

An

Doriedson

09-11-2024

Os dados

Trata-se de um conjunto extenso, com mais de 20 mil observações e 680 variáveis, sobre receitas culinárias do site *Epicurious*. Houve uma filtragem inicial no conjunto, restaram apenas 6 variáveis explicativas, as consideradas mais importantes: nota, a quantidade de calorias, de proteína, de gordura e de sódio. Há também o nome da receita e a variável resposta, que no caso é binária: trata-se de uma sobremesa? No site em questão, encontra-se categorias de pratos como *café da manhã*, *almoço*, *jantar*, *drinks* e, naturalmente, o objeto de interesse do trabalho, as *sobremesas*.

Em seguida, foi feita a exclusão de linhas que possuíam valores faltantes. Por fim, houve a retirada de valores absurdos (descrita mais detalhadamente abaixo). O conjunto final possui 15706 observações e 7 variáveis.

Obs.: a variável calorias está na unidade 'kcal'.

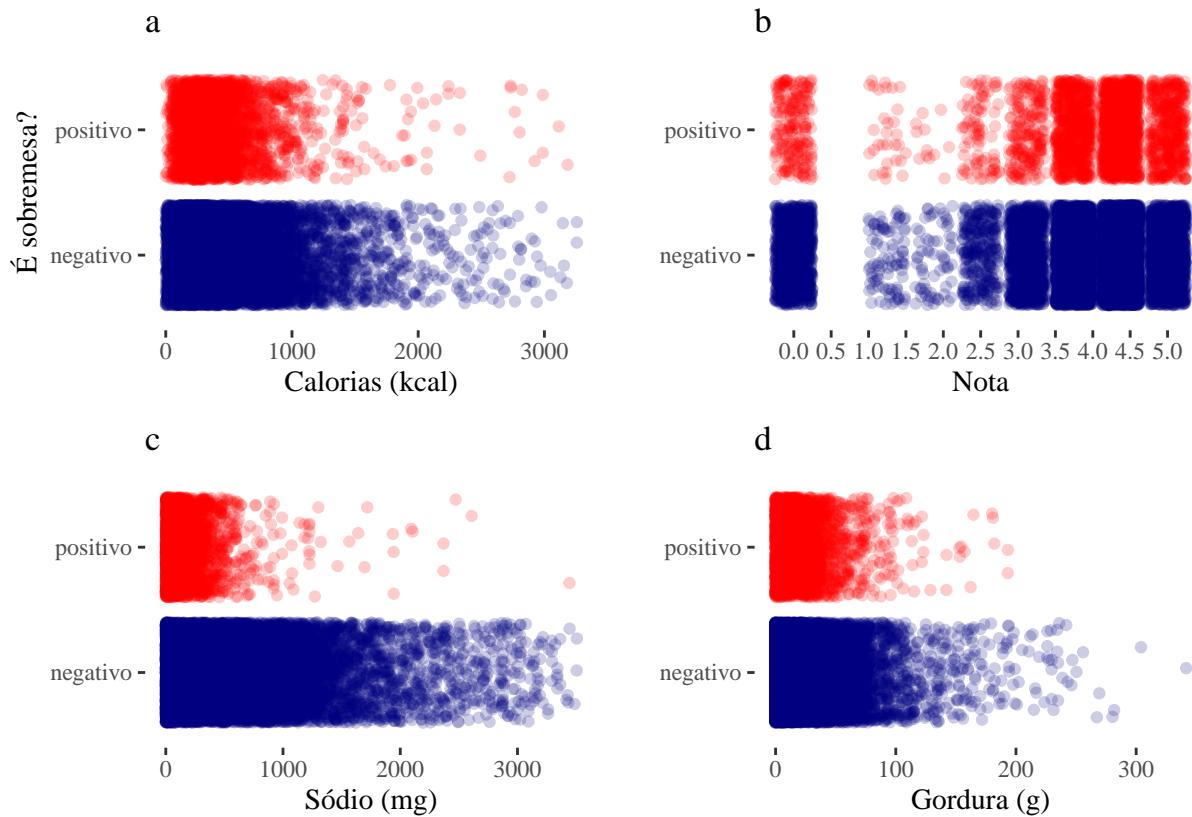
```
## [1] "title"          "rating"         "calories"       "protein"  
## [5] "fat"            "sodium"         "dados$dessert"  
  
##                                     titulo  nota calorias protein gordura  
## 1           Lentil, Apple, and Turkey Wrap 2.500    426     30      7  
## 2 Boudin Blanc Terrine with Red Onion Confit 4.375    403     18     23  
## 3           Potato and Fennel Soup Hodge 3.750    165      6      7  
## 5           Spinach Noodle Casserole 3.125    547     20     32  
## 6           The Best Blts 4.375    948     19     79  
## 9           Korean Marinated Beef 4.375    170      7     10  
##   sodio  sobremesa  
## 1   559      0  
## 2  1439      0  
## 3   165      0  
## 5   452      0  
## 6  1042      0  
## 9  1272      0  
  
## [1] 15864      7
```

Observando o percentil 99 tem-se que 99% dos valores são menores ou iguais a 3257 kcal. Tomarei como limite das observações de caloria, pois é improvável/inviável que receitas ultrapassem de maneira tão acentuada esse valor; receitas com milhões de kcal são claramente erros nessa base de dados. Foi feito um tratamento semelhante para a variável *sódio*.

```
## [1] 0.1872977
```

Análise descritiva

Como visto no gráfico 'b', a variável nota não aparenta ser influenciada pelo tipo da receita.



Modelo base

O modelo principal será analisado da seguinte maneira: o conjunto de dados selecionados será dividido em *conjunto_treino* (70%) e *conjunto_teste* (30%)

```
##  
## Call:  
## glm(formula = sobremesa ~ nota + calorias + proteina + gordura +  
##       sodio, family = binomial(link = "logit"), data = df1_treino)  
##  
## Coefficients:  
##              Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.0500559  0.0942438 -21.753 < 2e-16 ***  
## nota         0.1579778  0.0219800   7.187 6.61e-13 ***  
## calorias     0.0082228  0.0002679  30.695 < 2e-16 ***  
## proteina    -0.1687192  0.0073112 -23.077 < 2e-16 ***  
## gordura     -0.0364663  0.0029229 -12.476 < 2e-16 ***  
## sodio        -0.0032049  0.0001736 -18.466 < 2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## (Dispersion parameter for binomial family taken to be 1)  
##  
## Null deviance: 10510.4 on 10811 degrees of freedom  
## Residual deviance: 7046.7 on 10806 degrees of freedom  
## AIC: 7058.7  
##  
## Number of Fisher Scoring iterations: 8  
  
## Start:  AIC=7058.69  
## sobremesa ~ nota + calorias + proteina + gordura + sodio  
  
## Warning: glm.fit: probabilidades ajustadas numericamente 0 ou 1 ocorreu  
## Warning: glm.fit: probabilidades ajustadas numericamente 0 ou 1 ocorreu  
## Warning: glm.fit: probabilidades ajustadas numericamente 0 ou 1 ocorreu  
## Warning: glm.fit: probabilidades ajustadas numericamente 0 ou 1 ocorreu  
  
##              Df Deviance    AIC  
## <none>          7046.7 7058.7  
## - nota          1   7101.2 7111.2  
## - gordura       1   7204.3 7214.3  
## - sodio          1   7536.6 7546.6  
## - calorias       1   8290.8 8300.8  
## - proteina       1   8530.3 8540.3  
  
##  
## Call:  glm(formula = sobremesa ~ nota + calorias + proteina + gordura +  
##       sodio, family = binomial(link = "logit"), data = df1_treino)  
##  
## Coefficients:  
## (Intercept)      nota    calorias    proteina    gordura    sodio  
## -2.050056     0.157978   0.008223   -0.168719   -0.036466   -0.003205  
##
```

```

## Degrees of Freedom: 10811 Total (i.e. Null); 10806 Residual
## Null Deviance: 10510
## Residual Deviance: 7047 AIC: 7059

```

Teste com as outras funções de ligação.

```

## Warning: glm.fit: probabilidades ajustadas numericamente 0 ou 1 ocorreu

##
## Call:
## glm(formula = sobremesa ~ nota + calorias + proteina + gordura +
##      sodio, family = binomial(link = "probit"), data = df1_treino)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.167e+00 5.325e-02 -21.920 < 2e-16 ***
## nota        8.486e-02 1.264e-02   6.715 1.88e-11 ***
## calorias    4.530e-03 1.464e-04  30.940 < 2e-16 ***
## proteina    -8.965e-02 3.871e-03 -23.161 < 2e-16 ***
## gordura     -2.200e-02 1.651e-03 -13.327 < 2e-16 ***
## sodio       -1.785e-03 9.335e-05 -19.116 < 2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 10510.4 on 10811 degrees of freedom
## Residual deviance: 7092.4 on 10806 degrees of freedom
## AIC: 7104.4
##
## Number of Fisher Scoring iterations: 13

## Warning: glm.fit: algoritmo não convergiu

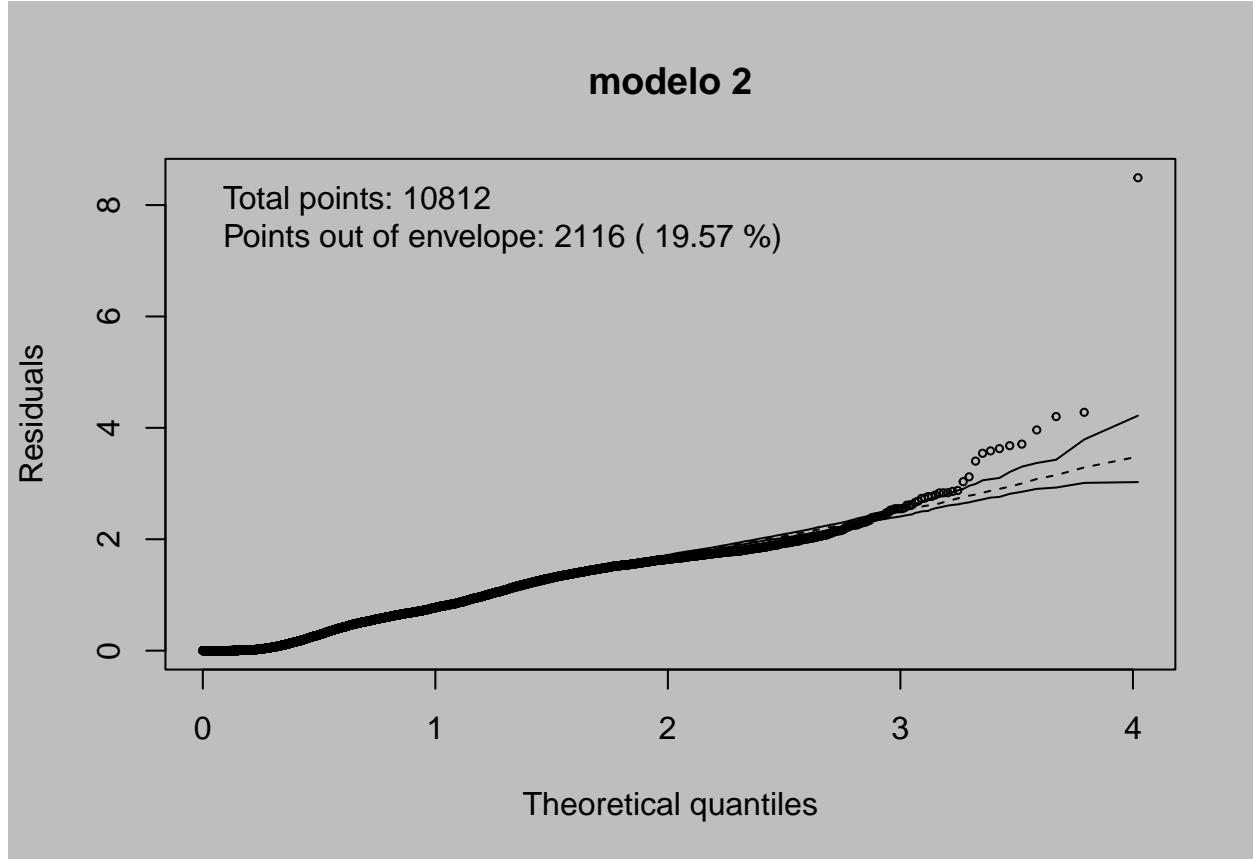
## Warning: glm.fit: probabilidades ajustadas numericamente 0 ou 1 ocorreu

##
## Call:
## glm(formula = sobremesa ~ nota + calorias + proteina + gordura +
##      sodio, family = binomial(link = "cloglog"), data = df1_treino)
##
## Coefficients:
##             Estimate Std. Error     z value Pr(>|z|)
## (Intercept) -3.572e+14 2.121e+06 -1.684e+08 <2e-16 ***
## nota        3.928e+13 5.076e+05  7.739e+07 <2e-16 ***
## calorias    2.279e+12 4.680e+03  4.871e+08 <2e-16 ***
## proteina    -5.547e+13 3.736e+04 -1.485e+09 <2e-16 ***
## gordura     -6.404e+12 5.404e+04 -1.185e+08 <2e-16 ***
## sodio       -1.185e+12 1.329e+03 -8.921e+08 <2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##

```

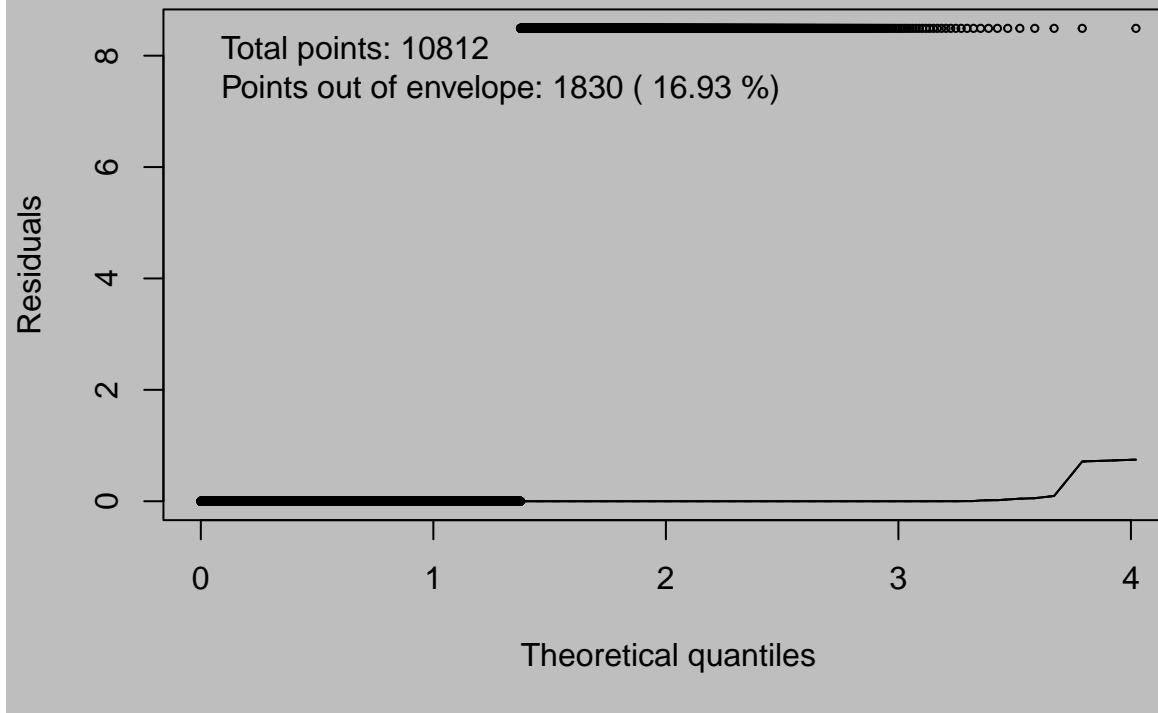
```
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 10510 on 10811 degrees of freedom
## Residual deviance: 131920 on 10806 degrees of freedom
## AIC: 131932
##
## Number of Fisher Scoring iterations: 25

## Binomial model
```



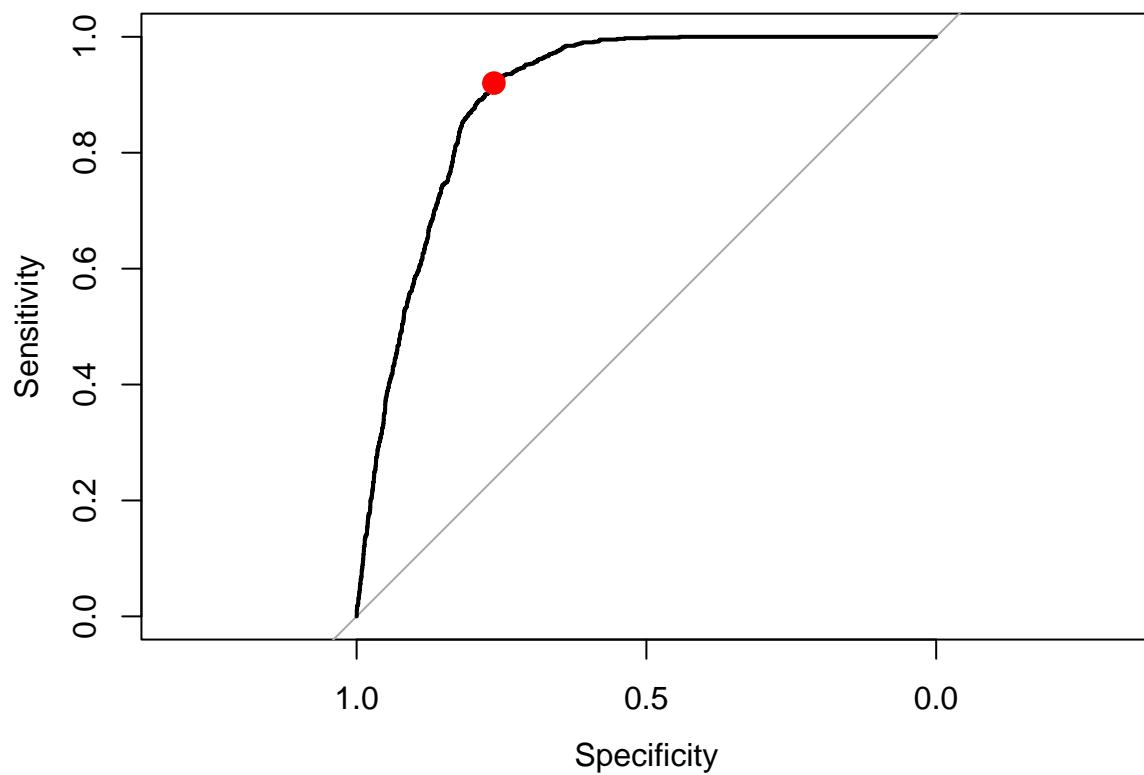
```
## Binomial model
```

modelo 3



```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```



```

## [1] 0.2011471

# tidymodels e tidyverse
df1_teste$predito <- ifelse(df1_teste$prob_pred < optimal_coords$threshold, 0, 1)

table(df1_teste$sobremesa)

##
##      0      1
## 3795  839

table(df1_teste$predito)

##
##      0      1
## 2963 1671

# Para o gráfico apenas
conf.m = caret::confusionMatrix(data = as.factor(df1_teste$predito),
                                 reference = as.factor(df1_teste$sobremesa))
df = as.data.frame(conf.m$table)

ggplot(df, aes(Prediction, Reference, fill=Freq)) +
  geom_tile() +

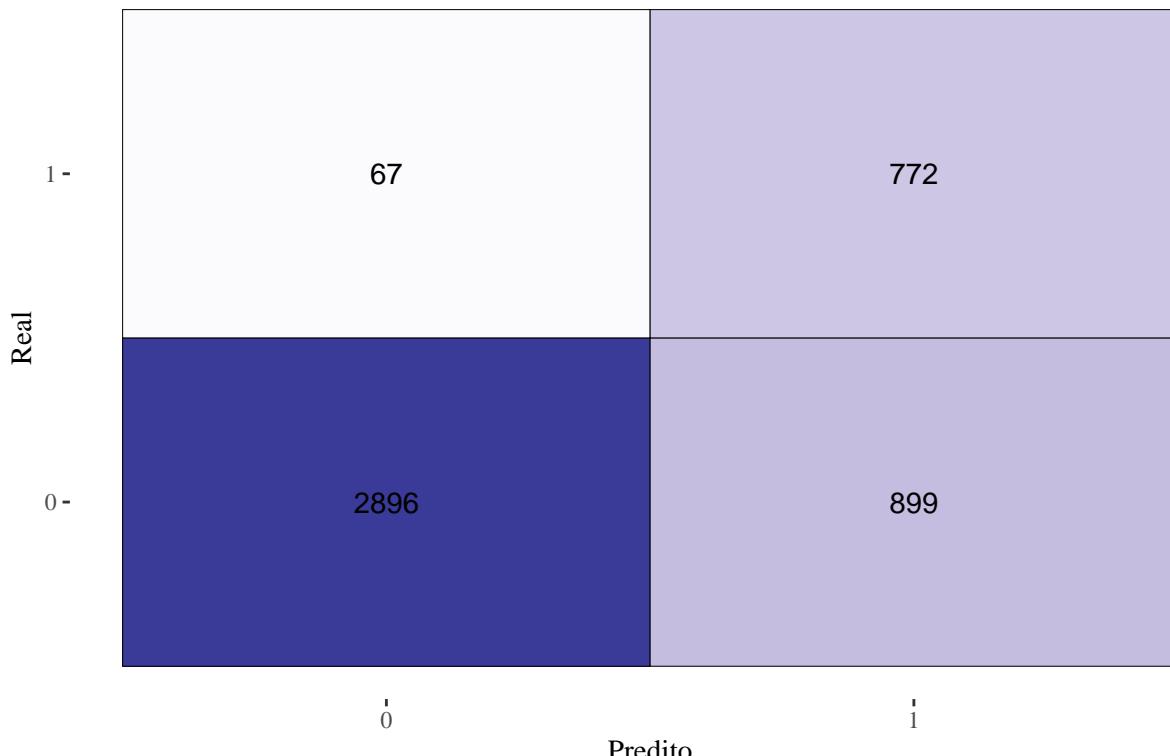
```

```

geom_tile(colour = 'black') +
ggthemes::theme_tufte() +
theme(legend.position = 'none') +
geom_text(aes(label = Freq)) + # write the values
scale_fill_gradient2(low = scales::muted("white"),
                      high = scales::muted("midnightblue")) +
labs(title = "Matriz de confusao", x="Predito", y = "Real")

```

Matriz de confusao



```
confundir(df1_teste$sobremesa, df1_teste$predito)
```

```

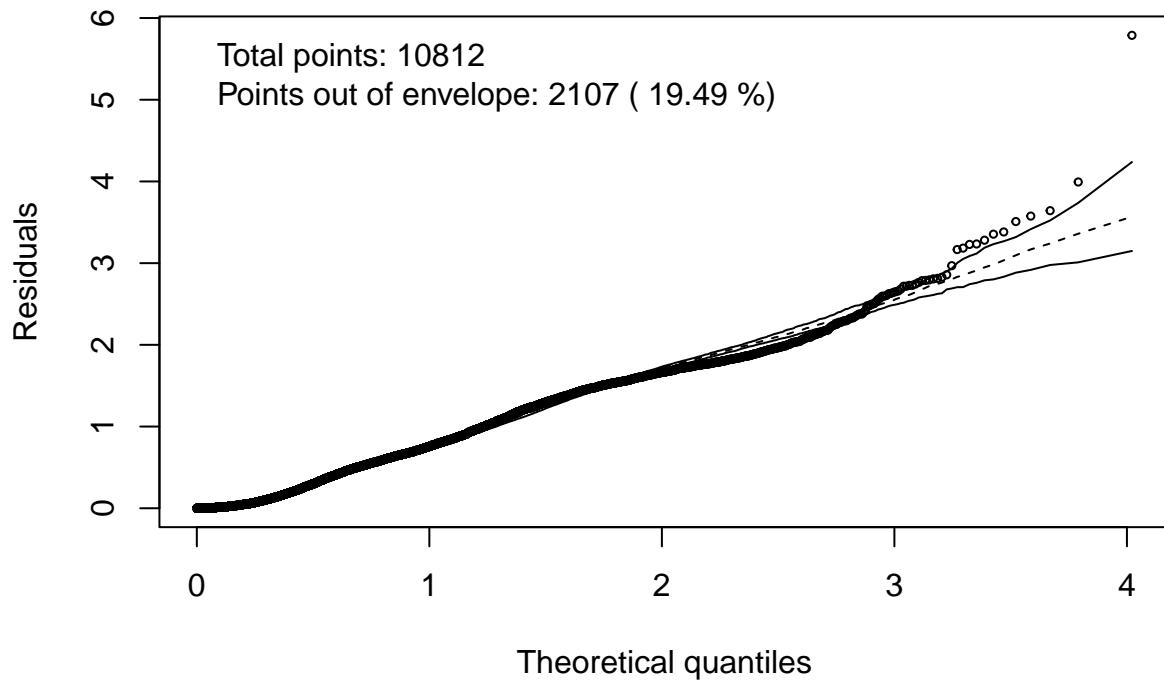
##   Evento.de.interesse Sensibilidade Especificidade Acuracidade Precisao
## 1          0.1810531      0.920143      0.7631094    0.7915408 0.4619988

```

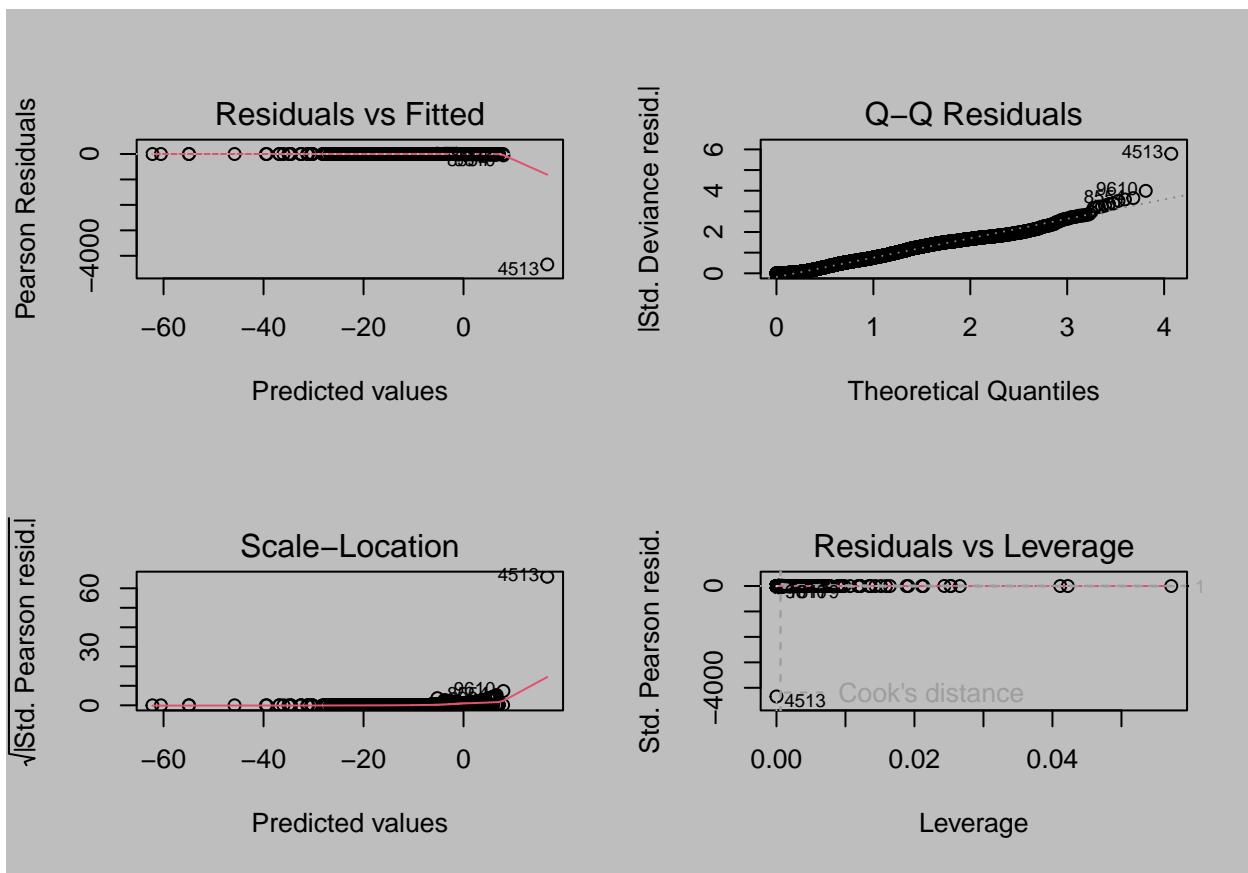
Destacar no texto qual tipo de erro mais ocorreu!

Análise de diagnóstico dos resíduos

```
## Binomial model
```



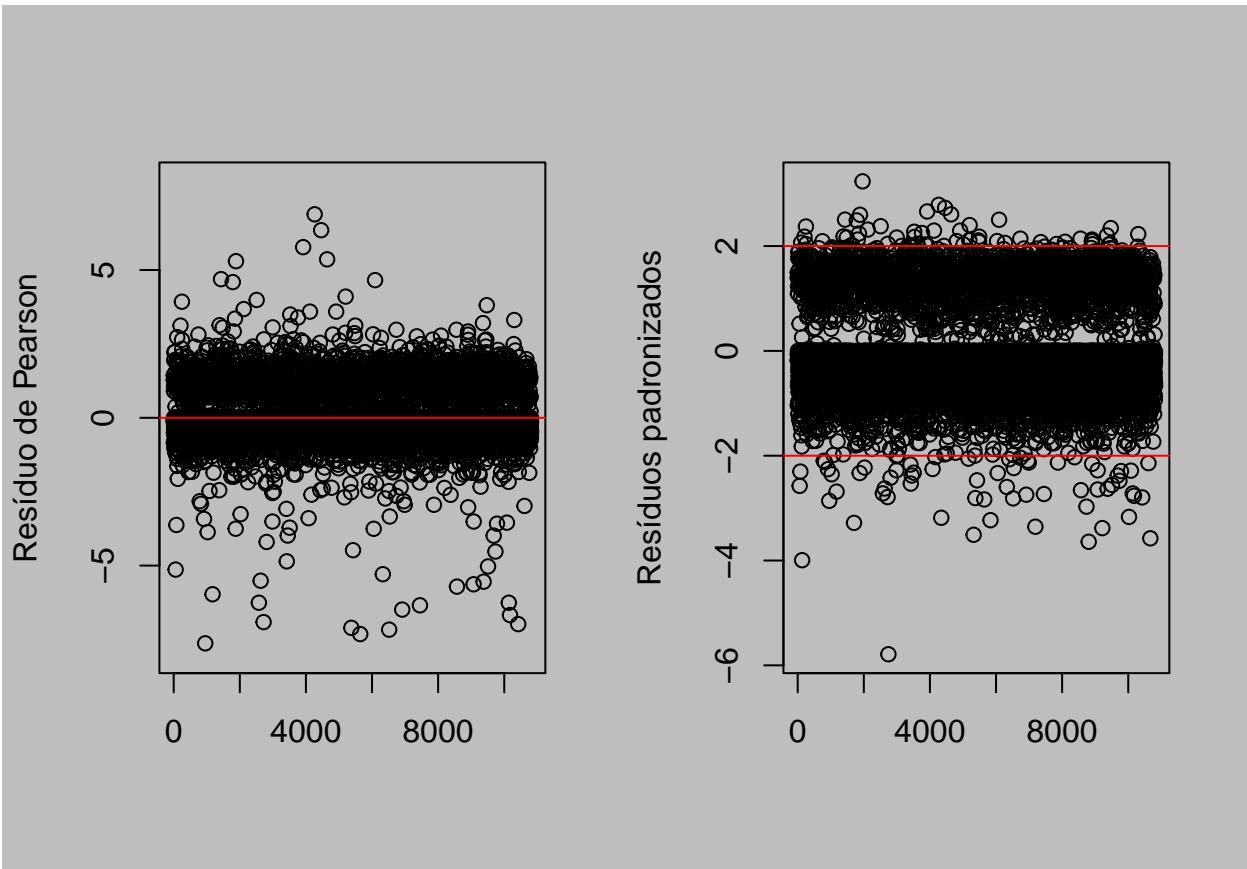
```
## [1] 0.1948761
```

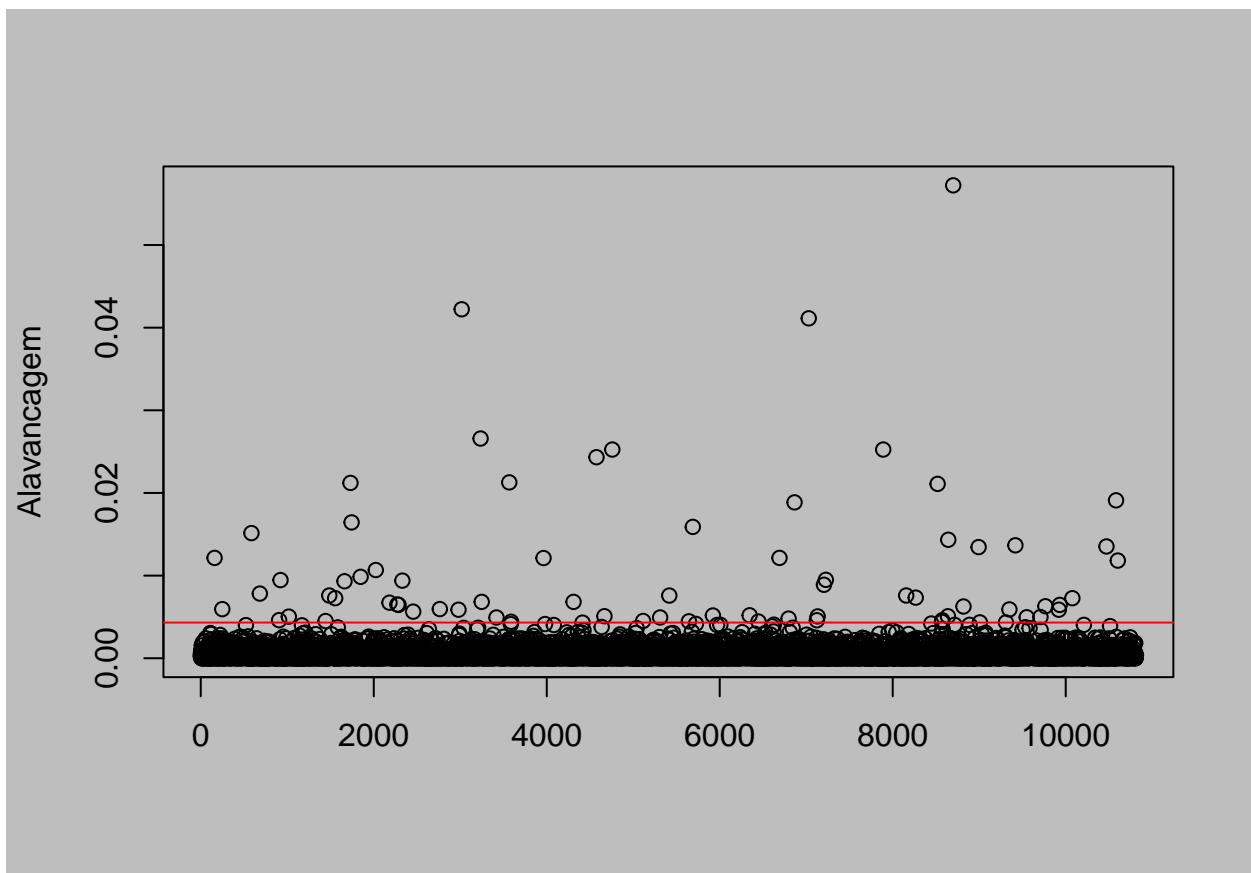


```
##
## Os outliers foram:
##
```

	titulo	nota	calorias	proteina
## 12202	Boston Brown Bread	5.000	255	6
## 3268	Haricots Verts, Roasted Fennel, and Shallots	4.375	298	6
## 13300	Molasses-Baked Onions	4.375	229	6
##	gordura sadio sobremesa status			
## 12202	4 635	0	negativo	
## 3268	18 96	0	negativo	
## 13300	14 357	0	negativo	

```
##
## Box-Ljung test
##
## data: modelo$residuals
## X-squared = 9.249e-05, df = 1, p-value = 0.9923
```





Parte final (teste com outros tamanhos amostrais)

Para esse conjunto de dados, realmente é necessário utilizar todos dados selecionados para ajustar um modelo satisfatório?

Tabulação das seguintes métricas para o mesmo modelo (mesmas variáveis explicativas): acurácia, precisão, sensibilidade, AIC, BIC. Deve conter qual o modelo final; qual o tamanho amostral do conjunto de treino e do de teste (proporção 70-30 sempre).

X	TamanhoEvento.de.interesse	Acuracidade	Sensibilidade	Especificidade	Precisão	Pontos.fora..envelope.
2	200	0.1956667	0.8163333	0.9486423	0.7841083	0.5284392
3	400	0.1909167	0.8090833	0.9369069	0.7788330	0.5055441
4	800	0.1855000	0.8035417	0.9141745	0.7782158	0.4881748
5	2000	0.1892833	0.7994000	0.9155959	0.7723026	0.4869713
6	4000	0.1880500	0.7939583	0.9170808	0.7654115	0.4764996
7	10000	0.1868433	0.7944167	0.9075442	0.7684313	0.4742298

Referências

<https://rpubs.com/mpfoley73/527573>

<https://www.kaggle.com/code/rtatman/regression-challenge-day-1>

[https://www.kaggle.com/code/rtatman/datasets-for-regression-analysis#Poisson-regression-\(predicting-a-count-value\)](https://www.kaggle.com/code/rtatman/datasets-for-regression-analysis#Poisson-regression-(predicting-a-count-value))

<https://www.kaggle.com/datasets/hugodarwood/epirecipes?resource=download>