x,y \in A \\ u,v \in B}} \beta_{xy|uv}, \min\limits_{\substack{y \in A \\ u,v \in B}} \beta_{yw|uv}, \min\limits_{\substack{x \in A \\ u,v \in B}} \beta_{xw|uv}\}$

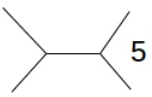
$\Rightarrow \alpha(A \cup \{w\}|B) \leq \alpha(A|B)$
Also: wenn $\alpha(A|B) \leq 0 \Rightarrow \alpha(A \cup \{w\}|B)$ auch ≤ 0
$\Rightarrow$ nur Splits auf X mit $\alpha(A|B) > 0$ müssen erwartet werden
Wenn d additiv $\Rightarrow$ Baum $\Rightarrow$ splits$\Sigma(T)$ kompatibel $\Rightarrow$ es gibt nicht mehr als $2|X|$ splits
$\Rightarrow$Die Isolationsindizes aller Splits mit $\alpha(A|B) > 0$ können in $\mathcal{O}(|x|^5)$ berechnet werden:
$|x|$ Erweiterungsschritte für $\mathcal{O}(|x|)$ splits mit Aufwand $\mathcal{O}(|x|^3)$

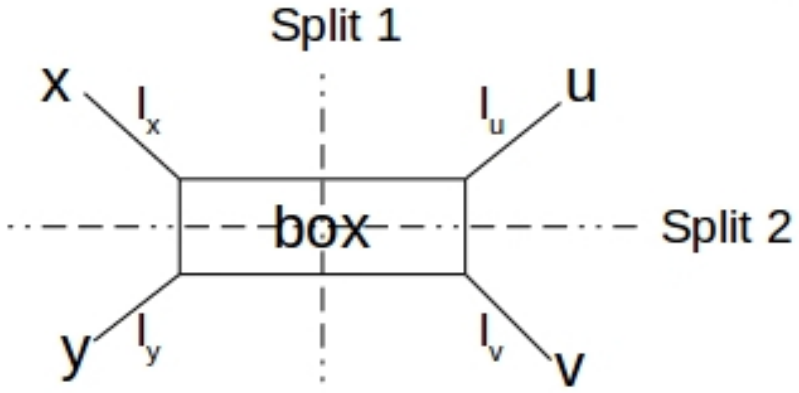
Theorem:[Bandelt,Dress]
Sei d eine Peusometrik auf X. Dann gibt es eine Pseudometrik d^0 auf X sodaß
$d(x,y) = \sum\limits_{A|B} \underbrace{\alpha(A|B)}_{*} \cdot \delta_{A|B}(x,y) + d^0(x,y)$
* $\alpha(A|B) = 0$ wenn $\min\limits_{\substack{x,y \in A \\ u,v \in B}} \beta_{xy|uv} < 0$

außerdem gilt: $\Sigma(d) = \{(A|B)\}$
$alpha(A|B) > 0$ hat höchstens $\mathcal{O}(|x|^2)$ Elemente
alle $\alpha(A|B) > 0$ können in $\mathcal{O}(|x|^6)$ Elemente berechnet werden.

- d additiv $\Rightarrow d^0 = 0$
- d^0 heißt split-primer
- d heißt total zerlegbar wenn $d^0 = 0$

allgemeine Pseudometrik auf 4 Punkten
Anzahl unabhängigen Distanzen: 6

Baum mit 4 Blättern: 5 



$d_{xu} + d_{xy} - d_{duy}$
$(l_x + a + l_u) + (l_x + b + l_y) - l_u - a - b - l_y = 2l_x$
$l_x = \frac{1}{2}[\underbrace{d_{xu} + d_{xy} - d_{uy}}_{\geq 0 (Dreiecksungleichung)}]$

Split 1:
$d_{xv} + d_{yu} - (d_{xy} + d_{uv}) =$
$l_x + a + b + l_v$
$+l_y + a + b + l_u$
$-l_x - b - l_y$
$-l_u - b - l_v = 2a$

Split 2:
$d_{xu} + d_{yv} - (d_{xy} + d_{uv}) =$
$l_x + a + l_u$
$+l_y + a + l_v$
$-l_x - b - l_y$
$-l_y - b - l_u = 2(a - b) \leq 2a$

$\alpha(\{xy\}|\{uv\}) = a$
$\alpha(\{xu\}|\{yv\}) = b$
Baum $\Rightarrow$ b=0

Messung der Baumartigkeit:

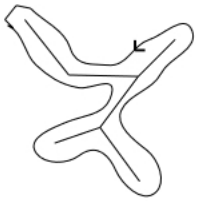
$$B := \frac{1}{\binom{n}{4}} \sum_{\substack{i<j<k<l \\ i,j,k,l \in X}} \frac{b_{ijkl}}{a_{ijkl} + b_{ijkl}}$$

Mittelwerte von in der Box
B $\approx$ Baumartig
B $\approx \frac{1}{2}$ völlig verrauscht, netzwerk-artig

Travelling sales person problem (TSP)
geschlossene Tour Voraussetzung
$|X| > 1$ (Anzahl der Städte größer 1)
Metrik d auf X gegeben
Tour: Permutation von X:π
$L(\pi) = \sum_{i=1}^{|X|} d_{\pi(i-1)\pi(i)}$ (lesen als indices modulo $|X|$)

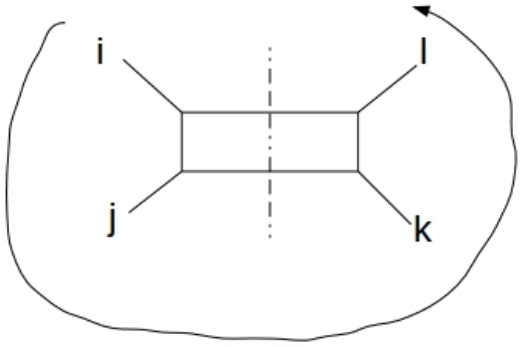
Definition Mastertour:
Einschränkung von π auf $X' \subseteq X$ löst das TSP auf X

Wenn d eine additive Metrik (Baum) ist dann existiert eine Mastertour (optimale Lösung) die genau ein Mal um den Baum herum führt.



Eine Metrik hat die KALMANSON-Eigenschaft, wenn man X so ordnen kann, daß
$d_{ij} + d_{kl} \leq d_{ik} + d_{jl} \forall i < j < k < l$
und
$d_{il} + d_{jk} \leq d_{ik} + d_{jl} \forall i < j < k < l$

$\rightarrow$ für jedes Quadrupel tauchen höchstens die Splits ij|kl, il|jk auf
d ist Kalmanson $\Leftrightarrow$ das TSP mit Distanz d einen Mastertour hat



Wenn d Kalmanson ist (zirkulär zerlegbar) $\Rightarrow$ d splitzerlegbar (planar darstellbar)
$\not\Leftarrow$ (Umkehr falsch)

$$d = \underbrace{\sum_{A|B} \alpha(A|B) \cdot \delta_{A|B}}_{fast\ immer\ Kalmanson} + \underbrace{\delta^0}_{\substack{Rauschen \\ (split\ Primaeranteil)}}$$

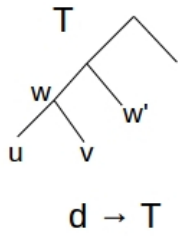
Anteil der Distanz ohne phylogenetische Information:
$$\frac{\sum_{x \neq y} \delta^0(x,y)}{\sum_{x \neq y} \delta(x,y)}$$
(Maß für die Größe des Rauschens → keine phylogenetische Information)

10 Vorlesung 21.12.2016

10.1 neighbor joining

geg: Distanzmatrix (d) auf Menge X von Taxa → Baum (ungewurzelt)
Iteration:

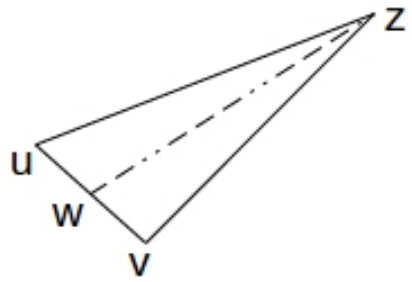
1. suche $argmin_{x,y}\tilde{d}_{xy} = \{u, v\}$
2. ersetze $\{u, v\} \rightarrow$ no (neuer Knoten)
3. brechne d_{wz} für $z \neq u, v \rightarrow$ Schritt 1



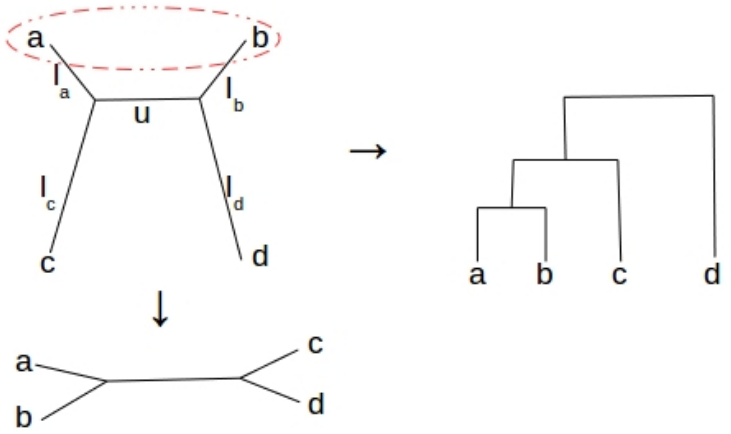
d → T

$\tilde{d}$ Transformation von d
F:$d \mapsto \tilde{d}$
$d_{wz} = \phi(d_{uz}, d_{vz}, d_{uz})$

Ein Baumrekonstruktionsalgorithmus $\mathcal{A} : d \mapsto T$ ist konsistent wenn:
Falls d ein additive Baum-Metrik mit Baum $\hat{T}$ ist, dann ist $\mathcal{A}(d) = \hat{T}$
Beispiel:
$\tilde{d} = d$
$d_{wz} = \frac{1}{2} \cdot d_{uz} + \frac{1}{2} \cdot d_{vz}$ (WPGMA)
$d_{wz} = \frac{|u|}{|u|+|v|} \cdot d_{uz} + \frac{|u|}{|u|+|v|} \cdot d_{vz}$ (UPGMA)



Ist der zugehörige Alogrithmus konsistent?
Gegenbeispiel:
$l_a, l_b, q << l_c, l_d \Rightarrow argmin_{x,y}\tilde{d}_{xy} = \{a, b\}$



(Problem: LBA - long branch attraction)

Lösung:
Abstand eines Punktes von allen anderen Punkten berechnen:$r(u) = \sum_{x \neq u} d(x,u)$

$\tilde{d}_{xy} = d_{xy} - \alpha \cdot r(x) - \beta \cdot r(y)$
$\alpha = \beta = \frac{1}{n-2}$ mit n=Zahl der Taxa

Lemma: Wenn d eine additive Baum-Metrik ist und $\{u,v\} = argmin_{x,y}\tilde{d}_{xy} = \{u,v\} \Rightarrow$ u,v wird cherry genannt.

$\{u,v\} \mapsto w$ (u und v mittels Vaterknoten w vereinigen)
$d(u,w) = \frac{1}{2} \cdot d(u,v) + \frac{1}{2} \cdot \frac{1}{n-2}[r(u) - r(v)]$
durch Symmetrie: $d(v,w) = \frac{1}{2} \cdot d(u,v) + \frac{1}{2} \cdot \frac{1}{n-2}[r(v) - r(u)]$
$d(w,z) = \frac{1}{2} \cdot [d(u,z) - d(u,w)] + \frac{1}{2} \cdot [d(v,z) - d(u,w)]$
$= \frac{1}{2} \cdot [d(u,z) + d(v,z)] - d(u,w)$

[Paper: Gascuel + Steel, Mol Biol Evol, 23 Seite 1997-2000 (2006)[2]]

10.2 Neighbor Net

Kalmanson Metrik
$\rightarrow$ zirkuläre Ordnung der Taxa

- Auswahl der Nachbarn
- Update der Distanzen

Initialisierung: Jeder Punkt ist in einem separaten Cluster C_i, mit Punkten x,y,...
$d(C_i, C_j) := \frac{1}{|C_i||C_j|} \sum_{\substack{x \in C_i \\ y \in C_j}} d(x,y)$

[2] http://mbe.oxfordjournals.org/content/23/11/1997.long

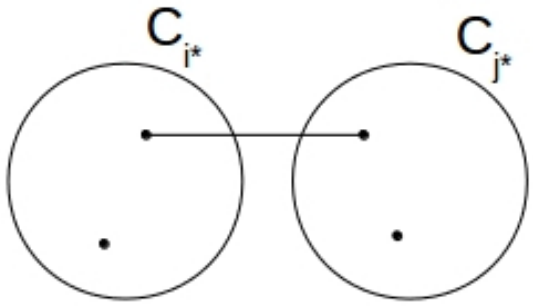
$$Q(C_i, C_j) := (m-2) \cdot d(C_i, C_j) - \underbrace{\sum_{k \neq i} d(C_i, C_k)}_{(m-2) \cdot r(C_i)} - \underbrace{\sum_{k \neq j} d(C_j, C_k)}_{(m-2) \cdot r(C_j)}$$

mit m= Anzahl Cluster
(NI-Formale für Cluster)
Bestimme i*,j* = $argmin_{i,j} Q(C_i, C_j)$
C_i, C_j enthält jeweils entweder 1 oder 2 Knoten
für Punkte in $x_i \in C_i$* und $x_j \in C_j$*

$$\hat{Q}(x_i, x_j) = (\hat{m} - 2) \cdot d(x_i, x_j) - \sum_k d(x_i, C_k) - \sum_k d(x_j, C_k)$$

$$\hat{m} = m - \underbrace{2}_{i*, j*} + |C_{i*}| + |C_{j*}|$$

Erkläre x*,y* mit $x* \in C_{i*}, y* \in C_{j*}$ (mit jedem Schritt eine Kante mehr)



y hat 2 (verschiedene) Nachbarn x,z
a≠x,y,z,u,v

$$d(u, a) = \alpha \cdot d(x, a) + \beta \cdot d(y, a)$$
$$d(v, a) = \beta \cdot d(y, a) + \gamma \cdot d(z, a)$$
$$d(u, v) = \alpha \cdot d(x, y) + \beta \cdot d(x, z) + \gamma \cdot d(y, z)$$

mit $\alpha + \beta + \gamma = 1;\ \alpha, \beta, \gamma \geq 0;\ \alpha = \beta = \gamma = \frac{1}{3}$

Theorem: Wenn d Kalmanson Eigenschaften hat
⇒ Neighbor Net erzeugt die zugehörige zirkuläre Ordnung und identifiziert damit alle Splits mit nichtnegativen $\beta_{A|B}$

letzter Schritt im Neighbor Net Algorithmus:

$$\min_{\substack{\beta_{A|B} \forall A|B \\ zirkuläre\ Splits}} \left(\sum_{x,y} (d(x, y) - \sum_{splits} \beta_{A|B} \cdot \delta_{A|B}(x, y))^2\right) \text{ mit } \beta_{A|B} \geq 0$$