離散数学 第 14 回
系統樹復元の離散数学

岡本 吉央
okamotoy@uec.ac.jp

電気通信大学

2013 年 7 月 30 日

最終更新：2013 年 7 月 29 日　17:54

目次

生物の多様性

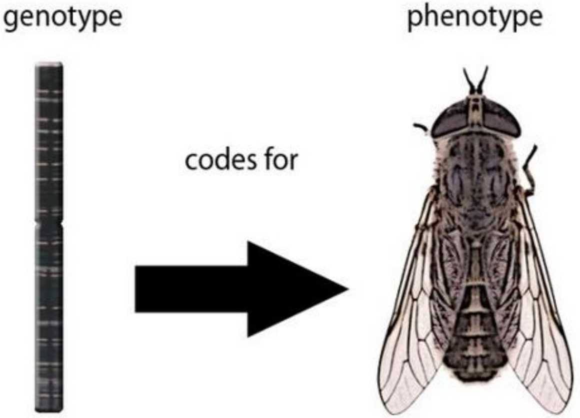
http://en.wikipedia.org/wiki/File:Fungi_of_Saskatchewan.JPG

http://mappery.com/Singapore-Zoo-Map-2

形態と形質

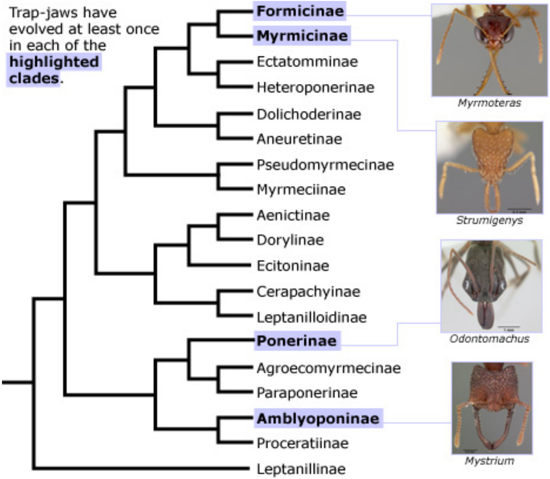
http://tolweb.org/treehouses/?treehouse_id=2482

遺伝子型と表現型



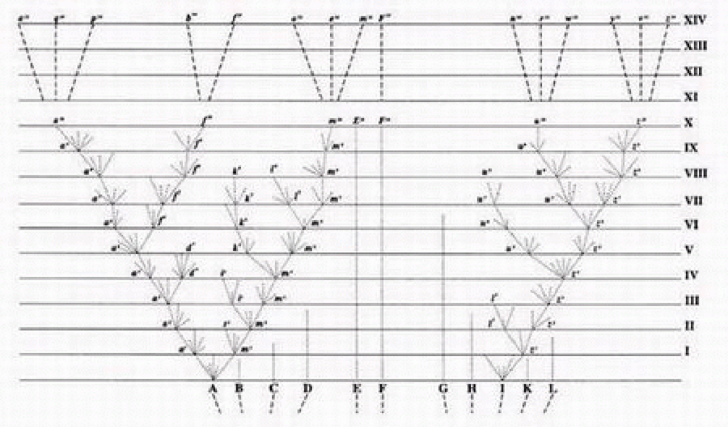
http://www.biotechlearn.org.nz/focus_stories/evolved_enzymes/images/fly_genotype_and_phenotype

系統樹



http://evolution.berkeley.edu/evolibrary/news/061001_trapjaw

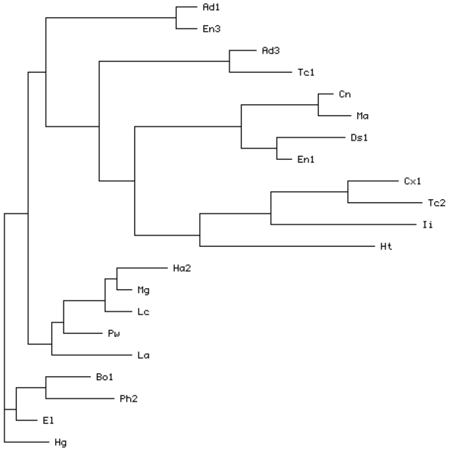
系統樹：ダーウィンの著書から



C. Darwin (1859) *On the Origin of Species by Means of Natural Selection*

系統樹：他分野での利用法

『カンタベリー物語』(Geoffrey Chaucer，14 世紀) の写本の系統樹



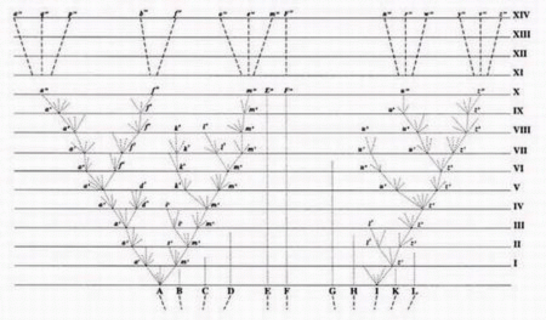
http://www.canterburytalesproject.org/pubs/desc2.html

目次

系統樹復元のお話

系統樹を復元するために使う手法の分類

1. **距離に基づく手法** distance-based approach
2. **形質に基づく手法** character-based approach



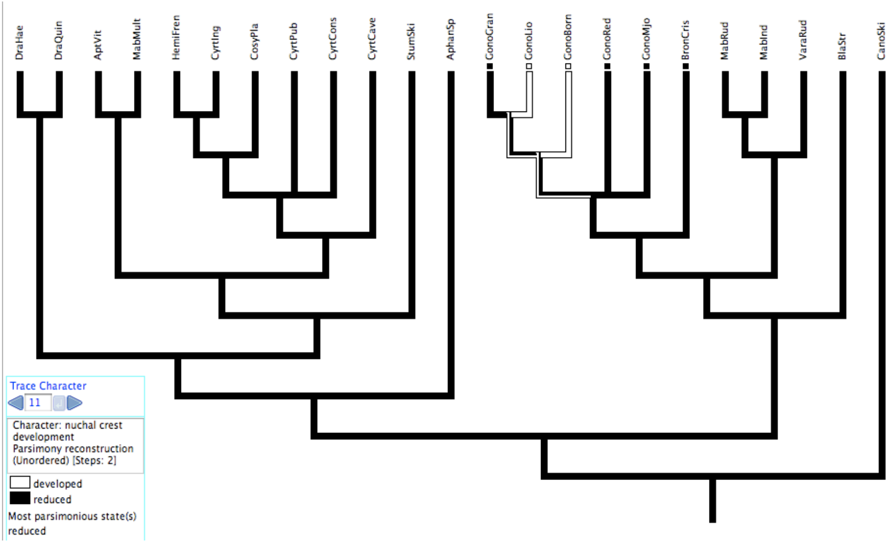
C. Darwin (1859) *On the Origin of Species by Means of Natural Selection*

形質に基づく手法 (1)：種形質行列

Taxon \ Character		1 tail compared to body	2 dorsal crest	3 dewlap	4 head	5 tympanum	6 leg length	7 nuchal crest	8 body type	9 leg type	10 dorsal crest developm	11 nuchal crest developm	12 size	13 eye shape	14 elongated ribs	15 dewlap size
1	BronCris	longer	absent	present	wide	visible	long	present	narrow	narrow	?	reduced	large	round	absent	small
2	DraQuin	same	absent	present	wide	hidden	long	absent	narrow	narrow	?	?	medium	two points	present	small
3	GonoBorn	longer	present	present	wide	visible	long	present	robust	robust	developed	developed	large	one point	absent	small
4	CyrtCons	same	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	small	round	absent	?
5	VaraRud	longer	absent	absent	wide	visible	long	absent	robust	robust	?	?	very large	round	absent	?
6	AphanSp	longer	absent	absent	wide	hidden	long	absent	robust	narrow	?	?	medium	one point	absent	?
7	DraHae	same	absent	absent	wide	hidden	long	absent	narrow	narrow	?	?	medium	two points	present	?
8	MabRud	shorter	absent	absent	narrow	visible	long	absent	robust	robust	?	?	medium	round	absent	?
9	BlaStr	longer	absent	absent	wide	visible	long	absent	narrow	narrow	?	?	large	round	absent	?
10	CyrtCave	same	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	medium	round	absent	?
11	CyrtIng	shorter	absent	absent	wide	visible	short	absent	narrow	narrow	?	?	very small	round	absent	?
12	GonoGran	longer	present	present	wide	visible	long	present	robust	robust	developed	reduced	very large	one point	absent	large
13	GonoLio	longer	present	present	wide	visible	long	present	robust	robust	reduced	developed	large	one point	absent	large
14	GonoMjo	longer	absent	present	wide	visible	long	present	robust	robust	?	reduced	large	one point	absent	small
15	GonoRed	longer	absent	present	wide	visible	long	present	robust	robust	?	reduced	large	one point	absent	small
16	CosyPla	shorter	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	small	round	absent	?
17	CyrtPub	same	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	small	round	absent	?
18	HemiFren	shorter	absent	absent	wide	hidden	short	absent	robust	narrow	?	?	very small	round	absent	?
19	AptVit	same	absent	absent	narrow	hidden	short	absent	robust	narrow	?	?	medium	round	absent	?
20	MabInd	longer	absent	absent	narrow	visible	short	absent	robust	robust	?	?	large	round	absent	?
21	MabMult	same	absent	absent	narrow	visible	short	absent	robust	narrow	?	?	medium	round	absent	?
22	CanoSki	longer	absent	absent	narrow	visible	long	absent	narrow	narrow	?	?	small	two points	absent	?
23	StumSki	same	absent	absent	narrow	hidden	long	absent	narrow	narrow	?	?	medium	one point	absent	?

http://phylodiversity.net/bb09/images/9/9a/Kristinachar.png

形質に基づく手法 (2)：復元された系統樹

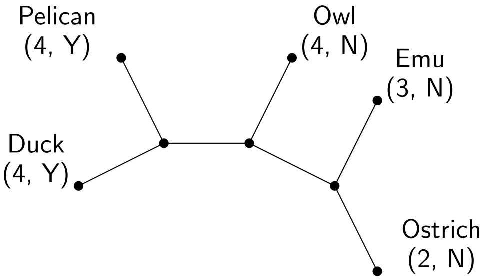


http://phylodiversity.net/bb09/images/f/f5/KristinaPhylo.PNG

Perfect phylogeny 問題

- **入力：種形質行列** M
- **出力：二分木で次を満たすもの**
  - **各葉が** 1 **つの種でラベル付けされている**
  - **各種はラベルとして一度だけ出現する**
  - **各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素**

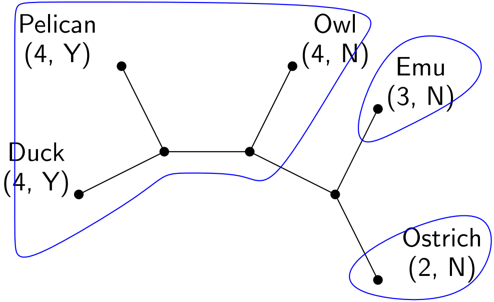
	# toes	Webbed feet
Ostrich	2	N
Emu	3	N
Pelican	4	Y
Duck	4	Y
Owl	4	N



Perfect phylogeny 問題

- **入力：種形質行列** M
- **出力：二分木で次を満たすもの**
  - **各葉が** 1 **つの種でラベル付けされている**
  - **各種はラベルとして一度だけ出現する**
  - **各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素**

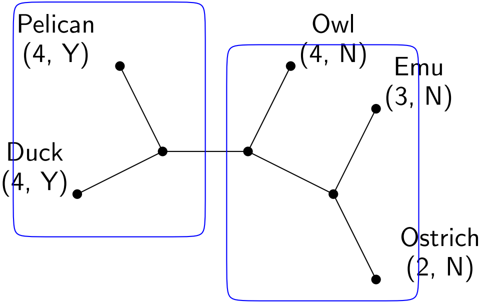
	# toes	Webbed feet
Ostrich	2	N
Emu	3	N
Pelican	4	Y
Duck	4	Y
Owl	4	N
	↑	



Perfect phylogeny 問題

- 入力：種形質行列 M
- 出力：二分木で次を満たすもの
  - 各葉が 1 つの種でラベル付けされている
  - 各種はラベルとして一度だけ出現する
  - 各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素

	# toes	Webbed feet
Ostrich	2	N
Emu	3	N
Pelican	4	Y
Duck	4	Y
Owl	4	N
		↑



問題のバリエーション

入力に対する仮定：形質は「二値」か「多値」か？

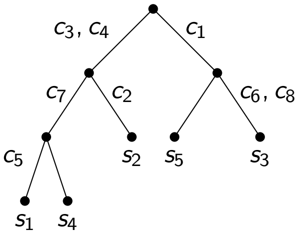
- **二値**：形質が 0 と 1 の値をとる (ように符号化される)
- 多値：形質のとりうる値に制限なし

出力に対する仮定：系統樹は「根付き」か「根無し」か？

- **根付き**：時間経過を考慮する
- 根無し：時間経過を考慮しない

$\rightsquigarrow$ Binary Directed Perfect Phylogeny 問題

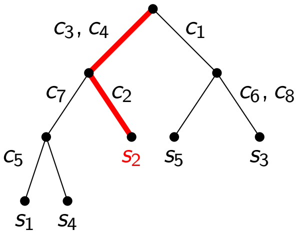
Binary Directed Perfect Phylogeny 問題



s：種，c：形質

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

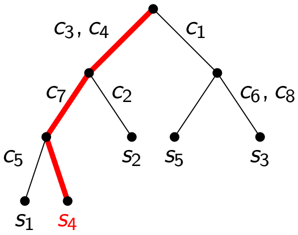
Binary Directed Perfect Phylogeny 問題



s：種，c：形質

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

Binary Directed Perfect Phylogeny 問題



s：種，c：形質

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

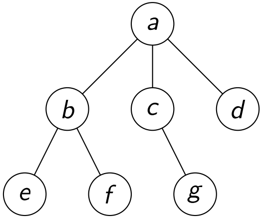
根つき木とは ?: 正確な定義

根つき木とは？

根つき木とは，ある有限集合 X 上の半順序集合 $T = (X, \preceq)$ で，以下を満たすもの

1. X の最大元が存在する
2. 任意の $x, y, z \in X$ に対して，
   y, z が x の上界であるならば，y, z は比較可能である

この半順序集合のハッセ図を根つき木と呼ぶこともある

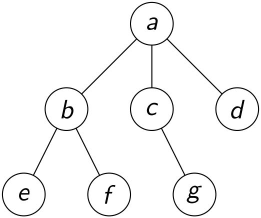


根つき木の根と葉

根つき木 $T = (X, \preceq)$

根つき木の根と葉とは？

- T の**根**とは X の最大元のこと
- T の**葉**とは X の極小元のこと

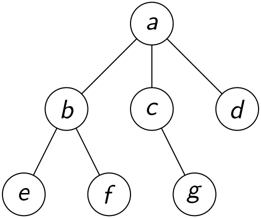


根つき木の性質

根つき木 $T = (X, \preceq)$

根つき木の性質

任意の $Y \subseteq X$ **に対して** Y **の最小上界が存在する**



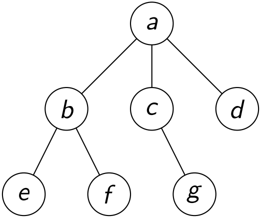
根つき木における子

根つき木 $T = (X, \preceq)$，$x \in X$

根つき木における子とは？

x の**子**とは，X の要素 $y \in X$ で以下を満たすもの

- $y \prec x$
- $y \prec z \prec x$ となる $z \in X$ が存在しない

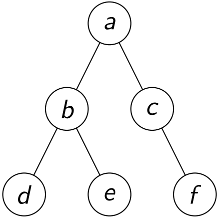


根つき二分木とは？

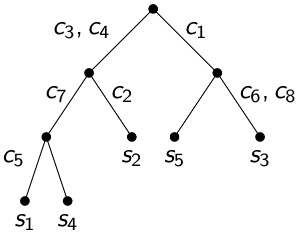
根つき木 $T = (X, \preceq)$

根つき木二分木とは？

T **が根つき二分木であるとは，**
葉ではない任意の $x \in X$ **の子の数がちょうど** 2 **であること**



Binary Directed Perfect Phylogeny 問題 (再掲)



s：種，c：形質

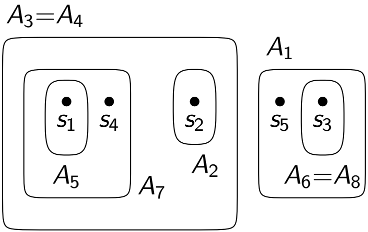
	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

目次

例 1：どのように系統樹を作るのか？

A_j = 形質 c_j を持つ種全体の集合
$(A_j = \{s_i \mid M_{i,j} = 1\})$

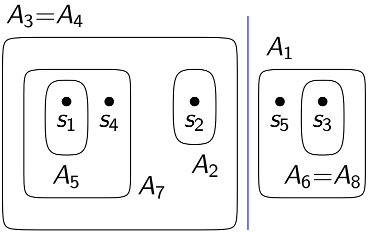
	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0



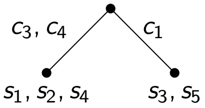
例 1：どのように系統樹を作るのか？

A_j = **形質** c_j **を持つ種全体の集合**
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

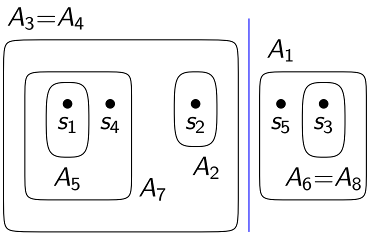


例 1：どのように系統樹を作るのか？

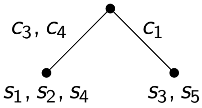


A_j = 形質 c_j を持つ種全体の集合
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

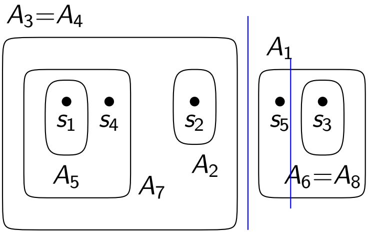


例 1：どのように系統樹を作るのか？

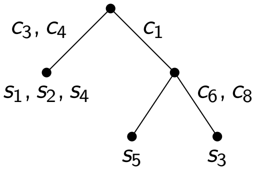


A_j = **形質** c_j **を持つ種全体の集合**
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

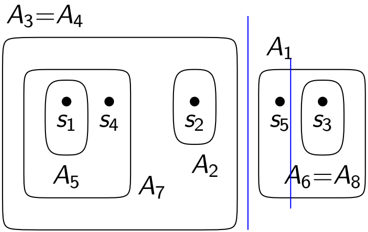


例 1：どのように系統樹を作るのか？

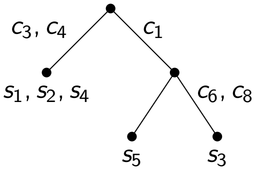


A_j = **形質** c_j **を持つ種全体の集合**
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

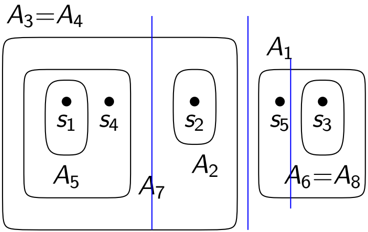


例 1：どのように系統樹を作るのか？

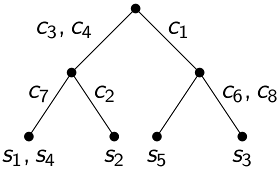


$A_j =$ **形質** c_j **を持つ種全体の集合**
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

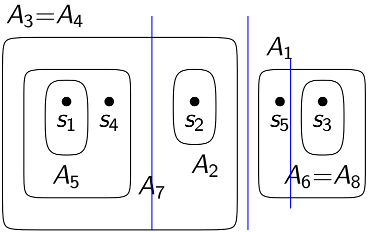


例 1：どのように系統樹を作るのか？

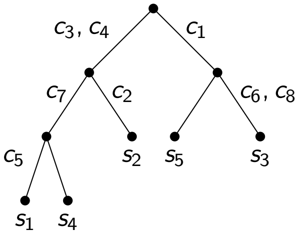


A_j = **形質** c_j **を持つ種全体の集合**
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

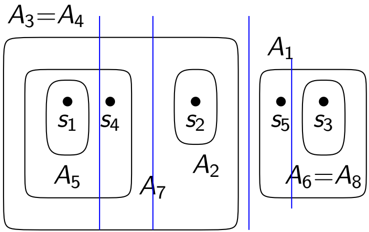


例 1：どのように系統樹を作るのか？



A_j = **形質 c_j を持つ種全体の集合**
$(A_j = \{s_i \mid M_{i,j} = 1\})$

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0



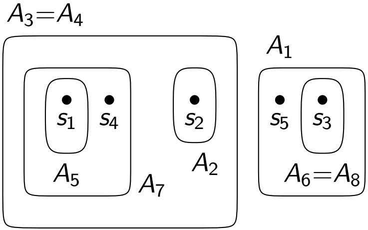
系統樹が作れるための必要十分条件

種 $s_1, \ldots, s_n$，形質 $c_1, \ldots, c_m$，種形質行列 $M \in \{0,1\}^{n\times m}$

系統樹が作れるための必要十分条件 (整合性条件)

M から系統樹を作ることができる $\Leftrightarrow$
集合 $A_1, \ldots, A_m$ の中の任意の 2 つ A_i と A_j が次のいずれかを満たす

$$A_i \subseteq A_j, \quad A_i \supseteq A_j, \quad A_i \cap A_j = \emptyset$$



- $A_1 \cap A_2 = \emptyset$
- $A_1 \cap A_3 = \emptyset$
- $A_1 \cap A_4 = \emptyset$
- $A_1 \cap A_5 = \emptyset$
- $A_1 \supseteq A_6$
- $A_1 \cap A_7 = \emptyset$
- $A_1 \supseteq A_8$
- $A_2 \subseteq A_3$
- $A_2 \subseteq A_4$
- $A_2 \cap A_5 = \emptyset$
- $A_2 \cap A_6 = \emptyset$
- $A_2 \cap A_7 = \emptyset$
- $A_2 \cap A_8 = \emptyset$
- $\cdots$

系統樹が作れるための必要十分条件：証明 (1)

種 $s_1, \ldots, s_n$**，形質** $c_1, \ldots, c_m$**，種形質行列** $M \in \{0,1\}^{n \times m}$

系統樹が作れるための必要十分条件 (整合性条件)

M **から系統樹を作ることができる** $\Leftrightarrow$
集合 $A_1, \ldots, A_m$ **の中の任意の 2 つ** A_i **と** A_j **が次のいずれかを満たす**

$$A_i \subseteq A_j, \quad A_i \supseteq A_j, \quad A_i \cap A_j = \emptyset$$

証明 ($\Rightarrow$**)：系統樹** T **が作れると仮定する**

- **任意の異なる形質** c_i **と** c_j **を選ぶ**
- **系統樹** T **において以下のいずれかが成り立っている**
  1. c_i **を割り当てている枝が** c_j **を割り当てている枝より下にある**
  2. c_i **を割り当てている枝が** c_j **を割り当てている枝より上にある**
  3. c_i **を割り当てている枝と** c_j **を割り当てている枝が同じである**
  4. c_i **を割り当てている枝と** c_j **を割り当てている枝が比較不能**

  図を板書

系統樹が作れるための必要十分条件：証明 (2)

種 $s_1, \dots, s_n$，形質 $c_1, \dots, c_m$，種形質行列 $M \in \{0,1\}^{n \times m}$

系統樹が作れるための必要十分条件 (整合性条件)

M から系統樹を作ることができる $\Leftrightarrow$
集合 $A_1, \dots, A_m$ の中の任意の 2 つ A_i と A_j が次のいずれかを満たす

$$A_i \subseteq A_j, \quad A_i \supseteq A_j, \quad A_i \cap A_j = \emptyset$$

証明 ($\Rightarrow$) 続き：系統樹 T が作れると仮定する

1. c_i を割り当てている枝が c_j を割り当てている枝より下にある
   - このとき，$A_i \subseteq A_j$ となる
2. c_i を割り当てている枝が c_j を割り当てている枝より上にある
   - このとき，$A_i \supseteq A_j$ となる
3. c_i を割り当てている枝と c_j を割り当てている枝が同じである
   - このとき，$A_i = A_j$ となる (すなわち，$A_i \subseteq A_j$ となる)
4. c_i を割り当てている枝と c_j を割り当てている枝が比較不能
   - このとき，$A_i \cap A_j = \emptyset$ となる

系統樹が作れるための必要十分条件：証明 (3)

種 $s_1, \ldots, s_n$，形質 $c_1, \ldots, c_m$，種形質行列 $M \in \{0,1\}^{n\times m}$

系統樹が作れるための必要十分条件 (整合性条件)

M から系統樹を作ることができる $\Leftrightarrow$
集合 $A_1, \ldots, A_m$ の中の任意の 2 つ A_i と A_j が次のいずれかを満たす

$$A_i \subseteq A_j, \quad A_i \supseteq A_j, \quad A_i \cap A_j = \emptyset$$

証明 $(\Leftarrow)$：$A_1, \ldots, A_m$ がこの条件を満たすと仮定

- 証明は m に関する帰納法で行う
- 基底段階：$m = 1$ のとき，
  A_1 とそれ以外を分けることで，系統樹は必ず作れる

系統樹が作れるための必要十分条件：証明 (3)

種 $s_1, \ldots, s_n$，形質 $c_1, \ldots, c_m$，種形質行列 $M \in \{0,1\}^{n \times m}$

系統樹が作れるための必要十分条件 (整合性条件)

M から系統樹を作ることができる $\Leftrightarrow$
集合 $A_1, \ldots, A_m$ の中の任意の 2 つ A_i と A_j が次のいずれかを満たす

$$A_i \subseteq A_j, \quad A_i \supseteq A_j, \quad A_i \cap A_j = \emptyset$$

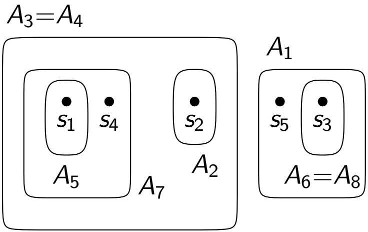
証明 ($\Leftarrow$) 帰納段階：$k < m$ である場合に成り立つと仮定する

- $A_1, \ldots, A_m$ の中で，$\subseteq$ に関して極大である集合を A_i とする
- 形質 c_i を持つ種と持たない種で分割し，系統樹の頭の部分を作る
- 残りの部分は帰納法の仮定を用いて作ることができる
(詳細は演習問題) □

証明における帰納段階と構成法 (あるいは，構成法の復習)

$A_j =$ 形質 c_j を持つ種全体の集合

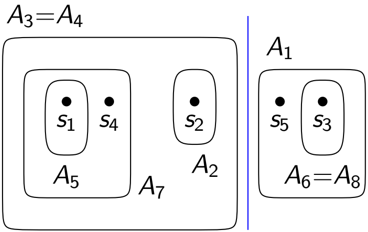
	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0



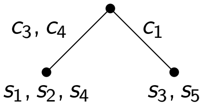
証明における帰納段階と構成法 (あるいは，構成法の復習)

$A_j =$ **形質** c_j **を持つ種全体の集合**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

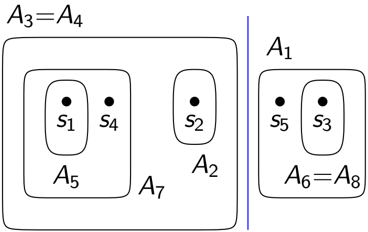


証明における帰納段階と構成法 (あるいは，構成法の復習)

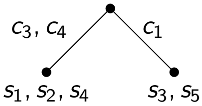


$A_j =$ **形質** c_j **を持つ種全体の集合**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

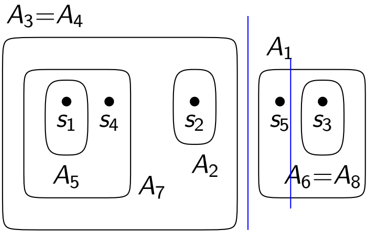


証明における帰納段階と構成法 (あるいは，構成法の復習)

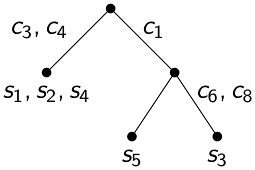


A_j = **形質** c_j **を持つ種全体の集合**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

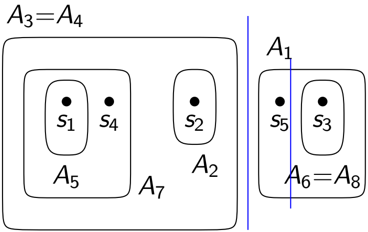


証明における帰納段階と構成法 (あるいは，構成法の復習)

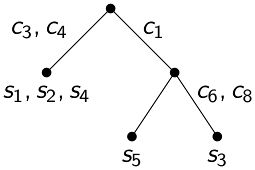


$A_j =$ **形質** c_j **を持つ種全体の集合**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

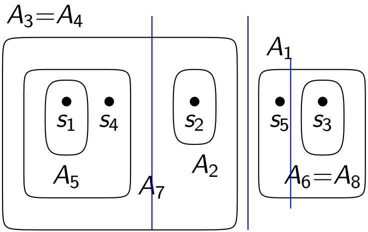


証明における帰納段階と構成法 (あるいは，構成法の復習)

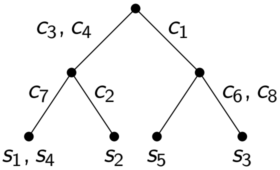


$A_j =$ **形質** c_j **を持つ種全体の集合**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

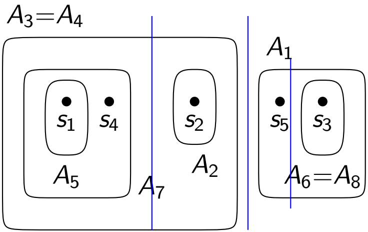


証明における帰納段階と構成法 (あるいは，構成法の復習)



$A_j =$ **形質** c_j **を持つ種全体の集合**

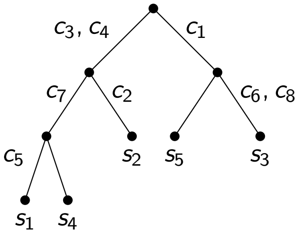
	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

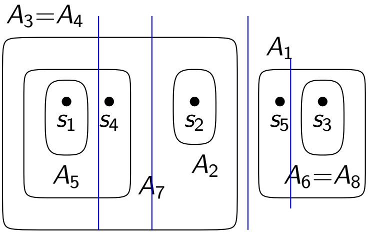


証明における帰納段階と構成法 (あるいは，構成法の復習)

A_j = 形質 c_j を持つ種全体の集合

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

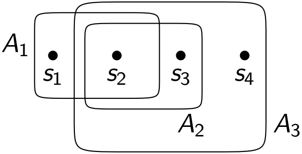




例 2：系統樹が作れない例

$A_j =$ 形質 c_j を持つ種全体の集合

	c_1	c_2	c_3
s_1	1	0	0
s_2	1	1	1
s_3	0	1	1
s_4	0	0	1



A_1, A_2 に対して

$$A_1 \subseteq A_2, A_1 \supseteq A_2, A_1 \cap A_2 = \emptyset$$

のどれも成り立ってないので，
系統樹が作れない

Binary Directed Perfect Phylogeny 問題の難しさ (易しさ)

定理 (Estabrook, Johnson, McMorris '76)

Binary Directed Perfect Phylogeny **問題は多項式時間で解ける**

- **より詳細には** $O(m^2 n)$ **時間**
- m **は形質の数，** n **は種の数**

後に， $O(mn)$ **時間アルゴリズム** (Gusfield '91)

展望

アルゴリズム，多項式時間， O **記法については**
『アルゴリズム・データ構造および演習』にて学習を

目次

現実のデータには欠損がある

Taxon \ Character		1 tail compared to body	2 dorsal crest	3 dewlap	4 head	5 tympanum	6 leg length	7 nuchal crest	8 body type	9 leg type	10 dorsal crest developm	11 nuchal crest developm	12 size	13 eye shape	14 elongated ribs	15 dewlap size
1	BronCris	longer	absent	present	wide	visible	long	present	narrow	narrow	?	reduced	large	round	absent	small
2	DraQuin	same	absent	present	wide	hidden	long	absent	narrow	narrow	?	?	medium	two points	present	small
3	GonoBorn	longer	present	present	wide	visible	long	present	robust	robust	developed	developed	large	one point	absent	small
4	CyrtCons	same	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	small	round	absent	?
5	VaraRud	longer	absent	absent	wide	visible	long	absent	robust	robust	?	?	very large	round	absent	?
6	AphanSp	longer	absent	absent	wide	hidden	long	absent	robust	narrow	?	?	medium	one point	absent	?
7	DraHae	same	absent	absent	wide	hidden	long	absent	narrow	narrow	?	?	medium	two points	present	?
8	MabRud	shorter	absent	absent	narrow	visible	long	absent	robust	robust	?	?	medium	round	absent	?
9	BlaStr	longer	absent	absent	wide	visible	long	absent	narrow	narrow	?	?	large	round	absent	?
10	CyrtCave	same	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	medium	round	absent	?
11	CyrtIng	shorter	absent	absent	wide	visible	short	absent	narrow	narrow	?	?	very small	round	absent	?
12	GonoGran	longer	present	present	wide	visible	long	present	robust	robust	developed	reduced	very large	one point	absent	large
13	GonoLio	longer	present	present	wide	visible	long	present	robust	robust	reduced	developed	large	one point	absent	large
14	GonoMjo	longer	absent	present	wide	visible	long	present	robust	robust	?	reduced	large	one point	absent	small
15	GonoRed	longer	absent	present	wide	visible	long	present	robust	robust	?	reduced	large	one point	absent	small
16	CosyPla	shorter	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	small	round	absent	?
17	CyrtPub	same	absent	absent	wide	hidden	short	absent	narrow	narrow	?	?	small	round	absent	?
18	HemiFren	shorter	absent	absent	wide	hidden	short	absent	robust	narrow	?	?	very small	round	absent	?
19	AptVit	same	absent	absent	narrow	hidden	short	absent	robust	narrow	?	?	medium	round	absent	?
20	MabInd	longer	absent	absent	narrow	visible	short	absent	robust	robust	?	?	large	round	absent	?
21	MabMult	same	absent	absent	narrow	visible	short	absent	robust	narrow	?	?	medium	round	absent	?
22	CanoSki	longer	absent	absent	narrow	visible	long	absent	narrow	narrow	?	?	small	two points	absent	?
23	StumSki	same	absent	absent	narrow	hidden	long	absent	narrow	narrow	?	?	medium	one point	absent	?

http://phylodiversity.net/bb09/images/9/9a/Kristinachar.png

不完全 Binary Directed Perfect Phylogeny 問題

- **入力：不完全な二値種形質行列** ($M \in \{0, 1, ?\}^{n\times m}$)
- **出力：根つき二分木で次を満たすもの**
  - **入力の不完全な部分を補う**
  - **各葉が** 1 **つの種でラベル付けされている**
  - **各種はラベルとして一度だけ出現する**
  - **各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	?	0
s_2	0	1	1	?	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	?	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

不完全 Binary Directed Perfect Phylogeny 問題

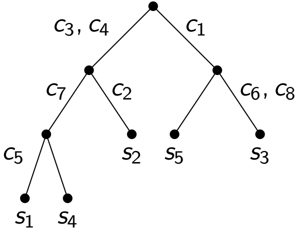
- 入力：不完全な二値種形質行列 ($M \in \{0, 1, ?\}^{n \times m}$)
- 出力：根つき二分木で次を満たすもの
  - 入力の不完全な部分を補う
  - 各葉が 1 つの種でラベル付けされている
  - 各種はラベルとして一度だけ出現する
  - 各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0

不完全 Binary Directed Perfect Phylogeny 問題

- **入力：不完全な二値種形質行列** $(M \in \{0,1,?\}^{n\times m})$
- **出力：根つき二分木で次を満たすもの**
  - **入力の不完全な部分を補う**
  - **各葉が** 1 **つの種でラベル付けされている**
  - **各種はラベルとして一度だけ出現する**
  - **各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素**

	c_1	c_2	c_3	c_4	c_5	c_6	c_7	c_8
s_1	0	0	1	1	1	0	1	0
s_2	0	1	1	1	0	0	0	0
s_3	1	0	0	0	0	1	0	1
s_4	0	0	1	1	0	0	1	0
s_5	1	0	0	0	0	0	0	0



不完全 Binary Directed Perfect Phylogeny 問題：より詳細な定義

- 入力：不完全な二値種形質行列 ($M \in \{0, 1, ?\}^{n \times m}$)
- 出力：根つき二分木で次を満たすもの
  - 入力の不完全な部分を補う
  - 各葉が 1 つの種でラベル付けされている
  - 各種はラベルとして一度だけ出現する
  - 各形質に対して，同じ状態の種が誘導する最小の部分木が互いに素

やりたいこと

行列 $M \in \{0, 1, ?\}^{n \times m}$ から行列 $M' \in \{0, 1\}^{n \times m}$ を次のように得ること

- $M_{i,j} = 0$ ならば $M'_{i,j} = 0$
- $M_{i,j} = 1$ ならば $M'_{i,j} = 1$
- M' から得られる集合 $A_1, \ldots, A_m$ が先ほどの整合性条件を満たすこと

例 1：どのように補えばよいか？

	c_1	c_2	c_3
s_1	1	0	0
s_2	1	1	0
s_3	?	1	1
s_4	0	?	1

- $\{s_1, s_2\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\{s_2, s_3\} \subseteq A_2 \subseteq \{s_2, s_3, s_4\}$
- $A_3 = \{s_3, s_4\}$

系統樹が作れるとすると

- A_1 と A_3 を見ると，$A_1 = \{s_1, s_2\}$ である
- A_2 と A_3 を見ると，$A_2 = \{s_2, s_3, s_4\}$ である
- しかし，このとき，A_1, A_2 に対して

$$A_1 \subseteq A_2, A_1 \supseteq A_2, A_1 \cap A_2 = \emptyset$$

のどれも成り立ってないので，系統樹が作れない

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	?	0	0	1
s_2	?	?	0	1	0
s_3	?	0	1	?	?

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	?	0	0	1
s_2	?	?	0	1	0
s_3	?	0	1	?	?

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

- $A_1 = \{s_1, s_2, s_3\}$ **とすれば** $A_1 \supseteq A_2, A_3, A_4, A_5$ **となるのでうれしそう**

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	?	0	0	1
s_2	1	?	0	1	0
s_3	1	0	1	?	?

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

- $A_1 = \{s_1, s_2, s_3\}$ **とすれば** $A_1 \supseteq A_2, A_3, A_4, A_5$ **となるので うれしそう**

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	?	0	0	1
s_2	1	?	0	1	0
s_3	1	0	1	?	?

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

- $A_1 = \{s_1, s_2, s_3\}$ **とすれば** $A_1 \supseteq A_2, A_3, A_4, A_5$ **となるので うれしそう**
- $A_2 = \emptyset$ **とすれば** $A_2 \subseteq A_3, A_4, A_5$ **となるので うれしそう**

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	0	0	0	1
s_2	1	0	0	1	0
s_3	1	0	1	?	?

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

- $A_1 = \{s_1, s_2, s_3\}$ **とすれば** $A_1 \supseteq A_2, A_3, A_4, A_5$ **となるので うれしそう**
- $A_2 = \emptyset$ **とすれば** $A_2 \subseteq A_3, A_4, A_5$ **となるので うれしそう**

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	0	0	0	1
s_2	1	0	0	1	0
s_3	1	0	1	?	?

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

- $A_1 = \{s_1, s_2, s_3\}$ **とすれば** $A_1 \supseteq A_2, A_3, A_4, A_5$ **となるので うれしそう**
- $A_2 = \emptyset$ **とすれば** $A_2 \subseteq A_3, A_4, A_5$ **となるので うれしそう**
- **他の「?」を 0 にすれば，** A_3, A_4, A_5 **は互いに素**

例 2：どのように補えばよいか？

	c_1	c_2	c_3	c_4	c_5
s_1	1	0	0	0	1
s_2	1	0	0	1	0
s_3	1	0	1	0	0

- $\{s_1\} \subseteq A_1 \subseteq \{s_1, s_2, s_3\}$
- $\emptyset \subseteq A_2 \subseteq \{s_1, s_2\}$
- $A_3 = \{s_3\}$
- $\{s_2\} \subseteq A_4 \subseteq \{s_2, s_3\}$
- $\{s_1\} \subseteq A_5 \subseteq \{s_1, s_3\}$

系統樹を作ろうとする

- $A_1 = \{s_1, s_2, s_3\}$ **とすれば** $A_1 \supseteq A_2, A_3, A_4, A_5$ **となるので うれしそう**
- $A_2 = \emptyset$ **とすれば** $A_2 \subseteq A_3, A_4, A_5$ **となるので うれしそう**
- **他の「?」を** 0 **にすれば，** A_3, A_4, A_5 **は互いに素**

条件を満たせた！

「うれしそう」であることをしてもよいという保証 (1)

種 $s_1, \ldots, s_n$，形質 $c_1, \ldots, c_m$，不完全種形質行列 $M \in \{0, 1, ?\}^{n \times m}$

性質

ある形質 c_k が存在して，$A_k = \emptyset$ となるように補完できる
このとき，

M に対して系統樹が作れる $\Rightarrow$ $A_k = \emptyset$ として系統樹が作れる

証明：M に対して系統樹が作れると仮定

- 集合 $A_1, \ldots, A_m$ が存在して，任意の i, j に次のどれかが成り立つ

$$A_i \subseteq A_j, A_i \supseteq A_j, A_i \cap A_j = \emptyset$$

- このとき，A_k を $\emptyset$ に変えても，この性質は成り立つ □

「うれしそう」であることをしてもよいという保証 (2)

種 $s_1, \ldots, s_n$，形質 $c_1, \ldots, c_m$，不完全種形質行列 $M \in \{0, 1, ?\}^{n \times m}$

性質

ある形質 c_k が存在して，$A_k = \{s_1, \ldots, s_n\}$ となるように補完できる
このとき，

M に対して系統樹が作れる $\Rightarrow$ $A_k = \{s_1, \ldots, s_n\}$ として系統樹が作れる

証明：M に対して系統樹が作れると仮定

- 集合 $A_1, \ldots, A_m$ が存在して，任意の i, j に次のどれかが成り立つ

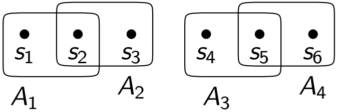
$$A_i \subseteq A_j, A_i \supseteq A_j, A_i \cap A_j = \emptyset$$

- このとき，A_k を $\{s_1, \ldots, s_n\}$ に変えても，この性質は成り立つ $\square$

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	?	0
s_2	1	1	?	?
s_3	?	1	0	?
s_4	0	?	1	?
s_5	?	0	1	1
s_6	?	0	?	1

? を全部 0 にしたときの状況

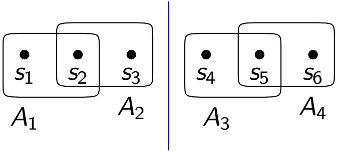


系統樹を作ろうとする

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	?	0
s_2	1	1	?	?
s_3	?	1	0	?
s_4	0	?	1	?
s_5	?	0	1	1
s_6	?	0	?	1

? を全部 0 にしたときの状況



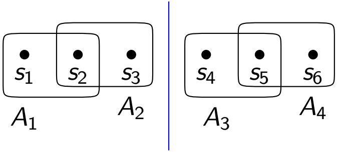
系統樹を作ろうとする

- 分けられるところで分けても問題なさそう

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	0	0
s_2	1	1	0	0
s_3	?	1	0	0
s_4	0	0	1	?
s_5	0	0	1	1
s_6	0	0	?	1

? を全部 0 にしたときの状況



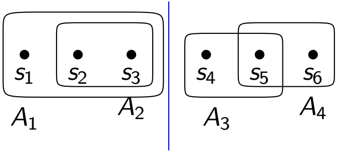
系統樹を作ろうとする

- 分けられるところで分けても問題なさそう

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	0	0
s_2	1	1	0	0
s_3	?	1	0	0
s_4	0	0	1	?
s_5	0	0	1	1
s_6	0	0	?	1

? を全部 0 にしたときの状況

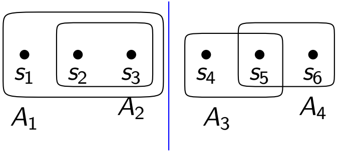


系統樹を作ろうとする

- **分けられるところで分けても問題なさそう**
- **左側では，$A_1 = \{s_1, s_2, s_3\}$ とすれば問題ない**

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	0	0
s_2	1	1	0	0
s_3	1	1	0	0
s_4	0	0	1	?
s_5	0	0	1	1
s_6	0	0	?	1

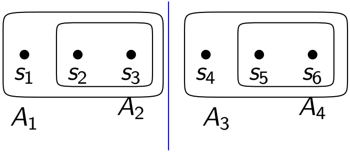


系統樹を作ろうとする

- 分けられるところで分けても問題なさそう
- 左側では，$A_1 = \{s_1, s_2, s_3\}$ とすれば問題ない

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	0	0
s_2	1	1	0	0
s_3	1	1	0	0
s_4	0	0	1	?
s_5	0	0	1	1
s_6	0	0	?	1

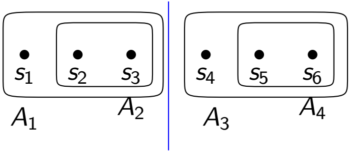


系統樹を作ろうとする

- **分けられるところで分けても問題なさそう**
- **左側では，$A_1 = \{s_1, s_2, s_3\}$ とすれば問題ない**
- **右側では，$A_3 = \{s_4, s_5, s_6\}$ とすれば問題ない**

例 3：どのように補えばよいか？

	c_1	c_2	c_3	c_4
s_1	1	0	0	0
s_2	1	1	0	0
s_3	1	1	0	0
s_4	0	0	1	0
s_5	0	0	1	1
s_6	0	0	1	1



系統樹を作ろうとする

- **分けられるところで分けても問題なさそう**
- **左側では，$A_1 = \{s_1, s_2, s_3\}$ とすれば問題ない**
- **右側では，$A_3 = \{s_4, s_5, s_6\}$ とすれば問題ない**

条件を満たせた！

「分けられるときに分ける」としてもよいという保証

種 $s_1, \ldots, s_n$，形質 $c_1, \ldots, c_m$，不完全種形質行列 $M \in \{0, 1, ?\}^{n \times m}$

性質

M における ? をすべて 0 にしたとき，
種の集合の分割 $\{S_1, S_2\}$ と形質の集合の分割 $\{C_1, C_2\}$ が存在して，
S_1 の種は C_2 の形質を持たない，そして，
S_2 の種は C_1 の形質を持たない，となるとする
このとき，

M に対して系統樹が作れる $\Rightarrow$ 上の分割から得られる補完を使って系統樹が作れる

「上の分割から得られる補完」とは？

その補完を M' とすると

- $s_i \in S_1$ かつ $c_j \in C_2$ のとき，$M'_{i,j} = 0$
- $s_i \in S_2$ かつ $c_j \in C_1$ のとき，$M'_{i,j} = 0$

不完全 Binary Directed Perfect Phylogeny 問題を解く手順

1. ある列の成分をすべて 1 に補完できるならば，そのように補完して，その列を削除
2. ある列の成分をすべて 0 に補完できるならば，そのように補完して，その列を削除
3. 残ったすべての ? を仮に 0 として，$A_1, \ldots, A_m$ を作る
4. うまく分割できるところで分割しようとする
   - 分割できた $\Rightarrow$ 小さくなった問題を同じ手順で解く
   - 分割できない $\Rightarrow$ 系統樹を作れない

不完全 Binary Directed Perfect Phylogeny 問題の難しさ (易しさ)

定理 (Benham, Kannan, Paterson, Warnow '95)

不完全 Binary Directed Perfect Phylogeny 問題は多項式時間で解ける

- より詳細には $O(mn^2)$ 時間
- m は形質の数，n は種の数

後に，$O(mn + k\log^2(m+n))$ 時間アルゴリズム

(Pe'er, Pupko, Shamir, Sharan, '04)

- k は「?」の総数

目次