

19 日目 : 偏相関係数

さて、今日は偏相関係数の算出について紹介します。個人的には、相関と偏相関の差異を眺めてみるのは好きなのですが、論文などではあまり見かけないです…

R では、いくつかのパッケージや自作関数などで偏相関係数を算出させることができます。これまで何度も使ってきた `psych` の中に偏相関係数を算出する関数がありますので、まずはこれを紹介します。

```
partial.r(x, c(2, 3), c(4))
```

このように入力すると、ファイル `x` の中の、2, 3 列目の変数間の偏相関係数を、4 列目の変数をコントロールして計算してくれます。カッコの中は、ファイル名、偏相関を求める変数、コントロールする変数という順で並べます。

結果はシンプルに右のようになります。

```
> library(psych)
> partial.r(x, c(2, 3), c(4))
partial correlations
              親の様子  家にある本
親の様子      1.00      0.65
家にある本    0.65      1.00
```

2, 3 列目の変数間の偏相関係数を、4, 5, 6 列目の変数をコントロールして計算したいならば…

```
partial.r(x, c(2, 3), c(4, 5, 6))
```

4, 5, 6 列目の変数間の偏相関係数を、2, 3 列目の変数をコントロールして計算したいならば…

```
partial.r(x, c(4, 5, 6), c(2, 3))
```

これだけなのでとても簡単に使えますが、表示は小数点以下2桁までのようです。もう1桁欲しい場合は、`print` でコントロールすればできるようです。

```
print(partial.r(x, c(2, 3), c(4)), 3)
```

無相関検定ですが、`partial.r` を `corr.p` に受け渡すことで結果を得ることができます。ただ `n` を入力しなければならないとか、なかなか難しいです。`partial.r` や `corr.p` のヘルプをしっかりと見てやってください。

```
par.r <- partial.r(x, c(2, 3), c(4))
corr.p(par.r, n=161, minlength=10)
```

偏相関に特化したパッケージには `ppcor` があり、これは検定まで一気にやってくれます。これをダウンロードし、`library` で呼び出しておきます。`pcor` 関数はリストに含まれる変数のうち、偏相関係数を算出する 2 変数以外のすべての影響をコントロールしたものを算出します (欠損値が含まれるデータはダメです)。以下の命令であれば、2 行目と 3 行目の偏相関を 4 行目をコントロールすることで算出し、2 行目と 4 行目の偏相関は 3 行目を、3 行目と 4 行目の偏相関は 2 行目をコントロールして算出することになります。

```
pcor(x[2:4])
```

結果の `$estimate` が偏相関係数、`$p.value` が p 値です。そのマトリクスの中から、先の `partial.r` を適用した部分を探し、結果を比べてください。同じになっています。

さて、検定まで一気にやってくれる `ppcor` パッケージですが、以下のようにして偏相関のマトリクスを作りたい場合にはとても便利です。

```
pcor(x[labels])
```

しかし、個別の偏相関を求めようとすると少しやりにくくなるようです。関数としては `pcor.test` を使うことになりますが、これが今一つ柔軟性にかけるようで…。`partial.r` の方が使いやすい場合もありそうです。たとえば、2, 3 列目の変数間の偏相関係数を、4 列目の変数をコントロールして計算したい場合、`partial.r`、`pcor.test` は、それぞれ以下のような命令になります。

```
partial.r(x, c(2, 3), c(4))
```

```
pcor.test(x[,2], x[,3], x[,4])
```

`partial.r` は、続くカッコの中で、データのセット、偏相関を求めたい変数 (3 変数以上可)、コントロールする変数 (複数指定可) となります。他方で `pcor.test` は、続くカッコの中に、偏相関を求めたい変数のペアを 1 番目と 2 番目に置き、最後にコントロールする変数 (複数指定可) を指定します。つまり、「4, 5, 6 列目の変数間の偏相関係数を、2, 3 列目の変数をコントロールして計算したい」といった場合、`partial.r` では一発で可能ですが、`pcor.test` ではいくつもの命令を出さなければならないということになります。うまく使い分けることが必要なのでしょう。

さて、これで相関、偏相関の両者が得られましたので、以下の表のように一覧にしてみました。昨日、一昨日にやった相関係数の値を表の左下側に、偏相関係数を右上側にいれてあります。検定の結果も合わせて示しておきました。論文で目にすることもある形式だと思います。相関係数と偏相関係数で大きく値が異なっているところもあります。

	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	—	.609 ***	-.157	.132	.237 ***	.053	.158 *	.235 ***
家にある本	.687 ***	—	.214 ***	.052	-.229 ***	-.123	.002	.008
知識欲	.288 ***	.340 ***	—	.496 ***	.352 ***	.218 ***	-.169 *	.382 ***
不可欠さ	.325 ***	.347 ***	.549 ***	—	-.215 ***	.058	.173 *	-.178 *
インターネット	.150	-.058	.275 ***	.034	—	.193 *	.307 ***	-.308 ***
マンガ	.044	-.046	.312 ***	.172 *	.301 ***	—	-.051	-.039
本好き	.475 ***	.321 ***	.223 **	.246 **	.237 **	.025	—	.465 ***
読書習慣	.493 ***	.422 ***	.395 ***	.234 **	-.030	.006	.547 ***	—

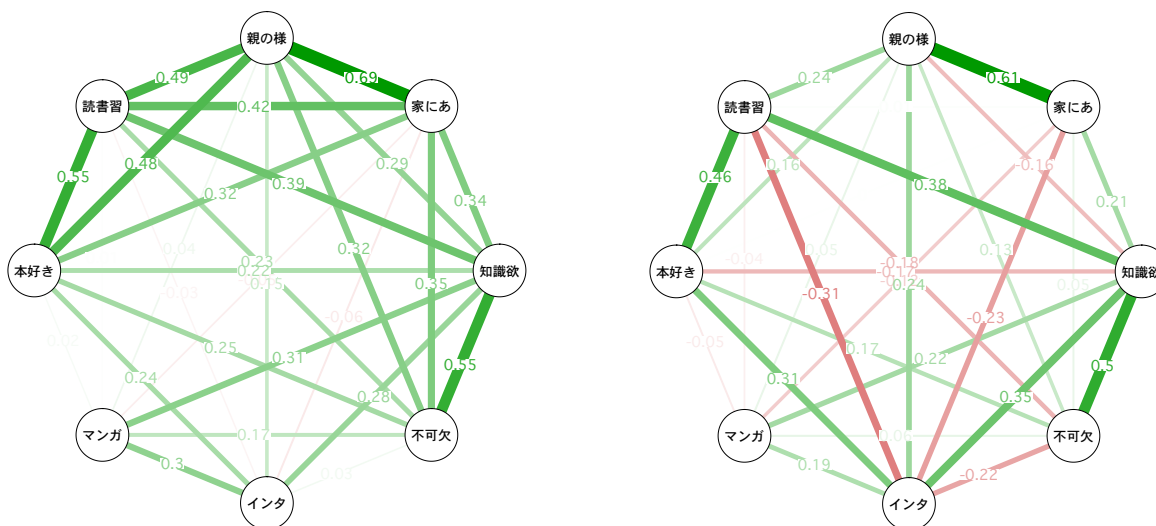
* $p < .05$, ** $p < .01$, *** $p < .001$

最後に、この相関係数、偏相関係数を使って変数間の関連を図示してみます。17日目に紹介した `qgraph` を使います。日本語が使われているので、まずは `par` でフォントを設定しておきます。相関の図は、`cor` の結果をそのまま使えますが、偏相関の方は `pcor(x[labels])` の結果を `per.r2` に代入しておき、そこから偏相関マトリクスの部分 (`$estimate`) だけを取り出して `qgraph` に読ませます。

```
par(family="Osaka")
qgraph(cor(x[labels]),edge.labels=TRUE)
```

```
per.r2 <- pcor(x[labels])
par(family="Osaka")
qgraph(per.r2$estimate,edge.labels=TRUE)
```

でき上がった図は以下のようです (左が相関、右が偏相関)。上の表を図式化しただけで同じ情報なのですが、どうでしょうか。頭の中で表から十分に情報が読み取れる人や、頭の中



で表を図式化できる人は別ですが、多くの人にとっては、こういう図を作りそれを眺めるということが、データが教えてくれることをうまく読み取ることにつながるように思います。もっといえば、自分が見やすいような図を自分で作ってみる（書いてみる）ことが一番だと思えます。

本日はここまでにします。明日は回帰分析を使ってみたいと思います。