言語情報処理入門（2013 年度前期木曜５時限目）

■ R による統計・言語処理入門（担当：杉浦）

1. ６月　６日　R による統計処理入門（R の基本的な使い方の説明）
2. ６月１３日　R によるテキスト処理入門（R を使ったテキスト処理の紹介）
   ・（一週とばして、というか山下先生の第３回目がここにきます）
3. ６月２７日　R による言語情報処理（分散分析と具体的な分析事例の紹介）

---

1. Rによる統計処理入門

WikiPedia

> **R 言語 (アールげんご) は、オープンソースでフリーソフトウェアの統計解析向けプログラミング言語、及びその開発実行環境である。**

1.1 はじめに

■インストール

・RjpWiki を検索
・http://www.r-project.org/
・（RStudio もインストールすると便利）

■起動と終了

起動：「スタート」から選ぶ
終了：「メニュー」の「ファイル」から「終了」を選ぶか、コマンド
　　（作業スペースの保存　＝＞　パソコン室での作業の場合、自分の USB メモリーに）

■コンソールでコマンド

「R Console」という窓
プロンプト　　の右にコマンド（命令）を書く

コマンドを組み合わせて、ひとまとめにしてプログラムにする（「関数」とよぶ）

■データを「処理」する

統計処理
テキスト処理

処理する前に、データを整形する
データは、文字か数字
数字のデータを整形する
　エクセルなどで、プログラムの「想定している」形式に合わせる。
文字のデータを整形する
　コーパス・データのフォーマット
　　CHILDES の CHAT フォーマットなど

■習得しないといけないこと

自分に必要なコマンド・プログラムは何か

1.2 R を使った処理の実例紹介

■「学部の違う二つのクラスで同じ試験を実施した。学部によって成績に差があるか？

1. エクセルにあるデータをタブ区切りのテキストで保存。（テキスト（タブ区切り）(*.txt)
2. データを読み込む。
    ・データフレーム名 <- (), header = T)

```
classes <- read.table(choose.files(), header = T)
```

1. t 検定
    ・差がないと仮定する（差がないことがあり得るか）
    ・クラスごと別々の受講生のデータ（()）

```
t.test(classes$ClassA, classes$ClassB)
```

・"Welch Two Sample t-test" （等分散でなくてもよい）

■「ある教材で学習したら成績が上がるか？：事前テストより事後テストの方が良い成績か？」

1. エクセルにあるデータをタブ区切りのテキストで保存。（テキスト（タブ区切り））
2. データを読み込む。
    ・データフレーム名 <- (), header = T)

```
prepost <- read.table(choose.files(), header = T)
```

1. t 検定
    ・差がないと想定する（差がないことがあり得るか）
    ・それぞれの学生ごとにデータがある（）：

```
t.test(prepost$pre, prepost$post, paired=T)
```

1.3 用意されたデータを使った演習（自分でやってみる）

■データの種類・フォーマットについて確認

http://sugiura-ken.org/wiki/wiki.cgi/exp?page=R.data

■サンプルデータ

1. 学部別クラスデータ
2. 事前・事後テスト

1.4 基礎的なコマンド

■参考

http://oscar.gsid.nagoya-u.ac.jp/~sugiura/ed/resource/introR.html

---

■データを取り出す

```
A <- classes$ClassA
B <- classes$ClassB
```

・変数が、すでに使われている場合、上書きされてしまう。
・使われているかどうかは、変数を入れて「リターン」で確かめればよい。
・データの一覧を見るには

■データを見てみる。

・いくつ要素があるか見る

```
length(x)
```

・平均を出す

```
mean(x)
```

・レンジ（範囲）を出す

```
range(x)
```

・標準偏差を出す

```
sd(x)
```

・データの全体像を見る

```
summary(x)
```

■グラフにしてみる

資料のちらばりを表す

・hist()
・stem()
　・縦棒の左側が１０（以上）のけた、右側が１のけた
・boxplot()
・barplot()
・plot()

図を順にかさねていくわざ

■正規性の検定

■等分散の検定（F test）

1.5 第１回目の課題（宿題）

■山下先生担当の授業で使ったデータを使って R で統計処理をしてみること。

1. 対応のあるサンプルの場合：「」で、事前テストと事後テストで平均点に差があるか？」
2. 独立したサンプルの場合　：「2013method1.xlsx」で、「二つのクラスの学力に差があるか？」

1. 対応のあるサンプル

```
> t.test(yamashitaPrePost$Pre, yamashitaPrePost$Post, paired=T)

        Paired t-test

data:  yamashitaPrePost$Pre and yamashitaPrePost$Post
t = -61.238, df = 290, p-value < 2.2e-16
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -14.57410 -13.66645
sample estimates:
mean of the differences
              -14.12027
```

2. 独立したサンプル

```
> t.test(dokuritu$one, dokuritu$two)

        Welch Two Sample t-test

data:  dokuritu$one and dokuritu$two
t = 2.7704, df = 37.238, p-value = 0.008683
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  2.177249 14.022751
sample estimates:
mean of x mean of y
    52.75     44.65
```

2.0 等分散を仮定する t 検定をしたい場合は、オプション をつける

```
> t.test(dokuritu$one, dokuritu$two, var.equal=T)

        Two Sample t-test

data:  dokuritu$one and dokuritu$two
t = 2.7704, df = 38, p-value = 0.008615
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
  2.181232 14.018768
sample estimates:
mean of x mean of y
    52.75     44.65
```

1.6 分散分析

■分散分析とは

- 「分散」（バラツキ）を分析する
- データの「バラツキ」が、何らかの理由（要因）によるものなのか、偶然の誤差なのかを判定する。
- 「要因」の数により、「一要因」から「三要因」までである。
- 集めたデータが、一人の被験者からとった複数のデータ（「被験者内」） なのか、複数の被験者からとったデータ（「被験者間」）なのかを区別する。

・要因が複数ある場合は、「混合」（被験者内と被験者間）の場合もある。

■分散分析の流れ

1. 分散分析をする
2. 交互作用があるか
    1. ない場合＝＞主効果が有意か
    2. ある場合＝＞交互作用の分析
        1. 主効果は意味がなくなる
        2. 単純主効果が有意か
3. ３水準（条件）以上の有意な主効果もしくは交互作用があるか
    1. ない場合＝＞ End
        1.2 水準で有意な場合、差があるのはどこなのかは明白なので、多重比較する必要はない。
    2. ある場合＝＞多重比較（二つずつ条件を組み合わせて、実際に有意な 差があるのはどこかを見つける）
        1. 主効果が有意であった場合＝＞主効果の多重比較
        2. 交互作用が有意であった場合＝＞単純主効果の多重比較

■ R で分散分析は ANOVA 君できまり

・js-STAR と同じ使い勝手・説明
    ・田中・山際 (2008)『ユーザーのための教育・心理統計と実験計画法―方法の理解から論文の書き方まで』で勉強できる。

・参考：JavaScript-STAR を 使った分散分析

■ anova 君のインストール（R で書かれたプログラムを読み込んで使う）

1.anovakun というソースコードをダウンロードする
2.R のメニューからソースコードを読み込む

```
source("http://www11    .atpages.jp/~riseki/pukiwikiplus/index.php?plugin=attach&refer=ANOVA%B7
%AF&openfile=anovakun_433.txt")
```

■分散分析のタイプと分析の手順

分散分析のタイプ一覧

ＡｓＢ（２要因参加者間）
ｓＡＢ（２要因参加者内）
ＡＢＣ（３要因参加者間）
ＡＢｓＣ（３要因混合）
ＡｓＢＣ（３要因混合）
ｓＡＢＣ（３要因参加者内）

1. 分析のタイプを決める。
    1. 要因はいくつあるか
    2. それぞれの要因は「被験者内」か「被験者間」か（「対応のある」データか「対応のない」データか）
    3. それぞれの要因に「水準」はいくつあるか
    4. 例
        1. トレーニングをする前と後で成績が上がるか（一要因・被験者内・二水準）
        2. トレーニングをする前と後、そして、しばらくしてからも効果が残るか（一要因・被験者内・三水準）

3. 文学部、工学部、理学部で、英語の成績に差があるか（一要因・被験者間・三水準）
4. 漢字の画数が多い熟語と少ない熟語で読む時間に差があるか（一要因・被験者内・二水準）
5. 漢字の画数が多い熟語と少ない熟語で、実在する語と実在しない語とで、読む時間に差があるか（二要因・被験者内・２×２）
6. 漢字の画数が多い熟語と少ない熟語を読むときに、日本語母語話者と学習者とで差があるか（二要因・混合・２×２）
7. 漢字の画数が多い熟語と少ない熟語で、実在する語と実在しない語とで、日本語母語話者と学習者とで読む時間に差があるか（三要因・混合・２×２×２）

2.。
3. データを読み込む。
4. 分析のタイプに合わせて、コマンドを実行する。

■ ANOVA 君の使い方

３．データファイルの作成

・ポイント
  ・被験者内は、横に並べる
  ・被験者間は、縦に並べる

■ anova 君を使った分析の実例「事前・事後・遅延テストで、成績に差があるか？」

・山下先生担当の授業で使ったデータ（対応のあるサンプルの場合：「」）を例に
・ポイントは、データをどういうフォーマットで並べた表を用意するか。

1. 同じ人から３種類のデータを取っているので「対応のあるデータ」（被験者内）（参加者内とも呼ぶ）
2. 要因は、タイミングだけなので、１要因（水準は三つ）
3. タイプとしては　(１要因参加者内)　タイプ。
4. 被験者内なので、平均の差の標準誤差が等しいことを確認するために球面性検定を行う。
    1. その際に、もし有意だった場合、自動で自由度の補正を行うオプションをつける
5. コマンド

```
anovakun( データ , " タイプ ", 水準数 , オプション )
```

・分析結果

■分析の実例２

・杉浦・岩崎（2003）日本語学習者のための擬音語・擬態語学習用マルチメディア CALL 教材の改善に向けて
  ・マルチメディア CALL 教材の学習効果を調べた。
    ・教材に含まれていて学習した項目（実験項目群）と教材に含まれていなかった項目（統制項目群）とで、学習の事前・事後・遅延で、成績に差があるか。
    ・結果
・サンプルデータ

・分析
  1. 実験項目群について、事前・事後・遅延の成績に差があるか？
  2. 統制項目群について、事前・事後・遅延の成績に差があるか？

3. 実験項目群と統制項目群とで、事前・事後・遅延の成績に差があるか？

■杉浦担当の授業に関しての課題

上の「分析」の１と２を R を使ってやってみること。
その手順と結果と考察をレポートにまとめること。
評価のポイントは、
１）分析手順がわかっているか、
２）結果のどこを見てどう判断すればよいかわかっているか、
３）結果について、自分で考察することができるか、
という点を評価しますので、「わかっている」ということが、わかるように書いてください。
枚数の制限は特に設けませんが、出てきた結果を、どこが必要か、どこは不要か、ということを
考えずにただコピペして枚数を増やすというのはよくありません。
必要なことを十分書いてください。

「分析」の３まで、やれた人はボーナスポイントを付けます。
授業の感想も、書いてくださいね。よろしく。

2. R によるテキスト処理入門

2.0 作業ディレクトリー、作業スペース（）、そして、履歴（）

■作業ディレクトリー

・パソコンの「ファイルシステム」の中のどこで作業をするか
・どのフォルダー内にあるデータを処理するか
・処理するデータのあるフォルダーへ移動しておく
・メニュー「ファイル」＝＞「」
　・USB メモリー内に授業用のフォルダーを作っておくとよい
・移動して、今どこにいるか、確認するには　と打ってみる。

```
> getwd()
[1] "C:/Users/sugiura/Documents"
```

■作業スペース（）（「ワークスペース」と呼ばれることもある）

・R で作業した「内容」（読み込んだり処理して「変数」に入れたデータ等）のすべて
・作業ディレクトリー内に自動的に保存される。
・終了するときに「作業スペースを保存しますか？」と聞かれる。「はい」と答えると、今いる作業ディレクトリー内に保存される。
・ファイル名は「」（拡張子を表示しないようにしていると、表面上は「見えない」）
・作業履歴も自動的に同じ場所にというファイル名で保存される。

■作業を再開するとき

・作業ディレクトリーへ移動する。
　・メニューの「ファイル」＝＞「ディレクトリーの変更」
・してみて、以前の作業スペースが復元されていることを確認する。
・自動的に復元されていない場合は、メニューの「ファイル」から明示的に「作業スペースの読み込み」をする。
・「履歴」も同様。矢印キー「↑」を押してみて、以前打ち込んだコマンドが出てくるか確認する。
　・出てこない場合は、メニューから読み込む。

2.1 学習者コーパス NICE (Nagoya Interlanguage Corpus of English)

■日本語母語英語学習者による英文エッセイを集めたコーパス（母語話者コーパスも含まれる）

http://sgr.gsid.nagoya-u.ac.jp/wordpress/?page_id=168

1.「Windows 版」をダウンロード
2.USB メモリーに入れる
3. マウスの右ボタンで「すべて展開」
4. 以下のフォルダー内のファイルを使います：

NICE-NNS（CHAT フォーマットで整理された学習者コーパス）
NICE-NS（比較のための英語母語話者コーパス）

・CHAT フォーマットとは、、、

2.2 R でのデータの取り扱い

■変数と代入

・変数という「入れ物」に「値」を代入する
・変数は、半角英数文字でつける（数字で始めてはいけない。大文字小文字は区別する。）
・変数への「代入」は、 記号であらわす。

```
seiseki <- 50
```

・「seiseki」という変数に数字「50」が入った。

```
namae <- "Hanako"
```

・「namae」という変数に文字「Hanako」が入った。（文字はダブルクオートの記号「」で囲む）

> abc <- 3　# abc という変数を作り、3 を代入。

> abc # abc の中身を表示させる。
[1] 3 # 3 が入っている。

> abc <- 300 # abc の値に 300 が上書きされる。

> abc # abc の中身を表示。
[1] 300 # 中身が 300 に変わった。

> efj <- 6

> jke <- 2987

> ls() # これまでに作った変数一覧を表示。

[1] "abc" "efj" "jke"

> rm(jke) # 変数 jke を削除。

> ls() [1] "abc" "efj"

文字と数字を区別する

> namae <- "sugiura" # namae には sugiura という文字列が入る。

> class(abc) # class(変数名) で、変数に入っているものが数字なら numeric を、

文字なら character を返す。

[1] "numeric"

> class(namae)

[1] "character"

■配列（R では「ベクトル」と呼ぶ）

配列＝入れ物が複数ならんでいる変数

<- c(, , ,) で入れる

> kazu <- c(2,4,6,8)

> kazu # kazu の中身を表示

[1] 2 4 6 8

> names <- c("I","you","he") # 配列の中身は文字列でもいい。

> names [1] "I" "you" "he"

> length(names) #length(配列名) で、該当配列の要素数を返す。

[1] 3 # 配列 names の要素は、 "I" "you" "he" の 3 つ。

変数や配列に何があるか調べる

ls()

要素がいくつ入っているか調べる

length(配列名）

2.3 ファイルとして保存されているデータの読み込み

scan(file=" ファイルの場所と名前 ", what="char")

文字データなので、what="char" を指定

ウインドを開いて、GUI で選ぶ

■セパレーターの話

データの単位を何で区切るか

デフォルトは　スペース

行単位で読み込むには、セパレーターを指定する

sep="\n"

```
> scan(choose.files(), what="char", sep="\n")   #セパレータを改行マーク (\n) に変更。 #デフォルトではスペース。
```

(出力例)

[1] "@Begin"

[2] "@Participants:\tNS001"

[3] "@Age:\t46"

[4] "@Sex:\tM"

[5] "@L1:\tAmE"

[6] "@FatherL1:\tAmE"

[7] "@MotherL1:\tAmE

（以下省略）

■読み込んだファイルを R の中にデータとして保存する

配列名「jp001」という名前にすることにする

```
jp001 <- scan(choose.files(), what="char", sep ="\n")
```

★ NICE のデータの中で、学習者３名分を、行単位で読み込んで、変数に入れてください。

変数	ファイル名
jp001	JPN001.txt
jp002	JPN002.txt
jp003	JPN003.txt

2.4 データの閲覧と検索

■読み込んだデータを見てみる

・全部見てみる

```
jp001
```

・最初の方を見てみる：

・最後の方を見てみる：

・特定の行だけ見てみる。

```
jp001[30]
```

```
jp001[30:40]
```

```
jp001[c(30,33,36,39)]
```

■文字列の検索

・grep(" 検索文字列 ", 変数名)

```
grep("JPN001", jp001)
```

要素番号（＝行番号）が表示される。

・中身そのものを出力するオプション

```
grep("JPN001", jp001, value=T)
```

・正規表現を使う

```
grep("\\*JPN001", jp001, value=T)
```

・正規表現「*」の「エスケープ」に「\」を二重に使う点に注意

・

```
jp001.body <- grep("\\*JPN001", jp001, value=T)
```

■不要文字列の削除：「削除」＝「何もなし」で置き換える：置き換える命令：

・不要なのは行頭の話者記号「*JPN001:\t」
・行頭を示す「*」が正規表現なので注意：を前につけて「エスケープ」する

```
gsub("\\*JPN001:\t", "", jp001.body)
```

```
jp001.data <- gsub("\\*JPN001:\t", "", jp001.body)
```

2.5 単語リストの作成

・「行単位」で入っている。
・これを「単語単位」にバラバラにする。
・文字列を切り離す命令：

```
strsplit(jp001.data, " ")
```

・バラバラにした結果は「リスト」に入る（２次元データ）

```
[[1]]
[1] [2] ...
[[2]]
[1] [2] ...
[[3]]
[1] [2] ...
```

・バラバラにしたデータを、新しい変数に入れる。

```
jp001.data.list <- strsplit(jp001.data, " ")
```

・「リスト」形式のデータを、リストの要素ごとにバラバラにする命令：
　・二次元でなく、一次元の一列に並んだデータとなる

```
unlist(jp001.data.list)
```

```
jp001.word <- unlist(jp001.data.list)
```

・要素を並べ替える命令：

```
sort(jp001.word)
```

```
jp001.sorted <- sort(jp001.word)
```

・重複する要素の重複をなくす命令：
　・★ UNIX のコマンド uniq とつづりが違うので注意

```
unique(jp001.sorted)
```

```
jp001.uni <- unique(jp001.sorted)
```

■ TTR (Type/Token Ratio)

・異なり語数÷述べ語数の比率：使われている単語のうちどのくらいが異なっているか：「多様性の指標」
　・異なり語数はどうやって出すか？
　・述べ語数はどうやって出すか？
　・計算はどうやってするか？

・ヒント：要素の数を出す命令は？

今日の課題：学習者３人分と母語話者３人分の TTR を出して比較してみる。

■便宜上、考えてない問題

1. 大文字・小文字
　・
2. 句読点などの記号
　・
3.TTR の指標としての注意すべき点

■一覧表を作る命令：

```
> table(jp001.word)
```