Chapter7 - Correspondence Analysis

张笑竹 / 201618070114

2018年12月3日

第七章实验采用课本例7-1、例7-2中给出的数据进行对应分析。例7-1是关于种族和学历的数据；例7-2是31个省、直辖市、自治区农民家庭人均纯收入的数据。
将本次的实验任务拆分如下：
1）调用ca包中的函数，对例7-1进行对应分析；
2）继续调用ca()函数，对例7-2中的分类汇总数据进行对应分析。

1 例7-1的分析

1.1 列联表及其检验

首先，在R中读入例7-1的数据集，并对其进行处理。其中，由于Degree变量的7、8、9为缺失值，故只选择Degree变量为1-4的数据。然后，对变量各个水平的标签进行更改。

```
#读取数据
X=read.csv("eg7-1.csv")
X=X[,-1]
X=subset(X,degree<5)
X$degree<-factor(X$degree,labels=c("Less than high school","High school","Junior college","Bache
lor","Graduate"))
X$race<-factor(X$race,labels=c("white","black","others"))
head(X)
```

```
##        degree  race
## 1 High school white
## 2    Bachelor black
## 3    Bachelor white
## 4 High school white
## 5    Graduate white
## 6 High school white
```

下面，输出Degree变量和Race变量的列联表，并进行卡方检验。

```
addmargins(table(X),c(1,2))
```

```
##                        race
## degree                  white black others  Sum
##   Less than high school   214    48     17  279
##   High school             658    92     30  780
##   Junior college           74    13      3   90
##   Bachelor                209     7     18  234
##   Graduate                 99     7      7  113
##   Sum                    1254   167     75 1496
```

```
addmargins(prop.table(table(X)))
```

```
##                          race
## degree                          white       black      others         Sum
##   Less than high school 0.143048128 0.032085561 0.011363636 0.186497326
##   High school           0.439839572 0.061497326 0.020053476 0.521390374
##   Junior college        0.049465241 0.008689840 0.002005348 0.060160428
##   Bachelor              0.139705882 0.004679144 0.012032086 0.156417112
##   Graduate              0.066176471 0.004679144 0.004679144 0.075534759
##   Sum                   0.838235294 0.111631016 0.050133690 1.000000000
```

```
chisq.test(table(X))
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(X)
## X-squared = 36.482, df = 8, p-value = 1.433e-05
```

根据输出的结果，$\chi^2 = 36.482$,得到的p值接近于零。因此，可以拒绝变量Degree和Race相互独立的假定，接下来的因子分析是有必要的。

1.2 利用函数ca()进行因子分析

加载ca包，直接调用ca()函数，进行因子分析。

```
library(ca)
ca.x=ca(table(X))
ca.x
```

```
##
##  Principal inertias (eigenvalues):
##            1        2
## Value      0.020786 0.003601
## Percentage 85.23%   14.77%
##
##
##  Rows:
##         Less than high school High school Junior college  Bachelor
## Mass                 0.186497    0.521390       0.060160  0.156417
## ChiDist              0.202654    0.055761       0.124825  0.278806
## Inertia              0.007659    0.001621       0.000937  0.012159
## Dim. 1              -1.217441   -0.206512      -0.801486  1.902870
## Dim. 2              -1.688145    0.785718       0.786794 -0.828163
##          Graduate
## Mass     0.075535
## ChiDist  0.163119
## Inertia  0.002010
## Dim. 1   1.129272
## Dim. 2  -0.167158
##
##
##  Columns:
##            white     black    others
## Mass    0.838235  0.111631  0.050134
## ChiDist 0.047063  0.400335  0.304183
## Inertia 0.001857  0.017891  0.004639
## Dim. 1  0.297415 -2.767419  1.189338
## Dim. 2  0.323306 -0.547246 -4.187140
```

根据输出的结果可以看出，一共得到了2个特征根，$\lambda_1 = 0.021, \lambda_2 = 0.004$，其中第一个特征根解释了原信息的85.23%.

首先，分析输出的行剖面信息“Rows”．“Mass”指列联表中的边缘概率，在这里是$p_{i.}$．而“Inertia”，指的是行剖面惯量，可以表达为

$$\sum_{j=1}^{p} p_{i.}\left(\frac{p_{ij}}{p_{i.}\sqrt{p_{.j}}} - \sqrt{p_{i.}}\right)^2 = \sum_{j=1}^{p} \frac{(p_{ij} - p_{i.}p_{j.})^2}{p_{i.}p_{j.}}$$

最后，输出结果中还有Dim.1和Dim.2，他们分别是λ_1和λ_2对应的特征向量u_1和u_2.

对于输出的列剖面信息“Columns”，所表达信息与行信息类似，只是这里的边缘概率为$p_{.j}$，而列剖面惯量变为

$$\sum_{i=1}^{n} p_{.j}\left(\frac{p_{ij}}{p_{.j}\sqrt{p_{i.}}} - \sqrt{p_{.j}}\right)^2 = \sum_{i=1}^{n} \frac{(p_{ij} - p_{i.}p_{j.})^2}{p_{i.}p_{j.}}$$

事实上，行剖面的特征向量u_1和u_2，与列剖面的特征向量$u_1^{'}$和$u_2^{'}$存在如下关系：

$$u_1^{'} = Zu_1, \quad u_2^{'} = Zu_2$$

这里，Σ_r和Σ_c分别为行剖面和列剖面的协方差矩阵，且

$$\Sigma_r = Z^{'}Z, \quad \Sigma_c = ZZ^{'}$$

$$Z = (z_{ij})_{n\times p} \qquad z_{ij} = \frac{p_{ij} - p_{i.}p_{.j}}{\sqrt{p_{i.}p_{.j}}}$$

此外，还可以看到，行剖面的总惯量与列剖面的总惯量存在如下的关系：

$$\sum_{i=1}^{n}\sum_{j=1}^{p}\frac{(p_{ij} - p_{i.}p_{j.})^2}{p_{i.}p_{j.}} = I_I = I_J = 0.024$$

而这恰好与$\lambda_1 + \lambda_2 = 0.024$相等，说明了2个特征根就是对原始信息的分解。
最后，结合1.1节中的数据，还可以发现

$$36.482 = \chi^2 = nI_I = nI_J = 1496 \times 0.024$$

这在一定程度上验证了I_I和I_J服从χ^2/n.

1.3 获取更多信息

1.3.1 二维图

首先，可以从结果列表隐藏的对象中，调取行与列各个状态在二维图中的坐标值。

```
ca.x$rowcoord
```

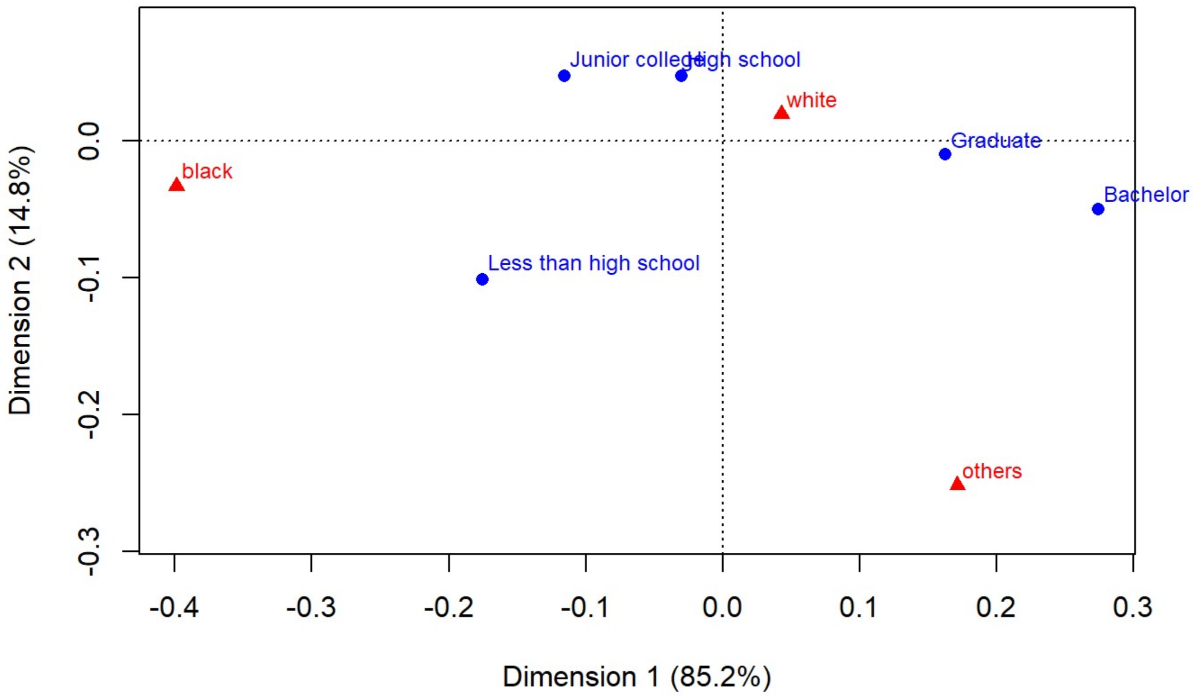
```
##                              Dim1       Dim2
## Less than high school -1.2174411 -1.6881450
## High school           -0.2065121  0.7857179
## Junior college        -0.8014865  0.7867943
## Bachelor               1.9028696 -0.8281632
## Graduate               1.1292722 -0.1671580
```

```
ca.x$colcoord
```

```
##               Dim1       Dim2
## white   0.2974151  0.3233059
## black  -2.7674186 -0.5472461
## others  1.1893380 -4.1871400
```

将各个状态所代表的点，大致地画入二维图。

```
plot(ca.x)
```



明显地，白人受教育程度一般较高，其与学历较高的点比较接近；而黑人学历较低，与Less than high school比较靠近；other的最高学历没有明显特点。

1.3.2 因子载荷矩阵

除了利用图形展示状态之间的关系，还可以定量地进行分析。首先，可以计算各个状态与因子的相关系数，即因子载荷矩阵。其表达式为

$$a_{ik} = u_{ik} \cdot \sqrt{\lambda_k}$$

对第一个变量degree，计算得到

```
##因子载荷(第一个变量)
dim11=c(-1.217441,-0.206512,-0.801486,1.902870,1.129272)
dim12=c(-1.688145,0.785718,0.786794,-0.828163,-0.167158)
a11=dim11*ca.x$sv[1]   #第1个因子与第1个变量
a12=dim12*ca.x$sv[2]   #第2个因子与第1个变量
loadings1<-data.frame(a11,a12)
rownames(loadings1)=c("Less than high school","High school","Junior college","Bachelor","Graduat
e")
colnames(loadings1)=c("dim1","dim2")
loadings1
```

```
##                              dim1        dim2
## Less than high school -0.17552154 -0.10129653
## High school           -0.02977336  0.04714672
## Junior college        -0.11555226  0.04721129
## Bachelor               0.27434156 -0.04969362
## Graduate               0.16280999 -0.01003026
```

可见维度1与Bachelor联系更为密切，而维度2与Less than high school联系更为密切。
对第二个变量race，计算得到

```
##因子载荷(第二个变量)
dim21=c(0.297415,-2.767419,1.189338)
dim22=c(0.323306,-0.547246,-4.187140)
a21=dim21*ca.x$sv[1]   #第1个因子与第2个变量
a22=dim22*ca.x$sv[2]   #第2个因子与第2个变量
loadings2<-data.frame(a21,a22)
rownames(loadings2)=c("white","black","others")
colnames(loadings2)=c("dim1","dim2")
loadings2
```

```
##               dim1        dim2
## white   0.04287907  0.01939986
## black  -0.39898577 -0.03283730
## others  0.17146985 -0.25124782
```

可见维度2与Black联系更为密切，而维度2与Others联系更为密切。

1.3.3 $CTR(i)$

状态i对公共因子的贡献为$CTR(i)$，$CTR(i)$的值越大，说明状态i对第k个公共因子的贡献越大。其表达式为

$$CTR(i)=p_i\cdot\frac{a_{ik}^2}{\lambda_k}$$

对第一个变量degree，计算得到

```
#CTR(i)
#第一个变量
CTR11=ca.x$rowmass*dim11^2   #第1个因子与第1个变量
CTR12=ca.x$rowmass*dim12^2   #第2个因子与第1个变量
CTR1=data.frame(CTR11,CTR12)
rownames(CTR1)=c("Less than high school","High school","Junior college","Bachelor","Graduate")
colnames(CTR1)=c("dim1","dim2")
CTR1
```

```
##                             dim1        dim2
## Less than high school 0.27641936 0.531486336
## High school           0.02223584 0.321881795
## Junior college        0.03864584 0.037242000
## Bachelor              0.56637295 0.107279295
## Graduate              0.09632610 0.002110577
```

根据结果，Bachelor对维度1的贡献最大，而Less than high school对维度2的贡献最大。

对第二个变量race，计算得到

```
#第二个变量
CTR21=ca.x$colmass*dim21^2  #第1个因子与第1个变量
CTR22=ca.x$colmass*dim22^2  #第2个因子与第1个变量
CTR2=data.frame(CTR21,CTR22)
rownames(CTR2)=c("white","black","others")
colnames(CTR2)=c("dim1","dim2")
CTR2
```

```
##              dim1       dim2
## white  0.07414667 0.08761803
## black  0.85493818 0.03343105
## others 0.07091535 0.87895094
```

类似地，Black对维度1的贡献最大，而Others对维度2的贡献最大。

1.4 行剖面和列剖面

1.4.1 行剖面

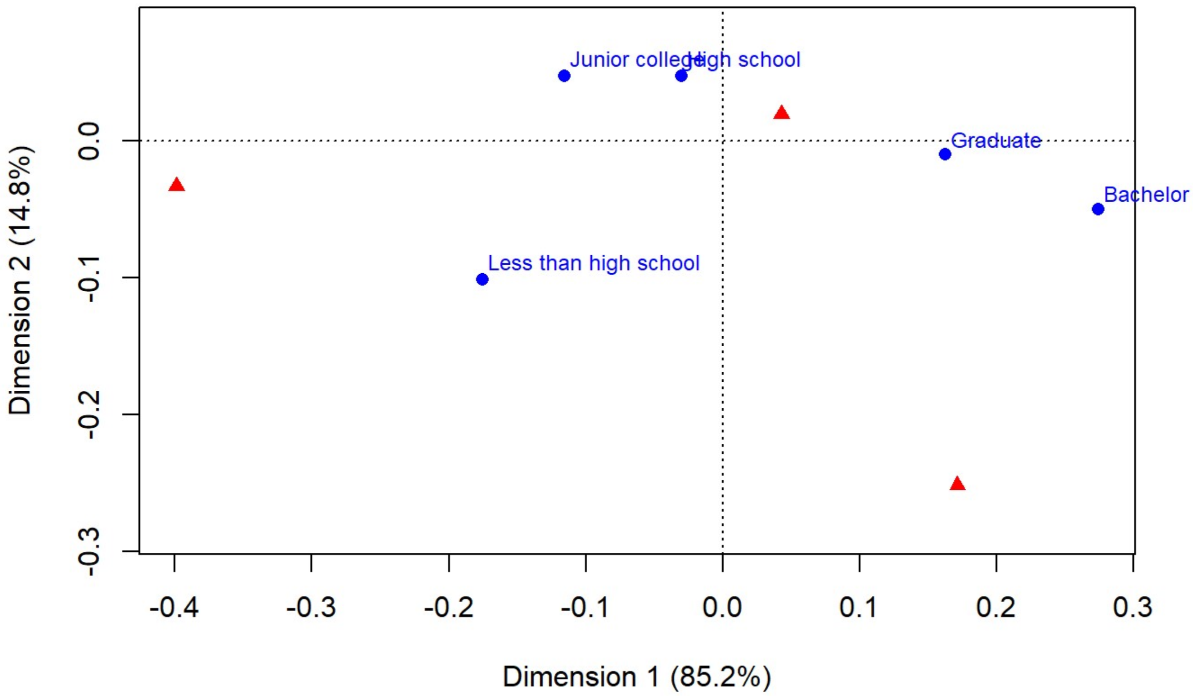
为了对行剖面有更清晰的了解，不妨计算行剖面并画出二维图。

```
RowPro=c()
for(i in 1:5){
  z1=table(X)[i,]/rowSums(table(X))[i]
  RowPro=rbind(RowPro,z1)
}
RowPro=rbind(RowPro,ca.x$colmass)
rownames(RowPro)=c("Less than high school","High school","Junior college","Bachelor","Graduate",
"MASS")
RowPro
```

```
##                          white      black     others
## Less than high school 0.7670251 0.17204301 0.06093190
## High school           0.8435897 0.11794872 0.03846154
## Junior college        0.8222222 0.14444444 0.03333333
## Bachelor              0.8931624 0.02991453 0.07692308
## Graduate              0.8761062 0.06194690 0.06194690
## MASS                  0.8382353 0.11163102 0.05013369
```

对行剖面进行横向观察，发现无论在那一种学历中，白人比例都多于黑人比例，而黑人比例又多于其他种族的比例。然而，这样的分析意义不大，因为在数据集中白人整体的比例就更高。一种好的分析方法是，对每一列进行观察。不难发现，白人在各个学历所占比例中，在Bachelor中的比例最大；黑人在各个学历所占比例中，在Less than high school中的比例最大；其他种族在各个学历所占比例中，在Bachelor中的比例也是最大的，但是和其他学历的区别不是非常明显。

```
Tab2=table(X)
colnames(Tab2)<-NULL
plot(ca(Tab2))
```



观察二维图，High School及以上学历状态之间的距离相近，而Less than high school可以单独归为一类。

1.4.2 列剖面

同样地，计算列剖面并画出二维图。

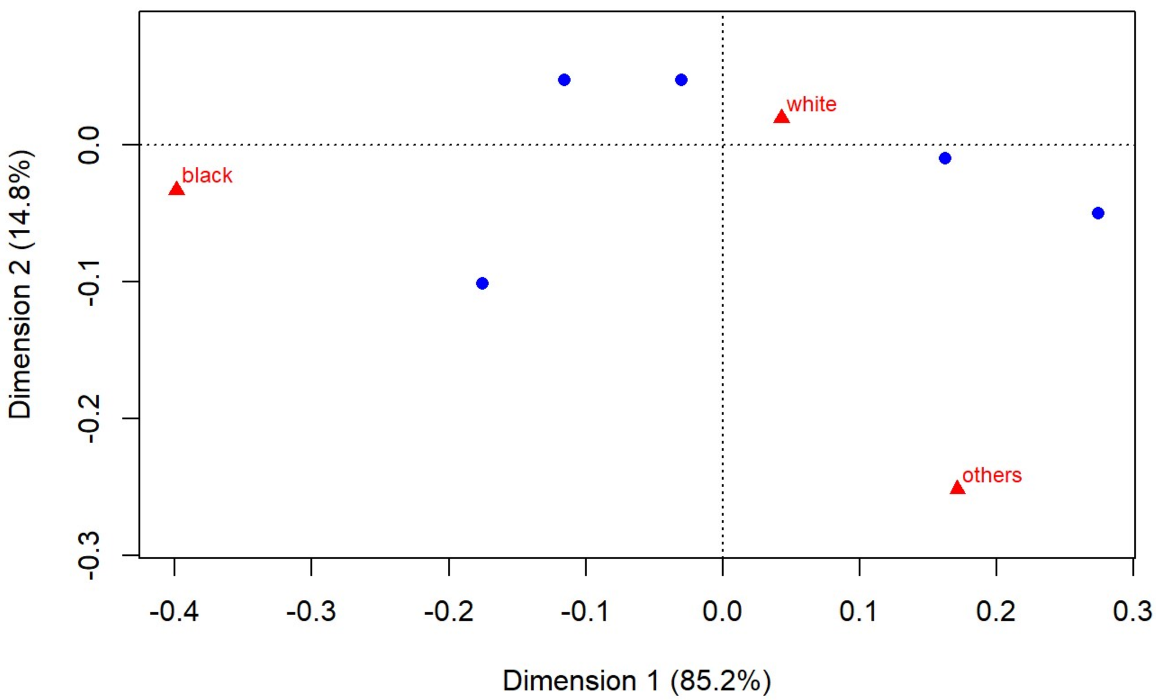
```
#列剖面
ColPro=c()
for(i in 1:3){
  z2=table(X)[,i]/colSums(table(X))[i]
  ColPro=cbind(ColPro,z2)
}
ColPro=cbind(ColPro,ca.x$rowmass)
colnames(ColPro)=c("white","black","others","MASS")
ColPro
```

```
##                            white      black     others       MASS
## Less than high school 0.17065391 0.28742515 0.22666667 0.18649733
## High school           0.52472089 0.55089820 0.40000000 0.52139037
## Junior college        0.05901116 0.07784431 0.04000000 0.06016043
## Bachelor              0.16666667 0.04191617 0.24000000 0.15641711
## Graduate              0.07894737 0.04191617 0.09333333 0.07553476
```

对列剖面进行纵向观察，发现无论在那一种族中，High school 的比例都是最大的。然而，这样的分析意义不大，因为在数据集中High School整体的比例就是最大的。一种好的分析方法是，对每一行进行观察。不难发现，Less than high shcool在各个种族所占比例中，在黑人中的比例最大；High school在各个种族所占比例中，仍然是在黑人中的比例最大；Junior College在各个种族所占比例中，依旧是在黑人中的比例最大；而Bachelor在各个种族所占比例中，在其他种族中的比例最大；最后，Graduate在各个种族所占比例中，也是在其他种族中的比例最大。

这样的分析结果似乎与先前的有所不同，白人不在在任何一个学历中占得上风，在高学历中反而是其他种族更具优势。结合实际分析，这或许是由于其他种族的范围过于宽泛，既包括普遍拥有高学历的亚裔，也包括普遍学历较低的拉丁裔等。

```
 Tab1=table(X)
rownames(Tab1)<-NULL
plot(ca(Tab1))
```



最后根据二维图，不难看出，白人、黑人和其他种族之间的距离都较远，很明显地形成了三大类。

2 例7-2的分析

2.1 利用ca()函数进行分析

首先，读入例7-2的数据，并对数据进行调整和处理。

```
Y=read.csv("eg7-2.csv")
rownames(Y)=Y[,1]
Y=Y[,-1]
head(Y)
```

```
##        工资性收入 家庭经营纯收入 财产性收入 转移性收入
## 北京    10843.484       1318.105  1716.3567  2597.7949
## 天津     7922.257       4126.286   920.9994  1055.9947
## 河北     4005.282       3254.567   218.3022   603.2344
## 山西     3175.504       2334.408   140.7999   705.9148
## 内蒙古   1459.055       4689.107   322.9771  1140.1715
## 辽宁     3630.236       4783.350   246.1730   723.9565
```

然后，重复上述做法，直接利用ca()函数，对例7-2进行分析。

```
ca.y=ca(Y,nd=2)
ca.y
```

```
##
##  Principal inertias (eigenvalues):
##            1        2        3
## Value      0.121937 0.023226 0.007294
## Percentage 79.98%   15.23%   4.78%
##
##
##  Rows:
##                北京      天津      河北      山西      内蒙古    辽宁
## Mass     0.062561  0.053257  0.030686  0.024137  0.028901 0.035632
## ChiDist  0.706095  0.291120  0.122350  0.133399  0.528191 0.210060
## Inertia  0.031191  0.004514  0.000459  0.000430  0.008063 0.001572
## Dim. 1  -1.959070 -0.709974 -0.031237 -0.211929  1.222764 0.591035
## Dim. 2  -0.774469  0.638398  0.779759  0.237759 -2.035455 0.256629
##                吉林      黑龙江    上海      江苏      浙江      安徽
## Mass     0.032649  0.032670  0.067604  0.046333  0.055256 0.027189
## ChiDist  0.528415  0.520635  0.769398  0.213300  0.168212 0.161680
## Inertia  0.009116  0.008856  0.040020  0.002108  0.001563 0.000711
## Dim. 1   1.405330  1.249265 -2.058933 -0.531172 -0.278503 0.292984
## Dim. 2  -1.029485 -1.220518 -1.703163  0.674396  0.877666 0.706191
##              福建     江西     山东     河南     湖北     湖南      广东
## Mass    0.037847 0.029730 0.035870 0.028574 0.029814  0.028252  0.040033
## ChiDist 0.153465 0.216032 0.154755 0.268649 0.279956  0.174688  0.415422
## Inertia 0.000891 0.001387 0.000859 0.002062 0.002337  0.000862  0.006909
## Dim. 1  0.278314 0.410061 0.225103 0.697283 0.694996 -0.086832 -0.996039
## Dim. 2  0.751536 1.038269 0.873947 0.743226 0.834947  0.964059  1.433808
##              广西     海南      重庆     四川     贵州      云南      西藏
## Mass    0.022812 0.028129  0.028036 0.026586 0.018048  0.020568  0.021717
## ChiDist 0.287645 0.318875  0.088029 0.087225 0.167799  0.431911  0.525834
## Inertia 0.001887 0.002860  0.000217 0.000202 0.000508  0.003837  0.006005
## Dim. 1  0.780417 0.912942 -0.006216 0.142136 0.403393  1.175319  1.393123
## Dim. 2  0.368303 0.040738  0.016936 0.044076 0.203863 -0.487736 -1.291848
##                陕西      甘肃      青海     宁夏      新疆
## Mass     0.021881  0.017113  0.020369 0.023468  0.024278
## ChiDist  0.048329  0.147034  0.344118 0.202563  0.616731
## Inertia  0.000051  0.000370  0.002412 0.000963  0.009234
## Dim. 1  -0.057978  0.374504  0.109610 0.531501  1.516544
## Dim. 2   0.268735 -0.225243 -1.746308 0.401427 -2.056830
##
##
##  Columns:
##         工资性收入 家庭经营纯收入 财产性收入 转移性收入
## Mass      0.451454       0.408573   0.039227   0.100746
## ChiDist   0.308352       0.417897   0.643768   0.466492
## Inertia   0.042925       0.071352   0.016257   0.021924
## Dim. 1   -0.826507       1.196315  -1.281792  -0.648884
## Dim. 2    0.704335       0.018102  -1.725246  -2.557861
```

根据输出结果，一共有3个因子，提取了前两个，共解释了原始信息的95.21%. 由于省份变量过于繁琐，故下面只分析收入变量的各个状态。

2.2 计算因子载荷

下面，利用公式

$$a_{ik} = u_{ik} \cdot \sqrt{\lambda_k}$$

计算得到“收入”变量的因子载荷：

```
##因子载荷(第二个变量)
dim1=c(-0.826507,1.196315,-1.281792,-0.648884)
dim2=c(0.704335,0.018102,-1.725246,-2.557861)
a1=dim1*ca.y$sv[1]  #第1个因子与第2个变量
a2=dim2*ca.y$sv[2]  #第2个因子与第2个变量
loadings<-data.frame(a1,a2)
rownames(loadings)=c("工资性收入","家庭经营纯收入","财产性收入","转移性收入")
colnames(loadings)=c("dim1","dim2")
loadings
```

```
##                     dim1        dim2
## 工资性收入     -0.2886124  0.107341384
## 家庭经营纯收入  0.4177476  0.002758764
## 财产性收入     -0.4475958 -0.262929277
## 转移性收入     -0.2265872 -0.389820666
```

通过结果不难发现，因子1与“财产性收入”联系更为密切，而因子2与“转移性收入”联系更为密切。

2.3 $CTR(i)$

此外，还可以计算状态i对公共因子的贡献为$CTR(i)$. 利用公式

$$CTR(i) = p_i \cdot \frac{a_{ik}^2}{\lambda_k}$$

计算得到：

```
CTR1=ca.y$colmass*dim1^2  #第1个因子与第2个变量
CTR2=ca.y$colmass*dim2^2  #第2个因子与第2个变量
CTR=data.frame(CTR1,CTR2)
rownames(CTR)=c("工资性收入","家庭经营纯收入","财产性收入","转移性收入")
colnames(CTR)=c("dim1","dim2")
CTR
```

```
##                     dim1         dim2
## 工资性收入     0.30839429 0.2239606910
## 家庭经营纯收入 0.58473772 0.0001338823
## 财产性收入     0.06444882 0.1167567247
## 转移性收入     0.04241932 0.6591487365
```

根据结果可以得到结论，“家庭经营纯收入”对因子1的贡献更大，而“转移性收入”对因子2的贡献更大。

2.4 二维图

最后，输出各个状态在二维图上的坐标，并画出二维图。

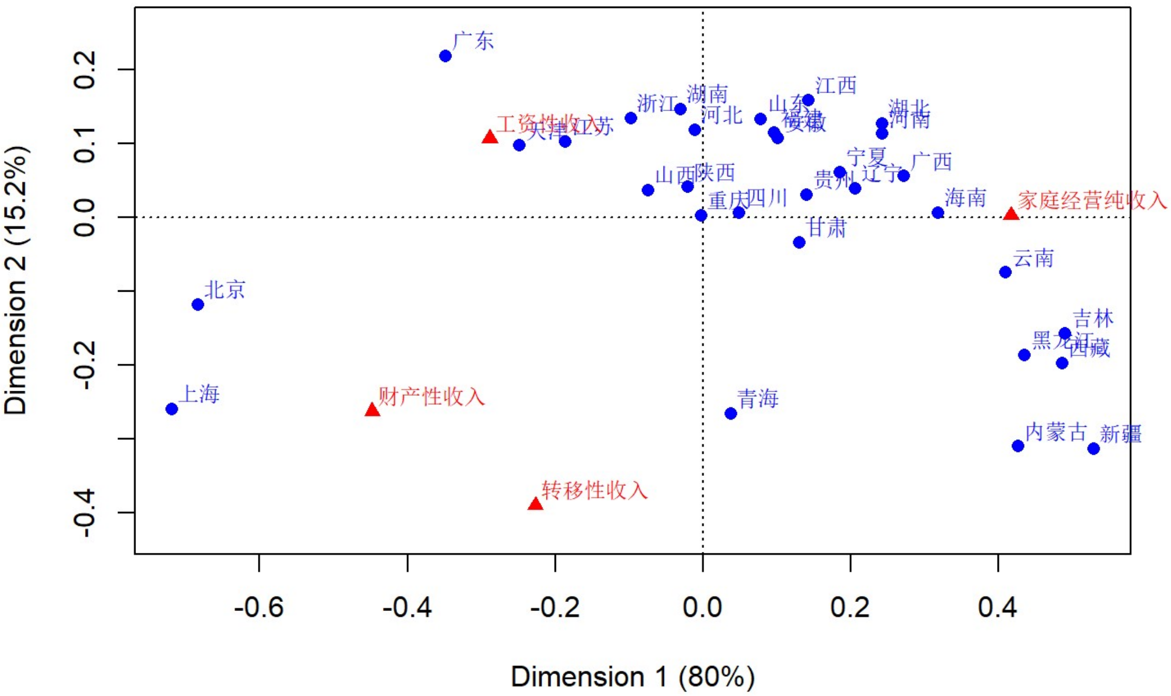
```
ca.y$rowcoord
```

```
##                Dim1        Dim2
## 北京   -1.959070020 -0.77446913
## 天津   -0.709974275  0.63839766
## 河北   -0.031237244  0.77975862
## 山西   -0.211929104  0.23775850
## 内蒙古  1.222763712 -2.03545452
## 辽宁    0.591035266  0.25662856
## 吉林    1.405329828 -1.02948543
## 黑龙江  1.249264567 -1.22051790
## 上海   -2.058933049 -1.70316270
## 江苏   -0.531172276  0.67439561
## 浙江   -0.278503062  0.87766597
## 安徽    0.292983554  0.70619115
## 福建    0.278314115  0.75153584
## 江西    0.410060890  1.03826891
## 山东    0.225103491  0.87394703
## 河南    0.697282599  0.74322582
## 湖北    0.694996075  0.83494741
## 湖南   -0.086832348  0.96405925
## 广东   -0.996039093  1.43380832
## 广西    0.780416508  0.36830273
## 海南    0.912941661  0.04073824
## 重庆   -0.006216169  0.01693571
## 四川    0.142135985  0.04407557
## 贵州    0.403393414  0.20386287
## 云南    1.175318778 -0.48773623
## 西藏    1.393123010 -1.29184779
## 陕西   -0.057978482  0.26873506
## 甘肃    0.374503602 -0.22524291
## 青海    0.109609643 -1.74630844
## 宁夏    0.531500802  0.40142715
## 新疆    1.516543662 -2.05682958
```

```
ca.y$colcoord
```

```
##                     Dim1        Dim2
## 工资性收入      -0.8265066  0.70433499
## 家庭经营纯收入   1.1963152  0.01810209
## 财产性收入      -1.2817922 -1.72524620
## 转移性收入      -0.6488840 -2.55786090
```

```
plot(ca.y)
```



从输出结果中不难看出，我国经济发达省区，如浙江、广东、江苏、天津等，主要以工资性收入为主；而广西、海南和云南等多依靠家庭经营性收入；个别省区，如上海、北京，经济发展迅速，会有相当部分的转移性收入和财产性收入。