title: 主成分分析

主成分分析

1. データXの主成分分析は，各点からの垂線の距離（情報損失量）が最小になるように，固有ベクトルを決める．
2. 主成分得点の分散の最大化も，これと同じことをしている．
3. 固有値は，最小化された情報損失量であり，最大化された主成分得点の分散でもある．
4. Rで主成分分析は`prcomp()`関数を使う．
5. 固有値の平方根（主成分得点の標準偏差）`$sdev`とノルム１で直交する固有ベクトル`$rotation`が出力される．
6. 主成分得点（スコア）は`$x`で出力される．
7. 通常，主成分分析では，各変数の分散は1に標準化した方がよいが，これは引数`scale=T`でできる．
8. 主成分負荷量は，標準化した変数と，それを使った主成分分析のスコアとの相関係数とするのが一般的．
9. `biplot()`関数を使うと，スコアと主成分負荷量という別種の値が１つのグラフにプロットされるが，これらの値は，特異値分解で得られるもので，上記のそれらとは値が異なる．
10. `biplot()`関数によるプロット図は，引数`scale=`に0～1の値で変わる．スコアを一般的な値で出力するなら`scale=0`．負荷量のプロットをXとスコアの相関係数の比率（値そのものは異なる）でプロットしたいなら`scale=1`（これがデフォルト）

考え方

主成分分析のかみ砕いた解説（咀嚼説明）は，有馬・石村（1987）[1]にあります．

以下はそのフォローです．

> 成分ベクトルとXとの距離の最小化であり，
>
> スコアの分散の最大化でもある．

情報損失量Uの最小化

$x_i = (x_{i1}, x_{i2})$と直線 $a_2 x_1 - a_1 x_2 + a_0 = 0$ との距離 をl
で表す．

ただし，$a_1^2 + a_2^2 = 1$ とする．

ヘッセの標準形から

$l_i = |a_2 x_{i1} - a_1 x_{i2} + a_0|$

> **ヘッセの標準形**
>
> 点 (x_1, y_1) から直線 $ax + by + c = 0$ に下した垂線の長さは
>
> $$\frac{|ax_1 + by_1 + c|}{\sqrt{a^2 + b^2}}$$

情報損失量 U を以下で定義して，

$U(a_2, a_1, a_0) = \sum_i l_i^2 = \sum_i (a_2 x_{i1} - a_1 x_{i2} + a_0)^2$

U を最小化するようなa_2, a_1, a_0 を見つけたい.

言い換えれば・・・

$$\min_{a_2, a_1, a_0} U(a_2, a_1, a_0)$$

s.t. $a_2^2 + a_1^2 = 1$

なので，ラグランジュ乗数法より

$$\min_{a_2, a_1, a_0} L = U - \lambda_0 (a_2^2 + a_1^2 - 1) \quad (1)$$

１階の条件として

$$\frac{\partial U}{\partial a_0} = 2 \sum_i (a_2 x_{i1} - a_1 x_{i2} + a_0) = 0 \quad (2)$$

なので，x_{i1}とx_{i2}の平均を，それぞれ$\bar{X}_1$,$\bar{X}_2$とすると，

$a_2 \bar{X}_1 - a_1 \bar{X}_1 + a_0 = 0$

ということであり，$z_{i1} = x_{i1} - \bar{X}_1$，$z_{i2} = x_{i2} - \bar{X}_{i2}$とすると，

$a_2 x_{i1} - a_1 x_{i2} + a_0 = a_2 z_{i1} - a_1 z_{i2}$

なので，

$$\frac{\partial U}{\partial a_2} = 2 \sum_i z_{i1} (a_2 z_{i1} - a_1 z_{i2}) - 2\lambda_0 a_2 = 0 \quad (3)$$

$$\frac{\partial U}{\partial a_1} = -2 \sum_i z_{i2} (a_2 z_{i1} - a_1 z_{i2}) - 2\lambda_0 a_1 = 0 \quad (4)$$

さらに，

$$\frac{\partial U}{\partial \lambda_0} = -(a_2^2 + a_1^2 - 1) = 0 \quad (5)$$

(3)式から

$$a_2 Var(X_1) - a_1 Cov(X_1, X_2) = \frac{\lambda_0}{n-1} a_2 \quad (6)$$

(4)式から

$$-a_2 Cov(X_1, X_2) + a_1 Var(X_2) = \frac{\lambda_0}{n-1} a_1 \quad (7)$$

$\lambda = \lambda_0 / (n-1)$ と書き換えると，

$$\begin{bmatrix} Var(X_1) & -Cov(X_1, X_2) \\ -Cov(X_1, X_2) & Var(X_2) \end{bmatrix} \begin{bmatrix} a_2 \\ a_1 \end{bmatrix} = \lambda \begin{bmatrix} a_2 \\ a_1 \end{bmatrix} \quad (8)$$

ということであり，(a_2, a_1)というのは，n行2列の行列$X = (X_1, -X_2)$の分散・共分散行列の**固有ベクトル**！

λは，Xの分散・共分散行列の**固有値**！

・・・ということ.

さらに，(6)式にa_2を掛けて，(7)式にa_1を掛けて足し合わせると

$$a_2^2 Var(X_1) - 2a_1 a_2 Cov(X_1, X_1) + a_1^2 Var(X_2) = \lambda(a_2^2 + a_1^2)$$

なので，

$$Var(a_2 X_1 - a_1 X_2) = \lambda \quad (9)$$

であり，これは，

$$\frac{1}{n-1} \sum_i (a_2 x - a_1 x_2 + a_0)^2 = \lambda$$

のことなので，λ_0は**情報損失量**Uということ．つまり，

$$U = \lambda_0 \quad (10)$$

スコアの最大化

$a_1 X_1 + a_2 X_2$を**主成分得点** (principal component score)
あるいは単に**スコア** (score) と呼ぶ.

n行2列の行列$X = (X1, X_2)$でスコア$a_1 X_1 + a_2 X_2$を最大化するノルム1の(a_1, a_2)を求める.

$$\max_{a_1, a_2} Var(a_1 X_1 + a_2 X_2)$$

1. 1. $a_1^2 + a_2^2 = 1$

であるから，ラグランジュ乗数法で

$$\max_{a_1, a_2} L = Var(a_1 X_1 + a_2 X_2) - \mu(a_1^2 + a_2^2 - 1)$$

$$= a_1^2 Var(X_1) + 2a_1 a_2 Cov(X_1, X_2) + a_2^2 Var(X_2) - \mu(a_1^2 + a_2^2 - 1)$$
(10)

1階の条件は

$$\frac{\partial L}{\partial a_1} = 2a_1 Var(X_1) + 2a_2 Cov(X_1, X_2) - 2\mu a_1 = 0$$
(11)

$$\frac{\partial L}{\partial a_2} = 2a_1 Cov(X_1, X_2) + 2a_2 Var(X_2) - 2\mu a_2 = 0$$
(12)

したがって，

$$\begin{bmatrix} Var(X_1) & Cov(X_1, X_2) \\ Cov(X_1, X_2) & Var(X_2) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \mu \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$
(13)

であり，(a_1, a_2)は，$X = (X_1, X_2)$ の分散・共分散行列の固有ベクトル

μは，その固有値.

$a_1^2 Var(X_1) + 2a_1 a_2 Cov(X_1, X_2) + a_2^2 Var(X_2) = \mu(a_1^2 + a_2^2)$

なので

$Var(a_1 X_1 + a_2 X_2) = \mu$ (14)

でもある.

Rで

```
library(tidyverse)
```

Rの`pcrcom`関数で

データ

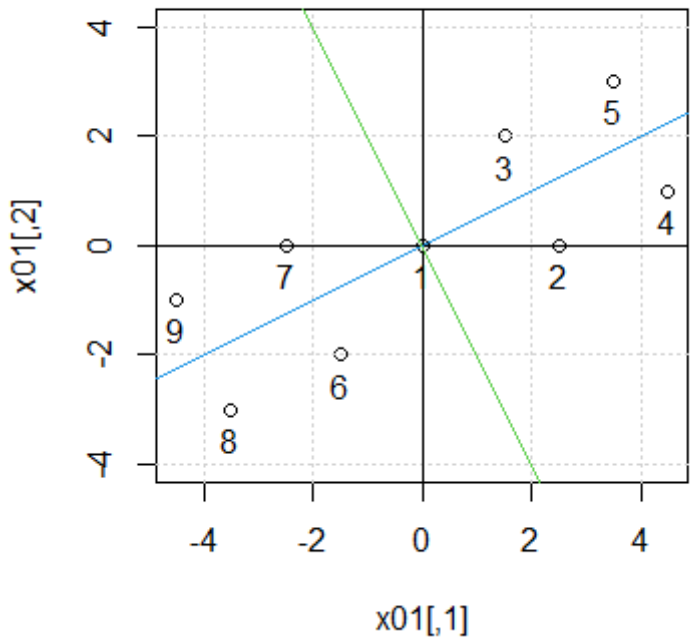
わかりやすいデータを作成.

$X = (X_1, X_2)$の2変数9個のデータ.

```
x01←c(0,0,
  2+0.5,1-1,
  2-0.5,1+1,
  4+0.5,2-1,
  4-0.5,2+1,
  -2+0.5,-1-1,
  -2-0.5,-1+1,
  -4+0.5,-2-1,
  -4-0.5,-2+1
  )%>%
  matrix(ncol=2,byrow=T)
```

こんな感じ

```
plot(x01,asp=1)
grid()
text(1:9,x=x01[,1],y=x01[,2],pos = 1)
abline(h=0)
abline(v=0)
abline(a=0,b=0.5,col=4)
abline(a=0,b=-2,col=3)
```



横軸がX_1（`x01[,1]`），縦軸がX_2（`x01[,2]`）

8つの点が，$(2,1)$の方向の原点を通る直線対して垂直に，長さ$\sqrt{0.5^2+1}=\sqrt{1.25}$で対称にばらついている．

分析しなくても主成分は，$(2,1)$の方向と$(-1,2)$の方向であることはわかる．

平均は２変数とも0．

```
apply(x01,2,mean)%>%
  round(4)
```

```
## [1] 0 0
```

分散・共分散行列Mは

```
var(x01)
```

```
##       [,1] [,2]
## [1,] 10.25  4.5
## [2,]  4.50  3.5
```

主成分分析を行う

`prcomp()` 関数

```
pr01←prcomp(x01)
pr01
```

```
## Standard deviations (1, .., p=2):
## [1] 3.535534 1.118034
##
## Rotation (n x k) = (2 x 2):
##             PC1        PC2
## [1,] -0.8944272 -0.4472136
## [2,] -0.4472136  0.8944272
```

固有値

`Standard deviations` というのは，スコアの標準偏差$\sqrt{\mu}$.

```
pr01$sdev
```

```
## [1] 3.535534 1.118034
```

二乗するとスコアの分散であり，Xの分散・共分散行列Mの固有値μ.

```
pr01$sdev^2
```

```
## [1] 12.50  1.25
```

この場合固有値は2個あるので，2つ出力．最初が第１主成分の，次のが第２主成分の固有値．第１主成分の方に分散が大きいことがわかる．

主成分のベクトル

`Rotation` は，主成分のベクトル (a_1, a_2).

```
pr01$rotation
```

```
##             PC1        PC2
## [1,] -0.8944272 -0.4472136
## [2,] -0.4472136  0.8944272
```

これも２種類あって，１列目が第１主成分で，２列目が第２主成分で，直交している．

第１主成分が$(-2, -1)$の方向で，第２主成分が$(-1, 2)$の方向で，ノルムが1となっている．

```
pr01$rotation[,1]^2
```

```
## [1] 0.8 0.2
```

（横軸がX_1，縦軸がX_2の場合）左下方向と左上方向で，思っていたのと逆だが，どっち方向にスコアをとるかだけの問題で，本質的に違いはない．

各点のスコア

各点のスコアは，`$x` で取り出せる．

```
pr01$x
```

```
##             PC1       PC2
## [1,]  0.000000  0.000000
## [2,] -2.236068 -1.118034
## [3,] -2.236068  1.118034
## [4,] -4.472136 -1.118034
## [5,] -4.472136  1.118034
## [6,]  2.236068 -1.118034
## [7,]  2.236068  1.118034
## [8,]  4.472136 -1.118034
## [9,]  4.472136  1.118034
```

スコアは$a_1 x_1 + b_1 x_2$のことだから，`x01` と `Rotation` を掛け合わせても得られる．

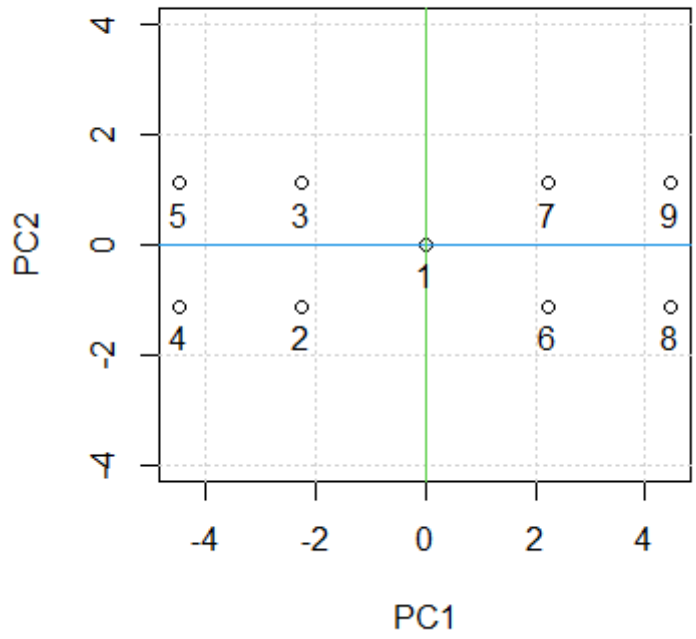
```
x01%*%pr01$rotation
```

```
##             PC1       PC2
## [1,]  0.000000  0.000000
## [2,] -2.236068 -1.118034
## [3,] -2.236068  1.118034
## [4,] -4.472136 -1.118034
## [5,] -4.472136  1.118034
## [6,]  2.236068 -1.118034
## [7,]  2.236068  1.118034
## [8,]  4.472136 -1.118034
## [9,]  4.472136  1.118034
```

各点のスコアのグラフ

第 1 主成分が横軸，第 2 主成分が縦軸

```
plot(pr01$x,asp=1)
grid()
abline(h=0,col=4)
abline(v=0,col=3)
text(1:9,x=pr01$x[,1],y=pr01$x[,2],pos=1)
```



スコアのプロットをマイナス45度回して，さらに＋－をひっくり返した形.
（青い線がもとの真ん中を通る直線）

例えば，２番目の点は，傾き1/2の直線（第１主成分方向）に垂線を下した点が$(2,1)$だから，第１主成分が$-\sqrt{2^2+1^2}=-\sqrt{5}$

```
-sqrt(2^2+1^2)
```

```
## [1] -2.236068
```

第２主成分は，$(2,1)$の点から$(0.5,-1)$だけずらした分だから，$-\sqrt{0.5^2+(-1)^2}=-\sqrt{1.25}$

```
-sqrt(0.5^2+(-1)^2)
```

```
## [1] -1.118034
```

スコアの分散は固有値

```
var(pr01$x)%>%
  round(4)
```

```
##       PC1  PC2
## PC1 12.5 0.00
## PC2  0.0 1.25
```

主成分負荷量

個々の点の特徴を，2つの主成分に分けて見れることはわかった．

それではX_1という変数は，主成分から見てどんな変数か，X_2はどうか，というのが**主成分負荷量** (principal component loading)．

因子負荷量 (factor loading)と呼ぶ人もいるが，因子分析の用語で気持ち悪い．

単に**負荷量** (loading) と呼ぶ場合も．

一般的な定義は

$$\left(\frac{a_1 \sqrt{\mu}}{Var(X_1)}, \frac{a_2 \sqrt{\mu}}{Var(X_2)} \right)$$

```
vinv01<-1/apply(x01,2,sd)
diag(vinv01)%*%pr01$rotation%*%diag(pr01$sdev)%>%
  round(4)
```

```
##         [,1]    [,2]
## [1,] -0.9877 -0.1562
## [2,] -0.8452  0.5345
```

これはXとスコアとの相関係数．

$$\frac{Cov(X_1, a_1 X_1 + a_2 X_2)}{\sqrt{Var(X_1)}\sqrt{Var(a_1 X_1 + a_2 X_2)}} = \frac{a_1 Var(X_1) + a_2 Cov(X_1, X_2)}{\sqrt{Var(X_1)}\sqrt{\mu}} = \frac{a_1 \mu}{\sqrt{Var(X_1)}\sqrt{\mu}}$$

3項目への変換は

$$\begin{bmatrix} Var(X_1) & Cov(X_1, X_2) \\ Cov(X_1, X_2) & Var(X_2) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \mu \begin{bmatrix} a_1 \\ a_2 \end{bmatrix}$$

からきている．

ふつうに相関係数を求めると．

```
cor(x01,pr01$x)%>%
  round(4)
```

```
##          PC1     PC2
## [1,] -0.9877 -0.1562
## [2,] -0.8452  0.5345
```

定義した負荷量と一致する．

なので，主成分負荷量は$[-1, 1]$の値をとる．

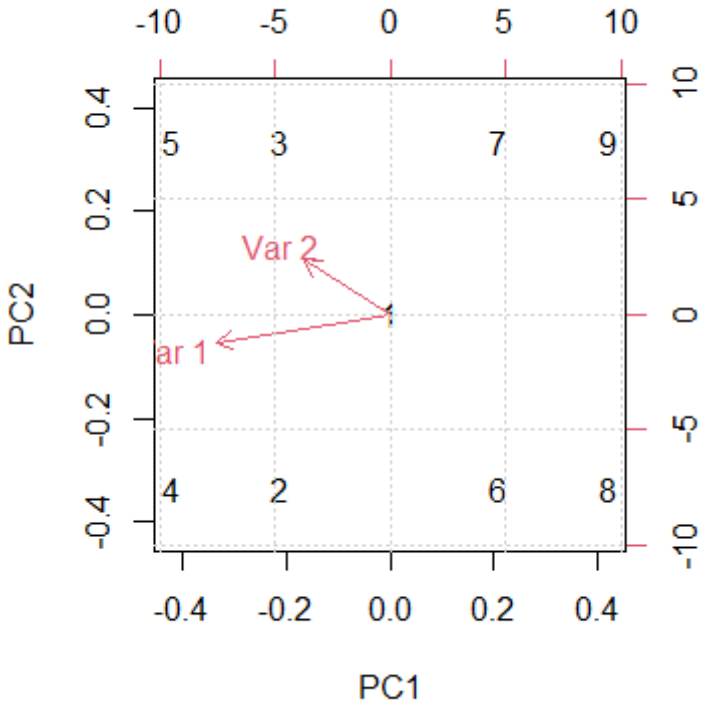
変数X_1は，第１主成分と-0.9877の相関があるが，第２主成分とは-0.1562しかない．

変数X_2は，第１主成分と-0.8452の相関があり，第２主成分とも0.5345の相関がある．

biplot

主成分分析のスコアと主成分負荷量は，`biplot()`関数で出力できる．

```
biplot(pr01,asp=1,scale=1)%>%
grid()
```



スコアと負荷量は全く別の数値だが，無理やり１つのグラフに入れてある．
下と左の目盛がスコア用で，上と右が負荷量のための目盛．

1～9の数字が各点のスコアで，赤い矢印が負荷量・・・だと思ったが，何か数値が違う．

この図を理解するには，Xの特異値分解をすればよくわかる．

特異値分解

Xの特異値分解$X = UDV'$は，Rを使えば`svd()`関数で簡単にできる．

```
svd01←svd(x01)
svd01%>%
  lapply(.,function(x) round(x,4))
```

```
## $d
## [1] 10.0000  3.1623
##
## $u
##         [,1]    [,2]
##  [1,]  0.0000  0.0000
##  [2,] -0.2236 -0.3536
```

```
##  [3,] -0.2236  0.3536
##  [4,] -0.4472 -0.3536
##  [5,] -0.4472  0.3536
##  [6,]  0.2236 -0.3536
##  [7,]  0.2236  0.3536
##  [8,]  0.4472 -0.3536
##  [9,]  0.4472  0.3536
##
## $v
##         [,1]    [,2]
## [1,] -0.8944 -0.4472
## [2,] -0.4472  0.8944
```

```
u01←svd01$u
d01←svd01$d
v01←svd01$v
```

UもVもノルム1の直交するベクトル

```
norm(u01,"2")
```

```
## [1] 1
```

```
norm(v01,"2")
```

```
## [1] 1
```

```
var(u01)%>%
  round(4)
```

```
##       [,1]  [,2]
## [1,] 0.125 0.000
## [2,] 0.000 0.125
```

DUで各点のスコアとなる． ただし，

$$D=\begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}$$

```
u01%*%diag(d01)%>%
  round(4)
```

```
##          [,1]   [,2]
##  [1,]  0.0000  0.000
##  [2,] -2.2361 -1.118
##  [3,] -2.2361  1.118
##  [4,] -4.4721 -1.118
##  [5,] -4.4721  1.118
##  [6,]  2.2361 -1.118
##  [7,]  2.2361  1.118
##  [8,]  4.4721 -1.118
##  [9,]  4.4721  1.118
```

・・・であるはずなのだが，`biplot()`関数では，スコアを以下で表し，

$$U\begin{bmatrix} d_1^{1-scale} & 0 \\ 0 & d_2^{scale} \end{bmatrix}$$

主成分負荷量を以下で表す．

$$V\begin{bmatrix} d_1^{scale} & 0 \\ 0 & d_2^{scale} \end{bmatrix}$$

ただし，$scale$は，$[0,1]$の実数で，通常のスコアの定義は，$scale=0$の場合である．

一方，`biplot()`のスコアは$scale=1$で出しているようだ．

この$scale$の値は，`biplot()`関数の中で，引数`scale=`で与えることができる．

$scale=0$の場合

スコアはUD

```
u01%*%diag(d01^(1-0))%>%
  round(4)
```

```
##         [,1]   [,2]
## [1,]  0.0000  0.000
## [2,] -2.2361 -1.118
## [3,] -2.2361  1.118
## [4,] -4.4721 -1.118
## [5,] -4.4721  1.118
## [6,]  2.2361 -1.118
## [7,]  2.2361  1.118
## [8,]  4.4721 -1.118
## [9,]  4.4721  1.118
```

・・・で，通常のスコアと一致する．

負荷量はV

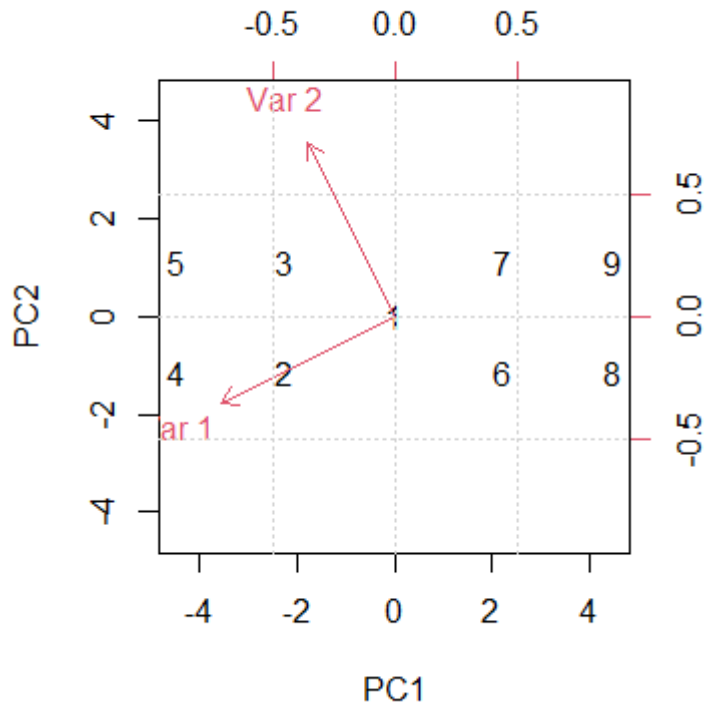
```
v01%*%diag(d01^0)%>%
  round(4)
```

```
##         [,1]    [,2]
## [1,] -0.8944 -0.4472
## [2,] -0.4472  0.8944
```

で，これは主成分ベクトルそのもの．

`biplot`で確認

```
biplot(pr01,asp=1,scale=0)%>%
grid()
```



スコアの値は正確.

負荷量は，定義した通りではあるのだが，90度で直交して，負荷量が相関係数を表していない．（だって，主成分ベクトルだから・・・）

「X_1は，第１主成分にくっついているので，X_1と第１主成分は相関が高い！」・・・とか，負荷量を赤い矢印の傾きで判断したいのなら，相関係数の傾きと同じになってほしい.

相関関係の傾きは

```
c(-0.1562/-0.9877,0.5345/-0.8452)
```

```
## [1]  0.1581452 -0.6323947
```

$scale = 0$の傾きは

```
c(-0.4472/-0.8944,0.8944/-0.4472)
```

```
## [1]  0.5 -2.0
```

・・・と，全然違う．（だって主成分ベクトルだから）

$scale = 1$の場合

スコアはU

```
u01%*%diag(d01^(1-1))%>%
  round(4)
```

```
##         [,1]   [,2]
## [1,]  0.0000  0.0000
## [2,] -0.2236 -0.3536
## [3,] -0.2236  0.3536
## [4,] -0.4472 -0.3536
## [5,] -0.4472  0.3536
## [6,]  0.2236 -0.3536
## [7,]  0.2236  0.3536
## [8,]  0.4472 -0.3536
## [9,]  0.4472  0.3536
```

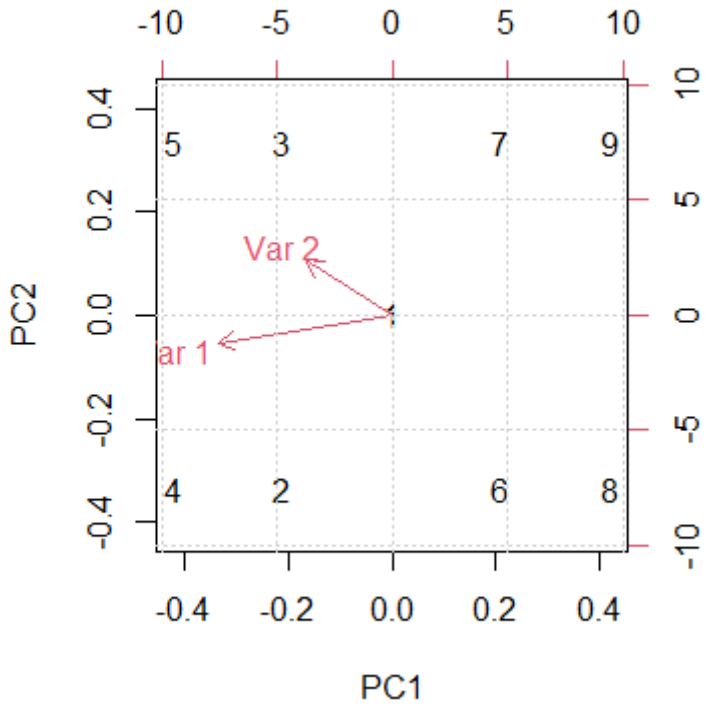
負荷量は，VD

```
v01%*%diag(d01^1)%>%
  round(4)
```

```
##         [,1]   [,2]
## [1,] -8.9443 -1.4142
## [2,] -4.4721  2.8284
```

`biplot` で確認すると

```
biplot(pr01,asp=1,scale=1)%>%
grid()
```



負荷量の傾きは・・・

```
c(-1.4142/-8.9443,2.8284/-4.4721)
```

```
## [1]  0.1581119 -0.6324546
```

相関係数の傾きとほぼ一緒．（値は違うけど）

$$scale = 0.5$$

Dの各要素の平方根をとったものを$D^{\frac{1}{2}}$と表すと・・・

スコアは$UD^{\frac{1}{2}}$

```
u01%*%diag(d01^(1-0.5))%>%
  round(4)
```

```
##         [,1]    [,2]
##  [1,]  0.0000  0.0000
##  [2,] -0.7071 -0.6287
##  [3,] -0.7071  0.6287
##  [4,] -1.4142 -0.6287
##  [5,] -1.4142  0.6287
##  [6,]  0.7071 -0.6287
##  [7,]  0.7071  0.6287
##  [8,]  1.4142 -0.6287
##  [9,]  1.4142  0.6287
```

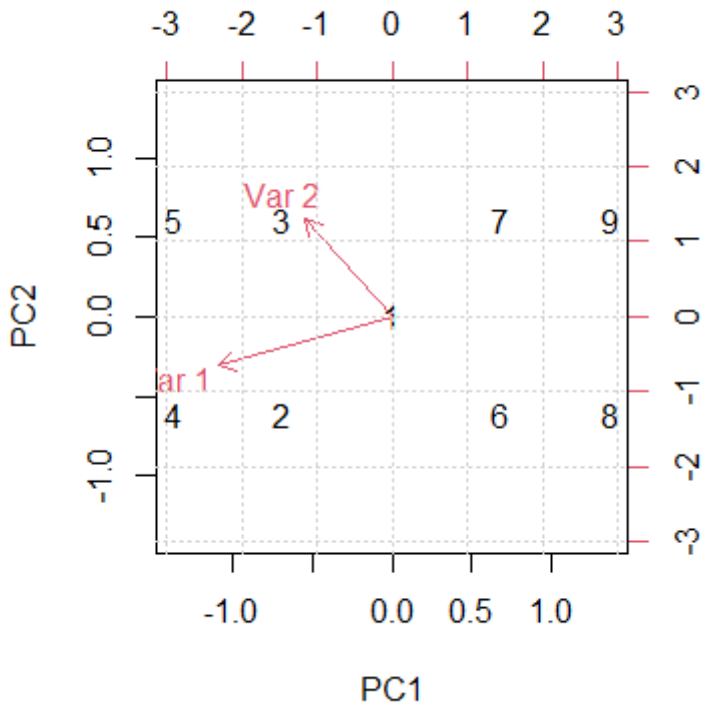
負荷量は$VD^{\frac{1}{2}}$

```
v01%*%diag(d01^0.5)%>%
  round(4)
```

```
##         [,1]    [,2]
## [1,] -2.8284 -0.7953
## [2,] -1.4142  1.5905
```

確認する

```
biplot(pr01,asp=1,scale=0.5)%>%
grid()
```



負荷量の傾きは

```
c(-0.7953/-2.8284,1.5905/-1.4142)
```

```
## [1]  0.2811837 -1.1246641
```

・・・微妙.

結局，負荷量を相関係数として理解するなら，$scale = 1$でbiplotを描くのが最もそれに近い．ただし，目盛は違うので要注意・・・ということ.

寄与率

スコアの分散が固有値だが，分散の和における個々の分散（固有値）の割合は**寄与率**と呼ばれる.

固有値が第２主成分まで，２つある場合 (μ_1, μ_2)
第１主成分の寄与率は

$$\frac{\mu_1}{\mu_1 + \mu_2}$$

２変数しかない場合，$\mu_1 + \mu_2$がXの分散（$Var(X_1) + Var(X_2)$）と一致するので，この寄与率は第１主成分がXの分散を説明する割合ということになる.

スコアの分散は

```
var(pr01$x)%>%
  round(4)
```

```
##      PC1  PC2
## PC1 12.5 0.00
## PC2  0.0 1.25
```

固有値は

```
pr01$sdev^2
```

```
## [1] 12.50  1.25
```

寄与率は

```
pr01$sdev^2/sum(pr01$sdev^2)%>%
  round(4)
```

```
## [1] 0.90909091 0.09090909
```

第１主成分の寄与率が90.9%で，第２主成分の寄与率が9.09%ということ．

寄与率は，`summary()`関数で出力できる．

```
summary(pr01)
```

```
## Importance of components:
##                           PC1     PC2
## Standard deviation     3.5355 1.11803
## Proportion of Variance 0.9091 0.09091
## Cumulative Proportion  0.9091 1.00000
```

`Proportion of Variance`が寄与率．それを大きい方から順に足していったのが累積寄与率．`Cumulative Proportion`がそれ．この場合２変数なので，主成分は高々２個．なので，第２主成分で累積寄与率は1になる．

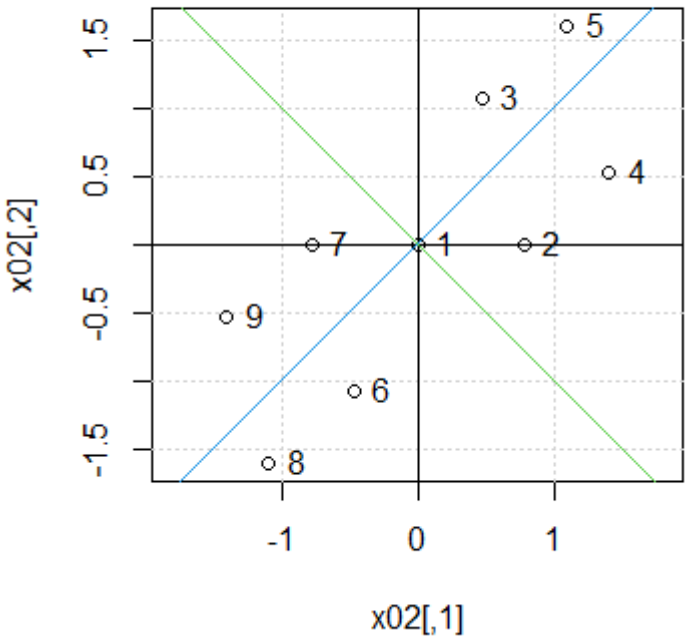
標準化

主成分分析は，Xの各変数の単位をどうとるかで，出力される値が変わってしまうので，（よほど変動を変えたくない場合を除き）通常は**全ての変数の分散を1**に標準化してから主成分分析を行うことが一般的なようだ．

Xを標準化する．

`scale()`

```
x02←scale(x01)
plot(x02,asp=1)
grid()
abline(h=0)
abline(v=0)
abline(a=0,b=1,col=4)
abline(a=0,b=-1,col=3)
text(1:9,x=x02[,1],y=x02[,2],pos=4)
```



平均

```
apply(x02,2,mean)
```

```
## [1] 0 0
```

分散

```
var(x02)
```

```
##           [,1]      [,2]
## [1,] 1.0000000 0.7513055
## [2,] 0.7513055 1.0000000
```

主成分分析

```
pr02←prcomp(x02)
pr02
```

```
## Standard deviations (1, .., p=2):
## [1] 1.3233690 0.4986928
##
## Rotation (n x k) = (2 x 2):
##             PC1        PC2
## [1,] -0.7071068 -0.7071068
## [2,] -0.7071068  0.7071068
```

２変数しかない場合，主成分のベクトルは，2本の45度線.

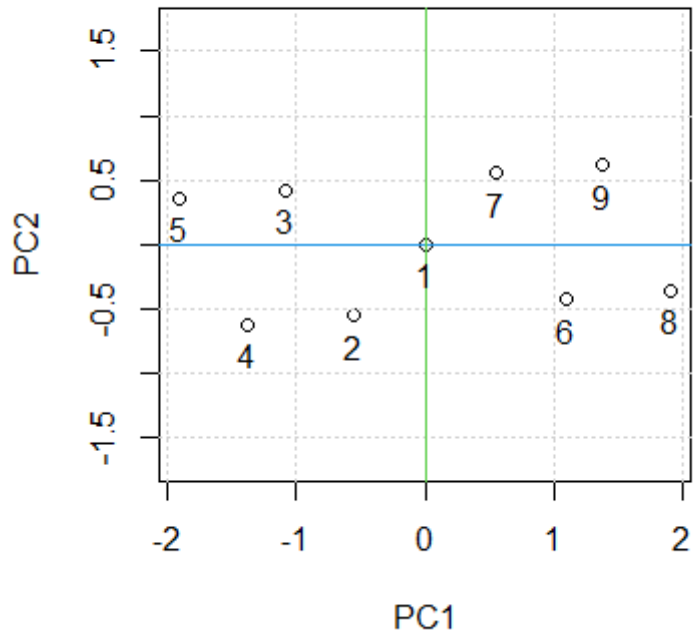
事前にXを標準化しなくても`prcomp()`関数に引数`scale=T`を与えても同じ結果となる.

```
prcomp(x01,scale=T)
```

```
## Standard deviations (1, .., p=2):
## [1] 1.3233690 0.4986928
##
## Rotation (n x k) = (2 x 2):
##             PC1        PC2
## [1,] -0.7071068 -0.7071068
## [2,] -0.7071068  0.7071068
```

スコアのプロット

```
plot(pr02$x,asp=1)
grid()
abline(h=0,col=4)
abline(v=0,col=3)
text(1:9,x=pr02$x[,1],y=pr02$x[,2],pos=1)
```



主成分負荷量

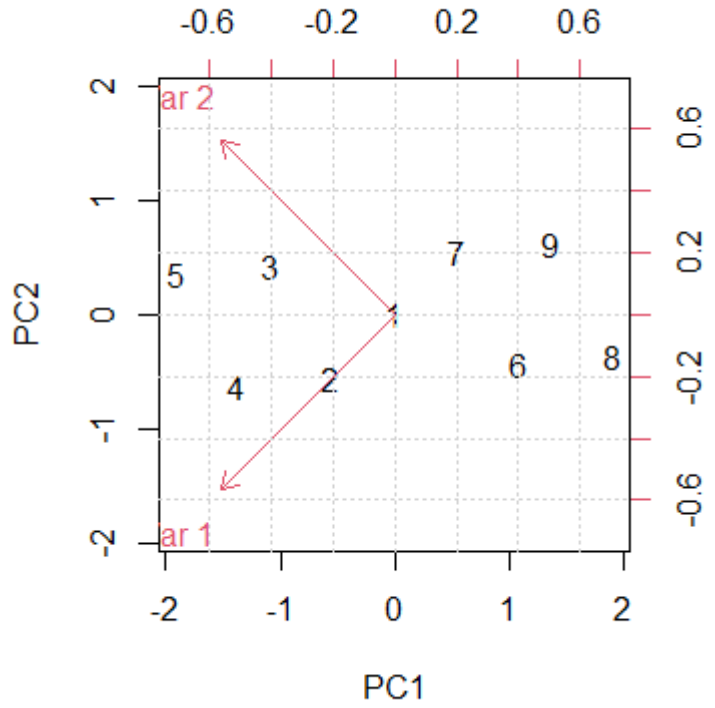
```
cor(x02,pr02$x)
```

```
##             PC1        PC2
## [1,] -0.9357632 -0.3526291
## [2,] -0.9357632  0.3526291
```

biplot

$scale = 0$の場合

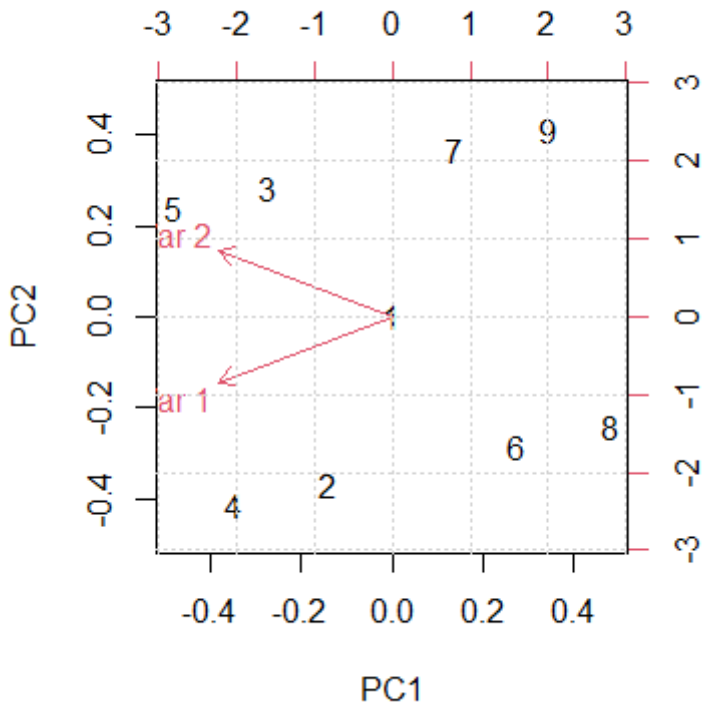
```
biplot(pr02,scale = 0)
grid()
```



負荷量（赤線）の値が，なんかちょっと違う気がするが・・・この辺りは謎.

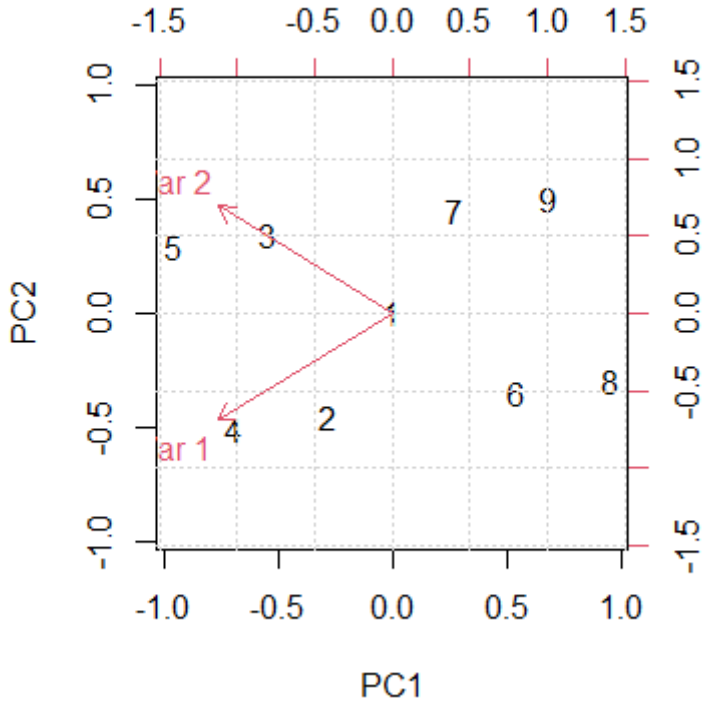
$scale = 1$の場合

```
biplot(pr02,scale = 1)
grid()
```



$scale = 0.5$

```
biplot(pr02,scale = 0.5)
grid()
```



寄与率

```
summary(pr02)
```

```
## Importance of components:
##                           PC1    PC2
## Standard deviation     1.3234 0.4987
## Proportion of Variance 0.8757 0.1244
## Cumulative Proportion  0.8757 1.0000
```

第１主成分の寄与率が，標準化していない場合よりも小さくなった．

主成分分析の結果は変数のスケールに応じて変わるということ．つまり同じデータでも，例えばキログラムで測るのか，それともトンで測るのかでも結果が変わるということ．

それはまずい，ということで，主成分分析では各変数の分散を１に揃えるのが一般的．

ただし，合計に意味があるような場合．例えば，入試の成績などでは，各科目の分散を１に揃えない方が，分析の意味づけがやりやすくなるはず，たぶん．

---

1. 有馬哲・石村貞夫『多変量解析のはなし』東京図書,1987 ↩