

```
> #example 1:  
> #install.packages("calibrate")  
> library(calibrate)
```

Loading required package: MASS

```
> data("heads")  
> data <- -heads[,1:2]  
> tail(data,4)
```

X1 X2

22 174 143

23 176 139

24 197 167

25 190 163

```
> apply(data,2,mean)
```

X1 X2

185.72 151.12

```
> r = cor(data)
```

```
> s = cov(data)
```

```
> eigen(s)
```

eigen() decomposition

\$values

[1] 131.5183 18.1350

\$vectors

[,1] [,2]

[1,] -0.8249295 0.5652357

[2,] -0.5652357 -0.8249295

```
> eigen(r)
```

eigen() decomposition

\$values

[1] 1.7345555 0.2654445

\$vectors

[,1] [,2]

[1,] 0.7071068 - 0.7071068

[2,] 0.7071068 0.7071068

> *pc* <- princomp(*data* , *scores* = *T* , *cor* = *TRUE*)

> *summary*(*pc*)

Importance of components:

Comp. 1 Comp. 2

Standard deviation 1.3170253 0.5152130

Proportion of Variance 0.8672778 0.1327222

Cumulative Proportion 0.8672778 1.0000000

> *pc*\$loadings

Loadings:

Comp. 1 Comp. 2

X1 0.707 0.707

X2 0.707 - 0.707

Comp. 1 Comp. 2

SS loadings 1.0 1.0

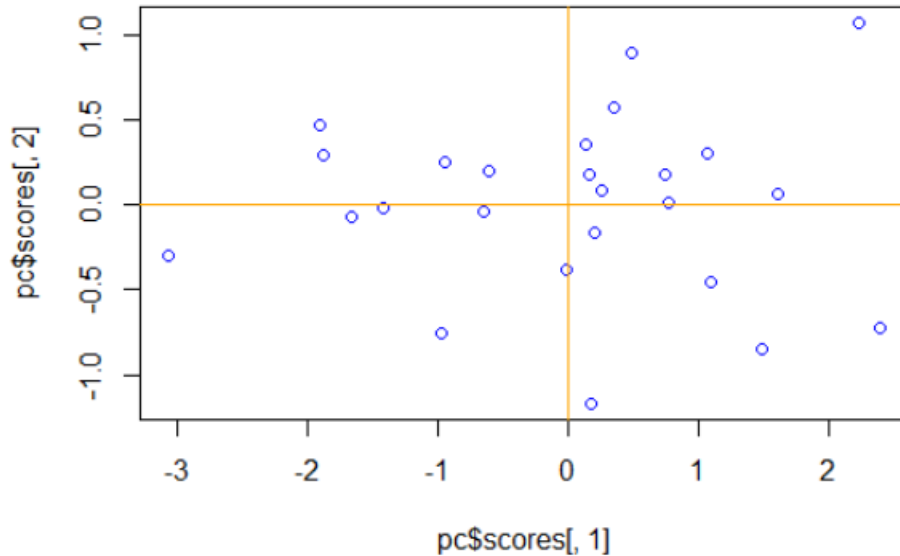
Proportion Var 0.5 0.5

Cumulative Var 0.5 1.0

> *plot*(*pc*\$scores[,1],*pc*\$scores[,2] , *col* = "blue")

> *#plot*(*pc*\$scores , *col* = "blue")

> *abline*(*h* = 0 , *v* = 0 , *col* = "orange")



```
> #Example2
> library(foreign)
> data <- as.data.frame(read.spss("F:/lessons/Multi countios Variate2/data
/ Table 8.3 football.sav"))

re - encoding from UTF - 8
> data = data[31:90,-1]
> tail(data,4)

WDIM CIRCUM FBEYE EYEHD EARHD JAW
87 15.4 55.0 18.8 10.7 14.2 10.8
88 15.5 58.4 19.8 13.1 14.5 11.7
89 15.7 59.0 20.4 12.1 13.0 12.7
90 17.3 61.7 20.7 11.9 13.3 13.3

> apply(data,2,mean)

WDIM CIRCUM FBEYE EYEHD EARHD
15.50000 57.57483 19.80667 10.51333 13.57500

JAW
```

11.87333

> (r = cor(data))

WDIM CIRCUM FBEYE

WDIM 1.00000000 0.61027644 0.36131523

CIRCUM 0.61027644 1.00000000 0.73019762

FBEYE 0.36131523 0.73019762 1.00000000

EYEHD 0.06012317 0.33816708 0.01370847

EARHD 0.25202932 0.09074389 - 0.02795587

JAW 0.60471992 0.40853148 0.31116595

EYEHD EARHD JAW

WDIM 0.06012317 0.25202932 0.60471992

CIRCUM 0.33816708 0.09074389 0.40853148

FBEYE 0.01370847 - 0.02795587 0.31116595

EYEHD 1.00000000 0.29739257 - 0.07874176

EARHD 0.29739257 1.00000000 - 0.08955570

JAW - 0.07874176 - 0.08955570 1.00000000

> (s = cov(data))

WDIM CIRCUM FBEYE

WDIM 0.37016949 0.6020339 0.14881356

CIRCUM 0.60203390 2.6289847 0.80147571

FBEYE 0.14881356 0.8014757 0.45825989

EYEHD 0.04440678 0.6656294 0.01126554

EARHD 0.10711864 0.1027839 - 0.01322034

JAW 0.20932203 0.3768599 0.11984181

EYEHD EARHD JAW

WDIM 0.04440678 0.10711864 0.20932203

CIRCUM 0.66562938 0.10278390 0.37685989

FBEYE 0.01126554 - 0.01322034 0.11984181

EYEHD 1.47371751 0.25220339 - 0.05438418

```

EARHD 0.25220339 0.48800847 - 0.03559322
JAW - 0.05438418 - 0.03559322 0.32368362
> eigen(s)
eigen() decomposition
$values
[1] 3.32341443 1.37430806 0.47606880 0.32468424
[5] 0.15649723 0.08785095

```

```

$vectors
      [1] [2] [3]
[1,] - 0.20744390 0.1415256 - 0.42155301
[2,] - 0.87284535 0.2191281 0.08433843
[3,] - 0.26126472 0.2314010 0.12087973
[4,] - 0.32586218 - 0.8911780 0.17308617
[5,] - 0.06563904 - 0.2220298 - 0.86746687
[6,] - 0.12788326 0.1868463 - 0.13457558
      [4] [5] [6]
[1,] 0.4425457 - 0.1682617 0.7314861
[2,] - 0.1309809 - 0.3304096 - 0.2380849
[3,] - 0.3819134 0.7676277 0.3584302
[4,] 0.1733080 0.1641025 0.1126568
[5,] - 0.3545187 0.1144323 - 0.2347707
[6,] 0.6967211 0.4829502 - 0.4603940

```

```

> eigen(r)
eigen() decomposition
$values
[1] 2.5677926 1.3691056 0.9324399 0.6779565
[5] 0.3220909 0.1306146

```

\$vectors

[1] [2] [3]

[1,] -0.5107369 0.008377599 0.4456068

[2,] -0.5613420 -0.086752441 -0.3196668

[3,] -0.4621079 0.146807619 -0.4753965

[4,] -0.1443386 -0.663982124 -0.3135873

[5,] -0.1096575 -0.644048720 0.4703556

[6,] -0.4214747 0.339382289 0.3920134

[4] [5] [6]

[1,] 0.03305842 -0.62058449 0.39280307

[2,] -0.02117623 -0.22757477 -0.72314844

[3,] -0.47302264 0.31139558 0.46710208

[4,] 0.59227988 0.09531954 0.28297679

[5,] -0.48807667 0.31147128 -0.12926295

[6,] 0.43092227 0.60001436 -0.08827771

> *pc* <- *princomp*(*data*, *scores* = *T*, *cor* = *TRUE*)

> *summary*(*pc*)

Importance of components:

Comp. 1 Comp. 2

Standard deviation 1.6024333 1.1700878

Proportion of Variance 0.4279654 0.2281843

Cumulative Proportion 0.4279654 0.6561497

Comp. 3 Comp. 4

Standard deviation 0.9656293 0.8233811

Proportion of Variance 0.1554066 0.1129927

Cumulative Proportion 0.8115563 0.9245491

Comp. 5 Comp. 6

Standard deviation 0.56753050 0.36140646
Proportion of Variance 0.05368181 0.02176911
Cumulative Proportion 0.97823089 1.00000000
 > *pc\$loadings*

Loadings:

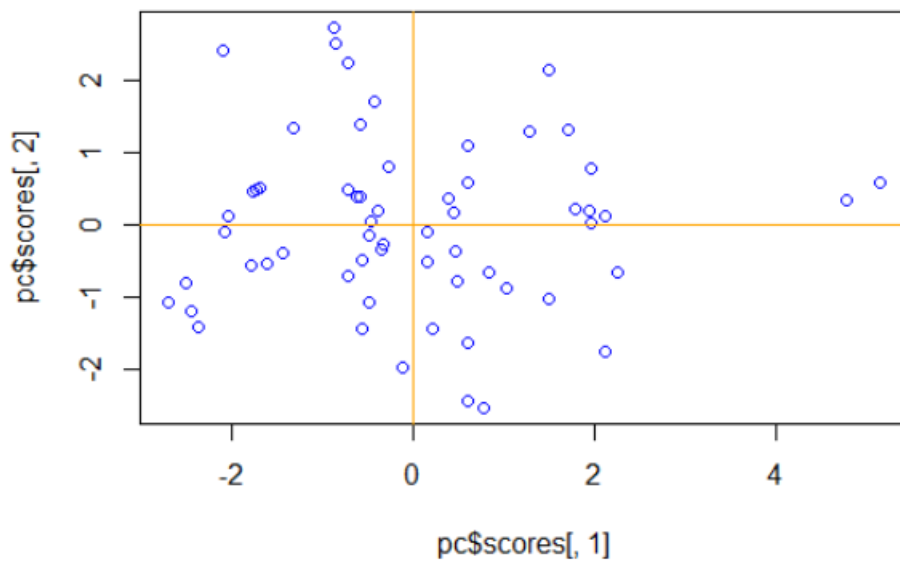
	<i>Comp. 1</i>	<i>Comp. 2</i>	<i>Comp. 3</i>	<i>Comp. 4</i>	<i>Comp. 5</i>
<i>WDIM</i>	0.511	0.446	0.621		
<i>CIRCUM</i>	0.561	- 0.320	0.228		
<i>FBEYE</i>	0.462	0.147	- 0.475	- 0.473	- 0.311
<i>EYEHD</i>	0.144	- 0.664	- 0.314	0.592	
<i>EARHD</i>	0.110	- 0.644	0.470	- 0.488	- 0.311
<i>JAW</i>	0.421	0.339	0.392	0.431	- 0.600

	<i>Comp. 6</i>
<i>WDIM</i>	0.393
<i>CIRCUM</i>	- 0.723
<i>FBEYE</i>	0.467
<i>EYEHD</i>	0.283
<i>EARHD</i>	- 0.129
<i>JAW</i>	

	<i>Comp. 1</i>	<i>Comp. 2</i>	<i>Comp. 3</i>	<i>Comp. 4</i>
<i>SS loadings</i>	1.000	1.000	1.000	1.000
<i>Proportion Var</i>	0.167	0.167	0.167	0.167
<i>Cumulative Var</i>	0.167	0.333	0.500	0.667

	<i>Comp. 5</i>	<i>Comp. 6</i>
<i>SS loadings</i>	1.000	1.000
<i>Proportion Var</i>	0.167	0.167
<i>Cumulative Var</i>	0.833	1.000

```
> plot(pc$scores[,1],pc$scores[,2],col="blue")  
> #plot(pc$scores,col="blue")  
> abline(h=0,v=0,col="orange")
```



تمامی خروجی‌های بالا مربوط به روش مولفه‌های اصلی به کمک ماتریس همبستگی هست و در فایل‌های قبلی نیز به کمک ماتریس واریانس کواریانس قرار داده شده است.

می‌دانیم که روشی که مبتنی بر ماتریس همبستگی هست، بهتر است زیرا دیگر به مقیاس‌ها ربطی ندارد. و اینجا هم برای دو سوال 2 مولفه اول مدنظر هست و تفاوت کمی در میزان درصد واریانس مولفه‌ها از کل واریانس وجود دارد و همانطور که مشاهده می‌کنید خروجی نمودارها هم تفاوت خیلی خاصی ندارد و فقط از همان لحاظ پراکندگی کمی داده‌ها متفاوت هستند.

تمامی خروجی‌های امتیاز بندی (همان مقادیری که به کمک آن رسم صورت گرفته) در خروجی دستور score مشاهده می‌کنید و خروجی‌های دستور loading هم ضرایب هریک از متغیرها در مولفه اصلی اول، دوم و.... را نشان می‌دهد.