Estudo do pacote **FactoMineR**
Disciplina: LCE5860 - Análise Multivariada

Docente: Dr. Afrânio vieira
Departamento: Estatística e Experimentação Agronômica - USP

Maria Letícia Salvador [mariale_salvador@usp.br]
Welinton Yoshio Hirai [wyhirai@usp.br]

2020-07-10

Contents


```
knitr::opts_chunk$set(
  dpi = 300,
  fig.retina = 1,
  fig.width = 8,
  fig.height = 6,
  cache = T
)

#pacotes utilizados
library(FactoMineR)
library(magrittr)
library(ggplot2)
library(corrplot)
library(knitr)
```

Introdução

Este relatório tem como objetivo apresentar um tutorial das funções do pacote `FactorMineR` (Husson, Josse, and Pages 2010) implementadas por linguagem R (R Core Team 2020). Ele se encontra na plataforma CRAN do R, desde Abril de 2006[1], estando atualmente na versão 2.3 publicada em 29/02/2020. Foi desenvolvido pelos autores: François Husson[2], Julie Josse[3] e Sébastien Lê[4].

O pacote tem como objetivo análises exploratórias de dados, utilizando métodos multivariadas como análise de componentes principais, métodos de agrupamentos e análise de correspondência (e múltipla).

Para efeito de aplicação utilizou-se dois conjuntos de dados o primeiro foi da tabela nutrícional dos sanduíches do MC Donalds para as técnicas de análise de componentes principais e de agrupamentos, e o banco de dados dos clientes da *Black Friday* para a análise de correspondência e análise de correspondência múltipla.

O relatório com os resultados das aplicações e algumas discussões realizadas pelo programa RStudio (R Core Team 2020), além dos conjuntos de dados que foram utilizados, foram salvos na pasta *Seminario_FactoMineR* da página pessoal do *github* para disciplina de LCE5860-6 Análise Multivariada[5].

```
#link para o github
link_github <- 'https://raw.githubusercontent.com/wyhirai/LCE5860_multivariate-analysis/master'
```

Além disto, para a melhor didática e facilidade de interpretações dos códigos e análises, foram utilizados outros pacotes além do `FactoMineR`, para uma organização no relatório as funções sempre foram especificadas com seu pacote, por exemplo `corrplot::corrplot()`.

Dados para aplicação

Hamburgueres do MC Donals

Importanto o conjunto de dados e verificando a estrutura das variáveis

```
MC_data <-
  link_github %>%
  paste('/Seminario_FactoMineR/data_sand.txt', sep = '') %>%
  url() %>%
  read.table(header = T, dec = ',')

str(MC_data)
```

```
## 'data.frame':    20 obs. of  11 variables:
##  $ Sanduiches : chr  "Big_Mac" "Big_Tasty" "Quarterao" "McNifico_Bacon" ...
##  $ valor_energ: num  502 837 528 571 468 338 404 390 345 402 ...
##  $ carboidrato: num  45 41 33 34 30 37 36 32 35 30 ...
##  $ proteina   : num  27 41 30 32 28 15 17 22 13 29 ...
##  $ g_totais   : num  25 57 31 34 27 15 22 19 17 18 ...
##  $ g_saturada : num  10 24 15 14 12 4.1 5.1 6.9 4.6 5.9 ...
##  $ g_trans    : num  0.6 1.7 1 0.9 0.9 0 0 0.4 0 0 ...
##  $ col        : num  66 85 69 70 65 27 37 55 21 75 ...
##  $ fibra_alim : num  5.4 4.6 2.6 3.8 2.4 2.2 3.4 3.2 3.5 3.1 ...
##  $ sodio      : num  1047 1345 1072 1099 819 ...
##  $ acucar     : num  8 9 9 10 5.3 5.6 6.3 7.9 6.7 7 ...
```

[1] http://factominer.free.fr/history.html
[2] https://husson.github.io/index.html
[3] http://juliejosse.com/
[4] http://sebastien.ledien.free.fr/
[5] https://github.com/wyhirai/LCE5860_multivariate-analysis

A tabela nutrícional dos sanduíches do MC Donals[6], possui 11 colunas e 20 observações. Sendo que a primeira coluna é referente aos nomes dos hambúrgueres, e tem-se 10 variáveis referente a valores nutricionais de todos os ingredientes que compõem os hambúrgueres.

Neste trabalho, utilizaremos os dados do MC Donals para o estudo das funções `PCA` e `HCPC`.

Dados da *Black Friday*

O segundo exemplo foi referente à um banco de dados público relacionada à perfis de compradores da *Black Friday*[7]. Este banco de dados contém 537.577 observações e 8 variáveis sendo elas:

- `User_ID`: identificação do usuário, contendo 5.891 usuários;
- `Product_ID`: identificação do produto contendo 3.623 produtos;
- `Gender`: gênero do usuário sendo que 1.666 indivíduos do sexo feminino e 4.225 indivíduos do sexo masculino;
- `Age`: idade dos usuários divido em 7 classes: 0-17, 18-25, 26-35, 36-45, 46-50, 51-55 e 55+;
- `Occupation`: ocupação 21 ocupações sem identificação;
- `City_Category`: categoria da cidade nomeadas como A, B e C também sem identificação;
- `Marital_Status`: estado civil com rótulo de 0 e 1 sem identificação (aparentemente 0 para solteiro e 1 para casado, pois não tem rótulo 1 para a faixa etária de 0-17);
- `Purchase`: valor de compra do produto (única variável numérica) sendo que seus valores de mínimo = 185 e máximo = 23.961, com mediana = 8.062 e média = 9.334;

```
#importanto os dados
BlackFriday_data2 <-
  link_github %>%
  paste('/Seminario_FactoMineR/Black_Friday.txt', sep = '') %>%
  url() %>%
  read.table(header = T, sep = '\t')

#convertendo para fator as variáveis caracter
BlackFriday_data <-
  BlackFriday_data2 %>%
  transform(User_ID = factor(User_ID),
            Product_ID = factor(Product_ID),
            Gender = factor(Gender),
            Age = factor(Age),
            Occupation = factor(Occupation),
            City_Category = factor(City_Category),
            Marital_Status = factor(Marital_Status),
            Stay_In_Current_City_Years = NULL)
```

Como o propósito do relatório foi o estudo do pacote (`FactoMineR`), a variável de valor de compra (`Purchase`) foi dividida em 10 classes utilizando a função `base::cut()` particionando a variável em partes pelo argumento `(breaks = ...)`. Afim de transforma-la para uma variável qualitativa (ordinal), como pode ser observado no gráfico de barras (Figura 1) dada pela contagem de observações particionadas para cada classes.

```
#particionando a variável em 10 classes
BlackFriday_data$Break_Purchase <- cut(BlackFriday_data$Purchase, breaks = 10)

#renomeando os rótulos para as classes da variável valor de compra
BlackFriday_data %<>%
  transform(Class_Purchase = factor(Break_Purchase, labels = paste('Purchase', 1:10, sep = '_')))
```

[6]https://github.com/wyhirai/LCE5860_multivariate-analysis/blob/master/first_homework/restaurante_br.pdf
[7]https://github.com/wyhirai/LCE5860_multivariate-analysis/tree/master/Seminario_FactoMineR

```r
#gráfico de barras
BlackFriday_data %>%
  ggplot(aes(x = Break_Purchase, fill = Class_Purchase)) +
  stat_count(col = 'black') +
  scale_y_continuous(n.breaks = 10) +
  coord_flip() +
  theme(legend.position = 'bottom') +
  labs(y = 'Contagem dos números de observações')
```

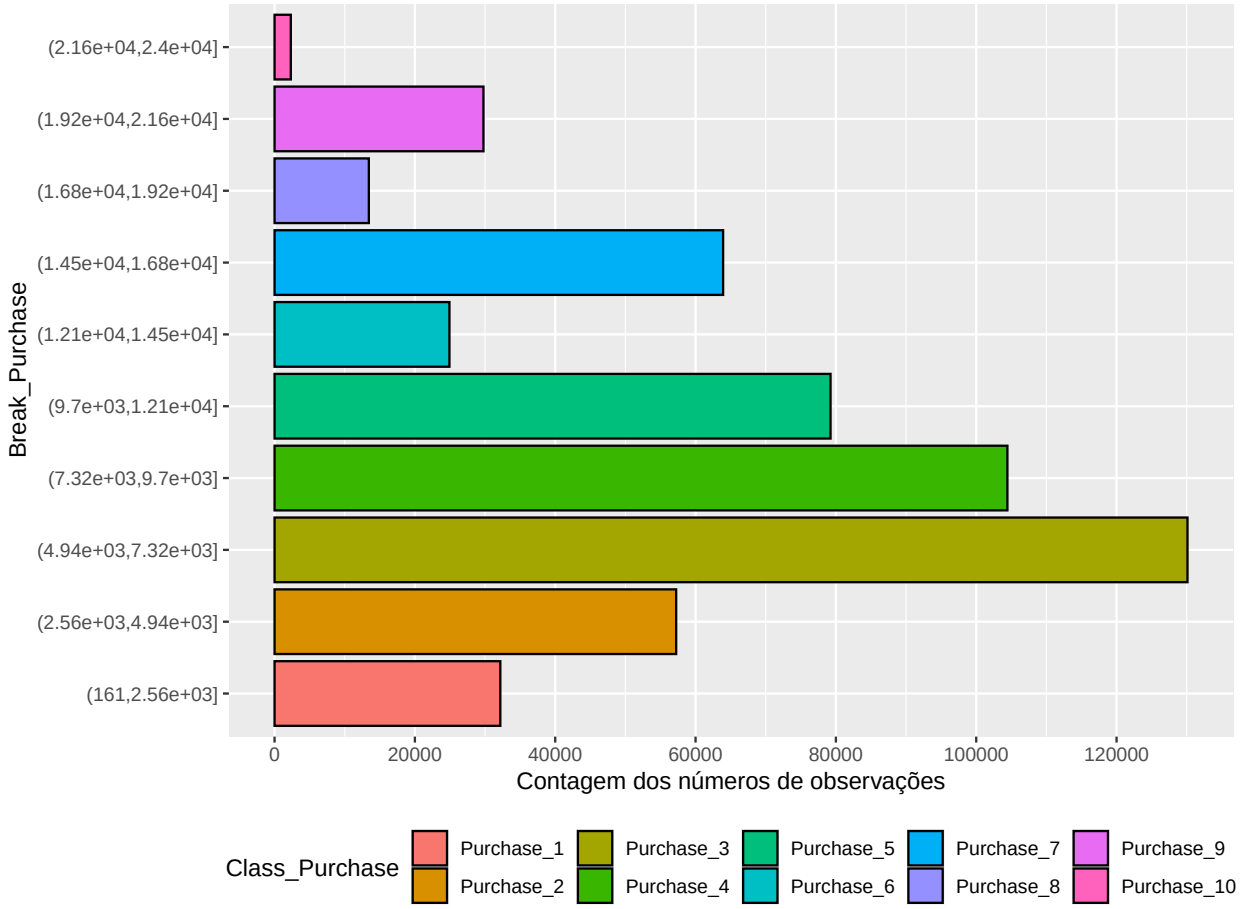


Figure 1: Gráfico de barras para contagem de observações para cada classes

Análise de Componentes Principais

A análise de componentes principais (ACP) busca explicar a estrutura de variância e covariância associada a um conjunto de variáveis através de algumas combinações lineares destas variáveis. O objetivo da ACP é encontrar uma maneira de condensar as informações contidas em várias variáveis originais em um conjunto menor de variáveis estatísticas (componentes) sem perder informações importantes.

Para o estudo de ACP utilizou-se o conjunto de dados do MC Donals, aplicando a função `FactoMineR::PCA`, conténdo os seguintes argumentos:

- `ncp =`: número de dimensões a serem consideradas nos resultados.
- `quali.sup =`: identifica as variáveis qualitativas.

- `quanti.sup =`: identifica as variáveis quantitativas.
- `scale.unit =` é um valor lógico. Se TRUE os dados são padronizados em uma mesma escala.
- `graph =`: para habilitar ou não o gráfico (*biplot*).

O código R abaixo calcula a análise de componentes principais:

```
MC_PCA <- FactoMineR::PCA(MC_data,
                          quali.sup = 1,
                          scale.unit = T,
                          graph = F)
```

A saída da função `FactoMineR::PCA` é uma lista composta pelos seguintes componentes:

```
print(MC_PCA)
```

```
## **Results for the Principal Component Analysis (PCA)**
## The analysis was performed on 20 individuals, described by 11 variables
## *The results are available in the following objects:
##
##    name                 description
## 1  "$eig"               "eigenvalues"
## 2  "$var"               "results for the variables"
## 3  "$var$coord"         "coord. for the variables"
## 4  "$var$cor"           "correlations variables - dimensions"
## 5  "$var$cos2"          "cos2 for the variables"
## 6  "$var$contrib"       "contributions of the variables"
## 7  "$ind"               "results for the individuals"
## 8  "$ind$coord"         "coord. for the individuals"
## 9  "$ind$cos2"          "cos2 for the individuals"
## 10 "$ind$contrib"       "contributions of the individuals"
## 11 "$quali.sup"         "results for the supplementary categorical variables"
## 12 "$quali.sup$coord"   "coord. for the supplementary categories"
## 13 "$quali.sup$v.test" "v-test of the supplementary categories"
## 14 "$call"              "summary statistics"
## 15 "$call$centre"       "mean of the variables"
## 16 "$call$ecart.type"   "standard error of the variables"
## 17 "$call$row.w"        "weights for the individuals"
## 18 "$call$col.w"        "weights for the variables"
```

O resultado `MC_PCA$eig`, extrai os autovalores associados aos componentes principais, a proporção da variância e a proporção acumulada.

```
MC_PCA %>%
  magrittr::extract2('eig') %>% #similar MC_PCA$eig
  knitr::kable(digits = 2) #gerando tabela automática com duas casas decimais
```

	eigenvalue	percentage of variance	cumulative percentage of variance
comp 1	6.69	66.89	66.89
comp 2	1.89	18.86	85.75
comp 3	0.72	7.18	92.93
comp 4	0.40	3.95	96.88
comp 5	0.12	1.24	98.12
comp 6	0.08	0.85	98.97
comp 7	0.06	0.61	99.59
comp 8	0.04	0.39	99.98

	eigenvalue	percentage of variance	cumulative percentage of variance
comp 9	0.00	0.02	100.00
comp 10	0.00	0.00	100.00

```
barplot(MC_PCA$eig[,2],
        main="Autovalores",
        names.arg=paste("dim",1:nrow(MC_PCA$eig)))
```

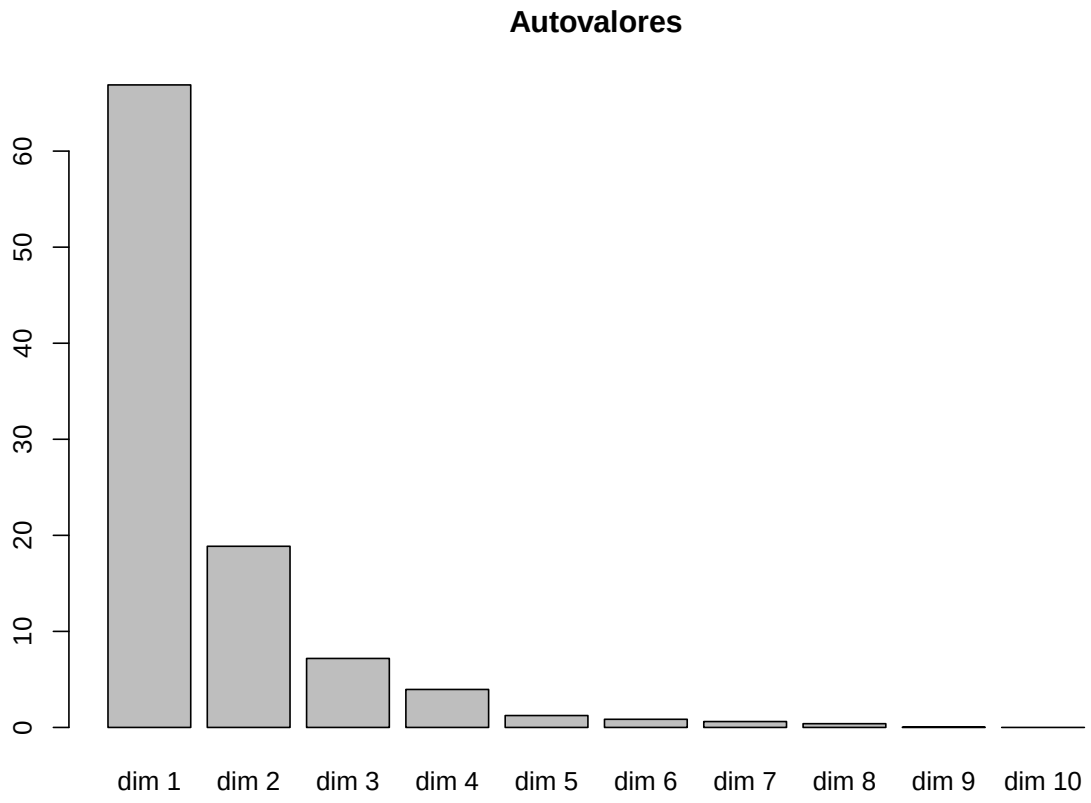


Figure 2: Autovalores associados a cada dimensão fornecida pelo ACP.

Discussão: Considerando os dos dados do MC Donals, observa-se que as duas primeiras componentes principais conjuntamente explicam 85,75 % da variância originais das variáveis. Além disto, com a Figura 2 viu-se a diferença entre os autovalores da dimensão 1 e 2 comparado-os aos outros.

Resultados sobre as observações

A função `FactoMineR::PCA` contém a lista `$ind`, que fornece uma sublista contendo todos os resultados para as observações:

- `$coord`: as coordenadas para as observações;
- `$cos2`: cosseno ao quadrado para as observações, utilizado para obter uma ideia da qualidade das projeções das observações para os componentes;

- `$contrib`: contribuição das observações (para saber o quanto uma observação contribui para a construção de um determinado componente);

Os diferentes argumentos podem ser acessados da seguinte forma:

```
MC_PCA %>%
  magrittr::extract2('ind') %>%
  magrittr::extract2('coord') %>%
  head() %>%
  knitr::kable(digits = 2)
```

Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
0.29	1.54	-0.70	1.56	-0.07
3.60	-0.80	-1.58	0.89	0.26
0.22	-0.89	-0.32	-0.58	-0.08
0.74	-0.16	-0.61	0.22	-0.18
-0.92	-1.29	-0.58	-0.32	-0.13
-2.96	0.26	-0.04	-0.53	0.60

```
MC_PCA %>%
  magrittr::extract2('ind') %>%
  magrittr::extract2('cos2') %>%
  head() %>%
  knitr::kable(digits = 2)
```

Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
0.01	0.40	0.08	0.42	0.00
0.76	0.04	0.15	0.05	0.00
0.03	0.55	0.07	0.24	0.00
0.42	0.02	0.28	0.04	0.02
0.28	0.54	0.11	0.03	0.01
0.90	0.01	0.00	0.03	0.04

```
MC_PCA %>%
  magrittr::extract2('ind') %>%
  magrittr::extract2('contrib') %>%
  head() %>%
  knitr::kable(digits = 2)
```

Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
0.06	6.26	3.40	30.94	0.21
9.70	1.70	17.47	9.96	2.74
0.04	2.09	0.73	4.32	0.26
0.41	0.06	2.58	0.63	1.26
0.63	4.39	2.32	1.31	0.69
6.55	0.18	0.01	3.56	14.34

Resultados sobre as variáveis

Por meio da função `FactoMineR::PCA`, pode se extrair uma lista de matrizes contendo todos os resultados para as variáveis com a indexação `$var`, sendo eles:

- `$coord`: as coordenadas para as variáveis;
- `$cor`: correlação entre as variáveis e as dimensões;
- `$cos2`: cosseno ao quadrado para as variáveis;
- `$contrib`: contribuição das variáveis;

Esses componentes da indexação `$var` podem ser usados no gráfico da seguinte maneira:

- `coord`: as coordenadas para variáveis para criar o gráfico de dispersão;
- `cos2`: mostra a qualidade da representação das variáveis no mapa de fatores;
- `contrib`: contém as contribuições (em porcentagem) das variáveis para os componentes principais. A contribuição de uma variável para um determinado componente principal é (em porcentagem): (var.cos2 * 100) / (cos2 total do componente);

Os diferentes componentes podem ser acessados da seguinte forma:

```
MC_PCA %>%
  purrr::pluck('var') %>%
  purrr::pluck('coord') %>%
  knitr::kable(digits = 2)
```

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
valor_energ	0.99	-0.09	-0.07	0.01	0.06
carboidrato	0.67	0.68	0.20	-0.04	0.16
proteina	0.95	-0.14	-0.06	-0.17	-0.13
g_totais	0.92	-0.30	-0.16	0.10	0.10
g_saturada	0.93	-0.35	-0.08	0.00	0.05
g_trans	0.82	-0.45	-0.31	-0.07	0.01
col	0.48	-0.55	0.61	0.30	-0.01
fibra_alim	0.55	0.64	-0.28	0.45	-0.05
sodio	0.89	0.32	0.16	-0.07	-0.24
acucar	0.82	0.43	0.24	-0.23	0.07

```
MC_PCA %>%
  purrr::pluck('var') %>%
  purrr::pluck('cor') %>%
  knitr::kable(digits = 2)
```

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
valor_energ	0.99	-0.09	-0.07	0.01	0.06
carboidrato	0.67	0.68	0.20	-0.04	0.16
proteina	0.95	-0.14	-0.06	-0.17	-0.13
g_totais	0.92	-0.30	-0.16	0.10	0.10
g_saturada	0.93	-0.35	-0.08	0.00	0.05
g_trans	0.82	-0.45	-0.31	-0.07	0.01
col	0.48	-0.55	0.61	0.30	-0.01
fibra_alim	0.55	0.64	-0.28	0.45	-0.05
sodio	0.89	0.32	0.16	-0.07	-0.24

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
acucar	0.82	0.43	0.24	-0.23	0.07

```
MC_PCA %>%
  purrr::pluck('var') %>%
  purrr::pluck('cos2') %>%
  knitr::kable(digits = 2)
```

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
valor_energ	0.98	0.01	0.01	0.00	0.00
carboidrato	0.45	0.46	0.04	0.00	0.03
proteina	0.90	0.02	0.00	0.03	0.02
g_totais	0.85	0.09	0.02	0.01	0.01
g_saturada	0.87	0.12	0.01	0.00	0.00
g_trans	0.67	0.20	0.10	0.01	0.00
col	0.23	0.30	0.38	0.09	0.00
fibra_alim	0.30	0.40	0.08	0.20	0.00
sodio	0.78	0.10	0.03	0.00	0.06
acucar	0.67	0.18	0.06	0.05	0.00

```
MC_PCA %>%
  purrr::pluck('var') %>%
  purrr::pluck('contrib') %>%
  knitr::kable(digits = 2)
```

	Dim.1	Dim.2	Dim.3	Dim.4	Dim.5
valor_energ	14.58	0.42	0.76	0.03	2.65
carboidrato	6.76	24.21	5.38	0.41	21.66
proteina	13.45	0.99	0.57	7.18	13.12
g_totais	12.69	4.65	3.47	2.46	8.00
g_saturada	12.94	6.37	0.96	0.00	1.64
g_trans	9.95	10.59	13.50	1.36	0.06
col	3.38	16.15	52.41	22.77	0.05
fibra_alim	4.53	21.44	11.15	51.35	1.92
sodio	11.73	5.47	3.78	1.09	46.91
acucar	9.99	9.72	8.02	13.35	3.99

Qualidade de representação das variáveis

A qualidade de representação das variáveis no mapa de fatores é dada pelo `$cos2`. E pelo pacote `corrplot` (Wei and Simko 2017) pode-se visualizar o `cos2` das variáveis em todas a dimensões (Kassambara 2017b, 2017a).

```
MC_PCA %>%
  magrittr::extract2('var') %>%
  magrittr::extract2('cos2') %>%
  t() %>%
  corrplot::corrplot(corr = .,
                     is.corr = F, #especificar que não é uma matrix de correlação
                     tl.col = 'black') #trocar a cor dos rótulos x e y
```

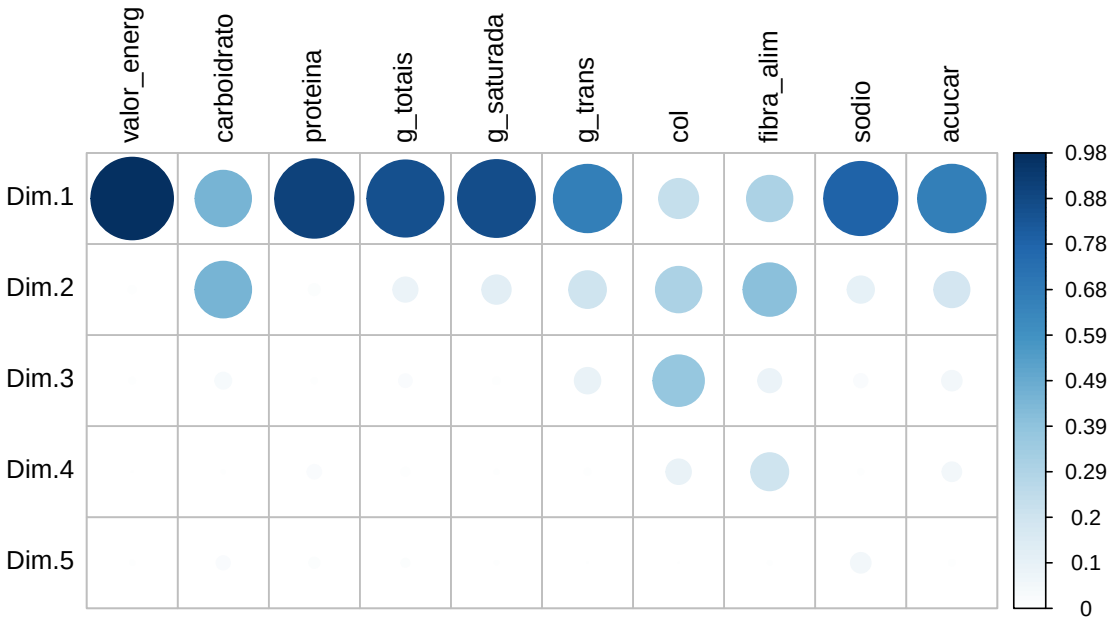


Figure 3: Gráfico para verificar a qualidade de representação das variáveis.

Note que, quando o `$cos2` possui um valor alto tem-se uma boa qualidade de representação da variável no componente principal, ou seja, nesse caso, a variável está posicionada próxima à circunferência do círculo de correlação. E quando o `$cos2` possui um valor baixo, isso indica que que a variável não é perfeitamente representada pelos componentes principais, isto é, nesse caso, a variável está próxima do centro do círculo.

Em resumo, pode-se dizer que:

- Os valores de `$cos2` são usados para estimar a qualidade da representação da variável no componente principal.
- Quanto mais próxima a variável estiver do círculo de correlações, melhor sua representação no mapa de fatores.

Discussão: Observe que, na dimensão 1 as variáveis que apresentam boa qualidade de representação são, valor energético, proteína, gorduras trans, gorduras saturadas, gorduras totais, sódio e o açúcar. E na dimensão 2 são, o carboidrato e a fibra alimentar.

Contribuição de variáveis para componentes principais

Como já visto, as contribuições das variáveis na contabilização da variabilidade em um determinado componente principal são expressas em porcentagem (Kassambara 2017b, 2017a).

- Variáveis correlacionadas com a primeira e a segunda componente principal são as mais importantes na explicação da variabilidade no conjunto de dados.
- Variáveis que não se correlacionam com nenhum componente principal ou com as últimas dimensões são variáveis com baixa contribuição e podem ser removidas para simplificar a análise geral.

Neste caso, também é possível usar a função `corrplot::corrplot` para destacar as variáveis que mais contribuem para cada dimensão (Kassambara 2017b, 2017a):

```
MC_PCA %>%
  magrittr::extract2('var') %>%
  magrittr::extract2('contrib') %>%
  t() %>%
  corrplot::corrplot(corr = .,
                     is.corr = F, #especificar que não é uma matrix de correlação
                     tl.col = 'black') #trocar a cor dos rótulos x e y
```

Discussão: Observe que, as variáveis valor energético carboidrato e fibra alimentar contribuem mais para as dimensões 1 e 2.

Análise Gráfica

Aplicando a função `FactoMineR::plot.PCA` para o objeto `MC_PCA`, pode-se escolher a partir do argumento `choix =` gerar o gráfico das variáveis (`$var`) projetadas nas 2 primeiras componentes principais, ou as observações (`$ind`). E nestes gráficos foi informado a proporção explicada das componentes principais em porcentagem.

```
plot1 <- plot(MC_PCA, choix = 'var')
plot2 <- plot(MC_PCA, choix = 'ind')
ggpubr::ggarrange(plotlist = list(plot1, plot2), labels = c('I', 'II'))
```

Discussão: A Figura 5 apresenta as variáveis projetadas em um plano bidimensional geradas pelas duas primeiras componentes principais. Observa-se que, o sentido dos vetores sugerem que a variáveis carboidrato tem correlação muito próxima de zero com a variável colesterol. E valor energético e proteína são variáveis muito correlacionadas, pois os vetores tem ângulos muito próximos de zero.

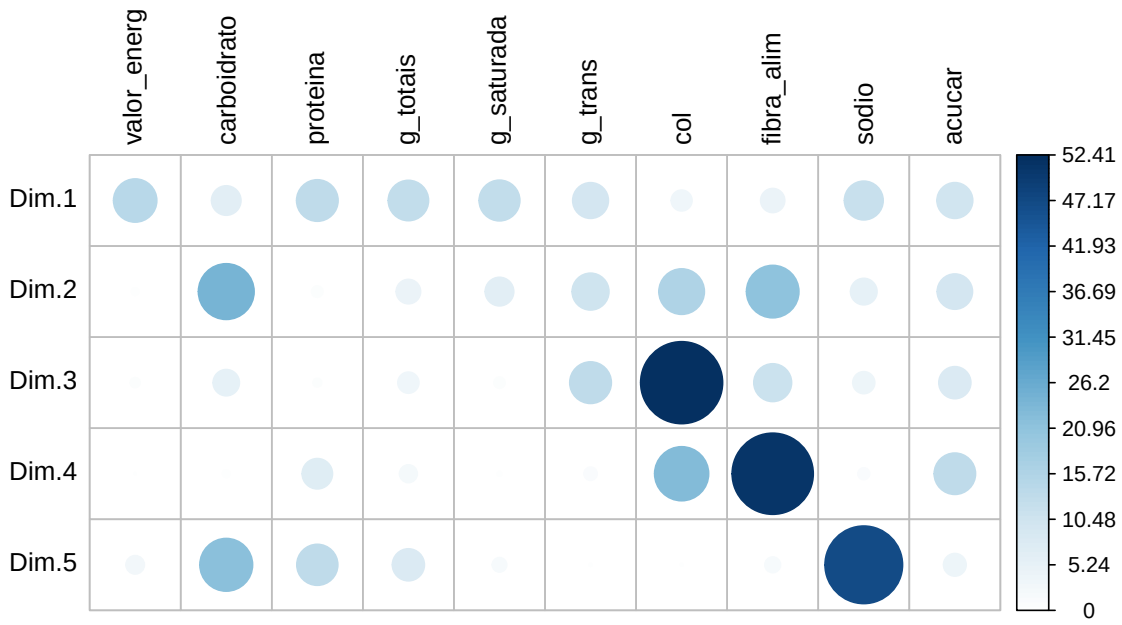


Figure 4: Gráfico para verificar quais variáveis mais contribuem para cada dimensão

I

II

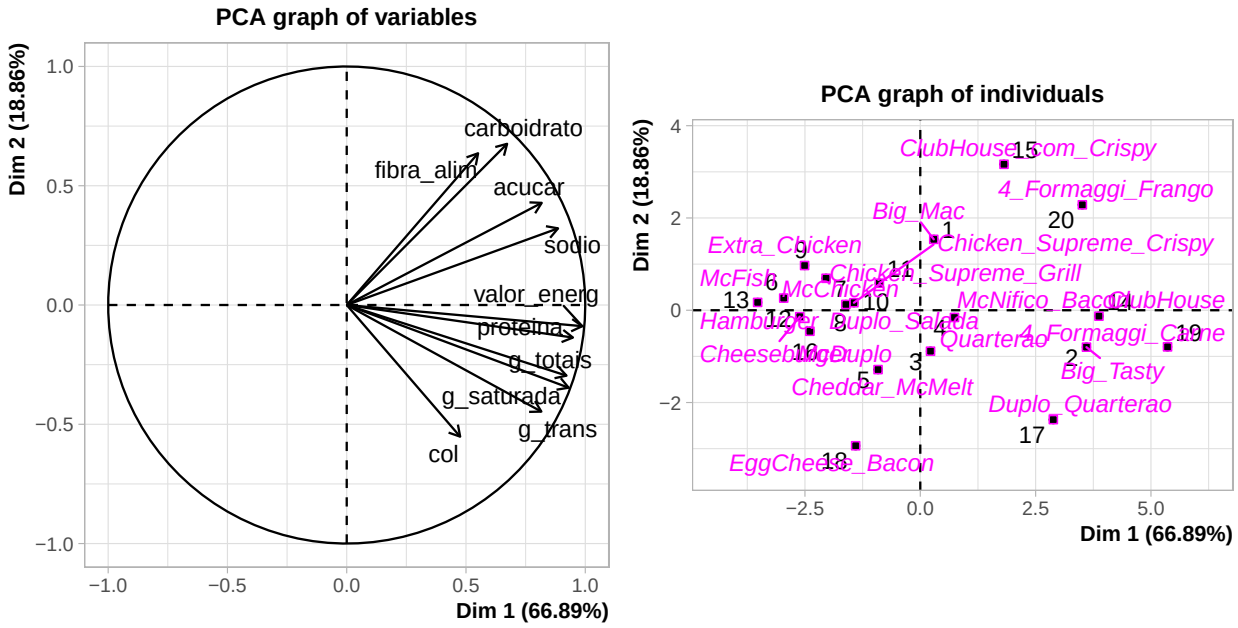


Figure 5: Projeção das variáveis na 1ª e 2ª componentes principal (I) Projeção dos indivíduos na 1ª e 2ª componentes principal (II)

Análise de agrupamento

A análise de agrupamento tem como objetivo identificar grupos com objetos semelhante em um conjunto de dados. As duas estratégias mais comuns são:

- Cluster hierárquico: identifica grupo de observações semelhantes.
- Cluster não hierárquico: divide um conjunto de dados em vários grupos, o mais utilizado é o algoritmo de k-means.

De acordo com (Husson, Josse, and Pages 2010), a abordagem HCPC (*Hierarchical Clustering on Principal Components*) combina três métodos usados na análise de dados multivariados: métodos de componentes principais, cluster hierárquico e cluster não hierárquico.

Neste trabalho, apresenta-se como função `FactoMineR::HCPC` pode ser usada para calcular o cluster hierárquico nos componentes principais. Esta função contém os seguintes argumentos:

- `nc.clust =`: é um número inteiro que especifica o número de grupos. Se 0, a árvore é cortada no número em que o indivíduo clica, se -1 a árvore é cortada automaticamente no nível sugerido e se é um número inteiro positivo a árvore é cortada com clusters nb.clusters;
- `min =` e `max =`: o número mínimo e máximo de clusters a serem exibidos;
- `graph =` se TRUE os gráficos são exibidos;
- `method =`: temos os seguintes métodos, `ward`, `average`, `complete` e o `single`. Em que o `ward` é o método padrão;

Para o estudo da função `HCPC`, considerou-se novamente o conjunto de dados do MC Donals[8].

Aplicando a função `plot` para o objeto `MC_data`, pode-se gerar o gráfico de agrupamento para as observações. E com o argumento `choice` pode-se escolher o tipo de gráfico a ser projetados, em que:

- `tree`: apresentar o gráfico de árvore;
- `map`: apresentar um mapa de fatores;
- `3D.map`: apresentar o mesmo mapa de fatores com as observações coloridas por cluster e a árvore acima;

Primeiramente, realizou-se a análise de agrupamentos considerando os dados do MC Donals sem levar nenhum método em consideração. Então, aplicou-se a função `HCPC` no conjunto de dados em estudo, considerando o método Ward, em seguida apresentou-se uma análise gráfica com os diferentes tipos de gráficos apresentados pela função.

```
row.names(MC_data) <- MC_data$Sanduiches

MC_HCPC <- HCPC(MC_data[,-1], nb.clust = -1, graph = F, method = 'ward')
```

```
plot(MC_HCPC, choice = 'tree')
```

```
plot(MC_HCPC, choice = 'map', draw.tree = F)
```

```
plot(MC_HCPC, choice = '3D.map')
```

Discussão: Os resultados nos indicam 3 clusters.

Inicialmente computou-se novamente a análise de componentes principais usando a função `FactoMineR::PCA`, em que o argumento `ncp = 2` indica que deve-se considerar apenas as duas primeiras componentes principais. Em seguida, a função `FactoMineR::HCPC` é aplicada no resultado do ACP.

[8] https://github.com/wyhirai/LCE5860_multivariate-analysis/blob/master/first_homework/restaurante_br.pdf

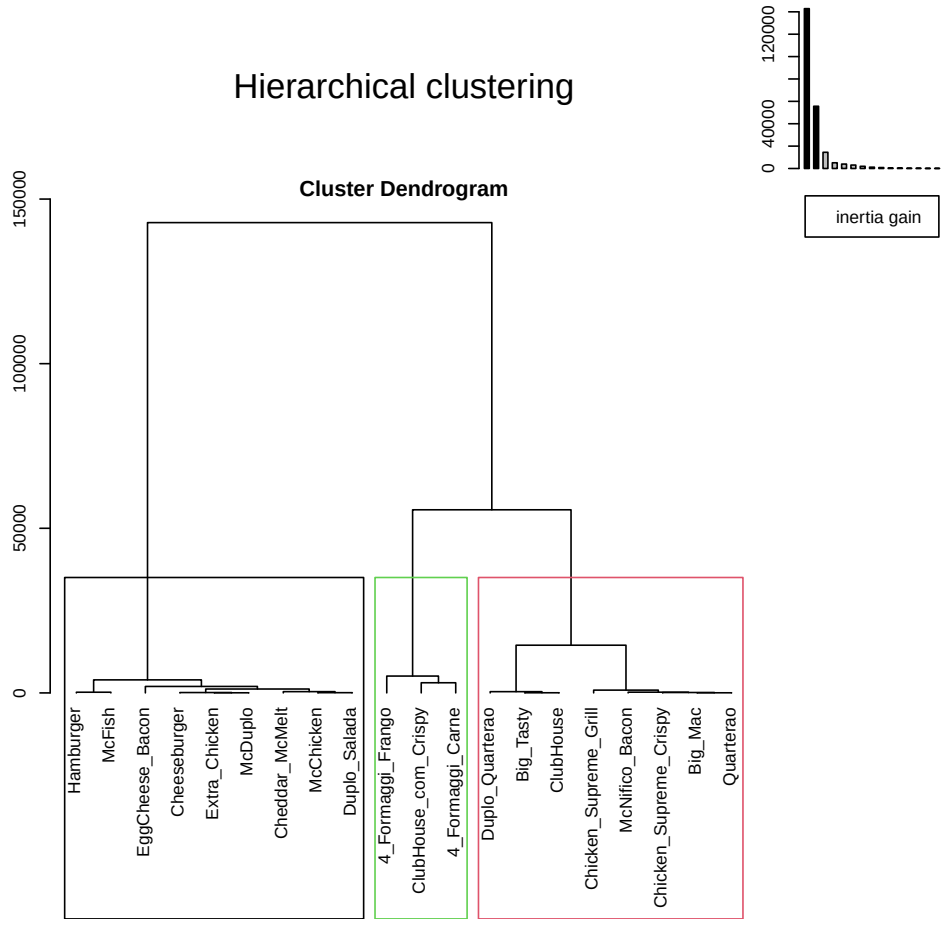


Figure 6: Dendograma, para os dados do MC Donals

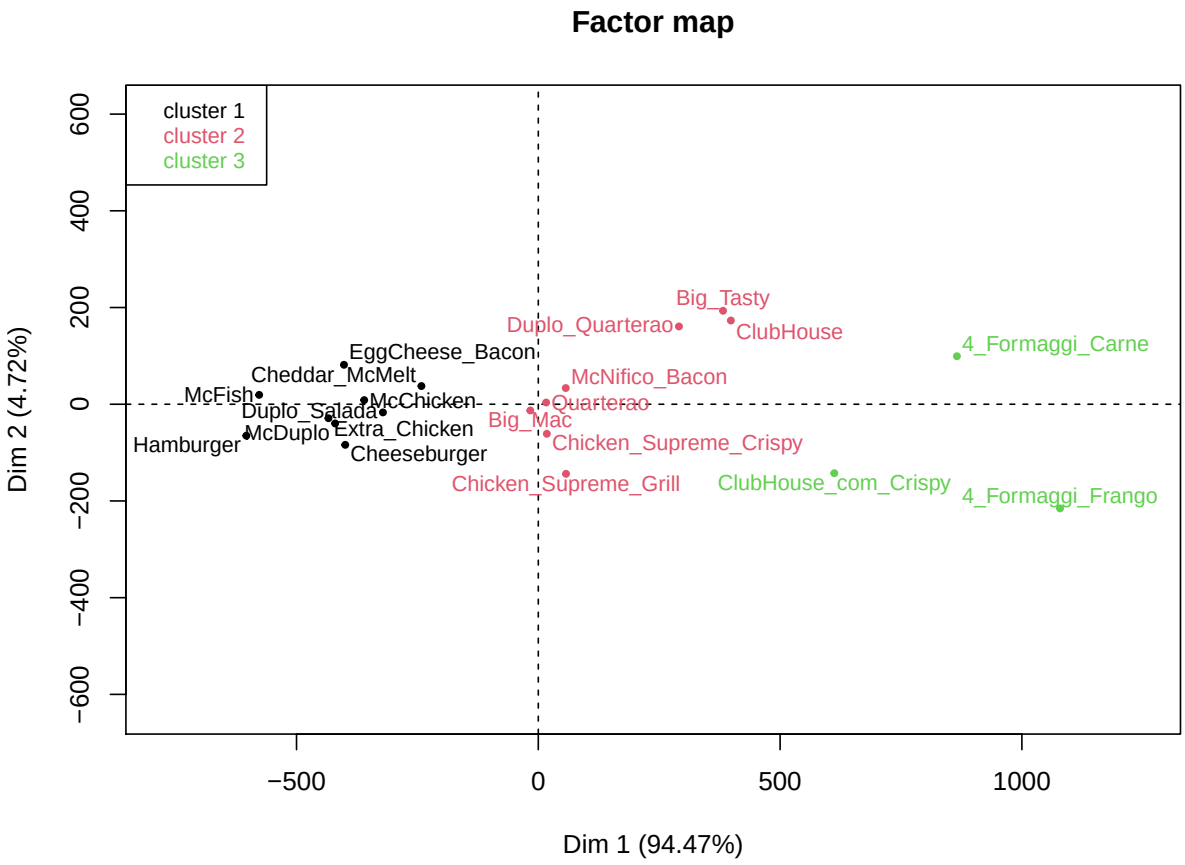


Figure 7: Mapa de fatores, para os dados do MC Donals

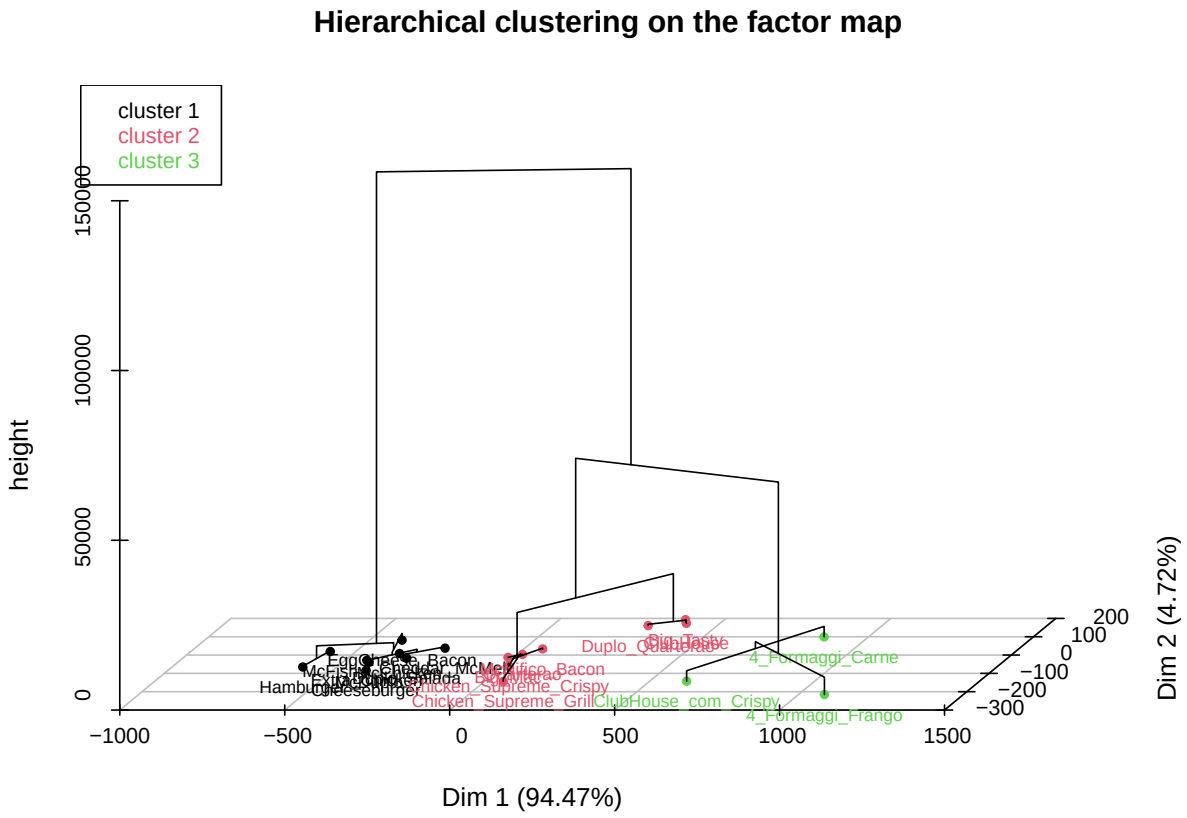


Figure 8: Mapa de fatores em 3D

```
# Cálculo da ACP com ncp=2
 MC_PCA2 <- FactoMineR::PCA(MC_data,
                            quali.sup = 1,
                            scale.unit = T, #padronizando as variáveis
                            graph = F)

# # Cálculo do agrupamento hierárquico em componentes principais
 MC_HCPC <- HCPC(MC_PCA2,
                 nb.clust = -1,
                 graph = F)
```

A função `FactoMineR::HCPC` tem as seguintes indexações:

- `data.clust`: extrai os dados originais com uma coluna suplementar que contém as atribuições de cluster;
- `desc.var`: que exibe as variáveis que descrevem cada cluster;
- `desc.ind`: mostra os indivíduos mais representativos de cada cluster;
- `desc.axe`: mostra as principais dimensões mais associadas a cada clusters;

Para exibir os dados originais com as atribuições de cluster, utiliza-se o seguinte comando:

```
MC_HCPC %>%
  magrittr::extract2('data.clust') %>%
  knitr::kable()
```

	Sanduiches	valor_energ	carboidrato	proteina	g_totais	g_saturada	g_trans	col	fibra_alim	sodio	acucares	clust
Big_Mac	Big_Mac	502	45	27	25.0	10.0	0.6	66	5.4	1047	8.0	3
Big_Tasty	Big_Tasty	837	41	41	57.0	24.0	1.7	85	4.6	1345	9.0	5
Quarterao	Quarterao	528	33	30	31.0	15.0	1.0	69	2.6	1072	9.0	3
McNifico_Bacon	McNifico_Bacon	571	34	32	34.0	14.0	0.9	70	3.8	1099	10.0	3
Cheddar_McMelt	Cheddar_McMelt	468	30	28	27.0	12.0	0.9	65	2.4	819	5.3	3
McFish	McFish	338	37	15	15.0	4.1	0.0	27	2.2	511	5.6	1
McChicken	McChicken	404	36	17	22.0	5.1	0.0	37	3.4	718	6.3	1
Duplo_Salada	Duplo_Salada	390	32	22	19.0	6.9	0.4	55	3.2	764	7.9	1
Extra_Chicken	Extra_Chicken	345	35	13	17.0	4.6	0.0	21	3.5	662	6.7	1
Chicken_Supreme_Grill	Chicken_Supreme_Grill	402	30	29	18.0	5.9	0.0	75	3.1	1164	7.0	1
Chicken_Supreme_Crispy	Chicken_Supreme_Crispy	472	37	19	27.0	9.1	0.3	48	3.6	1096	7.0	3
Cheeseburger	Cheeseburger	302	32	16	13.0	6.0	0.4	36	2.2	715	6.5	1
Hamburger	Hamburger	248	31	14	8.1	3.1	0.0	27	2.2	517	6.0	1
ClubHouse	ClubHouse	819	48	49	48.0	21.0	1.3	111	3.6	1368	16.0	5
ClubHouse_com_Crispy	ClubHouse_com_Crispy	607	60	33	26.0	10.0	0.0	64	4.4	1680	17.0	4
McDuplo	McDuplo	333	31	22	14.0	5.8	0.4	53	2.2	680	6.0	1
Duplo_Quarterao	Duplo_Quarterao	766	33	50	49.0	23.0	1.7	119	2.6	1273	9.1	5
EggCheese_Bacon	EggCheese_Bacon	417	27	17	27.0	12.0	0.3	243	1.6	660	5.1	2
4_Formaggi_Carne	4_Formaggi_Carne	911	49	55	55.0	27.0	1.7	143	3.6	1832	15.0	5
4_Formaggi_Frango	4_Formaggi_Frango	700	61	39	33.0	16.0	0.8	95	4.4	2143	16.0	4

Note que, a última coluna contém as atribuições do cluster.

Discussão: Então, por meio da saída acima, tem-se, por exemplo, que o cluster 1 contém os seguintes sanduiches: Mc Fish, McChicken, Duplo Salada, Extra Chicken , Chicken Supreme Grill, Cheeseburger, Hamburger, McDuplo.

Para se ter as variáveis que descrevem cada cluster, digita-se:

```
MC_HCPC %>%
  magrittr::extract2('desc.var') %>%
  magrittr::extract2('quanti')
```

```
## $`1`
##                v.test Mean in category Overall mean sd in category  Overall sd
## acucar      -2.280727           6.5000        8.925      0.6708204   3.7841611
## col         -2.449078          41.3750       75.450     17.0582641  49.5181532
## sodio       -2.780678         716.3750     1058.250    190.0164055 437.5703229
## proteina    -2.822454          18.5000       28.400      5.0744458  12.4835892
## g_trans     -2.837195           0.1500        0.620      0.1936492   0.5895761
## g_totais    -3.217217          15.7625       28.255      3.9679773  13.8197495
## valor_energ -3.226337         345.2500      518.000     50.2562185 190.5633753
## g_saturada  -3.298595           5.1875       11.730      1.1384611   7.0590438
##                    p.value
## acucar      0.0225646075
## col         0.0143222476
## sodio       0.0054245561
## proteina    0.0047657636
## g_trans     0.0045511845
## g_totais    0.0012944064
## valor_energ 0.0012538543
## g_saturada  0.0009716992
##
## $`2`
##       v.test Mean in category Overall mean sd in category Overall sd
## col 3.383608              243        75.45              0   49.51815
##          p.value
## col 0.0007154017
##
## $`3`
## NULL
##
## $`4`
##                v.test Mean in category Overall mean sd in category Overall sd
## carboidrato 3.438538             60.5       38.100            0.5   9.465199
## acucar      2.908497             16.5        8.925            0.5   3.784161
## sodio       2.833244           1911.5     1058.250          231.5 437.570323
##                   p.value
## carboidrato 0.000584865
## acucar      0.003631711
## sodio       0.004607819
##
## $`5`
##                v.test Mean in category Overall mean sd in category  Overall sd
## g_totais    3.784142            52.25       28.255      3.8324274  13.8197495
## g_saturada  3.711123            23.75       11.730      2.1650635   7.0590438
## g_trans     3.622705             1.60        0.620      0.1732051   0.5895761
## valor_energ 3.605475           833.25      518.000     51.9248255 190.5633753
## proteina    3.552808            48.75       28.400      5.0187150  12.4835892
## sodio       1.973641          1454.50     1058.250    220.7492922 437.5703229
##                    p.value
## g_totais    0.0001542399
```

```
## g_saturada  0.0002063415
## g_trans     0.0002915380
## valor_energ 0.0003115827
## proteina    0.0003811424
## sodio       0.0484225462
```

Discussão: As variáveis que descrevem o cluster 1 são: açúcar, colesterol, sódio, proteína, gorduras trans, gordura saturada, gorduras totais e valor energético estão significativamente associadas ao cluster 1. Por exemplo, o valor médio da variável açúcar é de 6,5 que é menor que a média geral (8,925) em todos os clusters, ou seja, o cluster 1 é caracterizado por sanduiches que possuem baixa taxa de açúcar. Pode-se concluir então que este cluster é caracterizado por sanduiches com baixa taxa de açúcar, colesterol, sódio, proteína, valor energético e também por hambúrgueres menos gordurosos O cluster 2 possui os sanduiches com alta taxa de colesterol. Já o cluster 4 os sanduiches possuem alta taxa de carboidrato, açúcar e sódio. E as variáveis que descrevem o cluster 5 são, sódio, proteína, gorduras trans, gordura saturada, gorduras totais, este cluster é caracterizado pelos sanduiches que são mais gordurosos.

Pode-se também, observar quais os eixos que descrevem os clusters, tem-se o seguinte comando:

```
MC_HCPC %>%
  magrittr::extract2('desc.axes')
```

```
##
## Link between the cluster variable and the quantitative variables
## ================================================================
##            Eta2      P-value
## Dim.1 0.9234081 3.391569e-08
## Dim.3 0.7933815 5.078975e-05
## Dim.2 0.7440663 2.394499e-04
##
## Description of each cluster by quantitative variables
## ========================================================
## $`1`
##          v.test Mean in category  Overall mean sd in category Overall sd
## Dim.1 -3.284269        -2.386662 -8.881784e-17      0.6429895   2.586327
##           p.value
## Dim.1 0.001022473
##
## $`2`
##          v.test Mean in category  Overall mean sd in category Overall sd
## Dim.3  3.076319         2.606895 -4.584007e-17              0  0.8474072
## Dim.2 -2.138488        -2.936770 -1.512679e-16              0  1.3732926
##           p.value
## Dim.3 0.002095736
## Dim.2 0.032477121
##
## $`3`
## NULL
##
## $`4`
##         v.test Mean in category  Overall mean sd in category Overall sd
## Dim.2 2.882424         2.724366 -1.512679e-16     0.43945305  1.3732926
## Dim.3 1.960930         1.143665 -4.584007e-17     0.07325806  0.8474072
##          p.value
## Dim.2 0.00394628
## Dim.3 0.04988717
##
```

```
## $`5`
##         v.test Mean in category  Overall mean sd in category Overall sd
## Dim.1 3.311085         3.929226 -8.881784e-17      0.9012594   2.586327
##              p.value
## Dim.1 0.0009293499
```

Discussão: Os resultados, indicam que os sanduiches nos clusters 1 e 5 tem coordenadas altas no primeiro eixo e os que pertencem ao cluster 2 e 4 possuem altas coordenadas no eixo dois e três.

E os sanduiches representativos de cada clusters podem ser extraídos da seguinte maneira:

```
MC_HCPC %>%
  magrittr::extract2('desc.ind')
```

```
## $para
## Cluster: 1
## Cheeseburger      McDuplo Duplo_Salada    McChicken       McFish
##    0.6478186    0.7906430    0.8290836    0.8913254    0.9080133
## ------------------------------------------------------------
## Cluster: 2
##  EggCheese_Bacon
##                0
## ------------------------------------------------------------
## Cluster: 3
##         McNifico_Bacon Chicken_Supreme_Crispy              Quarterao
##              0.8665229              1.0271177              1.2581626
##         Cheddar_McMelt                Big_Mac
##              1.5958498              2.0922539
## ------------------------------------------------------------
## Cluster: 4
## ClubHouse_com_Crispy    4_Formaggi_Frango
##            0.9984753            0.9984753
## ------------------------------------------------------------
## Cluster: 5
##        ClubHouse        Big_Tasty 4_Formaggi_Carne  Duplo_Quarterao
##         1.229733         1.624011         1.642874         1.794295
##
## $dist
## Cluster: 1
##     Hamburger        McFish  Cheeseburger Extra_Chicken       McDuplo
##      3.572882      3.095416      2.708456      2.628100      2.515255
## ------------------------------------------------------------
## Cluster: 2
## EggCheese_Bacon
##        4.510579
## ------------------------------------------------------------
## Cluster: 3
##                Big_Mac         McNifico_Bacon              Quarterao
##               3.511405               3.221319               2.872159
##         Cheddar_McMelt Chicken_Supreme_Crispy
##               2.168981               1.687564
## ------------------------------------------------------------
## Cluster: 4
## ClubHouse_com_Crispy    4_Formaggi_Frango
##             4.171048             3.672144
## ------------------------------------------------------------
```

```
## Cluster: 5
## 4_Formaggi_Carne        Big_Tasty  Duplo_Quarterao        ClubHouse
##         4.506038         4.003626         3.871407         3.305244
```

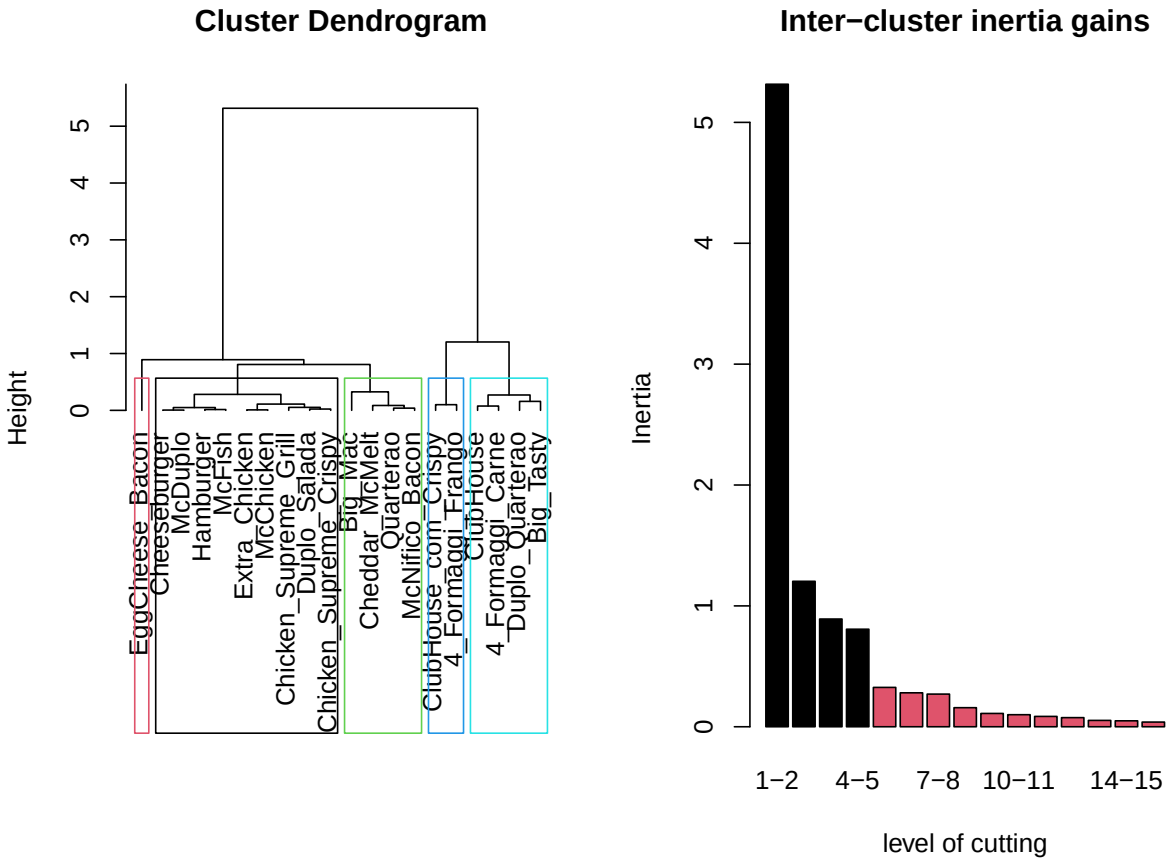
Tem-se que, comando `desc.ind$para`, indica quais são os sanduiches mais próximos do centro do cluster. E o comando `desc.ind$dist` indica os sanduiches que se encontram mais distantes do centro do cluster.

Análise Gráfica

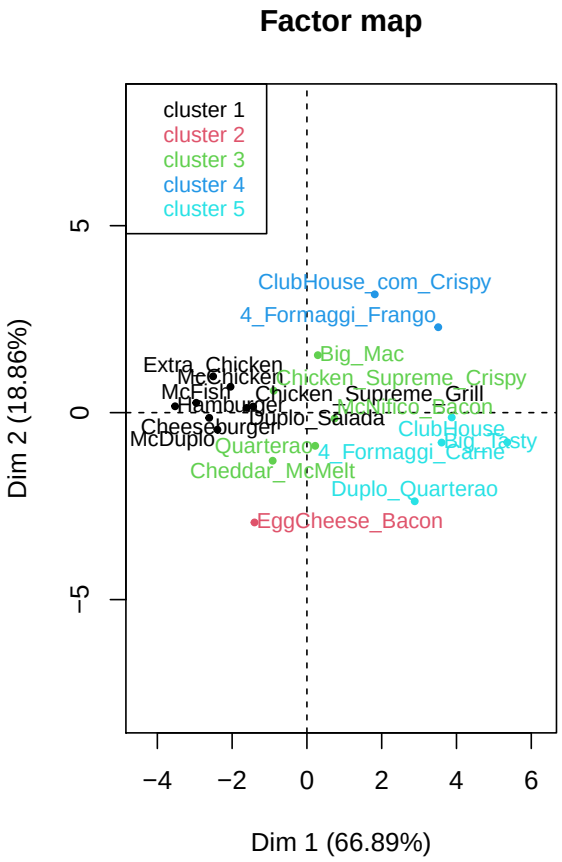
Aplicando a função `plot.HCPC` para o objeto `MC_HCPC`, pode-se escolher por meio do argumento `choice` gerar o gráfico das variáveis (`map`) projetadas nas 2 primeiras componentes principais, ou as observações (`3D.map`). E nestes gráficos pode visualizar em quais clusters se encotram as observações.

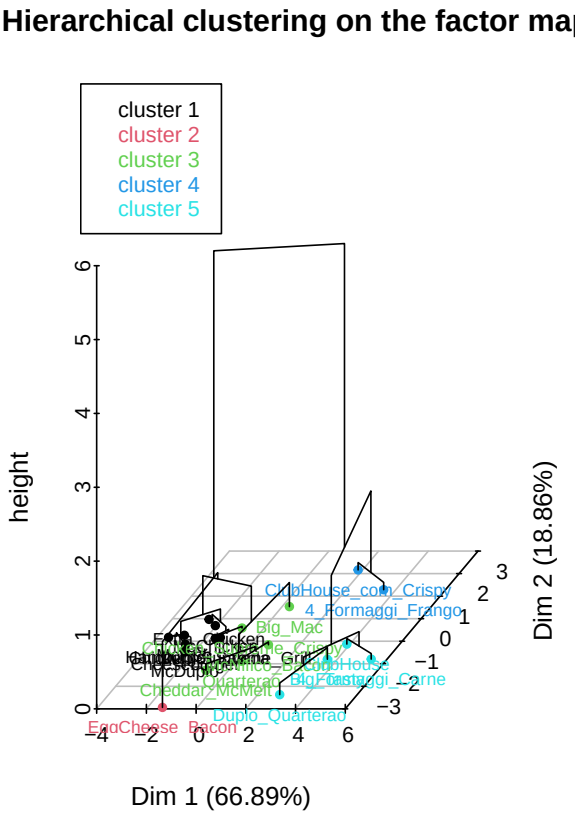
```
MC_HCPC <- HCPC(MC_PCA2,
                nb.clust=-1,
                graph = F)

par(mfrow = c(1, 2))
plot(MC_HCPC, choice = 'tree', tree.barplot = F)
plot(MC_HCPC, choice = 'bar')
```




```
par(mfrow = c(1, 2))
plot(MC_HCPC, choice = 'map', draw.tree = F)
plot(MC_HCPC, choice = '3D.map')
```





Análise de Correspondência

A Análise de Correspondência (*Correspondence Analysis* - CA) é uma técnica multivariada exploratória para análise numérica e gráfica de dados com a forma de matriz (sem valores negativos), mas é amplamente utilizadas para tabelas de frequências e contagens (Greenacre and Blasius 2006). Assim, para este método, foi utilizado os dados da *Black Friday*.

Foi utilizada a CA, afim de verificar como está caracterizada a relação de linha e coluna entre as variáveis de classes do valor de compra (`Class_Purchase`) e idade dos clientes da *Black Friday* (`Age`), como hipótese verificar se a faixa etária, tem diferentes perfis de compra. Para isto, foi utilizada uma tabela de contagens de quantas compras houveram para cada classe de compra (linha), em relação a cada idade (colunas).

```
BlackFriday_data %>%
  dplyr::select(Class_Purchase, Age) %>%
  table() %>%
  as.data.frame() %>%
  ggplot(aes(x = Age,
             y = Class_Purchase,
             fill = Freq)) +
  geom_tile(colour = 'black') +
  geom_text(aes(label = Freq),
            colour = 'black ') +
  scale_fill_gradient2(name = 'Contagem',
                       n.breaks = 10,
```

```
                    mid = 'white', high = 'darkblue')
```

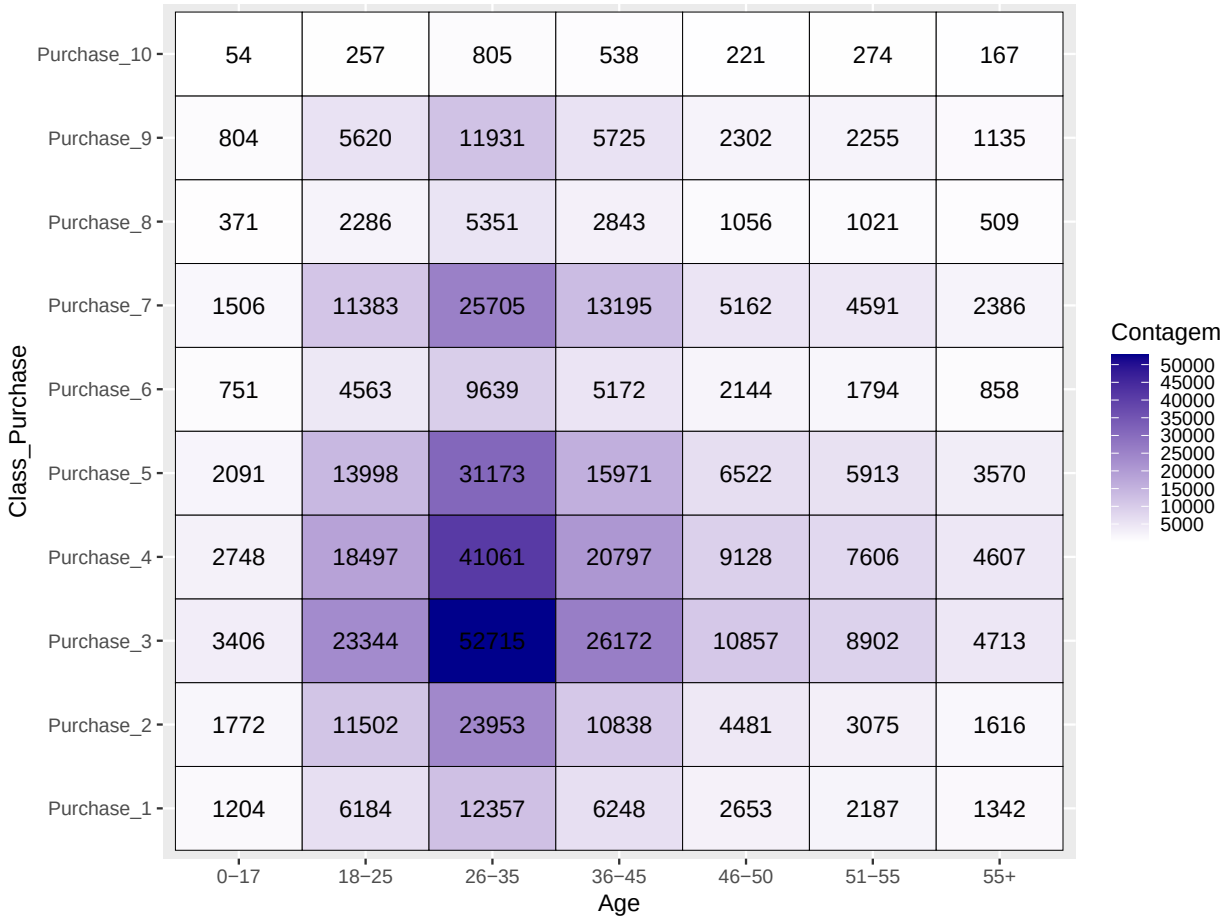


Figure 9: Gráfico de retângulos para a tabela de contagens das classes de compra, em relação as idades

Para isso o pacote `FactoMineR` (Lê, Josse, and Husson 2008) tem a função `FactoMineR::CA()`, em que o objeto de entrada para a função foi a tabela de contagens. Os principais argumentos da função (mais informações no `base::help(CA, "FactoMineR")`) são:

- `X =`: onde se insere a tabela;
- `ncp =`: nº de dimensões que os *resultados* da análise apresenta (padrão é 5);
- `graph =`: (lógico) só assume valores `TRUE` ou `FALSE`, é o argumento para apresentar o gráfico da CA (afim de didática foi apresentado mais a frente);

```
BlackFriday_CA <-
  BlackFriday_data %>%
  dplyr::select(Class_Purchase, Age) %>%
  table() %>%
  FactoMineR::CA(X = .,
                 ncp = 5,
                 graph = F)
```

O primeiro resultado que vamos observar é o próprio objeto que foi atribuída a função `FactoMiner::CA()`. Inicialmente pode-se observar que foi realizado o teste de χ^2. Além disto, exibe as demais listas que podem ser apresentados, como por exemplo `$eig`, `$col`, `$col$coord` e entre outros.

Discussão: para o nosso exemplo o teste mostrou-se um valor de probabilidade < 0, assim rejeitou-se a hipótese nula (H_0) de indepêndência entre as linhas e colunas.

```
BlackFriday_CA
```

```
## **Results of the Correspondence Analysis (CA)**
## The row variable has  10  categories; the column variable has 7 categories
## The chi square of independence between the two variables is equal to 1528.142 (p-value =  3.45974e-28
## *The results are available in the following objects:
##
##    name              description
## 1  "$eig"            "eigenvalues"
## 2  "$col"            "results for the columns"
## 3  "$col$coord"      "coord. for the columns"
## 4  "$col$cos2"       "cos2 for the columns"
## 5  "$col$contrib"    "contributions of the columns"
## 6  "$row"            "results for the rows"
## 7  "$row$coord"      "coord. for the rows"
## 8  "$row$cos2"       "cos2 for the rows"
## 9  "$row$contrib"    "contributions of the rows"
## 10 "$call"           "summary called parameters"
## 11 "$call$marge.col" "weights of the columns"
## 12 "$call$marge.row" "weights of the rows"
```

Na lista `$eig` apresentou-se os autovalores, o percentual da variância, e o percentual acumulado.

Discussão: pode-se observar que apenas 2 dimensões foi o bastante para explicar, aproximadamente, 89.12 % da variância percentual acumulada.

```
BlackFriday_CA %>%
  magrittr::extract2('eig') %>%
  knitr::kable()
```

	eigenvalue	percentage of variance	cumulative percentage of variance
dim 1	0.0020484	72.0603994	72.06040
dim 2	0.0004852	17.0703233	89.13072
dim 3	0.0001438	5.0571914	94.18791
dim 4	0.0001090	3.8328610	98.02078
dim 5	0.0000380	1.3356915	99.35647
dim 6	0.0000183	0.6435333	100.00000

Abaixo são os valores das coordenadas (`$row$coord`) dos indivíduos das linhas (classes de valores de compra) na projeção que tenha máxima inercia total contido no espaço de dimensão das colunas (idade dos clientes). Similar ao apresentado na seção de análise de componentes principais.

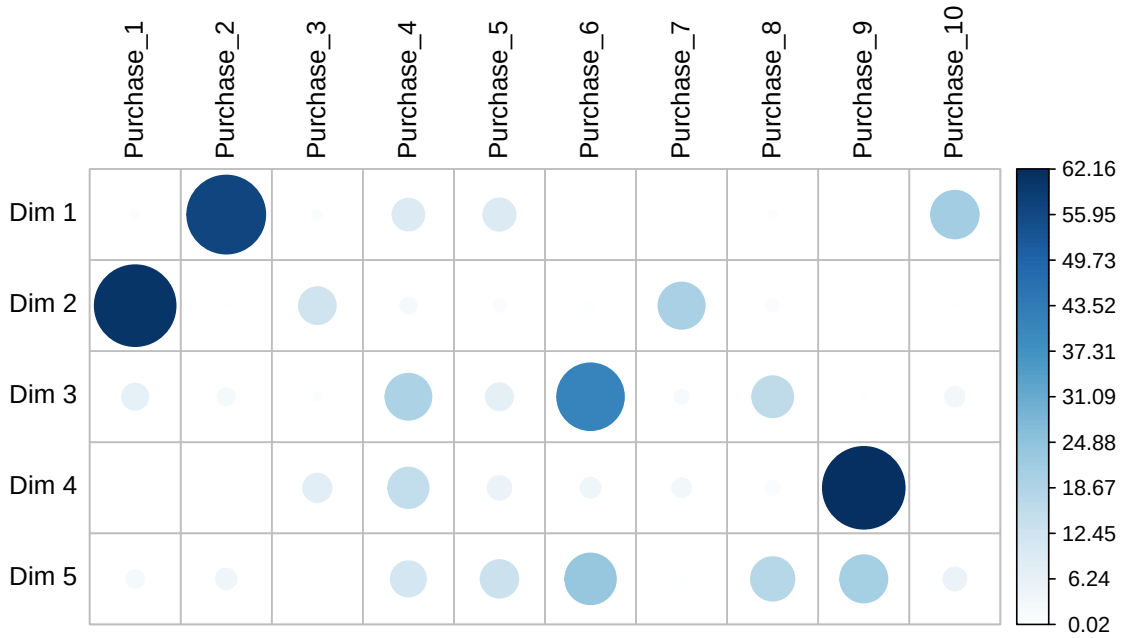
```
BlackFriday_CA %>%
  magrittr::extract2('row') %>%
  magrittr::extract2('coord') %>%
  knitr::kable(digits = 2)
```

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5
Purchase_1	-0.02	0.07	0.01	0.00	0.00
Purchase_2	-0.10	0.00	-0.01	0.00	0.00
Purchase_3	-0.01	-0.02	0.00	0.01	0.00
Purchase_4	0.03	0.01	-0.01	0.01	0.00

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5
Purchase_5	0.04	0.01	-0.01	-0.01	-0.01
Purchase_6	0.00	0.01	0.04	0.01	0.01
Purchase_7	0.01	-0.03	0.00	-0.01	0.00
Purchase_8	0.02	-0.02	0.03	-0.01	-0.02
Purchase_9	0.00	0.00	0.00	-0.03	0.01
Purchase_10	0.32	0.01	0.03	0.01	-0.02

O próximo resultado são cálculos de contribuição (`$row$contrib`) que as variáveis das linhas apresentam para cada dimensão. Afim de facilitar a visualização, como foi realizado na PCA, construiu-se um gráfico para visualização de matrizes com a função `corrplot::corrplot()`, apresentando circunferências maiores de com cores mais fortes (azul), para aqueles que apresentam valores altos.

```
BlackFriday_CA %>%
  magrittr::extract2('row') %>%
  magrittr::extract2('contrib') %>%
  t() %>%
  corrplot::corrplot(is.corr = F, tl.col = 'black', mar = c(0, 0, .5, 0))
```



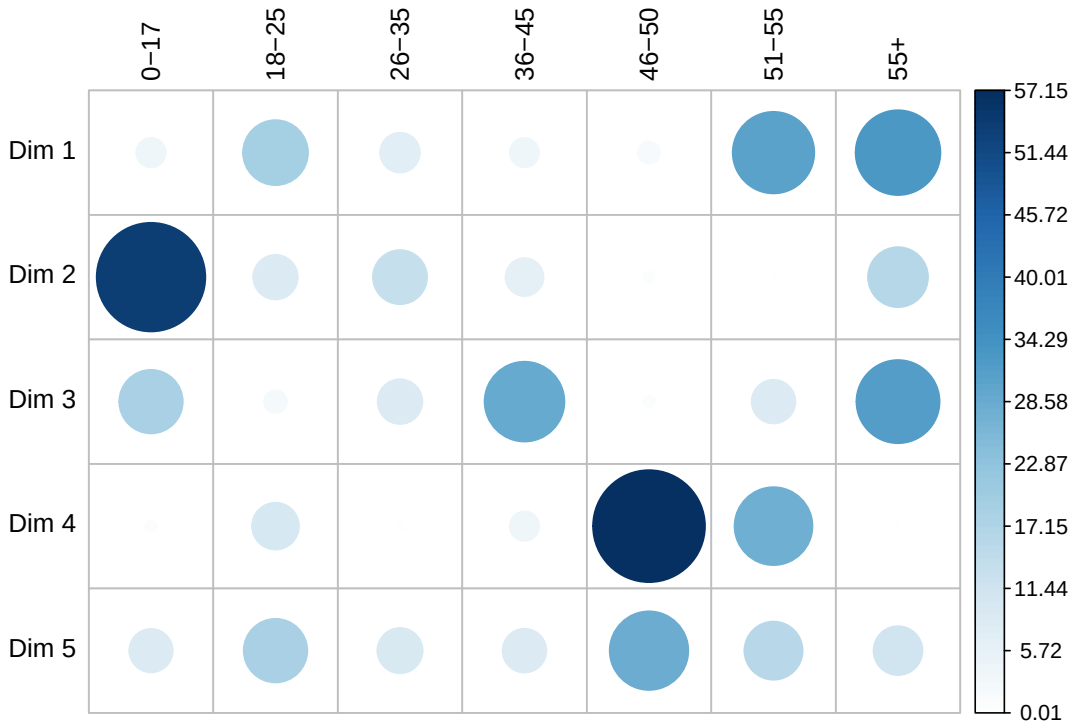
E de forma similar, tem-se os mesmo resultados de coordenadas e contribuição para as variáveis de linhas

```
BlackFriday_CA %>%
  magrittr::extract2('col') %>%
  magrittr::extract2('coord') %>%
```

```
  knitr::kable(digits = 2)
```

	Dim 1	Dim 2	Dim 3	Dim 4	Dim 5
0-17	-0.05	0.10	0.03	0.00	-0.01
18-25	-0.05	0.02	0.00	-0.01	0.01
26-35	-0.02	-0.01	-0.01	0.00	0.00
36-45	0.02	-0.01	0.01	0.00	0.00
46-50	0.02	0.01	0.00	0.03	0.01
51-55	0.09	0.00	0.01	-0.02	0.01
55+	0.13	0.05	-0.03	0.00	-0.01

```
BlackFriday_CA %>%
  magrittr::extract2('col') %>%
  magrittr::extract2('contrib') %>%
  t() %>%
  corrplot::corrplot(is.corr = F, tl.col = 'black', mar = c(0, 0, .5, 0))
```



Resumindo todas as saídas, pode-se pedir o `summary.CA()` para o objeto que foi atribuído a CA, além disto, para não deixa carregado de informações, pode-se pedir o nº de dimensões que foi avaliado como significativo para a análise utilizado o argumento `ncp =`.

```
summary(BlackFriday_CA, ncp = 2)
```

```
##
## Call:
## FactoMineR::CA(X = ., ncp = 5, graph = F)
##
## The chi square of independence between the two variables is equal to 1528.142 (p-value =  3.45974e-28
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4   Dim.5   Dim.6
## Variance               0.002   0.000   0.000   0.000   0.000   0.000
## % of var.             72.060  17.070   5.057   3.833   1.336   0.644
## Cumulative % of var.  72.060  89.131  94.188  98.021  99.356 100.000
##
## Rows
##                Iner*1000    Dim.1    ctr   cos2    Dim.2    ctr   cos2
## Purchase_1  |     0.319 | -0.015  0.683  0.044 |  0.070 60.752  0.923 |
## Purchase_2  |     1.166 | -0.104 56.600  0.994 |  0.003  0.185  0.001 |
## Purchase_3  |     0.094 | -0.009  0.936  0.203 | -0.016 12.776  0.657 |
## Purchase_4  |     0.257 |  0.032  9.529  0.759 |  0.008  2.555  0.048 |
## Purchase_5  |     0.230 |  0.037  9.786  0.871 |  0.007  1.447  0.030 |
## Purchase_6  |     0.077 | -0.003  0.024  0.006 |  0.008  0.584  0.037 |
## Purchase_7  |     0.117 |  0.008  0.389  0.068 | -0.029 20.003  0.827 |
## Purchase_8  |     0.054 |  0.025  0.739  0.280 | -0.017  1.439  0.129 |
## Purchase_9  |     0.080 | -0.004  0.043  0.011 |  0.004  0.147  0.009 |
## Purchase_10 |     0.447 |  0.318 21.271  0.975 |  0.011  0.112  0.001 |
##
## Columns
##                Iner*1000    Dim.1    ctr   cos2    Dim.2    ctr   cos2
## 0-17        |     0.376 | -0.055  4.006  0.218 |  0.098 53.829  0.695 |
## 18-25       |     0.465 | -0.047 19.319  0.850 |  0.016  9.082  0.095 |
## 26-35       |     0.235 | -0.019  7.271  0.634 | -0.013 13.374  0.276 |
## 36-45       |     0.168 |  0.020  3.963  0.484 | -0.013  6.611  0.191 |
## 46-50       |     0.123 |  0.024  2.234  0.372 |  0.006  0.589  0.023 |
## 51-55       |     0.673 |  0.094 30.381  0.925 | -0.002  0.048  0.000 |
## 55+         |     0.803 |  0.131 32.824  0.838 |  0.045 16.468  0.100 |
```

Além disto, de uma maneira mais didática e apresentável o gráfico da CA, em que são apresentadas as coordenadas da variáveis de linhas e colunas, nas dimensões escolhidas (`axes =`) além dos seus respectivos o percentual de variância.

Discussão: Com o gráfico abaixo, podemos observar como foi caracterizada as variáveis de linhas e coluna na dimensão 1. Em relação as classes de valor de compra, pode-se observar uma diferença entre a classe 10 em relação com seus demais, principalmente com a classe 2 apresentado um sentido oposto, agora para idade houve indícios de uma relação antagonista para as idade de clientes com faixa etária de jovens (18-25, 0-17, 26-35) e adultos (46-50, 51-55, 55+).

Na segunda dimensão, houve uma caracterização para a idade 0-17 que se destoa dos demais, além de uma aproximação em relação à classe 1, indicando uma possível aproximação destas informações. Mais informações e expressões foram apresentadas nos livros (Greenacre and Blasius 2006; Le Roux and Rouanet 2010; Lebart, Morineau, and Piron 1995)

Além disto, o pacote apresenta um função `FactoMineR::ellipseCA()` para construção de elipses de confiança utilizando o método de *bootstrap* para as categorias de cada variáveis (linhas e colunas), a partir das coordenadas da projeção nas dimensões 1 e 2.

```
plot1 <- plot(BlackFriday_CA, axes = c(1, 2))
plot2 <- FactoMineR::ellipseCA(BlackFriday_CA, method = 'boot')
```

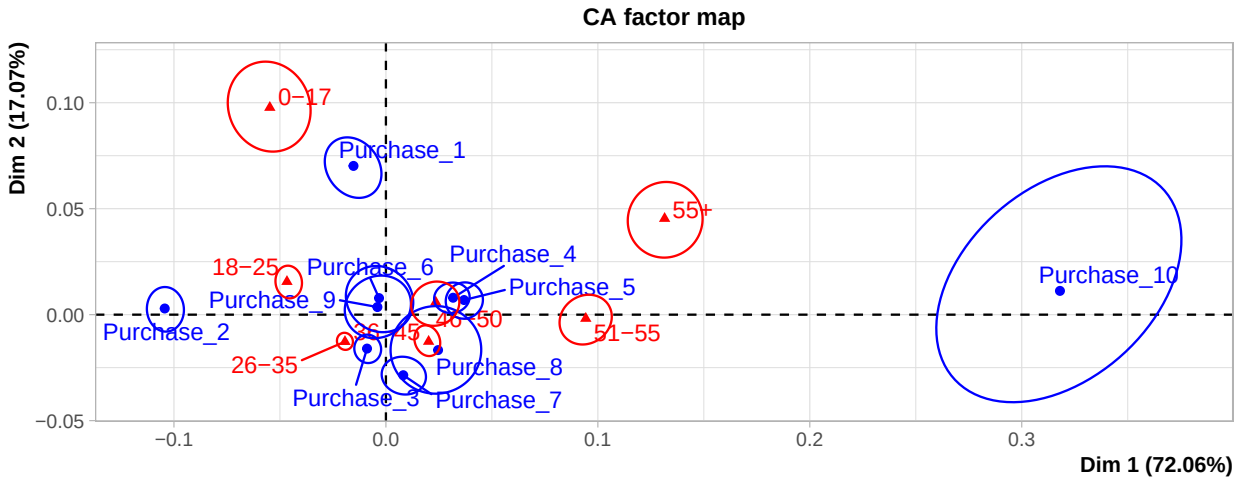


Figure 10: Gráfico da projeção pelo método CA das variáveis categóricas nas dimensões 1 e 2 (I) e suas respectivas elipse de confiança(I)

```
ggpubr::ggarrange(plotlist = list(plot1, plot2), ncol = 1, labels = c('I', 'II'))
```

Análise de Correspondência Múltipla

A Análise de Correspondência Multipla (*Multiple Correspondence Analysis* - MCA) é aplicado para tabelas em que os indivíduos estão nas linhas e categorias nas colunas (Lê, Josse, and Husson 2008).

Sintetizando a o método que é realizado a MCA, constrói-se uma matriz *dummy* a partir dos dados para cada indivíduo nas linhas em relação aos nível das categorias nas colunas, e depois é realizado o algoritmo a CA (Greenacre and Blasius 2006; Le Roux and Rouanet 2010)

Como o método é intensivo computacionalmente, foi selecionado de forma aleatória uma amostra de 500 indivíduos para a análise, a partir da função `base::sample()`.

```
set.seed(2020)
n_select <- sample(1:nrow(BlackFriday_data), size = 500)

BlackFriday_data %>%
```

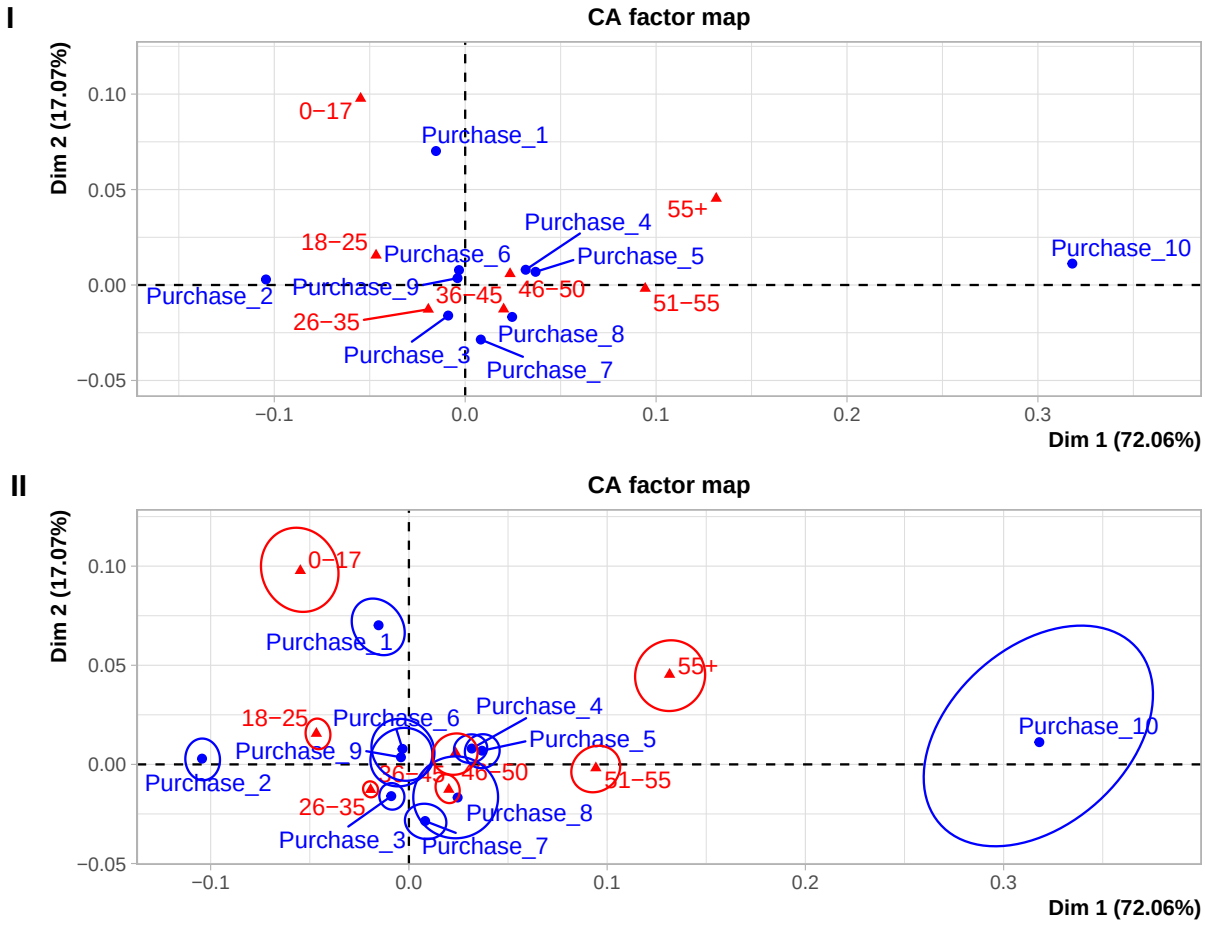


Figure 11: Gráfico da projeção pelo método CA das variáveis categóricas nas dimensões 1 e 2 (I) e suas respectivas elipse de confiança(I)

```
  dplyr::slice(n_select) %>%
  dplyr::select(Gender, Age, Occupation, Marital_Status, City_Category, Class_Purchase) %>%
  summary(maxsum = 10)
```

```
##  Gender     Age        Occupation  Marital_Status City_Category
##  F:119   0-17 : 17   4      : 64   0:300          A:126
##  M:381   18-25: 96   0      : 60   1:200          B:213
##          26-35:194   7      : 44                  C:161
##          36-45: 95   1      : 42
##          46-50: 45   20     : 36
##          51-55: 38   12     : 32
##          55+  : 15   2      : 29
##                      17     : 27
##                      14     : 25
##                      (Other):141
##      Class_Purchase
##  Purchase_1 : 26
##  Purchase_2 : 50
##  Purchase_3 :130
##  Purchase_4 : 98
##  Purchase_5 : 65
##  Purchase_6 : 24
##  Purchase_7 : 60
##  Purchase_8 : 15
##  Purchase_9 : 29
##  Purchase_10:  3
```

Pode-se visualizar como os 500 grupos foram agrupados em relação as variáveis categóricas de gênero, idade, ocupação, estado civil, categoria da cidade e classe de valor de compra.

```
BlackFriday_MCA <-
  BlackFriday_data %>%
  dplyr::slice(n_select) %>%
  dplyr::select(Gender, Age, Occupation, Marital_Status, City_Category, Class_Purchase) %>%
  FactoMineR::MCA(X =.,
                  ncp = 25,
                  graph = F,
                  axes = c(1, 2))
```

A análise foi alocada no objeto `BlackFriday_MCA`, e de modo similar ao CA podemos visualizar o resumo das informações (`summary.MCA()`) sobre as variâncias de cada dimensão, a coordenada das variáveis de linhas e colunas e suas respectivas contribuições. De maneira similar, também pode-se pedir os resultados das contribuições `$contrib`, coordenadas `$coord` e cossenos `$cos2` tanto das variáveis `$var` quantos do indivíduos `$ind`, porém como o conjunto de dados possuí grande quantidade de informação, optou-se apenas de gerar o gráfico de matriz para as contribuições das variáveis (Figura 12).

```
summary(BlackFriday_MCA)
```

```
##
## Call:
## FactoMineR::MCA(X = ., ncp = 25, graph = F, axes = c(1, 2))
##
##
## Eigenvalues
##                        Dim.1   Dim.2   Dim.3   Dim.4   Dim.5   Dim.6   Dim.7
## Variance               0.332   0.307   0.257   0.235   0.227   0.223   0.215
```

```
## % of var.              5.101   4.720   3.958   3.622   3.485   3.432   3.306
## Cumulative % of var.   5.101   9.821  13.779  17.401  20.887  24.319  27.625
##                        Dim.8   Dim.9  Dim.10  Dim.11  Dim.12  Dim.13  Dim.14
## Variance               0.209   0.202   0.196   0.194   0.192   0.186   0.182
## % of var.              3.214   3.107   3.015   2.985   2.952   2.865   2.800
## Cumulative % of var.  30.839  33.946  36.961  39.946  42.897  45.762  48.563
##                       Dim.15  Dim.16  Dim.17  Dim.18  Dim.19  Dim.20  Dim.21
## Variance               0.180   0.176   0.173   0.172   0.168   0.167   0.165
## % of var.              2.765   2.700   2.665   2.645   2.586   2.564   2.538
## Cumulative % of var.  51.328  54.029  56.694  59.339  61.925  64.489  67.026
##                       Dim.22  Dim.23  Dim.24  Dim.25  Dim.26  Dim.27  Dim.28
## Variance               0.162   0.154   0.148   0.146   0.142   0.140   0.137
## % of var.              2.492   2.374   2.281   2.250   2.191   2.152   2.100
## Cumulative % of var.  69.518  71.892  74.173  76.423  78.614  80.767  82.867
##                       Dim.29  Dim.30  Dim.31  Dim.32  Dim.33  Dim.34  Dim.35
## Variance               0.132   0.127   0.124   0.122   0.118   0.109   0.103
## % of var.              2.032   1.955   1.914   1.874   1.816   1.678   1.585
## Cumulative % of var.  84.899  86.854  88.769  90.642  92.458  94.136  95.722
##                       Dim.36  Dim.37  Dim.38  Dim.39
## Variance               0.098   0.087   0.064   0.029
## % of var.              1.507   1.339   0.987   0.445
## Cumulative % of var.  97.229  98.567  99.555 100.000
##
## Individuals (the 10 first)
##                  Dim.1    ctr   cos2    Dim.2    ctr   cos2    Dim.3    ctr
## 1             | -0.404  0.099  0.025 |  0.609  0.242  0.058 | -0.225  0.039
## 2             |  0.525  0.166  0.037 | -0.660  0.284  0.059 |  0.663  0.341
## 3             | -0.079  0.004  0.001 |  0.100  0.006  0.001 |  0.930  0.673
## 4             | -0.038  0.001  0.000 | -0.704  0.324  0.160 |  0.504  0.198
## 5             | -0.423  0.108  0.020 |  0.769  0.386  0.065 |  0.913  0.648
## 6             |  0.137  0.011  0.006 | -0.891  0.518  0.242 |  0.019  0.000
## 7             | -0.263  0.042  0.014 |  0.037  0.001  0.000 | -0.227  0.040
## 8             | -0.286  0.049  0.014 |  0.381  0.095  0.024 | -0.488  0.185
## 9             |  0.076  0.003  0.002 | -0.499  0.162  0.074 | -0.494  0.190
## 10            | -0.269  0.044  0.016 | -0.190  0.023  0.008 | -0.323  0.081
##                cos2
## 1             0.008 |
## 2             0.059 |
## 3             0.106 |
## 4             0.082 |
## 5             0.091 |
## 6             0.000 |
## 7             0.010 |
## 8             0.040 |
## 9             0.072 |
## 10            0.023 |
##
## Categories (the 10 first)
##                   Dim.1     ctr    cos2  v.test     Dim.2     ctr    cos2
## F             |  -0.286   0.976   0.025  -3.566 |   0.153   0.302   0.007
## M             |   0.089   0.305   0.025   3.566 |  -0.048   0.094   0.007
## 0-17          |   4.503  34.654   0.714  18.872 |   1.815   6.082   0.116
## 18-25         |   0.388   1.455   0.036   4.228 |  -1.112  12.904   0.294
## 26-35         |  -0.144   0.402   0.013  -2.553 |  -0.288   1.751   0.053
```

```
## 36-45        |  -0.206   0.406   0.010  -2.231 |   0.117   0.140   0.003
## 46-50        |  -0.666   2.008   0.044  -4.681 |   0.908   4.033   0.082
## 51-55        |  -0.669   1.710   0.037  -4.286 |   1.561  10.058   0.200
## 55+          |  -0.732   0.808   0.017  -2.875 |   1.372   3.069   0.058
## Occupation_0 |   0.030   0.005   0.000   0.244 |   0.052   0.017   0.000
##                v.test     Dim.3     ctr    cos2  v.test
## F               1.908 |   0.115   0.205   0.004   1.441 |
## M              -1.908 |  -0.036   0.064   0.004  -1.441 |
## 0-17            7.605 |  -0.362   0.288   0.005  -1.517 |
## 18-25         -12.111 |   1.076  14.389   0.275  11.712 |
## 26-35          -5.127 |  -0.507   6.463   0.163  -9.019 |
## 36-45           1.262 |  -0.568   3.972   0.076  -6.146 |
## 46-50           6.380 |  -0.447   1.163   0.020  -3.138 |
## 51-55           9.999 |   1.456  10.431   0.174   9.325 |
## 55+             5.391 |   1.335   3.463   0.055   5.244 |
## Occupation_0    0.427 |  -0.383   1.138   0.020  -3.157 |
##
## Categorical variables (eta2)
##                   Dim.1 Dim.2 Dim.3
## Gender          | 0.025 0.007 0.004 |
## Age             | 0.824 0.700 0.620 |
## Occupation      | 0.818 0.591 0.576 |
## Marital_Status  | 0.229 0.219 0.000 |
## City_Category   | 0.016 0.177 0.002 |
## Class_Purchase  | 0.076 0.146 0.341 |
```

```
BlackFriday_MCA %>%
  magrittr::extract2('var') %>%
  magrittr::extract2('cos2') %>%
  t() %>%
  corrplot::corrplot(is.corr = F, tl.col = 'black')
```

Para critérios de didática para apenas a visualização dos possíveis resultados, será considerado apenas as duas primeiras dimensões, porém seria necessário a utilização de mais dimensões que pode ser modificado utilizando o argumento `axes =` . Mesmo assim o *biplot* ficou poluído visualmente devido à grande quantidade de informações que é apresentado, com isto, pode-se utilizar o argumento `invisible = c('ind', 'var')` para selecionar quais das informações (indivíduos ou variáveis) não se demonstra na figura.

```
plot(BlackFriday_MCA, axes = c(1, 2))
```

```
plot_list <- lapply(c('ind', 'var'),
                    function(x) plot(BlackFriday_MCA, invisible = x, title = paste('MCA graph:', x)))

ggpubr::ggarrange(plotlist = plot_list, ncol = 1, labels = c('I', 'II'))
```

Discussão: foi observado que na dimensão 1 a faixa etária de 0-17 e a ocupação 10 foram as mais expressivas. Indicando que o perfil de clientes da *Black Friday* que tem entre 0-17 apresenta tipo de ocupação 10. De forma que, para a dimensão 2, o classe 10 de valor de compra, com a ocupação 10 e 13 e a faixa etária de 0-17 mostram-se significativas para a dimensão 2.

Além disto, com a função `FactoMineR::plotellipses`pode-se gerar os gráficos de elipse de confiança para as variáveis categóricas, utilizando as coordenadas relacionadas as respectivas dimensões, sendo possível ser alterado utilizando o argumento `axes =`.

```
plotellipses(BlackFriday_MCA,
             axes = c(1, 2))
```

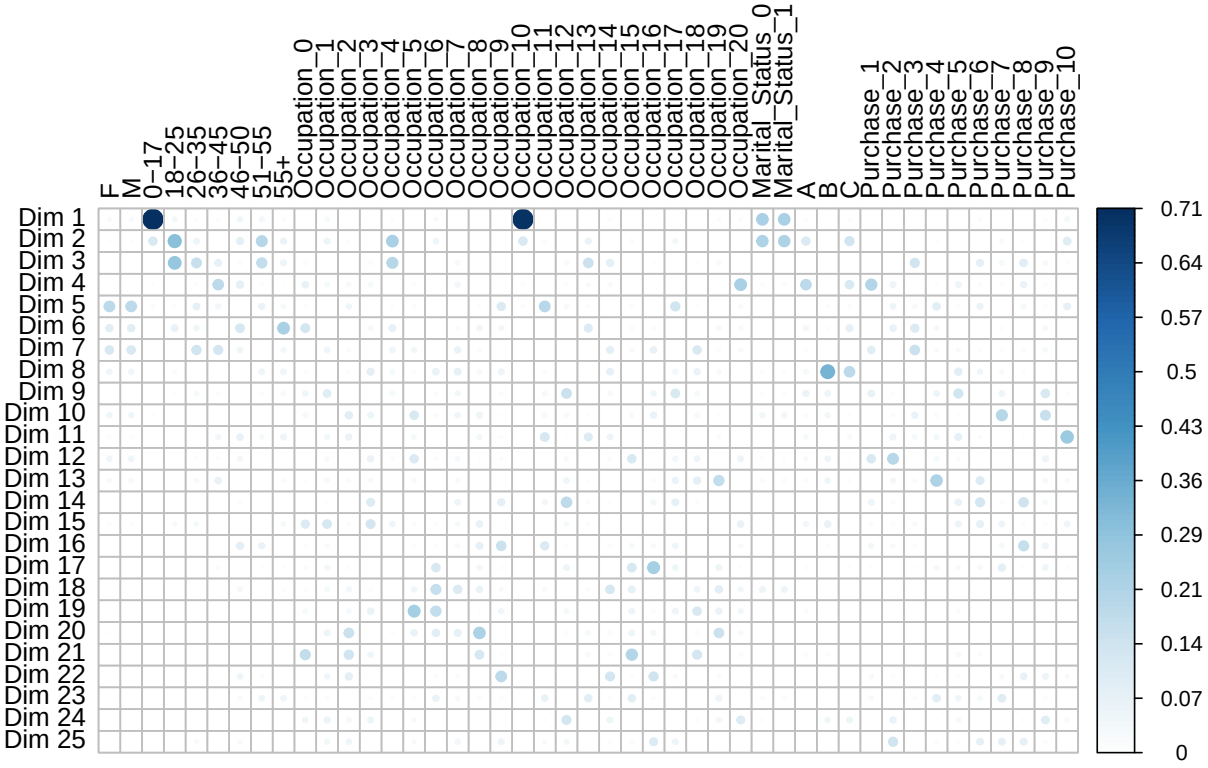


Figure 12: Gráfico para verificar quais variáveis mais contribuem para cada dimensão (cosseno)

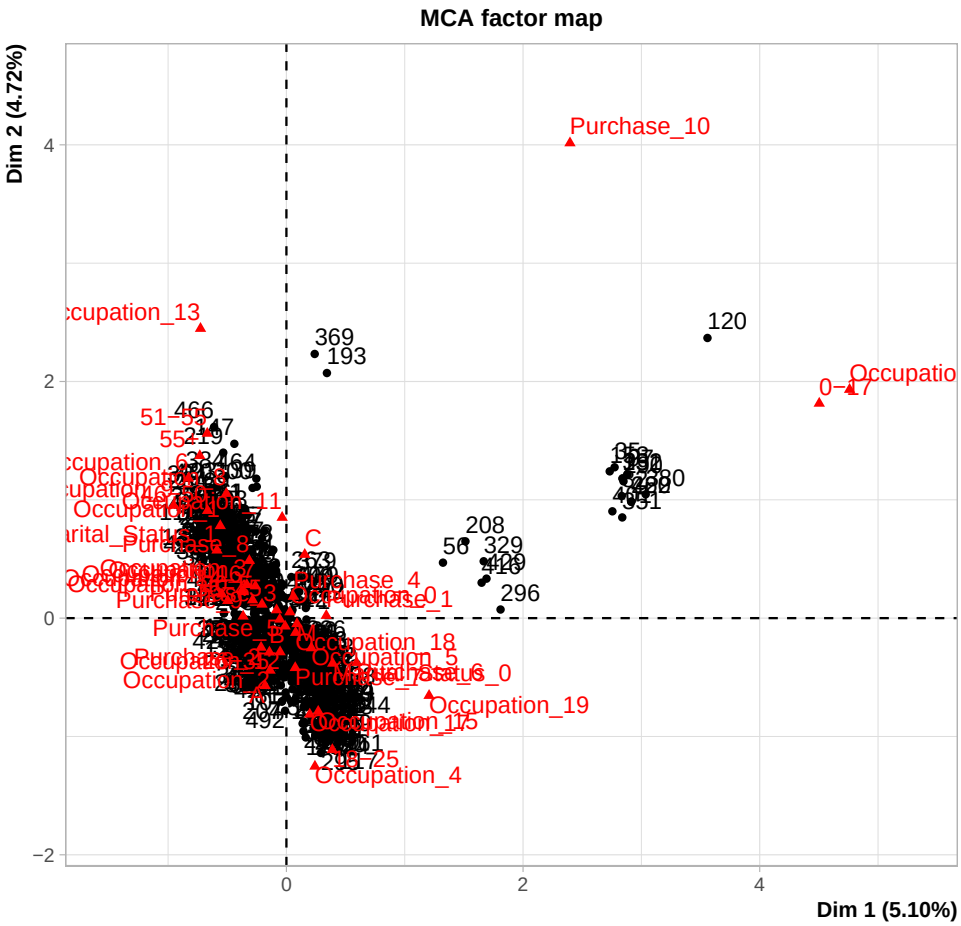
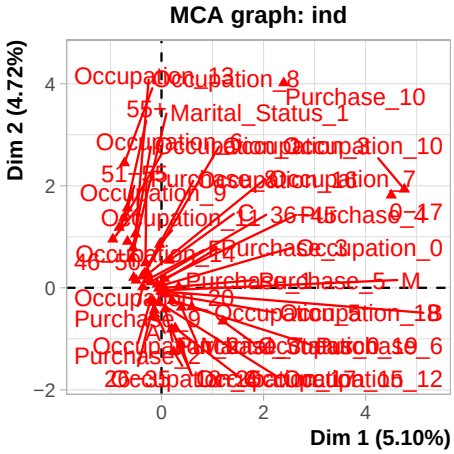


Figure 13: Gráfico da projeção pelo método MCA para os indivíduos e as variáveis nas dimensões 1 e 2

I



II

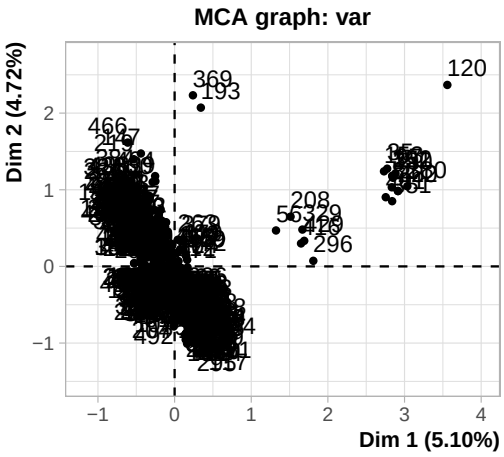


Figure 14: Gráfico da projeção pelo método MCA para os indivíduos (I) e as variáveis (II) nas dimensões 1 e 2

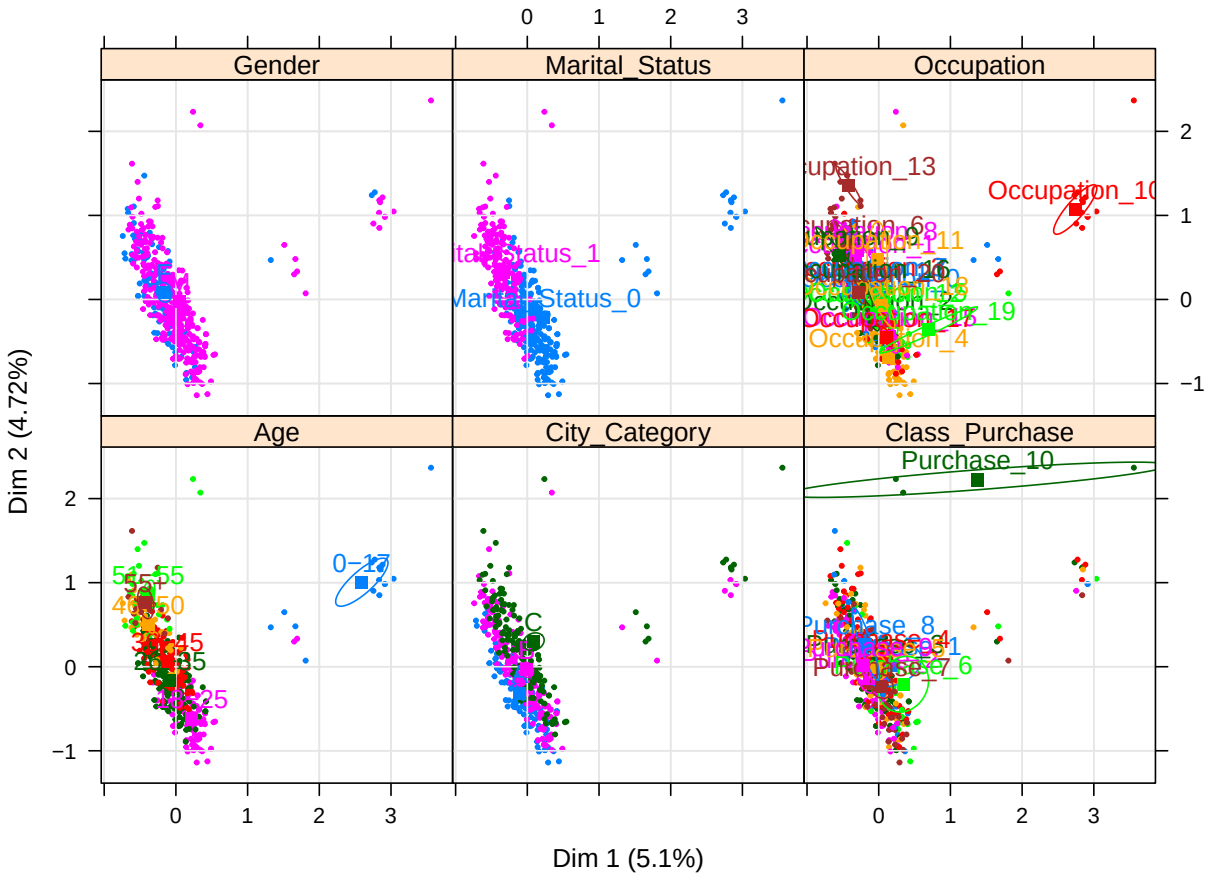


Figure 15: Elipse de confiança para MCA, para as categorias das variáveis

Considerações Finais

O pacote `FactoMineR` é completo para a análises exploratória de dados com estrutura multivariada, e atualmente isto é uma ferramente muito útil devido ao aumento de método para mineração de dados, além da sua facilidade para analisar tantos variáveis quantitativas, como qualitativas, ou o conjunto entre elas.

Porém deve se ter cuidado para quantidade de dados absurdamente grandes, para os dados da *Black Friday* a análise de correspondência múltipla não foi possível de ser realizada, devido a quantidade enorme de observações.

Mas com uma comunidade bastante ativa, tanto para tutoriais sobre o pacote, sanar dúvidas em forúns, e para novas implementações que estão sendo realizadas, é pacote bem confiável e com credibilidade, qualidades importantes para alguém que necessita destas análises, devido a quantidades grandes de bibliotecas que cresce na comunidade.

Outras funções podem ser abordadas futuramentes para estudo, como a função `FactoMineR::FAMD` que é dedicado à uma exploração de dados com variáveis contínuas e categóricas, que aborda a análise de ACP e MCA conjuntamente para balancear as influencias das diferentes características das variáveis. Ou até uma biblioteca nova que os autores estão trabalhando, a `missMDA` (Josse and Husson 2016) para imputação de dados devido a problemas que o método de ACP e MCA sofre quando tem observações com valores ausentes em algumas variáveis.

Referências

Greenacre, Michael, and Jorg Blasius. 2006. *Multiple Correspondence Analysis and Related Methods*. CRC press.

Husson, Francois, Julie Josse, and Jerome Pages. 2010. "Principal Component Methods-Hierarchical Clustering-Partitional Clustering: Why Would We Need to Choose for Visualizing Data." *Applied Mathematics Department*, 1–17.

Josse, Julie, and François Husson. 2016. "missMDA: A Package for Handling Missing Values in Multivariate Data Analysis." *Journal of Statistical Software* 70 (1): 1–31. https://doi.org/10.18637/jss.v070.i01.

Kassambara, Alboukadel. 2017a. *Practical Guide to Cluster Analysis in R: Unsupervised Machine Learning*. Vol. 1. Sthda.

———. 2017b. *Practical Guide to Principal Component Methods in R: PCA, M (ca), Famd, Mfa, Hcpc, Factoextra*. Vol. 2. STHDA.

Lebart, Ludovic, Alain Morineau, and Marie Piron. 1995. *Statistique Exploratoire Multidimensionnelle*. Vol. 3. Dunod Paris.

Le Roux, Brigitte, and Henry Rouanet. 2010. *Multiple Correspondence Analysis*. Vol. 163. Sage.

Lê, Sébastien, Julie Josse, and François Husson. 2008. "FactoMineR: An R Package for Multivariate Analysis." *Journal of Statistical Software, Articles* 25 (1): 1–18. https://doi.org/10.18637/jss.v025.i01.

R Core Team. 2020. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wei, Taiyun, and Viliam Simko. 2017. *R Package "Corrplot": Visualization of a Correlation Matrix*. https://github.com/taiyun/corrplot.