17 日目：相関係数（2）

昨日まで，因子分析から平均値の比較へと進めました。本日からは方向性を変えて，相関系の分析をやってみたいと思います。

データも本日から変えてみたいと思います。大学生を対象に，読書習慣に関する調査を行ったとします。読書習慣に関連すると考えられる諸変数も測定し，読書習慣形成との関わりから整理することを狙ってみます。このプロセスで，相関係数，偏相関係数，回帰分析をやってみたいと思います。もちろん，でっち上げのダミーデータですが。

データを入手したら最初にやるべきことをやっておきましょう。基礎統計量の算出と度数分布の確認です。ヒストグラムははぶきますが，基礎統計量は以下のようです。

	n	平均値	標準偏差	最小値	最大値	標準誤差
親の様子	162	2.59	1.01	1	4	0.08
家にある本	162	2.50	0.94	1	4	0.07
知識欲	162	14.10	3.37	6	20	0.26
不可欠さ	162	14.13	2.79	5	20	0.22
インターネット	162	4.51	1.59	1	7	0.12
マンガ	162	2.76	1.36	0	7	0.11
本好き	162	14.69	4.73	5	25	0.37
読書習慣	162	2.35	1.77	0	7	0.14

では本日は相関係数をやりたいのですが，相関係数を出す時にはまず散布図を眺めておきます。この段階を端折る人も多いのではないかと思うのですが（あくまでも私の想像），*t*検定の前に平均値や標準偏差，分布の様子を確認しておくように，やはりデータの状況をきちんと眺めておいてから相関係数を算出し，その意味を把握することが大事だと思います。

散布図を作るのは簡単ですから，必ずやっておきましょう。また，いろいろな図がつくれます。

データは x という名前で読み込んだとします。

```
par(family="Osaka")
plot(x$知識欲, x$不可欠)
```

変数名が日本語なので，par でフォントを指定しておきます。簡単な散布図は，plot() だけで作ってくれます。

plot(x$知識欲, x$不可欠)のように２変数を併記すれば OK。３変数以上を一気にやりたいなら，plot(x[2:9])などと複数列を指定するか，変数を別のファイルに入れておくいつものやり方もあります。

```
labels <- c("親の様子", "家にある本", "知識欲", "不可欠さ", "インターネット", "マンガ", "本好き", "読書習慣")
plot(x[labels])
```

複数列を指定すると，興味深い結果を返してきます。散布図マトリックスともいえるような出力が得られます。指定した変数が多すぎると，一つひとつが小さくなりすぎて見にくくなりますが，ある程度までであれば多少小さくても十分ではないでしょうか。なお，右上と左下の部分は，同じ図ではないことがわかると思います。X 軸と Y 軸の変数が入れ替えてあります。

なお，plot のデフォルトでは，データの重なり具合が円の太さ（濃さ）で表現されるのですが，わかりやすいとは言いがたいものです。ネット上には，データの重なり具合，集まり具合を表現するいろいろな方法が紹介されていますので参考にしてみてください。合うものを見つけると，一気に見やすくなると思います。個人的には以下が好みです。col の部分の最後の 2 桁を，20 から 30 くらいの間で動かしてみるとよいでしょう。

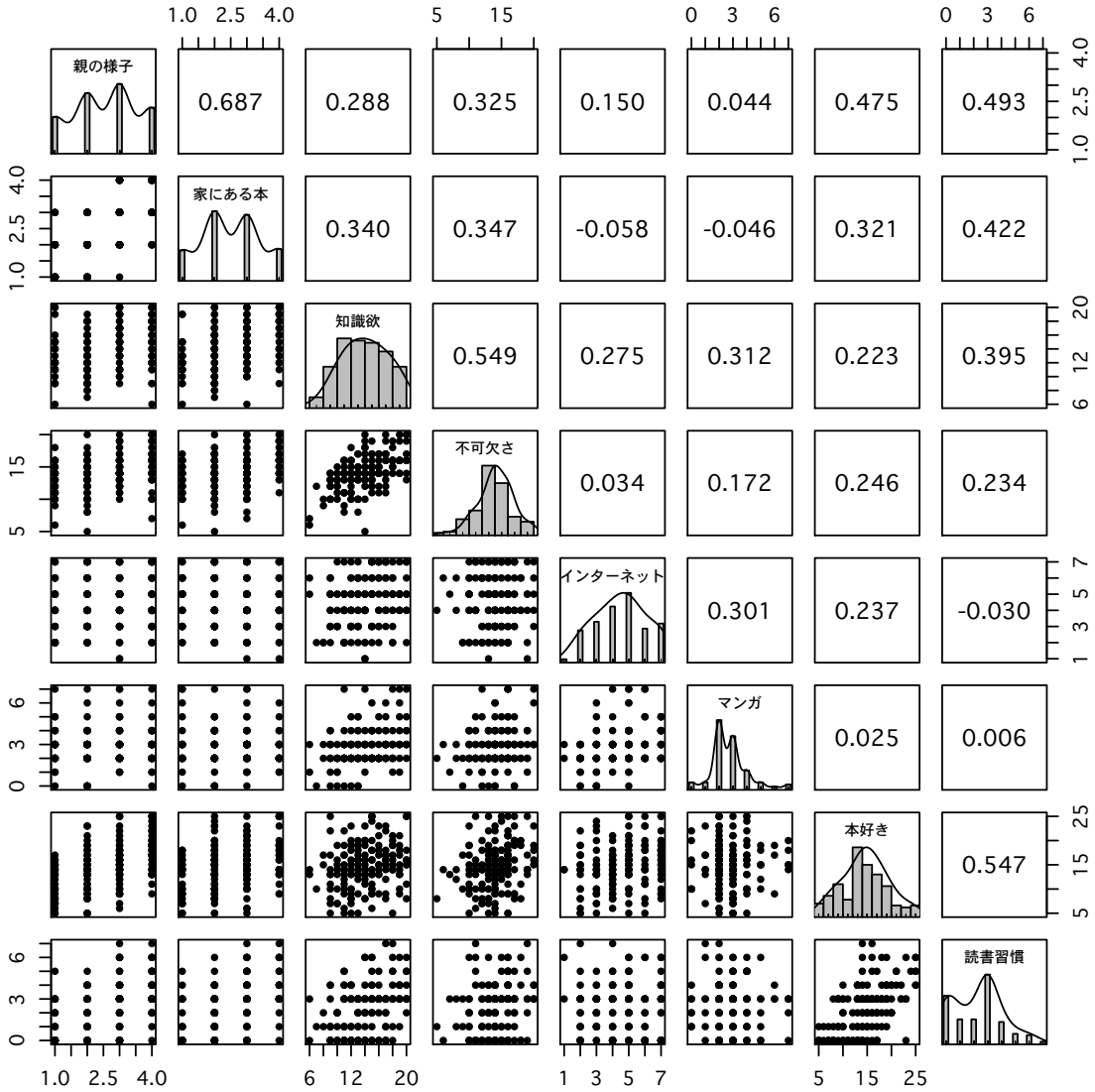
```
plot(x[labels], col="#0000FF25", pch=19)
```

少しフライング気味ですが，他にも状況を概観するために便利な関数があります。psych パッケージの，pairs.panels です。

```
pairs.panels(x[labels])
```

plot の出力と似たような結果が得られます。散布図だけでなく，各変数のヒストグラム，相関係数も同時に表示してくれるところが便利といえるでしょう。ただ，ヒストグラムの色や，不要という感じの情報もあるので，ヘルプを参考に以下のように修正してみました。ここは各自の好みでしょうから，工夫してみてください。その出力を示しておきます。ただ，残念ながら先の散布図のようにコントロールすることはできないようです…。

```
pairs.panels(x[labels], smooth=FALSE, ellipses=FALSE, digits=3, hist.col="gray")
```



さて，これで分布状況を把握したら，相関係数を求めてみます。今回のデータでは，曲線的関係にあると見なせるような分布はないようですから。

10 日目に，cor()という命令を紹介しました。これで作成される相関行列は，いろんな使い道のあるものです。以下のような表（相関マトリクス）が簡単に作れます。

	親の様子	家にある本	知識欲	不可欠さ	インターネット	マンガ	本好き	読書習慣
親の様子	1.000							
家にある本	.687	1.000						
知識欲	.288	.340	1.000					
不可欠さ	.325	.347	.549	1.000				
インターネット	.150	-.058	.275	.034	1.000			
マンガ	.044	-.046	.312	.172	.301	1.000		
本好き	.475	.321	.223	.246	.237	.025	1.000	
読書習慣	.493	.422	.395	.234	-.030	.006	.547	1.000

散布図で分布の様子を確認したら，次は相関マトリクスで変数間の関連を把握するという順になりますが，この行列を眺めているだけではわかりにくいというのも事実でしょう。変数を書き出し，関連の強い変数を近くに置いたり線で結ぶといった図示をすることも，変数同士の関連や構造を把握する方法です。こういう作業をやってくれるような関数 qgraph もあります。これはパッケージなので，qgraph パッケージをダウンロードして，読み込んでください。そして以下を。

```
qgraph(cor(x[labels]),edge.labels=TRUE)
```

日本語を使っているので，フォントを先に指定しておくと文字化けせずに図が作れます。相関係数の正負，強さを，線の色や濃さや，太さで表現してくれます。変数間の関連を把握するたには便利だと思います。

cor()は先の表のような行列を作成するには便利なのですが，論文などに記載する情報が不足します。そこで無相関検定もやってくれる２つのコマンドを取り上げます。

まず一つ目は，

```
cor.test(x$知識欲, x$不可欠)
```

```
> cor.test(x$知識欲, x$不可欠)

    Pearson's product-moment correlation

data:  x$知識欲 and x$不可欠
t = 8.3055, df = 160, p-value = 3.976e-14
alternative hypothesis: true correlation is not equal to 0
95 percent confidence interval:
 0.4311574 0.6482000
sample estimates:
      cor
0.5488631
```

結果は図のように表示されます。なお，cor()と同じで，method=で違う方法を指定できますし，指定しなければピアソンの積率相関係数を求めます。

相関係数とともに，無相関検定の t 値，自由度，p 値を表示しています（3.975e-14 は指数表示です。$3.975 \times 1/10^{14}$ を意味します）。この出力のよいところは，95 percent confidence interval，つまりこの相関係数の 95%信頼区間も出してくれるところです。今回のデータから，母集団における相関係数を推定した場合，95%の確率で.431 から.648 の間にあるという推定です。有意だけれども弱い相関が得られた場合，これをみて，落ち着いて解釈をしてください。

これはかなり必要な値をカバーしてくれる出力をしてくれる（n が出てきませんが…）の

ですが，面倒なのは複数列に渡る変数指定ができないところです。cor.test(x[2:9])とするとエラーになります…。

もうひとつは，psych パッケージにある，corr.test()です（先よりも r がひとつ多い！）。こちらは複数列の指定も受け付けてくれます。なお，method による指定も，cor()や cor.test()と同じようにできます。

```
corr.test (x$知識欲, x$不可欠)
corr.test(x[2:9])
corr.test(x[labels])
```

なお，今回のデータでは次のような警告メッセージが出ると思います。変数名が長すぎることによるようですが，計算結果には影響していないようです。気になるなら，minlength の数値を大きくしてください（デフォルトは 5）。

```
corr.test(x[labels], minlength=10)
```

```
警告メッセージ:
abbreviate(colnames(r), minlength = 5) で:
  ASCII 文字でないものが省略名として使われました
```

この場合の出力は，相関係数，データ数，有意確率の３つの指標になります。なかなかに便利なのですが，相関係数も有意確率も小数点以下２桁で表示されます。ここは好みの分かれるところでしょう。4 桁という論文にはまず出会いませんが，3 桁表示は多くあります。これは，print でコントロールできるようです。

```
print(corr.test(x[labels]), digits=3)
```

さて，相関係数を求める場合には，行と列に同じ変数を入れたい場合もありますが，別の変数を指定したい場合も多いでしょう。そのような場合には，corr.test(x[8:9], x[2:7])というような指定もできます，指定の前半部が行（つまり縦）になり，後半部が列（つまり横）になります。もちろん以下のようなこともできます。なお，cor も同様で，cor(x[8:9], x[2:7])という指定ができます。

```
labels_r <- c("親の様子", "家にある本", "知識欲", "不可欠さ", "インターネット", "マンガ")
labels_l <- c("本好き", "読書習慣")
corr.test(x[labels_l], x[labels_r])
```

```
> labels_r <- c("親の様子", "家にある本", "知識欲", "不可欠さ", "インターネット", "マンガ")
> labels_l <- c("本好き", "読書習慣")
> corr.test(x[labels_l], x[labels_r], minlength=10)
Call:corr.test(x = x[labels_l], y = x[labels_r], minlength = 10)
Correlation matrix
         親の様子 家にある本 知識欲 不可欠さ インターネット マンガ
本好き       0.48       0.32   0.22     0.25           0.24   0.02
読書習慣     0.49       0.42   0.39     0.23          -0.03   0.01
Sample Size
[1] 162
Probability values  adjusted for multiple tests.
         親の様子 家にある本 知識欲 不可欠さ インターネット マンガ
本好き          0          0   0.02     0.01           0.01      1
読書習慣        0          0   0.00     0.01           1.00      1

 To see confidence intervals of the correlations, print with the short=FALSE option
```

最後の部分が気になるならば，print(corr.test(x[labels_l], x[labels_r]), short=FALSE)を実行してみてください，

さて，cor，　cor.test，corr.test の結果を比較してみると，corr.test が一番見やすいのかなと思います。

しかし，論文に記載する表に加工しようとすると，いずれもちょっと手間がかかりそうで，また転記ミスをする可能性もありそうです。

本日はここまでにします。明日は論文に掲載する表の形式に近いものを出力する自作関数を紹介しようと思います。