Rで学ぶベイズ統計学入門
第1章　R入門

茨城大学工学部
佐々木稔

あらまし

- Rとは
  - 統計計算やグラフィック表示を行う言語、環境

- 数多くの統計手法を簡単に使うことができる
  - 線形・非線形モデル、統計検定、時系列解析など
  - 基本パッケージのみでも多くの解析手法が存在
  - パッケージの追加で他の手法も使える

この章での目標

- Rでデータの要約、グラフ化するための方法
  - 学生アンケートで得られたデータセットを利用
- プログラミング環境としてのRの利用方法
  - モンテカルロ・シミュレーション
  - 通常の仮説から逸脱した母集団の検定

- データ分析手法、シミュレーションの利用
  - ベイズモデルの推定に役に立つ

データセット

- 学生データセット
  - Bowling Green 州立大学での統計学初級クラス受講生657名全員に行ったアンケート結果（タブ区切り）
- アンケート内容（一部）
  - 性別
  - 身長（インチ）
  - 1から10までの自然数の中からひとつ選ぶ
  - 昨晩の就寝時間
  - 今朝目覚めた時間
  - 夕食時飲みたいのは、水、炭酸飲料、ミルクのどれ

Rにデータを読み込む

- データセット（ファイル名 “studentdata.txt”）
  - テキストデータで、列間はタブ区切り
  - データの先頭行は、列名が入ったヘッダ部分
- Rによるデータの読み込み
  - read.table()関数を使う
  - 第2引数 sep は区切り文字
  - 第3引数 header はヘッダが先頭行にあるか

```
> studentdata <- read.table(“studentdata.txt”,
+    sep = “¥t”, header = TRUE)
```

パッケージからのデータ読み込み

- パッケージ LearnBayes に studentdata が存在
- パッケージの利用方法
  - library(パッケージ名)
  - メニューバーの「パッケージ」から「パッケージの読み込み」を選択し、使用するものを選択
- パッケージにあるデータの読み込み
  - data(studentdata)
  - 変数名はパッケージ内で設定済み
  - achievement, sluggerdata, soccergoals などあり

データの表示

- データの表示方法
  - 変数名[行番号, 列名（列番号）]で指定
  - 変数studentdataの1行目を表示

```
> studentdata[1,]
  Student Height Gender Shoes Number Dvds ToSleep WakeUp Haircut Job Drink
1       1     67 female    10      5   10    -2.5    5.5      60  30 water
```

  - 変数studentdataの１４行目～１６行目を表示

```
> studentdata[14:16,]
   Student Height Gender Shoes Number Dvds ToSleep WakeUp Haircut Job Drink
14      14   63.5 female    20      6    3    -1.0    7.5     110  27  milk
15      15   72.0   male     5     NA   83     2.5    8.5      15  28   pop
16      16   65.0 female    40      7   50    -0.5    7.0      15   0 water
```

データの列名の表示

- 変数内の列名を変数として使うには
  - attach() 関数を使用する
  - 繰り返し呼び出すと、少し怒られます

```
> attach(studentdata)
The following object(s) are masked from 'studentdata (position 3)':

    Drink, Dvds, Gender, Haircut, Height, Job, Number, Shoes,
    Student, ToSleep, WakeUp
```

グループ別に要約

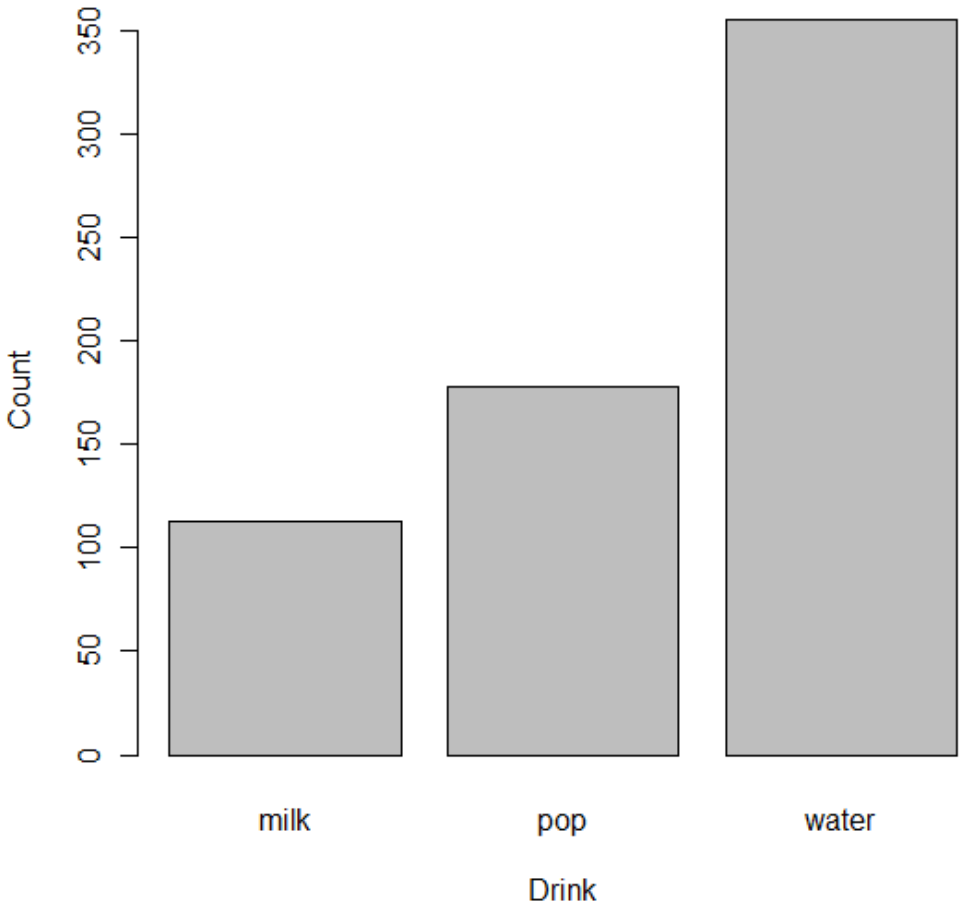
- カテゴリカル変数をカテゴリ毎に集計
  - 変数 student 内の変数 Drink
  - 集計は table() 関数を使用する

```
> table(Drink)
Drink
 milk   pop water
  113   178   355
```

  - 集計結果をバープロットでグラフ化する
  - 棒グラフの表示は barplot() 関数を使用する

```
> barplot(table(Drink), xlab = "Drink", ylab = "Count")
```

棒グラフで表示



睡眠時間の分析

- データセットには睡眠時間の質問はない
  - 起床時間と就寝時間の差を求める

```
> attach(studentdata)
> hours.of.sleep <- WakeUp - ToSleep
```

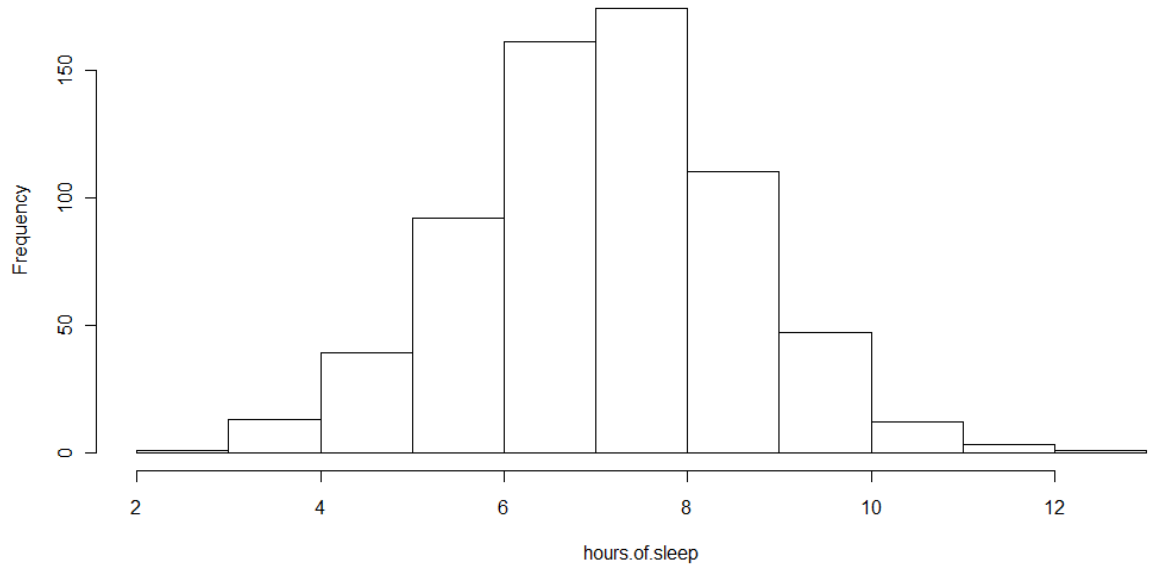
- 量的変数の要約
  - summary() 関数を使用する

```
> summary(hours.of.sleep)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
  2.500   6.500   7.500   7.385   8.500  12.500   4.000
```

ヒストグラムの表示

- 睡眠時間の分布を確認
  - ヒストグラム表示は hist() 関数を使用する

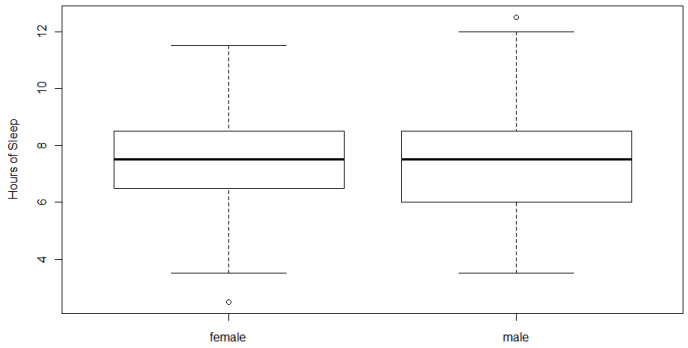
```
> hist(hours.of.sleep, main ="")
```



グループ同士で比較

- どの量的変数も男女差の比較が可能
  - 男性は女性よりも長く寝る傾向を調べる
  - 男女の睡眠時間を表す並行箱ヒゲ図を作成
  - 並行箱ヒゲ図は boxplot() 関数を使用する

```
> boxplot(hours.of.sleep ~ Gender, ylab = "Hours of Sleep")
```



- 男女間での散髪代の分析
- 男女それぞれの散髪代を抽出
  - 論理演算子を使用する

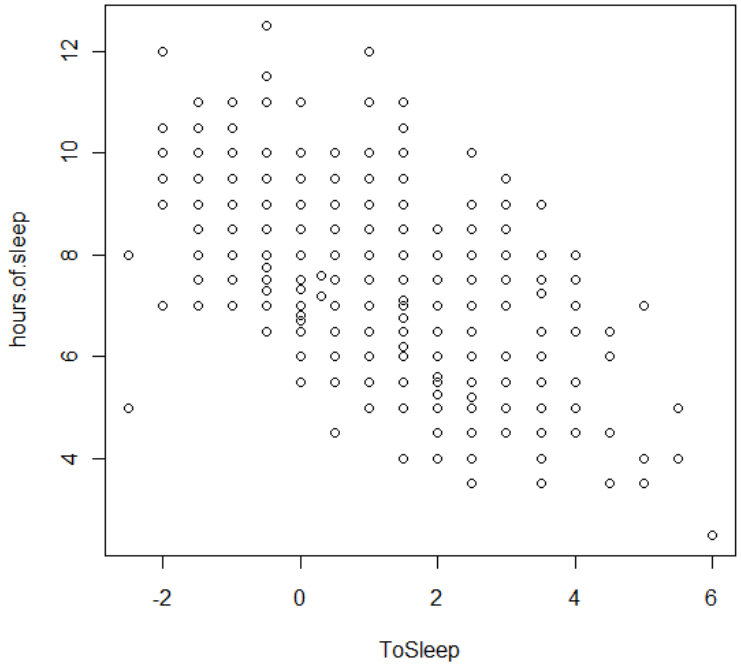
```
> female.Haircut <- Haircut[Gender == "female"]
> male.Haircut <- Haircut[Gender == "male"]
> summary(female.Haircut)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00   15.00   25.00   34.08   45.00  180.00   19.00
> summary(male.Haircut)
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
   0.00    0.00   12.00   10.54   15.00   75.00    1.00
```

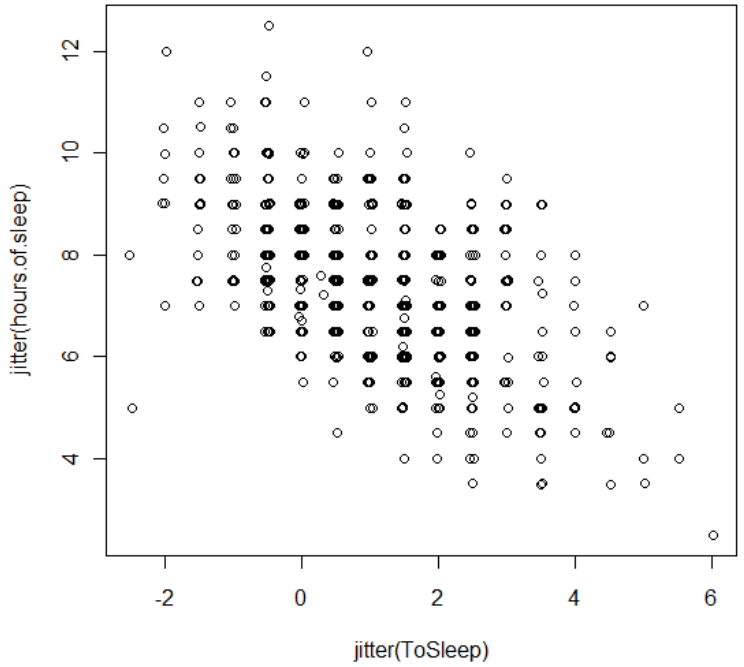
関連性を調べる

- 睡眠時間の長さと就寝時間との関連性
  - 変数 hours.of.sleep と ToSleep の関係
- 散布図を使って表示
  - plot(ToSleep, hours.of.sleep)
    - 同じ点が重なって分かりにくい
  - plot(jitter(ToSleep), jitter(hours.of.sleep))
    - jitter() 関数でノイズを加えて違いを出す

jitter関数による違い

```
> plot(ToSleep, hours.of.sleep)
> plot(jitter(ToSleep), jitter(hours.of.sleep))
```





直線への当てはめ

- 最小二乗法による直線近似
  - lm() 関数を使う

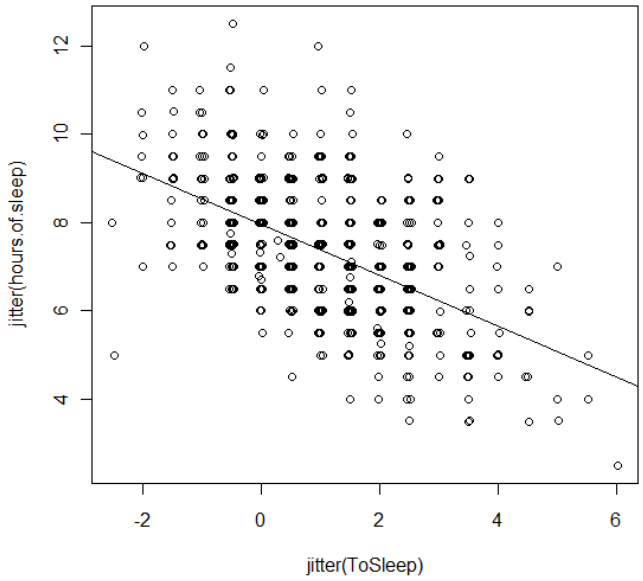
```
> fit <- lm(hours.of.sleep ~ ToSleep)
> fit

Call:
lm(formula = hours.of.sleep ~ ToSleep)

Coefficients:
(Intercept)     ToSleep
     7.9628      -0.5753
```

- 直線の傾き -0.58
  - 就寝時間が1時間遅くなると、睡眠時間がおおよそ30分短くなる

直線への当てはめ

- 近似曲線は散布図に追加できる
  - 散布図を表示した後で、abline() 関数を使う

```
> abline(fit)
```



t統計量の頑健性の分析

- 2つの独立した標本 $x_1, \ldots, x_m$ と $y_1, \ldots y_n$
  - それぞれの母集団平均が等しいと仮説
    - $H_0 : \mu_x = \mu_y$
  - $\bar{X}, \bar{Y}$ を x, y の標本平均、s_x, s_y を標準偏差
  - 仮説 H_0 に基づく標準的な t 統計量 T
    - 2つの条件を満たすと、自由度 $m+n-2$ の t分布になる
    - 条件1：正規分布からのランダム標本
    - 条件2：母集団の各標準偏差 σ_x, σ_y は等しい

$$T = \frac{\bar{X} - \bar{Y}}{s_p \sqrt{1/m + 1/n}} \qquad s_p = \sqrt{\frac{(m-1)s_x^2 + (n-1)s_y^2}{m+n-2}}$$

仮説の検定

- 検定統計量 T による仮説 H_0 の（両側）検定
  - 以下の不等式を満たす場合は、仮説 H_0 を棄却

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

  - $t_{df,\alpha}$ は自由度 df の t 分布における $(1-\alpha)$確率点
  - α は t 検定の有意水準
- t統計量は仮説が疑わしい場合にも利用
  - 強いランダム性のある分布の時の真の α は？
  - 等分散の仮定が成り立たない時の真の α は？
    - これらの疑問はモンテカルロ・シミュレーションで分析

t統計量の計算

- t統計量を計算する関数 tstatistic を定義
  - テキストファイル tstatistic.R に保存

```
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.stat <- (mean(x) - mean(y)) / (sp * sqrt(1/m + 1/n))
   return(t.stat)
}
```

t統計量の計算

- ベクトル形式の仮想データで計算
  - tstatistic() は前スライドの関数

```
> data.x <- c(1, 4, 3, 6, 5)
> data.y <- c(5, 4, 7, 6, 10)
> tstatistic(data.x, data.y)
[1] -1.937926
```

モンテカルロ・シミュレーション

- 母集団が正規分布や等分散ではない場合
  - 真の有意水準の推定（α^T を計算）

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

- α^T のシミュレーションアルゴリズム
  - 第1母集団から $x_1, \ldots, x_m$ を抽出
  - 第2母集団から $y_1, \ldots, y_n$ を抽出
  - 2つの標本から t 統計量を計算する
  - |T| が棄却値を超えていれば、H_0 を棄却
  - 上記の判定を N 回繰り返し、真の有意水準を推定

$$\alpha^T = H_0\text{の棄却回数}/N$$

異なる仮説下での真の有意水準の挙動

- 母集団が異なる場合の有意水準 α^T を分析
  - 平均 0 で、$\sigma_x = \sigma_y = 1$ の正規分布
  - 平均 0 で、$\sigma_x = 1, \sigma_y = 10$ の正規分布
  - 自由度 4 で広がり方の等しい T 母集団
  - $\mu_x = \mu_y = 1$ の指数母集団
  - 正規母集団 ($\mu_x = 1, \sigma_x = 2$) と指数母集団 ($\mu_x = 10$)

真の有意水準の推定結果

- 有意水準 $\alpha = 0.1$ と比較
  - シミュレーション標準誤差は 0.003
- 推定の結果
  - 広がり方が等しい場合 α とほぼ等しい値
  - 広がり方が異なる場合は α を超える値

母集団	真の有意水準
広がり方の等しい正規母集団	0.0986
広がり方の異なる正規母集団	0.1127
広がり方の等しい自由度4のt分布	0.0968
広がり方の等しい指数母集団	0.1019
広がり方の異なる正規母集団と指数母集団	0.1563

正規母集団と指数母集団

- 真の有意水準よりも50%大きい値
  - t統計量の正確なサンプリング分布を求める
  - 正規母集団と指数母集団の t 統計量

```
> m=10
> n=10
> my.tsimulation <- function()
+     tstatistic(rnorm(m,mean=10,sd=2), rexp(n,rate=1/10))
```

  - replicate()関数を使って、10,000回繰り返す

```
> tstat.vector <- replicate(10000, my.tsimulation())
```

t統計量の密度分布

- 自由度18のt分布密度関数を同時にプロット
  - サンプリング分布が右に歪んでいる
  - そのため、真の有意水準が大きくなっている