Chapter 1. An introduction to R

2007/11/12 (Mon.) 飯島勇人[*1*2 *3]

はじめに断っておくべきこと

- この本では「代入」記号として、=が用いられています。しかし=はそれが使われた階層でしか有効ではありません。オンラインヘルプによれば、

  Rjpwiki からの引用

  The operators <- and = assign into the environment in which they are evaluated. The operator <- can be used anywhere, whereas the operator = is only allowed at the top level (e.g., in the complete expression typed at the command prompt) or as one of the subexpressions in a braced list of expressions.

  とされています。すなわち、普遍性の高いのは<-であるということです。そのため、私のレジュメ中では基本的に=は<-に置き換えます。
- 作図の際の記述方法として、できるだけ y ~ x という書き方を用います。これは線形モデル式の書き方として一般的であり、多くの作図方法にも適用できるためです。
- 必要に応じて、本文にないことを補っています。

1.1 この章の目的

1. R でのデータの要約方法と作図方法を簡単に示す
2. Monte Carlo シミュレーションを行うプログラムを書くための環境としての R の使い方を示す。

1.2 R での簡単なデータの見方

1.2.1 使用データの導入

この章で用いるデータ studentdata は R のパッケージである LearnBayes の中に含まれている。

```
> library(LearnBayes) #LearnBayes の読み込み
> data(studentdata) #LearnBayes 内の studentdata の読み出し
```

このデータはある大学の統計学を受講している学生に対する質問の答えであり、性別、身長、就寝の時間などが示されている。なお、データフレームの各列にアクセスするのを容易にするため、これ以降ではデータフレームをアタッチする。

```
> studentdata[1, ] #1 行目の表示
```

[*1] 北海道大学大学院農学研究院専門研究員
[*2] 連絡先: hayato-i@for.agr.hokudai.ac.jp または http://www.geocities.jp/iijima_web/index.html
[*3] 本ゼミのサポートページ: http://www7.atwiki.jp/hayatoiijima/pages/37.html

```
> attach(studentdata) #これ以降はラベル名だけでデータが取り出せる
```

1.2.3 データの見方

得られたデータを概観する方法としては、統計要約量を見る、あるいは作図を行う方法がある。

- カテゴリーに含まれるデータ数を見る: `table(Drink)`
- 列同士で演算を行い、新たな関係を検討する: `hours.of.sleep <- WakeUp - ToSleep`
- 統計要約量を見る: `summary(hours.of.sleep)`
- あるデータの分布を見る: `hist(hours.of.sleep, main="")`

1.2.4 群間比較のための R のコマンド

- 箱ひげ図: `boxplot(hours.of.sleep ~ Gender)`
- ある群だけを取り出す場合: `[ ]` を使う。例えば女性のデータだけ抜き出す場合、
  `> female.Haircut <- Haircut[Gender=="female"]`
  男性なら
  `> male.Haircut <- Haircut[Gender=="male"]`

1.2.5 関係性を検討するための R のコマンド

- 散布図: 2 変量の関係を検討するための作図コマンド。例えば睡眠時刻と睡眠時間の関係は、`plot(hours.of.sleep ~ ToSleep)` で作図することができる。ちなみに、
  `plot(jitter(hours.of.sleep) ~ jitter(ToSleep))`
  とすると、点を微妙にずらして描画することができる。
- 回帰直線: `lm()` によって回帰を行い、`abline()` によって回帰直線を描画することができる。

```
> fit <- lm(hours.of.sleep ~ ToSleep)
> fit #ずらずらと回帰の結果が出てくる
> abline(7.9628, -0.5753) #切片と傾きを入れるだけで描画してくれる

#プログラミング化を考えるなら、こうした方が汎用性が高い
> coef <- fit$coef
> abline(coef[1], coef[2])
```

1.3 R でのプログラミング〜T 検定統計量の頑健性の探索を通して〜

1.3.1 T 検定統計量の概説と問題点

T 検定統計量は、2 群の平均値の差を検定するためによく用いられる（いわゆる t 検定)。X と Y と言う 2 群からデータを得たときに、検定統計量 T は

$$T = \frac{\bar{X} - \bar{Y}}{s_q \sqrt{(1/m + 1/n)}}$$

ここで s_q は X と Y をプールした標準偏差であり、

$$s_q = \sqrt{\frac{(m-1)s_x^2 + (n-1)s_y^2}{m+n-2}}$$

と定義される。帰無仮説の元で、検定統計量 T は、

- x と y が正規分布する集団から独立にランダムサンプリングされている
- x と y の標準偏差、すなわち σ_x と σ_y は等しい

が満たされるとき、自由度 m + n - 2 の t 分布に従う。帰無仮説は

$$|T| \geq t_{n+m-2,\alpha/2}$$

のときに棄却される。しかし実際には、これらの仮定が疑われるような状況でもしばしば t 検定が行われている。そのため、これらの仮定が崩れた状況での T 検定統計量の頑健性、あるいは感受性は興味のある問題である。これを検討するために有効なのが Monte Carlo シミュレーションである。そこで以下では、Monte Carlo シミュレーションによって T 検定統計量の頑健性を検証する。

1.3.2 t 統計量を計算する関数を書く

まずは X と Y の 2 群から、正規分布に従うデータを 10 個取ってくる。乱数の充実も R の優れた点の一つである。

```
> x <- rnorm(10, 50, 10) #(サンプル数，平均，標準偏差)
> y <- rnorm(10, 50, 10)
```

わかっていることではあるが、x と y のサンプル数を計算する（できるだけ汎用性のある書き方をすることがプログラミングには重要である)。

```
> m <- length(x) #length() は要素数を計算する関数
> n <- length(y)
```

同じようにして、t 検定に必要なものを計算していく。

```
> sp <- sqrt(((m - 1)*sd(x)^2 + (n - 1)*sd(y)^2)/(m + n - 2))
> t <- (mean(x) - mean(y))/(sq*sqrt(1/m + 1/n))
```

これらを組み合わせると、検定統計量 T を計算するのに必要なプログラムをかくことができる。

```
> tstatistic <- function(x,y)
+   {
+   m <- length(x)
+   n <- length(y)
+   sp <- sqrt(((m - 1)*sd(x)^2 + (n - 1)*sd(y)^2)/(m + n - 2))
+   t <- (mean(x) - mean(y))/(sp*sqrt(1/m + 1/n))
+   return(t)
+   }
```

架空のデータを使って動作を試してみる。

```
> data.x <- c(1, 4, 3, 6, 5)
> data.y <- c(5, 4, 7, 6, 10)
> tstatistic(data.x, data.y)

[1]  -1.937926
```

1.3.3 Monte Carlo シミュレーションのプログラミング

モンテカルロ法（Monte Carlo method, MC）とはシミュレーションや数値計算を乱数を用いて行なう手法の総称である。

通常、（2 群比較における）真の有意水準（α^T）は

- 設定する有意水準 α
- （データを取る）集団の形（正規、歪んだ、長く尾を引くなど）
- 2 つの集団の分布の広がり（ようはばらつき）
- サンプルサイズ

に依存するので、

$$\alpha^T = P(|T| \geq t_{n+m-2,\alpha/2})$$

と定義できる。これを求めるためには、

1. 第一の集団からデータ x_1,...,x_m、第二の集団からデータ y_1,...,y_m を取る。
2. 2 つのサンプルから検定統計量 T を計算する。
3. $|T|$ が危険率を超え、帰無仮説が棄却されるかを決定する。

というプロセスを何度も繰り返し、以下の方法で真の有意確率を決定する。

$$\hat{\alpha}^T = \frac{\text{number of rejections of } H_0}{N}$$

$|T| \geq t_{n+m-2,\alpha/2}$ を R で実行するためには、検定統計量 T を計算し、`abs(t) > qt(1 - alpha/2, n+m+2)` を実行すればよい。`qt()` は自由度 n+m+2 のときの t 分布における 1 - alpha/2 分位数を算出する関数であり、得られた t の絶対値（`abs()`）がそれより大きいかどうかで判定できる。以上を踏まえ、Monte Carlo シミュレーションを R 上で行うプログラムを以下に示す。

```
> alpha <- 0.1; m <- 10; n <- 10 #alpha と m と n に値を代入
> N <- 10000 #繰り返し回数を設定
> n.reject <- 0 #帰無仮説が棄却された回数を 0 に設定
> for (i in 1:N)
+  {
+  x <- rnorm(m, 0, 1)
+  y <- rnorm(n, 0, 1)
+  t <- tstatistic(x,y)
```

```
+  if (abs(t) > qt(1 - alpha/2, n + m - 2))
+  n.reject <- n.reject + 1 #
+  }
> true.sig.level <- n.reject/N #真の有意確率を算出
```

1.3.4 異なる仮定の下での有意確率の挙動

今回は 5 パターンについて検証を行った。

1. 正規分布から得られた平均も標準偏差も変わらないデータ。
2. 正規分布から得られた平均は同じだが標準偏差が異なるデータ。
3. 両データとも自由度 4 の t 分布から得られたデータ。
4. 平均 1 の指数分布から得られたデータ。
5. 平均 10、標準偏差 2 の正規分布から得られたデータと平均 10 の指数分布から得られたデータ。

結果は p.13（当然多少の誤差は生じます)。特に分布形が異なる場合、その実際の有意確率は大きくなってしまっている。これを図で示すためには、以下のようにする（本文のとおりではうまくいきません)。

```
> m <- 10; n <- 10; N <- 10000
> tstat <- numeric(0)
> for (i in 1:N) {
        x <- rnorm(m, mean=10, sd=2)
        y <- rexp(n, rate=1/10)
        tstat[i] <- tstatistic(x, y)
        }
> plot(density(tstat), xlim=c(-5, 8), ylim=c(0, 0.4), lwd=3)
> lines(density(tstat)$x, dt(density(tstat)$x, df=18))
> legend(4, 0.3, c("exact", "t(18)"), lwd=c(3, 1))
```