

Dados pareados

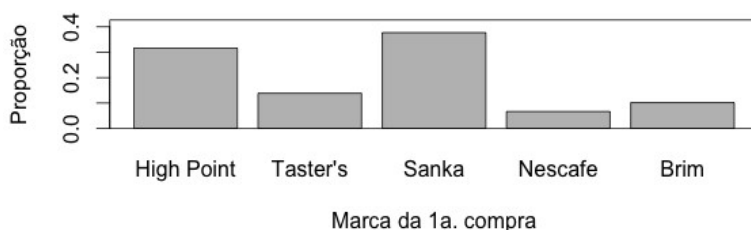
```
## Dados
## Tabela 9.5, p. 236 em Agresti (1996), Introduction to Categorical
## Data Analysis
```

```
tab95 <- matrix(c(93, 17, 44, 7, 10,
                  9, 46, 11, 0, 9,
                  17, 11, 155, 9, 12,
                  6, 4, 9, 15, 2,
                  10, 4, 12, 2, 27), ncol = 5, byrow = TRUE)
rownames(tab95) <- colnames(tab95) <- c("High Point", "Taster's",
    "Sanka", "Nescafe", "Brim")
addmargins(tab95)
n <- sum(tab95)
cat("\n n =", n, "\n")
```

```
n = 541
```

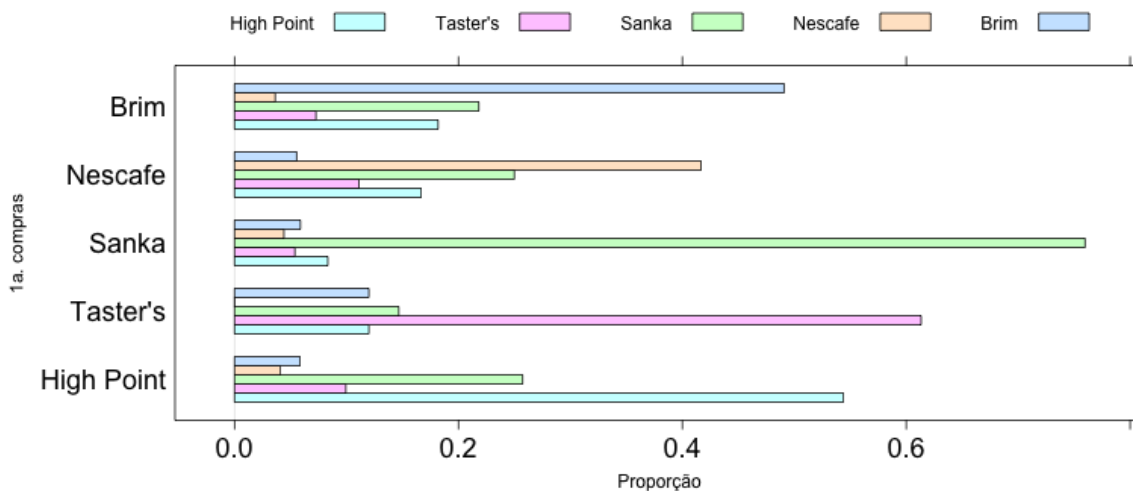
```
## Distribuições marginais
marg1 <- margin.table(tab95, margin = 1) / n
marg2 <- margin.table(tab95, margin = 2) / n
r12 <- c(0, max(marg1, marg2))
```

```
par(mfrow = c(2, 1))
barplot(marg1, xlab = "Marca da 1a. compra", ylab = "Proporção",
    ylim = r12)
box()
barplot(marg2, xlab = "Marca da 2a. compra", ylab = "Proporção",
    ylim = r12)
box()
```



```
## Distribuições condicionais
library(lattice)

tab95c <- prop.table(tab95, margin = 1)
barchart(tab95c, xlab = "Proporção", ylab = "1a. compras", stack = FALSE,
  scale = list(cex = 1.5), auto.key = list(space = "top",
  columns = length(colnames(tab95c))))
```



Nota 1. Interprete o gráfico acima e comente sobre a independência entre as variáveis.

1. Modelos

Em seguida os dados são organizados de uma forma conveniente para o ajuste de modelos log-lineares. A função `as.vector` empilha as colunas de uma matriz.

```
## Dados
compra1 <- factor(colnames(tab95), levels = colnames(tab95))
compra2 <- compra1
dados <- expand.grid(compra1 = compra1, compra2 = compra2)
dados$freq <- as.vector(tab95)
dados
```

	compra1	compra2	freq
1	High Point	High Point	93
2	Taster's	High Point	9
3	Sanka	High Point	17
4	Nescafe	High Point	6
5	Brim	High Point	10
6	High Point	Taster's	17
7	Taster's	Taster's	46
8	Sanka	Taster's	11
9	Nescafe	Taster's	4
10	Brim	Taster's	4
11	High Point	Sanka	44
12	Taster's	Sanka	11

```

13      Sanka      Sanka  155
14      Nescafe    Sanka   9
15      Brim       Sanka  12
16 High Point    Nescafe   7
17 Taster's     Nescafe   0
18      Sanka     Nescafe   9
19      Nescafe   Nescafe  15
20      Brim      Nescafe   2
21 High Point    Brim    10
22 Taster's     Brim     9
23      Sanka     Brim    12
24      Nescafe   Brim     2
25      Brim      Brim    27

```

```

## Modelo de independência
mind <- glm(freq ~ compra1 + compra2, family = poisson, data = dados)

```

```

# G2 e X2
X2ind <- sum(resid(mind, type = "pearson")^2)
cat("\n Modelo de independência (g.l. = ", mind$df.residual,")")
cat("\n G2 = ", mind$deviance, " (p =", pchisq(mind$deviance,
mind$df.residual, lower.tail = FALSE), ")")
cat("\n X2 = ", X2ind, " (p =", pchisq(X2ind, mind$df.residual,
lower.tail = FALSE), ")")

```

```

Modelo de independência (g.l. = 16 )
G2 = 346.381 (p = 5.879941e-64 )
X2 = 463.3044 (p = 1.817251e-88 )

```

O ajuste do modelo de simetria requer uma variável auxiliar formada por pares de linhas e colunas (i,j) e tais que o par (j,i) seja igual ao par (i,j) sempre que $i \neq j$, para i e $j = 1, \dots, I$. O número de pares é I^2 , que correspondem a $I + I(I-1)/2 = 15$ pares não redundantes (ou únicos).

```

## Modelo de simetria
# Variável auxiliar com um valor para cada diferente probabilidade
dados$aux <- factor(paste(pmin(as.numeric(dados$compra1),
as.numeric(dados$compra2)), pmax(as.numeric(dados$compra1),
as.numeric(dados$compra2))), sep = ",")

```

```
length(unique(dados$aux))
```

15

```
msim <- glm(freq ~ aux, family = poisson, data = dados)
```

```

# G2 e X2
X2s <- sum(resid(msim, type = "pearson")^2)
cat("\n Modelo de simetria (g.l. = ", msim$df.residual,")")
cat("\n G2 = ", msim$deviance, " (p =", round(pchisq(msim$deviance,
msim$df.residual, lower.tail = FALSE), 4), ")")
cat("\n X2 = ", X2s, " (p =", round(pchisq(X2s, msim$df.residual,
lower.tail = FALSE), 4), ")")

```

```

Modelo de simetria (g.l. = 10 )
G2 = 22.47293 (p = 0.0129 )
X2 = 20.41236 (p = 0.0256 )

```

Para ajustar o modelo de quase independência são necessárias $I (= 5)$ variáveis auxiliares assumindo em cada par (i,j) valor 0 se $i \neq j$ e valor 1 se $i = j$.

```

## Modelo de quase independência
# Variáveis auxiliares
dados$D1 <- as.numeric(dados$aux == "1,1")
dados$D2 <- as.numeric(dados$aux == "2,2")
dados$D3 <- as.numeric(dados$aux == "3,3")
dados$D4 <- as.numeric(dados$aux == "4,4")
dados$D5 <- as.numeric(dados$aux == "5,5")

mqind <- glm(freq ~ compra1 + compra2 + D1 + D2 + D3 + D4 + D5,
             family = poisson, data = dados)

# G2 e X2
X2qind <- sum(resid(mqind, type = "pearson")^2)
cat("\n Modelo de quase independência (g.l. = ", mqind$df.residual,")")
cat("\n G2 = ", mqind$deviance, "(p =", round(pchisq(mqind$deviance,
mqind$df.residual, lower.tail = FALSE), 4), ")")
cat("\n X2 = ", X2qind, "(p =", round(pchisq(X2qind, mqind$df.residual,
lower.tail = FALSE), 4), ")")

```

```

Modelo de quase independência (g.l. = 11 )
G2 = 13.78563 (p = 0.2451 )
X2 = 12.24792 (p = 0.3453 )

```

Nota 2. O modelo de quase independência ajusta bem os dados? Compare as frequências observadas e as frequências esperadas estimadas.

```

## Teste de homogeneidade marginal
g2hm <- msim$deviance - mqsim$deviance
gl <- msim$df.residual - mqsim$df.residual
cat("\n Teste de homogeneidade marginal \n G2 = ", g2hm,
    "(p =", round(pchisq(g2hm, gl, lower.tail = FALSE), 4),
    ", g.l. = ", gl,")")

Teste de homogeneidade marginal
G2 = 12.49889 (p = 0.014 , g.l. = 4 )

```

2. kappa de Cohen

```

# kappa de Cohen
library(vcd)
(k95 <- Kappa(tab95))
confint(k95)

```

Nos resultados abaixo, ASE indica o erro padrão assintótico. Os limites do intervalo de confiança assintótico de 95% são indicados por *lwr* (*lower*: inferior) e *upr* (*upper*: superior).

```

      value      ASE      z  Pr(>|z|)
Unweighted 0.4765 0.02805 16.99 1.060e-64
Weighted   0.4527 0.03297 13.73 6.473e-43

```

```
confint(k95)
```

```

Kappa      lwr      upr
Unweighted 0.4214736 0.5314329
Weighted   0.3880910 0.5173130

```

Nota 3. Para estes dados, a medida kappa ponderada (Weighted) é aplicável?

3. Inferência em uma tabela 2 × 2 com dados pareados

```

## Teste de McNemar
# Dados
# Tabela 10.1, p. 350 em Agresti (1990), Categorical Data Analysis
tab101 <- matrix(c(794, 150, 86, 570), byrow = TRUE, ncol = 2)
rownames(tab101) <- c("1. Aprovação", "1. Desaprovação")
colnames(tab101) <- c("2. Aprovação", "2. Desaprovação")
addmargins(tab101)

```

	2. Aprovação	2. Desaprovação	Sum
1. Aprovação	794	150	944
1. Desaprovação	86	570	656
Sum	880	720	1600

```

n <- sum(tab101)
cat("\n n =", n, "\n")

```

```
n = 1600
```

```

# Proporções amostrais e teste
prop101 <- prop.table(tab101)
round(prop101, 3)

```

	2. Aprovação	2. Desaprovação
1. Aprovação	0.496	0.094
1. Desaprovação	0.054	0.356

```
mcnemar.test(tab101, correct = FALSE)
```

```
McNemar's chi-squared = 17.3559, df = 1, p-value = 3.099e-05
```

```

# Teste exato (distribuição binomial)
# H1: pi1+ > pi+1 (aprovação 1o. > aprovação 2o.)
ns <- tab101[1, 2] + tab101[2, 1] # n*

```

```

valorp <- pbinom(tab101[1, 2] - 1, ns, 0.5, lower.tail = FALSE)
cat("\n Teste exato para H1: pi1+ > pi+1 \n n12 =", tab101[1, 2],
    ", n* =", ns, "(p =", valorp, ") \n")

    Teste exato para H1: pi1+ > pi+1
    n12 = 150 , n* = 236 (p = 1.857968e-05 )

# Diferença de aprovação (2o. - 1o.)
(estd <- prop101[2, 1] - prop101[1, 2])

    -0.04

# Erro padrão da estimativa
plm <- sum(prop101[1, ]) # p1+
pm1 <- sum(prop101[, 1]) # p+1
epest <- sqrt((plm * (1 - plm) + pm1 * (1 - pm1) - 2 *
    abs((prod(diag(prop101)) - prop101[1, 2] * prop101[2, 1]))) / n)
cat("\n e.p.(d) =", epest, "\n")

    e.p.(d) = 0.009549215

# IC para a diferença de aprovação (2o. - 1o.)
conf <- 0.95
icdif <- estd + c(-1, 1) * qnorm((1 + conf) / 2) * epest
cat("\n IC de", conf * 100, "% para a diferença (2o. - 1o.): \n", icdif,
    "\n")

    IC de 95 % para a diferença (2o. - 1o.):
    -0.05871612 -0.02128388

```

Nota 4. Interprete o resultado acima.