

Rで始める心理統計

徳永智子・国里愛彦・蔵永 瞳・高村真広・深瀬裕子・宮谷真人

Introduction of statistical methods in psychological research with R

Satoko Tokunaga, Yoshihiko Kunisato, Hitomi Kuranaga, Masahiro Takamura, Yuko Fukase,
and Makoto Miyatani

Rは高度で多様な統計解析処理が可能なフリーソフトである。このような利点から、近年心理統計を行うソフトとしての使用も増えている。特に、教育現場において、Rを導入する試みもなされ始めている。しかしながら、RはCUIを用いており、プログラミングの基礎知識を必要とするため、初学者には敷居が高い印象がある。また、実際にRを扱ってみても、思いがけない点でつまずきやすい。本論文は、心理学を学ぶ大学生がRを用いて心理統計の基礎を学び、実際の解析を行えるよう、Rの基礎的な操作について具体的に紹介することを目的とした。Rの基礎知識、操作方法および簡単な記述統計やデータの図示方法について紹介する。

キーワード： R, 心理統計, 記述統計

1. はじめに

心理学研究や心理学における統計教育においては、SPSSやSASが用いられることが多い。しかし、SPSSやSASは学生が個人で購入しないし契約するには高価なソフトウェアであり、学んだ内容を卒業後の生活で活用していくことは困難である。一方で、Rはオープンソースなプロジェクトで開発されているフリー統計ソフトであり、無料で利用することができる。そのため、SPSSやSASとは違い、学んだ学習内容を卒業後の生活の中でも活用することができる。また、Rは多くの研究者や開発者によって提供されるパッケージによって、多彩で最新の分析方法も取り入れることができる。そのため、最新の解析方法についていち早く学ぶという点に関して、Rは優れているといえる。

Rを用いて心理統計を行うことには、フリーソフトであるという点以外にも以下のようなメリットが挙げられる。まず、Rでは他のソフトではできない多様な分析が可能である。具体的には、SPSSには用意されていないポリシリアル相関、ポリコリック相関、項目反応理論といった分析も行うことができる。また、Rでは実際にキーボードから関数を入力し分析を行っていくので、分析法の理解を深める効果も期待できる。

このような利点から、心理学の授業でもRが導入され始めている。最近ではRに関する書籍も増

えてきており、心理統計についても山田・杉澤・村井（2008）といった R 初学者向けの本が出版され、以前より非常に学習しやすくなった。しかし、心理学を学ぶ者が実際に R を用いて解析を行う際、これらの書籍での学習だけでは理解しにくい点が存在する。例えば、SPSS と異なり、R では CUI（Character User Interface）を用いる。CUI とは、画面上のアイコンなどをマウスでクリックして操作するのではなく、行いたい処理の命令をキーボードから入力することでコンピュータを操作するインターフェースのことである。心理学を専攻する学生の多くは文系出身者であり、このようなコマンド入力が必要なパソコン操作は馴染みが薄い。しかし、こういった R の使用以前の基本的な事柄について、多くの書籍では詳しく説明されておらず、実際に R を扱おうとする際のつまづきやすさの一因となっている。そこで本論文は、心理学を学ぶ大学生が R を用いた心理統計を行えるよう、R の初学者がつまづきやすい点を挙げてその対処法を紹介しながら、R の基本的な操作方法を紹介することを目的とする。特に、R のインストールと簡単な操作方法、外部データの読み込み、基本統計量の算出、およびデータの図示方法について説明する。なお、実際に解析を行いながら R の操作を学べるよう、本論文で扱う仮想データのファイルを、<http://sites.google.com/site/yoshihikokunisato/rde-hajimeru-shinri-toukei> からダウンロードできるようにした。

2. R のインストール方法

最初に、R をダウンロードして、パソコンにインストールする必要がある。R は CRAN（The Comprehensive R Archive Network）のサイト（<http://cran.r-project.org/index.html>）からダウンロードできる。まず OS（Windows, Mac, Linux）を選択し、“base” をクリックしてダウンロードする。続いて、ダウンロードしたファイルをダブルクリックして、インストールする。インストールに関しては、最初は指示どおりにデフォルトでインストールするとよい。Windows 用 R のインストール手順を Table 1 に簡単に示す。Mac OS の場合や、より詳しい手順については、山田他（2008）を参照していただきたい。

Table 1
Windows 用 R のインストール手順

-
- 1 セキュリティの警告が出現するが、“実行” をクリックし、継続する。
 - 2 セットアップの続行、セットアップウィザードの開始では、“次へ” をクリックし、先に進む。
 - 3 インストール先を指定する。
 - 4 コンポーネントの選択は特に行わず、“次へ” をクリックする。
 - 5 セットアップに使用する言語として“Japanese” を選び、“OK” をクリックする。
 - 6 起動時オプションは“いいえ” を選択し、次へ進む。
 - 7 プログラムグループの設定、追加タスクの選択も、特に設定せず次へ進む。
-

以上でインストールが終了する。インストールが終わって、デスクトップに“R”というアイコンができたなら、それをダブルクリックして R Console を起動する。もし、文字化けが生じたら、RGui

ウィンドウの“編集”メニューから“GUI プリファレンス”を選択し，“Rgui 設定エディター”で“Font”から“MS Gothic”を選び、保存する。

3. R の操作方法

3.1 基本的な操作

1 節でも少し述べたが、R を実際に扱おうとする際、最もとっつきにくい印象を与えるのは、R が CUI であるということである。このような敷居の高さを減らすため、まず CUI と GUI (Graphical User Interface) について説明する。

CUI は、キーボードから文字を入力してコンピュータを操作するものである。一方 GUI とは、情報の表示にウィンドウやアイコンなどのグラフィックを使用し、操作の多くをマウスで行うことのできるインターフェースである。Word や Excel といった Office ソフトなど、普段行っているパソコン操作のほとんどが GUI である。

つまり R は、キーボードから命令 (コマンドと呼ぶ) を入力することで解析を行うのである。R は我々が入力したコマンドの内容を実行し、返答する。SPSS などの統計ソフトは GUI であるため、データをシートに入力し、表示されるボタンをマウスでクリックしていくことで処理が行われるのに対し、R の操作画面には表示されるボタンはなく、R Console というウィンドウ (Figure 1) に文字が並んでいるだけである。GUI に慣れているとイメージしにくいのが、実際に R を使用することで、R の操作に徐々に慣れることができると考えられる。まずは加減乗除の方法から紹介する。

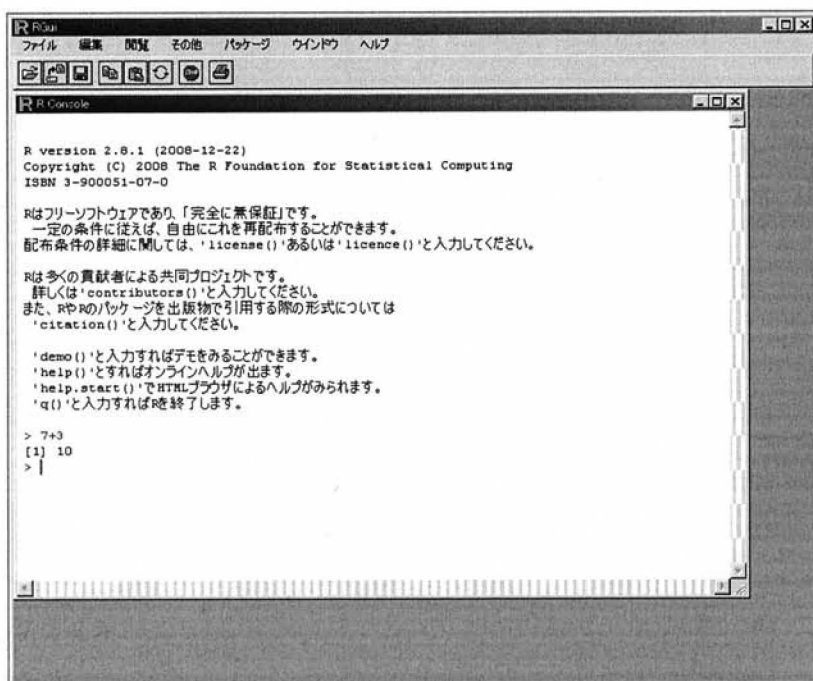


Figure 1. R Console

まず、デスクトップの R アイコンをダブルクリックする。R が立ち上がると、RGui ウィンドウが開き、R Console ウィンドウが表示され、最下段に、自動的に “>” (プロンプト) が表示される。このプロンプトに続けて数字や記号を使ってコマンドを入力し、<Enter>キーを押すことで、コマンドが実行される。また、コマンドは基本的に半角英数字で入力する。例えば、7+3 を計算する場合、

```
> 7+3
```

と入力し、<Enter>キーを押す。すると以下のように表示される。

```
> 7+3
[1] 10
```

“7+3 は?” というコマンドに対し、R が “10 です” と返答したということである。このように、キーボードから入力するコマンドと R の返答の繰り返しで操作する。加減乗除は “+,-,*,/” で可能である。数字や記号を入力することで答えを計算してくれる一種の電卓と考え、使用するとよい。

3.2 関数について

ただし、このような単純なコマンドだけでは、行える処理に限界があるし、行いたい処理によっては非常に手間がかかる。そこで、関数というものが用いられる。関数とは、データに対してあらかじめ定められた処理を実行し、結果を返すコマンドである。Excel でデータ処理をする際、SUM や AVERAGE といった関数をよく用いるのではないだろうか。R でも、例えば数の合計を算出したとき、

```
> sum(数値)
```

と入力する。このような場合の “sum” を関数と呼ぶ。

より複雑な統計解析も、このような関数を用いて行う。当然のことであるが、1 文字でもスペルミスがあると正しく動作しないので気をつける。また、関数の入力は必ず半角で行う。

3.3 R エディタ

R ではこのようにコマンドを入力して作業を進めるのであるが、その際に R Console に打ち込むよりも R エディタを用いるのが便利である。R エディタとは、R Console に読み込んで処理を行う前の準備や、編集作業を行えるメモパッドのようなものである。R エディタを用いて作業を行うことの利点には、次のようなものがある。例えば、複数行に渡るような長いコマンドを入力し実行する際、直接 R Console 上に入力して作業を行うと、途中でコマンドを間違った場合、再度初めからやり直さなければならない。しかし R エディタで編集したものを読み込めば、そのような手間はかからない。また、コマンドしか入力できない R Console と比べ、R エディタはメモ帳のように自由に記述でき、一度行った処理を再度部分的に行いたい場合には、R エディタの内容をさかのぼって選択して簡単に実行することができるなど、非常に扱いやすい。

R エディタは、RGui ウィンドウの “ファイル” メニューの中から “新しいスクリプト” を選択すると出現する。この R エディタに直接コマンドを入力する (Figure 2)。そして、R エディタに入力

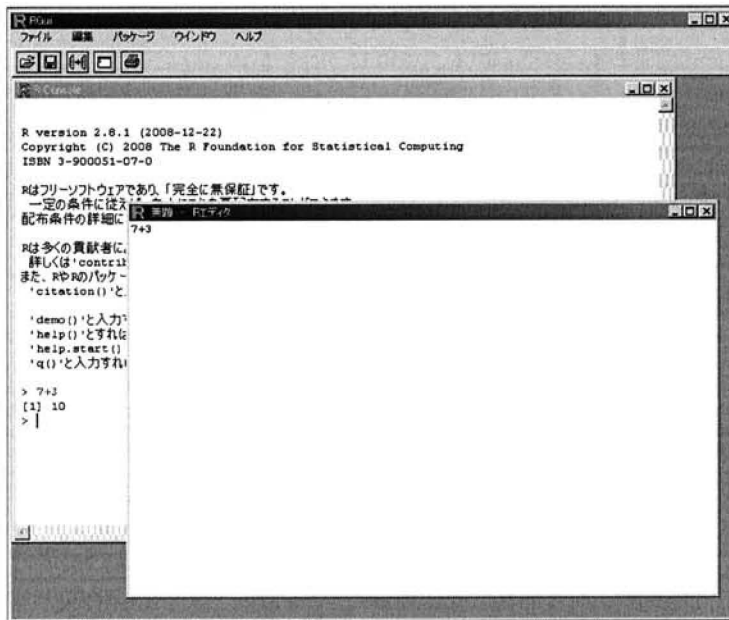


Figure 2. R エディタ

した内容を R Console 上に書き込むには、書き込みたい部分を選択し、<Ctrl>キー+<R>キーまたは <F5>キーを押す。すると、選択した内容が R Console 上に表示される。なお、R エディタの内容を保存しておき、後日 R を実行するときに RGui ファイルメニューの“スクリプトを開く”から呼び出すことも可能である。

4. データの読み込み

実際に関数を用いて様々な統計処理を行う前に、R でのデータの読み込み方について紹介する。以下に示すように、R ではデータを読み込む方法が複数あるので、場合によって使い分けたり、最も扱いやすい方法を用いたりするとよい。

4.1 ローデータをそのまま入力する

例えば、10 人分のテスト得点のデータを扱う場合のことを考える。その合計や平均など、複数の処理を行いたい場合に、処理を行うたびに 10 人分の得点を入力してもよいが、手間もかかるミスも生じやすくなる。よって、ローデータを“テスト得点”という名前のひとまとまりのデータセットとしてまとめる。この作業を、データを格納すると呼ぶ。データを格納する関数は“任意のデータ名<-c(各データ)”である。このときの任意のデータ名には全角文字も用いてよい。例えば、10 人分のテスト得点を“テスト得点”という名前のデータセットにまとめるには、以下のようになる。

```
> テスト得点<-c(38,40,43,35,29,35,37,38,46,32)
```

これで、このローデータに対して何らかの解析を行う際には、“テスト得点”と入力するだけで、

ひとまとまりの 10 人分のデータを扱うことが可能となる。なお、ここでの“テスト得点”といったデータセットの名称を“変数名”と呼ぶ。R では、このように変数名を指定する必要がある。変数名を指定する作業は、特にその後の統計解析において重要になるので覚えておきたい。

4.2 Excel のデータをコピーペーストする

実際に実験や調査で得られたデータは、Excel ファイルとして保存していることが多いだろう。先ほどのようにローデータを 1 つずつ入力せずとも、Excel ファイルのデータをコピーペーストすることでもデータの読み込みが可能である。そのためには、まず Excel のデータを開く。そして解析に使いたいデータの部分を選択し、コピーする。この際、使いたいデータの最上段には変数名を入れておき、変数名も一緒に選択する。この作業によって、Windows のクリップボードにデータが一時的に格納される。そして、以下のように入力する。

```
> 変数名<-read.delim("clipboard")
```

クリップボードに格納されたデータを読み込むことにより、選択した部分のデータが読み込める。このコマンドは R エディタに入力してから読み込むと不具合が生じるため、直接 R Console に入力する。このようにクリップボードへコピーすることでデータを読み込む方法は、あるデータファイルの一部のデータについて解析を行いたいときに便利である。ただし、クリップボードへのコピーを複数回行う場合には、クリップボード内のデータが実際に読み込みたい最新のものであるか、履歴に気をつけながら用いる必要がある。

4.3 Excel のデータを csv 形式で保存して読み込む

R で Excel ファイルをそのまま読み込むことはできないが、csv 形式に変更すると、簡単に読み込むことができる。そのためには、Excel ファイルの保存時に“名前をつけて保存”を選び、csv 形式で保存すればよい。この際“データ名.csv.xls”と xls の拡張子が付かないよう注意する。また、Excel ファイルから csv 形式に変更する際には、保存時に警告が出るが、無視して保存してよい。読み込む際には以下のように入力する。

```
> 任意のデータ名<-read.csv("読み込みたいファイル名.csv")
```

これで<Enter>キーを押し、エラーが出なければファイルが読み込まれたことになる。よくあるエラーとして、“ファイル 'データ.csv' を開くことができません、理由は 'No such file or directory' です”と表示されるものがある。ファイル名が正しいにも関わらず、このような表示が出る場合、作業ディレクトリとファイルが置かれている場所が違うと考えられる。作業ディレクトリとは、作業をしている場所であると考えよう。作業している場所と同じ所にファイルがなければ読み込めないためである。作業ディレクトリは、カーソルが R Console にある状態にしてファイルメニューを選択し、“ディレクトリの変更”で変更することができるので、csv ファイルのある場所とディレクトリとを一致させる。

このように、csv 形式のファイルを読み込む方法は、1 人に関して、質的変数と量的変数が混在した複数のデータがある場合に便利である。例えば、Table 2 のようなデータがあり、これがアルパイ

Table 2

アルバイト調査のデータ (仮想データ)

アルバイトの種類	満足度	時給	アルバイトの種類	満足度	時給
家庭教師	4	1300	スーパー	4	780
飲食店	3	750	飲食店	5	900
飲食店	2	780	家庭教師	5	1360
スーパー	2	690	スーパー	3	690
家庭教師	3	1100	家庭教師	4	1100
塾講師	3	1250	塾講師	4	1250
塾講師	5	1500	スーパー	2	750
スーパー	3	720	飲食店	3	790
飲食店	4	850	家庭教師	3	1150
スーパー	2	750	飲食店	4	810
スーパー	4	780	スーパー	2	790
飲食店	5	900	塾講師	4	1300
家庭教師	5	1360	スーパー	3	690
塾講師	3	1450	飲食店	3	1450
スーパー	2	790	スーパー	2	790
スーパー	1	760	家庭教師	3	760
飲食店	1	720	飲食店	3	720
スーパー	4	790	塾講師	5	1500
飲食店	3	820	スーパー	2	720
塾講師	3	1250	飲食店	4	850
家庭教師	4	1300	スーパー	2	750
家庭教師	3	1350	塾講師	3	1200

ト調査.csv という csv ファイルで保存されているとする。このデータを読み込む際、各変数（アルバイトの種類、満足度、時給）について毎回読み込まなくても、ファイルの全てのデータを一度に読み込み、その後変数ごとに処理を行うことが可能である。まず、次のように入力する。

```
> アルバイトデータ<-read.csv("アルバイト調査.csv")
```

読み込まれたかどうかを確認するため、

```
> アルバイトデータ
```

と入力し、<Enter>キーを押すと、R Console にファイルの内容が表示される。

4.4 Excel データをそのまま使用する

Excel データを csv 形式に変換せず、そのまま使用することもできる。ただし、そのためには xlsReadWrite という、Excel ファイルを読み込むことを可能にするパッケージをインストールする必要がある。

パッケージとは、R のプログラムをまとめ、配布できる形式にして保存されたものである。ある複雑な処理についてコマンドを入力して実行しようとする大変な手間がかかるところを、複雑なコマンドをひとまとめにし、簡単に実行できるようにされたものであると考えるとよい。標準インストールでもいくつかのパッケージを使用できるが、インターネット上から新たにインストールすることも可能である。xlsReadWrite パッケージは、インターネット上からインストールする必要がある。

ある。インターネットに接続した状態で R を開き、RGui ウィンドウの“パッケージ”メニューの中から“パッケージのインストール”を選択する。まずどのサイトからインストールするかを選択する画面が出現するので、“Japan(Tsukuba)”を選択する。次に、どのパッケージをインストールするか選択すると、インストールが実行される。

インストールが完了したら、以下のようにコマンドを入力する。まず `library()` という、パッケージを指定するための関数によってパッケージを読み込み、その後ファイルを読み込む。

```
> library(xlsReadWrite)
> 任意のデータ名<-read.xls("読み込みたいファイル名.xls")
```

この方法は、パッケージのインストールなど少し複雑であるが、Excel ファイルを csv ファイルに変換する必要がないので、比較的扱いやすい。

5. 基本統計量と図の作成

データの解析には、まず基本統計量を算出することが必要である。また、データを分析する際には、基本統計量を見るだけでなく、データの分布を図示するなどしてその全体像を把握することが重要である。R では、関数を用いて簡単に基本統計量を算出し、さらに美しい図を出力することができる。本節では、基本統計量の算出とデータの図示の方法を紹介する。なお、図の作成については中澤 (2007) に詳しく紹介されているので、参照して欲しい。

5.1 代表値 (平均値, 中央値, 最頻値)

データを代表する値を代表値といい、平均値, 中央値, 最頻値の3つがある。その中でもよく用いられるのは平均値であり、データの総和をデータの個数で割った値である。平均値は“`mean(変数名)`”という関数で求められる。例えば、43, 21, 18 の平均値を求める場合は以下ようになる。1行目で3つの数値を“データ”という変数に格納し、2行目でその変数の平均値を算出する。

```
> データ<-c(43,21,18)
> mean(データ)
[1] 27.33333
```

実際に統計解析に使用する際は、各データの数値を直接入力するよりも、前節で紹介したように、すでに作成されたデータファイルを読み込む場合が多い。例えば、前節の csv ファイルのデータの読み込みの方法を用い、アルバイト調査のデータファイルの中の“時給”の平均値を求める場合には、以下ようになる。

```
> アルバイトデータ<-read.csv("アルバイト調査.csv")
> mean(アルバイトデータ$時給)
[1] 978.6364
```

“\$”は、“アルバイトデータ”というデータセットの中の“時給”という変数を指定するという

意味である。このように毎回データセット内の変数を入力するのが面倒な場合は、“attach(データ名)”によって変数を登録しておくとし、以下“\$”を用いなくても参照できるようになる。一度登録すると、“detach(データ名)”で登録を消去するまで、変数の参照が有効である。例えば attach 関数を用いた場合、先ほどの平均値の算出は以下ようになる。

```
> attach(アルバイトデータ)
> mean(時給)
[1] 978.6364
```

平均値は外れ値の影響を受けやすいため、外れ値がある場合には中央値が代表値として用いられる。中央値は、データを小さいものから順に並べたときに、ちょうど中央に位置する値である。データ数が偶数の場合には、中央に位置する2つのデータの平均値となる。中央値は“median(変数名)”で求められる。今度は、アルバイトデータの中の“時給”の中央値を算出する。すでに“attach(アルバイトデータ)”で変数が参照可能になっているので、以下のように入力すればよい。

```
> median(時給)
[1] 815
```

最頻値は最も度数の多いデータの値である。度数分布表を作成することで求めることができる。度数分布表を作成する関数は table(変数名)である。例えばアルバイトデータの“満足度”について最頻値を調べる場合、

```
> table(満足度)
```

と入力すると、R Console 上に以下の度数分布表が出力され、最頻値は3であることがわかる。

```
満足度
 1  2  3  4  5
 2  9 16 11  6
```

5.2 データの散らばり（分散、標準偏差、四分位偏差）とその図示

データの散らばり具合を知るために、散布度を算出する。平均値を代表値としたときには、平均値の辺りにどれくらいデータが集まっているかを示す分散を算出する。分散を算出する関数は“var(変数名)”である。平均値を算出したものと同じデータについて分散を求めるには、

```
> var(データ)
```

と入力する。すると、

```
[1] 186.3333
```

と返答がある。

また、アルバイトデータにおける“時給”について分散を求める場合は、以下のように変数名を指定する。

```
> var(時給)
[1] 76114.38
```

なお、var 関数で算出できる分散は不偏分散である。不偏分散とは、偏差の 2 乗和をデータの個数 n ではなく $n-1$ で割ったものである。一方で、偏差の 2 乗和を n で割ったものを標本分散と呼ぶ。なお、不偏分散と標本分散の違いについては、山田・村井 (2004) に詳しい。

また、分散の平方根をとったものが標準偏差である。標準偏差は“sd(変数名)”という関数で算出できる。“時給”について標準偏差を求めると、以下のようになる。

```
> sd(時給)
[1] 275.8883
```

このような連続変数の分布状態を見るには、ヒストグラムが用いられる。ヒストグラムを描く関数は“hist(変数名)”である。アルバイト調査について、“時給”のヒストグラムを描く場合、

```
> hist(時給)
```

となる。これを実行すると、Figure 3 のように別ウィンドウが現れ、ヒストグラムが出力される。

このようにして作成したヒストグラムを、Word など他のプログラムに貼り付ける場合には、図が出力されたウィンドウにカーソルを合わせて右クリックし、メタファイルまたはビットマップにコピーする。Word 側で“編集”メニューから“形式を選択して貼り付け”を選び、貼り付ける。プレ

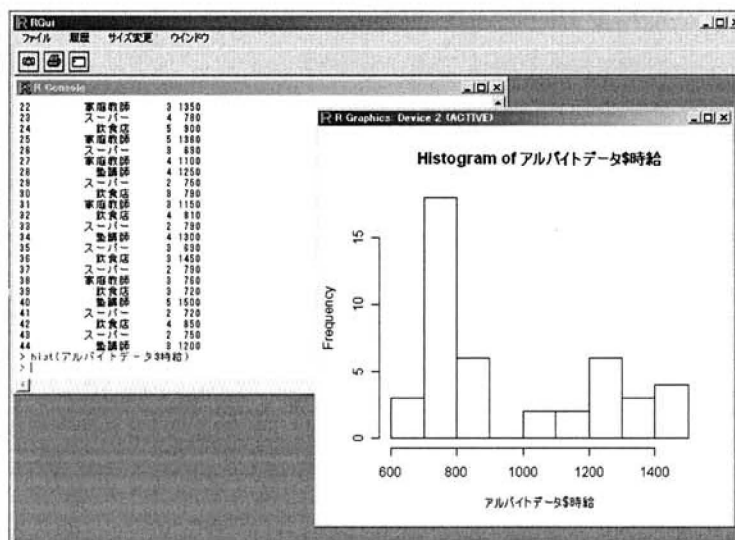


Figure 3. R における図の出力ウィンドウ

ゼンテーションソフトや描画ソフトに貼り付けて仕上げるとよい。データの図示については中澤(2007)で詳しく説明されている。

中央値を代表値とした際の散布度には、四分位偏差を用いる。Rで四分位偏差を算出するには、まず“fivenum(変数名)”という関数を用いて最小値、第1四分位、第2四分位、第3四分位と最大値という5つの要約値を算出する。それを利用し、四分位範囲を算出し、それを2で割って求める。例えば、“満足度”のデータであれば、以下のように入力する。

```
> 5数要約値<-fivenum(満足度)
> 四分位範囲<-5数要約値[4]-5数要約値[2]
> 四分位偏差<-四分位範囲/2
> 四分位偏差
```

これで、

```
[1] 0.75
```

という四分位偏差を導くことができる。なお、この手順は山田他(2008)を参考にした。

また、中央値や四分位とともにデータの分布を図示する方法として箱ひげ図がある。分布の中心や対称性といったおおよその様子と外れ値を比較できるので、ばらつきを捉える上で有益である。箱ひげ図の箱内の太線が中央値、箱の下が第1四分位、上が第3四分位となる。ひげは第1四分位と第3四分位の差を1.5倍した線分となる。箱ひげ図を作成する関数は“boxplot(変数名)”である。アルバイトデータの“満足度”の箱ひげ図を作成する場合を例にとってみると、

```
> boxplot(時給,ylab="時給(円)")
```

となる。ylabで縦軸の項目名を指定する。これを実行すると、Figure 4の箱ひげ図が描ける。

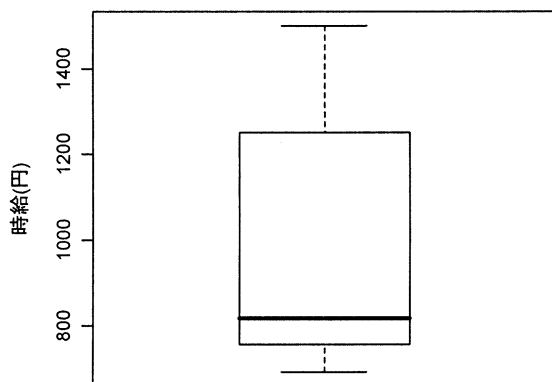


Figure 4. R を用いて作成した箱ひげ図の例

また、R ではストリップチャートを作成することもできる。ストリップチャートは、カテゴリごとに各データが正確にプロットされるので、分布やカテゴリ間の差を把握する上で便利である。ストリップチャートを作成するには、以下のような手順をとる。

まず、アルバイトの種類別に時給の平均、標準偏差を算出する。

```
> 時給平均<-tapply(時給,アルバイトの種類,mean)
> 時給標準偏差<-tapply(時給,アルバイトの種類,sd)
```

次に“stripchart()”関数でストリップチャートを作成する。“vert=T”は縦軸と横軸を入れ替えるオプションである。

```
> stripchart(時給~アルバイトの種類,method="jitter",vert=T,ylab="時給(円)")
```

さらに、エラーバーについて、横軸上の座標を計算する。例の場合、チャートの中心から右に、4つのチャートの間隔の15%ずらした位置を設定している。

```
> エラーバー<-c(1,2,3,4)+0.15
```

“point()”でエラーバーや平均をプロットする。pchはプロットする点の形状を指定する。

```
> points(エラーバー,時給平均,pch=18)
```

最後に、“arrows()”で平均の上下にエラーバーを引く。code, angle, lengthはそれぞれ、エラーバーの上下 (code=3) に、angleの角度でlengthの長さの横線を付与するオプションである。エラーバーの作成手順は、中澤 (2007) を参考にした。

```
> arrows(エラーバー,時給平均-時給標準偏差,エラーバー,時給平均+時給標準偏差,code=3,angle=90,length=.1)
```

作成されたストリップチャートを Figure 5 に示す。

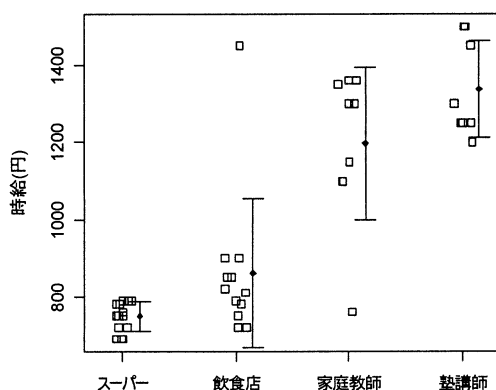


Figure 5. R で作成したストリップチャートの例

5.3 度数分布

名義尺度のデータを扱う際、各カテゴリのデータ数を把握するために、度数分布を示す必要がある。度数分布表は、5.1 で最頻値について説明した際に紹介した“table(変数名)”という関数で作成できる。アルバイトデータの“アルバイトの種類”という変数について度数分布表を作成する場合、

```
> table(アルバイトの種類)
```

と入力する。すると、R Console 上に以下のような度数分布表が作成される。

```
アルバイトの種類
スーパー  飲食店  家庭教師  塾講師
      15      12      9      8
```

さらに、これを棒グラフにする際には、以下のように記述する。

```
> アルバイト<-c(15,12,9,8)
> names(アルバイト)<-c("スーパー","飲食店","家庭教師","塾講師")
> barplot(アルバイト,ylim=c(0,8),main="アルバイトの種類(人数)")
```

まず、度数の変数を作成し、次に各度数のラベルを作成する。その後、棒グラフを作成する関数である“barplot(変数名)”を用い、図を作成する。その際、ylim=c(x,y)で縦軸の目盛 (x が最小値, y が最大値)、main で図のタイトルを指定する。これで Figure 6 の棒グラフを作成できる。

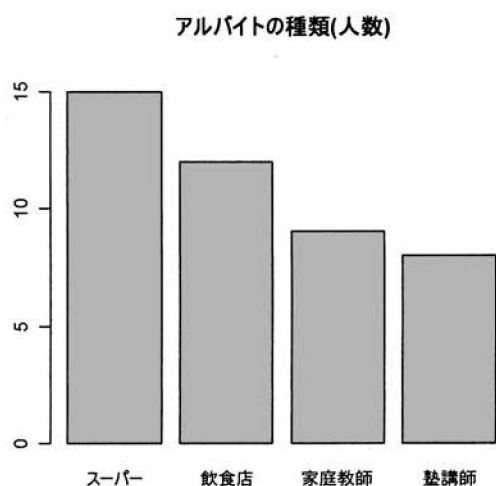


Figure 6. R を用いて作成した棒グラフの例

6. まとめ

以上、Rを用いた心理統計について簡単に紹介してきた。このようにRを心理統計の学習で使い、能動的な作業を行うことで、心理統計に対する理解を促すことができると考えられる。図表も容易に作成できるため、データを視覚化してその全体像を捉えることができ、その後の統計解析について思慮しやすい。また、分析にも用いやすく、他のソフトより融通が利く。例えば、高村・国里・徳永・蔵永・深瀬・宮谷（2008）が紹介するように、一要因分散分析の後に多重比較を行う際、その方法を自由に選択することができる。今後、心理統計教育の現場でもRを用いる機会が増えると思われる。その際、本論文で紹介したような作業を行うことによって、コンピュータの扱いに不慣れた学生でも抵抗なく心理統計に取り組むことができると期待できる。

引用文献

- 中澤 港 (2007). Rによる保健医療データ解析演習 ピアソン・エデュケーション
- 高村真広・国里愛彦・徳永智子・蔵永 瞳・深瀬裕子・宮谷真人 (2008). Rによる一要因分散分析と多重比較 広島大学心理学研究, **8**, 177-190.
- 山田剛史・村井潤一郎 (2004). よくわかる心理統計 ミネルヴァ書房
- 山田剛史・杉澤武俊・村井潤一郎 (2008). Rによるやさしい統計学 オーム社