Guia para Reprodução de Tabelas, Gráficos e Figuras

Tabelas

Tabela 1

Apresenta as frequências dos perfis "puros" e mistos, após a classificação dos indivíduos com base em seus escores de GoM (g_{ik}) e segundo os critérios descritos no artigo. Os escores de GoM para cada indivíduo da amostra se encontram no arquivo "GoMK3(1).TXT".

Para importar os dados desse arquivo para o ambiente ativo do R, pode-se usar a função `read.cvs2` do pacote `utils`, que já vem pré-instalado com o programa:

```
ESEB2018Giks <- read.csv2("GoMK3(1).TXT", sep="")

# O arquivo "GoMK3(1).TXT" deve estar no diretório de trabalho.
# Caso não esteja, especificar o caminho.
```

A manipulação dos dados foi feita com o `dplyr`, um dos principais pacotes do `tidyverse`. Para instalar o `tidyverse` e todos os pacotes que o acompanham, basta usar o código `install.packages("tidyverse")`.

```
library(tidyverse)
```

O código a seguir cria a variável *Perfis_Eleitores*, com base nos escores de GoM dos indivíduos e nos critérios explicitados no artigo:

```
Giksperf <- ESEB2018Giks %>%
  mutate(Perfis_Eleitores = case_when(
    final_gik1 >= 0.7500 ~ "P1",
    final_gik2 >= 0.7500 ~ "P2",
    final_gik3 >= 0.7500 ~ "P3",
    final_gik1 > 0.5000 & final_gik1 < 0.7500 &
      final_gik2 <= 0.2500 &
      final_gik3 <= 0.2500 ~ "P1",
    final_gik2 > 0.5000 & final_gik2 < 0.7500 &
      final_gik1 <= 0.2500 &
      final_gik3 <= 0.2500 ~ "P2",
    final_gik3 > 0.5000 & final_gik3 < 0.7500 &
      final_gik1 <= 0.2500 &
      final_gik2 <= 0.2500 ~ "P3", #PX
    final_gik1 > 0.5000 & final_gik1 <= 0.7500 &
      final_gik2 >= 0.2500 &
      final_gik2 < 0.5000 ~ "MP12", #MP12
    final_gik1 > 0.500 & final_gik1 <= 0.7500 &
      final_gik3 >= 0.2500 &
      final_gik3 < 0.5000 ~ "MP13", #MP13
    final_gik2 > 0.5000 & final_gik2 <= 0.7500 &
```

```
      final_gik1 >= 0.2500 &
      final_gik1 < 0.5000 ~ "MP21", #MP21
    final_gik2 > 0.5000 & final_gik2 <= 0.7500 &
      final_gik3 >= 0.2500 &
      final_gik3 < 0.5000 ~ "MP23", #MP23
    final_gik3 > 0.5000 & final_gik3 <= 0.7500 &
      final_gik1 >= 0.2500 &
      final_gik1 < 0.5000 ~ "MP31", #MP31
    final_gik3 > 0.5000 & final_gik3 <= 0.7500 &
      final_gik2 >= 0.2500 &
      final_gik2 < 0.5000 ~ "MP32", #MP32
    final_gik1 < 0.5000 &
      final_gik2 < 0.5000 &
      final_gik3 < 0.5000 ~ "MSP" #MSP
  ))
```

Por fim, a tabela pode ser obtida por meio do código abaixo:

```
X <- c("P1", "MP12", "MP13", "P2", "MP21", "MP23", "P3", "MP31", "MP32", "MSP")

Tabela1 <- Giksperf %>%
  group_by(Perfis_Eleitores) %>%
  summarise(n=n()) %>%
  mutate(prop = round(prop.table(n)*100, digits = 2)) %>%
  slice(match(X, Perfis_Eleitores))

knitr::kable(Tabela1)
```

Perfis_Eleitores	n	prop
P1	242	9.66
MP12	81	3.23
MP13	140	5.59
P2	494	19.71
MP21	82	3.27
MP23	253	10.10
P3	553	22.07
MP31	156	6.23
MP32	279	11.13
MSP	226	9.02

A tabela foi salva como planilha do Microsoft Excel e então formatada (arquivo "Tabela1.xlsx"). Isso pode ser feito com a função `write.xlsx` do pacote `openxlsx`.

Observação: o separador decimal padrão no R é o ponto, enquanto no Excel costuma ser a vírgula. Ao salvar a tabela como planilha do Excel, basta substituir os pontos por vírgulas dentro do programa ou alterar as configurações do separador decimal. Isso permitirá realizar operações aritméticas e/ou gerar gráficos.

Tabela 2

Apresenta as probabilidades estimadas (λ) e Razões E/O (ou RLFM) para cada categoria l, de cada variável j e cada perfil extremo k. A tabela foi criada a partir da formatação no Excel do arquivo "LogGoMK3(1).TXT" (gerando o arquivo "Tabela2.xlsx").

Gráficos

Gráfico 1

Apresenta as chances de cada perfil extremo exibir um conjunto de características. Foi construído no Excel a partir dos valores das Razões E/O para uma seleção de categorias de resposta de variáveis. Para tanto, multiplicou-se o valor da Razão por 100 e em seguida se subtraiu o valor resultante por 100 (arquivo "Gráfico1.xlsx").

Gráfico 2

Apresenta a composição dos perfis "puros" e mistos de acordo com a variável sexo. Foi construído a partir de uma tabela de contingência entre as variáveis "Perfis dos Eleitores [Perfis_Eleitores]" e "sexo", salva posteriormente como planilha do Excel para formatação (arquivo "Gráfico2.xlsx").

Para a criação da tabela de contingência, foi utilizado o pacote `janitor`:

```
if(!require(janitor)) install.packages("janitor")
library(janitor)
```

Para criar uma tabela de contingência com o `janitor`, basta usar a função `tabyl` e selecionar as variáveis desejadas:

```
# Atribuindo os rótulos "Homem" e "Mulher" para a variável "sexo"
Giksperf$sexo <- factor(Giksperf$sexo, levels = c(1,2), labels = c("Homem", "Mulher"))

# Gerando a tabela de contingência
Tab_Grafico2 <- Giksperf %>%
  tabyl(Perfis_Eleitores, sexo) %>%
  adorn_percentages("row") %>%
  adorn_pct_formatting()

knitr::kable(
  Tab_Grafico2 %>%
  slice(match(X, Perfis_Eleitores))
)
```

Perfis_Eleitores	Homem	Mulher
P1	59.9%	40.1%
MP12	44.4%	55.6%
MP13	67.9%	32.1%
P2	32.4%	67.6%
MP21	39.0%	61.0%
MP23	36.8%	63.2%
P3	56.1%	43.9%
MP31	67.9%	32.1%
MP32	37.6%	62.4%
MSP	47.8%	52.2%

Figuras

Figura 1

Apresenta uma análise de correspondências múltiplas (MCA) utilizando duas tipologias de eleitores: aquela construída aqui e aquela apresentada por David Samuels e César Zucco Jr em seu livro *Partisans, Antipartisans, and Nonpartisans: Voting Behavior in Brazil* (Cambridge; New York: Cambridge University Press, 2018).

A partir da descrição desta última tipologia realizada no artigo e das questões sobre simpatia e rejeição partidária do ESEB 2018, é possível criar a variável "TipologiaSamuels":

```
Giksperf <- Giksperf %>%
  mutate(TipologiaSamuels = case_when(
    Q22A == 1 & P1 == 1 ~ "Partidário extremo",
    Q22A == 1 & P1 > 1 ~ "Partidário positivo",
    Q22A > 1 & P1 == 1 ~ "Partidário negativo",
    Q22A > 1 & P1 > 1 ~ "Não partidário")
    )
```

A figura pode ser obtida utilizando os pacotes `FactoMineR` e `factoextra` no R.

Para instalar e carregar os pacotes:

```
if(!require(FactoMineR)) install.packages("FactoMineR")
if(!require(factoextra)) install.packages("factoextra")
library(FactoMineR)
library(factoextra)
```

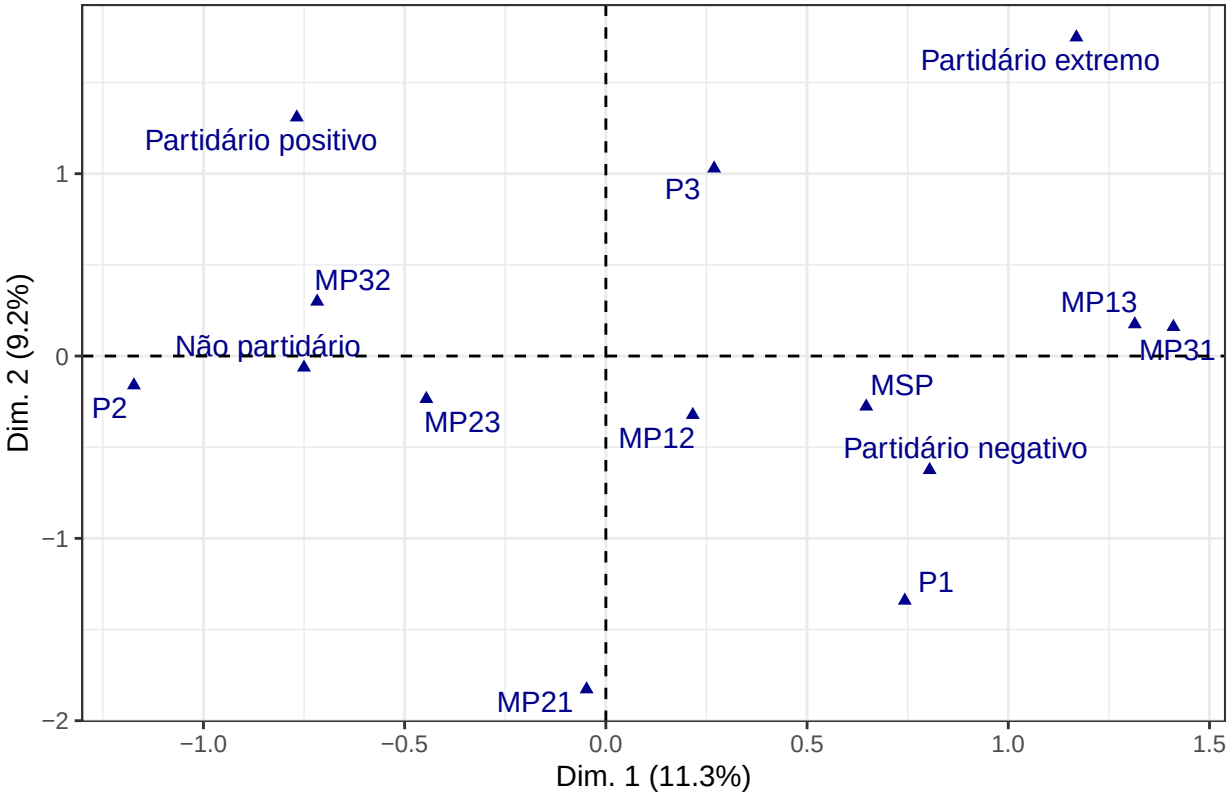
Para selecionar as duas tipologias de interesse e realizar a análise de correspondências múltiplas:

```
PerfisTipos <- Giksperf %>%
  select(Perfis_Eleitores, TipologiaSamuels)

AnaliseMult <- MCA(PerfisTipos)
```

Finalmente, a figura é gerada pelo código abaixo:

```
fviz_mca_var(AnaliseMult, col.var = "darkblue", repel = T) +
  theme_bw() +
  labs(title = "", x = "Dim. 1 (11.3%)", y = "Dim. 2 (9.2%)")
```



Informações de versões

- R version 3.6.2 (2019-12-12), `x86_64-w64-mingw32`
- Running under: `Windows 10 x64 (build 19043)`
- Matrix products: default
- Base packages: base, datasets, graphics, grDevices, methods, stats, utils
- Other packages: dplyr 1.0.6, factoextra 1.0.7, FactoMineR 2.4, forcats 0.5.1, ggplot2 3.3.5, janitor 2.1.0, purrr 0.3.4, readr 1.4.0, stringr 1.4.0, tibble 3.1.1, tidyr 1.1.3, tidyverse 1.3.1
- Loaded via a namespace (and not attached): abind 1.4-5, assertthat 0.2.1, backports 1.2.1, broom 0.7.8, car 3.0-11, carData 3.0-4, cellranger 1.1.0, cli 3.0.0, cluster 2.1.0, colorspace 2.0-1, compiler 3.6.2, crayon 1.4.1, curl 4.3.1, data.table 1.14.0, DBI 1.1.1, dbplyr 2.1.1, digest 0.6.27, DT 0.18, ellipsis 0.3.2, evaluate 0.14, fansi 0.4.2, farver 2.1.0, flashClust 1.01-2, foreign 0.8-72, fs 1.5.0, generics 0.1.0, ggpubr 0.4.0, ggrepel 0.9.1, ggsignif 0.6.2, glue 1.4.2, grid 3.6.2, gtable 0.3.0, haven 2.4.1, hms 1.1.0, htmltools 0.5.1.1, htmlwidgets 1.5.3, httr 1.4.2, jsonlite 1.7.2, knitr 1.33, labeling 0.4.2, lattice 0.20-38, leaps 3.1, lifecycle 1.0.0, lubridate 1.7.10, magrittr 2.0.1, MASS 7.3-51.4, modelr 0.1.8, munsell 0.5.0, openxlsx 4.2.3, pillar 1.6.1, pkgconfig 2.0.3, R6 2.5.0, Rcpp 1.0.6, readxl 1.3.1, reprex 2.0.0, rio 0.5.27, rlang 0.4.11, rmarkdown 2.9, rstatix 0.7.0, rstudioapi 0.13, rvest 1.0.0, scales 1.1.1, scatterplot3d 0.3-41, snakecase 0.11.0, stringi 1.6.1, tidyselect 1.1.1, tools 3.6.2, utf8 1.2.1, vctrs 0.3.8, withr 2.4.2, xfun 0.24, xml2 1.3.2, yaml 2.2.1, zip 2.1.1