Классификация без обучения

Alexeyeva N.P.

24 04 2020

Подготовка данных

```
library("knitr", lib.loc="/Library/Frameworks/R.framework/Versions/3.3/Resources/library")
life <- read.csv(
  "~/Documents/share/R/lessions/A_PrincipleComponents/life.csv", sep=";")

data<-data.frame(L=life$life,
                 M=life$manager,
                 P=life$pover,
                 A=life$auto,
                 V=life$vodka); row.names(data)<-life$X
```

```
kable(data,caption="Данные о средней продолжительности жизни и
      сопутствующих факторах.",label="tab:12")
```

Таблица 1: Данные о средней продолжительности жизни и сопутствующих факторах.

	L	M	P	A	V
1970	68.9	1060	7.8	5.5	25.3
1975	68.1	1101	9.5	15.3	28.0
1980	67.6	1147	10.1	30.2	30.0
1985	69.2	1204	10.0	44.5	23.5
1990	69.2	1602	9.8	58.6	18.0
1995	64.6	1893	5.5	93.3	38.4
1998	67.0	2777	6.2	122.0	29.6

Основной алгоритм метода главных компонент

Нормируем и центрируем данные, вычисляем корреляционную матрицу.

```
data.0<-apply(data,2,function(x)(x-mean(x))/sd(x))
Sigma<-cov(data.0)
kable(Sigma, digits=3,caption="Корреляционная матрица.")
```

Таблица 2: Корреляционная матрица.

	L	M	P	A	V
L	1.000	-0.501	0.769	-0.595	-0.934
M	-0.501	1.000	-0.700	0.954	0.297
P	0.769	-0.700	1.000	-0.669	-0.681
A	-0.595	0.954	-0.669	1.000	0.370
V	-0.934	0.297	-0.681	0.370	1.000

Вычисляем собственные числа ковариационной матрицы и собственные векторы. Поскольку данные нормированы и центрированы, то ковариационная матрицв совпадает с корреляционной.

```
ei<-eigen(Sigma)
df<-data.frame('С.Ч'=ei$values,'Процент'=cumsum(ei$values)/sum(ei$values)*100,
               row.names = as.character(seq(dim(Sigma)[1])));
kable(df,
caption="Собственные числа и суммарный  вклад компонент в общую дисперсию.")
```

Таблица 3: Собственные числа и суммарный вклад компонент в общую дисперсию.

С.Ч	Процент
3.6000120	72.00024
1.0899853	93.79995
0.2449248	98.69844
0.0465064	99.62857
0.0185715	100.00000

```
kable(data.frame(ei$vectors),digits=3,caption="Собственные векторы.")
```

Таблица 4: Собственные векторы.

X1	X2	X3	X4	X5
0.472	-0.381	0.297	0.499	-0.543
-0.430	-0.540	-0.033	0.538	0.483
0.477	-0.043	-0.852	0.167	0.133
-0.446	-0.465	-0.381	-0.331	-0.574
-0.407	0.587	-0.201	0.569	-0.354

Убедимся в том, что сумма собственных чисел совпадает с суммарной дисперсий нормированных признаков.

```
c(sum(ei$values),sum(diag(cov(data.0))) )
```

```
## [1] 5 5
```

Вычисляем главные компоненты,

```
Scores<-data.0 %*% ei$vectors
```

проверяем, действительно ли дисперсии главных компонент совпадают с собственными числами корреляционной матрицы,

```
apply(Scores,2,var)
```

```
## [1] 3.60001200 1.08998530 0.24492480 0.04650638 0.01857152
```

```
ei$values
```

```
## [1] 3.60001200 1.08998530 0.24492480 0.04650638 0.01857152
```

затем факторы (нормированные главные компоненты)

```
factors<-apply(Scores,2, function(x) x/sd(x))
kable(data.frame(factors), digits=3,caption="Значения факторов")
```

Таблица 5: Значения факторов

	X1	X2	X3	X4	X5
1970	0.600	0.465	2.008	0.170	-0.061
1975	0.540	0.710	-0.161	0.658	0.887
1980	0.375	0.790	-1.285	0.669	0.357
1985	0.724	-0.339	-0.460	-0.113	-2.020
1990	0.662	-1.301	-0.227	-1.408	0.977
1995	-1.587	1.019	-0.038	-1.232	-0.206
1998	-1.314	-1.345	0.162	1.256	0.065

и факторные нагрузки как коэффициенты корреляции между признаками и факторами.

```
Matr<-apply(Scores,2,function(y)apply(data.0,2,function(x)cor(x,y)))
kable(data.frame(Matr), caption="Матрица факторных нагрузок")
```

Таблица 6: Матрица факторных нагрузок

	X1	X2	X3	X4	X5
L	0.8961099	-0.3979026	0.1470056	0.1076214	-0.0739427
M	-0.8150340	-0.5636009	-0.0161356	0.1161283	0.0657835
P	0.9048438	-0.0447356	-0.4214645	0.0360203	0.0180717
A	-0.8469578	-0.4857210	-0.1884832	-0.0713071	-0.0782748
V	-0.7724239	0.6132590	-0.0994764	0.1227032	-0.0481976

Функция в R для факторного анализа

```
data.0<-scale(data)

pc<-prcomp(data.0)


kable(data.frame(lambda=pc$sdev^2,EI=ei$values),
      digits=3,
      caption="Дисперсии главных компонент, вычисленные разными методами")
```

Таблица 7: Дисперсии главных компонент, вычисленные разными методами

lambda	EI
3.600	3.600
1.090	1.090
0.245	0.245
0.047	0.047
0.019	0.019

```
kable(pc$rotation[,seq(2)],digits=3,
      caption="Матрица собственных векторов  из prcomp.")
```

Таблица 8: Матрица собственных векторов из prcomp.

	PC1	PC2
L	-0.472	0.381
M	0.430	0.540
P	-0.477	0.043
A	0.446	0.465
V	0.407	-0.587

```
kable(data.frame(ei$vectors[,seq(2)]),digits=3,
    caption="Матрица собственных векторов ковариационной матрицы.")
```

Таблица 9: Матрица собственных векторов ковариационной матрицы.

X1	X2
0.472	-0.381
-0.430	-0.540
0.477	-0.043
-0.446	-0.465
-0.407	0.587

```
kable(pc$x[,seq(2)],digits=3,
    caption="Значения главных компонент  из prcomp.")
```

Таблица 10: Значения главных компонент из prcomp.

	PC1	PC2
1970	-1.139	-0.486
1975	-1.024	-0.741
1980	-0.711	-0.825
1985	-1.374	0.354
1990	-1.256	1.358
1995	3.012	-1.064
1998	2.492	1.404

```
kable(data.frame(Scores[,seq(2)]),digits=3,
    caption="Значения главных компонент  из преобразования при помощи с.в.")
```

Таблица 11: Значения главных компонент из преобразования при помощи с.в.

	X1	X2
1970	1.139	0.486
1975	1.024	0.741
1980	0.711	0.825
1985	1.374	-0.354
1990	1.256	-1.358
1995	-3.012	1.064

	X1	X2
1998	-2.492	-1.404

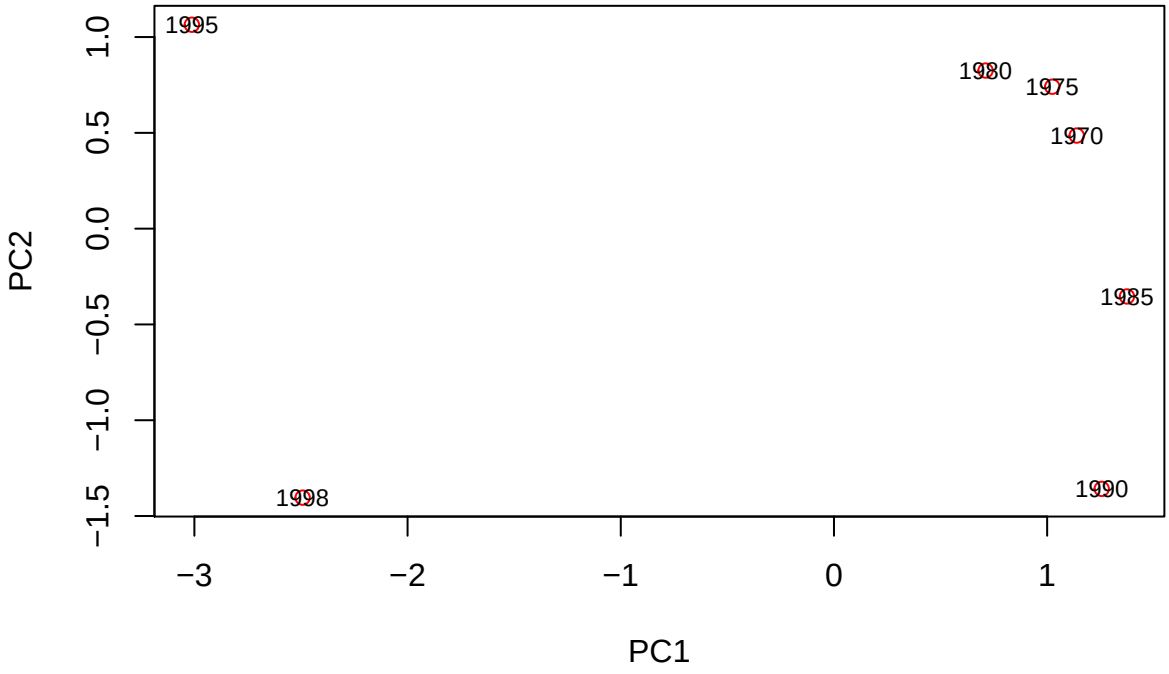
Чтобы получить факторные нагрузки, нужно элементы с.в. умножить на корень из соответствующего собственного числа.

```
MatrPC<-apply(rbind(pc$rotation[,seq(2)],pc$sdev[seq(2)]),2,function(x)x[-length(x)]*x[length(x)])
kable(MatrPC,caption="Вычисление факторных нагрузок")
```

Таблица 12: Вычисление факторных нагрузок

	PC1	PC2
L	-0.8961099	0.3979026
M	0.8150340	0.5636009
P	-0.9048438	0.0447356
A	0.8469578	0.4857210
V	0.7724239	-0.6132590

```
plot(Scores[,seq(2)],type="n",xlab="PC1",ylab="PC2")
lines((-1)*pc$x[,seq(2)],col=2,type="p")
text(Scores[,seq(2)],as.character(life[,1]),cex=0.75)
```

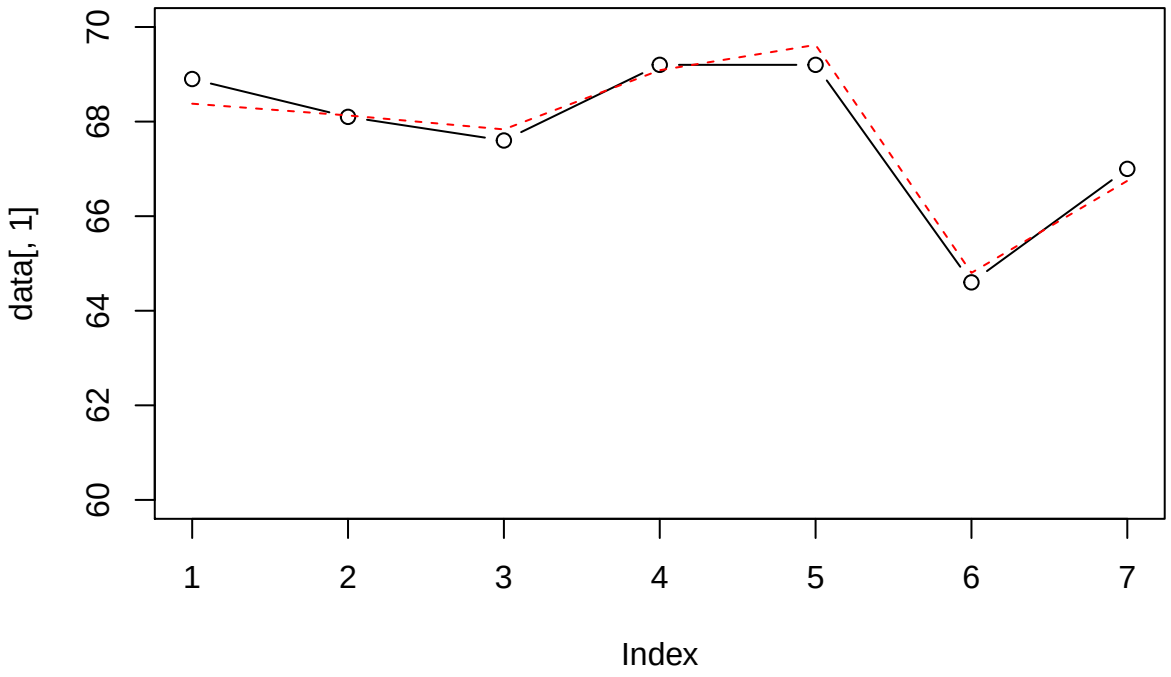


Восстановление переменных по первым двум факторам

Для восстановления данных по главным компонентам используем формулу произведения матрицы факторных нагрузок на значения факторов.

```
sigma2<-apply(data,2,var)
Rest<-function(k,sigma2,Matr,factors,data)
  {
  XX<-Matr[,seq(k)]%*%t(factors[,seq(k)])
  XX.1<-apply(cbind(XX,sqrt(sigma2)),1,function(x)x[-length(x)]*x[length(x)])
  XX.2<-apply(rbind(XX.1,colMeans(data)),2,function(x)x[-length(x)]+x[length(x)])
return(XX.2)}
```

```
plot(data[,1],type="b",ylim=c(60,70))
lines(Rest(2,sigma2,Matr,factors,data)[,1],lty=2,col=2)
```



Интерпретация факторов

Признаки: L – средняя продолжительность жизни; M – количество чиновников; A – количество автомобилей; P – доходы бедных; V – объемы продажи водки.

Вклад первого фактора равен 0.72. Значения в i-й строке и j-м столбце соответствуют коэффициенту корреляции между i-м признаком и j-й главной компонентой. Чем больше первый фактор, тем больше продолжительность жизни и доходы бедных, меньше чиновников и автомобилей и не много водки – фактор какого-то благополучия, во всяком случае корреляция первого фактора с числом самоубийц отрицательная.

```
kable(cor(cbind(life[,c("rich","auto","suicide")],Scores[,seq(2)])),
      caption="Корреляционная матрица факторов с другими признаками.")
```

Таблица 13: Корреляционная матрица факторов с другими признаками.

	rich	auto	suicide	1	2
rich	1.0000000	-0.5539969	-0.8677292	0.8531189	-0.2109657
auto	-0.5539969	1.0000000	0.6333100	-0.8469578	-0.4857210

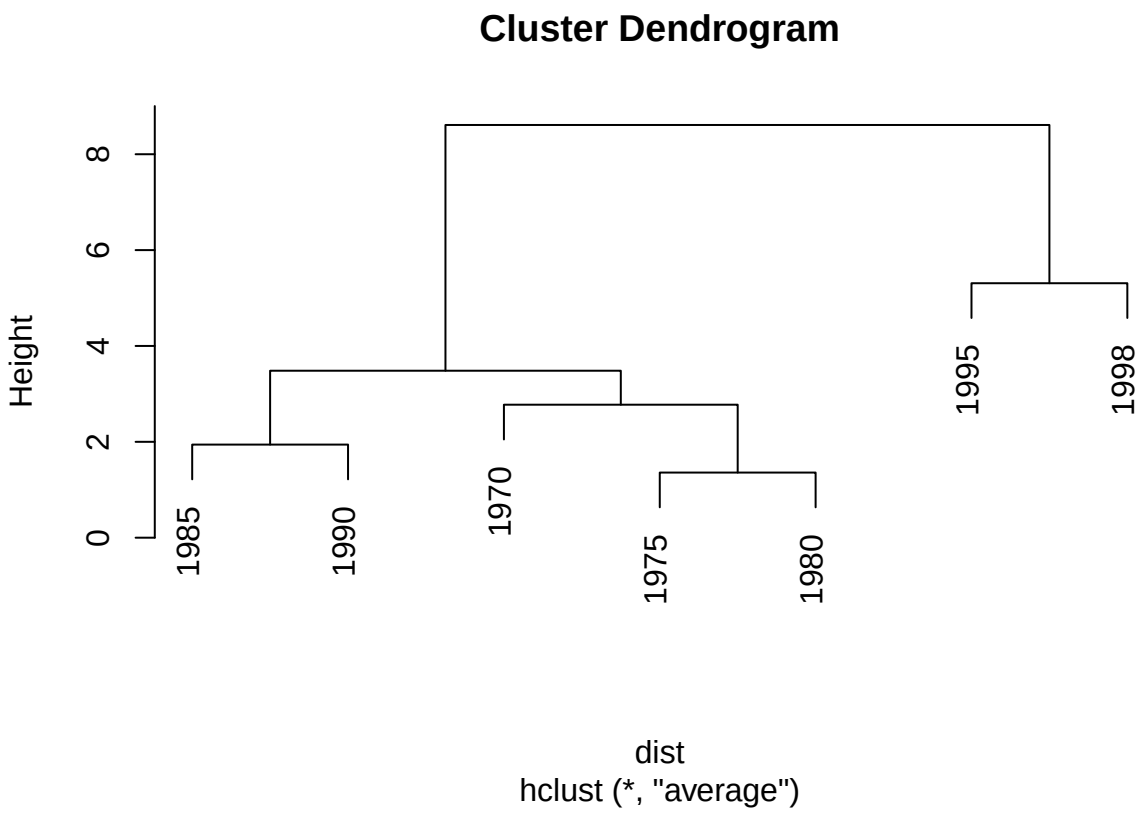
	rich	auto	suicide	1	2
suicide	-0.8677292	0.6333100	1.0000000	-0.7175992	-0.2485356
1	0.8531189	-0.8469578	-0.7175992	1.0000000	0.0000000
2	-0.2109657	-0.4857210	-0.2485356	0.0000000	1.0000000

Кластерный анализ

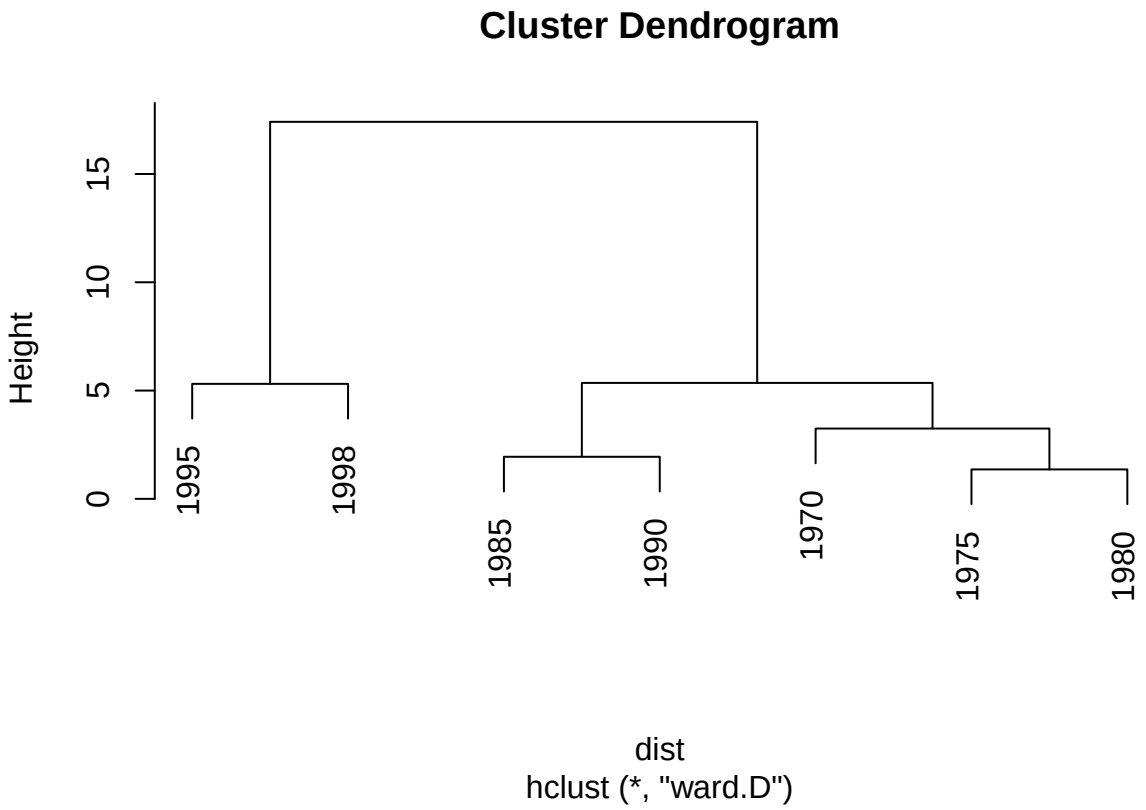
```
library("cluster")


dist<-daisy(data.0,metric="manhattan");dist #матрица расстояний
```

```
## Dissimilarities :
##          1970     1975     1980     1985     1990     1995
## 1975 2.094934
## 1980 3.453804 1.358870
## 1985 2.764317 2.497320 2.483477
## 1990 4.498783 4.231787 4.425760 1.942283
## 1995 9.296350 8.967855 8.232435 9.752107 9.546196
## 1998 8.173134 7.844639 7.235500 8.628891 8.422980 5.308948
##
## Metric :  manhattan
## Number of objects : 7
```

```
h<-hclust(dist,method="average")
plot(h)
```



```
dist.T<-daisy(t(data.0),metric="euclidean")  #матрица расстояний
h<-hclust(dist,method="ward.D")
plot(h)
```



Если нужно выделить нужное число кластеров, то используется метод k-средних.

```
km<-kmeans(data.0,2)
km
```

```
## K-means clustering with 2 clusters of sizes 2, 5
##
## Cluster means:
##           L          M          P          A          V
## 1 -1.2194215  1.2696413 -1.3322516  1.3003991  1.019273
## 2  0.4877686 -0.5078565  0.5329007 -0.5201596 -0.407709
##
## Clustering vector:
## 1970 1975 1980 1985 1990 1995 1998
##    2    2    2    2    2    1    1
##
## Within cluster sum of squares by cluster:
## [1] 3.330800 5.378485
##  (between_SS / total_SS =  71.0 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
km<-kmeans(data.0,3)
km
```

```
## K-means clustering with 3 clusters of sizes 2, 3, 2
##
## Cluster means:
##           L          M          P          A           V
## 1 -1.2194215  1.2696413 -1.3322516  1.30039905  1.01927261
## 2  0.2438843 -0.6998514  0.3735747 -0.84763744  0.03532877
## 3  0.8535951 -0.2198642  0.7718895 -0.02894289 -1.07226577
##
## Clustering vector:
## 1970 1975 1980 1985 1990 1995 1998
##    2    2    2    3    3    1    1
##
## Within cluster sum of squares by cluster:
## [1] 3.3308001 1.5487219 0.6403839
##  (between_SS / total_SS =  81.6 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

Задание для самостоятельной работы

Нужно выполнить систематизацию наблюдений по нескольким признакам (от 5 до 10) при помощи методов главных компонент и кластерного анализа. Важно выяснить, насколько можно доверять восстановлению данных по первым двум-трем главным компонентам. В кластерном анализе выполнить кластеризацию индивидов, признаков и выделить оптимальное число кластеров по индивидам. В качестве рабочего материала можно взять кардиологические данные финских алкоголиков. Использование других данных не возбраняется.