MAE 5905: Introdução à Ciência de Dados - Lista 4

Leonardo Lima - 14334311 Leonardo Makoto - 7180679

2023-06-20

Questão 1

Determine as componentes principais para o conjunto de dados *iris* disponível por meio do comando **`data(iris)`** no pacote R.

```r
# Carregando os pacotes de manipulação de dados
library(tidyverse)

# vamos carregar o pacote para produção de gráfico de correlação corrplot
library(corrplot)

# carregando a base de dados
data("iris")
```

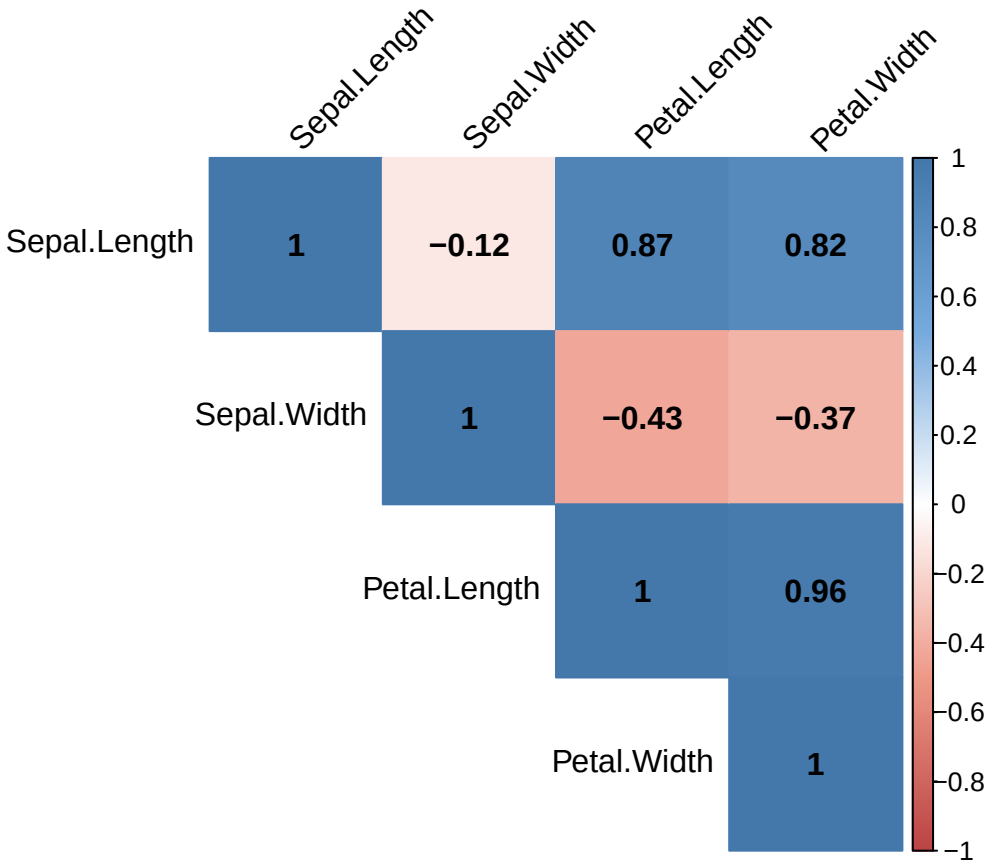
Para os dados Iris, a variável dependente dos modelos é a factor **Species**, que contém 3 categorias: setosa, versicolor e virginica. A análise de componente principal (ACP), assim como análise fatorial (AF) e Análise de Componentes Independentes (ACI) é um método que tem o objetivo de reduzir a dimensionalidade de observações multivariadas com base em sua estrutura de dependência.

Nesse sentido, a primeira coisa a se fazer durante a aplicação do PCA é observar a correlação linear entre as variáveis explicativas do modelo que buscamos implementar:

```r
# criando a correlação entre as variáveis
correlacao <- cor(iris[,1:4], method = "pearson")

# paleta de cores pasteis para usar no gráfico de correlação
col <- colorRampPalette(c("#BB4444", "#EE9988", "#FFFFFF", "#77AADD", "#4477AA"))

# produzindo um gráfico para visualização
corrplot(correlacao, method = "color",
         type = "upper", col = col(200),
         addCoef.col = "black",
         tl.col="black", tl.srt=45)
```



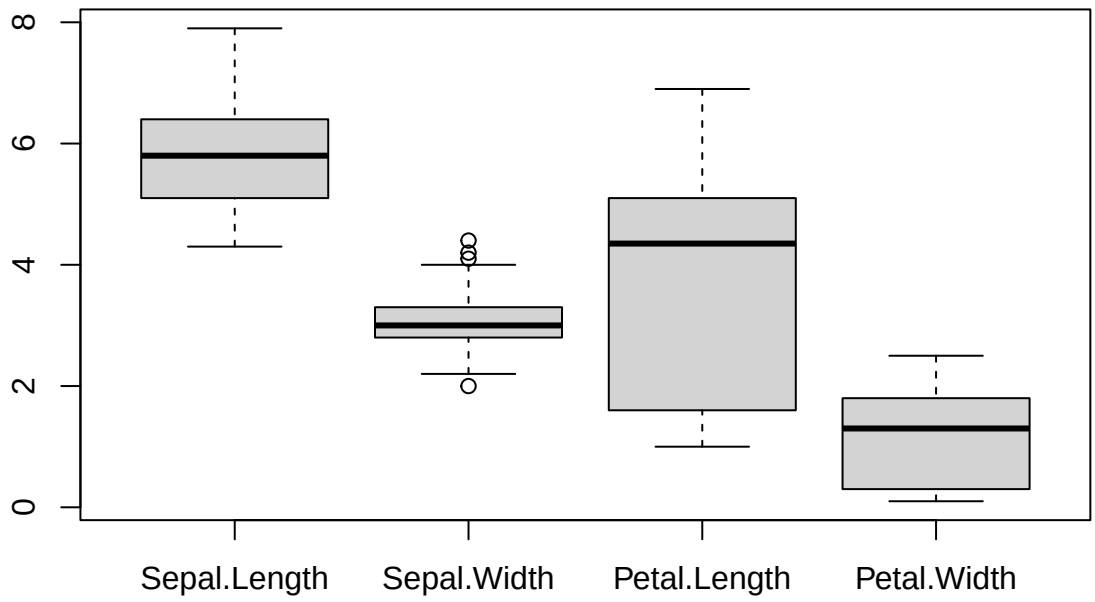

```
# observações:

# 1. method = color determina o formato do gráfico, para ser quadrados coloridos;
# 2. type = upper determina que só deve aparecer a parte superior do correlograma;
# 3. col(200) determina que o espectro de cores entre 1 e -1 tenha 200 bandas;
# 4. add.coef.col faz com que o valor da correlação seja reportado
#    junto com as cores com, além de determinar a cor do número;
# 5. tl.col r tl.srt determinam, respectivamente,
#    a cor e a inclinação do nome dos vetores.
```

De acordo com os resultados do correlograma, há uma correlação forte entre Sepal Lenght com Petal Lenght e Petal Width, assim como Petal Lenght e Petal Width. O único vetor que parece ter um comportamento significantemente distinto dos demais do ponto de vista linear é Sepal Width. No caso dessa variável, os índices de correlação são negativos com as demais e ela possuí uma correlação mais fraca.

Antes de realizar as estimativas, é preciso avaliar a dispersão dos dados para saber se é necessário padronizá-los para facilitar a interpretação dos componentes principais. Nesse caso, vamos criar um boxplot para analisar os dados

```
# criando o boxplot
boxplot(iris[,-5])
```



Embora a dispersão dos dados não seja tão grande, há uma diferença significativa na distribuição entre Sepal Lenght e Petal Width. Nesse sentido, iremos padronizar os dados para facilitar a interpretação dos coeficientes principais.

Como a questão não solicita que os dados sejam separados em diferentes amostras - para teste e treino, vamos encontrar os componentes principais utilizando todo o conjunto de dados iris.

```
set.seed(9845)
# Como a redução de dimensionalidade é feita apenas para as variáveis independentes,
# iremos remover a variável Species do cálculo.

# calculando os componentes principais
acp <- prcomp(iris[,-5],
              center = TRUE,
              scale. = TRUE)

# observação: as opções center e scale. servem para padronizar os dados.
# 1. Center centraliza os dados ao redor de zero
# 2. scale. torna a variância das variáveis unitária


# vejamos as estimativas dos componentes principais
acp
```

```
## Standard deviations (1, .., p=4):
## [1] 1.7083611 0.9560494 0.3830886 0.1439265
##
```

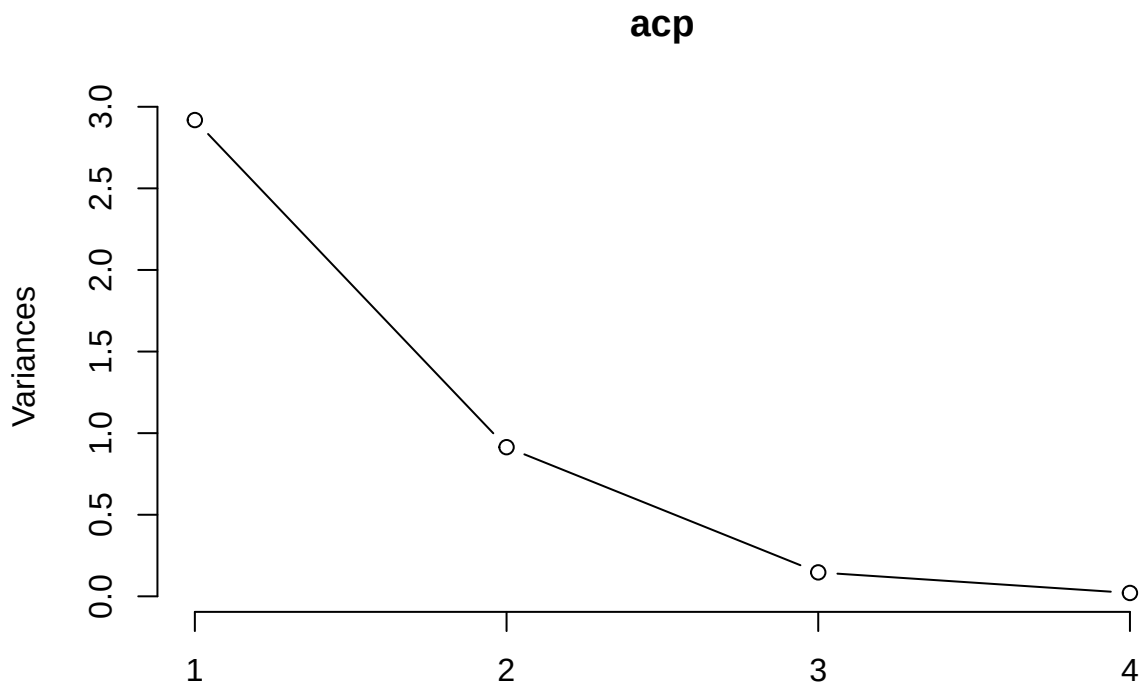
```
## Rotation (n x k) = (4 x 4):
##                     PC1         PC2        PC3        PC4
## Sepal.Length  0.5210659 -0.37741762  0.7195664  0.2612863
## Sepal.Width  -0.2693474 -0.92329566 -0.2443818 -0.1235096
## Petal.Length  0.5804131 -0.02449161 -0.1421264 -0.8014492
## Petal.Width   0.5648565 -0.06694199 -0.6342727  0.5235971
```

```
summary(acp)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.7084 0.9560 0.38309 0.14393
## Proportion of Variance 0.7296 0.2285 0.03669 0.00518
## Cumulative Proportion  0.7296 0.9581 0.99482 1.00000
```

Os resultados reportam 4 componentes principais. A primeira componente corresponde a aprox. 73% da variância total dos dados, enquanto a segunda corresponde a aproximadamente 23%. Em conjunto, os dois componentes respondem por aprox. 96% de toda a variabilidade das 4 variáveis, indicando que os demais seriam desnecessários, por explicarem uma parcela muito pequena dos dados. Vejamos o gráfico com os autovalores (variâncias) dos componentes principais:

```
screeplot(acp, type = "lines")
```

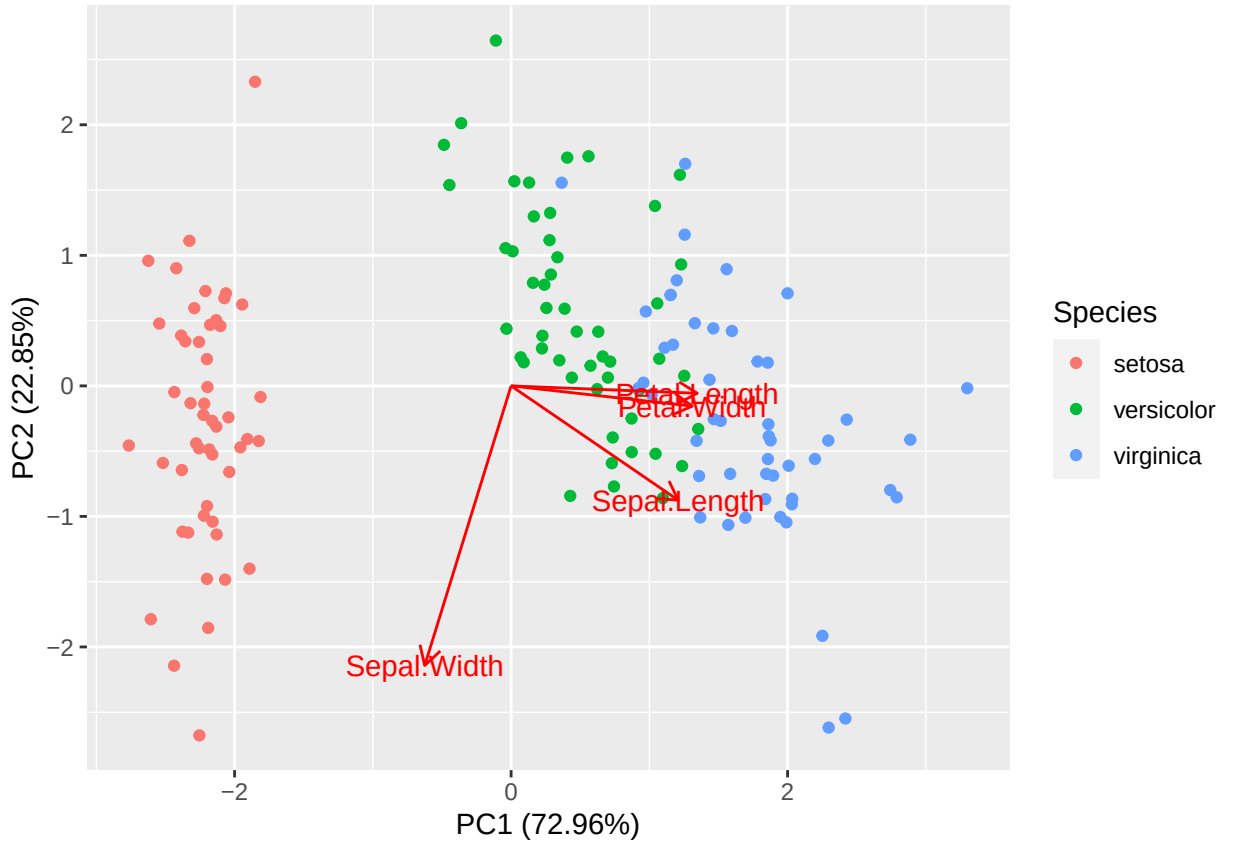


Os autovalores são obtidos através do cálculo do quadrado dos coeficientes reportados como "standard deviation", anteriormente. Já a proporção da variância explicada por cada componente principal pode ser calculada através da razão entre o autovalor do componente e o somatório dos autovalores de todos os componentes. Combinando os resultados do gráfico dos autovalores (usando o Teste Scree de Cattel (1966)) com os resultados presentes na tabela anterior, confirma-se que somente os 2 primeiros componentes

são necessários para o modelo. Vejamos o gráfico que mostra a relevância de cada variável em relação aos componentes:

```
# Para isso, usaremos o pacote ggfortify,
# que permite ao ggplot interpretar os coeficientes do ACP.
library(ggfortify)

# gráfico com os autovetores e os componentes principais.
autoplot(acp, data = iris, colour = "Species",
         loadings = TRUE, loadings.label = TRUE,
         scale = 0)
```




```
# Observações:
# 1. Loadings = TRUE determina que os autovetores devem ser reportados;
# 2. loadings.label = TRUE reporta o nome das variáveis ligadas ao vetor;
# 3. scale = 0 serve para remover a padronização dos autovetores.
```

O gráfico anterior possuí diversas características interessantes. Os valores projetados de cada vetor nos componentes principais determinam o seu nível de influência sobre aquele componente. No caso em questão, o componente principal 2 é determinado majoritariamente pelo comportamento de Sepal Width, enquanto o componente principal 1 apresenta pesos próximos para Petal Width, Lenght e Sepal Length. Além disso, o ângulo entre os vetores reportados mostra como essas variáveis são correlacionadas. Como é possível observar, Petal Lenght e Width são altamente correlacionadas e todas são pouco correlacionadas com Sepal Width. Caso houvesse um ângulo de 90° graus entre os vetores, seria indicativo de que eles não são correlacionados. O mais próximo disso é a relação entre Sepal Lenght e Sepal Width.

Por fim, é importante destacar como o valor de cada um dos componentes principais é calculado. De acordo com os coeficientes reportados, o Componente principal 1 pode ser definido da seguinte maneira:

$$CP_1 = 0,52 * Sepal.Lenght - 0,27 * Sepal.Width + 0,58 * Petal.Lenght + 0,56 * Petal.Width$$

O segundo componente segue a mesma lógica:

$$CP_2 = -0,38 * Sepal.Lenght - 0,92 * Sepal.Width - 0,02 * Petal.Lenght - 0,07 * Petal.Width$$

Como os demais vetores explicam uma parcela insignificante da variabilidade e seguem a mesma lógica, não serão reportados.

Questão 2

Realize análise fatorial para os dados do problema anterior.

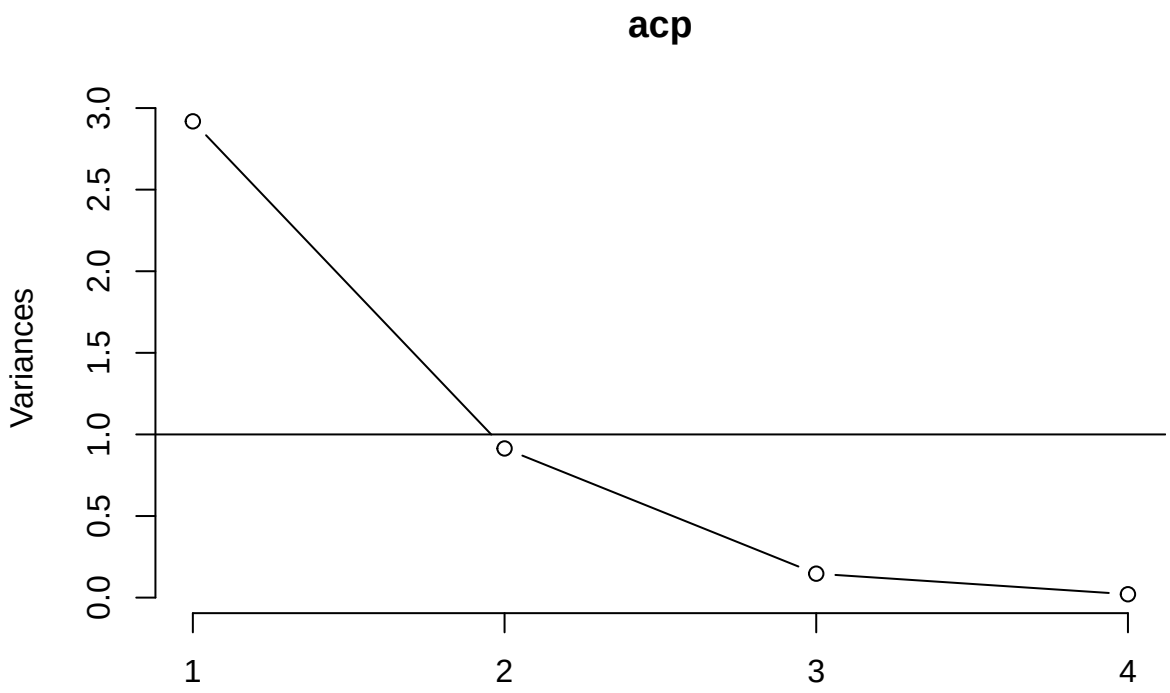
Assim como no exemplo da aula 14, para o caso da análise aplicada a **iris** não é possível realizar a análise fatorial considerando 2 fatores, pois o pacote **stats** não aceita valor superior a 1 para 4 variáveis:

```
AF <- factanal(iris[,-5], factors = 2, rotation = "varimax")
```

```
## Error in factanal(iris[, -5], factors = 2, rotation = "varimax"): 2 factors are too many for 4 variab
```

Porém, a aplicação para apenas 1 fator não é problemática, dado que a escolha do número de fatores adequada usando a Regra de Kaiser-Guttman, em que se consideram apenas os fatores com autovalores maiores que 1, indica que o número de fatores adequado é 1, divergindo da análise gráfica através do Teste Scree:

```
screeplot(acp, type = "lines")
abline(h=1)
```



Sendo assim, realizaremos as estimativas usando apenas um fator:

```
set.seed(9845)

# Análise fatorial considerando apenas 1 fator
AF <- factanal(iris[,-5], factors = 1)

# Observações:
# 1. Como há apenas um fator, não há uma matriz de cargas fatoriais, mas apenas um vetor.
#    Assim, não é possível fazer nenhum tipo de rotação de fatores para simplificar
#    a interpretação.

AF
```

```
##
## Call:
## factanal(x = iris[, -5], factors = 1)
##
## Uniquenesses:
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##        0.240        0.822        0.005        0.069
##
## Loadings:
##              Factor1
## Sepal.Length  0.872
## Sepal.Width  -0.422
## Petal.Length  0.998
```

```
## Petal.Width   0.965
##
##                Factor1
## SS loadings      2.864
## Proportion Var   0.716
##
## Test of the hypothesis that 1 factor is sufficient.
## The chi square statistic is 85.51 on 2 degrees of freedom.
## The p-value is 2.7e-19
```

Há diversas informações pertinentes a serem consideradas:

1. **Uniqueness** se refere aos ruídos do modelo. É a proporção da variabilidade de cada variável (a variância específica) que não pode ser explicada pelo único fator que criamos. Nota-se que o fator explica consideravelmente bem a variabilidade de Petal Lenght e Width, além de explicar grande parte da variabilidade de Sepal Lenght. No entanto, o fator contribui menos de 20% para a variância de Sepal Width.
2. **Loadings** se refere as cargas fatoriais. Esses valores indicam a importância do fator 1 na composição de cada uma das variáveis. Valores (em módulo) próximos de 1 indicam que o fator é muito relevante para explicar a variável. Já próximos a zero, baixa. Assim como adiantado no resultado sobre Uniqueness, as cargas fatoriais são consideravelmente elevadas para as variáveis Petal Lenght, Width e Sepal Lenght, indicando que elas são bem explicadas pelo fator 1. Já Sepal Width apresenta um valor, em módulo, consideravelmente menor que os demais, indicando que ela não é bem explicada pelo fator 1.
3. **Comunalidade**: a comunalidade de cada variável não é reportada diretamente no output, mas pode ser calculada por duas maneiras: (i) através da soma dos quadrados das cargas fatoriais de cada fator; e (ii) fazendo a conta: 1 - **Uniqueness** de cada variável. A comunalidade se refere a parcela da variância da variável que é explicada pelos fatores. No caso em questão, a comunalidade será:

```r
# cálculo de comunalidade
apply(AF$loadings^2,1,sum)
```

```
## Sepal.Length  Sepal.Width Petal.Length  Petal.Width
##    0.7597716    0.1781358    0.9950964    0.9306666
```

Como é possível perceber, o fator explica praticamente toda a variabilidade de Petal Lenght e Width e a maior parte de Sepal Lenght, mas explica apenas 18% de Sepal Width, indicando que não é apropriado para endereçar a variabilidade desta variável.

4. **SS Loadings e Proportion of Var**: essa parte da tabela indica a proporção da variabilidade das variáveis explicadas por cada fator. Como há apenas um, não há a linha que reporta a variabilidade cumulativa. Os resultados indicam que o fator 1 explica aproximadamente 72% da variabilidade das variáveis.
    - SS Loadings é a soma dos quadrados das cargas fatoriais. Pode ser obtida através da conta:

```r
sum(AF$loadings^2)
```

```
## [1] 2.86367
```

5. A última parte do output se refere a um teste de hipótese que avalia se o número de fatores no modelo é suficiente para capturar a dimensionalidade dos dados. Com o p-valor é próximo de zero, rejeitamos a hipótese nula, o que indica que o número de fatores do modelo é pequeno demais. Esse teste só é reportado porque as estimações dos parâmetros do modelo fatorial do pacote `stats` são feitas utilizando o método de máxima verossimilhança.

Podemos estimar as matrizes de covariâncias $\hat{\Sigma}$ e a residual através dos seguintes comandos:

```r
# matriz com Lambdas (cargas fatoriais)
Lambda <- AF$loadings
```

```r
# matriz de ruídos
Psi <- diag(AF$uniquenesses)

# matriz de covariâncias amostral
S <- AF$correlation

# matriz de covariâncias estimada
Sigma <- Lambda %*% t(Lambda) + Psi

# Observação: t(Lambda) transpõe a matriz de cargas fatoriais

# vejamos a matriz de covariância estimada
Sigma
```

```
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length    1.0000003  -0.3678893    0.8695090   0.8408888
## Sepal.Width    -0.3678893   1.0000011   -0.4210253  -0.4071671
## Petal.Length    0.8695090  -0.4210253    1.0000964   0.9623424
## Petal.Width     0.8408888  -0.4071671    0.9623424   1.0000000
```

```r
# matriz residual
mat_residual <- round(S - Sigma, 6)

mat_residual
```

```
##              Sepal.Length Sepal.Width Petal.Length Petal.Width
## Sepal.Length     0.000000    0.250320     0.002245   -0.022948
## Sepal.Width      0.250320   -0.000001    -0.007415    0.041041
## Petal.Length     0.002245   -0.007415    -0.000096    0.000523
## Petal.Width     -0.022948    0.041041     0.000523    0.000000
```

Como é possível observar para a matriz residual, os valores que relacionam Sepal Width e Length não são próximos de zero, indicando que o modelo fatorial precisaria de um fator adicional para contemplar esta relação. Para as demais, o modelo para ser adequado.

Há a possibilidade de usar 2 fatores através do pacote `psych`, mas como será apresentado abaixo, a depender do método utilizado para estimação dos parâmetros, eles produzem casos ultra-Heywood (quando a comunalidade excede 1). Um caso ultra-Heywood implica que um dos fatores únicos possuí uma variância negativa, que é um indicativo claro que algo está errado e as estimativas não são confiáveis. Abaixo segue um exemplo do resultado usando 2 fatores e o método de fatoração de minimização dos resíduos (default do pacote):

```r
library(psych)
```

```r
set.seed(9845)

# aplicação da análise com dois fatores usando minres
fa2_minres <- fa(iris[,-5], nfactors = 2, rotate = "varimax")
```

```
## Warning in fa.stats(r = r, f = f, phi = phi, n.obs = n.obs, np.obs = np.obs, :
## The estimated weights for the factor scores are probably incorrect.  Try a
## different factor score estimation method.
```

```
## Warning in fac(r = r, nfactors = nfactors, n.obs = n.obs, rotate = rotate, : An
## ultra-Heywood case was detected.  Examine the results carefully
```

```r
fa2_minres
```

```
## Factor Analysis using method =  minres
```

```
## Call: fa(r = iris[, -5], nfactors = 2, rotate = "varimax")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                MR1   MR2   h2     u2 com
## Sepal.Length  0.90  0.01 0.81  0.188 1.0
## Sepal.Width  -0.14  0.97 0.97  0.031 1.0
## Petal.Length  0.96 -0.29 1.01 -0.011 1.2
## Petal.Width   0.92 -0.24 0.90  0.097 1.1
##
##                        MR1  MR2
## SS loadings           2.60 1.09
## Proportion Var        0.65 0.27
## Cumulative Var        0.65 0.92
## Proportion Explained  0.70 0.30
## Cumulative Proportion 0.70 1.00
##
## Mean item complexity =  1.1
## Test of the hypothesis that 2 factors are sufficient.
##
## df null model =  6  with the objective function =  4.81 with Chi Square =  706.96
## df of  the model are -1  and the objective function was  0.11
##
## The root mean square of the residuals (RMSR) is  0.01
## The df corrected root mean square of the residuals is  NA
##
## The harmonic n.obs is  150 with the empirical chi square  0.06  with prob <  NA
## The total n.obs was  150  with Likelihood Chi Square =  15.81  with prob <  NA
##
## Tucker Lewis Index of factoring reliability =  1.145
## Fit based upon off diagonal values = 1
```

Mesmo alterando a especificação do modelo com relação a forma com que os escores e cargas fatoriais são calculados o algorítmo continua chegando a uma solução do tipo Heywood:

```
# aplicação da análise com dois fatores utilizando método de fator principal
fa2_pa <- fa(iris[,-5], nfactors = 2, rotate = "varimax", fm = "pa")
```

```
## Warning in fac(r = r, nfactors = nfactors, n.obs = n.obs, rotate = rotate, : An
## ultra-Heywood case was detected.  Examine the results carefully
```

```
fa2_pa
```

```
## Factor Analysis using method =  pa
## Call: fa(r = iris[, -5], nfactors = 2, rotate = "varimax", fm = "pa")
## Standardized loadings (pattern matrix) based upon correlation matrix
##                PA1   PA2   h2     u2 com
## Sepal.Length  0.94 -0.01 0.88  0.120 1.0
## Sepal.Width  -0.13 -0.72 0.54  0.463 1.1
## Petal.Length  0.93  0.42 1.05 -0.046 1.4
## Petal.Width   0.88  0.35 0.89  0.111 1.3
##
##                        PA1  PA2
## SS loadings           2.54 0.82
## Proportion Var        0.63 0.20
## Cumulative Var        0.63 0.84
## Proportion Explained  0.76 0.24
## Cumulative Proportion 0.76 1.00
```

```
##
## Mean item complexity =  1.2
## Test of the hypothesis that 2 factors are sufficient.
##
## df null model =  6  with the objective function =  4.81 with Chi Square =  706.96
## df of  the model are -1  and the objective function was  0
##
## The root mean square of the residuals (RMSR) is  0
## The df corrected root mean square of the residuals is  NA
##
## The harmonic n.obs is  150 with the empirical chi square  0  with prob <  NA
## The total n.obs was  150  with Likelihood Chi Square =  0.32  with prob <  NA
##
## Tucker Lewis Index of factoring reliability =  1.011
## Fit based upon off diagonal values = 1
## Measures of factor score adequacy
##                                                  PA1  PA2
## Correlation of (regression) scores with factors   0.99 0.94
## Multiple R square of scores with factors          0.98 0.89
## Minimum correlation of possible factor scores     0.96 0.79
```

Como a convergência dos resultados é muito dependente do método aplicado ao utilizar 2 fatores (e com base no resultado do teste de Kaiser-Guttman), optou-se por realizar a análise com base em apenas um fator, assim como apresentado anteriormente.

Questão 3

Obtenha as componentes independentes para os dados do Problema 1.

Questão 4

Considere o conjunto de dados *Boston* do pacote *ISLR*, contendo 506 amostras e 14 variáveis. Escolha variáveis que você acha que são importantes para descrever os dados. Faça uma análise de CP e uma análise fatorial e tente interpretar as componentes e os fatores.