バイオスタティスティクス基礎論
第 4 回　講義テキスト

岩田洋佳
aiwata@mail.ecc.u-tokyo.ac.jp

＜階層的クラスタ解析＞

多数の対象について、それらのもつ多次元の特徴をもとに「似たもの」どうしをグループ（クラスタ cluster）に分類すると便利なことがあります。例えば、DNA 多型のデータに基づき遺伝資源に含まれる品種や系統をグループ分けできれば、遺伝資源のもつ形質の変異について、グループの情報をもとに整理し、体系化することができます。

前回の講義でもお話ししたように、多数のサンプルがもつ多数の特徴の変異について、データを眺めるだけで把握するのは困難です。主成分分析では、多数の特徴を低次元の変数で表現することでデータのもつ変異の要約を試みました。クラスタ解析では、多数のデータを少数のグループにまとめることで、データのもつ変異の要約を試みます。今回の講義では、まず、多数のデータを階層的にグループに分類する階層的クラスタ解析について概説します。

今回の講義では、前回までと同様にイネのデータ（Zhao et al. 2011, Nature Communications 2:467）を用いて説明を進めていきます。今回の講義では、品種・系統データ（RiceDiversityLine.csv）、表現型データ（RiceDiversityPheno.csv）、マーカー遺伝子型データ（RiceDiversityGeno.csv）の 3 つのデータを用います。いずれも、Rice Diversity の web ページ http://www.ricediversity.org/からダウンロードして整理したデータです。前回も説明したようにマーカー遺伝子型データは、ソフトウエア fastPHASE（Scheet and Stephens 2006, Am J Hum Genet 78: 629）を用いて欠測値の補間を行ってあります。

まずは、前回と同様に、3 種類のデータを読み込んで、それらを結合してみましょう。

```
> line <- read.csv("RiceDiversityLine.csv")
                          # 系統データを line として読み込む
> pheno <- read.csv("RiceDiversityPheno.csv")
                          # 形質データを pheno として読み込む
> geno <- read.csv("RiceDiversityGeno.csv")
                          # 遺伝子型データを geno として読み込む
> line.pheno <- merge(line, pheno, by.x = "NSFTV.ID", by.y = "NSFTVID")
        # line の NSFTV.ID と pheno の NSFTVID が一致するようにデータを結合
> alldata <- merge(line.pheno, geno, by.x = "NSFTV.ID", by.y = "NSFTVID")
        # line と pheno を結合したデータにさらに geno を結合
> rownames(alldata) <- alldata$NSFTV.ID
        #  alldata の行の名前を品種・系統の ID で置き換える
```

最初に、DNA マーカー（1,311 SNPs）に見られた変異に基づいて、374 品種・系統をクラスタに分類してみましょう。まずは、そのためのデータを準備します。

```
> data.mk <- alldata[, 50:ncol(alldata)]
        # alldata の 50 列目から最後までがマーカーデータ。関数 ncol は列数を返す
> subpop <- alldata$Sub.population
        #  分集団データも抜き出して subpop に代入しておく
> dim(data.mk)              #  データのサイズを表示する
[1]  374 1311               #  374 行×1311 列のデータ
```

クラスタ解析には様々な方法がありますが、ここではまず 1 つの方法でクラスタ解析を行ってみます。

まずは、DNA マーカーのデータをもとに、品種・系統間の距離を計算します。

```
> d <- dist(data.mk)        #  374 品種・系統の全組合せ間でユークリッド距離を計算
> head(d)                   #  距離の計算結果は、行列ではないため、うまく表示されない
[1] 54.47141 53.08033 44.70547 52.82571 45.40700 44.36904
> head(as.matrix(d))[,1:6]          #  最初の 6 品種・系統間の距離の表示
        1       3       4       5       6       7
1  0.00000 54.47141 53.08033 44.70547 52.82571 45.40700
3 54.47141  0.00000 37.53194 46.79940 37.68502 49.82169
4 53.08033 37.53194  0.00000 44.38481 17.58133 46.49073
5 44.70547 46.79940 44.38481  0.00000 43.85254 42.87989
6 52.82571 37.68502 17.58133 43.85254  0.00000 46.69070
7 45.40700 49.82169 46.49073 42.87989 46.69070  0.00000
```

なお、関数 dist の返す値は行列（matrix）形式でなく、距離行列特有の形式になっていることに注意して下さい。したがって、例えば、最初の 6 品種間の総

当たりの距離を 6×6 行列で表示したい場合には、上記のように関数 as.matrix で距離行列特有の形式から matrix 形式に変換する必要があります。

では、クラスタ解析を行ってみましょう。

```
> tre <- hclust(d)          #  関数 hclust で階層的クラスタリングを行う
> tre                       #  結果の表示

Call:
hclust(d = d)

Cluster method   : complete
Distance         : euclidean
Number of objects: 374
```

Call には、回帰分析などと同様に、実行したコマンドがそのまま表示されます。また、Cluster method にはクラスタ解析の方法（クラスタ間の距離の定義）、Distance には距離の計算法が表示されます。また、Number of objects は分類を行った対象（ここでは、品種・系統）の数です。

クラスタ解析の結果を樹形図（dendrogram）で表示してみましょう。

```
> plot(tre)                 #  単に hclust の結果を関数 plot に入力するだけ
```

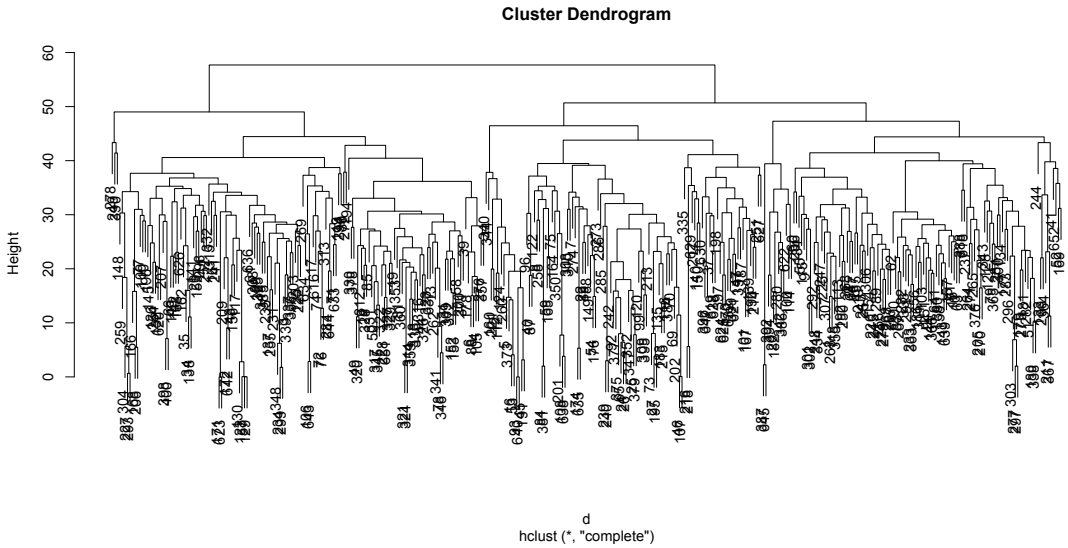

図 1. マーカー遺伝子型データをもとに得られた 374 品種・系統の樹形図

図 1 は関数 hclust で得られた結果をそのまま樹形図にしたものです。パッケージ ape を用いると、様々な表現様式で樹形図を描くことができます。そのためには、まず関数 hclust で得られた結果をパッケージ ape で定義されている phylo とよばれるクラスに変換する必要があります。

```
> require(ape)                          #  パッケージ ape を読み込む
> phy <- as.phylo(tre)                  #  関数 hclust の結果をクラス phylo に変換
```

では、phylo クラスに変換された結果をプロットしてみましょう。

```
> plot(phy)                             #  クラス phylo に変換したものを plot する
```

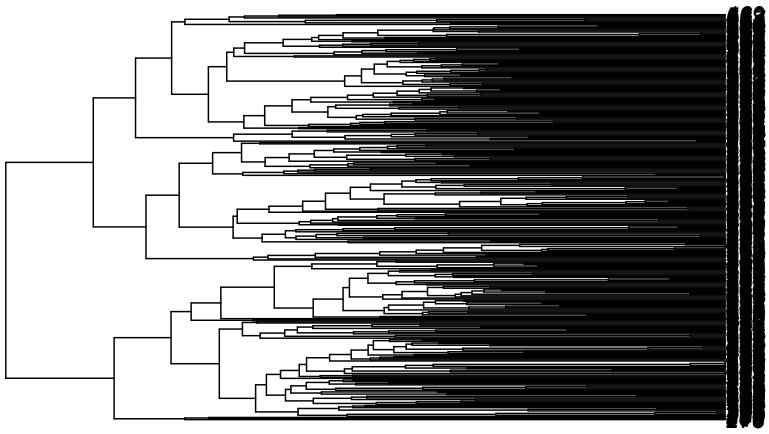

図 2. パッケージ ape の phylo クラスに変換して描いた樹形図

図 2 は、品種・系統数が多いこともあり、非常に見にくい図になっています。各品種・系統の遺伝的背景（所属する分集団）と樹形図での位置の関係を枝の色で確認できるようにして、少し見やすい図に描き換えてみましょう。

```
> phy$edge                    #  phylo クラスの edge 内に枝の結合関係が記述されている
（結果は省略）
> subpop[phy$edge[,2]]        #  phy$edge の 2 列目が枝の下流側の ID を表す
                              #  これを利用して末端の枝と分集団の関係を紐付ける
                              #  その枝が末端の枝で無い場合には<NA>になる
（結果は省略）
> col <- as.numeric(subpop[phy$edge[,2]])
                              #  分集団を数値に変換して、色コードとする
> edge.col <- ifelse(is.na(col), "gray", col)
                              #  NA になっている枝の色を、灰色“gray”に変換する
> plot(phy, edge.color = edge.col, show.tip.label = F)
                              #  オプション edge.color で枝の色を指定する
          #  オプション show.tip.label を False にすると末端のラベルが省略される
```

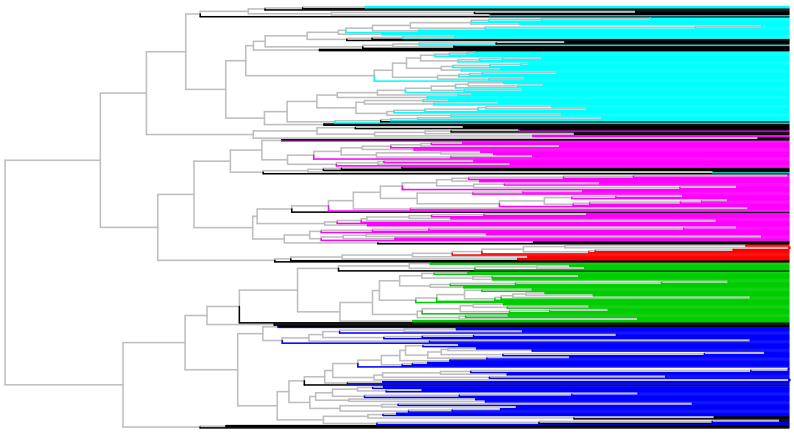

図 3. 品種・系統の所属する分集団毎に色付けした樹形図

図 3 を見ると、同じ分集団に含まれる品種・系統が同じクラスタに含まれる傾向が確認でき、品種・系統のもつ遺伝的背景の違いがクラスタ解析の結果によく反映していることが分かります。

パッケージ ape の phylo クラスは、様々な表現の仕方で樹形図を描くことができます。異なるタイプの樹形図を試してみましょう。

```
> pdf("fig4.pdf", width = 10, height = 10)
         #  グラフが大きくグラフウィンドウでは確認しにくいので pdf ファイルとして出力する
> op <- par(mfrow = c(2, 2), mar = rep(0, 4))
                  #  2 行 2 列でグラフを配置, mar は余白の設定。4 方向とも 0。
> plot(phy, edge.color = edge.col, type = "phylogram", show.tip.label = F)
                                    #  デフォルトの様式
> plot(phy, edge.color = edge.col, type = "cladogram", show.tip.label = F)
> plot(phy, edge.color = edge.col, type = "fan", show.tip.label = F)
> plot(phy, edge.color = edge.col, type = "unrooted", show.tip.label = F)
> par(op)                           #  グラフパラメータを元に戻す
> dev.off()                         #  pdf ファイルを閉じる（忘れないこと！）
```

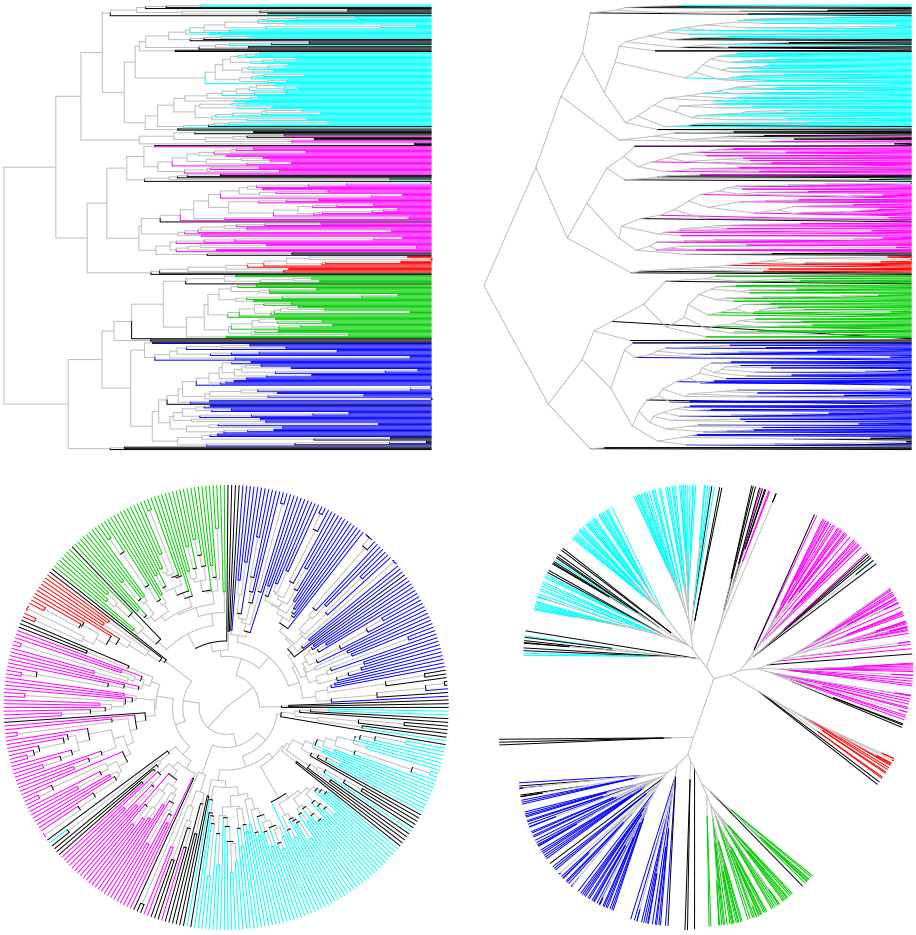

図 4. パッケージ ape を用いて描いた様々な様式の樹形図

図 4 は、同じクラスタ解析の結果を異なる様式の樹形図で描いたものです。様式が異なると受ける印象も分かりやすさも異なります。品種・系統の遺伝的関係を大域的に把握したい場合には、4 番目の“unrooted”タイプの樹形図が最も目的に適っているのではないかと思われます。

クラスタ解析の結果についてパッケージ ape を利用して図にする手順は、途中に phylo クラスへの変換などを必要とするため、少し面倒です。そこで、一連の作業を自作の関数として定義して、クラスタ解析の結果の図示を簡略化してみましょう。

```
> myplot <- function(tre, subpop, type = "unrooted", ...) {
                  #  関数 function を用いて自作関数を定義する
                  #  まずは自作関数の引数を指定する。ここでは、tre, subpop, type
                  #  type についてはデフォルトの値（"unrooted"）を設定してある
                  #  引き続いて{}で囲まれた部分に関数で実行する処理を記述する
         phy <- as.phylo(tre)       #  phylo クラスへの変換
         col <- as.numeric(subpop[phy$edge[,2]])
                       #  phylo 内の edge の情報を使って色コードを指定
         edge.col <- ifelse(is.na(col), "gray", col)
                       #  末端の枝以外（色コードが NA になっている）を灰色にする
         plot(phy, edge.color = edge.col, type = type, show.tip.label = F, ...)
                       #  樹形図を描く。設定した枝の色をオプションとして指定
         #  type = type に注意。2 番目の type には引数で指定された type が代入される
 }
```

では、自作の関数 myplot を使って樹形図を描いてみましょう。

```
> d <- dist(data.mk)                #  サンプル間の距離の計算
> tre <- hclust(d)                  #  クラスタ解析
> myplot(tre, subpop)               #  自作関数 myplot で樹形図を描く
> myplot(tre, subpop, type = "cladogram")
                                    #  オプション type を指定して樹形図を描く
```

＜距離の定義＞

クラスタ解析では、サンプル間やクラスタ間の距離を計算し、計算された距離に基づいてクラスタリングを行います。したがって、距離の定義が異なると異なる結果が得られることになります。ここでは、サンプル間やクラスタ間の距離の定義について解説します。

まずは、サンプル間の距離についてです。サンプル間の距離を計算するのに、様々な定義があります。まずは、異なる定義の距離に基づいて樹形図を描いてみましょう。

```
> pdf("fig5.pdf", width = 10, height = 10)  #  図が大きいので pdf に出力
> op <- par(mfrow = c(2, 2), , mar = rep(0, 4))
> d <- dist(data.mk, method = "euclidean")  #  ユークリッド距離（デフォルト設定）
> myplot(hclust(d), subpop)         #  自作関数で樹形図を描く
> d <- dist(data.mk, method = "manhattan")  #  マンハッタン距離
> myplot(hclust(d), subpop)
> d <- dist(data.mk, method = "minkowski", p = 1.5)   #  ミンコフスキー距離
> myplot(hclust(d), subpop)
> d <- as.dist(1 - cor(t(data.mk)))
                  #  相関係数に基づく距離。相関係数を r とすと 1-r を距離とする
                  #  関数 dist で計算できないが、関数 as.dist で dist クラスに変換できる
> myplot(hclust(d), subpop)
> par(op)
> dev.off()                         #  pdf ファイルを閉じる
```

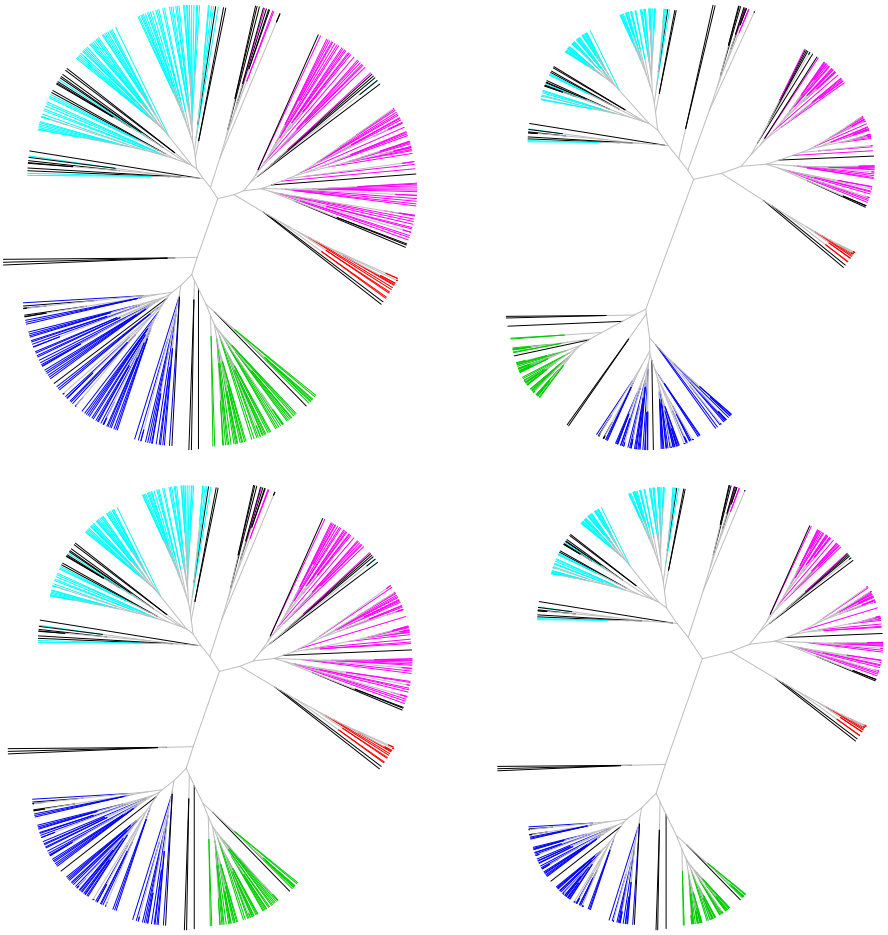

図 5. サンプル間の距離の異なる定義に基づいて計算された樹形図

今回のデータでは距離の定義が異なっても樹形図のトポロジー（topology）は大きく変わりませんが、データによっては距離の定義が大きく影響する場合があります。

上で用いたサンプル間の距離について、その定義を以下に示します。なお、各サンプルが q 個の特徴で記述されており、i 番目のサンプルのデータベクトルを $\mathbf{x}_i = (x_{i1},...,x_{iq})^{\mathrm{T}}$、j 番目のサンプルのデータベクトルを $\mathbf{x}_j = (x_{j1},...,x_{jq})^{\mathrm{T}}$ と表すこととします。このとき、サンプル i, j 間の距離 $d(\mathbf{x}_i,\mathbf{x}_j)$ は、以下のように定義されます。

- ユークリッド（Euclidian）距離

$$d(\mathbf{x}_i,\mathbf{x}_j) = \sqrt{\sum_{k=1}^{q}(x_{ik} - x_{jk})^2}$$

- マンハッタン（Manhattan）距離

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sum_{k=1}^{q}\left|x_{ik}-x_{jk}\right|$$

- ミンコフスキー距離

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sqrt[p]{\sum_{k=1}^{q}\left|x_{ik}-x_{jk}\right|^p}$$

- 相関に基づく距離

$$d(\mathbf{x}_i,\mathbf{x}_j)=1-r_{ij}=1-\frac{\sum_{k=1}^{q}(x_{ik}-\overline{x}_i)(x_{jk}-\overline{x}_j)}{\sqrt{\sum_{k=1}^{q}(x_{ik}-\overline{x}_i)^2}\sqrt{\sum_{k=1}^{q}(x_{jk}-\overline{x}_j)^2}}$$

ここで、$\overline{x}_i=\frac{1}{n}\sum_{k=1}^{q}x_{iq}$, $\overline{x}_j=\frac{1}{n}\sum_{k=1}^{q}x_{jq}$

マンハッタン距離は、ニューヨーク市の Manhattan のような正方形に区分された市街地を移動する場合の距離というのがその名の由来です。そのような市街地では、例えば、地点（0,0）から地点（2,3）に移動する場合に、建物があるために斜めに移動（ユークリッド距離$\sqrt{13}$）することができず、道に沿って移動（マンハッタン距離 5）する必要があるためです。ミンコフスキー距離は、ユークリッド距離とマンハッタン距離の一般化されたかたちです。$p=1$ のときはマンハッタン距離、$p=2$ のときはユークリッド距離に一致します。

相関に基づく距離では、「変数間ではなくサンプル間で」相関係数を計算して、それを 1 から減じたものを距離とします。相関が 1 の場合は距離 0、相関が 0 のときは距離 1、相関が−1 のときには距離が 2 となります。すなわち、相関係数に基づく距離では最大値が 2 となります。なお、遺伝子間で発現パターンの類似性からクラスタ解析を行う場合には、1 から相関を減ずる代わりに「相関の絶対値」を減ずる場合があります。この場合、相関が−1 または 1 のときには距離 0、相関が 0 のときに最も距離が遠くなり 1 となります。

関数 dist では、次のような距離も計算できます。今回のデータには不向きであったので利用しませんでしたが、解析するデータの性質によっては、以下に紹介する距離が適切な場合もあります。

- チェビシェフ（Chebyshev）距離
  （関数 dist で `method="maximum"` を指定）

$$d(\mathbf{x}_i,\mathbf{x}_j)=\max_k\left(\left|x_{ik}-x_{jk}\right|\right)$$

- キャンベラ（Canberra）距離
  （関数 dist で `method="canberra"` を指定）

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sum_{k=1}^{q}\frac{\left|x_{ik}-x_{jk}\right|}{\left|x_{ik}\right|+\left|x_{jk}\right|}$$

- ハミング距離（Hamming）距離
  （関数 dist で `method="binary"` を指定）

$$d(\mathbf{x}_i,\mathbf{x}_j)=\sum_{k=1}^{q}(1-\delta_{x_{ik},x_{jk}})$$

  ここで、

$$\delta_{a,b}=\begin{cases}1 & (a=b)\\ 0 & (a\neq b)\end{cases}$$

チェビシェフ距離は q 個の特徴のうち最も異なっている 1 つの特徴の違いだけに基づく距離です。この距離は、ミンコフスキー距離の $p\to\infty$ の極限となっています。ハミング距離は情報科学でよく用いられる距離で、同じ長さの数列について、同じ位置の値を比較したときに一致しない位置の数を数えあげたものです。ハミング距離を用いるようなデータでは、x_{ik} は、連続値ではなく、離散値（0, 1）である場合がほとんどです。

ここまではサンプル間の距離の定義について説明してきました。階層的クラスタ解析では、距離の近いサンプルどうしを 1 つのクラスタにまとめながら、さらに、サンプルとクラスタ、または、クラスタどうしを、上位の階層のクラスタとしてまとめあげていきます。したがって、サンプル間の距離だけでなく、サンプルとクラスタ、または、クラスタ間の距離を定義しておく必要があります。

ここでは、まず、クラスタ間距離の様々な定義に基づいて樹形図を描いてみます。関数 hclust では、クラスタ間の距離の計算方法（定義）をオプション method で指定することができます。

```
> pdf("fig6.pdf", width = 10, height = 10)  #  図が大きいので pdf ファイルに出力
> d <- dist(data.mk)                        #  ユークリッド距離を計算
> op <- par(mfrow = c(2, 3), mar = rep(0, 4))
> tre <- hclust(d, method = "complete")     #  最長距離法（完全連結法）
> myplot(tre, subpop)
> tre <- hclust(d, method = "single")       #  最短距離法（単連結法）
> myplot(tre, subpop)
> tre <- hclust(d, method = "average")      #  平均距離法
> myplot(tre, subpop)
> tre <- hclust(d, method = "median")       #  メディアン法
> myplot(tre, subpop)
> tre <- hclust(d, method = "centroid")     #  重心法
> myplot(tre, subpop)
> tre <- hclust(d, method = "ward.D2")      #  ウォード（Ward）法
> myplot(tre, subpop)
> par(op)
> dev.off()                                 #  pdf ファイルを閉じる
```

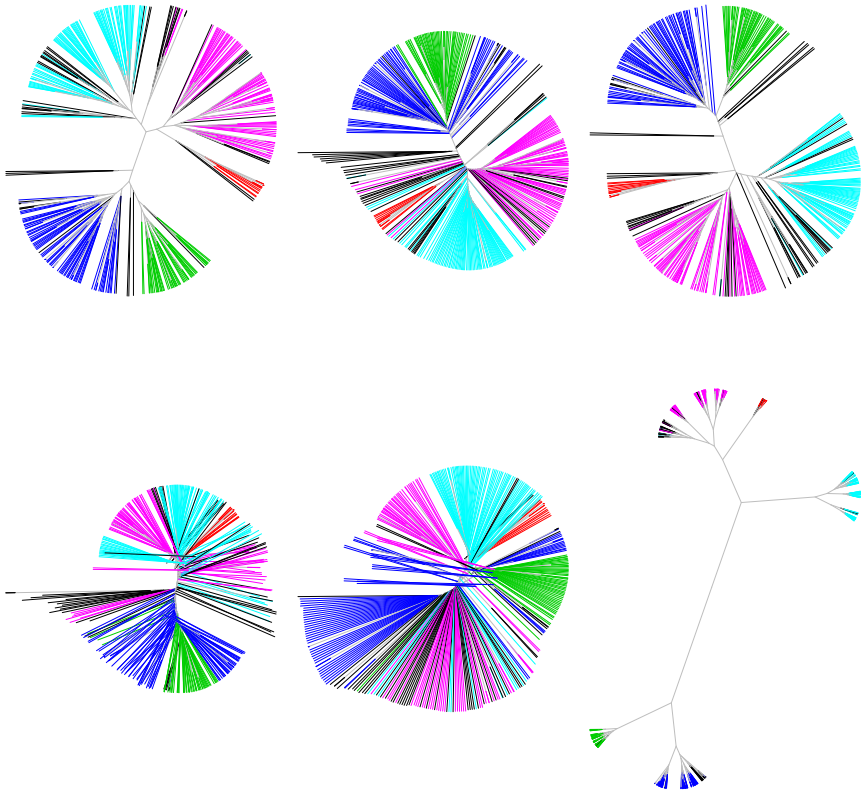

図 6. 様々なクラスタ間距離の定義に基づく樹形図

図 6 を見ると、クラスタ間距離の定義の違いは、サンプル間距離の定義の違いと異なり、樹形図のトポロジーが大きく変化することが分かります。中には、枝長が負の値になりおかしな樹形図になっている場合もあります（左下、下中央)。また、クラスタ間の違いが非常に強調される場合もあります（右下)。この中からどの手法を選択するかは難しい問題ですが、多くの場合、既知の情報と大きく矛盾が無いものが選ばれます。例えば、ここでは、品種・系統が所属している分集団と矛盾が小さいものを選ぶとよいでしょう。

関数 hclust で指定できるクラスタ間の距離の定義を示します。サンプル間の距離 $d(\mathbf{x}_i,\mathbf{x}_j)$ に基づき、クラスタ A と B の距離を d_{AB} は以下のように計算されます。

- 最長距離法（完全連結法）
  （関数 hclust で `method="complete"` を指定）

$$d_{\mathrm{AB}} = \max_{\substack{i\in A \\ j\in B}} \left(d(\mathbf{x}_i,\mathbf{x}_j) \right)$$

- 最短距離法（単連結法）
  （関数 hclust で `method="single"` を指定）

$$d_{\mathrm{AB}} = \min_{\substack{i\in A \\ j\in B}} \left(d(\mathbf{x}_i,\mathbf{x}_j) \right)$$

- 平均距離法
  （関数 hclust で `method="average"` を指定）

$$d_{\mathrm{AB}} = \frac{1}{n_{\mathrm{A}} n_{\mathrm{B}}} \sum_{i\in A} \sum_{j\in B} d(\mathbf{x}_i,\mathbf{x}_j)$$

  ここで、$n_{\mathrm{A}}, n_{\mathrm{B}}$ はクラスタ A, B に含まれるサンプルの数を表す。

以下の 3 つの定義では、クラスタ A, B が融合して新しくクラスタ C ができるときに、新しいクラスタ C と A,B 以外のクラスタ O 間の距離 d_{CO} を次のように定義する。なお、クラスタ A と B の距離を d_{AB}、クラスタ A と O の距離を d_{AO}、クラスタ B と O の距離を d_{BO}、クラスタ A、B、O に含まれるサンプルの数を $n_{\mathrm{A}}, n_{\mathrm{B}}, n_{\mathrm{O}}$ と表す。

- 重心法

（関数 hclust で `method="centroid"` を指定）

$$d_{CO}^2 = \frac{n_A}{n_A + n_B} d_{AO}^2 + \frac{n_B}{n_A + n_B} d_{BO}^2 - \frac{n_A n_B}{(n_A + n_B)^2} d_{AB}^2$$

- メディアン法

  （関数 hclust で `method="median"` を指定）

$$d_{CO} = \frac{1}{2} d_{AO} + \frac{1}{2} d_{BO} - \frac{1}{4} d_{AB}$$

- ウォード（Ward）法

  （関数 hclust で `method="ward.D2"` を指定）

$$d_{CO}^2 = \frac{n_A + n_O}{n_A + n_B + n_O} d_{AO}^2 + \frac{n_B + n_O}{n_A + n_B + n_O} d_{BO}^2 - \frac{n_O}{n_A + n_B + n_O} d_{AB}^2$$

図 6 において分集団との対応が明瞭と思われる 2 つの手法について、もう少し詳細に比較してみましょう。

```
> op <- par(mfrow = c(1, 2))          #  グラフを 1 行 2 列で配置
> d <- dist(data.mk)                  #  ユークリッド距離を計算
> tre <- hclust(d, method = "complete")      #  最長距離法
> myplot(tre, subpop, type = "phylogram")    #  phylogram として樹形図を描く
> tre <- hclust(d, method = "ward")          #  ウォード法
> myplot(tre, subpop, type = "phylogram")
> par(op)                             #  グラフィックオプションをリセットする
```

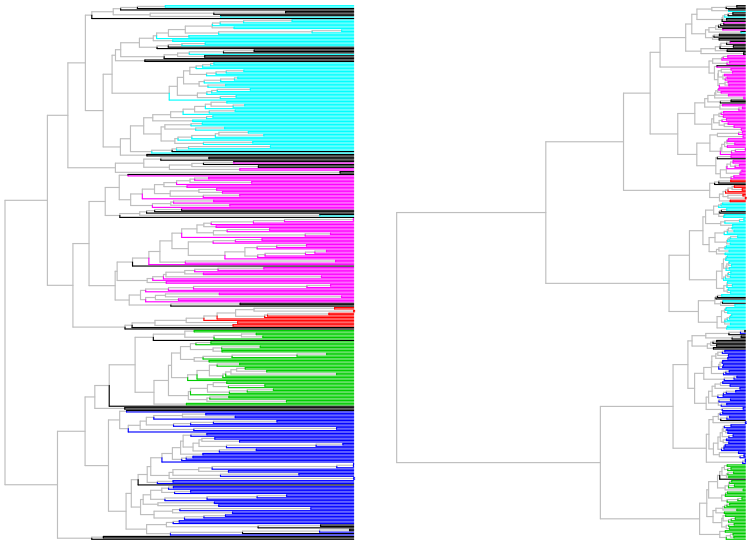

図 7. 最長距離法（左）とウォード法（右）による樹形図

＜多次元データの両側からのクラスタ解析＞

ここまでは、DNA マーカーデータをもとに品種や系統をクラスタに分類しました。品種や系統のクラスタ解析は、DNA マーカーデータだけでなく、形質データをもとにしても行うことができます。また、全く同じデータについて、品種・系統ではなく、形質を分類する対象とみなして、品種・系統間で似たような変異のパターンをもつ形質どうしを同じクラスタに分類することもできます。ここでは、このようなアプローチについて説明を行います。

まずは、形質データを準備しましょう。全データ（alldata）から、このような解析に適さない形質を除き、形質データを抜き出しましょう。

```
> required.traits <- c("Flowering.time.at.Arkansas",
         "Flowering.time.at.Faridpur", "Flowering.time.at.Aberdeen",
         "Culm.habit", "Flag.leaf.length", "Flag.leaf.width",
         "Panicle.number.per.plant", "Plant.height", "Panicle.length",
         "Primary.panicle.branch.number", "Seed.number.per.panicle",
         "Florets.per.panicle", "Panicle.fertility", "Seed.length",
         "Seed.width","Brown.rice.seed.length", "Brown.rice.seed.width",
         "Straighthead.suseptability","Blast.resistance",
         "Amylose.content", "Alkali.spreading.value", "Protein.content")
> data.tr <- alldata[, required.traits]      #  必要な形質だけを抜き出す
> missing <- apply(is.na(data.tr), 1, sum) > 0    #  欠測をもつサンプルを見つける
> data.tr <- data.tr[!missing, ]             #  欠測をもつサンプルを除く
> subpop.tr <- alldata$Sub.population[!missing]   #  分集団データも準備しておく
```

形質データは、形質によって変動の大きさ（分散）が異なります。このデータをそのまま用いると、分散の大きな形質は距離の計算に大きな影響を与え、分散の小さな形質は距離の計算への寄与が小さくなります。そこで、全ての形質について、分散 1 に基準化しておきます。

```
> data.tr <- scale(data.tr)          #  データを基準化しておく
```

では、形質データについて、品種・系統を分類の対象としたクラスタ解析と、形質を分類の対象としたクラスタ解析を行ってみましょう。

```
> d <- dist(data.tr)                    #  形質データから品種・系統間の距離を計算
> tre.var <- hclust(d, method = "ward.D2")
                                #  ウォード法によるクラスタ解析（品種・系統を分類）
> d <- dist(t(data.tr))                 #  形質データから形質間の距離を計算
> tre.tra <- hclust(d, "ward.D2")
                                #  ウォード法によるクラスタ解析（形質を分類）
> op <- par(mfrow = c(1, 2))            #  図を 1 行 2 列で配置する
> myplot(tre.var, subpop.tr, type = "phylogram")
                                #  自作関数 myplot を使って品種・系統間の関係を樹形図で表示
> plot(tre.tra, cex = 0.5)              #  hclust の結果をそのままプロット
                                        #  形質間の関係を樹形図で表示
> par(op)
```

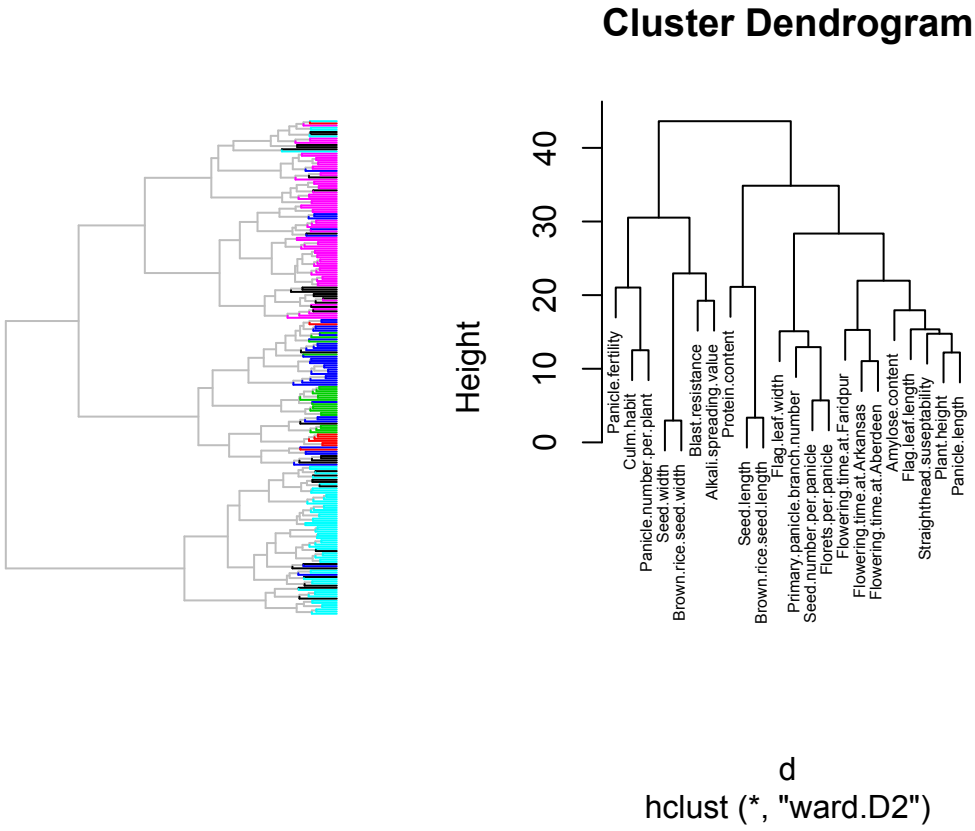

図 8. 形質データをもとにしたクラスタ解析
品種・系統の関係（左）と形質間の関係（右）を表す樹形図

図 8 の右側の樹形図から、植物体のサイズに関わる形質（Plant.height, Panicle.length, Flag.leaf.length）のは互いに関係が強いことが分かります。また、そのクラスタの近くに、3 つの環境で計測された開花のタイミング（Flowering.time.at.*****）が位置していることも分かります。また、止め葉

の幅（Flag.leaf.width）は、他のサイズ関連形質と異なり、穂の特徴を表す形質との関連が強いことも分かります。このように、多次元データはどちら側からもクラスタ解析を行うことができます。このことを覚えておくと、同じデータを少し違った視点から眺めることができるでしょう。

なお、上述した解析は、関数 heatmap を用いてより視覚的に結果を表示できます。

```
> pdf("fig9.pdf")                    #  グラフが大きいので pdf に出力
> heatmap(data.tr, margins = c(12,2))
          #  関数 heatmap による両側からのクラスタ解析とデータのヒートマップ表示
> dev.off()
```

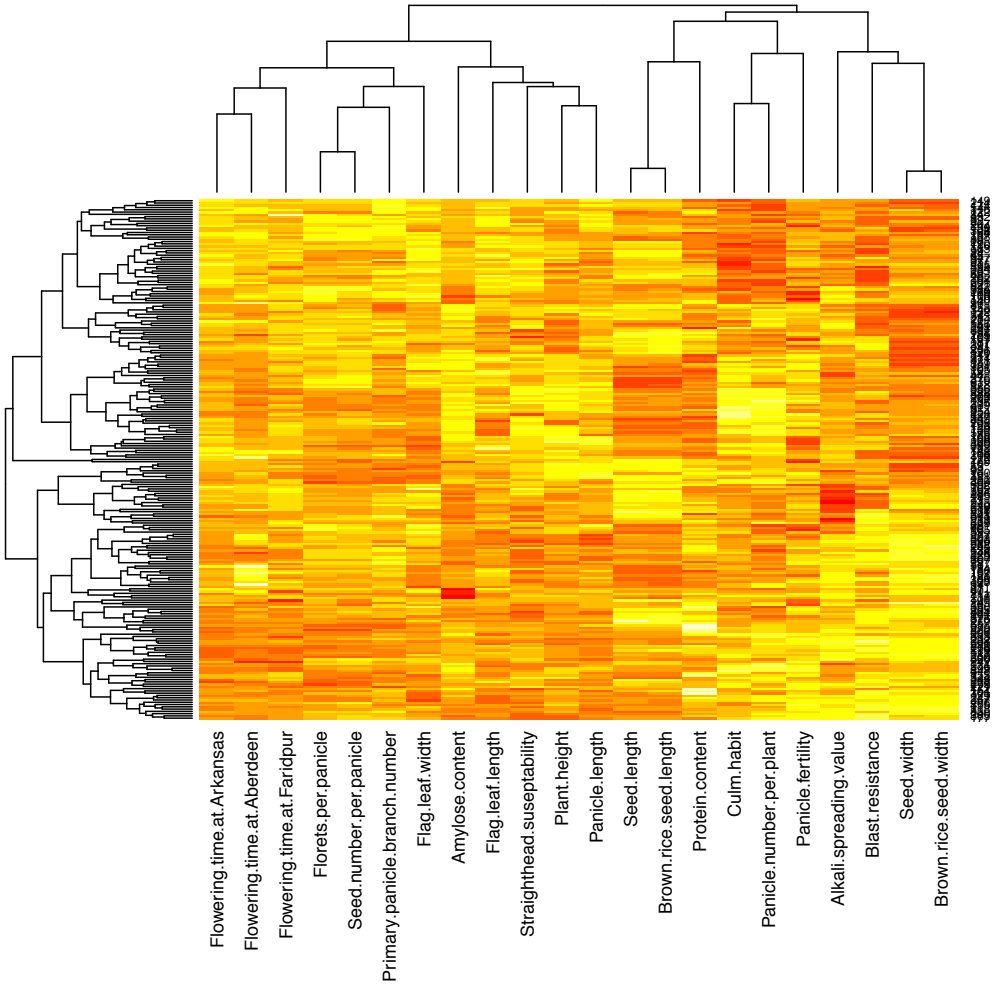

図 9. 形質データのクラスタ解析の結果とヒートマップの表示

以下のようにすると、別に行ったクラスタ解析の結果を反映させることができます。先ほど、同データについて行ったクラスタ解析の結果をヒートマップ表示に反映させてみましょう。

```
> pdf("fig10.pdf")                    #  グラフを pdf に出力する
> heatmap(data.tr,                    #  形質データを指定
          Rowv = as.dendrogram(tre.var),      #  品種・系統を分類した樹形図
          Colv = as.dendrogram(tre.tra),      #  形質を分類した樹形図
          RowSideColors = as.character(as.numeric(subpop.tr)),
                                      #  分集団の情報を色付きのバーで表示
          labRow = subpop.tr,         #  行側のラベルを分集団名に置き換える
          margins = c(12, 2))         #  グラフの余白のサイズを指定
> dev.off()
```

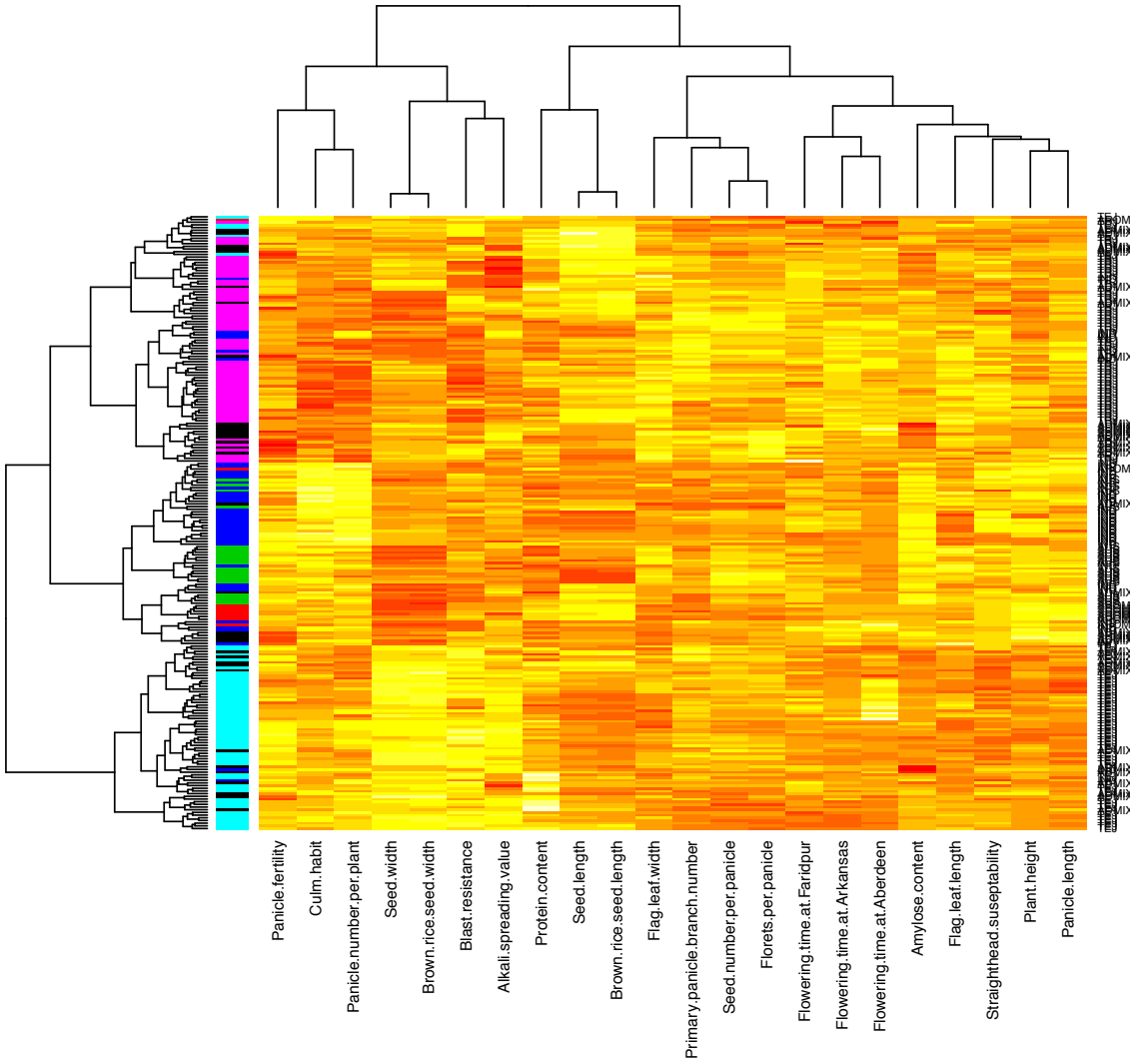

図 10. 図 9 のクラスタ解析の手法をウォード法に変更した結果

関数 heatmap では、必ずしも縦横同じデータでクラスタ解析をする必要はありません。例えば、行側について、DNA マーカーデータを用いたクラスタ解析の結果をあてはめることもできます。

```
> data.mk2 <- data.mk[!missing, ]   #  表現型データに欠測があるものを除く
                                    #  これを忘れるとサンプル数が一致しないのでエラーになる
> d <- dist(data.mk2)                #  DNA データでの距離を計算する
> tre.mrk <- hclust(d, method = "ward.D2")   # ウォード法でクラスタ解析
> pdf("fig11.pdf")
> heatmap(data.tr, Rowv = as.dendrogram(tre.mrk),     #  ここが、先ほどと異なる
                   Colv = as.dendrogram(tre.tra),     #  後は同じ
                   RowSideColors = as.character(as.numeric(subpop.tr)),
                   labRow = subpop.tr,
                   margins = c(12, 2))
> dev.off()
```

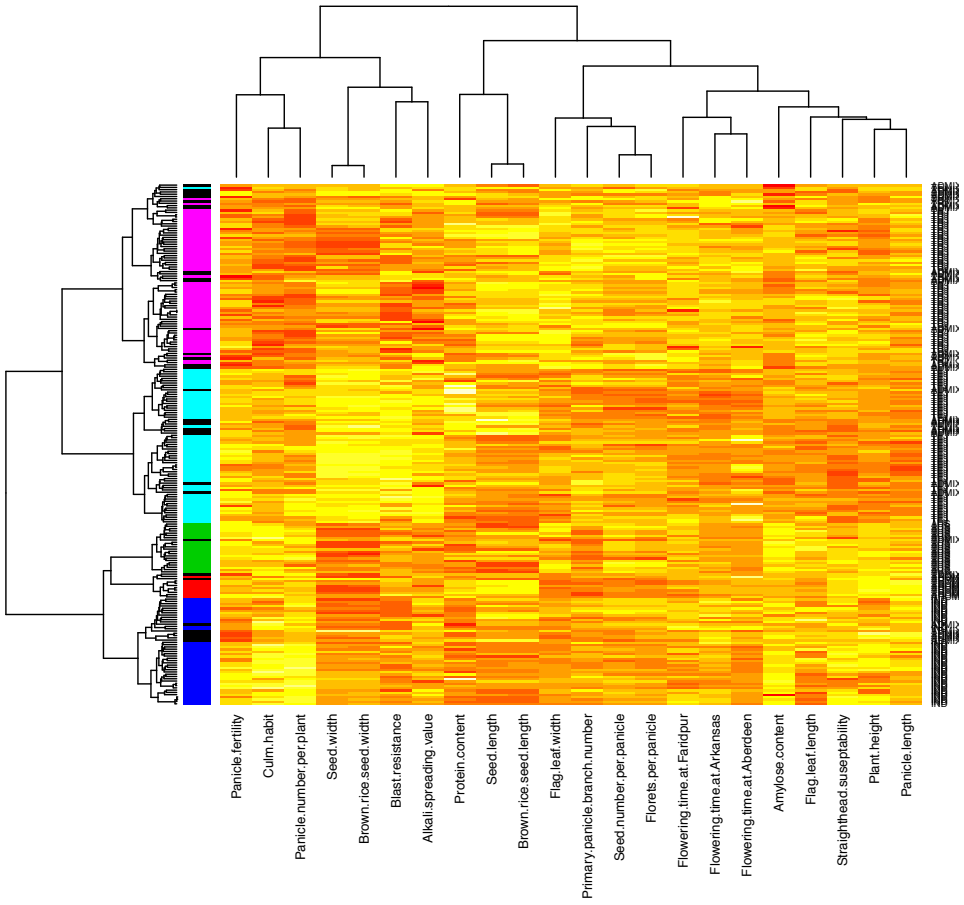

図 11. 遺伝マーカーデータを基にしたクラスタ解析の結果と形質の関係

＜階層的クラスタ解析に基づく分類＞

サンプル間の類似性についてあるところで線引きし、サンプルを離散的にグループに分類したい場合があります。ここでは、階層的クラスタ解析の結果に基づいてサンプルを決められた数のクラスタに分類する方法について説明します。

DNA マーカーデータに基づく階層的クラスタ解析の結果に基づき、品種・系統を 5 つのグループに分類してみましょう。5 というのは、品種・系統の所属する分集団の数に合わせた数字です。階層的クラスタ解析の結果から、離散的なグループを求めるには関数 cutree を用います。

```
> d <- dist(data.mk)                    #  DNA マーカーデータからユークリッド距離を計算
> tre <- hclust(d, method = "ward.D2")                  #  ウォード法によるクラスタ解析
> cluster.id <- cutree(tre, k = 5)
          #  関数 cutree を用いて樹形図に基づきサンプルを 5 つのクラスタに分類
```

クラスタ解析に基づき 5 つのグループに分類した結果を図示してみましょう。

```
> op <- par(mfrow = c(1,2), mar = rep(0, 4))  #  グラフィックオプションの設定
> myplot(tre, cluster.id, type = "phylogram")
                    #  関数 cutree の分類結果（cluster.id）で色づけ
> myplot(tre, subpop, type = "phylogram", direction = "leftwards")
          #  所属する分集団（subpop）で色づけ。オプション direction で樹形図の向きを指定
> par(op)
```

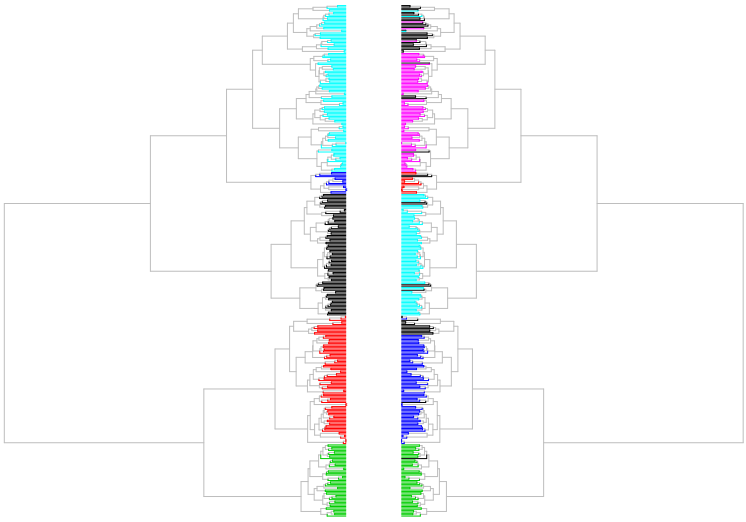

図 12. クラスタ解析による分類（左）と分集団（右）の関係

クラスタ解析に基づく分類と分集団の関係を、クロス集計表を作成して確認してみましょう。

```
> table(cluster.id, subpop)      #  cluster.id と subpop のクロス集計表を表示
                subpop
cluster.id ADMIX AROMATIC AUS IND TEJ TRJ
        1     5        0   0   0  84   0
        2    14        0   0  80   0   0
        3     1        0  52   0   0   0
        4     2       14   0   0   0   0
        5    34        0   0   0   3  85
```

両者は、インディカ（IND）の 3 品種・系統を除いて、非常によく一致していることが分かります。これは、分集団構造そのものが DNA マーカーデータに基づいて推定されたためだと考えられます。なお、複数の分集団の混合（ADMIX）と推定されている品種については様々なグループに分類されていることも分かります。

階層的クラスタ解析による分類の結果を主成分軸上で確認してみましょう。

```
> pca <- prcomp(data.mk)              #  主成分分析
> op <- par(mfrow = c(1,2))           #  グラフを 1 行 2 列に並べる
> plot(pca$x[,1:2], pch = cluster.id, col = as.numeric(subpop))
                    #  第 1、2 主成分の散布図を描く
                    #  点のタイプで分類の結果を、点の色で分集団を表す
> plot(pca$x[,3:4], pch = cluster.id, col = as.numeric(subpop))
> par(op)
```

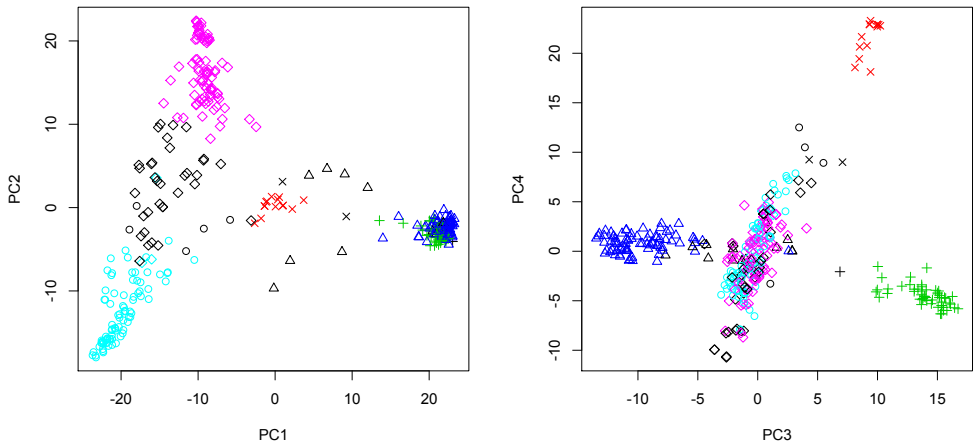

図 13. クラスタ解析による分類と分集団間の関係

＜非階層的クラスタリング＞
ある決められたグループ数に分類する場合には、階層的に分類を行う必要は必ずしもありません。ここでは、非階層的クラスタ解析手法の一つである k-平均（k-means）法を紹介します。

先ほどと同じデータについて、関数 kmeans を用いて 5 つのグループへの分類を行ってみましょう。

```
> kms <- kmeans(data.mk, centers = 5)        #  関数 kmeans で 5 グループに分類
> kms                                        #  結果の表示
（結果は省略）
```

k-平均法では、以下のようなアルゴリズムで決められたグループ数への分類を行います。

1. k 個のクラスタ中心として、k 個のサンプルを無作為に選びだす
2. すべてのデータ点と k 個のクラスタ中心間の距離を求め、各データ点を中心（重心を中心とする）が最も近いクラスタに分類する
3. 形成されたクラスタの中心（重心）を更新する
4. クラスタの中心（重心）が変化しなくなるまで、2-3 を繰り返す

k-平均法では、最初に無作為に選ばれるサンプルによって結果が変化することがあります。実際に、同じデータで解析を繰り返して、結果のバラツキを確認してみましょう。

```
for(i in 1:5) {            #  同じ解析を 5 回繰り返す
        kms <- kmeans(data.mk, centers = 5, nstart = 50)
           #  nstart = 50 は、最初に選択されるサンプルを 50 セット別のもので比較する
        print(table(kms$cluster, subpop))
 }
```

先ほどと異なり、結果が安定していることが分かると思います。なお、各サンプルが分類されるグループの「番号」は異なる解析の間で異なってきますが、これは 5 つのグループに任意につけられている番号なので特に問題はありません。

では、k-平均法で分類されたグループと、階層的クラスタ解析で分類されたグループ、および、品種・系統が所属する分集団の関係について、クロス集計表を作成して確認してみましょう。

```
> table(kms$cluster, subpop)        #  k-平均法と分集団間でクロス集計表を作る
   subpop
    ADMIX AROMATIC AUS IND TEJ TRJ
  1     1        0  52   0   0   0
  2    23        0   0   0   1  85
  3    17        0   0   0  86   0
  4    12        0   0  80   0   0
  5     3       14   0   0   0   0
 > table(cluster.id, subpop)        #  階層的クラスタ解析の結果と分集団の比較
         subpop
cluster.id ADMIX AROMATIC AUS IND TEJ TRJ
        1     5        0   0   0  84   0
        2    14        0   0  80   0   0
        3     1        0  52   0   0   0
        4     2       14   0   0   0   0
        5    34        0   0   0   3  85
> table(kms$cluster, cluster.id)    #  階層的クラスタ解析の結果と k-平均法の比較
   cluster.id
      1   2   3   4   5
  1   0   0  53   0   0
  2   0   1   0   0 108
  3  88   1   0   0  14
  4   0  92   0   0   0
  5   1   0   0  16   0
```

クロス集計表を作成してみると、ADMIX と IND 以外の分集団に所属する品種・系統については、k-平均法と階層的クラスタ解析で同じように分類されていることが分かります。3 つ目のクロス集計表を見ると、両手法の分類結果はほぼ一致していますが、一部違いが見られます。これは、分集団が ADMIX（混合）となっている品種・系統の分類が両手法で異なることが主な原因です。

では、k-平均法と階層的クラスタ解析による分類の結果を、主成分軸上にプロットすることにより確認してみましょう。

```
> convert.table <- apply(table(kms$cluster, cluster.id), 1, which.max)
   #  クラスタの ID を合わせるために、両方法のクロス集計表で最頻の組合せの番号を調べる
> convert.table            #  ID の読み換えをするための読み換えテーブル
1 2 3 4 5
5 3 4 2 1
> cluster.id.kms <- convert.table[kms$cluster]        #  ID の変換
> pdf("fig14.pdf", width = 8, height = 8)    #  グラフを pdf ファイルとして出力
> op <- par(mfrow = c(2,2))
> plot(pca$x[,1:2], pch = cluster.id, col = as.numeric(subpop),
                  main = "hclust")           #  階層的クラスタ解析の結果
> plot(pca$x[,3:4], pch = cluster.id, col = as.numeric(subpop),
                  main = "hclust")
> plot(pca$x[,1:2], pch = cluster.id.kms, col = as.numeric(subpop),
                  main = "kmeans")           #  k-平均法の結果
> plot(pca$x[,3:4], pch = cluster.id.kms, col = as.numeric(subpop),
                  main = "kmeans")
> par(op)
> dev.off()
```

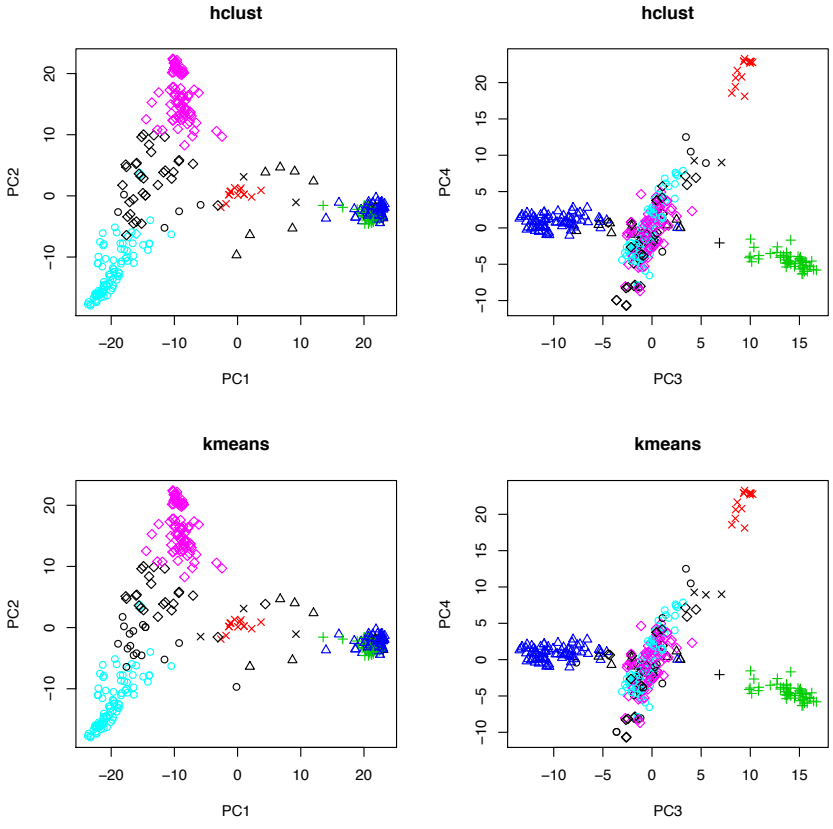

図 14. 階層的クラスタ解析（上）および k-平均法（下）による分類と主成分得点の関係

＜適切なグループ数の決定＞

ここまで分類するグループ数を分集団の数に合わせて 5 としてきました。では、この 5 というグループ数が本当に適切な数かどうかを確認するには、どのようにするとよいのでしょうか。適切なグループ数を決めるための 1 つの方法として、様々な数のグループに分類して、そのときの群（グループ）内平方和（Within groups sum of squares）の減少の程度を基準に決めるという方法があります。

分散分析の際に説明したのと同じように、全平方和は群間平方和と群内平方和に分割されます。したがって、グループ数が 1 のときは、全平方和が群内平方和となります。その後、2, 3, 4…とグループ数が増えていくと、群間平方和が大きくなり、群内平方和は小さくなって行きます。最終的にグループ数がサンプル数に一致すると、群内平方和は 0 となります。したがって、群内平方和を最小化するというルールではグループ数が常にサンプル数となってしまい意味がありません。そこで、主成分分析において主成分の数を決めたときのルールと同じように、群内平方和の減少が、急な変化からなだらかな変化に変わる点を見つけ、それを採用するグループ数とします。

では、実際にグループの数を 1～10 に変化させて群内平方和を計算し、その減少の様子を図示してみましょう。

```
> n <- nrow(data.mk)                    #  サンプルの数を n とする
> wss <- rep(NA, 10)                    #  群内分散を代入する入れ物（配列）を準備
> wss[1] <- (n - 1) * sum(apply(data.mk, 2, var))
                              #  全分散を計算して、それに n-1 を乗じて平方和に戻す
> for(i in 2:10) {
         print(i)             #  i を表示させて、計算の経過が分かるようにする
         res <- kmeans(data.mk, centers = i, nstart = 50)
                                    #  k = 2-10 で k-平均法を適用
         wss[i] <- sum(res$withinss)
                              #  群内平方和を配列 wss の i 番目の要素として代入
 }
（結果は省略）
> plot(1:10, wss, type = "b", xlab = "Number of groups",
                               ylab = "Within groups sum of squares")
         #  x 軸を 1:10 に、y 軸を群内平方和としてグラフを描く
         #  オプション type = “b”は、プロットと折れ線でグラフを描く
```

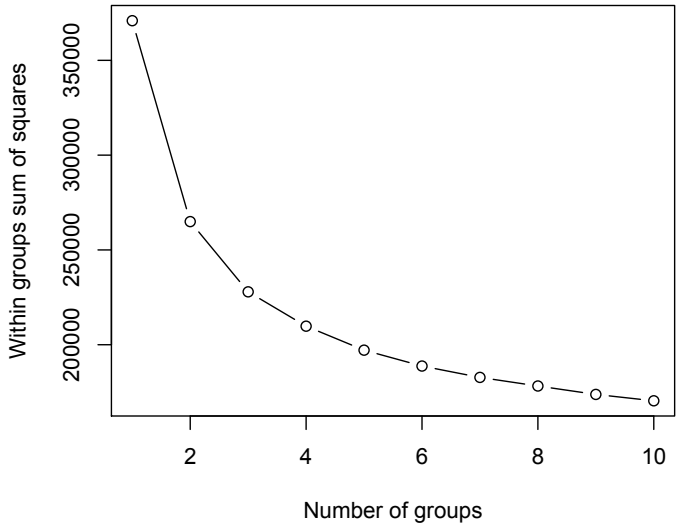

図 15. k-平均法でクラスタ数を 1〜10 にした場合の群内分散の変化

図 15 を見ると、クラスタ数が 5 を過ぎたあたりから群内分散の減少が直線的になるのが分かります。この図からも、5 という数が適切なグループ数であると考えられます。

＜分類が曖昧なサンプルの検出＞

ここまでは、各サンプルを必ず 1 つのグループに分類してきました。すると、あるグループに分類されたサンプルの中にも、明確にそのグループに分類されたものと、「かろうじて」そのグループに分類されたものが存在することになります。分類の確かさを明示するためにも、後者のように分類が曖昧なサンプルを何らかの基準で検出できると便利です。ここでは、shadow value（Everitt and Hothorn 2011, An introduction to applied multivariate analysis with R. Springer）という統計量をもとに分類の曖昧さを評価する方法を紹介します。

関数 kms では、k-平均法で求められた各グループの重心の位置が、計算されています。ここでは、各サンプルからこれらグループの重心までの距離を計算し、最も近いグループまでの距離と、次に近いグループまでの距離を計算し、その違いをもとに曖昧さを評価してみます。

まずは、パッケージ fields に含まれている関数 rdist を用いて、全サンプルとグループ重心間の距離を計算します。

```
> require(fields)                    #  パッケージ fields の読み込み
> d2ctr <- rdist(kms$centers, data.mk)
         #  グループ重心（kms$centers）とサンプル（data.mk）間の距離を全通り計算
> d2ctr                              # 内容確認
（結果は省略）
> apply(d2ctr, 2, which.min)         #  最も近い（距離が最小の）グループを表示
（結果は省略）
> kms$cluster                        #  k-平均法の結果の表示
（結果は省略）
```

k-平均法では、先に述べたように、重心までの距離が最も近いグループに各サンプルを分類します。したがって、上のボックスで表示される 2 つの結果は、一致することに注意しましょう。

各サンプルについて計算された距離から、もっとも近い重心までの距離を取り出すには、関数 min を使うことができます。しかし、2 番目に近い重心までの距離を取り出すのにはどうすればよいのでしょうか。ここでは、自作関数 nth.min を作成して、これを実現します。自作関数 nth.min は、引数 x として与えられた配列を大きさ順（昇順）に並べ直し、その n 番目の値を返すという関数です。

```
> nth.min <- function(x, n) {       #  x と n が引数の自作関数 nth.min を定義
         sort(x)[n]                 #  関数 sort で x を昇順に並べ、その n 番目を返す
 }
> nth.min(-10:10, 3)               # 試しに使ってみる
> d.1st <- apply(d2ctr, 2, min)     #  関数 min を使って最近の重心までの距離を得る
> d.2nd <- apply(d2ctr, 2, nth.min, n = 2)
                        #  自作関数 nth.min を使って 2 番目に近い重心までの距離を得る
```

上のボックスにあるコマンドを実行すると、d.1st には、各サンプルから最も近い重心までの距離が、d.2nd には、2 番目に近い重心までの距離が代入されます。

次に、shadow value を計算してみましょう。i 番目のサンプルの shadow value は以下のように定義されます

$$s(\mathbf{x}_i) = \frac{2d(\mathbf{x}_i, c(\mathbf{x}_i))}{d(\mathbf{x}_i, c(\mathbf{x}_i)) + d(\mathbf{x}_i, \tilde{c}(\mathbf{x}_i))}.$$

ここで、$d(\mathbf{x}_i, c(\mathbf{x}_i))$ は、i 番目のサンプルの観察値 $\mathbf{x}_i$ から最も近いグループの重心（そのサンプルが分類されたグループの重心）までの距離、$d(\mathbf{x}_i, \tilde{c}(\mathbf{x}_i))$ は、2 番目に近いグループの重心までの距離を表しています。この値は、0〜1 までの値をとります。この値が 0 に近ければ、そのサンプルが分類されたグループの重心付近に位置していることを示しており、逆に、1 に近ければ、分類されたグループの重心と 2 番目に近いグループの重心までの距離がほとんど同じであることを意味します。したがって、分類が曖昧なサンプルを検出するには、shadow value が 1 に近いサンプルを見つければよいということになります。

先ほど計算しておいた距離 d.1st および d.2nd を用いて shadow value を計算し、その値が 0.9 以上になるものを検出してみましょう。

```
> shadow <- 2 * d.1st / (d.1st + d.2nd)
> unclear <- shadow > 0.9          #  shadow が 0.9 より大きいものを T（真）にする
```

検出の結果は、unclear に代入されます。この値が、T（真）であれば分類が曖昧、F（偽）であれば分類が比較的明瞭であると考えられます。

では、分類が曖昧と判断されたサンプルの●で表して、主成分軸上の散布図を描いてみましょう。

```
> cluster.id.kms[unclear] <- 20     #  unclear が T（真）のものに 20 を代入
                                    #  20 は●での散布を表すコード番号
> op <- par(mfrow = c(1,2))
> plot(pca$x[,1:2], pch = cluster.id.kms, col = as.numeric(subpop),
                  main = "kmeans")  #  図 15 と同様に散布図を描く
> plot(pca$x[,3:4], pch = cluster.id.kms, col = as.numeric(subpop),
                  main = "kmeans")
> par(op)
```

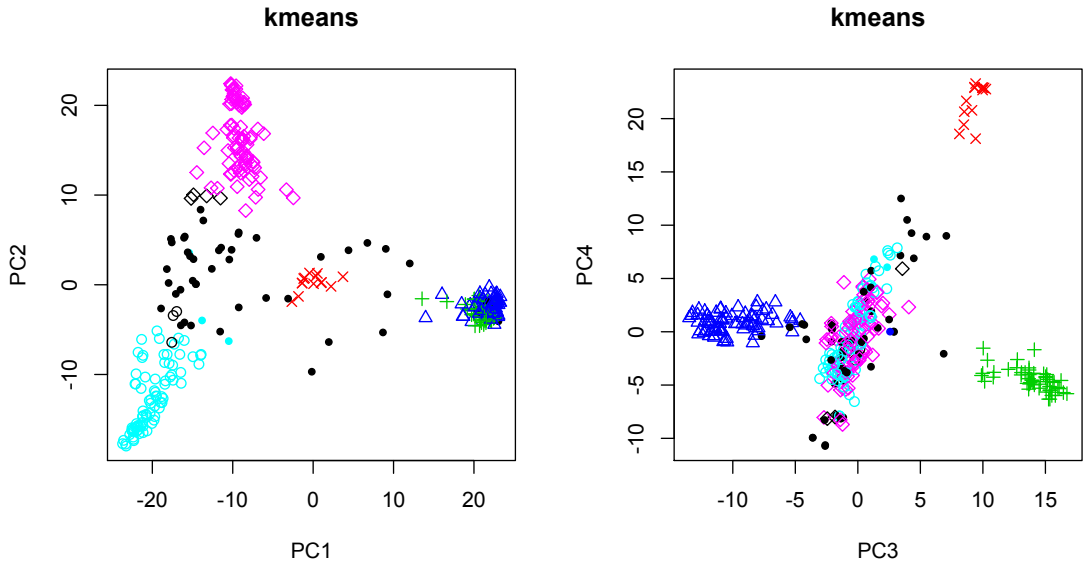

図 16. 分類が曖昧なサンプルを●で表した散布図

図 16 をみると分集団が ADMIX（混合）となっている品種・系統（黒色の点）のほとんどが●で散布されていることが分かります。このように、各サンプルの分類の曖昧さ（逆に言うと、確からしさ）を評価することで、より詳細に分類結果を把握することがでるようになります。この例では、複数の分集団に起因するゲノムが混合されていると考えられる品種・系統を見つけ出すことができました。

＜代表サンプルの選択＞

クラスタ解析は、多数のサンプルから、少数の代表的なサンプルを選び出すためにも利用できます。例えば、多数の遺伝資源について収集された既存のデータをもとにクラスタ解析で分類を行い、その分類結果に基づいて代表的な品種・系統を選び出すことができます。こうして代表的な品種・系統を選び出しておいて、それら品種・系統を用いて時間やコストを要する圃場試験や分子生物学的実験を行うことがよく行われます。

ここでは、このような代表サンプルの選択に適したクラスタ解析の方法として、k-medoids 法を紹介します。k-medoids 法は、k-平均法に似ていますが、グループの重心までの距離をもとにグループ分けするのではなく、グループの代表サンプル（medoids）までの距離をもとにグループ分けします。より具体的には、クラスタの中心を重心とするのではなく、グループの代表サンプルの座標点とするアルゴリズムです。

では、表現型データ（data.tr）に含まれる 229 品種・系統の中から、k-medoids 法で代表となる 48 サンプルを選出してみましょう。k-medoids 法を実行する関数 pam はパッケージ cluster に含まれています。k-medoids 法で得られる結果のうち、medoids の id（id.med）が代表として選ばれたサンプルの ID です。

```
> require(cluster)    #  k-medoids 法のための関数 pam が含まれるパッケージを呼び出す
> kmed <- pam(data.tr, k = 48)      #  関数 pam を使って k-medoids 法を実行
                                    #  k = 48 が分類するグループ数
> kmed                              # 結果の表示
 （結果は省略）
> kmed$id.med                       #  代表として選ばれたサンプルの ID を表示
 [1]   1  28   8 192 121  80   7  53  10 218  33  15  63 191  18  83 126  27
[19]  93  98  36  38 202 106  52  54 148 136 101  62 211 161  86 145  85  91
[37]  92 207 123 203 124 159 178 137 138 162 166 214
```

なお、ここで用いられた表現型データ（data.tr）は既に分散 1 に基準化されていたことに注意しましょう。もし、自分の手持ちのデータで同様の解析を行う場合には、データを基準化する必要があるかどうかをよく検討して、必要であれば、関数 scale を用いて基準化しておきます。

では、代表として選ばれたサンプルのもつ変異を、主成分軸上の散布図として図示してみましょう。

```
> pca.tr <- prcomp(data.tr)          #  主成分分析を実行
> mypch <- rep(1, nrow(data.tr))     #  データの数だけ 1 が並ぶ配列を作成する
> mypch[kmed$id.med] <- 19           #  そのうち代表として選ばれたものだけ 19 に変更
                                     #  1 は○での散布を表すコード、19 は●を表す
> op <- par(mfrow = c(1,2))
> plot(pca.tr$x[,1:2], col = as.numeric(subpop.tr), pch = mypch)
                                     #  準備しておいたコード mypch で散布図を描く
> plot(pca.tr$x[,3:4], col = as.numeric(subpop.tr), pch = mypch)
> par(op)
```

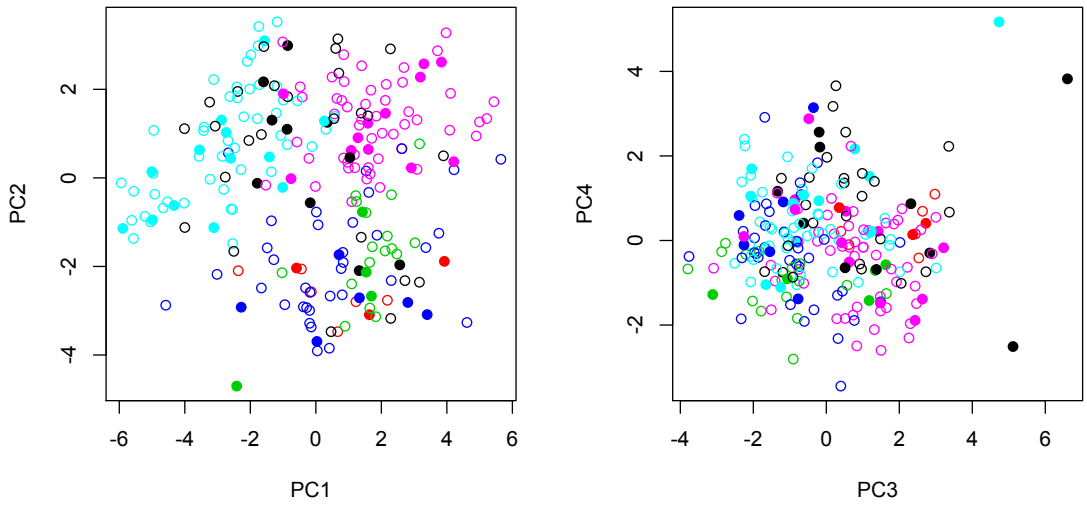

図 17. k-medoids 法で選出された代表 48 品種・系統の分布

最後に、k-medoids 法を用いて選出された 48 品種・系統のもつ主成分得点の分布と、全品種・系統のもつ主成分得点の分布を、ヒストグラムを描いて比較してみましょう。

```
> op <- par(mfcol = c(2,4))          #  グラフを 2 行 4 列で並べる。
                  #  mfrow と mfcol の違いは後者は列方向からグラフを並べる点
> for(i in 1:4) {          #  第 1～4 主成分のヒストグラムを、for 文を用いて描く
         res <- hist(pca.tr$x[,i], main = paste("PC", i, "all"))
                  #  全データのヒストグラム
                  #  関数 hist の結果を res に代入しておき、次行のコマンドで利用。
         hist(pca.tr$x[kmed$id.med, i], breaks = res$breaks,
                  main = paste("PC", i, "k-medoids"))
                  #  代表として選ばれた品種・系統のヒストグラム
                  #  全データのヒストグラムの区切り位置（res$breaks）を踏襲する
}
> par(op)
```

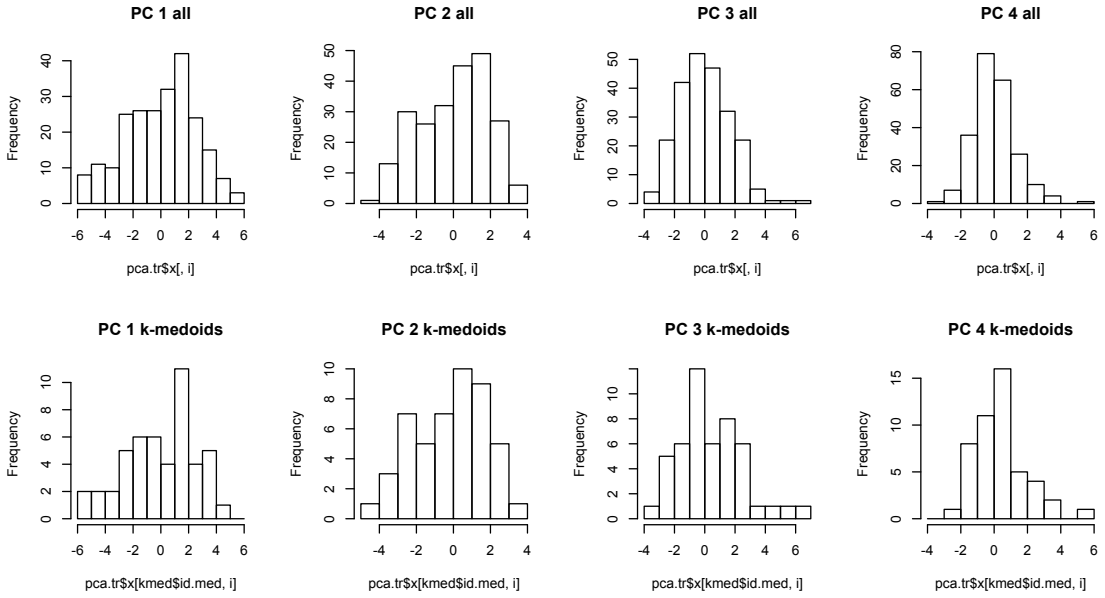

図 19. 全品種・系統（上）および代表として選ばれた品種・系統（下）の主成分得点の分布

図 19 を見ると、k-medoids 法で選ばれた 48 品種・系統が、全品種・系統がもつ形質の変異をよく代表できていることが分かります。

このように、クラスタ解析は、多数のサンプルから少数の代表を選出するのにも用いることができます。クラスタ解析のこのような利用法についても、覚えておくと便利でしょう。

＜レポート課題＞

(1) 階層的クラスタ解析と非階層的クラスタ解析を用いて、イネの表現型データ data.tr（15 ページ）に基づき、品種・系統を「5 群」に分類してみましょう。なお、階層的クラスタ解析については、いくつかのクラスタ間の距離の定義に基づいて分類を行ってみましょう。また、クロス集計表を用いて、これらが分集団（subpop）とどのような関係にあるかを調べてみましょう。

(2) k-medoids 法を用いて data.mk2（19 ページ）に含まれる 229 品種・系統の中から DNA マーカーに見られた変異に基づき代表的な 48 品種・系統を選んでみましょう。また、DNA マーカーに見られたに基づく主成分分析を行い、代表として選ばれた品種・系統のもつ遺伝変異を、主成分軸上の散布図として図示してみましょう。

(3) (2)選ばれた品種・系統について、各分集団（subpop）に属しているものがいくつずつ選ばれているか、table 関数を用いて集計してみましょう。また、(2)で選ばれた品種・系統について、図 19 （全品種・系統と選ばれた品種・系統の形質の変異のヒストグラム）を描き、全品種・系統のもつ形質の変異が(2)で選ばれた品種・系統によって、どの程度代表されているかを調べてみましょう。

提出方法：

- レポートは「pdf ファイル」として作成し、メール添付で提出する。
- メールは、「report@iu.a.u-tokyo.ac.jp 宛」に送る。
- レポートの最初に、「所属、学生番号、名前を忘れず」に。
- 提出期限は、5 月 7 日