



来て見て使って：R  
～データ入力から計算まで～

20090617

森林生態生理学 research 分野 M2 徳本雄史

内容

1. 拡張子の表示
2. データの保存・R への読み込み
3. データの集計・計算
4. ちょっと遊んでみよう！(時間があれば)+ 補足

**0. おことわり**

今回はほぼ生データを使って、いろいろ計算してみます。

“初級編”とするには面倒なことが多いですがついてきてください。

ちなみに、今回使用するデータは、3年生実習の毎木調査のデータを改変したものです。

**1. 拡張子の表示**

○Windows XP: フォルダ > ツール > フォルダオプション > 表示 > 詳細設定の「登録されている拡張子は表示しない」のチェックを外す。

○Windows Vista: コントロールパネル > フォルダオプション > 表示 > 詳細設定の「登録されている拡張子は表示しない」のチェックを外す。

○Mac OS X: Finder 環境設定... > 詳細の「すべての拡張子を表示」をチェック

**2. データの保存・R への読み込み**

R\_lecture\_2009 フォルダの中にある『plot\_data.xls』を開く(欠損値には「NA」と入力する)。

ファイル > 別名で保存 > その他の形式 > ファイルの種類を選択 > .txt 形式で保存

→フォルダの中に plot\_data.txt というファイルが出てきましたか？

次にそのファイルを R で読み込んでみましょう。

R を起動させ、次のコマンドを入力。

```
> setwd("D:/Data/R_lecture_2009")      #読み込むフォルダを選択
> d <- read.table("plot_data.txt", header = T)
      # "d" という名前にデータを格納
```

データが読み込めるか確かめてみましょう。

```
> d          #これだけ！
```

### 3. データの集計・計算

#### 3. 1 集計

Rにあるいろいろな関数を使って、データをいろいろな角度から見てみましょう。

まずは、データ全体の要約を見る関数 `str()` を使って、

```
> str(d)
```

データ数が 298 で、変数が 5 つで、などなど。

その下には、変数を見ることができます。

変数だけだと、`names` 関数を使っても見ることができます。

```
> names(d)
```

では、次に、ある変数の中には、どんな分類が何個あるかを見ていきましょう。

具体的には、データの中にある樹木種には何があるのか、そしてあれば何個あるのかを見ていきます。

ここでは関数 `levels`、`nlevels`、`table` という関数を使います。

```
> levels(d$species)    # "species" の中にある分類名  
> nlevels(d$species)   # 分類は何個あるのか  
> table(d$species)     # それぞれの種が何個あるのか
```

次に見たいデータだけを取り出してみましょう。

ここでは、コナラだけのデータについて取り出してみます。

```
> d[d$species == "konara",]    # species が "konara" の行のみを取り出す
```

#### 3. 2 計算

次にコナラの 2008 年の GBH の平均値と標準偏差を計算してみましょう。

```
> mean(d[d$species == "konara", "gbh_2008"])  
# species が、"konara" の、gbh_2008 の行の平均値を計算する。  
> sd(d[d$species == "konara", "gbh_2008"])  
# 標準偏差を計算する。
```

一個一個やっていくのは面倒なので、`tapply` 関数を使って、一気に計算してみましょう。

```
> tapply(d$gbh_2008, d$species, mean)
```

では、R 上で周囲長から胸高直径(DBH)と胸高断面積(BA)を求めて、ファイルの中に入れてみましょう。

```
> d$dbh_2008 <- d$gbh_2008 / pi          #周囲長 ÷ π  
> d$ba_2008 <- (d$gbh_2008 / 2)^2 * pi  #半径 × 半径 × π
```

データがどう変わったか確認してみましょう。以前のように“d”だけでもいいのですが、関数 `head`, `tail` を使ってみてみましょう。

```
> head(d)          #データの上から 6 行目まで表示  
> tail(d)         #データの下から 6 行目まで表示
```

ここで一旦上書き保存してみます。

```
> write.table(d, file = "plot_data_sec.txt", quote = F,  
+ row.names = F, sep = "¥t")
```

フォルダに戻ってファイルが保存されているか確認してみましょう。

データの計算などはエクセルでもできますが、慣れてくればこちらの方が早い時もあります。

#### 4. ちょっと遊んでみよう！

毎木調査のデータが 2 年分あるので、1 年間の樹木の成長量を計算してみましょう！

まずは、求めていなかった 2009 年の DBH、BA を求めましょう。

以前打ったコマンドは残っているので、上キーを押して、2008 年の 8 を 9 に変えましょう！

```
> d$dbh_2009 <- d$gbh_2009 / pi          #周囲長÷π  
> d$ba_2009 <- (d$gbh_2009 / 2)^2 * pi  #半径×半径×π
```

データを確認してみましょう。

```
> head(d)
```

次に、1年間でどれだけ成長したかを計算してみましょう(図-1)。

```
> d$growth_dbh <- d$dbh_2009 - d$dbh_2008      #DBH の成長量  
> d$growth_ba   <- d$ba_2009 - d$ba_2008     #BA の成長量
```

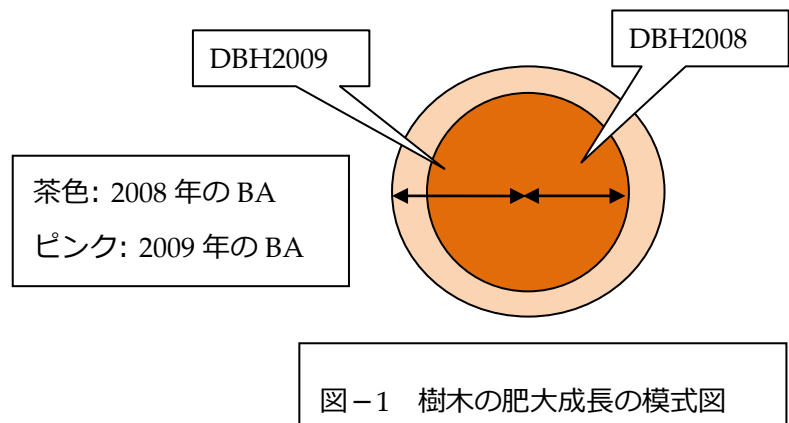
では、樹種によって成長は違うのでしょうか???

前に使った `tapply` 関数を使って、

```
> tapply(d$growth_dbh, d$species, mean)  
> tapply(d$growth_ba, d$species, mean)
```

最後に、

```
> plot(d$species, d$growth_dbh, las = 2)      #las は軸の表示の向きです
```



補足：

### 1. ファイル形式

今回は.txt 形式のタブ区切りを使ってきました。

エクセルの名前を付けて保存の時に、カンマ区切り(.csv)形式でも保存できます。

.txt だと

```
> read.table("*****.txt", header = T)
```

.csv だと

```
> read.csv("*****.csv")
```

### 2. write.table について

今回指定した式の中で、ファイル名以外にもつらつら書いてました。

○quote = は、一つ一つのデータをクォーテーションマーク(“”)で区切るかどうか。

○row.names = はいちばん左の行番号を入れるかどうか。

○sep = はデータの区切りをタブ区切り(“\t”)にするのか、コンマ区切り(“,”)にするのかを指定します。

いろいろ試してみてください。

### 3. attach()

データの項目を、「ファイル名\$項目」とせずに扱うことができる関数です。

たとえば一年間の DBH 成長を見たい時には、

これまで            > d\$growth\_d

使うと                > growth\_d

ファイル名がどうしても長くなってきたときなどは、よく使えます。

ただし、一つの解析が終わったあとには

```
> detach()
```

で項目を消す方がベターです。(他の変数、関数と重なってしまうため。)