

## 第4章

### 観察して評価すること

#### 4.1 データの種類と構成

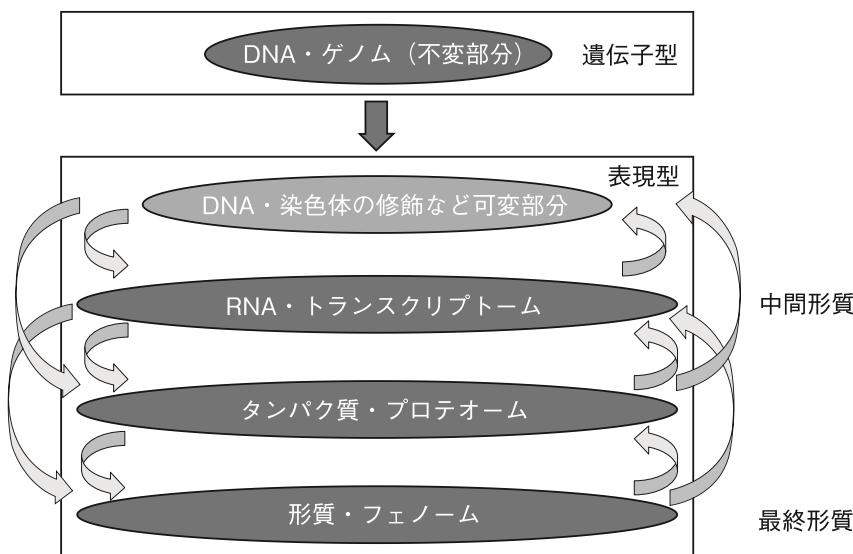
##### ● 4.1.1 遺伝子から見たデータの種類——遺伝子型と表現型、最終形質と中間形質

遺伝統計学では生命現象を遺伝子という観点で整理・解釈するので、データの種類と構成を考えるときにも、その視点から種類分けをします。2つに分かれます。

- ジェノタイプ（遺伝子型）
- フェノタイプ（表現型）

ジェノタイプというのは、親から子に伝えられる情報である「遺伝情報」に関するデータである、ということです。これは個体が生まれるとき（受精したとき）から死ぬまで、変わらずに保持する「何か」です。生まれるときからあるので、形質に影響を与えます。変わらずに存在するので、形質から影響を受けることはありません。ゲノムの型と言い換えてよいかもしれません。DNA分子とそれを含む染色体は化学修飾を受けたりして、変化することもあるので、DNA分子・染色体の状態のうち、変化しないものの型をジェノタイプと考えましょう。塩基配列がそれに相当します。

そして、ジェノタイプでないものをすべてひっくるめて、フェノタイプとし



データはジェノタイプ（遺伝子型）とフェノタイプ（表現型）に分けられます。ジェノタイプは不变。フェノタイプは可変。フェノタイプは形質のものですが、それらは、ジェノタイプからの遠近関係により、最終形質とその中間形質という関係で見ることができます。

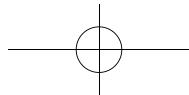
図4.1 ジェノタイプとフェノタイプ

ます。表現型は相互に影響を及ぼしたり、及ぼされたりします。RNAの発現の状態（トランスクリプトーム）はDNAの配列に影響を受けますが、発現調節をするタンパク質の影響を受けますし、そのタンパク質の発現は、さまざまなタンパク質とRNAの影響を受けます<sup>※1</sup>。

同じフェノタイプと言っても、それらは、段階的に構成されています。DNAから転写されてmRNAができます。さらに翻訳されてタンパク質ができます。そしてさらに、タンパク質をはじめとするさまざまな分子がさまざまな現象を起こします。ときには、その分子の量や機能を測定する検査などがあります。その検査は病気などの判断の有力な根拠になっているかもしれません。

このように、遺伝子から段階的に離れていきますが、mRNAもタンパク質も、検査も病気もすべて形質です。今、病気に特に興味がある場合、それは最終的に興味がある形質であるとして、最終形質と呼ばれます。このようなと

※1 塩基配列以外で親から子に伝達する情報もあります。その要素の影響をエピジェネティクスと言います。そのことを考えると、DNAの塩基配列とDNA・染色体の塩基配列以外の状態とを分けてしまうのは、乱暴に過ぎますが、本書では「割り切り方」の一方法として、このようにしておきます。



き、最終形質と関係する、最終形質以外の形質は**中間形質**とも呼ばれます。

「遺伝子から最終形質までの道のり」の中間にあるからです。

遺伝子発現を研究標的にするときには、mRNA 自体が最終形質になります。

### ● 4.1.2 解析対象としてのデータの種類——データ型

#### (1) データ型

データを遺伝子の視点で分類しましたが、解釈・解析する視点から分類することにします。その分類が**データ型**です。分類の視点は次の 2 つです。

- とりうる値が限られている（**離散的**）かそうでない（**連続的**）か
- **順序**があるかないか

離散的なデータ型は 2 個以上のカテゴリからなります。

表 4.1 データ型の分類

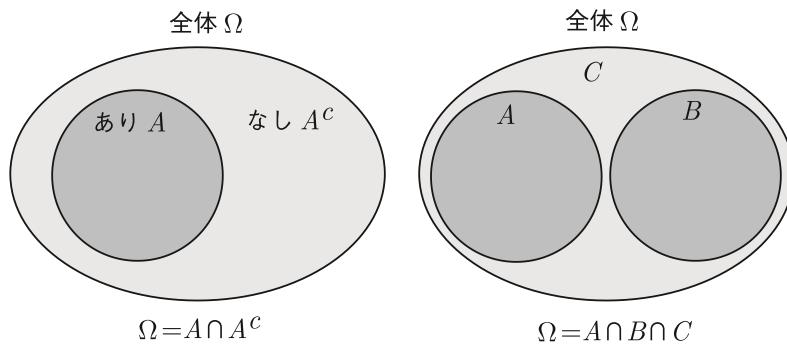
	離散		連続
順序の有無	2 カテゴリ	3 以上カテゴリ	量的
順序あり	△	○	○
一部順序あり	×	○	×
順序なし	△	○	×

この表では、○印はそのようなデータ型があることを示し、×印はないことを、△印は事情によって、あるともないとも言えることを示しています。

#### (2) 2 カテゴリ型は順序があるともないとも言える

- A さんは病気 X と診断されています（されていません）
- A さんの遺伝子 G の mRNA は検出されています（されていません）

このデータは、数え上げができる場合（カテゴリ）のどちらかに該当するので離散的です。場合の数が 2 つしかないので、カテゴリ数は 2 です。必ずどちらかのカテゴリに属します。どちらにも属さなかったり、両方に属することはできません。2 カテゴリ型は相互に排他的で、2 つのカテゴリを合わせると全体になります。全体集合に対して、部分集合とその補集合になっている、とも言います（図 4.2）。



全体を3カテゴリに分けると、相互に重なりのない（相互に排他的な）3つの集合に分かれます。 $A^c$ は $A$ の補集合。

図4.2 全体を2つのカテゴリに分けると、集合とその補集合に分かれる

表4.1で、2カテゴリ型は、順序がある、ないどちらにも△印がついています。これは、「診断・検出されている」ことを「されていない」ことに対して、「何かがある」と考えると、「ある」ことに1点、「ない」ことに0点をつければ、順序があることになりますし、「ある」と「ない」ことは並列な2のこと、と考えれば順序がないことになります。以降の話では、順序があるデータ型と一緒に取り扱われることもあり、順序がないデータ型と一緒に取り扱われることもあります。

### (3) 3以上カテゴリ型

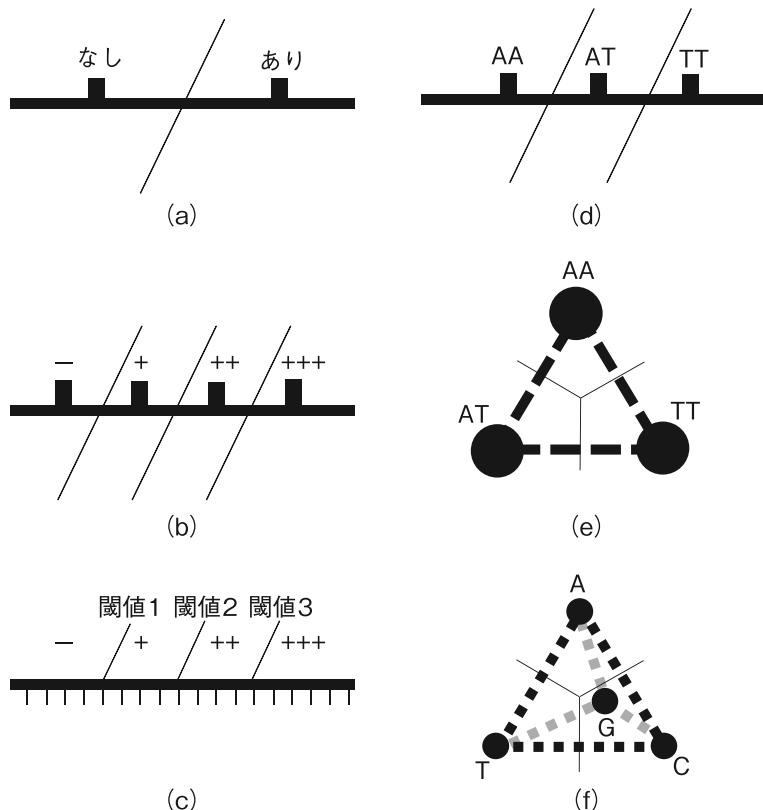
次の例を見てみます。

- Aさんの遺伝子Gの遺伝子多型Pの父由来の塩基は"A"です

塩基は"A" "T" "G" "C"の4つの可能性があります。そしてそれ以外の可能性はなく、必ずどれか1つをとります。ですから、4つのカテゴリを持つデータ型です。ここで、"A" "T" "G" "C"には特に順番の定めようもないで、順序なしです。図4.3 (f) のように3次元空間を4つの部分に分けることに相当します。

- Aさんの遺伝子多型Pのジェノタイプは"AT"です

この例では、3つのジェノタイプ "AA", "AT", "TT" がありえます。離散的な3つのカテゴリで、順序がないと見ることができます。この場合は図4.3 (e) のように平面を3つの部分に分けることに相当します。別の見方では、この遺



直線を区切る線を引くことによっていくつかの線分に分けると、順序のあるカテゴリに分けられます (a) (b) (d)。区切りの線を引かずに数直線として扱えば連続型です (c)。3以上のカテゴリは、全体が直線では表せず、平面・立体・高次元空間になります。この場合も全体を複数の相互に排他的なカテゴリに仕切る点は同じです。(e) (f) の2つの図は2次元平面を3カテゴリに分ける様子と、3次元空間を4カテゴリに分ける様子です。

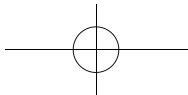
図 4.3 全体が直線で表せるときは、順序ありのデータ型とみなせる

伝子多型の遺伝子型としてはアレル "T" の本数に着目すると、0, 1, 2 本というように、数値を割り当てることができます。この場合は、順序のあるカテゴリです。この場合は、直線を3つに仕切ることに相当します (図 4.3 (d))。どちらで扱うかは、どのような点に着目したいかによって変わります。

一方、次の例は

- Aさんの検査 T (定性) の結果は "++" (強い) です

カテゴリ数は3個以上ですが、順序があります。"++" (強く陽性 (20 ~ 30)) は、"−" (陰性 (0-10)) , "+" (陽性 (10-20)) , "++" (強く陽性 (20 ~ 30)) , "+++" (非常に強く陽性 (30 以上)) という4個のカテゴリのうち、



2番目に強いカテゴリです。順序があるのは、直線で仕切ったものであるとして定義されているからです。量的データに、仕切り線の位置を決める値（閾値（10, 20, 30））が存在する場合もこのようになります。

#### (4) 連続型

- Aさんの検査T（タンパク質T：遺伝子Gの翻訳産物）（定量）の結果は256ユニットです

量で量ると数直線という1本の直線で表すことができ、データはこの直線上の点として表すことができます。では、AさんとBさんの2人のデータが、ともに256ユニットであったとき、2人のタンパク質量は同じなのでしょうか。たとえ256ユニットと同じ値だとしても、十分に精密に較べれば、全く同じということはないでしょう。そういう意味で、量的データの値は、数直線上の点で表される値を適当な精度で丸めた値であると考えておくのがよいでしょう。精度という細かさで多くの順序ありカテゴリに分けられたカテゴリ型のデータであるが、そのカテゴリ数が十分多いので、カテゴリであることを忘れてよいというように考えてもよいかもしれません。

### ◎ 4.1.3 一部に順序があること

では、順序が一部にある場合というのはどういう場合でしょうか。2つの2アレル型の多型のジェノタイプのデータを見てみます。

	CC	CG	GG		0	1	2
AA	AA, CC	AA, CG	AA, GG	0	(0, 0)	(0, 1)	(0, 2)
AT	AT, CC	AT, CG	AT, GG	1	(1, 0)	(1, 1)	(1, 2)
TT	TT, CC	TT, CG	TT, GG	2	(2, 0)	(2, 1)	(2, 2)

この表では、縦に多型P（A/T）、横に多型Q（C/G）のディプロタイプが表されています。これは、3個のカテゴリを持つ2つの因子の組み合わせのカテゴリ型データで、そのカテゴリは $3^2 = 9$ 個ある場合です。では、Pのアレル "T" の本数とQのアレル "G" の本数に着目してPもQも順序ありと考えます。9通りのカテゴリはそれぞれPとQとのアレルの本数のパターンで表すこともできます。特に特殊事情を考慮しなければ、Pの "T" の本数に順序があつ

て、Q の "G" の本数に順序を考えてよく、P の "T" も Q の "G" も病気 X のリストを上げているとき、

$$(0, 0) < (0, 1) < (0, 2)$$

$$(0, 0) < (1, 0) < (2, 0)$$

$$(0, 0) < (1, 1) < (2, 2)$$

の順序は問題がないでしょう。

はっきりしないのは、(0, 1) と (1, 0) の順序、(0, 2) と (1, 1) と (2, 0) の順序、(1, 2) と (2, 1) の順序です。これが、一部に順序があるけれども、一部に順序がない場合です。このような場合を**半順序**と言います。全体に順序をつけるためには、何かしらのルールを入れる必要があります。すべてに順序がある場合は**全順序**と言います<sup>※2</sup>。

#### ● 4.1.4 カテゴリの組み合わせ

多型 P, Q の例では、2つのカテゴリ型を組み合わせていました。カテゴリ数が  $k_i$  であるようなカテゴリ変数を  $N$  個組み合わせれば、

$$\prod_{i=1}^N k_i = k_1 \times k_2 \times \dots \times k_N$$

通りのカテゴリが生じます。

#### ● 4.1.5 唯一選択、重複選択

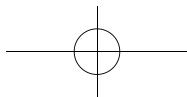
複数の項目からなる診断基準の例を見てみます。

- Aさんは5項目の診断基準のうち、3項目を満足しています

これは5項目のうちの該当する項目をいくつでも選ぶというタイプのデータ型です。それらはいくつかのパターンに分けられます。5カテゴリ ("a", "b", "c", "d", "e") から1つ選ぶときに得られたデータの記録方法としては次の表のような方法が考えられます。

---

※2 順序、全順序、半順序は数学で定義された概念です。ここでは、データの型を考えるためにあたって、順序という既存の概念を使えることを理解することが大事であって、その詳細には立ち入りません。詳細は成書を参照してください。



サンプル	カテゴリ
サンプル 1	b
サンプル 2	a
サンプル 3	a
サンプル 4	c

「 $n$ 個の中から選んでください。選ぶ数はいくつでも構いません」というような選択肢の場合もよく見かけます。選ぶ数に上限を設けて、「 $n$ 個の中から選んでください。ただし、最大3個までとします」ということも比較的多いかもしれません。こんな場合は、次のように記録することになるでしょう。

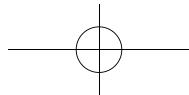
サンプル	選択カテゴリ 1	選択カテゴリ 2	選択カテゴリ 3
サンプル 1	b	—	—
サンプル 2	a	d	c
サンプル 3	a	c	d
サンプル 4	c	e	—

サンプルによって、選択カテゴリ数が異なると表の形で記録するのが難しくなります。サンプル2とサンプル3はともにa, c, dを選んでいますが、表の上では異なっています。次のようにしてみます。

サンプル	a	b	c	d	e
サンプル 1	0	1	0	0	0
サンプル 2	1	0	1	1	0
サンプル 3	1	0	1	1	0
サンプル 4	0	0	1	0	1

列の数が増えましたが、表の中はシンプルになりましたし、選択された項目の順番について気をつかわなくとも同じ選択パターンは同じレコードになっています。

この方式だと、「最大3個を選び、それに順序をつけなさい」という選択方法にも簡単に対応できます。「選ばないカテゴリを0点、1番目に選んだ項目に3点、2番目に選んだ項目に2点、3番目に選んだ項目に1点」と重みを付けることにして、



サンプル	a	b	c	d	e
サンプル 1	0	3	0	0	0
サンプル 2	3	0	1	2	0
サンプル 3	3	0	1	2	0
サンプル 4	0	0	3	0	2

のように、表の中の値に重みを反映するだけでうまく記録ができます。

言い方を変えると、順序のない 3 以上カテゴリ型の変数は、カテゴリごとに 2 カテゴリ型変数を割り当てて、それを組み合わせたものとして捉えることができることが、データの記録方式からわかります。

ただし、個々のカテゴリに割り当てた変数同士が独立かというと、そうではありません。5 カテゴリから 1 個だけを選択する条件の場合には、5 カテゴリに入れる値は 0 か 1 で、その和は 1 でなくてはならない、という制約が 5 つの変数の間にありますし、最大 3 個を選ぶという条件の場合には、5 カテゴリに入れる値は 0 か 1 で、その和は 3 以下でなくてはならない、という制約があります。

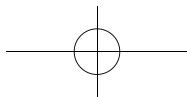
$N$  個のカテゴリから、 $k$  個を選ぶとき、その選び方は  $\binom{N}{k} = \frac{N!}{k!(N-k)!}$  通りあります（組み合わせ。選んだ項目が何番目かも問題にするしたら、 $\frac{N!}{(N-k)!}$  通りあります（順列））。

A, T, G, C を並べて長さ  $L$  の塩基配列を作るとき、その配列の場合の数は、重複順列で  $4^L$  通りでした。これに対して、3 以上カテゴリから選択するときの場合の数はこれより小さくなっています。この違いは、塩基配列のときに組み合わせた各塩基同士は独立であったのに対して、複数項目からの選択の場合には、選択項目間は独立でないことを反映しているとも言えます。

### ● 4.1.6 2 倍体という特殊性——ハーディ・ワインバーグ平衡 (HWE) の正確検定

カテゴリに注意して表を作ることが便利であることの例として、HWE の正確確率検定を挙げることにします。

ジェノタイプは父母由来のアレルの組み合わせです。今、ある 2 アレル型多型があり、その 3 ディプロタイプ "AA", "AT", "TT" の人数がそれぞれ、 $n_0$ ,  $n_1$ ,  $n_2$ ;  $n_0+n_1+n_2=N$  人だとします。アレル A と T の染色体本数が  $n_A$ ,  $n_T$  とし



ます。 $N$ 行2列の表があり、各行の和がすべて2で、第1、2列の和が $n_A$ ,  $n_T$ で、総和が $2N$ である表ができました。

サンプル	Aの本数	Tの本数	計
AA1	2	0	2
AA2	2	0	2
...	...	...	...
AA $n_0$	2	0	2
AT1	1	1	2
AT2	1	1	2
...	...	...	...
AT $n_1$	1	1	2
TT1	0	2	2
TT2	0	2	2
...	...	...	...
TT $n_2$	0	2	2
計	$n_A$	$n_T$	$2N$

これは「2カテゴリから、重複を許して必ず2個選べ」というタイプのデータ型です。

第1列の値が0, 1, 2のときには、第2列の値は必ず2, 1, 0という関係になっていて、片方の列はなくてもわかります。2列分の情報は冗長です。では、この2列は全く不要なのかというと、そうではありません。2アレル型多型のハーディ・ワインバーグ平衡を考えるときには、この2列を持つ表が活用されます。この表を用いて、HWEの正確確率検定をしてみます。

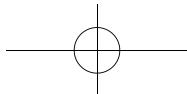
一般的な分割表の正確確率検定については、第13章を参照してください。

このような分割表の生起確率は、2のセルが $n_0 + n_2$ 個、0のセルが $n_0 + n_2$ 個、1のセルが $2 \times n_1$ 個、それぞれあるので、

$$\frac{n_A!n_T!\times\prod_{i=1}^N(2!)^{n_i}}{(2N)!(2!)^{n_0+n_2}(0!)^{n_0+n_2}(1!)^{2\times n_1}} = \frac{n_A!n_T!\times 2^{n_1}}{(2N)!}$$

となります。

$0!=1$ ,  $1!=1$ ,  $2!=2$ ,  $N=n_0+n_1+n_2$ を使って式変形しました。



ここまででは、 $N$ 行を区別していました。これは  $N$ 人のそれぞれを区別していたことになるのですが、今は  $N$ 人のうち 3 ジェノタイプの内訳が  $n_0, n_1, n_2$  である場合を区別しないので、 $n_0, n_1, n_2$  人で観測する場合は  $\frac{N!}{n_0!n_1!n_2!}$  通りあり、これらの区別がつきません。したがって、そのような確率は結局、

$$\frac{N!}{n_0!n_1!n_2!} \times \frac{2^{n_1} n_A! n_T!}{(2N)!} = \frac{2^{n_1} n_A! n_T! N!}{(2N)! n_0! n_1! n_2!}$$

となります。

観測した表の生起確率がわかりましたので、正確検定をするためには、次のようにします。

まず、与えられた条件（ $n_0, n_1, n_2$  が与えられた条件。必然的に  $n_A, n_T$  も決まります）のもとで、起こりえるすべての表を数え上げ、その生起確率を求めます。その上で、観測表の生起確率以下の生起確率をとる表について足し合わせます。

今、 $n_A + n_T = 2N$  なので、 $n_A$  と  $n_T$  はともに偶数か、ともに奇数のどちらかです。 $n_A, n_T$  が奇数のときには、ヘテロ型の人数  $n_1$  は奇数しかとらず、 $n_A, n_T$  が偶数のときには、 $n_1$  は偶数しかありません。

そのことを用いて HWE の正確確率検定を R で行ってみるソースが R4-1.R です。とりうる場合の確率のすべてを足し合わせると確かに 1 になっていることも確認できます。

#### R ソース 4.1 R4-1.R : HWE 正確確率検定

```

1 hweExact<-function(g=c(813,182,5)){ # 3 ジェノタイプの人数
2   n<-sum(g) # 総人数
3   nA<-2*g[1]+g[2] # A アレル本数
4   na<-2*g[3]+g[2] # a アレル本数
5   evod<-g[2]%%2 # ヘテロ人数の偶数奇数判断
6   maxAa<-min(nA,na)-evod
7   Aa<-seq(from=evod,to=maxAa,by=2) # 観測しうるヘテロ人数のベクトル
8   AA<-(nA-Aa)/2 # 観測しうる AA 人数
9   aa<-(na-Aa)/2 # 観測しうる aa 人数
10  obs<-(g[2]-evod)/2+1 # 観察データのヘテロ人数は Aa [obs]
11  prob<-rep(0,length(Aa)) # 観測しうる表の生起確率
12  prob<-exp(n*lgamma(2+1)+lgamma(nA+1)+lgamma(na+1)-

```

```

    lgamma(2*n+1) - (AA*lgamma(2+1)+Aa*lgamma(1+1)
    +aa*lgamma(2+1))+lgamma(n+1)-(lgamma(AA+1) +
    lgamma(Aa+1)+lgamma(aa+1)))
13 p.value<-sum(prob [prob<=prob [obs]])
    # 観測表の生起確率以下の生起確率を持つ表の生起確率の和
14 # Aa 観測しうるヘテロ人数リスト
15 # prob ヘテロ人数別の生起確率
16 # obsprob 観察テーブルの生起確率
17 # p.value 正確検定p値
18 list(Aa=Aa,prob=prob,obsprob=prob [obs],p.value=p.value)
19 }
20 xx<-hweExact(c(813,182,5))
21 xx$p.value # 検定p値
22 sum(xx$prob) # 全表の生起確率の和は1になります

```

### ● 4.1.7 親項目と子項目

複数の項目を組み合わせたときに、それらが独立である場合もあれば、相互に関係がある場合もあることはすでに述べてきました。

- Aさんは病気Xのタイプbと診断されています

このカテゴリ型項目はどうでしょうか。この項目のデータは病気Xという項目が「あり」の場合にのみ存在します。病気Xという項目を親項目とすれば、その子項目と言えます。アンケート調査などにもこのような項目間の帰属関係があります。「前の設問で（1）と答えた場合に、以下の設間に答えよ」という形式の場合です。データの構造が複雑になってきたときには、このような項目間の相互帰属関係を明確にして取り扱うことも適切です。

このような項目の帰属関係はフェノタイプにのみあるわけではありません。ジェノタイプの場合には、次のような例があります。Y染色体上にあって、Y染色体固有の遺伝子に関するジェノタイプは、Yを持っていて初めて意味をなしますから、Y染色体の保有に関する項目と、Y染色体固有のジェノタイプに関する項目は親子関係です。また、挿入型の遺伝子多型があって、その挿入配列がさらに多型性を持つ場合にも、遺伝子型の間に親子関係があります。親子関係はグラフ理論で言うところの木で表すことができますので、第5章で扱うグラフに関する説明が参考になります。

### ● 4.1.8 カテゴリの配置、カテゴリ間の非独立性、正単体

5 カテゴリから 3 カテゴリを選択するときに、5 列の表でデータを表すときには、表に 0 か 1 の値を入れ、行についての和が 3 になるようにするという制約がありました。ある列に入れる値によって、別の列に入れる値が影響を受けています。これが、カテゴリ間の非独立性です。このことを異なる視点から眺めてみます。

相互に関係のない  $n$  カテゴリは、**正単体**の頂点に配置できることが知られています。正単体とは、正三角形を多次元一般化したものと考えればよく、次のように説明できます。

- $k$  次元空間にある  $k$ -正単体は  $k+1$  個の頂点を持ち、すべての頂点は中心から等距離にあって、中心から各頂点への方向はどの 2 頂点を取り上げても等しい角度  $\theta$  をなす ( $\cos\theta = -\frac{1}{k}$ ) <sup>※3</sup>。

<sup>※3</sup>  $k+1$  次元空間の  $k+1$  個のベクトルを、 $(e_i = 0, 0, \dots, 0, 1, 0, \dots, 0); i = 1, 2, \dots, k+1$

という単位ベクトルで考えます。この  $k+1$  個のベクトルは  $k+1$  次元空間にあって、すべてのベクトルは直交しています。今、このすべての中心を

$$o = \frac{1}{k+1} \sum_{i=1}^{k+1} e_i = \frac{1}{k+1} (1, 1, 1, \dots, 1)$$

とします。ここで、

$$\begin{aligned} f_i &= e_i - o \\ &= \left( -\frac{1}{k+1}, -\frac{1}{k+1}, \dots, -\frac{1}{k+1}, 1 - \frac{1}{k+1}, -\frac{1}{k+1}, \dots, -\frac{1}{k+1} \right) \end{aligned}$$

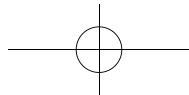
とします。 $\sum_{i=1}^k f_i = -f_{k+1}$  なので、これらは、 $k$  次元亜空間に納まっており、すべての  $f_i$  は同じ長さ

$$|f_i| = \sqrt{\frac{1}{(k+1)^2} (k * 1 + k^2)} = \sqrt{\frac{k}{k+1}}$$

を持ち、任意の  $i, j, i \neq j$  について、

$$f_i f_j = (k-1) * \left( \frac{1}{k+1} \right)^2 + 2 \times \left( -\frac{1}{k+1} \right) \left( 1 - \frac{1}{k+1} \right) = -\frac{1}{k+1}$$

であるから、そのなす角  $\theta_{i,j}$  は  $\cos(\theta_{i,j}) = \frac{f_i f_j}{|f_i| |f_j|} = -\frac{1}{k+1} \frac{k+1}{k} = -\frac{1}{k}$  です。



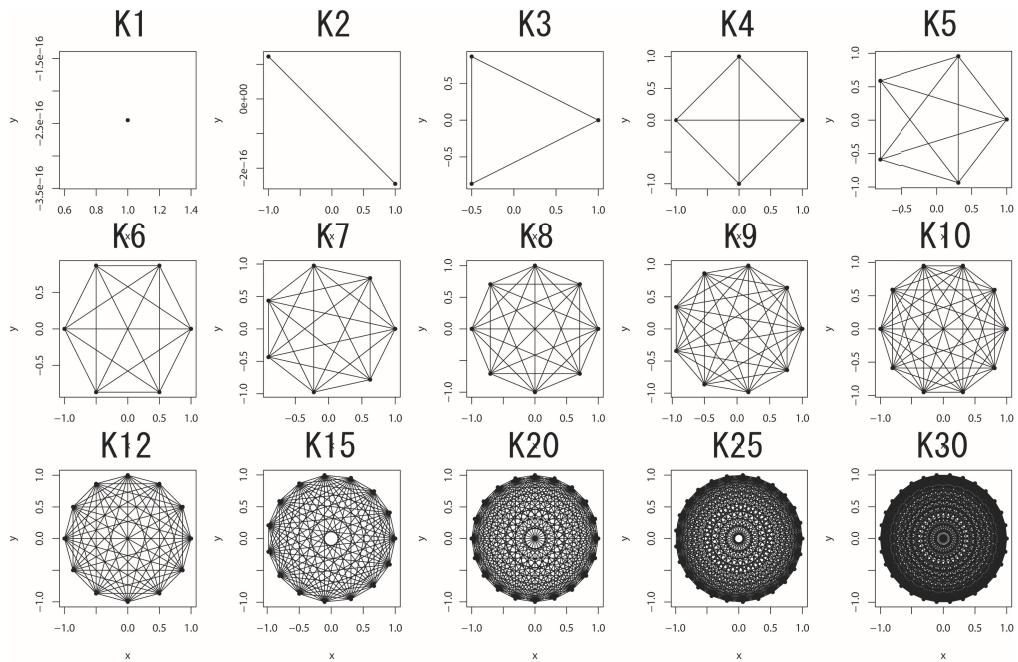
低次元の場合には次のようにになります。

- 1次元空間（直線）：1- 正単体 線分
- 2次元空間（平面）：2- 正単体 正三角形
- 3次元空間（空間）：3- 正単体 正四面体

$k+1$  個のカテゴリデータ型ではすべてのカテゴリが相互に対等な関係にあって、 $k$  次元空間中の  $k$ - 正単体の頂点とみなすことができます。

$k=1$  のとき（カテゴリ数が 2 のとき）は、2- 正単体の頂点に配置できるわけですが、2- 正単体とは、ただの線分で、その両端に 2 つのカテゴリが配置されています。直線配置できる点が連続データ型と共通であって、直線配置できることが「順序あり」扱いできることですので、2 カテゴリカルデータ型は必ず順序ありなのです。また、2 カテゴリは相互に逆向きのベクトルなので、片方のデータを知れば、もう片方のデータは自ずから確定するのも、こういう事情です。

さて、カテゴリ数  $k$  を大きくしていくと、 $\cos(\theta_{i,j}) = -\frac{1}{k}$  は 0 に収束しています。 $\theta_{i,j}$  は直角に近づいていくということです。直交関係は独立な関係なので、カテゴリ数が大きくなると、カテゴリ同士は相互に独立な関係に近づいていくことがわかります。第5章で扱うグラフで正単体を描けば、頂点数  $k+1$  の完全グラフとなります。すべての頂点の間に辺を引いたグラフで、**完全グラフ**という名前で呼ばれます。また、正単体という高次元表現にも、グラフという平面表現にも共通するのは、すべての頂点を相互に平等に扱う方法であるという点です。



頂点数 : 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 12, 15, 20, 25, 30。 $K_n$  の  $n$  が頂点数。

図 4.4 カテゴリを表す正単体は完全グラフ

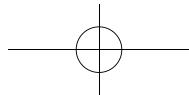
## 4.2 データを比較する

### ● 4.2.1 2つのデータの関係——対称的な関係と非対称的な関係

前節ではデータの分類についてまとめました。今度はデータを引き較べることにします。引き較べることを、**関係**<sup>※4</sup>を見る作業と呼ぶことにします。まず、関係には方向性があって、A のデータの B のデータに対する関係を見るのか、B のデータの A のデータに対する関係を見るのかの 2通りのやり方があります。

- 病気 X と診断されているかどうかという項目について、A さんは "1" (はい、そうである)、B さんも "1" です

※4 ここで言う「関係」は数学的には**二項関係**と定義されています。



A と B との関係は、 $A = B$  か  $A \neq B$  のいずれかです。どちらの場合も A のデータの B のデータに対する関係と B のデータの A のデータに対する関係とが同じです。これを、「**対称的**」な関係と言います。次は「**非対称的**」な関係の例です。

- A さんは X を 1つ持ち、B さんは 2つ持ちます

このデータは順序があります。そして、A さんのデータは B さんのそれに對して「小さく」、B さんのデータは A さんのそれに対して「大きい」です。2 方向の関係が異なりますから、これは、「**非対称的**」な関係です。

### ● 4.2.2 非対称な関係を対称にする——距離

順序のあるデータ型は非対称な関係をもたらしました。次の例を見てみます。

- A さんの検査 T (定性) の結果は "++" で、B さんのそれは "+++" です

順序のあるカテゴリ型データで、非対称な関係です。これを

- A さんの検査 T (定性) の結果は B さんのそれと "1" 段階違います
- B さんの検査 T (定性) の結果は A さんのそれと "1" 段階違います

と評価するとします。こうすると、「対称的」な関係になります。順序があるデータについて「差」をとることによって、「対称的」にしています。その結果、負の値をとらなくなりました。絶対値を採用したとも言えます。これは、**距離**と呼ばれるものです。

距離を扱いやすいように定義します。

#### (1) 距離の定義

- 2つの何か (A, B) の間に定められ
- 非負 (0 以上) の値をとり
- その値は、A の B に対する値も B の A に対する値も同じであり
- A と B とが同じときには 0 をとり
- 3つの何か (A, B, C) があるときには、A-B, B-C, C-A の間の距離は、A, B, C の3つを平面に配置して三角形 (つぶれてしまって直線になってしまっても

よい) が描けるような値をとるもの (三角不等式を満足する)

## (2) 三角不等式

ここで、距離を定義するのに三角不等式という用語が出てきました。これを説明します。

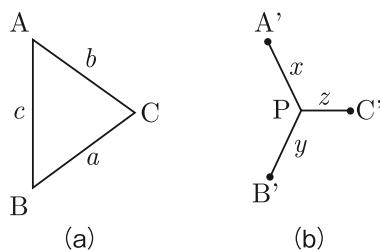


図 4.5 距離

図 4.5 (a) を見てください。平面上に 3 点 A, B, C があります。A から C へ行くのに、まっすぐ行く方が B を経由して行くよりも距離が短いです。この関係がどの点からどの点へ行く場合にも当てはまります。 $\triangle ABC$  の辺 AB, BC, CA の長さがそれぞれ  $c, a, b$  とすると、

$$b + c > a, c + a > b, a + b > c$$

が成り立ちます。このとき、

$$x = \frac{a+b+c}{2} - a = \frac{1}{2}(b+c-a) \geq 0$$

$$y = \frac{a+b+c}{2} - b = \frac{1}{2}(c+a-b) \geq 0$$

$$z = \frac{a+b+c}{2} - c = \frac{1}{2}(a+b-c) \geq 0$$

$$x + y = c, y + z = a, z + x = b$$

が成り立ちますから、 $A', B', C'$  の 3 点を点 P からそれぞれ  $x, y, z$  の距離に配置すれば (図 4.5 (b))、 $A'$  から  $B'$  へ P を経由して行く道のりの長さは、AB の道のりと同じです。 $B'C', C'A'$  の場合も同様です。したがって、**三角不等式** が成り立つような 3 点の距離関係は、3 点間の道のりを変えることなく、枝分かれで描けることがわかります。本章の後半では、複数の点の間の距離の情報

から、この枝分かれ道（木）を描く手法のことを述べますが、距離の定義はそのような木を作れることを保証しています。

### ● 4.2.3 ユークリッド距離とそれ以外の距離

データが数直線上にあるとき、関係を距離として表すことができました。私たちが日常生活で使っているいわゆる距離は、平面上（2次元）にも空間中（3次元）にも定義されています。それは**ユークリッド距離**と言われるもので、 $k$ 次元空間においては点  $A = (a_1, a_2, \dots, a_k)$  と点  $B = (b_1, b_2, \dots, b_k)$  の間のユークリッド距離は

$$d_E(A, B) = \sqrt{\sum_{i=1}^k (a_i - b_i)^2} = \sqrt{\sum_{i=1}^k \delta_i^2}$$

ただし、 $\delta_i = |a_i - b_i|$  と定義されます。

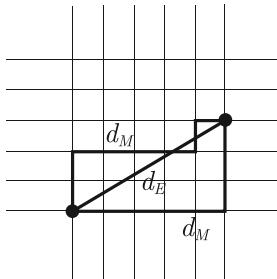
しかしながら、距離の定義はすでに述べた通りで、この式を必ずしも要求していません。ですから、遺伝子型や表現型のデータを使って距離を定めるときも、ユークリッド距離に限る必要はありません。たとえば、

$$d_M(A, B) = \sum_{i=1}^k |a_i - b_i| = \sum_{i=1}^k \delta_i$$

で定義される**マンハッタン距離**というものがあります。マンハッタン距離は、図4.6のように格子の辺をたどっていくときの距離です。

ユークリッド距離とマンハッタン距離以外にも距離はいくつも定義できます。脚注を参考にRを活用していろいろな距離の名称と定義を確認してみてください<sup>※5</sup>。

<sup>※5</sup> Rで距離を扱う関数であるdist()関数では、"euclidean", "maximum", "manhattan", "canberra", "binary", "minkowski"の6種類から選択することができます。dist()関数の説明文書（Rのヘルプ記事）を表示したり、dist()関数のソースコード（Rのソース）を表示することで、これらの内容がわかります。



2つの黒丸の間に直線で結んだ距離がユークリッド距離 ( $d_E$ ) で、格子（マンハッタン街区のような）を最短でたどるのがマンハッタン距離 ( $d_M$ ) です。ユークリッド距離の道は1通りですが、マンハッタン距離の道のとり方は1通りではありません。今、 $k$  次元で  $d_M = \sum_{i=1}^k \delta_i$  であるとすると、通り道のとり方の場合の数は、 $\binom{d_M}{\delta_1 \delta_2 \dots \delta_k} = \frac{d_M!}{\prod_{i=1}^k \delta_i!}$  です（第15章参照）。

図 4.6 マンハッタン距離

#### ● 4.2.4 配列の違いとマンハッタン距離

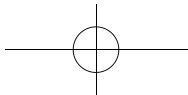
たとえば、DNA 塩基配列同士の距離は、塩基の異なる箇所数で表すことができます。

- ヒト : cccggaCAcCgActtcccGgggctcatt
- マウス : cccggaTGcAgGttcccAgggctcatt

この部分の配列の長さは 29 塩基で、5 塩基が違います。これは、29 次元空間で、マンハッタン距離が 5 と考ることができます。変異が 5 箇所に起きたことをマンハッタン距離 5 とみなしているわけです。配列の比較には、塩基の置換の他に、挿入・欠失も考慮が必要です。

- ヒト : ...cccggaCAcCgActtcccGgggctcattACcctCAc...
- マウス : ...cccggaTGcAgGttcccAgggctcatt=Tcct=Tc...

この場合には、置換と挿入・欠失という異質な出来事を総合的に考えて距離を決める必要があります。それぞれの出来事に、生物学的な背景を考慮して重み付けをします。また、塩基配列の比較では、2 配列がランダムに作られたという仮説に立つと、ありえないほど似ており、そのありえなさを数値にして、配列の似ている程度を数値化しますが、その珍しさを極値分布に照らして  $p$



値として数値化することができます。よく使われる公共サイトのblastなどがその例です。

### ● 4.2.5 距離の代わりに角度で表す——相関係数

距離は対称な関係を0以上の値で表現しました。負の数も含めて関係を量的に評価することはできないでしょうか。カテゴリ型を空間に配置したときに、

すべてのカテゴリが相互に  $\theta = \cos^{-1} \frac{1}{k}$  なる関係<sup>※6</sup>にあって、この角度が均等

であることが、カテゴリの平等な関係を意味していると述べました。このように角度も2つのものの間の関係を表すことができます。 $\theta$ をその関係を表す量とすることができますし、 $\cos \theta$ をその量としてもよいでしょう。今、2つのデータがベクトル形式であれば、ベクトル同士の内積と長さで表されます。

$$\cos \theta = \frac{\sum_{i=1}^k x_i y_i}{\|x\| \|y\|}$$

これは相関係数と呼ばれる値です。



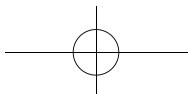
## 4.3 複数のサンプル、たくさんの比較

### ● 4.3.1 1対N-1とN対N

2個のサンプルを比較してきました。ここからは、N個のサンプルを較べます。

ある1個に特に興味があって、それ以外の( $N-1$ )個との関係が調べたいときもあります。N個の全部に興味があるときには、すべての関係を調べることになります。関係が対称でないときには、 $N \times N$ (重複順列)の比較をしますし、関係が対称であるときには自身との関係の情報もとりたければ、 $\frac{N(N+1)}{2}$ ペア(重複組み合わせ)の関係を調べますし、自身との関係は"0"

※6  $\cos \theta = x$  のとき  $\theta = \cos^{-1} x$



であって調べるには及ばないということであれば、 $\frac{N(N-1)}{2}$  ペア（組み合わせ）の関係を調べます。 $N \times N$  の関係を図 4.7 のように行列で表せば、 $N \times N$  行列全体、対称的な関係のときは**下三角行列**部分のみ、1 対その他ならば 1 行のみを使います。（●?本文中は N、図中は n、OKor どちらかに統一? ●）

$v(1, 1)$	$v(1, 2)$	...	$v(1, i)$	...	$v(1, n)$
$v(2, 1)$	$v(2, 2)$	...	$v(2, i)$	...	$v(2, n)$
...	...	...	...	...	...
$v(i, 1)$	$v(i, 2)$	...	$v(i, i)$	...	$v(i, n)$
...	...	...	...	...	...
$v(n, 1)$	$v(n, 2)$	...	$v(n, i)$	...	$v(n, n)$

すべての組み合わせは正方形行列全体 ( $N \times N$ )、対称な関係を見るときは、三角行列部分のみが対象。

図 4.7 比較の仕方

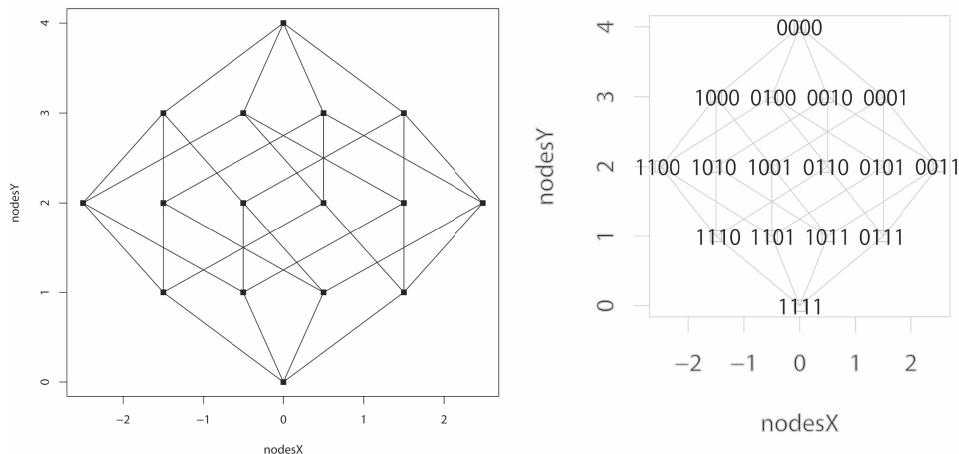
### ● 4.3.2 一部に関係がないとき——半順序

$N \times N$  のすべてに関係が定義できれば行列状に関係を見ればよいですが、関係が定義できない場合が含まれることもあります。

例を挙げます。4 個の選択肢からの選択を考えます。1 個も選ばない場合を  $\{0, 0, 0, 0\}$ 、1 番と 3 番を選ぶ場合を  $\{1, 0, 1, 0\}$  と表すことにします。こうすると、選び方は全部で  $2^4 = 16$  通りあります。1, 2, 3 番を選んだ場合と 1, 3 番を選んだ場合は、前者の選択は後者の選択を含みます（包含関係にあります  $\{1, 2, 3, 0\} \subset \{1, 0, 1, 0\}$ ）。しかしながら、 $\{1, 1, 1, 0\}$  と  $\{1, 0, 0, 1\}$  との関係は、包含関係が成り立ちません。このように包含関係が定まる場合と定まらない場合があります。 $2^4$  通りを 4 次元空間の点とみなして、 $k=4$  次元の長さ 1 の**立方格子**\*7 の点として表すとします。

要素数が 1 個だけ異なっていて、包含関係にある点同士が辺で結ばれます。 $2^k$  個の点を持つ多次元の格子は平面に描くのが難しいので、図 4.8 のように平面に押しつぶして描くことができます。図の上部にある点から下向きの辺だけをたどって下部にある点たどりつける場合には、上の点は下の点に含まれる関

\*7 1 次元の立方格子点  $(0), (1)$  を結んだ線分、2 次元立方格子は点  $(0, 0), (0, 1), (1, 0), (1, 1)$  を結んだ正方形、3 次元立方格子は点  $(0, 0, 0), (0, 0, 1), (0, 1, 0), (1, 0, 0), (0, 1, 1), (1, 0, 1), (1, 1, 0), (1, 1, 1)$  を結んだ立方体。 $k$  次元立方格子は  $2^k$  個の点からなり、その点の座標の成分は 0 か 1 でできている。



ハッセ図とも言います。要素数 4 の集合の各要素を含むか否かを 0, 1 で表して、ハッセ図の各頂点に部分集合を割り当てれば（右図）、ハッセ図は部分集合間の包含関係を表します。0 と 1 の数値の並びをハプロタイプとみなせば、1 回の変異で変化しうる間柄のハプロタイプ同士が結ばれていることになります（R4-sup1.R（この R ソースはダウンロードすることができますが、本書には掲載していません）でハッセ図を描けます）。

図 4.8 4 次元立方格子を平面に描く

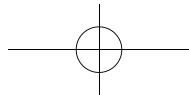
係になっています。

このように一部に順序があり、一部に順序がないときに、このような図で表されることがわかります。ハッセ図と言います。「一部に順序がある」という呼び方をしていましたが、それを半順序と言います。「全部に順序がある」のが全順序です。

"0, 1" を 2 つのアレルとすれば、4箇所の 2 アレル型多型が作る 16 種類のハプロタイプを変異回数で結んだ場合も多次元立方格子で表されます。

### ● 4.3.3 距離行列と木

今、 $N$  個の要素同士のすべてに対称的な関係があつて距離として表されるとします。 $N$  個のサンプルの  $N$  個のサンプルに対する関係を調べるとします。距離で表す関係は対称なので、（下）三角行列と呼ばれる行列に納めることができます。さらに、距離であるときには、自身と自身との関係は 0 であつて不要なので、以下に示すように、対角成分のない三角行列成分として表すことができます。これが距離行列です。行番号（2, 3, 4, 5）と列番号（1, 2, 3, 4）とともに距離が表示されています。



## ■ R の出力結果

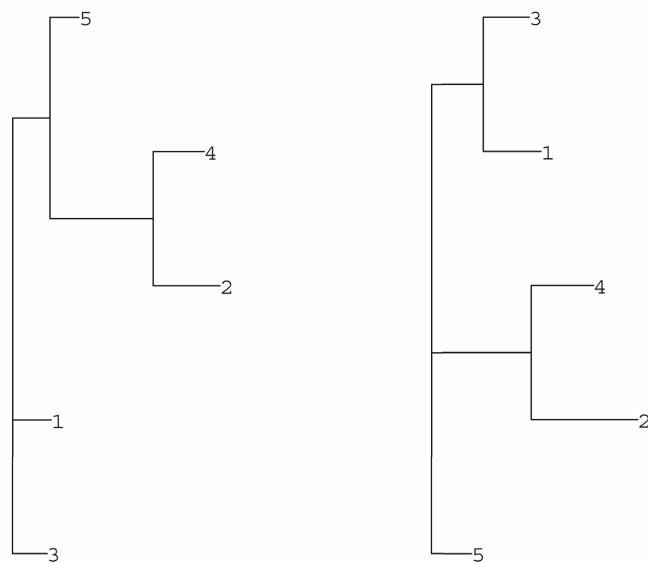
```
1          2          3          4
2 4.725108
3 1.372943 4.600435
4 4.135837 2.220284 4.311534
5 2.020896 3.608861 1.844573 3.591590
```

距離の定義としてユークリッド距離とマンハッタン距離を用いて、木を作つてみます。R のパッケージ<sup>※8</sup>ape を取り込み、ape が持つ、近隣結合法 (Neighbor-Joining) による木作成関数 nj() を使います。距離の定義が違うので、距離行列も異なりますから、そこから作られる木も少し違ってきます。木の作成はサンプルの階層的クラスタリングの一種です。階層的クラスタリングは引き続き第 5 章で扱います。

### R ソース 4.2 R4-2.R : 距離行列と近隣結合法による木の作成

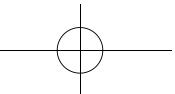
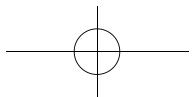
```
1 # サンプル数 Ns=5 が 5 次元のデータを持っているようなデータセットを作成
2 Ns<-5;k<-5;x <- matrix(rnorm(Ns*k), nrow=Ns)
3 dist(x,method="euclidean") # ユークリッド距離の距離行列
4 library(ape) # 木を作る関数 nj() を持つパッケージ ape の読み込み
5 treu<-nj(dist(x,method="euclidean"))
               # ape の近隣結合法関数 nj により木を作る
6 trman<-nj(dist(x,method="manhattan"))
               # マンハッタン距離で木を作る
7 par(mfcol=c(1,2)) # 画面を 1 行 2 列に分割
8 plot(treu);plot(trman) # 2 つの距離法で木の表示
9 par(mfcol=c(1,1)) # 画面分割を 1 行 1 列に戻す
```

※8 R のパッケージの取り込みは付録 A の A.2 節を参照。



同じデータを使っても距離の定義が異なると、作成される木の形状が異なることがわかります。

図 4.9 ユークリッド距離（左）とマンハッタン距離（右）による距離行列に基づく近隣結合法による木



# 第 5 章

## サンプルを個別に捉える

サンプルからデータをとて、データから意味を取り出すのが遺伝統計学の役割ですが、サンプルの取り扱いには大きく2つのやり方があります。

1つは、個々のサンプルを個別に扱う方法で、もう1つは、サンプルを集団として扱う方法です。第5章では個別に扱う方法についてを、第6章では集団として扱う方法についてを取り上げます。



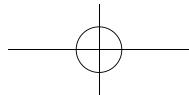
### 5.1 グラフとは

#### ● 5.1.1 グラフの定義

この章では、個々のサンプルを区別して扱います。区別して扱うために、個々のサンプルはそれぞれを点として扱います。サンプルを点で表して、それらの関係を表す方法の1つが**グラフ**です。すでに、この本でもいくつかのグラフの例が登場しています。

用語を整理する意味も込めて、グラフ理論の基礎事項をまとめます。

- 「頂点」と「辺」とで構成されたものをグラフと言います
- 頂点は点であり、辺は頂点と頂点を結んだものです
- 辺は2つの異なる頂点を結ぶ場合と、1つの頂点から出て同じ頂点に戻る場合とがあります



- 異なる2頂点を結ぶ辺のみを許せば、辺は頂点ペアの間の関係を表すことになります
- 辺に向きがあるとする場合と、向きがないとする場合があります
- 向きのあるグラフを有向グラフ、ないグラフを無向グラフと言います
- 辺は頂点に「接続」し、ある頂点に「接続」している辺の数をその頂点の「次数」と言います
- 辺で結ばれた頂点は互いに「隣」であると言います
- ある頂点からある頂点まで辺をたどるとき、両端の頂点と途中の頂点と辺とを合わせて「道」と言います
- 「道」がある点から元の点に戻ってくるとき、それは「サイクル」と言います
- グラフ上での2頂点間の距離は、2頂点を結ぶ最短の道の長さです
- 特徴的なグラフには名前がついています
- すべての頂点の間に辺があるグラフは、「完全グラフ」<sup>※1</sup>と呼ばれます
- すべての頂点同士に道があり、サイクルがないグラフは「木」と呼ばれます
- 関係をグラフにすることの利点の1つは、グラフを扱うアルゴリズムやグラフに関する証明済みの事項を利用できることです

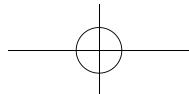


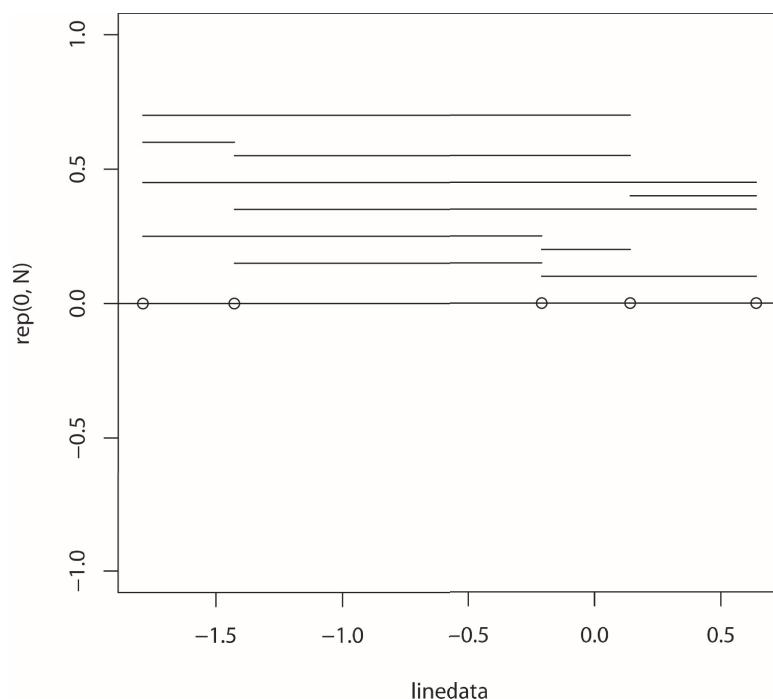
## 5.2 サンプルを並べる——数直線というグラフ

順序のある連続データ型の値を持つサンプルセットがあるとします。サンプルを値に応じて数直線上に並べます（図5.1の中央の水平線とその上の点）。この並べ方はどのように読み取ればよいでしょうか。

数直線上の点のすべての点のペアを取り出して、その長さが見えるように数直線の上部に並べたのが、図5.1の中央の数直線の上部に描かれた線分です。このようにしますと、点を乗せた数直線には、 $N \times N$ の関係が距離として埋め込まれていることがわかります。このことから、量的データを用いて定めたサンプル間距離を表したグラフは、直線状のグラフとして表せることがわかります。

※1　図4.4は完全グラフの例です。





中央の数直線上に 5 点あります。それらが作る  $5 \times 4/2=10$  個のペアの距離を表す線分が数直線上に示されています。この 10 ペアの情報を埋め込んだのが、この数直線状のグラフです。

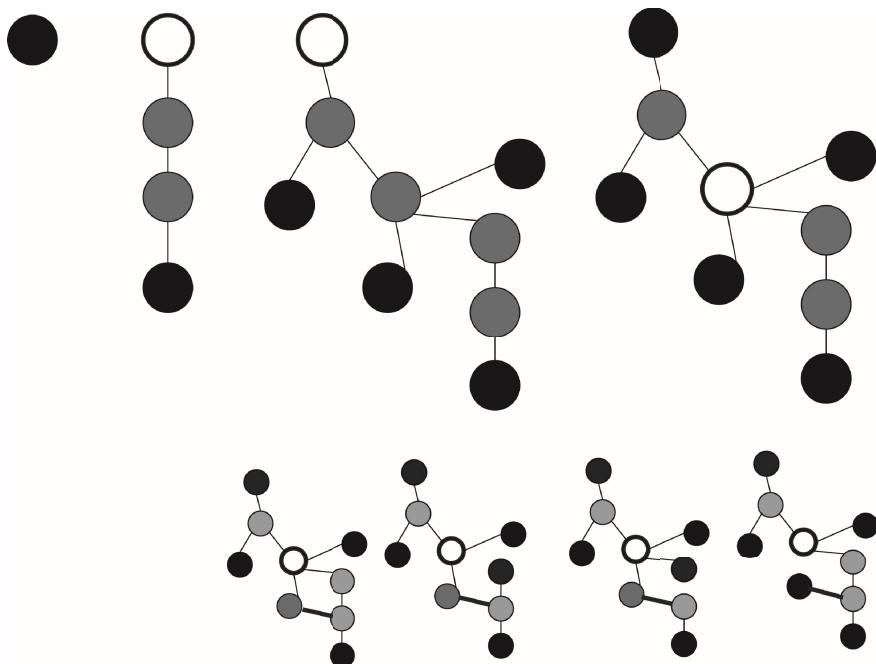
図 5.1 数直線

## 5.3 木というグラフ

### ● 5.3.1 木とは

前節の数直線の例もグラフには違いありませんが、ずいぶんと単純な形をしていました。今度は、グラフの中で**木**と呼ばれるタイプのものを扱います。

数直線も木の特別な場合です。図 5.2 を見てください。グラフで言うところの木は、植物の木と同様に、ある点から成長してできる形をとります。その成長にあたっては、先端を伸ばすか、枝分かれするかの 2 通りの方法があります。成長するときにできる辺の先は必ず新しい頂点であって、既存の頂点には接続しないので、木にはサイクルはできません。木の上の 2 点間に辺を渡すと、必ずサイクルができるので、木ではなくなります。そのサイクルを構成する 1 辺を取り去ると、再び木に戻ります。木の頂点の数と辺の数は



頂点1のグラフも木です。白丸が「根」、黒丸が「葉」、灰色丸がそれ以外の点です。上段の右2つのグラフは無向グラフとしては同じですが、「根」を取り替えて有向グラフとすると、グラフとして異なるものになります。下段の最左は、1本の辺を2頂点間に差し渡して、木でなくしたグラフです。生じたサイクルの辺を1本取り去ると、必ず木に戻ります。

図5.2 木

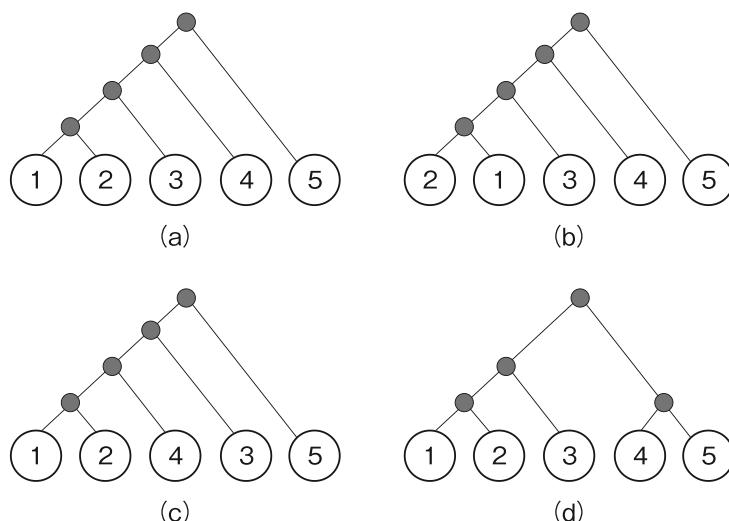
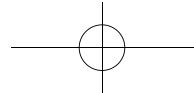
$$(T \text{ の辺の個数}) = (T \text{ の節点の個数}) - 1$$

の関係にあります。

木には「根 (root)」がある場合（「根付き木」）とない場合があります。根付き木は、木であって、その1つの頂点を特別扱いして、それを「根」と呼びます。根付き木は辺に向きがあり（有向）、辺の向きは「根」から離れる方向です。「根つき木」の先端には、次数1（接続している辺の数が1）の頂点（端末点）があり、それは「葉」と呼ばれます。

### ◎ 5.3.2 木の形状——トポロジー

木でデータを理解するためには、木の違いがわかる必要があります。木の違いは、枝分かれの具合と、分岐間の長さ・末梢の枝の長さで決まります。長さには意味を持たせずに枝分かれの具合だけに着目したとき、その形状を木のトポロジーと言います（図5.3）。



木としてのトポロジーと分岐木としてのトポロジーは違います。分岐木は階層的分類を表しています。その場合には、トーナメント方式の試合をしている場合を考えるとわかりやすいです。トーナメント戦のやり方が変わらなければ、分岐木のトポロジーは同じです。それが変われば、分岐木のトポロジーは異なります。(b) を (a) と較べると、1, 2 のラベルの付け方は変わりましたが、トーナメント戦の対戦順は同じです。(a) と (c) は、3, 4 のラベルを付け替えました。この場合は、トーナメント戦の対戦順が変わります。(d) は、見るからに形が変わっていて、トーナメント戦の対戦順は変わります。

図 5.3 分岐木としてのトポロジー

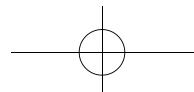


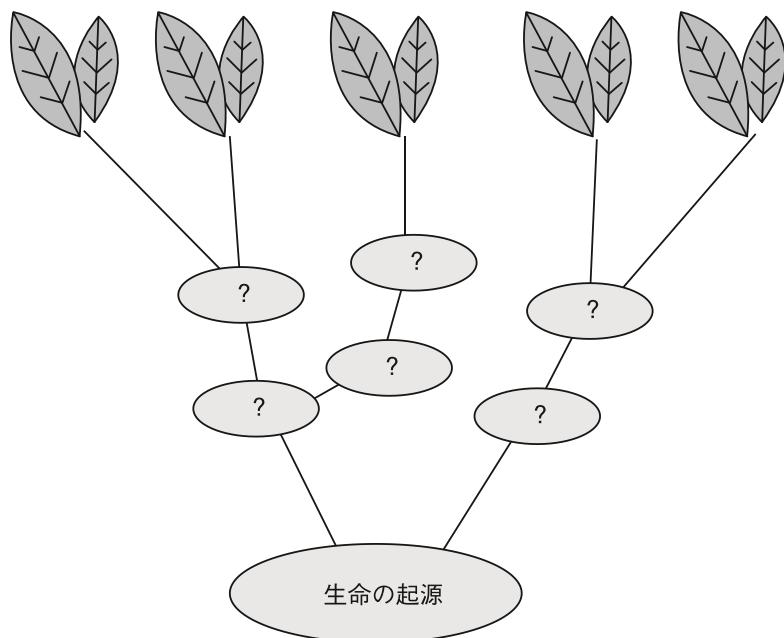
## 5.4 木構造でのデータの理解——階層的クラスタリング

### ● 5.4.1 進化系統樹

種の進化の過程を樹で表したものを**系統樹**と言います。系統樹は「根付き」の木グラフで、現在の種は「葉」にあたります。

複数サンプル（今の場合子種）の外見的特長・塩基配列など、何かしらで遠近関係があるときに、その関係を木として捉えようとした例です。木の構造を、階層的クラスタ構造とも言います。枝分かれがサンプルの分類（クラスタ化）をしており、その枝分かれが、根元から末梢に向かって段階的（階層的）に繰り返されているからです。**分岐図** (cladogram) とも呼ばれます。サンプルを分類することで理解しようとする手法全体を**クラスタ解析・クラスタリング**と言いますが、階層的クラスタリングは木構造を作ってサンプルを分類する方法の1つです。





進化系統樹は生命の共通祖先から、現在の種まで木状の関係があると考えて作られるものです。現在の種は系統樹の葉に相当します。過去には、共通の祖先がいるとみなしています。

図 5.4 進化系統樹

### ● 5.4.2 階層的クラスタリング

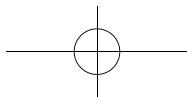
クラスタリング手法には、階層的クラスタリング以外に非階層的クラスタリングもあります。非階層的クラスタリングはサンプルを集団として扱うので、第6章で改めて取り上げます。

階層的クラスタリングは、サンプルペア間の遠近関係・類似関係から、木のトポロジーと辺の長さを答えとして出します。初めは、すべての要素を独立したクラスタとし、クラスタ同士をだんだんに合併することを繰り返すことで、最終的に1つにまとめ上げます。

この処理を実行するにあたって、原則として次の3つの決まりを定める必要があります。

- 遠近関係・類似関係の測り方
- 合併してできたクラスタと、その他のクラスタとの距離の決め方
- 合併の順序の規則

です。サンプル同士の遠近関係・類似関係の測り方については、対称的で量的



な関係であれば、距離でもよいですし、データベクトルのなす角に基づく値をとってもよいです。4.2節（距離、相関係数）の議論に基づいて選べばよいです。

個々のサンプル同士の間の遠近関係・類似関係は選びましたが、複数サンプルが作るクラスタと、他のクラスタとの間の関係を量として表すためには、クラスタ内のどの点の情報をどのように用いるのかを決める必要があります。その方法を説明したのが図5.5です。

図5.5に示す通り、生じたクラスタとその他のサンプルや他のクラスタとの間の遠近関係・類似関係を測るにあたり、クラスタの代表点を新たに定める場合と、定めない場合があります。定める場合には、その定め方の定義が必要です。本書の目的は、個々の手法の理解ではありませんから、細かい点は不要です。手法ごとにどういう点が違うのかをつかんでください。

(b) は**近隣結合法** (NJ法) を示しています。あくまでも、分岐図であることを意識した方法ですので、まとめる（結合する）2点の根元側に共通な点を定め、そこから2点が分岐したものとして新しい点を定めます。

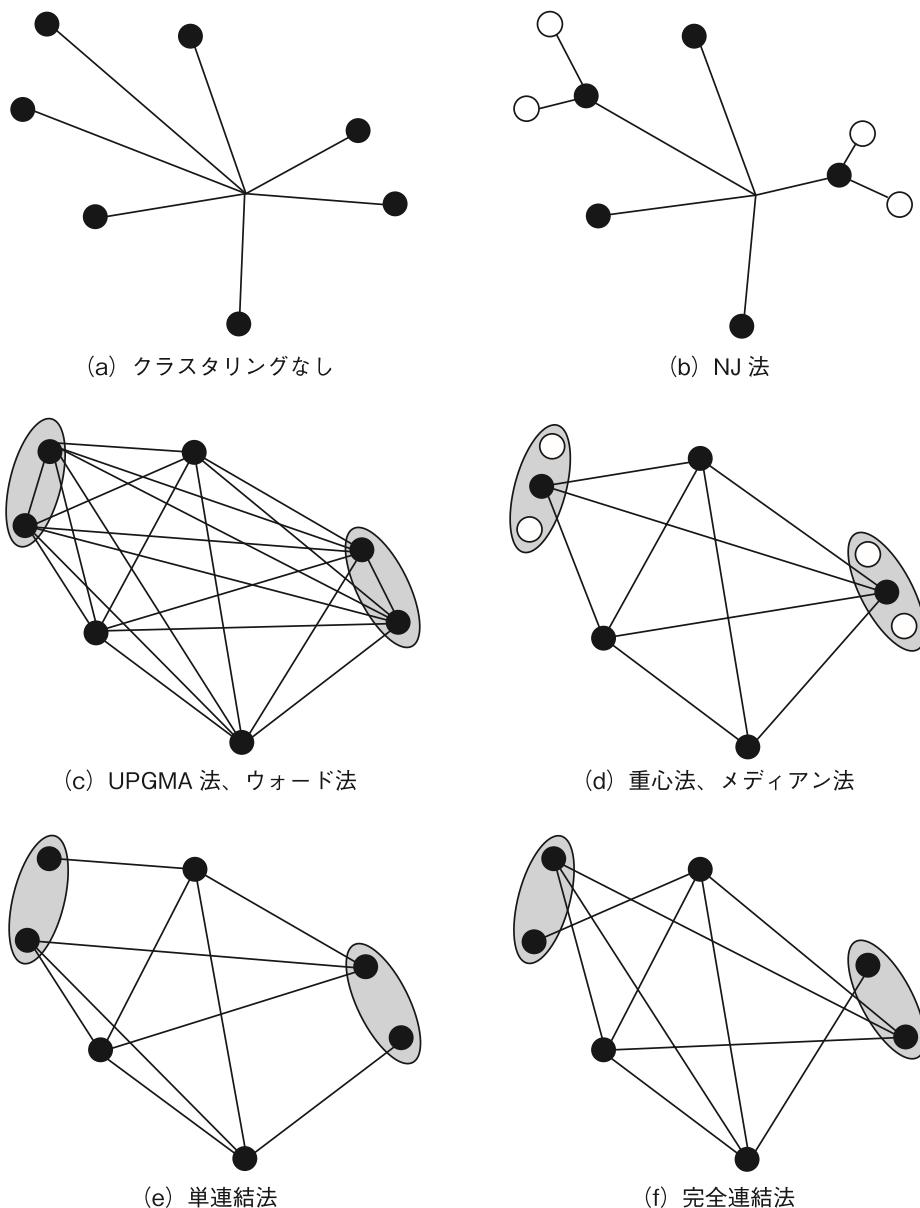
(d) は、根元に近いところに新点を置くのではなく、新たに生じたクラスタの「真ん中」にクラスタを代表する点を置きます (**重心法・メディアン法**)。この「真ん中」の定義もいくつかの定め方があります。

他方、新たな代表点を定めない場合には、クラスタが範囲を持った領域であるとみなします。このやり方では、クラスタを表す領域とクラスタ外部の点ないし領域との関係を測る方法を定める必要があります。(c) は、クラスタ化した後も、クラスタの構成要素のすべてが測定に意味を持つ方法です (**群平均法** (**UPGMA** (Unweighted Pair-Group Method using Arithmetic averages) 法)、**ウォード法**)。 (e) はクラスタとクラスタ外部の関係は最小関係を、(f) は最大関係をとるように定めます (**単連結法**、**完全連結法**)。

合併の順序のルールは、最も近いペア同士を合併していくのが素直な考え方なので、そのようにします。

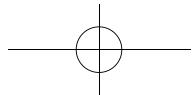
このように、距離・関係の定義がいろいろあり、クラスタ間の距離・関係を決める方法にもいろいろなやり方があるので、出来上がるクラスタもいろいろです。

第4章で、距離行列の扱いのときには、進化・集団遺伝学系のクラスタリング法であるNJ法を用いました。ここでは、データマイニング系のRの



- (a) は、7 個のサンプルを空間に配置しただけで、クラスタリングしていない状態です。その他の5つの図は、それぞれの方法でクラスタを進めている途中経過を示しています。
- (b) (NJ 法) では、クラスタの生成に伴って、クラスタを代表する点を中心寄りに定めています。クラスタとその他の点やクラスタとの関係は、この代表点によって決められます。クラスタリングされたサンプルが白丸で表され、新たに作られた代表点が黒丸で表されています。
- (d) (重心法、メディアン法) でも、クラスタを代表する点が生成されますが、それは、クラスタを構成する要素の分布の「真ん中」に選ばれます。中心寄りに選ぶ NJ 法とはここが違います。クラスタとその他の点やクラスタの関係は、この代表点によって決められます。
- その他の3つの図は、クラスタリングした後に、クラスタに新たな代表点を生成しない点で共通しています。クラスタとその他の点やクラスタの関係を定める方法の違いで名称が異なります。その説明は本文を参照してください。

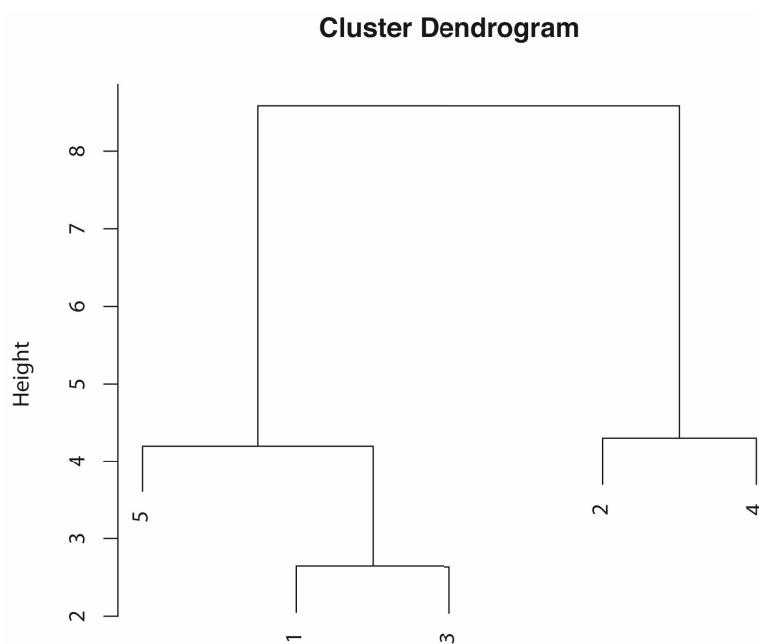
図 5.5 クラスタリングの方法



`hclust()` 関数を使ってみます<sup>※2</sup>(R5-1.R、図 5.6)。

#### R ソース 5.1 R5-1.R : クラスタリング

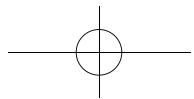
```
1 distMatrix<-dist(x,method="manhattan")
   # 距離にはマンハッタン距離を使用
2 trclust<-hclust(distMatrix,method="ward")
   # クラスタ間距離の定義にはウォード法を使用
3 plot(trclust)
```



dman  
hclust(\*, "complete")

図 5.6 クラスタリング結果

※2 `hclust()` の R のヘルプを見ることにより、複数の手法 "ward", "single", "complete", "average", "mcquitty", "median", "centroid" が利用可能であることがわかります。R のソースを確認すれば、その定義が確認できます。



## 5.5 行列データを眺める

### ○ 5.5.1 要素を並び替えて眺める——ヒートマップ

$N \times N$  の関係を階層的クラスタリングにて木にまとめ上げましたが、 $N$  個のサンプルの間の関係が複数 ( $M$  個) の項目によって決まるときに、項目の方の関係にも興味があったらどうしたらよいでしょうか？ 両方で階層的クラスタリングをして、 $N \times M$  個のデータを表示させてみればよいでしょう。

図 5.7 は、**ヒートマップ** という図示方法です。データは  $6 \times 20$  の長方形型の行列です (2 アレル型多型が 6 箇所あり、20 本の染色体のハプロタイプを想定しています)。R5-2.R のようにしてデータを作り、ヒートマップを作る `heatmap()` 関数に処理させています。

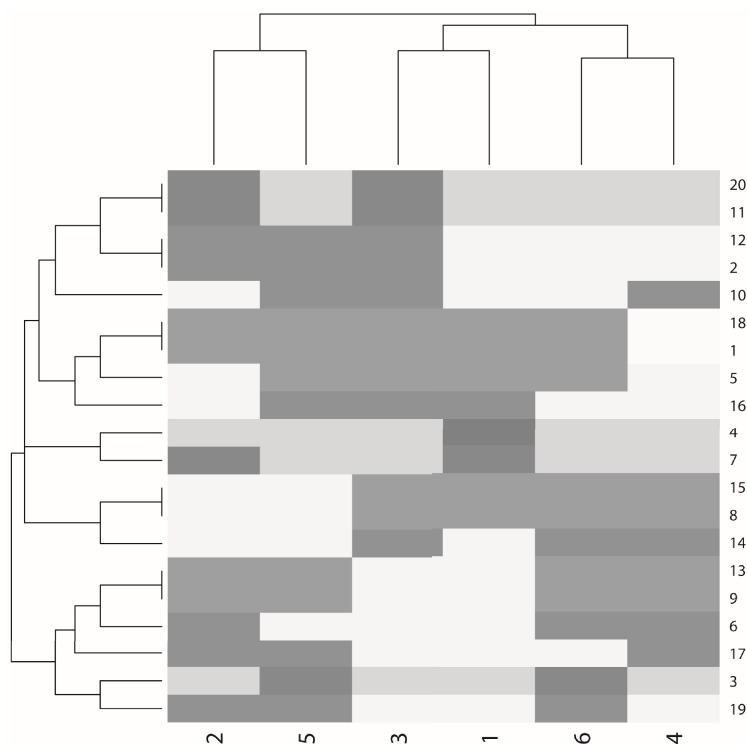
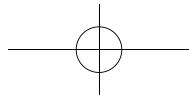


図 5.7 6 多型・20 染色体の多型データのヒートマップ



### R ソース 5.2 R5-2.R : ヒートマップ

```
1 m<-matrix(rbinom(120,1,0.5),20,6)
2 heatmap(m)
```

水平軸では 6 個の多型がクラスタリングされ、垂直軸では 20 本の染色体がクラスタリングされています。 $6 \times 20$  個のデータはその値に応じて濃淡がついています<sup>※3</sup>。

### ● 5.5.2 要素を並び替えずに眺める——連鎖不平衡係数プロット

前項のヒートマップでは  $N \times M$  行列の情報を、 $N$  側と  $M$  側の両方でクラスタリングし、 $N \times M$  をカラースケールで視覚的に表現しました。クラスタリングの結果、要素の並び順が変更されました。

要素の並び順に意味があるときは、並び順を変えずに表示します。多型マークーが染色体上に並んでいるときに、マークー間の  $N \times N$  の関係を表示してみます。マークー間の関係として連鎖不平衡係数を使って、図示することにします。連鎖不平衡の係数のうち  $r$  と呼ばれるものは相関係数でしたので、R5-3.R の処理により図 5.8 のように表示することができます。

実際には、相関係数が 1 のときと -1 のときは、片方の多型のアレルの "0", "1" のラベル付けを取り替えれば同じことです。 $r^2$  を連鎖不平衡係数としてよく使いますので、図 5.8 もそのようになっています。これは、連鎖不平衡の視覚的表現で最もよく使われるプロットです。

### R ソース 5.3 R5-3.R : 連鎖不平衡マップ

```
1 cormatrix<-cor(m);rsqmatrix<-cormatrix^2
2 image(1:nrow(rsqmatrix),1:ncol(rsqmatrix),rsqmatrix,
       col=gray((100:0)/100))
```

※3 `heatmap()` 関数は、内部で、距離行列を作る部分に `dist()` 関数を、クラスタリングに `hclust()` 関数を用いています。R のソースを確認するとわかります。

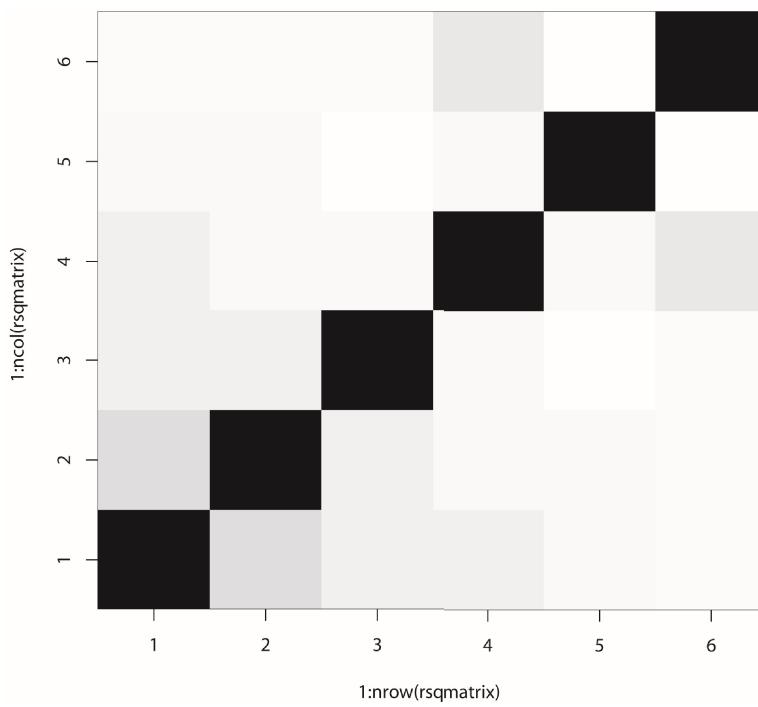
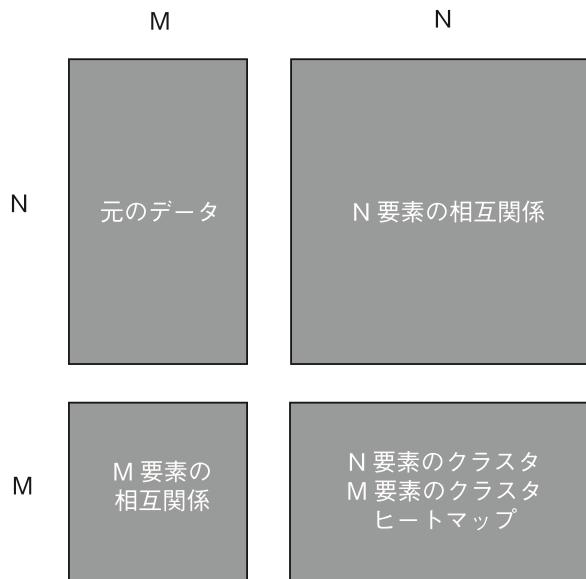
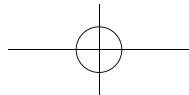


図 5.8 6 多型の連鎖不平衡マップ

### ◎ 5.5.3 片方の軸に着目、両方の軸に着目

もともとのデータが  $N \times M$  の形をしていたときにクラスタリングをすれば、要素の関係の強弱により、要素の順序が変わります。行についてクラスタリングを行ったり、列についてクラスタリングを行ったり、行・列の両方についてクラスタリングを行ったり（ヒートマップ）することができました。

要素順を変更せずに、 $N \times N$  の関係表示をすることもできました（連鎖不平衡係数プロット）。 $N \times N$  とは逆に  $M \times M$  で相関をとって図示することもできます。これらの関係は図 5.9 のようになっています。



2次元データそのものは左上。行の  $N$  個の要素について処理すれば、右上になります。要素の順序を変えればクラスタリングですし、変えなければ、相関パターンの図示になります。列の  $M$  個の要素についても同様です（左下）。両方でクラスタリングをするとヒートマップ表示になります。

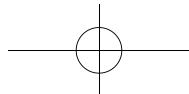
図 5.9  $N \times M$  の形のデータ

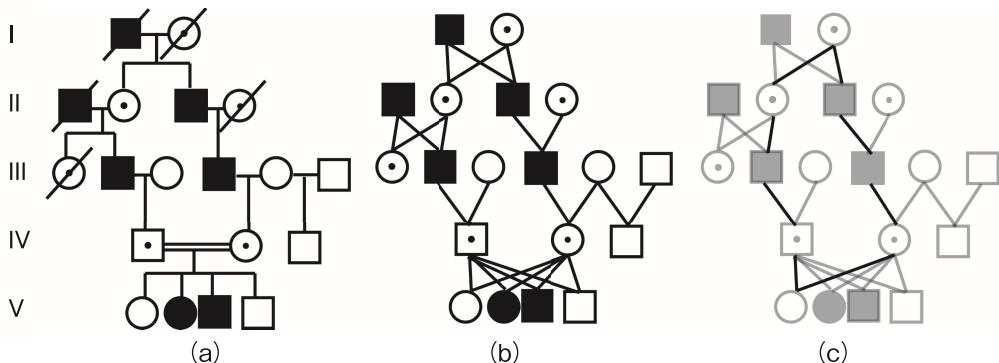
## 5.6 個体の家系図、アレルの系図——同一種の中のグラフ

親子・血縁関係を図で表した**家系図**というものがあります（図 5.10）。遺伝学では血縁関係は非常に重要ですから、家系図も大変重要です。

図 5.10 は点（丸と四角）が線で結ばれていますが、すべての線が点と点とで結ばれているわけではないのでグラフではありません。また点にフェノタイプ（発病・非発病、死亡・生存）の情報が付随し、夫婦間の線に近親婚か否かの情報があるなど、複雑な情報を搭載している点も、データ解析的には取り扱いを難しくしています。家系図をグラフとして扱うためにはどのようにすればよいでしょう。

図 5.10 (b) が血縁関係をグラフ化した図です。遺伝因子の伝達関係を表すためのグラフとしたので、夫婦の関係を表す水平線は消え、子を表す頂点を介して間接的に夫婦の関係が示されるようになりました。近親婚を表す2重線も消えました。近親婚であることは、近親婚で生まれた子から出発すると、グラ





左のⅠ、Ⅱ、Ⅲ、Ⅳ、Ⅴは世代番号です。四角は男性、丸は女性を表します。黒塗りは発病者、白抜きは非発病者です。白抜きのうち、中央に点がある場合は、遺伝因子を有する人（保因者）です。男女が水平線で結ばれているときには、生物学的な父母であることを示しています。男女の水平線から垂直に下りた線の先に子が続きます。子が複数あるときには、その線に水平線が接続し、水平線から垂直線が枝分かれしてその先に子が接続します。両親を同じくする子は、このように水平線からの枝分かれで示されます。

第Ⅲ世代の右から2番目の女性は2人の男性と水平線で接続し、それぞれの関係から子を持っています。この場合、第Ⅳ世代の右端と中央の2人は半同胞です。第Ⅳ世代の夫婦は二重線です。これは、近親婚関係を表しています。斜めの線は死亡を表しています。

個人を頂点に、親子関係を辺に変えたのが（b）です。見やすくするために死亡の斜線は省略しています。（c）は、サイクルの1つを強調した図です。

図 5.10 常染色体性劣性遺伝病の家系図の例

フをぐるりと回って、本人に戻ってこられることからわかります。なお、辺に向きをつけていませんが、遺伝子の伝達は親から子へ伝わりますから、伝達関係を考慮すれば有向グラフです。木ではありません。

### ◎ 5.6.1 個人の関係のグラフと染色体の関係のグラフ

#### （1）個人の関係のグラフ——家系図

図 5.11（a）は常染色体劣性遺伝病の家系図です。それをグラフにしたのが図 5.11（b）です。

図 5.12 は個人が持つ染色体を三角で表して、その伝達関係をグラフにしたものです。子に伝わる染色体は、必ず親の染色体セットの両方から一部分ずつを受け継ぎます。したがって、すべての染色体は2本の辺を親の染色体から受け継ぎます。染色体数は個体数の2倍あるので、その分だけグラフが混雑していますが、ある染色体に着目して祖先の方向に向かってたどると、世代ごとに祖先染色体が2倍になる2分岐木であることがわかります。

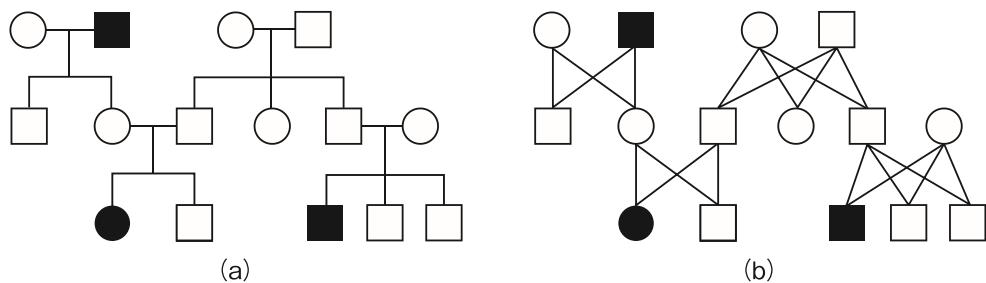
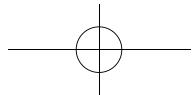


図 5.11 常染色体性劣性遺伝形式の形質の家系図とそのグラフ化表現

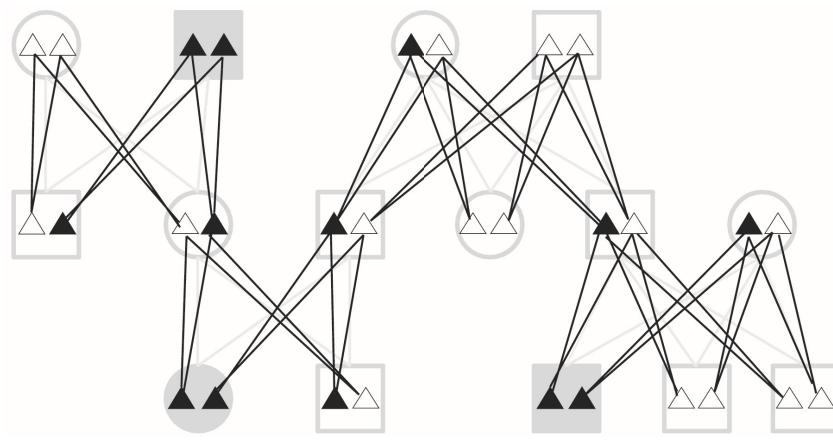


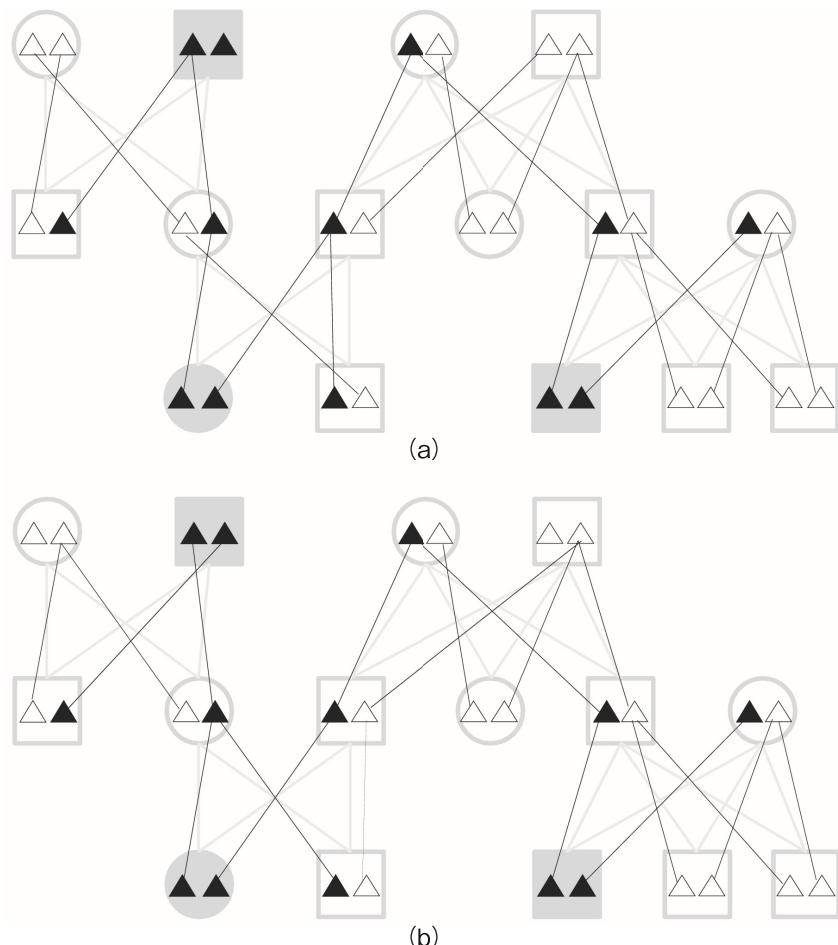
図 5.12 染色体の伝達グラフ

## (2) 個人の関係のグラフは複数の染色体の関係のグラフを含む

ある親の2倍体染色体から、精子もしくは卵子の1倍体染色体が取り出される過程を観測することはできません。したがって、その取り出しパターンは、推測するしかありません。

ある家系において、ジェノタイプを調べたとします。そのジェノタイプを満足する、染色体の伝達グラフのパターンは1つとは限りません。

2アレル型多型の例を示します。図5.13を見てください。2アレル型の多型のジェノタイプを観察したときに、(a)と(b)と、どちらも白黒の三角パターン（ジェノタイプパターン）は同じですが、辺の結び方（伝達パターン）が違います。遺伝因子解析では、ジェノタイプデータに基づいて、アレルがどのように伝達してきたかを推定しますが、なぜ推定しなければいけないのかをこの図は示しています。



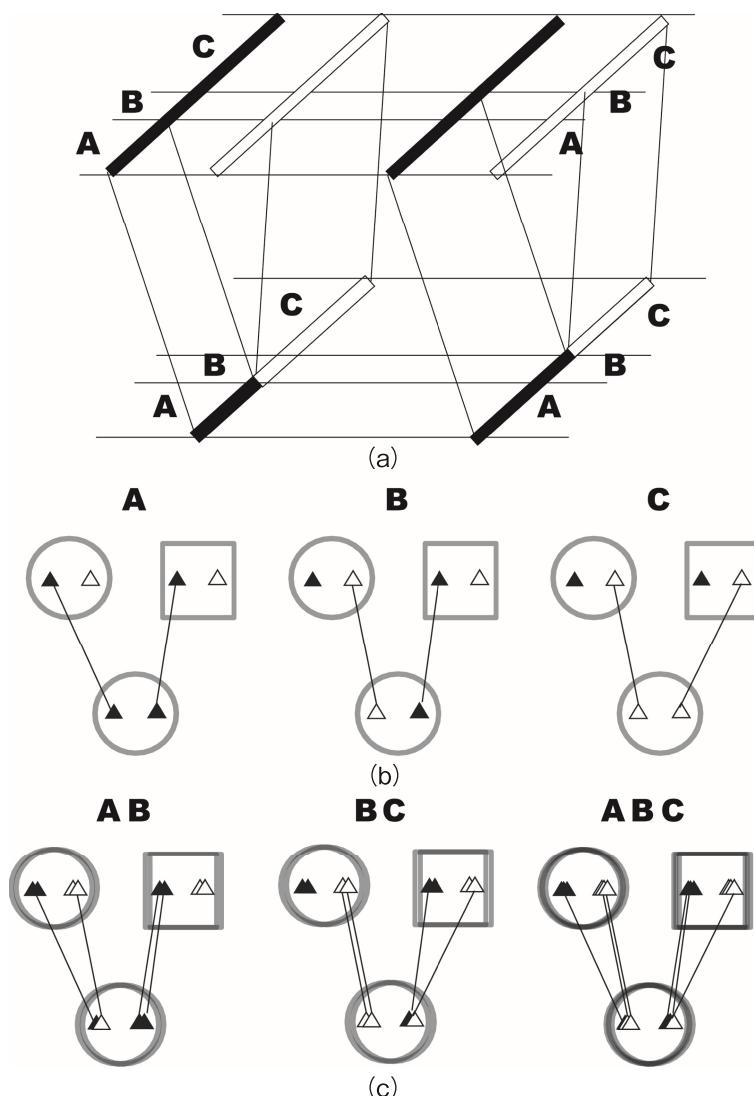
第Ⅱ世代の左から3番目の個体が受け取る白△の染色体が、(b)では同胞と共有されているのに対して、(a)では共有されていません。また、第Ⅲ世代の左から2番目の個体の黒▲の染色体は、(a)では父親から受け取っていますが、(b)では母親から受け取っています。

図 5.13 2アレル型多型の伝達グラフ

### ● 5.6.2 染色体の伝達グラフと組み換え

染色体の伝達グラフを、染色体の場所ごとに分解してみます。

図 5.14 (a) は、2組の染色体対（黒と白）から交叉が1箇所で起きて、組み換え染色体ができる様子を示しています。染色体を、3つの部分 A, B, C に分けて考えます。それぞれの部分での染色体の伝達の様子を (b) に示します。A, B, C に分けてグラフを描くと、子染色体と親染色体との関係は1対1で、枝分かれがありません。逆に、交叉・組み換えが起きた点をまたいで伝達グラフを描くと、(c) にあるように、枝分かれが生じます。交叉・組み換えが

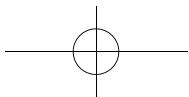


- (a) 6本の白黒棒が描かれています。それぞれが染色体で、上側の4本は親染色体で、2本ずつのペアになって、下側の子染色体をそれぞれ作ります。左側のペアは、AとBの間で交叉・組み換えを起こしているために、子染色体はA部分が黒、B,C部分が白です。右側のペアはBとCの間で交叉・組み換えを起こしています。
- (b) 染色体のA,B,Cのそれぞれの部分での、染色体の伝達をグラフにしたものです。アレルの伝達と同様で、親染色体と子染色体は1対1対応になっています。
- (c) (b) のAとB、BとC、AとBとCとを、重ね合わせた図です。

図 5.14 染色体の伝達グラフ

起きていらない範囲だけを取り出すと、何世代さかのぼっても、枝分かれのない一本道の伝達関係になります。

(c) ではAとB、BとC、AとBとCのすべてを重ね合わせました。このように重ね合わせると、染色体のグラフが分岐木になります。なお、長さ1塩



基の線分の内部で交叉・組み換えは起きませんので、塩基に関する伝達グラフは必ず枝分かれのない一本道になります。

### ● 5.6.3 祖先にさかのぼる——コアレント

染色体の伝達グラフをより多くの世代にわたって考えてみることにします。染色体の集団があって、交叉・組み換えをしつつ、次世代の染色体を作る過程をシミュレーションしてみます。

図5.15は、8本の染色体で10世代の経過を見ています。縦軸が世代経過を表しています。各世代にある8個の点が染色体を表しています。(a)は、ある特定の塩基についての伝達グラフです。1塩基の伝達グラフなので一本道です。最も若い世代(最下端の世代)の8本の染色体の親をたどると、一番上の世代では、2本の染色体に行き着いていることがわかります。現在の染色体の祖先染色体をさかのぼるとだんだんに同じ染色体に行き着くわけですが、これを**コアレント**(合体・統合)と呼びます。

(b)のグラフは、(a)のグラフの塩基の近傍の塩基での伝達の様子です。木の形はよく似ていますが、(a)の方が(b)よりも濃い木が大きいことがわかります。この2つのグラフを重ね合わせたのが、(c)のグラフです。第6世代の左から2番目の染色体だけが、上から2本の辺を受けています。祖先に向かって、一本道ではなくなり、分岐が生じています。これは、2つの塩基の間で交叉・組み換えが起きたことを示しています。

(d)のグラフでは、薄い色の木の途中に変異を表す★印が描き込まれ、その変異を引き継ぐ部分が太い木として描いてあります。この木はこの★印が「根」になります。

このように、特定の塩基に着目すると、複数の木が独立に存在して、根から枝を介する部分は過去にあり、現在は、その末端である葉として現れています。この葉を2つ持っているのが個人です。すべての塩基について、伝達の木を描くことができて、隣合う塩基の木々のパターンはほぼ同じで、交叉・組み換えが起きたときにのみ、パターンが変わります。塩基ごとの伝達グラフはゲノム全体にわたって存在しており、そのすべての木を重ね合わせたものが、塩基配列情報の現在と過去とその伝達に関する情報のすべてです。

個人はフェノタイプを持っています。したがって、ある変異がフェノタイプ

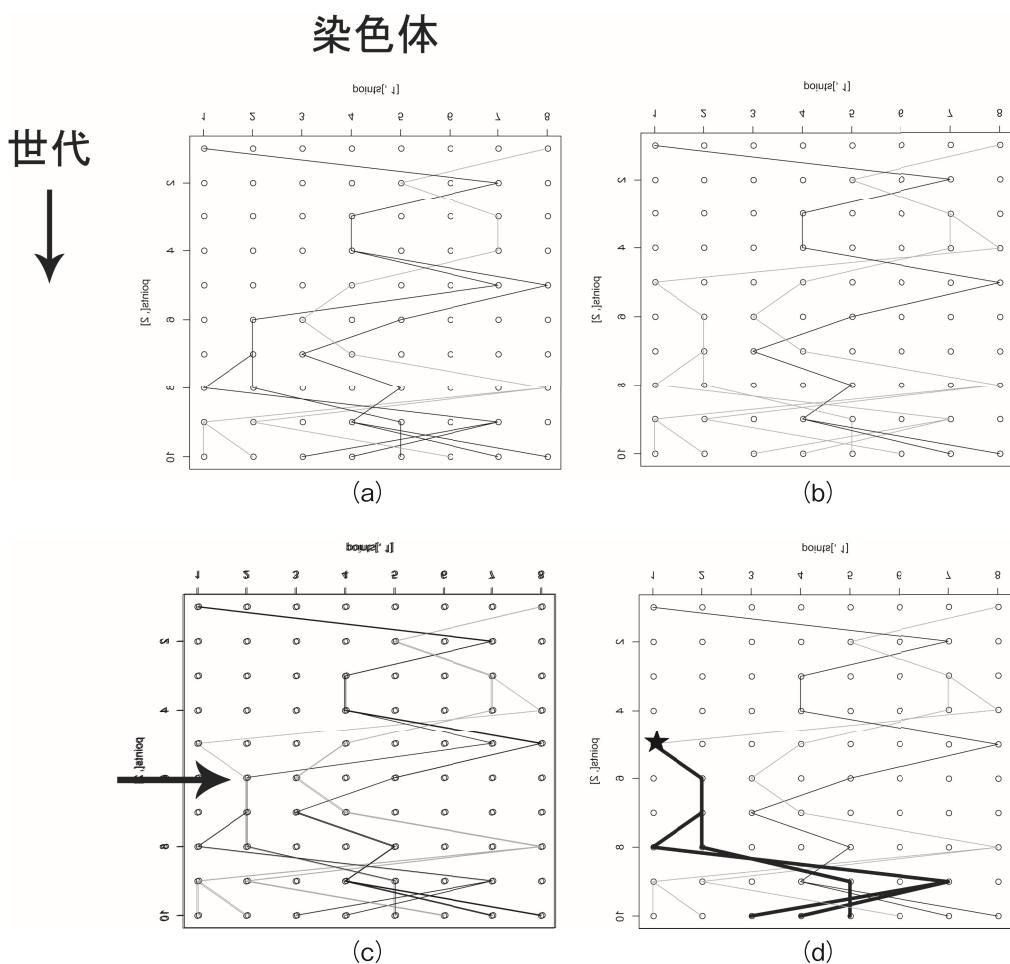
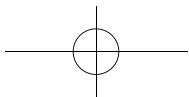


図 5.15 コアレント

と関係していることを調べる作業は、この過去にさかのぼるたくさんの木々のパターンのうちのどれが、葉に現れているフェノタイプのパターンとよく合致するかを判断することと同じです。この点は、家系サンプルを用いた連鎖解析でも、集団からのサンプルを用いた関連解析でも同じです。

この図を描く R のソースは掲載しませんが、ダウンロードできます (R5-sup1.R)。



## 5.7 ネットワーク

個々のサンプルを個別に取り扱う方法としてのグラフに関する最後のトピックとして、**ネットワーク**を取り上げます。木はサイクルを持たないグラフですが、サイクルを持ったグラフを活用することもできます。サイクルを持った有向グラフでそこに何かしらの流れを想定するとき、それはネットワークと呼ばれます。

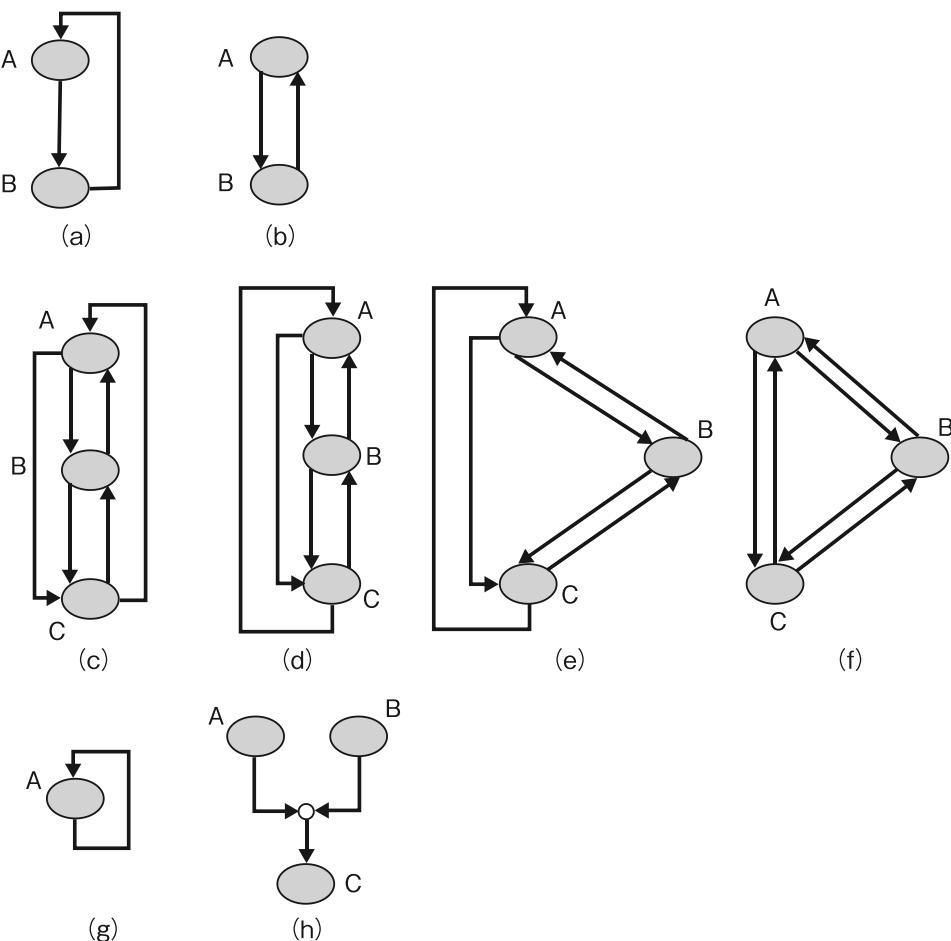
分子としてのDNA、RNA、タンパク質、その他の因子は、同時に存在して相互に影響を及ぼし合うことができます。これらの関係を表現しようとすると、**サイクル**が生じます。AがBを促し、BがAを促すという関係は**ポジティブフィードバック**と呼ばれ、AがBを促し、BがAを抑制するという関係は**ネガティブフィードバック**と呼ばれ、どちらも単純に表せば、図5.16(a) (b) のようになります。要素を増やしてA、B、Cの3要素にしてみます。CがAに影響を与える様子を「作用している」印象を強めて図にしたのが図5.16の(c)～(f)です。

少しずつ配置を変えていくと、3頂点の間に方向の異なる2つの辺を引いた状態という意味では、変わりがないことがわかります。

要素間のネットワークの解析では、図5.16(h)のような関係も登場します。2つの要素を組み合わせたときに初めて、第3の要素への効果が生じる、という関係です。このような場合には、「2要素の組み合わせ」に対応した点をグラフに付け加えることになります。遺伝形式を取り扱うときに列を追加する必要があったことと同じことです(図3.7参照)。

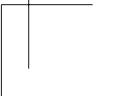
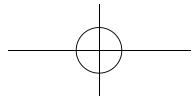
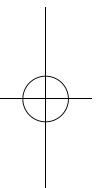
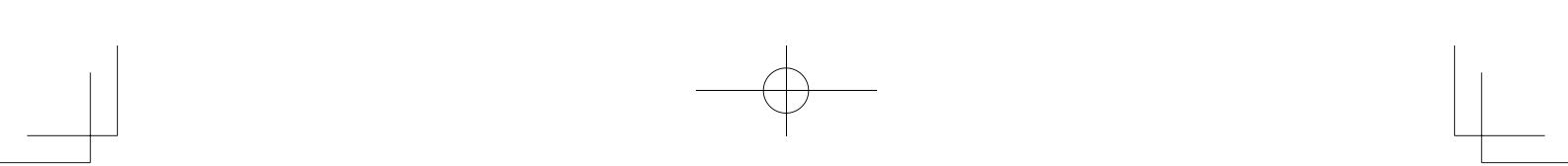
要素の組み合わせを考えるときには、組み合わせの数が要素数に応じて猛烈な勢いで増えしていくためにグラフが巨大となり、全部を数え上げることは現実的でなくなることが多いです。

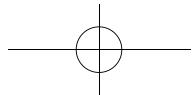
また、サイクルのあるグラフは無向にしろ有向にしろ、読み取りが難しくなります。どこから見始めて、どのように見終えたらよいかがわかりにくいくらいです。ですから、サイクルのあるグラフを理解するときには、そのグラフを特徴づけたり、説明したりする指標を取り出すという作業が発生します。



(a) がポジティブフィードバック、(b) がネガティブフィードバック、(c) ~ (f) は 3 点間の相互関係を点の配置を換えて描いたもの、(g) は自身へのフィードバック、(h) は 2 つの要素の作用に相互の影響がある場合を示しています。

図 5.16 フィードバックとグラフ





# 第 6 章

## サンプルを集団として捉える

第5章では個々のサンプルを区別しました。今度は、サンプルの集まりに着目します。

### 6.1 分布として捉える

#### ● 6.1.1 1次元

複数のサンプルについて1つの量的データ型のデータがあるものとします。そのデータの集まりの特徴を知るために、**箱ひげ図**と**密度分布**と**累積密度分布**を描いてみます（R6-1.R）。

R ソース 6.1 R6-1.R : 記述統計

```
1 n1<-1000;n2<-500 # サンプル数
2 # 1峰性サンプル作成
3 popdata1<-rnorm(n1,0,0.5) # 正規分布からの乱数発生
4 par(mfcol=c(1,3)) # 画面を1x3に分割
5 boxplot(popdata1) # 箱ひげ図
6 plot(ecdf(popdata1)) # 標本の累積分布
7 plot(density(popdata1)) # 密度分布
8 par(mfcol=c(1,1))
9 summary(popdata1) # データの基本統計
10 #2 峰性サンプル作成
```

