バイオスタティスティクス基礎論
第 1 回　講義テキスト

岩田洋佳
aiwata@mail.ecc.u-tokyo.ac.jp

最近では、農学や生命科学の分野において、様々な種類のデータが大量に収集・蓄積されるようになってきています。こうしたデータに潜む未発見の「知」を見逃さずに確実に引き出すためには、研究の目的やデータのもつ性質に適した方法を用いてデータを解析する必要があります。

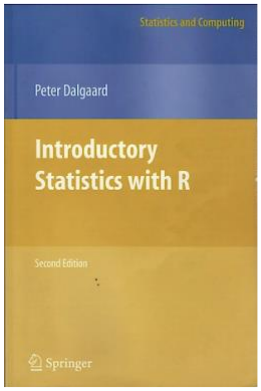
統計解析には様々な手法がありますが、各手法の特徴を把握し、解析の原理を理解し、得られた解析結果を適切に解釈できるようになるためには、相応の学習を必要とします。また、その学習をより効果的なものにするには、実際のデータを自分で解析してみるという経験も不可欠です。自分で計測したデータを解析してはじめて、講義や参考書で学んだことが明瞭に理解できるようになることは少なくなりません。

本講義は、受講生の皆さんが自らデータ解析を行い、統計解析のスキルを高めていくための「最初の第一歩」を提供することを目的としています。具体的には、今後の研究で必要となると考えられるいくつかの統計手法について、R を使った実践的なデータ解析の方法に重点をおいて解説していきます。本講義の目標は、回帰分析、分散分析、主成分分析などの汎用的な統計解析手法について、それを自分のデータ解析に利用するためのスキルを身につけること、さらには、より発展したデータ解析を行うための足場をかためることです。全 4 回と短い講義ではありますが、統計解析の面白さや巧みさについて興味をもってもらえるように講義を進めていきたいと思っています。

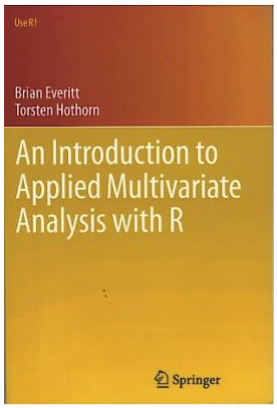
＜R＞

R は統計解析のためのフリーソフトウエアです（少しだけ正確にいうと、R とはコンピュータ言語の名称であり、パソコン上にソフトウェアとしてインストールされる R は R 言語を利用するための“環境”となります)。R には数多くの機能が備わっており、その利用場面は、統計解析だけでなく、データの前処理から、データの俯瞰、さらには、論文用のグラフ作成にまで及びます。また、パッケージ（package）として配布されている拡張プログラムをインストールすることで、様々な解析を容易に実行することができます。新しく開発された統計手法が R では比較的早く利用できるようになります。このようなことから、R を使うためのスキルは、農学や生命科学の研究者にとって非常に有用なものとなってきています。

なお、R については、現在、非常に多くの参考書が出版されています。私のおすすめの入門書は、以下の通りです。



Peter Dalgaard 著、Introductory Statistics with R (Statistics and Computing) Second Edition, Springer, 2008, ISBN: 978-0387790534



Brian Everitt, Torsten Hothorn 著、An Introduction to Applied Multivariate Analysis with R (Use R!), Springer, 2011, ISBN: 978-1441996497

＜R を用いた簡単な計算＞

R では、基本的には、コマンド（命令文）を順次入力しながら対話的に解析を進めていきます（ただし、実際に解析を行う場合は、R スクリプトとして一連のコマンドを先に入力しておき、それを実行する方が部分的修正や履歴の確認ができて便利です)。

ここでは、コマンド入力で簡単な計算を行いながら、R に慣れるところから始めて見ましょう。

R の最も簡単な利用方法は、簡単な算術表現を入力し、その答えを得ることです。例えば、

```
> 3 + 5 * 3
[1] 18
```

得られた結果をもとに次の計算をしたい場合には、次のように値を変数に代入しておきます。

```
> x <- 1 + 2
> x
[1] 3
```

代入しておいた値は、変数名を介して別の計算に用いることができます。

```
> x + 5 * x
[1] 18
```

関数を用いて様々な計算を行うことができます。

```
> abs(x)                          #  絶対値を求める
[1] 3
> sin(x)                          #  正弦（sine）を求める
[1] 0.14112
> atan(x)                  #  逆正接（arctangent）を求める
[1] 1.249046
> log(x)                          #  自然対数を求める
[1] 1.098612
> log10(x)                 #  底 10 の対数を求める
[1] 0.4771213
```

では、少し複雑な計算をしてみましょう。平均 μ、分散 σ^2 の正規分布の確率密度関数（図 1）は、

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(- \frac{(x - \mu)^2}{2\sigma^2} \right)$$

ですが、これを R で計算してみましょう。

```
> mu <- 3
> s2 <- 2
> x <- 5
> 1 / sqrt(2 * pi * s2) * exp(- (x - mu)^2 / (2 * s2))
[1] 0.1037769
```

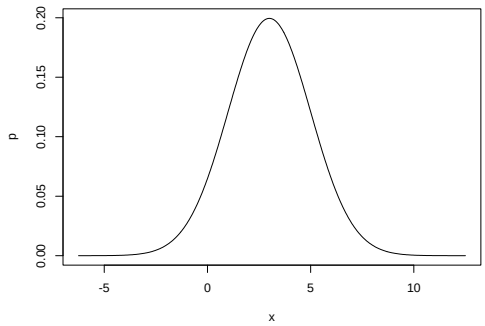


図 1. 平均 3、分散 2 の正規分布

確認のために正規分布の確率密度を計算する関数 dnorm で計算してみると同じ値が得られます。

```
> dnorm(x, mu, sqrt(s2))
[1] 0.1037769
```

＜ベクトルや行列を用いた計算＞

R の優れた点のひとつは、ベクトルや行列の演算を非常に簡単に実行できることです。ここでは、ベクトルや行列の演算を用いていくつかの要約等計量を計算してみましょう。

例えば、6 個の数値からなるベクトルを以下のように簡単に作成できます。なお、このデータは、6 品種・系統のイネの籾長を mm 単位で計測したデータです（データの出典は後述します）。

```
> length <- c(8.1, 7.7, 8.2, 9.7, 7.1, 7.3)       # mm scale
> length
[1] 8.1 7.7 8.2 9.7 7.1 7.3
```

同じ品種・系統の籾幅を計測したデータも入力し、籾長と籾幅の比を計算します。

```
> width <- c(3.7, 3.0, 2.9, 2.4, 3.3, 2.5)
> ratio <- length / width
> ratio
[1] 2.189189 2.566667 2.827586 4.041667 2.151515 2.9200000
```

まず、籾長と籾幅の比の平均を計算してみましょう。母平均の推定値は、

$$\sum_i^n x_i / n$$

として計算できます。ここで、x_i は i 番目のサンプルの値、n はサンプル数です。

```
> sum(ratio)                          # 総和を求める
[1] 16.69662
> length(ratio)                                  # ベクトルの長さ、すなわち、サンプル数を得る
[1] 6
> sum(ratio) / length(ratio)
[1] 2.782771
```

平均は、関数 mean を使って計算できます。

```
> mean(ratio)
[1] 2.782771
```

次に、分散を計算してみましょう。母分散の推定値は、

$$\sum_i^n (x_i - \bar{x})^2 \Big/ (n - 1)$$

として計算できます。ここで、$\bar{x}$ は先ほど計算した平均です。

```
> xbar <- mean(ratio)                    # 平均の代入
> (ratio - xbar)^2                       # 平均からの差の 2 乗
[1] 0.352338947 0.046700930 0.002008434 1.584819189 0.398483500 0.018831895
> sum((ratio - xbar)^2)                  # 平均からの差の 2 乗の和を計算
[1] 2.403183
> sum((ratio - xbar)^2) / (length(ratio) - 1)
[1] 0.4806366
```

分散は、関数 var を使って計算できます。

```
> var(ratio)
[1] 0.4806366
```

次に、共分散を計算してみましょう。2 変量 x と y 間の共分散の推定値は、

$$\sum_i^n (x_i - \bar{x})(y_i - \bar{y}) \Big/ (n - 1)$$

として計算できます。ここで、$\bar{x}$ および $\bar{y}$ は各変量の平均を表します。

```
> xbar <- mean(length)                   # 平均の代入
> ybar <- mean(width)                    # 平均の代入
> sum((length - xbar) * (width - ybar)) / (length(length) - 1)   #   共分散
[1] -0.1773333
```

なお、R の関数 cov を使って共分散を計算することもできます。

```
> cov(length, width)
[1] -0.1773333
```

共分散に続いて Pearson の積率相関係数（以下、相関係数）を計算してみましょう。相関係数を式で書くと、$\dfrac{\sum_i^n (x_{1i} - \bar{x}_1)(x_{2i} - \bar{x}_2)}{\sqrt{\sum_i^n (x_{1i} - \bar{x}_1)^2}\sqrt{\sum_i^n (x_{2i} - \bar{x}_2)^2}}$ となります。

```
> s12 <- sum((length - xbar) * (width - ybar))
> s1 <- sum((length - xbar)^2)
> s2 <- sum((width - ybar)^2)
> s12 / (sqrt(s1) * sqrt(s2))
[1] -0.3901388
```

式を見て分かるように、相関係数は共分散を両変数の標準偏差で割ったかたちになっています。実際に計算して確認してみましょう。

```
> cov(length, width) / (sd(length) * sd(width))
[1] -0.3901388
```

相関係数では、両変数の標準偏差で割ることにより基準化してあるために、共分散と異なり、計測値のスケールに影響されずに変数間の関係を把握できます。したがって、異なる尺度（重さと長さなど）で計測された変数間で関係の強さを比較するのに適しています。

なお、R の関数 cor を使って相関係数を計算することもできます。

```
> cor(length, width)
[1] -0.3901388
```

では、行列計算を用いて分散と共分散を計算してみましょう。まずは、length と width を結合して 6×2 の行列を作成します。

```
> x <- cbind(length, width)
> x
（結果は省略）
```

次に、関数 apply を用いて各列の平均を求めます。

```
> m <- apply(x, 2, mean)
> m
   length     width
8.016667 2.966667
```

求めた列平均を各列から引き算します。

```
> z <- sweep(x, 2, m)
> z
（結果は省略）
```

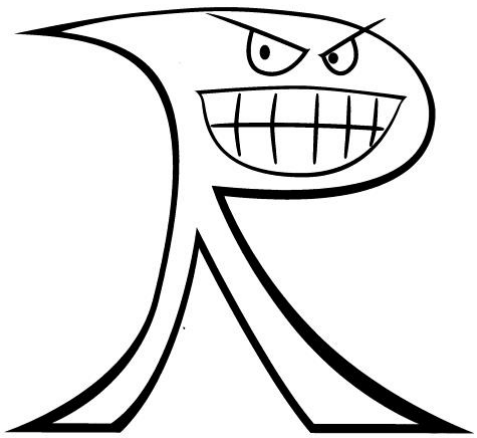
あとは行列の積を用いることで分散と共分散（分散共分散行列）を計算できます。

```
> t(z) %*% z / (nrow(z) - 1)
                length          width
length    0.8656667 -0.1773333
width    -0.1773333    0.2386667
```

対角成分が分散、非対角成分が共分散です。

分散共分散行列は関数 cov で計算することができます。

```
> cov(x)
                length          width
length    0.8656667 -0.1773333
width    -0.1773333    0.2386667
```



K.W.

＜外部データを読み込んで解析する＞

自分の研究のために R を利用する場合は、表計算ソフト等で整理されたデータを読み込んで解析する場合がほとんどだと思います。ここでは、他のソフトで保存されたデータを R に読み込み解析するための手順を説明します。

なお、ここでは、Zhao ら（2011; Nature Communications 2:467）がイネ遺伝資源を用いたゲノムワイドアソシエーション解析に用いられたデータ（Rice Diversity http://www.ricediversity.org/data/ からダウンロードできる）をデータ例として用います。

csv 形式で保存されたファイルの読み込みには read.csv という関数を用います。

```
> pheno <- read.csv("RiceDiversityPheno.csv")        # csv ファイルの読み込み
```

読み込んだデータのサイズやデータの一部を確認するには以下のようにします。

```
> dim(pheno)                                         #  データの次元を調べる
[1] 413   38
> head(pheno)                                        #  最初の 6 行を表示する
（結果は省略）
```

このデータには、各遺伝資源の由来などが記述されたファイルが別に存在します。ここでは、そのファイルを読み込んで pheno データに結合してみます。まずは、ファイルを読み込みます。

```
> line <- read.csv("RiceDiversityLine.csv")
> head(line)
（結果は省略）
```

line データの NSFTV.ID と pheno データの NSFTVID が対応しているので、この列の情報をもとに 2 つのデータを結合します。

```
> data <- merge(line, pheno, by.x = "NSFTV.ID", by.y = "NSFTVID")
> head(data)
（結果は省略）
```

＜読み込んだデータの解析＞

計測データセットには、実験上の都合から欠測したデータが含まれる場合が少なくありません。また、数少ない変数について解析をするだけでなく、たくさんの変数についてその分布や変数間の関係をみたい場合が少なくありません。ここでは、先ほど読み込んだデータを用いて、データ解析を行ってみましょう。

では、先ほどと同じようにして、籾長と籾幅の比を計算し、その平均を計算してみましょう。

```
> ratio <- data$Seed.length / data$Seed.width
                                                    # data 内の変数は$を使って指定する
> mean(ratio)
[1] NA
```

すると NA と表示されるだけで平均が計算できません。何故でしょうか。

これは、ratio に欠測値（R では NA として表す）が含まれているためです。

```
> ratio                                             #   中身を確認
（結果は省略）
```

このような場合は、na.rm というオプションを指定して計算します。

```
> mean(ratio, na.rm = T)                  # na.rm = T は NA を無視して計算せよの意味
[1] 2.752084
```

data 内の全ての変数について平均を求めるには以下のようにします。

```
> sapply(data, mean, na.rm = T)
（結果は省略）
```

数値データでないデータについては警告メッセージが表示されて計算結果は NA となります。

なお、次のコマンドを用いると、数値（numeric）データについては平均だけでなく、四分位点、最小値、最大値が表示され、因子（factor）データについては、各階級に属するサンプルの数え上げ結果が表示されます。

```
> summary(data)
（結果は省略）
```

では、全変数の総当たりで相関係数を計算してみましょう。

```
> cor(data)
 以下にエラー cor(data) :  'x' は数値でなければなりません
```

するとエラーメッセージが出て計算ができません。これは数値データと因子データが混在しており、因子データでは相関を計算できないためです。

そこで数値データの列だけを抜き出してみましょう。

```
> selector <- sapply(data, is.numeric)
                                        #  is.numeric 関数は数値データのときに TRUE を返す
> selector
（結果は省略）
> num.data <- data[,selector]
> head(num.data)
（結果は省略）
```

相関を計算してみます。

```
> cor(num.data)
（結果は省略）
```

ほとんどの組合せで結果が NA となってしまいます。これは先ほどと同様欠測値によるものです。

欠測値に対する対処の仕方を指定して再度計算してみます。

```
> cor(num.data, use = "pair")          # ペアワイズで欠測が無いサンプルを用いて計算
（結果は省略）
```

一部を除いて無事計算されました。一部の組合せでは欠測したサンプルを除くと一方の変数の分散が 0 になってしまい、相関が計算できないようです。

＜データの視覚化＞

実際に統計解析を行う前に、データをいろいろな角度から眺めてみることは非常に重要です。例えば、上述した平均や分散といった統計量は要約のための統計量であり、同じような平均や分散をもつ変数であっても、観察値の分布が大きく異なる場合もあります。したがって、まずはデータをじっくり眺めるということが、そのデータのもつ特性を理解するためにも非常に重要です。また、データの視覚化はデータ解析の結果を論文等にまとめる際にも必要です。ここでは、様々なデータ視覚化手法について説明します。

まず、視覚化手法の説明の前に、data 内にあるデータを直接呼び出せるようにしましょう。

```
> attach(data)                    #   data 内にあるデータを直接呼び出せるようにする
> search()                  # R の search path 内に data が登録されている
```

こうしておくことで、例えば、これまで data$Plant.height と指定していたところを、data$無しの Plant.height として入力できるようになります。

では、まずヒストグラムを描いてみましょう。

```
> hist(Plant.height)
```

stem-and-leaf プロットを描いてみましょう。

```
> stem(Plant.height)
（結果は省略）
```

こちらは図ではなくテキスト表示で結果が示されます。

箱ひげ図（box plot）を描いてみましょう。

```
> boxplot(Plant.height)
```

次に、いもち病抵抗性（Blast.resistance）についてヒストグラムを描いてみま

す。

```
> hist(Blast.resistance)
```

うまく分布が図示できているように見えますが、実は落とし穴があります。

いもち病抵抗性データは抵抗性の強さを 9 段階（0-9）のスコアで表されています。そこで、まずは、9 段階のどの階級に何品種・系統が含まれているのか集計してみましょう。

```
> t <- table(Blast.resistance)              #  各スコアをとるサンプル数を集計できる
> t
Blast.resistance
 0  1  2  3  4  5  6  7  8  9
 3 77 23 34 36 24 39 36 52 61
```

さきほど描いたヒストグラムでは全階級をうまく表せていなかったことが分かります。

上記のように table 関数を用いて集計されたデータから、棒グラフ（bar plot）を描くことができます。

```
> plot(t)       # table の結果で plot 関数を使うと棒グラフが描かれる
> plot(t, xlab = "Blast resistance scores", ylab = "Frequency")
                                                    #  棒グラフに軸タイトルを付ける
```

棒グラフは barplot 関数を用いて描くこともできます。ただし、上記の棒グラフと少し見た目が異なります。

```
> barplot(t)
```

円グラフを描くと各スコアの割合を図示できます。

```
> pie(t)
> pie(t, main = "Blast resistance")                  # 円グラフにタイトルを付ける
```

ここからは、2 変数間の関係を見て行きましょう。

```
> plot(Plant.height, Panicle.length)      #   最初の変数が横軸、2 番目が縦軸になる
```

回帰分析により直線をあてはめて重ね描きします。

```
> abline(lm(Panicle.length ~ Plant.height))
                #   lm は回帰分析を行う関数、abline は傾きと切片を指定して直線を描く関数
```

ラグ（織物）プロットを重ね描きします。分布の疎密を視覚化するのに便利です。

```
> rug(Plant.height, side = 1)             #   side = 1 は x 軸
> rug(Panicle.length, side = 2)           #   side = 2 は y 軸
```

では、少し複雑な図を描いてみましょう。散布図と箱ひげ図を併せて描いてみましょう。

```
> def.par <- par(no.readonly = T)         # 現在の描画パラメータを保存しておく
> layout(matrix(c(2, 0, 1, 3), nrow = 2, byrow = T),
                widths = c(2, 1), heights = c(1, 2), respect = T)
                # 2×2 の描画範囲をつくる。 左下、左上、右下（右上は描画されない）の順で描画
> plot(Plant.height, Panicle.length)      #   散布図を描く
> boxplot(Plant.height, horizontal = T)   #   x 軸の変数の箱ひげ図を描く
> boxplot(Panicle.length)                           #   y 軸の変数の箱ひげ図を描く
> par(def.par)                                      # 保存しておいた描画パラメータに戻す
```

Plant.height についても Panicle.length についても外れ値（outlier）が○で示されています。

2 変数の分布を同時に考慮しながら外れ値を見つけることもできます。次に描くのは 2 次元版の箱ひげ図です。

```
> require(MVA)                            # パッケージ MVA を読み込む(あらかじめインストールしておく)
> x <- cbind(Plant.height, Panicle.length)          # 2 つの変数を結合して x に代入
> x <- na.omit(x)                                   # 欠測値を除く
> bvbox(x, xlab = "Plant.height", ylab = "Panicle.length")
                                                    #   bivariate boxplot を描く
```

外側の楕円（fence 柵とよばれる）の外の点は外れ値の「可能性」があるデータ

です（柵の外に散布されただけでは外れ値とは限りません）。

2 変数間の関係を、カーネルを用いた平滑化（kernel smoothing）を用いて図示してみましょう。

```
> require("KernSmooth")
> d <- bkde2D(x, bandwidth = 4)
> plot(x)
> contour(d$x1, d$x2, d$fhat, add = T)
```

等高線のように表されているのがカーネルで平滑化された点の密度です。

では、この平滑化された密度を 3 次元で表示してみましょう。

```
> persp(d$x1, d$x2, d$fhat, xlab = "Plant.height",
                ylab = "Panicle.length", zlab = "density",
                theta = -30, phi = 30)
```

読み込まれた Zhao ら（2011）のデータには、形質データだけでなく、遺伝資源の遺伝的背景に関するデータも含まれています。遺伝的背景と形質の間にどのような関係があるのか、両データを併せて図示して調べてみましょう。

Sub.population という変数は、各遺伝資源の遺伝的背景の違いを表しています。これは Structure 解析（Pritchard et al. 2000, Genetics 155:945)を用いて推定されたものです。では、遺伝的背景と草丈や穂長にどのような関係があるのか視覚化して見てみましょう。

```
> pop.id <- as.numeric(Sub.population)
                              # 因子データである Sub.population を数値に変換
> plot(Plant.height, Panicle.length, col = pop.id) # 数値で色を指定している
> levels(Sub.population)   # 因子の水準を表示すると 6 つの分類があることが分かる
[1] "ADMIX"       "AROMATIC" "AUS"          "IND"          "TEJ"          "TRJ"
> legend(locator(1), levels(Sub.population),
                col = 1:nlevels(Sub.population), pch = 1)
    #   クリックした位置に凡例を加える。locator(1)がクリックした場所を座標値に変換する。
```

遺伝的背景の違いにより値にどのような違いがあるのかを箱ひげ図で示してみましょう。

```
> boxplot(Plant.height ~ Sub.population)
```

先ほど描いた散布図と箱ひげ図の組合せを、遺伝的背景の違いも分かるように作成し直してみましょう。

```
> def.par <- par(no.readonly = T)          # 描画オプションの保存
> layout(matrix(c(2, 0, 1, 3), nrow = 2, byrow = T),
          widths = c(2, 1), heights = c(1, 2), respect = T)
                                                  #  描画レイアウトを変更する
> plot(Plant.height, Panicle.length, col = pop.id)   # 遺伝的背景の違いで色分け
> boxplot(Plant.height ~ Sub.population,
          border = 1:nlevels(Sub.population), horizontal = T)
                                                  # 分集団毎に箱ひげ図を描画
> boxplot(Panicle.length ~ Sub.population, border = 1:nlevels(Sub.population))
> par(def.par)                                    # 描画オプションを元に戻す
```

草丈（Plant.height）と穂長（Panicle.length）に加え、止め葉の長さ（Flag.leaf.length）の 3 変数の間の関係がどのようになっているかをバブルプロット（bubble plot）によって確かめてみましょう。ここでは、バブルの大きさが止め葉の長さを表しています。

```
> symbols(Plant.height, Panicle.length,
          circles = Flag.leaf.length, inches = 0.1, fg = pop.id)
                                  # バブルで表したい変数を circles オプションで指定する
```

3 変数間の関係をスタープロット（star plot）として描いてみましょう。

```
> x <- data.frame(Plant.height, Panicle.length, Flag.leaf.length)
> stars(x)
                        # star plot の場合はあらかじめ描画したい変数を束ねておく
```

三角形の形の違いより大きさの違いが大きいことから、3 形質間の関係は強く、1 つの形質で大きいものは他の形質も大きい傾向が見てとれます。なお、スタープロットでは、4 つ以上の変数の関係を同時にみることができます。

3 変数間の関係を折れ線グラフとして描くこともできます。

```
> matplot(t(x), type = "l", lty = 1, col = pop.id)
                        # t(x)は x の転置を表す
                        # type = "l"で折れ線を指定。lty = 1 は線種を表す。
```

3 変数間の関係を総当たりの散布図で描いてみましょう。

```
> pairs(x, col = pop.id)
```

回帰直線を加えた少し複雑な散布図にしてみましょう。

```
> pairs(x, panel = function(x, y, ...) {
                points(x, y, ...)                                    # 点を散布する
                abline(lm(y ~ x), col = "gray")          # 回帰係数を描く
 }, col = pop.id)                                                    # 少し複雑ですね…
```

3 変数間の関係を 3 次元の散布図を描いてながめてみましょう。

```
> require(scatterplot3d)                    # scatterplot3d パッケージが必要
> scatterplot3d(Plant.height, Panicle.length, Flag.leaf.length, color = pop.id)
```

次に、散布図と star plot を重ねてみましょう。なお、散布図の点の位置は草丈（Plant.height）と穂長（Panicle.length）にしたがい、star plot ではマーカー遺伝子型をもとにした主成分分析のスコア（遺伝的背景の違いを定量化したもの）を表現することとします。また、star plot も色を付けしてみることにします。

```
> plot(Plant.height, Panicle.length, col = pop.id, pch = ".")
                                        # まずは点を散布する
> stars(cbind(PC1, PC2, PC3, PC4),                  #PC1 ～ PC4 は主成分スコア
                locations = cbind(Plant.height, Panicle.length),
                add = T, col.stars = pop.id)            # add = T は上書きするという意味
> legend(locator(1), levels(Sub.population), col = 1:nlevels(Sub.population), pch = 1)
                # マウスでクリックした位置に凡例を加える
```

逆に主成分スコアの散布図に対して形質の値を star plot として重ねて表示することもできます。

```
> plot(PC1, PC2, col = pop.id, pch = ".") # PC1 と PC2 で散布図を作成
> stars(cbind(Plant.height, Panicle.length, Flag.leaf.length),
                locations = cbind(PC1, PC2), add = T, col.stars = pop.id, len = 0.005)
                                        # 草丈、穂長、止め葉の長さを star plot で表示
> legend(locator(1), levels(Sub.population), col = 1:nlevels(Sub.population), pch = 1)
```

読み込まれているデータには、各遺伝資源の由来している場所の緯度経度のデータも含まれています。そこで、各遺伝資源の由来を世界地図上にマップして確認してみましょう。

```
> require(maps)                                    # map パッケージが必要
> require(mapdata)                  # mapdata パッケージも必要
> map('worldHires')                 #  世界地図をプロットする
> points(Longitude, Latitude, col = pop.id)
                                                   #  緯度経度を指定すると対応する場所に点をうてる
> legend(locator(1), levels(Sub.population), col = 1:nlevels(Sub.population), pch = 1)
```

上のコマンドでは、遺伝資源数よりもずっと少ない数の点しか描かれません。これは、同じ地域からの遺伝資源が互いに重なり合って表示されているためです。重なり合いを防ぐには関数 jitter で重なっている点を少しだけ動かします。

```
> map('worldHires')
> points(jitter(Longitude, 200), Latitude, col = pop.id)
                                                   #  関数 jitter で x 方向に少しずらす
> legend(locator(1), levels(Sub.population), col = 1:nlevels(Sub.population), pch = 1)
```

＜図のファイルへの出力＞

作成した図を論文やプレゼン用資料などを利用するためには、図を PDF ファイルなどに出力できると便利です。ここでは、簡単にその方法を説明します。

先ほど描いた図を map.pdf というファイルに出力してみましょう。

```
> pdf("map.pdf")                                    # pdf ファイルへの出力を指定
> map('worldHires')                          #  描画してもグラフウィンドウには表示されない
> points(jitter(Longitude, 200), Latitude, col = pop.id)
> legend(-175, 5, levels(Sub.population), col = 1:nlevels(Sub.population), pch = 1)
> dev.off()                                  #  重要！：かならず最後に出力ファイルを閉じる
null device
          1
```

上のコマンドを実行すると map.pdf というファイルが R の作業ディレクトリに出力されます。

関数 pdf では、出力する図のサイズを指定することができます。今回の図のように横長のほうが合っていて、かつ、大きなサイズで出力したほうがよい場合には、サイズを指定して出力したほうがきれいな図が描けます。

```
> pdf("map_large.pdf", width = 20, height = 10)    # 20 インチ×10 インチで出力
> map('worldHires')
> points(jitter(Longitude, 200), Latitude, col = pop.id)
> legend(-175, 5, levels(Sub.population), col = 1:nlevels(Sub.population), pch = 1)
> dev.off()
null device
          1
```

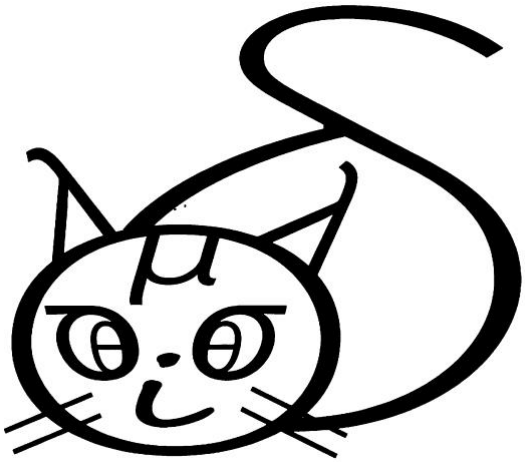
なお、複数の図を同じ pdf ファイルに繰り返し出力すると複数ページの pdf ファイルとして保存されます。同種の図を繰り返し大量に出力したい場合には、1 つの pdf ファイルにまとめておく方が便利かもしれません。

＜レポート課題＞

講義で学んだ様々なデータ視覚化法を用いて形質間の関係や、形質と遺伝的背景間の関係について図を描いてください。また、描いた図から読み取ることができる関係について記述してください。

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、4 月 20 日



K.W.