



CE073 - Análise de Dados Categóricos

Trabalho No. 3

Luiz Henrique Barretta Francisco - GRR20213026

novembro/2023

8-a) O que significaria independência e dependência entre as linhas e colunas no contexto de DAVFs? Qual conclusão você acha que os pesquisadores prefeririam para justificar seu sistema de classificação? Explique.

A independência entre linhas e colunas mostraria a ineficiência das classificações em relação aos sintomas, dado que nesse cenário, qual fosse a classificação assinalada não resultaria em diferença nas probabilidades dos sintomas. No exemplo, sejam S e C as categorias para os sintomas e as classificações, respectivamente, e s_0 o evento equivalente à hemorragia. A independência nesse caso seria dada se $P(S = s_0 | C = c_i) = P(S = s_0)$, ou seja, a probabilidade de um sintoma de hemorragia é independente da classificação aplicada.

Um resultado estatístico que rejeita a independência, ou seja, que admite a dependência entre linhas e colunas, é suficiente para justificar a proposta do sistema de classificação e, portanto, é o cenário desejado pelos pesquisadores.

8-b) Cognard et al. (1995) afirmam que “a tabela de contingência mostra uma diferença estatisticamente significativa ($P = 0,0001$)” sem mencionar o teste estatístico específico realizado. Realize um teste qui-quadrado de Pearson para independência para produzir um resultado de significância semelhante.

```
require(knitr)
sintomas <- c("Hemorragia", "Hipertensão intracraniana", "Déficit neurológico",
              "Convulsões", "Deficiência cardíaca", "Mielopatia", "Sintomas não agressivos")
classificacoes <- c("1", "2a", "2b", "2a e 2b", "3", "4", "5")
davf <- matrix(c(0, 1, 0, 0, 0, 0, 83, 0, 8, 0, 1, 1, 0, 17, 2, 1, 0, 0, 0, 0, 7,
                 1, 2, 6, 2, 1, 0, 6, 10, 0, 8, 1, 0, 0, 6, 19, 4, 2, 3, 0, 0, 1,
                 5, 1, 0, 0, 0, 0, 6, 0),
               nrow = length(sintomas), ncol = length(classificacoes),
               dimnames = list(sintomas, classificacoes))
kable(davf)
```

	1	2a	2b	2a e 2b	3	4	5
Hemorragia	0	0	2	1	10	19	5
Hipertensão intracraniana	1	8	1	2	0	4	1
Déficit neurológico	0	0	0	6	8	2	0
Convulsões	0	1	0	2	1	3	0
Deficiência cardíaca	0	1	0	1	0	0	0
Mielopatia	0	0	0	0	0	0	6
Sintomas não agressivos	83	17	7	6	6	1	0

```
chisq.test(x = davf, correct = FALSE)

##
## Pearson's Chi-squared test
##
## data:  davf
## X-squared = 303.2, df = 36, p-value < 2.2e-16
```

8-c) É improvável que pacientes com DAVF apresentem apenas um sintoma. Por exemplo, um paciente pode ter hipertensão intracraniana e convulsões enquanto possui uma classificação de nível 3. De fato, as contagens totais da tabela podem ser maiores ou menores que o número de pacientes. Esse cenário alternativo invalidaria o uso do teste qui-quadrado de Pearson nessa situação? Explique.

O uso do teste qui-quadrado de Pearson é apropriado quando cada observação contribui para uma única célula em uma tabela de contingência, onde se busca analisar a independência entre duas variáveis categóricas. No entanto, se os pacientes com DAVF puderem contribuir para nenhuma, uma ou mais de uma célula na tabela devido à possibilidade de apresentarem múltiplos sintomas e/ou classificações, os pressupostos do teste qui-quadrado são violados. Isso ocorre porque as observações não são independentes, tornando a análise inadequada. Nesse cenário, seria mais apropriado utilizar métodos estatísticos como a análise de regressão logística, que leva em consideração múltiplas variáveis independentes e permite modelar a probabilidade de ocorrência de sintomas de forma mais adequada. Portanto, o teste qui-quadrado de Pearson não seria válido para avaliar a independência entre as variáveis nessas condições.

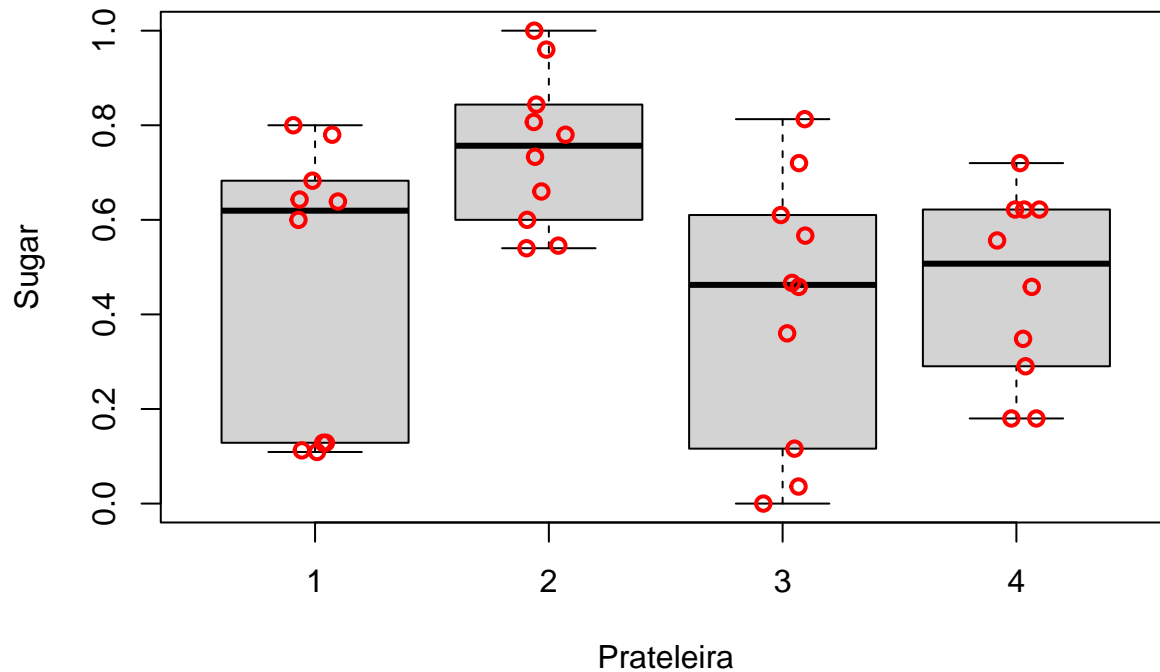
11-a) As variáveis explicativas precisam ser reformatadas antes de continuar.

```
cereal = read.csv(file = "http://leg.ufpr.br/~lucambio/ADC/cereal_dillons.csv")
stand01 <- function (x) { (x - min(x))/( max(x) - min(x)) }
cereal <- data.frame (Shelf = cereal$Shelf,
                     sugar = stand01 (x = cereal$sugar_g / cereal$size_g ),
                     fat = stand01 (x = cereal$fat_g / cereal$size_g ),
                     sodium = stand01 (x = cereal$sodium_mg / cereal$size_g ))
head(cereal)
```

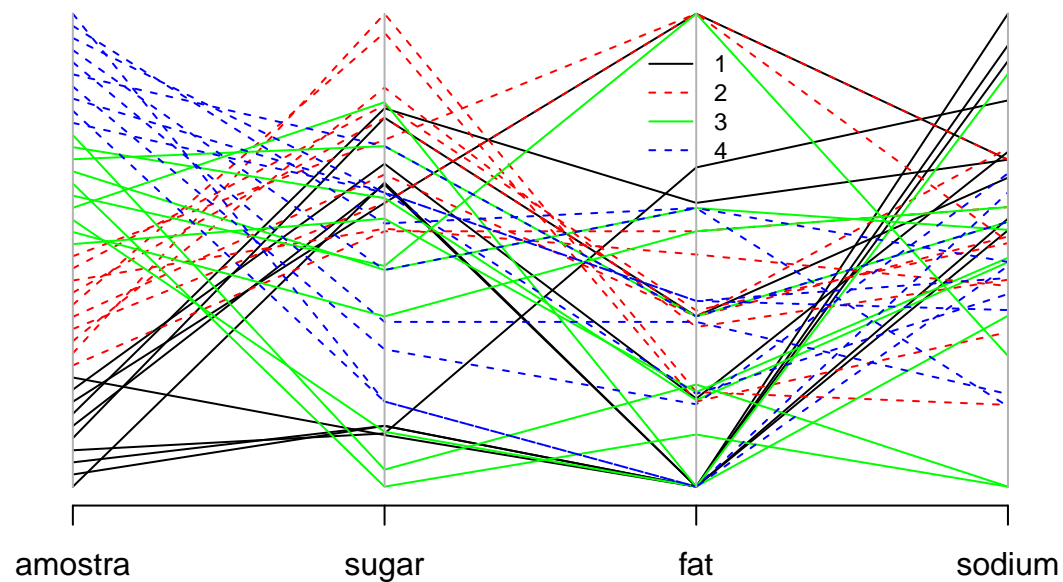
```
##  Shelf      sugar    fat    sodium
## 1      1 0.6428571 0.000 0.5666667
## 2      1 0.1285714 0.000 0.9000000
## 3      1 0.1285714 0.000 1.0000000
## 4      1 0.1125000 0.675 0.8166667
## 5      1 0.7800000 0.360 0.6533333
## 6      1 0.6387097 0.000 0.5419355
```

11-b) Construa gráficos de box-plots (caixas) lado a lado com gráficos de pontos sobrepostos para cada uma das variáveis explicativas. Além disso, construa um gráfico de coordenadas paralelas para as variáveis explicativas e o número da prateleira. Discuta se existem possíveis diferenças de conteúdo entre as prateleiras.

```
boxplot(formula = sugar ~ Shelf , data = cereal , ylab = "Sugar", xlab = "Prateleira",
        pars = list (outpch = NA))
stripchart(x = cereal$sugar ~ cereal$Shelf , lwd = 2, col = "red",
           method = "jitter", vertical = TRUE , pch = 1, add = TRUE)
```



```
require(MASS)
cereal2 <- data.frame(amostra = 1:nrow(cereal), cereal[,2:4])
cereal.colors <- ifelse(cereal$Shelf=="1", "black",
                        ifelse(cereal$Shelf=="2", "red",
                                ifelse(cereal$Shelf=="3", "green",
                                        "blue"))))
cereal.lty <- ifelse(cereal$Shelf=="1", "solid",
                     ifelse(cereal$Shelf=="2", "dashed",
                             ifelse(cereal$Shelf=="3", "solid",
                                     "dashed"))))
parcoord(x = cereal2, col = cereal.colors, lty = cereal.lty)
legend(x = 2.78, y = 0.955, legend = c("1", "2", "3", "4"), lty = unique(cereal.lty),
       col = unique(cereal.colors), cex = 0.75, bty = "n", text.col = "black")
```



Observou-se as seguintes tendências nas prateleiras:

1. Prateleira 1: Baixo nível de *fat* e alto de *sodium*.
2. Prateleira 2: Alto nível de *sugar*.
3. Prateleira 3: Aparentemente aleatório, sem nenhum padrão sistemático.
4. Prateleira 4: Níveis entre baixo e médio das três variáveis: *sugar*, *fat* e *sodium*.

11-c) A resposta tem valores de 1, 2, 3 e 4. Em que cenário seria desejável levar em conta a ordinalidade. Você acha que isso ocorre aqui?

A consideração da ordinalidade é relevante no cenário em que a variável resposta é o número da prateleira em um supermercado. Nesse contexto, a ordem das prateleiras, numeradas de baixo (1) para cima (4), é intrinsecamente significativa, uma vez que a localização das caixas de cereais nas diferentes prateleiras afeta a visibilidade e a probabilidade de compra. Portanto, a ordem natural das prateleiras desempenha um papel fundamental na interpretação dos resultados, tornando a consideração da ordinalidade apropriada ao analisar como os níveis de teor de açúcar, gordura e sódio dos cereais influenciam sua localização nas prateleiras.

11-d) Estime um modelo de regressão multinomial com formas lineares das variáveis açúcar, gordura e sódio. Realize testes da razão de verossimilhanças (LRTs) para examinar a importância de cada variável explicativa.

```
require(package = nnet)
mod.fit <- multinom(Shelf ~ sugar + fat + sodium, data = cereal, trace = FALSE)
summary(mod.fit)
```

```
## Call:
## multinom(formula = Shelf ~ sugar + fat + sodium, data = cereal,
##          trace = FALSE)
##
## Coefficients:
##      (Intercept)      sugar      fat      sodium
## 2      6.900708    2.693071    4.0647092 -17.49373
## 3     21.680680   -12.216442   -0.5571273 -24.97850
## 4     21.288343   -11.393710   -0.8701180 -24.67385
##
## Std. Errors:
##      (Intercept)      sugar      fat      sodium
## 2      6.487408    5.051689    2.307250    7.097098
## 3      7.450885    4.887954    2.414963    8.080261
## 4      7.435125    4.871338    2.405710    8.062295
##
## Residual Deviance: 67.19028
## AIC: 91.19028
```

```
require(package = car)
Anova(mod.fit)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Shelf
##          LR Chisq Df Pr(>Chisq)
## sugar    22.7648  3  4.521e-05 ***
## fat       5.2836  3    0.1522
## sodium   26.6197  3  7.073e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com base na análise de deviance (LRT) para o modelo de regressão multinomial, pode-se observar que as variáveis explicativas *sugar* e *sodium* são estatisticamente significativas, uma vez que apresentam p-valores muito baixos, indicando que essas variáveis têm um impacto significativo nas categorias da variável resposta *Shelf*. Em contraste, a variável *fat* não é estatisticamente significativa, pois seu p-valor é relativamente alto (0.1522), sugerindo que a presença ou ausência de gordura nos cereais pode não ter um efeito significativo na determinação da prateleira em que são colocados nas mercearias. Portanto, *sugar* e *sodium* emergem como as variáveis explicativas mais relevantes na explicação das prateleiras de cereais matinais em supermercados, enquanto *fat* não demonstra significância estatística.

11-e) Mostre que não há interações significativas entre as variáveis explicativas (incluindo uma interação entre as três variáveis).

```
mod.fit2 <- multinom(Shelf ~ sugar + sodium, data = cereal, trace = FALSE)
mod.fit.iter <- multinom(Shelf ~ sugar*fat*sodium, data = cereal, trace = FALSE)
summary(mod.fit.iter)
```

```
## Call:
## multinom(formula = Shelf ~ sugar * fat * sodium, data = cereal,
##      trace = FALSE)
##
## Coefficients:
##      (Intercept)      sugar      fat      sodium sugar:fat sugar:sodium fat:sodium
## 2    -4.563627    8.944868 22.063003   1.030077  35.60873  -12.250084  -23.75955
## 3    24.498320 -22.248456 35.981865 -27.899087 -17.12487   13.253103  -59.54150
## 4    27.246742 -21.852777  7.298799 -29.106797  41.08251    2.887805  -30.85250
##      sugar:fat:sodium
## 2          -55.88455
## 3           37.71571
## 4          -22.59552
##
## Std. Errors:
##      (Intercept)      sugar      fat      sodium sugar:fat sugar:sodium fat:sodium
## 2    25.21113 29.72894  96.57821 27.29915 135.1117   31.98647 116.0776
## 3    22.83750 25.81043 101.17670 24.61166 150.1228   26.89827 138.0237
## 4    22.80359 26.00692 100.83444 24.51538 150.6750   28.86631 138.5448
##      sugar:fat:sodium
## 2          158.8091
## 3          212.2222
## 4          217.3953
##
## Residual Deviance: 53.54496
## AIC: 101.545
```

```
Anova(mod.fit.iter)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Shelf
##
##      LR Chisq Df Pr(>Chisq)
## sugar      19.2525  3 0.0002424 ***
## fat         6.1167  3 0.1060686
## sodium     30.8407  3 9.183e-07 ***
## sugar:fat    3.2309  3 0.3573733
## sugar:sodium 3.0185  3 0.3887844
## fat:sodium   3.1586  3 0.3678151
```

```
## sugar:fat:sodium    2.5884  3  0.4595299
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

anova(mod.fit2, mod.fit.iter)

## Likelihood ratio tests of Multinomial Models
##
## Response: Shelf
##
##      Model Resid. df Resid. Dev   Test      Df LR stat.  Pr(Chi)
## 1      sugar + sodium      111   72.47384
## 2 sugar * fat * sodium      96   53.54496 1 vs 2      15 18.92888 0.216975
```

O teste de razão de verossimilhanças (LRT) entre dois modelos multinomiais, um com apenas as variáveis *sugar* e *sodium* e outro mais complexo com a inclusão das interações todas as três variáveis, não revelou uma melhora significativa ao adicionar as interações. O teste de razão de verossimilhanças comparando os dois modelos não mostrou diferença estatisticamente significativa entre eles. Isso sugere que as interações entre as variáveis explicativas não desempenham um papel significativo na explicação das categorias da variável resposta *Shelf* no contexto do modelo multinomial. Portanto, o modelo mais simples, sem as interações, é preferível, uma vez que fornece uma explicação adequada para os dados.

11-f) Para uma porção de 28 gramas, seu teor de açúcar é de 12 gramas, o teor de gordura é de 0.5 gramas e o teor de sódio é de 130 miligramas. Estime as probabilidades de prateleira para Apple Jacks.

```
cereal3 = read.csv(file = "http://leg.ufpr.br/~lucambio/ADC/cereal_dillons.csv")
kellogs <- data.frame(sugar = (12/28 - min(cereal3$sugar_g/cereal3$size_g))/
  (max(cereal3$sugar_g/cereal3$size_g)-min(cereal3$sugar_g/cereal3$size_g)),
  fat = (0.5/28 - min(cereal3$fat_g/cereal3$size_g))/
  (max(cereal3$fat_g/cereal3$size_g)-min(cereal3$fat_g/cereal3$size_g)),
  sodium = (130/28 - min(cereal3$sodium_g/cereal3$size_g))/
  (max(cereal3$sodium_g/cereal3$size_g)-min(cereal3$sodium_g/cereal3$size_g)))
round(predict(object = mod.fit2, newdata = kellogs, type = "probs"),2)

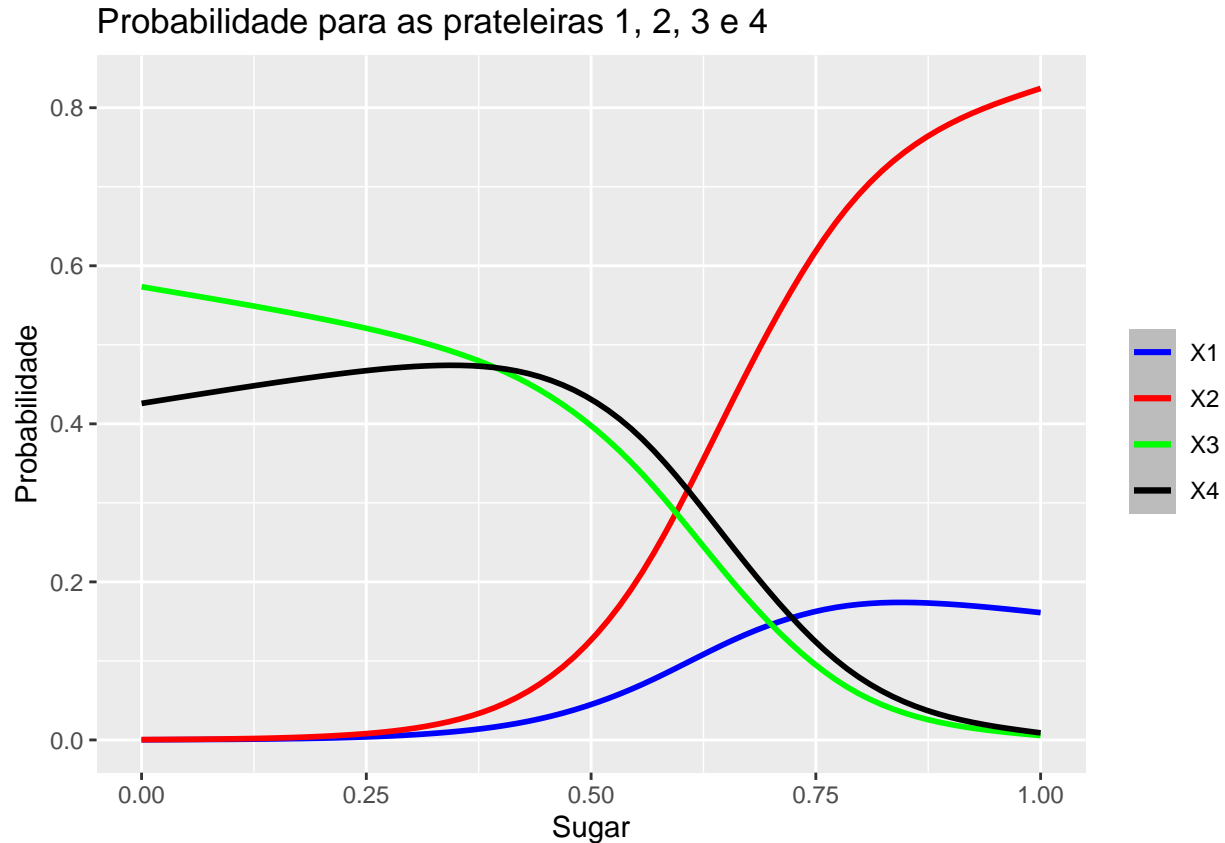
##      1      2      3      4
## 0.00 0.03 0.46 0.51
```

Para o cereal especificado, as probabilidades de estar na prateleira 3 e 4 são de 46% e 51%, respectivamente.

11-g) Construa um gráfico onde a probabilidade estimada de uma prateleira está no eixo y e o teor de açúcar está no eixo x. Use o conteúdo médio geral de gordura e sódio como os valores variáveis correspondentes no modelo. Interprete o gráfico em relação ao teor de açúcar.

```
sugar.values <- seq(from = 0, to = 1, by = 0.01)
pi.sugar <- data.frame(sugar.values, predict(object = mod.fit2,
  newdata = data.frame(sugar = sugar.values,
    fat = mean(cereal$fat),
    sodium = mean(cereal$sodium)),
  type = "probs"))
legend_data <- data.frame(label = c("X1", "X2", "X3", "X4"),
  color = c("blue", "red", "green", "black"))
require(ggplot2)
p <- ggplot(data = pi.sugar, aes(x = sugar.values))
p + geom_smooth(aes(y = X1, color = "X1"), alpha = 0.2, method = "gam") +
  geom_smooth(aes(y = X2, color = "X2"), alpha = 0.2, method = "gam") +
  geom_smooth(aes(y = X3, color = "X3"), alpha = 0.2, method = "gam") +
```

```
geom_smooth(aes(y = X4, color = "X4"), alpha = 0.2, method = "gam") +
scale_color_manual(values = c("X1" = "blue", "X2" = "red",
                              "X3" = "green", "X4" = "black"), name = NULL) +
labs(title = "Probabilidade para as prateleiras 1, 2, 3 e 4",
     x = "Sugar", y = "Probabilidade")
```



11-h) Estime as razões de chances e calcule os intervalos de confiança correspondentes para cada variável explicativa. Relacione suas interpretações com os gráficos construídos para este exercício.

```
sd.cereal <- apply (X = cereal2[, -c(1,3)], MARGIN = 2, FUN = sd)
c.value <- sd.cereal
round(c.value, 2)
```

```
## sugar sodium
## 0.27 0.23
```

```
beta.hat2 <- coefficients (mod.fit2) [1 ,2:3]
beta.hat3 <- coefficients (mod.fit2) [2 ,2:3]
beta.hat4 <- coefficients (mod.fit2) [3 ,2:3]

conf.beta <- confint (object = mod.fit2 , level = 0.95)
ci.OR2 <- exp(c.value * conf.beta [2:3 ,1:2 ,1])
ci.OR3 <- exp(c.value * conf.beta [2:3 ,1:2 ,2])
ci.OR4 <- exp(c.value * conf.beta [2:3 ,1:2 ,3])

results_df2 <- data.frame(
```



```
Odds_Ratio = round(exp(c.value * beta.hat2), 2),
Inverse_Odds_Ratio = round(1 / exp(c.value * beta.hat2), 2))
results_ci2 <- data.frame(
  CI = round(data.frame(low = ci.OR2[,1], up = ci.OR2[,2]), 2),
  Inverse_CI = round(data.frame(low = 1/ci.OR2[,2], up = 1/ci.OR2[,1]), 2))
kable(results_df2, caption = "Razão de Chances para Prateleira 2 vs. Prateleira 1")
```

Tabela 2: Razão de Chances para Prateleira 2 vs. Prateleira 1

	Odds_Ratio	Inverse_Odds_Ratio
sugar	1.38	0.73
sodium	0.04	22.99

```
kable(results_ci2, caption = "Intervalos de Confiança para Prateleira 2 vs. Prateleira 1")
```

Tabela 3: Intervalos de Confiança para Prateleira 2 vs. Prateleira 1

	CI.low	CI.up	Inverse_CI.low	Inverse_CI.up
sugar	0.14	13.92	0.07	7.34
sodium	0.00	0.77	1.30	405.50

1. As chances estimadas de Prateleira 2 vs Prateleira 1 mudam em 1.38 vezes para um aumento de 0.27 em *sugar* mantendo as outras variáveis constantes. De forma equivalente, as chances estimadas de Prateleira 2 vs Prateleira 1 mudam em 0.73 vezes para uma diminuição de 0.27 em *sugar* mantendo as outras variáveis constantes.
2. As chances estimadas de Prateleira 2 vs Prateleira 1 mudam em 0.04 vezes para um aumento de 0.23 em *sodium* mantendo as outras variáveis constantes. De forma equivalente, as chances estimadas de Prateleira 2 vs Prateleira 1 mudam em 22.99 vezes para uma diminuição de 0.23 em *sodium* mantendo as outras variáveis constantes.

```
results_df3 <- data.frame(
  Odds_Ratio = round(exp(c.value * beta.hat3), 2),
  Inverse_Odds_Ratio = round(1 / exp(c.value * beta.hat3), 2))
results_ci3 <- data.frame(
  CI = round(data.frame(low = ci.OR3[,1], up = ci.OR3[,2]), 2),
  Inverse_CI = round(data.frame(low = 1/ci.OR3[,2], up = 1/ci.OR3[,1]), 2))
kable(results_df3, caption = "Razão de Chances para Prateleira 3 vs. Prateleira 1")
```

Tabela 4: Razão de Chances para Prateleira 3 vs. Prateleira 1

	Odds_Ratio	Inverse_Odds_Ratio
sugar	0.05	18.91
sodium	0.01	189.13

```
kable(results_ci3, caption = "Intervalos de Confiança para Prateleira 3 vs. Prateleira 1")
```

Tabela 5: Intervalos de Confiança para Prateleira 3 vs. Prateleira 1

	CI.low	CI.up	Inverse_CI.low	Inverse_CI.up
sugar	0.01	0.54	1.84	194.01
sodium	0.00	0.14	7.15	5000.89

3. As chances estimadas de Prateleira 3 vs Prateleira 1 mudam em 0.05 vezes para um aumento de 0.27 em *sugar* mantendo as outras variáveis constantes. De forma equivalente, as chances estimadas de Prateleira 3 vs Prateleira 1 mudam em 18.91 vezes para uma diminuição de 0.27 em *sugar* mantendo as outras variáveis constantes.
4. As chances estimadas de Prateleira 3 vs Prateleira 1 mudam em 0.01 vezes para um aumento de 0.23 em *sodium* mantendo as outras variáveis constantes. De forma equivalente, as chances estimadas de Prateleira 3 vs Prateleira 1 mudam em 189.13 vezes para uma diminuição de 0.23 em *sodium* mantendo as outras variáveis constantes.

```
results_df4 <- data.frame(
  Odds_Ratio = round(exp(c.value * beta.hat4), 2),
  Inverse_Odds_Ratio = round(1 / exp(c.value * beta.hat4), 2))
results_ci4 <- data.frame(
  CI = round(data.frame(low = ci.OR4[,1], up = ci.OR4[,2]), 2),
  Inverse_CI = round(data.frame(low = 1/ci.OR4[,2], up = 1/ci.OR4[,1]), 2))
kable(results_df4, caption = "Razão de Chances para Prateleira 4 vs. Prateleira 1")
```

Tabela 6: Razão de Chances para Prateleira 4 vs. Prateleira 1

	Odds_Ratio	Inverse_Odds_Ratio
sugar	0.06	15.43
sodium	0.01	174.66

```
kable(results_ci4, caption = "Intervalos de Confiança para Prateleira 4 vs. Prateleira 1")
```

Tabela 7: Intervalos de Confiança para Prateleira 4 vs. Prateleira 1

	CI.low	CI.up	Inverse_CI.low	Inverse_CI.up
sugar	0.01	0.66	1.52	156.24
sodium	0.00	0.15	6.67	4571.35

5. As chances estimadas de Prateleira 4 vs Prateleira 1 mudam em 0.06 vezes para um aumento de 0.27 em *sugar* mantendo as outras variáveis constantes. De forma equivalente, as chances estimadas de Prateleira 4 vs Prateleira 1 mudam em 15.43 vezes para uma diminuição de 0.27 em *sugar* mantendo as outras variáveis constantes.
6. As chances estimadas de Prateleira 4 vs Prateleira 1 mudam em 0.01 vezes para um aumento de 0.23 em *sodium* mantendo as outras variáveis constantes. De forma equivalente, as chances estimadas de Prateleira 4 vs Prateleira 1 mudam em 174.66 vezes para uma diminuição de 0.23 em *sodium* mantendo as outras variáveis constantes.

As interpretações 1, 3 e 5, apresentadas acima, estão de acordo com o gráfico de Probabilidade para as prateleiras de acordo com os valores de *sugar*, onde a razão de chances, comparada com a Prateleira 1, é muito maior para as Prateleiras 3 e 4 para valores pequenos de *sugar*, e muito maior para a Prateleira 2 para valores altos de *sugar*, exatamente como o gráfico mostra.

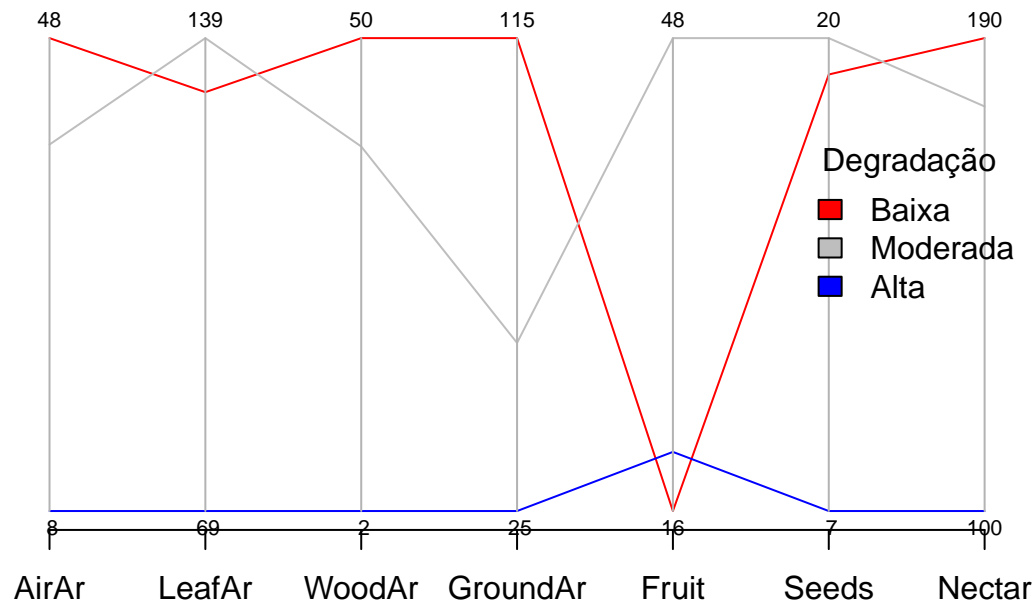
19) Realize uma análise para determinar quais guildas são mais afetadas ou menos afetadas pela degradação do habitat.

```
df <- data.frame(
  Baixa = c(48, 131, 50, 115, 16, 19, 190),
  Moderada = c(39, 139, 39, 57, 48, 20, 177),
  Alta = c(8, 69, 2, 25, 20, 7, 100)
)
Guildas <- c("AirAr", "LeafAr", "WoodAr", "GroundAr", "Fruit", "Seeds", "Nectar")
kable(cbind(Guildas, df), caption = "Tabela cruzada de Aves Equatorianas capturadas por
Degradação de Habitat e Guilda de Fonte de Alimento")
```

Tabela 8: Tabela cruzada de Aves Equatorianas capturadas por Degradação de Habitat e Guilda de Fonte de Alimento

Guildas	Baixa	Moderada	Alta
AirAr	48	39	8
LeafAr	131	139	69
WoodAr	50	39	2
GroundAr	115	57	25
Fruit	16	48	20
Seeds	19	20	7
Nectar	190	177	100

```
df_t <- t(df)
colnames(df_t) <- Guildas
parcoord(x = df_t, col = c("red", "grey", "blue"), var.label = TRUE)
legend("topright", legend = c("Baixa", "Moderada", "Alta"),
  fill = c("red", "grey", "blue"), bty = "n", title = "Degradação", x = 5.75, y = .8)
```



Conforme o gráfico de coordenadas paralelas mostra acima, é possível perceber uma correlação em que os ambientes com maior degradação tendem a ter menos aves capturadas para todas as Guildas, exceto a *Fruit*. Essa observação sugere que a degradação ambiental está associada a uma redução na diversidade de aves nas diferentes guildas ecológicas.

Conforme sugere o enunciado, as quatro primeiras guildas são artrópodes, em contraponto com as 3 últimas guildas. A seguir, será adicionado uma variável indicativa representando quais Guildas são *Arth* e a aplicação do teste Anova para modelos encaixados, que testa se a inclusão dela é necessária ou não no modelo.

```
data <- data.frame(Guilda = rep(Guildas, 3),
  Arth = rep(c(rep(1,4),rep(0,3)),3),
  Degracao = c(rep("Baixa",7), rep("Moderada",7), rep("Alta",7)),
  Contagem = c(df[,1], df[,2], df[,3]))
data$Degracao <- factor(data$Degracao, levels = c("Baixa", "Moderada", "Alta"))
data$Arth <- factor(data$Arth)

mod.fit.ord.arth <- polr(Degracao ~ Contagem + Guilda + Arth, data, method = "logistic")
mod.fit.ord.guilha <- polr(Degracao ~ Contagem + Guilda, data, method = "logistic")
summary(mod.fit.ord.guilha)
```

```
## Call:
## polr(formula = Degracao ~ Contagem + Guilda, data = data, method = "logistic")
##
## Coefficients:
##              Value Std. Error  t value
## Contagem      -0.13017    0.05202 -2.502460
## GuildaFruit     0.06800    1.84332  0.036889
## GuildaGroundAr  3.24119    2.28506  1.418428
## GuildaLeafAr   11.79249    5.32367  2.215107
## GuildaNectar   17.92732    7.70215  2.327573
## GuildaSeeds    -2.25918    1.97603 -1.143295
## GuildaWoodAr   -0.01293    1.78664 -0.007236
##
## Intercepts:
##              Value Std. Error t value
## Baixa|Moderada -5.7148    2.4110  -2.3703
## Moderada|Alta  -2.9432    2.1124  -1.3933
##
## Residual Deviance: 27.49513
## AIC: 45.49513
```

```
anova(mod.fit.ord.arth, mod.fit.ord.guilha)
```

```
## Likelihood ratio tests of ordinal regression models
##
## Response: Degracao
##              Model Resid. df Resid. Dev  Test    Df LR stat. Pr(Chi)
## 1 Contagem + Guilda + Arth          12    27.49513
## 2      Contagem + Guilda          12    27.49513 1 vs 2      0      0      1
```

```
Anova(mod.fit.ord.guilha)
```

```
## Analysis of Deviance Table (Type II tests)
##
## Response: Degracao
##              LR Chisq Df Pr(>Chisq)
## Contagem    18.647  1  1.573e-05 ***
## Guilda      15.937  6    0.0141 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com base na análise do teste ANOVA para modelos encaixados, a inclusão da variável *Arth* não se mostrou

necessária no modelo, indicando que não há diferença estatisticamente significativa entre as guildas ecológicas que são do tipo *Arth* com as que não são.

```
pi.hat <- predict(mod.fit.ord.guilde,type = "probs")
pi.hat.class <- predict(mod.fit.ord.guilde,type = "class")
comparacao <- data.frame(pi.hat,Degradacao=data$Degradacao,pi.hat.class)
kable(comparacao)
```

Baixa	Moderada	Alta	Degradacao	pi.hat.class
0.6302411	0.3343551	0.0354038	Baixa	Baixa
0.3880943	0.5221244	0.0897813	Baixa	Moderada
0.6913663	0.2814652	0.0271685	Baixa	Baixa
0.9975599	0.0022871	0.0001530	Baixa	Baixa
0.0241244	0.2591102	0.7167654	Baixa	Alta
0.2724105	0.5844200	0.1431695	Baixa	Moderada
0.7483315	0.2310629	0.0206056	Baixa	Baixa
0.3456406	0.5484649	0.1058945	Moderada	Moderada
0.6424494	0.3239050	0.0336456	Moderada	Baixa
0.3485704	0.5467529	0.1046766	Moderada	Moderada
0.1770550	0.5976740	0.2252711	Moderada	Moderada
0.6142595	0.3479397	0.0378008	Moderada	Baixa
0.2989592	0.5731100	0.1279309	Moderada	Moderada
0.3537844	0.5436643	0.1025513	Moderada	Moderada
0.0092535	0.1206496	0.8700968	Alta	Alta
0.0001983	0.0029615	0.9968402	Alta	Alta
0.0043142	0.0604593	0.9352265	Alta	Alta
0.0033289	0.0473540	0.9493171	Alta	Alta
0.0399468	0.3594924	0.6005609	Alta	Alta
0.0728010	0.4837559	0.4434431	Alta	Moderada
0.0000243	0.0003639	0.9996118	Alta	Alta

```
require(dplyr)
comparacao <- comparacao %>% mutate(igual=ifelse(Degradacao==pi.hat.class,1,0))
sum(comparacao$igual)/nrow(comparacao)
```

```
## [1] 0.7142857
```

O modelo acertou a predição para 71,4% das observações, o que aparenta ser um resultado condizente com o problema, dado que se trata de um exercício com poucas variáveis explicativas.