



**Programa Aulas Abertas - Departamento de Matemática Aplicada e Estatística - ICMC - USP**

SME0823 Modelos de Regressão e Aprendizado Supervisionado II

## **Modelos de regressão para classificar e-mails como spam**

**(Uma introdução à disciplina)**

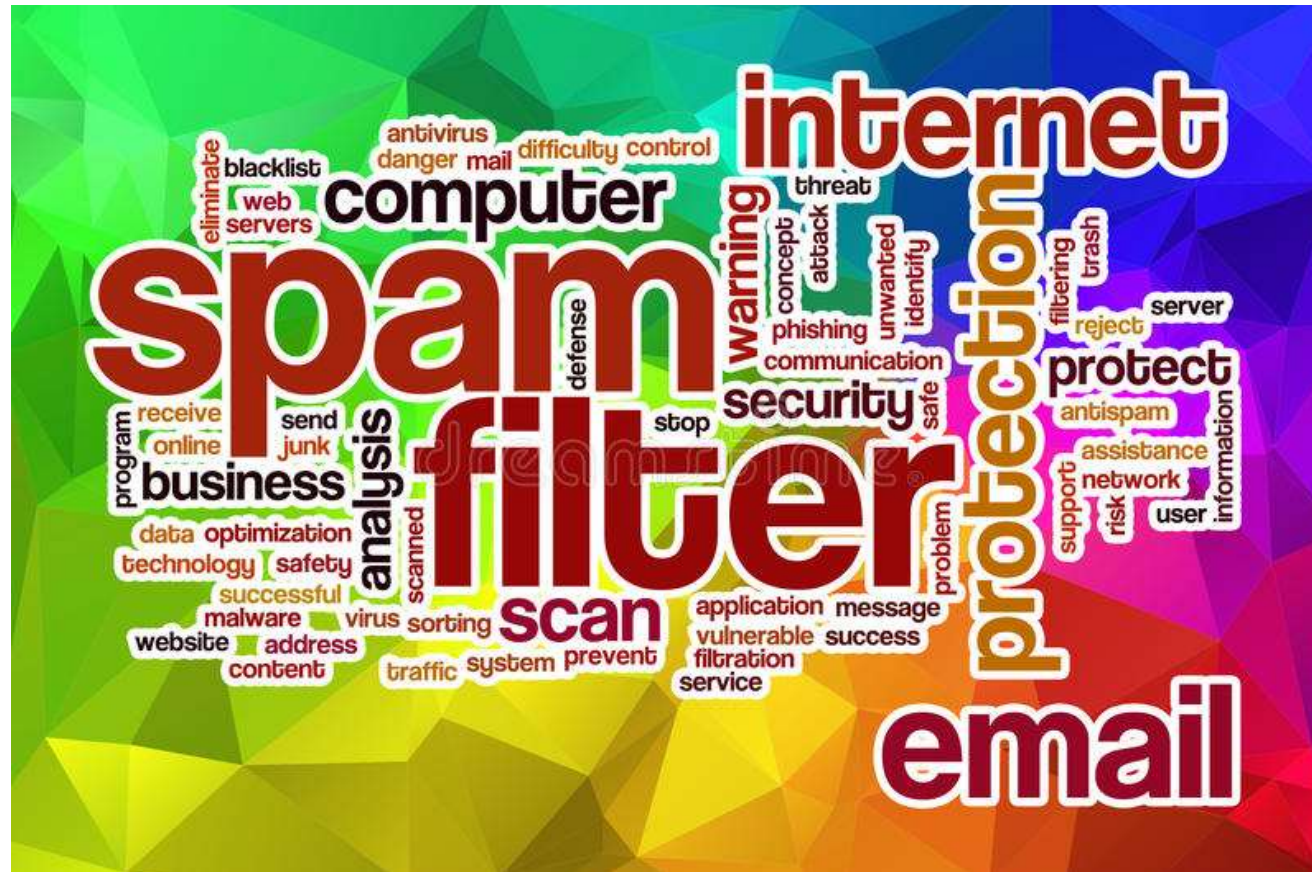
**Prof. Jorge Luis Bazán**

<https://sites.icmc.usp.br/jlbazan/>

<https://jorgeluisbazan.weebly.com>

## **Como citar este documento**

Bazán, J. L. (2021). Modelos de regressão para classificar e-mails como spam. Uma introdução à disciplina SME0823 Modelos de Regressão e Aprendizado Supervisionado II. Programa Aulas Abertas - Departamento de Matemática Aplicada e Estatística - ICMC – USP. 16 de agosto de 2020. Disponível em [Download](#).



## **Conteúdos**

- 1. Introdução**
- 2 Descrição do Problema e os dados**
- 3. Análise descritiva das variáveis do modelo**
- 4. Formulação do modelo (de Classificação)**
- 5 Proposta de modelos alternativos usando diferentes ligações**
- 6. Análise de seleção de variáveis para o modelo proposto**
- 7. Análise diagnostica para identificar pontos problemáticos no modelo reduzido**
- 8. Modelo final e interpretação de parâmetros**
- 9. Conclusões acerca da metodologia**
- 10. Outras considerações do problema**
- 11. Referenciais**

## 1. Introdução

O principal objetivo desta apresentação é analisar um conjunto de dados de Spam usando as ferramentas que serão desenvolvidas na disciplina SME0823 – Modelos de Regressão e Aprendizado Supervisionado II do curso de Bachelarelado em Estatística do ICMC - USP oferecida no segundo semestre do ano 2020.

Para isso seguiremos o seguinte roteiro

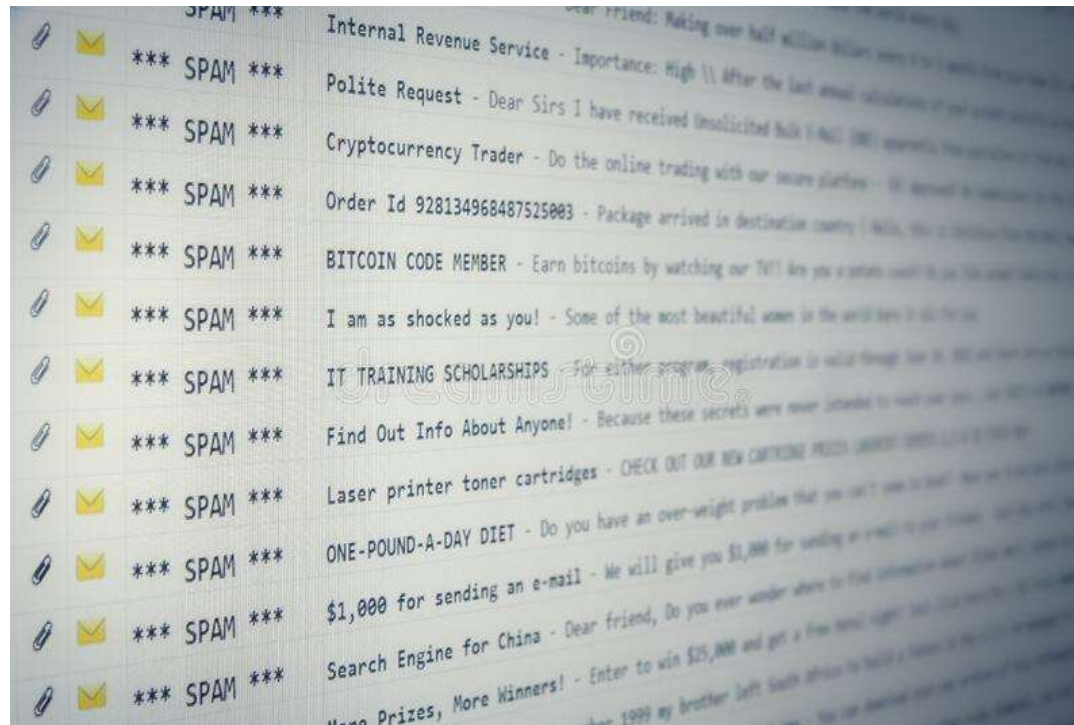
- Descrição do Problema e os dados
- Análise descritiva das variáveis do modelo
- Formulação do modelo
- Proposta de modelos alternativos usando diferentes ligações. (Escolha do modelo usando diferentes critérios)
- Para o modelo escolhido, análise de seleção de variáveis. (Escolha das variáveis significativas para o modelo reduzido)
- Para o modelo reduzido, desenvolver uma análise diagnostica para identificar pontos problemáticos
- Propor o modelo final e interpretar os parâmetros do modelo
- Avaliar o modelo final
- Formular conclusões acerca do modelo adotado, a metodologia e melhoras do modelo.

O banco de dados fornecido é referente à classificação de e-mail em spam e não spam.

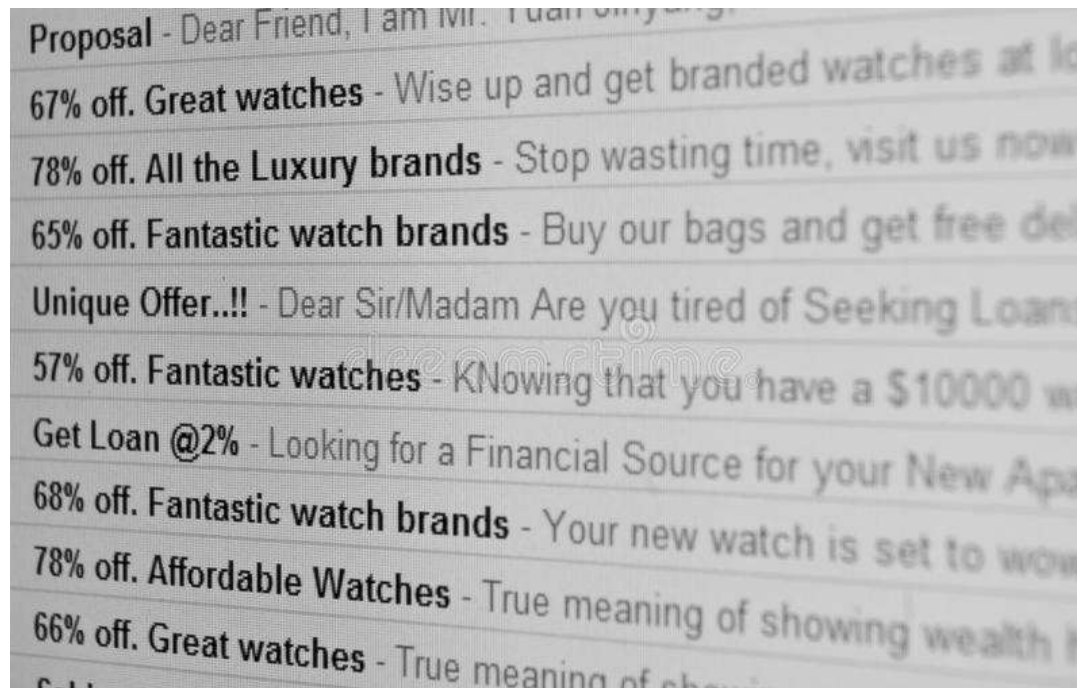


<https://www.dreamstime.com/best-stock-photos>

## 2. Descrição do Problema e os dados



<https://www.dreamstime.com/best-stock-photos>



<https://www.dreamstime.com/best-stock-photos>





<https://www.dreamstime.com/best-stock-photos>

No repositório (<https://archive.ics.uci.edu/ml/datasets/Spambase>) são disponibilizados o conjunto de dados spam. Os dados são também disponíveis em <https://www.r-bloggers.com/build-a-spam-filter-with-r/>

**Spambase Data Set**  
 Download: [Data Folder](#), [Data Set Description](#)

**Abstract:** Classifying Email as Spam or Non-Spam

<b>Data Set Characteristics:</b>	Multivariate	<b>Number of Instances:</b>	4601	<b>Area:</b>	Computer
<b>Attribute Characteristics:</b>	Integer, Real	<b>Number of Attributes:</b>	57	<b>Date Donated</b>	1999-07-01
<b>Associated Tasks:</b>	Classification	<b>Missing Values?</b>	Yes	<b>Number of Web Hits:</b>	521856

**Source:**

Creators:

Mark Hopkins, Erik Reeber, George Forman, Jaap Suermondt  
 Hewlett-Packard Labs, 1501 Page Mill Rd., Palo Alto, CA 94304

Donor:

O objetivo deste conjunto de dados é explicar uma variável resposta spam que indica se um e-mail pode ser considerado sem spam (classe 0) ou com spam (classe 1). Sendo que dos 4601 e-mails, 1813 foram considerados spam.

Três conjunto de diferentes covariáveis contínuas associadas ao texto foram consideradas. As primeiras 48 correspondem a frequências de algumas. As seguintes 6 correspondem as frequências de alguns, e por último, as 3 últimas correspondem a frequência de uso de algumas.

Para simplificar, um índice para cada conjunto foi elaborado de modo a ter média 0 e desvio padrão 1. Eles são o `Indice_word`, `Indice_char` e `Indice_capital` respectivamente. A maior valor do índice, indica maior frequência usando palavras chaves, caracteres chaves o uso de maiúsculas nos textos dos e-mails.

(A metodologia de construção desses índices está baseada em conhecimentos das disciplinas (SME0803 Visualização e Exploração de Dados, SME0820 Modelos de Regressão e Aprendizado Supervisionado I, SME0806 Estatística Computacional e não são mostradas aqui)

Assim A base de dados possui 4 variáveis quantitativas (discretas e contínuas) e 4601 observações.

Neste caso, temos como variável resposta a variável spam (discreta) que indica se um e-mail pode ser considerado sem spam (classe 0) ou com spam (classe 1), já para as covariáveis contínuas temos: `Indice_word`, `Indice_char` e `Indice_capital`.

Um exemplo dos dados

```
## e-mail spam Indice_word Indice_char Indice_capital
## 1 1 1 -0.2931382 0.66430579 0.7153379
## 2 2 1 0.4272653 0.61482211 1.5186001
## 3 3 1 0.4034456 0.48923709 2.1141300
## 4 4 1 -0.2017293 -0.04493053 0.4606454
## 5 5 1 -0.2017293 -0.05307403 0.4606454
## 6 6 1 -0.9329178 -0.15498843 -0.3321171

## 'data.frame': 4601 obs. of 5 variables:
## $ e-mail : int 1 2 3 4 5 6 7 8 9 10 ...
## $ spam : int 1 1 1 1 1 1 1 1 1 1 ...
## $ Indice_word : num -0.293 0.427 0.403 -0.202 -0.202 ...
## $ Indice_char : num 0.6643 0.6148 0.4892 -0.0449 -0.0531 ...
## $ Indice_capital: num 0.715 1.519 2.114 0.461 0.461 ...
```

Note que as covariáveis são quantitativas contínuas (numéricas) e a variável resposta é quantitativa discreta (inteira). Além disso, a coluna e-mail na base de dados indica o correspondente identificador do e-mail e embora numérica e uma variável qualitativa nominal.

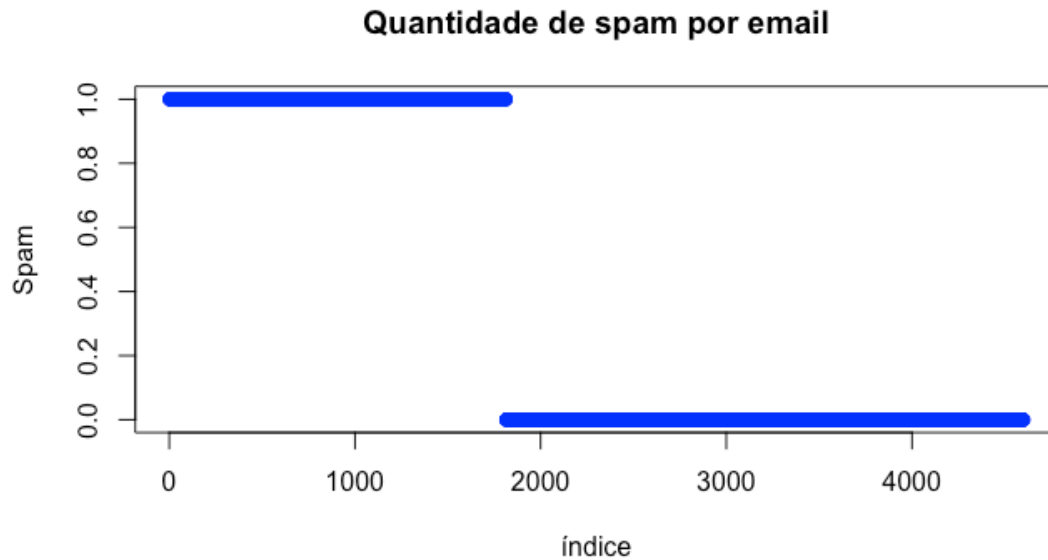
### 3. Análise descritiva das variáveis do modelo

A primeira etapa consiste em entender melhor os dados que estão sendo trabalhados, a partir disso fizemos uma análise descritiva.

#### 3.1 Análise da variável resposta

Para nossa análise descritiva usaremos

```
sum(dados$spam ==1)
## [1] 1813
mean(dados$spam)
## [1] 0.3940448
```



Observamos que a quantidade de e-mails considerados spam é de 1813 e-mails o que corresponde a uma proporção de 0.39.

Observando a figura acima visualizamos que a quantidade de e-mails sem spam é maior, isso ocorre porque os dados são desbalanceados, ou seja, há uma quantidade maior de e-mails sem spam em comparação com os e-mails com spam.

### 3.2 Análise das covariáveis

A partir da função `summary` do R, podemos ter informações sobre mínimo, máximo, média, mediana e quantis das variáveis explicativas.

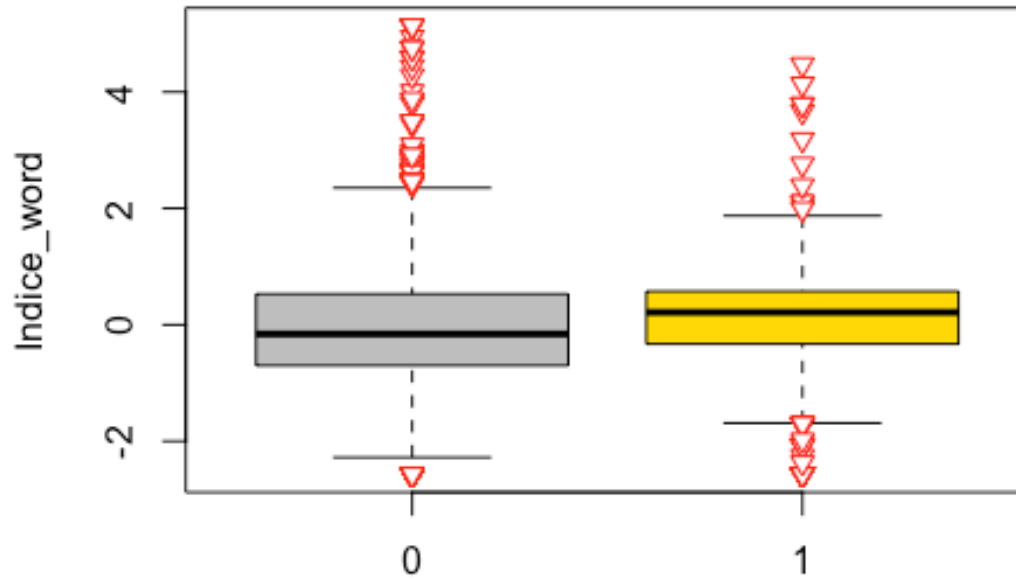
```
summary(dados[,-c(1,2)])  
##   Indice_word      Indice_char      Indice_capital  
##   Min.      :-2.570190   Min.      :-1.52567   Min.      :-2.465525  
##   1st Qu.: -0.585478   1st Qu.: -0.50344   1st Qu.: -0.649121  
##   Median :  0.001673   Median :  0.08871   Median : -0.001055  
##   Mean   :  0.000000   Mean   :  0.00000   Mean   :  0.000000  
##   3rd Qu.:  0.558472   3rd Qu.:  0.61372   3rd Qu.:  0.687094  
##   Max.    :  5.138182   Max.    :  7.34931   Max.    :  3.454816
```

Notamos por exemplo que o Índice de palavras varia entre -2,57 até 5,14, que o índice de caracteres varia entre -1,53 até 7,35 e que o índice de letras maiúsculas varia entre -2,47 até 3,45. A média em todos os casos é zero.

Agora, vamos fazer a análise descritiva para as covariáveis segundo a variável de resposta

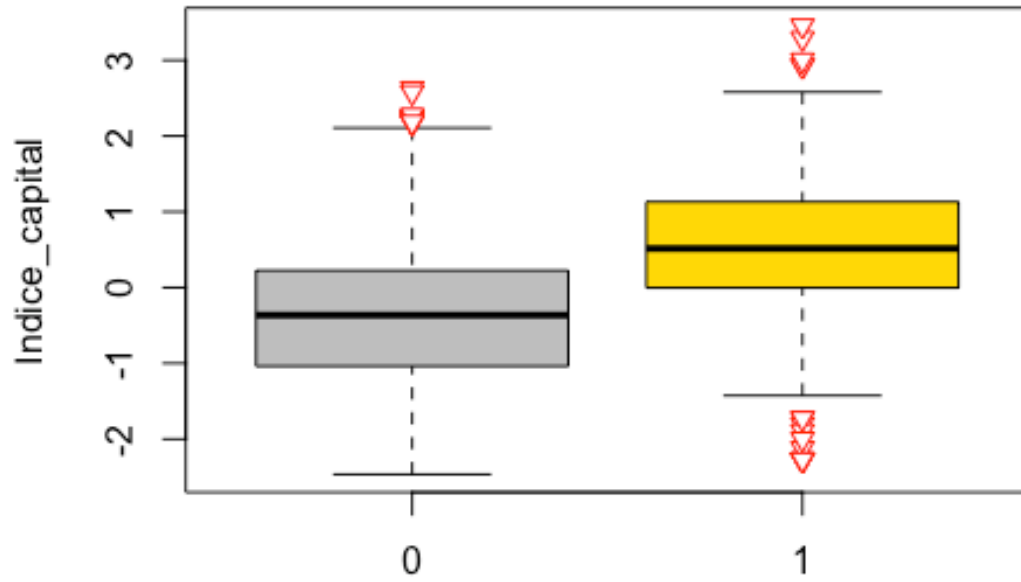
```
## $`0`  
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## -2.5702 -0.7010 -0.1589 -0.0731  0.5297  5.1382  
##  
## $`1`  
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.  
## -2.5702 -0.3306  0.2132  0.1124  0.5756  4.4726
```

Boxplot: Indice\_word para a quantidade de spam



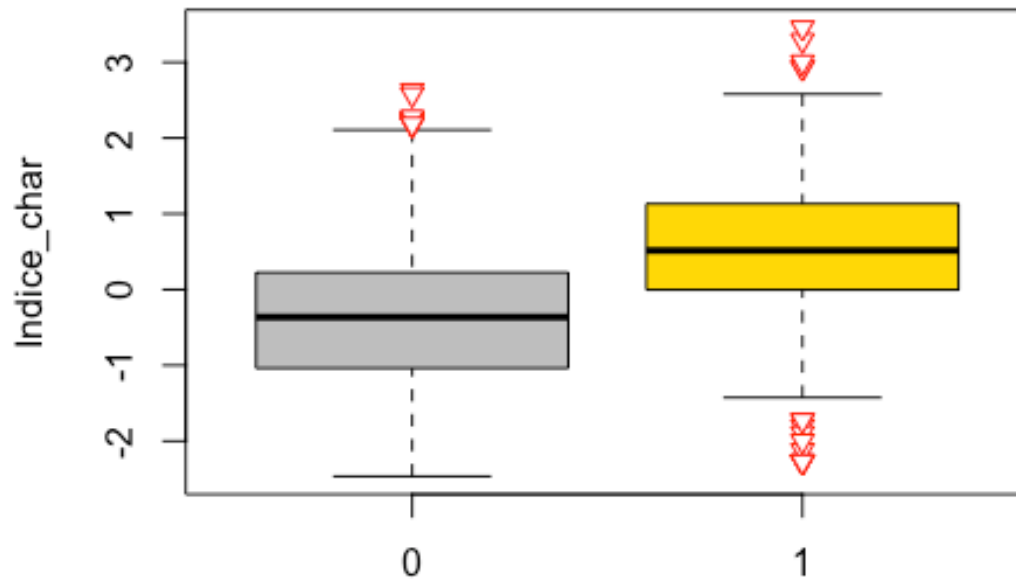
```
## $`0`
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -2.4655 -1.0385 -0.3646 -0.3669  0.2251  2.6268
##
## $`1`
##   Min.  1st Qu.   Median     Mean  3rd Qu.    Max.
## -2.272406 -0.003908  0.514677  0.564235  1.131243  3.454816
```

Boxplot: Indice\_capital para a quantidade de spam





Boxplot: Indice\_char para a quantidade de spam



```
## $`0`
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.5257 -1.5257 -0.1760 -0.3451  0.2733  7.3493
##
## $`1`
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## -1.52567  0.05477  0.55073  0.53072  0.98410  6.09588
```

Nas três covariáveis, para a quantidade de e-mails com spam, usando o boxplot verificamos que há uma assimetria a esquerda, isso implica que a média < mediana e também notamos a presença de outliers nos pontos altos dos índices. Já para o grupo sem spam, temos que há uma pequena assimetria à direita, isso mostra que a média > mediana, assim como observamos a presença de outliers nos valores altos. Notamos também que as covariáveis índices de caracteres e palavras em maiúsculas se mostram diferentes segundo seja spam ou não indicando que podem estar associadas nesta característica e serão importantes no modelo a serem proposto. Pelo contrário, o índice de palavras não se ve tão diferente entre os e-mails com e sem spam indicando que pode ser uma variável que pode não ser importante na formulação do modelo.

## 4. Formulação do modelo (de Classificação)

### 4.1 O Modelo e regressão binária

Seja  $Y_i, i = 1, \dots, n$  a variável binária definida por

$$Y_i = \begin{cases} 1, & \text{e. mail com spam.} \\ 0, & \text{caso contrário} \end{cases}$$

com  $n = 4601$  sendo o número de e-mails.

Para a formulação de nosso modelo nós assumimos que esta variável segue uma distribuição de Bernoulli denotada por  $Y_i \sim \text{Bernoulli}(\mu_i)$ , a qual assume dois valores 0 e 1, sendo 1 para e-mails com spam e 0 caso contrário, com probabilidade  $\mu_i \in [0,1]$ .

Sabemos que para uma resposta binária temos que,  $E(Y_i) = \sum_{y=0}^1 P(Y_i = y)y = 1 \times P(Y_i = 1) + 0 \times P(Y_i = 0) = \mu_i \in (0,1)$ . Então, temos interesse em estimar  $\hat{\mu}_i$ , em que  $y = 1$  significa que o e-mail é considerado spam.

Assim, o modelo de regressão binária diz que,

$$Y_i \sim \text{Bernoulli}(\mu_i)$$

com

$$\mu_i = F(\eta_i) = F(x_i^T \beta), \quad i = 1, \dots, n$$

em que

O modelo proposto é chamado modelo de regressão binária. Este modelo é um modelo de classificação que faz parte dos chamados modelos lineares generalizados. Existem também outros modelos de classificação no aprendizado supervisionado.

Especificamente, vamos considerar o modelo de regressão binária com uma função de ligação logito.

- Componente aleatório:  $y_1, \dots, y_{4601}$  é uma amostra aleatória  $Y_i \sim \text{Bernoulli}(\mu_i)$
- Componente sistemático:  $\eta_i = \beta_0 + \beta_1 \cdot \text{Indice\_word}_i + \beta_2 \cdot \text{Indice\_char}_i + \beta_3 \cdot \text{Indice\_capital}_i$

- Função de ligação  $\text{logit}(\mu_i) = \ln\left(\frac{\mu_i}{1-\mu_i}\right)$  ou

$$\eta_i = \log\left(\frac{\mu_i}{1-\mu_i}\right) = x_i^T \beta, \quad i = 1, \dots, 4601$$

Função de ligação logito
Preditor linear

Considerando a função distribuição acumulada da distribuição logística, tem-se a função de ligação logito, ou seja, em que o modelo é

$$\mu_i = F(\eta_i) = \frac{\exp(\eta_i)}{1 + \exp(\eta_i)} = \frac{\exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3)}{1 + \exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3)}$$

O nosso interesse é utilizar o chamado modelo de regressão binária para modelar  $\mu_i = E(Y_i|X)$ ,  $i = 1, \dots, n$  e estimar os coeficientes de regressão associados com as variáveis explicativas considerando uma determinada função de ligação.

## 4.2 Ajuste do modelo usando a função glm do R

Para ajustar o modelo proposto usando a função de ligação logito (canônica) usaremos a função glm do pacote R.

```
## [1] "e-mail"          "spam"              "Indice_word"      "Indice_char"
## [5] "Indice_capital"

##
## Call:
## glm(formula = spam ~ Indice_word + Indice_char + Indice_capital,
##      family = binomial(link = "logit"), data = dados)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.9384  -0.8460  -0.3040   0.8724   2.8620
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)    -0.66201    0.03819 -17.335 < 2e-16 ***
## Indice_word    0.13692    0.03881   3.528 0.000419 ***
## Indice_char    0.87578    0.04855  18.039 < 2e-16 ***
## Indice_capital 0.93577    0.04466  20.954 < 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: 6170.2  on 4600  degrees of freedom
## Residual deviance: 4679.5  on 4597  degrees of freedom
## AIC: 4687.5
##
## Number of Fisher Scoring iterations: 5
```

### 4.3 Testes de hipóteses de cada variável e intervalos de confiança

Para um nível  $\alpha = 0.05$  de significância, estamos querendo testar a seguinte hipótese:

$$\begin{cases} H_0: & \beta_j = 0, j = 0,1,2,3 \\ H_1: & \text{Para qualquer } \beta_j \neq 0 \end{cases}$$

Considerando 5% de nível de significância, observamos que, como comentado na análise descritiva, a covariável `Indice_word` não é significativa para o modelo, e as outras covariáveis do modelo (`Indice_char` e `Indice_capital`) são significativas.

Além disso, o resíduo está no intervalo  $(-3,3)$ , e o AIC desse ajuste é 4687,5.

Observamos também que as covariáveis `Indice_char` e `Indice_capital` possuem coeficiente de regressão positivo, isso indica que quanto maior os valores do índice de caracteres e de letras maiúsculas, maior a chance de o e-mail ser considerado spam.

Da mesma forma, a covariável `Indice_word` possui coeficiente de regressão negativo, o que indica que quanto menor os valores do índice de palavras dessa covariável, maior a chance de o e-mail ser considerado spam.

Estes resultados também podem ser conferidos considerando os intervalos de confiança dos coeficientes de regressão

```
(IC1 <- confint.default(fit.modell, level=0.95))
```

```
##           2.5 %    97.5 %
## (Intercept) -0.73686541 -0.5871631
## Indice_word  0.06084606  0.2129859
## Indice_char  0.78062836  0.9709384
## Indice_capital 0.84823884  1.0232981
```

Observamos que os intervalos de confiança dos coeficientes para estimar  $\beta_0, \beta_2, \beta_3$  não contém o valor zero, o que confirma que essas covariáveis são significativas, enquanto o IC do estimador de  $\beta_1$  possui o valor zero, o que confirma que essa covariável não é significativa para o modelo.

Vamos verificar a significância das variáveis utilizando um teste alternativo baseado na análise de deviance usando a estatística qui-quadrado.

```
# Teste chisq para o modelo 1
anova(fit.modell, test = 'Chisq')

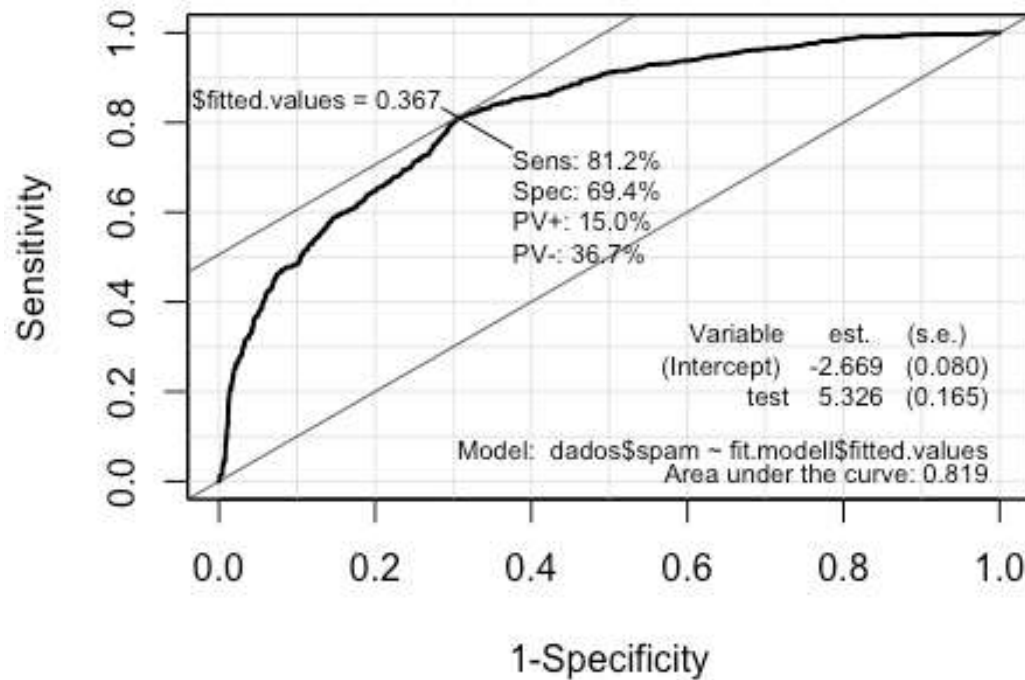
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: spam
##
## Terms added sequentially (first to last)
##
##
##           Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                4600      6170.2
## Indice_word      1    37.99      4599      6132.2 7.127e-10 ***
## Indice_char      1   932.92      4598      5199.3 < 2.2e-16 ***
## Indice_capital   1   519.78      4597      4679.5 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Neste caso podemos notar que as variáveis `Indice_char` e `Indice_capital` foram significativas para o modelo, já a variável `Indice_word` não foi significativa. Em outras palavras, rejeitamos  $H_0$  a um nível de significância para as duas últimas covariáveis (`Indice_char` e `Indice_capital`), isto significa que elas são significantes para o modelo e os coeficientes de regressão podem ser diferentes de 0. Agora, para a primeira covariável (`Indice_word`), não rejeitamos  $H_0$ , e, portanto, há indícios de que o coeficiente de regressão seja igual à 0.

#### 4.4 Análises preditivas

Usando os seguintes comandos obtemos as Curvas ROC (receiver operating characteristic) para o modelo

```
library(Epi)
ROC(fit.modell$fitted.values, dados$spam, plot= "ROC")
```



Obtivemos uma área abaixo da curva de 0.819, uma sensibilidade de 81,2% e uma especificidade de 69,4%.

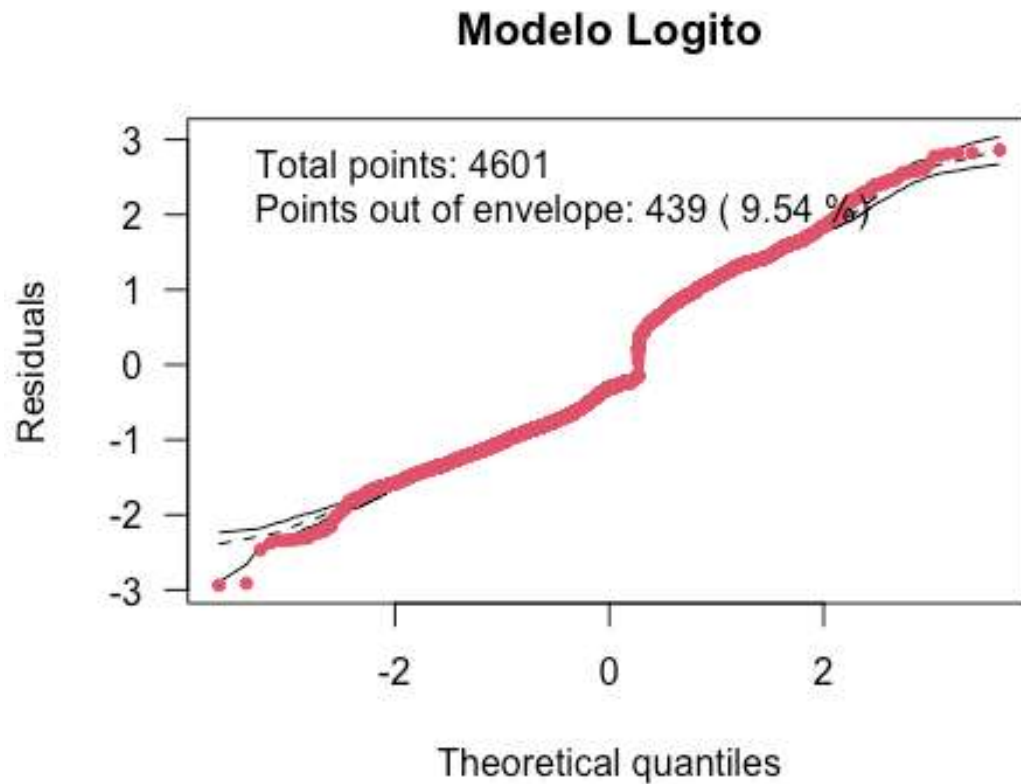
Adicionalmente, obtemos o gráfico de envelope considerando os seguintes comandos

```
library(hnp)
hnp.fit.modell = hnp(fit.modell, print.on=TRUE, plot=FALSE,
halfnormal=F)
```



```
## Binomial model
```

```
plot(hnp.fit.modell,main="Modelo Logito",las=1,pch=20,cex=1,col=c(1,1,1,2))
```



Conforme observado no gráfico acima, nenhum ponto está fora dos limites do envelope, o que indica bom ajuste dos dados ao modelo.

O gráfico de controle mostra que o modelo de regressão binária com a função de ligação logito não faz um bom ajuste do modelo, pois apresenta um 9,54% de observações fora do envelope.

## 5 Proposta de modelos alternativos usando diferentes ligações

### 5.1 Ajustando o modelo de regressão binária com ligação Probit

$$\eta_i = \Phi^{-1}(\mu_i) = x_i^T \beta, \quad i = 1, \dots, n$$

Função de ligação probito      Preditor linear

$$\mu_i = F(\eta_i) = \Phi(\eta_i) = \Phi(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3)$$

utilizando a função de ligação probito, temos o seguinte trecho de código:

```
# Função de ligação probito
fit.modelp<-glm(spam~Indice_word+Indice_char+Indice_capital, family = binomial(link = 'probit'))
#summary(fit.modelp)
```

Utilizando o teste anova.

```
# Teste chisq para o modelo 2
anova(fit.modelp, test = 'Chisq')

## Analysis of Deviance Table
##
## Model: binomial, link: probit
##
## Response: spam
##
## Terms added sequentially (first to last)
##
##
##          Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                4600      6170.2
## Indice_word         1    38.99      4599    6131.2 4.267e-10 ***
## Indice_char         1   899.96      4598    5231.2 < 2.2e-16 ***
## Indice_capital     1   534.52      4597    4696.7 < 2.2e-16 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com a função de ligação probito, não rejeitamos  $H_0$  para um nível de significância 5%, ou seja, há indícios de que o coeficiente de regressão seja igual à 0, já para as outras covariáveis, rejeitamos  $H_0$ , isso mostra que os coeficientes podem ser diferentes de 0.

## 5.2 Ajustando o modelo de regressão binária com ligação Cauchito

$$\eta_i = \tan(\pi(p_i - 0.5)) = x_i^T \beta, \quad i = 1, \dots, n$$

Função de ligação cauchito
Preditor linear

Considerando a função distribuição acumulada da distribuição Cauchy, tem-se a função de ligação cauchito, ou seja, em que o modelo é

$$\mu_i = F(\eta_i) = \frac{1}{2} + \frac{\arctan(\eta_i)}{\pi} = \frac{1}{2} + \frac{\arctan(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3)}{\pi}$$

utilizando a função de ligação cauchito, temos o seguinte trecho de código:

```
# Função de ligação cauchito
fit.modelc<-glm(spam~Indice_word+Indice_char+Indice_capital, family = binomial(link = 'cauchit'))
#summary(fit.modelc)
```

Utilizando o teste anova.

```
# Teste chisq para o modelo 3
anova(fit.modelc, test = 'Chisq')

## Analysis of Deviance Table
##
## Model: binomial, link: cauchit
##
## Response: spam
```

```
##
## Terms added sequentially (first to last)
##
##
##           Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                4600      6170.2
## Indice_word      1   32.18      4599      6138.0 1.402e-08 ***
## Indice_char      1 1012.50      4598      5125.5 < 2.2e-16 ***
## Indice_capital   1   478.85      4597      4646.6 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Dessa forma, rejeitamos  $H_0$  para as covariáveis `Indice_char` e `Indice_capital`, com isso pode-se dizer que os coeficientes de regressão são diferentes de 0, logo para a variável `Indice_word`, não rejeitamos  $H_0$ , então, há indícios de que o coeficiente seja igual à 0.

### 5.3 Ajustando o modelo de regressão binária com ligação Cloglog

$$\eta_i = \log(-\log(1 - \mu_i)) = \underbrace{x_i^T \beta}_{\text{Função de ligação cloglog}}, \quad i = 1, \dots, n$$

Considerando a função  $F(\eta) = 1 - \exp(-\exp(\eta))$ , tem-se a função de ligação cloglog, ou seja, em que o modelo é

$$\mu_i = F(\eta_i) = 1 - \exp(-\exp(\eta)) = 1 - \exp(-\exp(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3))$$

utilizando a função de ligação cloglog, temos o seguinte trecho de código:

```
# Função de Ligação cloglog
fit.modelcl<-glm(spam~Indice_word+Indice_char+Indice_capital, family = binomial(link = 'cloglog'))
#summary(fit.modelcl)
```

Utilizando o teste anova.

```
# Teste chisq para o modelo 4
anova(fit.modelcl, test = 'Chisq')

## Analysis of Deviance Table
##
## Model: binomial, link: cloglog
##
## Response: spam
##
## Terms added sequentially (first to last)
##
##
##              Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                4600      6170.2
## Indice_word         1   33.13      4599      6137.0 8.626e-09 ***
## Indice_char         1  737.77      4598      5399.3 < 2.2e-16 ***
## Indice_capital     1  678.72      4597      4720.5 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com a função de ligação cloglog, não rejeitamos  $H_0$  para a variável `Indice_word`, e assim como, pode-se dizer que há indícios que o coeficiente seja igual à 0, já para as covariáveis restantes, rejeitamos  $H_0$ , dessa forma os coeficientes de regressão podem ser diferentes de 0.

#### 5.4 Ajustando o modelo de regressão binária com ligação loglog

Considerando a função  $F(\eta) = \exp(\exp(\eta))$ , tem-se a função de ligação loglog, ou seja, em que o modelo é

$$\eta_i = -\log(-\log(\mu_i)) = x_i^T \beta, \quad i = 1, \dots, n$$

Função de ligação loglog      Preditor linear

$$\mu_i = F(\eta_i) = \exp(-\exp(-\eta_i)) = \exp(-\exp(-\beta_0 - \beta_1 x_1 - \beta_2 x_2 - \beta_3 x_3))$$

utilizando a função de ligação cauchito, temos o seguinte trecho de código:

```
# Função de Ligação LogLog

# Geradora para a função de Ligação LogLog
loglog <- function( ) structure(list(
  linkfun = function(mu) -log(-log(mu)),
  linkinv = function(eta)
    pmax(pmin(exp(-exp(-eta)), 1 - .Machine$double.eps),
          .Machine$double.eps),
  mu.eta = function(eta) {
    eta <- pmin(eta, 700)
    pmax(exp(-eta - exp(-eta)), .Machine$double.eps)
  },
  dmu.deta = function(eta)
    pmax(exp(-exp(-eta) - eta) * expm1(-eta),
          .Machine$double.eps),
  valideta = function(eta) TRUE,
  name = "loglog"
), class = "link-glm")

fit.modelll<-glm(spam~Indice_word+Indice_char+Indice_capital, family = binomial(link = loglog()))
#summary(fit.modelll)
```

Utilizando o teste anova.

```
# Teste chisq para o modelo 5
anova(fit.modelll,test = 'Chisq')

## Analysis of Deviance Table
##
## Model: binomial, link: loglog
##
## Response: spam
##
```

```
## Terms added sequentially (first to last)
##
##
##           Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
## NULL                4600      6170.2
## Indice_word      1    46.11      4599      6124.0 1.116e-11 ***
## Indice_char      1   928.28      4598      5195.8 < 2.2e-16 ***
## Indice_capital   1   436.58      4597      4759.2 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com a função de ligação loglog, não rejeitamos  $H_0$  para a variável `Indice_word`, e assim como, pode-se dizer que há indícios que o coeficiente  $\beta_1$  seja igual à 0, já para as covariáveis restantes, rejeitamos  $H_0$ . Sendo assim, podemos dizer que, provavelmente os coeficientes  $\beta_2$  e  $\beta_3$  sejam diferentes de 0. Em outras palavras, note que os coeficientes significativos são positivos e influenciam diretamente na variável resposta.

### 5.5 Escolhendo o modelo de regressão binária com diferentes ligações

Agora, vamos fazer um dataframe e verificar qual o modelo que obteve o menor AIC.

```
# Dataframe para verificar o AIC
data.frame(Modelo=c("Modelo logito", "Modelo probito", "Modelo cauchito", "Modelo cloglog", "Modelo loglog"),
           AIC = c(AIC(fit.modell), AIC(fit.modelp), AIC(fit.modelc),
                 AIC(fit.modelcl), AIC(fit.modelll)))

##           Modelo      AIC
## 1  Modelo logito 4687.472
## 2  Modelo probito 4704.685
## 3  Modelo cauchito 4654.618
## 4  Modelo cloglog 4728.527
## 5  Modelo loglog 4767.179
```

Portanto, escolhemos o modelo de regressão binária com ligação cauchito por ser porque seu AIC foi o menor dentre todos os outros modelos. Neste caso, no momento temos que a interpretação dos coeficientes: Para encontrar spam no e-mail é:

- Diminui conforme o coeficiente do `Indice_word` decresce;
- Aumenta conforme há um acréscimo no coeficiente do `Indice_char`;
- Aumenta conforme há um acréscimo no coeficiente do `Indice_capital`.



## 6. Análise de seleção de variáveis para o modelo proposto

Para o modelo escolhido,

- $y_i|x \stackrel{\text{iid}}{\sim} \text{Bernoulli}(p_i)$
- $\tan(\pi(p_i - 0.5)) = \beta_0 + \beta_1 \times \text{Indice.word} + \beta_2 \times \text{Indice.char} + \beta_3 \times \text{Indice.capital}$

vamos fazer uma análise de seleção de variáveis utilizando a função do R `stepAIC`, que nos ajuda a detectar os melhores preditores.

Utilizando a função `stepAIC`, temos:

```
# stepAIC
stepAIC(fit.modelc)

## Start:  AIC=4654.62
## spam ~ Indice_word + Indice_char + Indice_capital
##
##           Df Deviance   AIC
## <none>           4646.6 4654.6
## - Indice_word     1  4648.7 4654.7
## - Indice_capital  1  5125.5 5131.5
## - Indice_char     1  5136.8 5142.8
##
## Call:  glm(formula = spam ~ Indice_word + Indice_char + Indice_capital,
##           family = binomial(link = "cauchit"))
##
## Coefficients:
## (Intercept)  Indice_word  Indice_char  Indice_capital
##      -0.8296         0.0567         1.2888         0.9520
##
## Degrees of Freedom: 4600 Total (i.e. Null);  4597 Residual
```

```
## Null Deviance:      6170
## Residual Deviance: 4647  AIC: 4655
```

Note que realmente quando retiramos a variável `Indice_word` o AIC é menor de quando retiramos outras variáveis. Assim, a sugestão é retirar esta variável do modelo. Assim, concluímos que as variáveis mais significativas para o modelo proposto são: - `Indice_char` - `Indice_capital`

O modelo reduzido é obtido usando

```
#modelo reduzido
fit.modelcr<-glm(spam~Indice_char+Indice_capital,
                 family = binomial(link = 'cauchit'))
summary(fit.modelcr)

##
## Call:
## glm(formula = spam ~ Indice_char + Indice_capital, family = binomial(link = "cauchit"))
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.4960  -0.7602  -0.4100   0.7914   2.3524
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -0.83214    0.05140  -16.19  <2e-16 ***
## Indice_char   1.30210    0.07655   17.01  <2e-16 ***
## Indice_capital 0.95901    0.05779   16.59  <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: 6170.2  on 4600  degrees of freedom
## Residual deviance: 4648.7  on 4598  degrees of freedom
## AIC: 4654.7
```

```
##  
## Number of Fisher Scoring iterations: 7
```

Assim o modelo reduzido é

- $y_i|x \stackrel{\text{iid}}{\sim} \text{Bernoulli}(p_i)$
- $\tan(\pi(\hat{\mu}_i - 0.5)) = -0.83214 + 1.30210 \times \text{Indice.char} + 0.95901 \times \text{Indice.capital}$

## 7. Análise diagnóstica para identificar pontos problemáticos no modelo reduzido

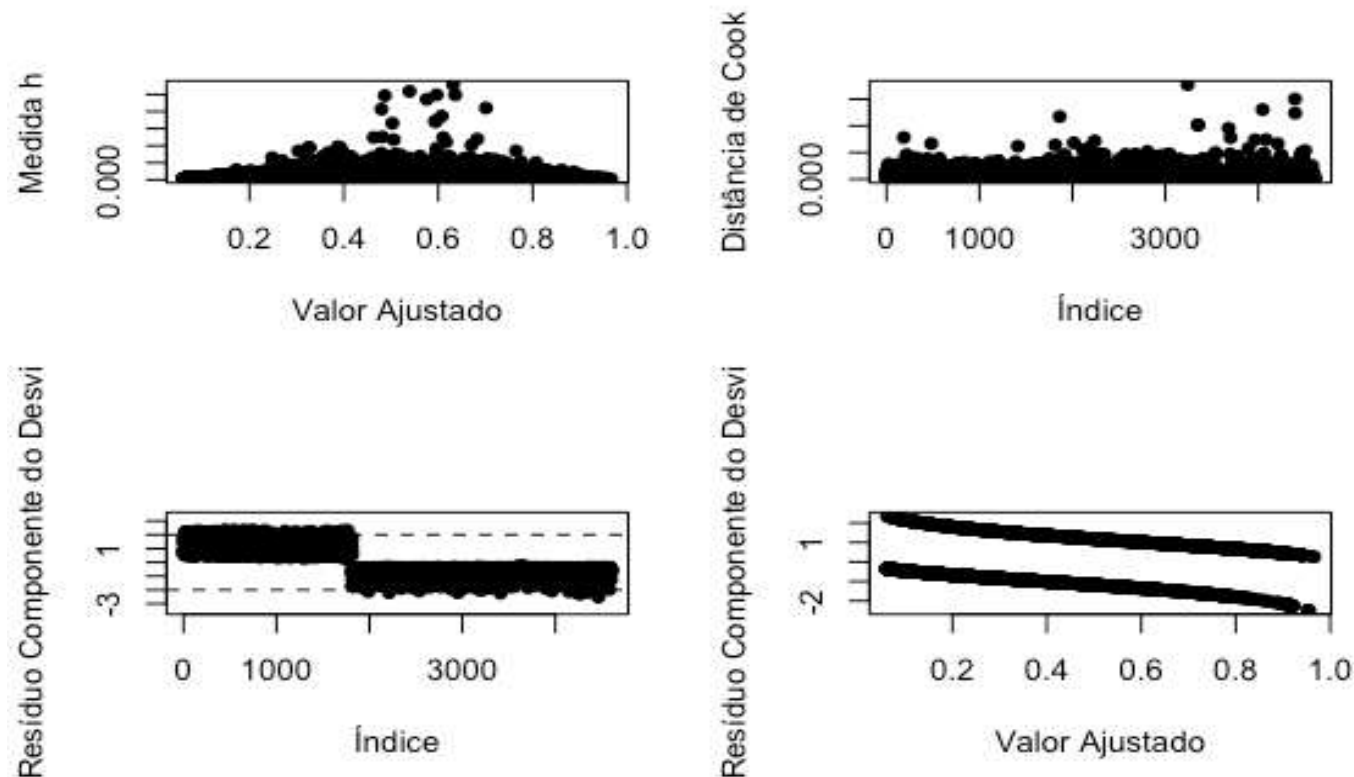
### 7.1 Análise diagnóstico considerando diferentes medidas

Para o modelo reduzido, temos a seguinte análise de diagnóstico. A saída terá quatro gráficos: de pontos de alavanca, de pontos influentes e dois de resíduos.

```
# Análise de diagnostico Modelo reduzido
#source("http://www.ime.usp.br/~giapaula/diag_bino")
setwd("/Users/jorgebazan/OneDrive/SEMESTRE20202/AulaAberta/DadosSpam/")
fit.model<-fit.modelcr
attach(dados)

## The following objects are masked from dados (pos = 3):
##
##      e-mail, Indice_capital, Indice_char, Indice_word, spam

source("diag_bino.txt")
```



A figura anterior não é fácil de interpretar dado a grande quantidade de pontos observados.

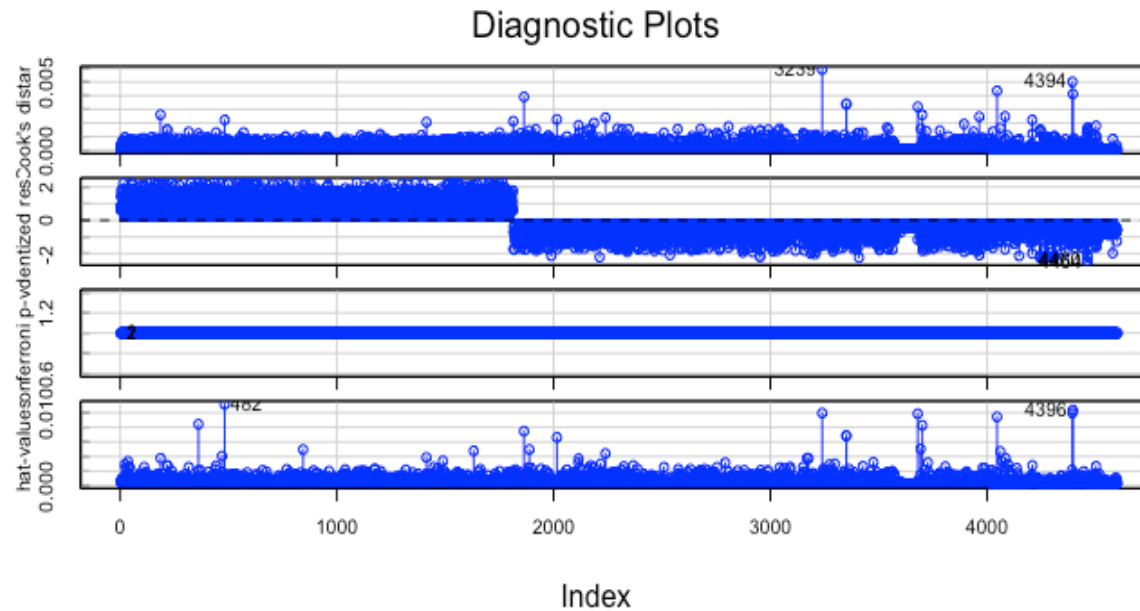
## 7.2 Identificação de Pontos problemáticos

Agora, vamos fazer uma análise diagnóstica para identificar pontos problemáticos do modelo reduzido. Para isso, vamos utilizar a função `InfluenceIndexPlot` do pacote `car` no R.

A figura a seguir apresenta diferentes quantidades calculadas para cada uma das observações usando medidas de diagnóstico de pontos influentes usualmente apresentadas nos modelos lineares generalizados. A quantidade “Cook” corresponde a distância de Cook (para

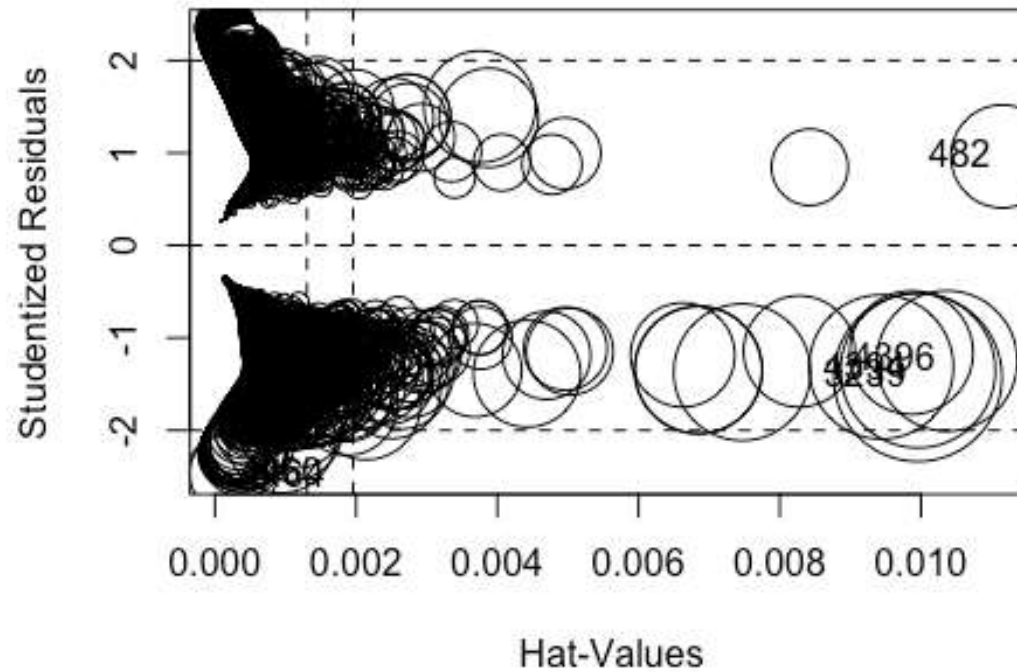
detectar pontos influentes), “Studentized” corresponde aos resíduos studentizados (para detectar homocedasticidade), “Bonf” corresponde aos valores p do teste Bonferroni para outliers e , por fim, “hat” para os valores-hat values (ou pontos de alavanca).

```
# gráfico de influência e alavanca  
influenceIndexPlot(fit.modelcr,col='blue')
```



Para identificar quais são os pontos influentes dentre os apresentados nos 4 gráficos anteriores, utilizamos a função Influenceplot:

```
influencePlot(fit.modelcr)
```



##	StudRes	Hat	CookD
## 482	0.9623373	0.0111542181	0.002219420
## 3239	-1.4274729	0.0099550735	0.005910185
## 4394	-1.3527323	0.0099569329	0.005005098
## 4396	-1.2495112	0.0103683347	0.004127181
## 4460	-2.4684301	0.0001861994	0.001242037
## 4464	-2.4970006	0.0002320677	0.001666523

A figura que mostra as observações segundo os resíduos studentizados e valor  $h$ , mostra também círculos proporcionais ao valor da distância de Cook.

Considerando os valores dos resíduos studentizados, percebemos que os pontos 4460 e 4464 se encontram fora do intervalo  $(-2,2)$ .

Para identificar os pontos influentes, precisamos encontrar aqueles com valor  $\hat{h} > \frac{2p}{n} = \frac{6}{4601} = .001$ , onde  $p = 3$  é o número de coeficientes de regressão e  $n = 4601$  é o número de observações. Neste caso, identificamos como ponto de alavanca ( $\hat{h}$ ) os pontos: 482, 3239, 4394 e 4396, já para os pontos de influência (Distância de Cook): 3239, 4394 e 4396, e assim, levando em consideração os pontos que têm mais de uma indicação problemática, concluímos que estes pontos requerem uma análise mais detalhada.

#### 7.4 Ajuste do modelo retirando alguns pontos

```
# Retirada do ponto 3239
ajuste2<-glm(spam~Indice_char+Indice_capital,
             subset = -c(3239),
             family = binomial(link='cauchit'),
             data=dados)

#summary(ajuste2)

# Retirada do ponto 4394
ajuste3<-glm(spam~Indice_char+Indice_capital,
             subset = -c(4394),
             family = binomial(link='cauchit'),
             data=dados)

#summary(ajuste3)

# Retirada do ponto 4396
ajuste4<-glm(spam~Indice_char+Indice_capital,
             subset = -c(4396),
             family = binomial(link='cauchit'),
             data=dados)

#summary(ajuste4)

# Retirada do ponto 3239 e 4394
ajuste5<-glm(spam~Indice_char+Indice_capital,
             subset = -c(3239,4394),
```



```
        family = binomial(link='cauchit'),
        data=dados)
#summary(ajuste5)

# Retirada do ponto 3239 e 4396
ajuste6<-glm(spam~Indice_char+Indice_capital,
             subset = -c(3239,4396),
             family = binomial(link='cauchit'),
             data=dados)
#summary(ajuste6)

# Retirada do ponto 4394 e 4396
ajuste7<-glm(spam~Indice_char+Indice_capital,
             subset = -c(4394,4396),
             family = binomial(link='cauchit'),
             data=dados)
#summary(ajuste7)

# Retirada do ponto 3239, 4394 e 4396
ajuste8<-glm(spam~Indice_char+Indice_capital,
             subset = -c(3239,4394,4396),
             family = binomial(link='cauchit'),
             data=dados)
#summary(ajuste8)
```

Vamos comparar os coeficientes de todos os modelos acima, baseados na retirada de pontos e no modelo que não foram retirados pontos.

```
compareCoefs(fit.modelo1,ajuste2, ajuste3, ajuste4,
             ajuste5, ajuste6, ajuste7,
             ajuste8)

## Calls:
## 1: glm(formula = spam ~ Indice_char + Indice_capital, family =
##    binomial(link = "cauchit"))
## 2: glm(formula = spam ~ Indice_char + Indice_capital, family =
```

```
## binomial(link = "cauchit"), data = dados, subset = -c(3239))
## 3: glm(formula = spam ~ Indice_char + Indice_capital, family =
## binomial(link = "cauchit"), data = dados, subset = -c(4394))
## 4: glm(formula = spam ~ Indice_char + Indice_capital, family =
## binomial(link = "cauchit"), data = dados, subset = -c(4396))
## 5: glm(formula = spam ~ Indice_char + Indice_capital, family =
## binomial(link = "cauchit"), data = dados, subset = -c(3239, 4394))
## 6: glm(formula = spam ~ Indice_char + Indice_capital, family =
## binomial(link = "cauchit"), data = dados, subset = -c(3239, 4396))
## 7: glm(formula = spam ~ Indice_char + Indice_capital, family =
## binomial(link = "cauchit"), data = dados, subset = -c(4394, 4396))
## 8: glm(formula = spam ~ Indice_char + Indice_capital, family =
## binomial(link = "cauchit"), data = dados, subset = -c(3239, 4394, 4396))
##
##
##           Model 1 Model 2 Model 3 Model 4 Model 5 Model 6 Model 7 Model 8
## (Intercept) -0.8321 -0.8332 -0.8329 -0.8326 -0.8340 -0.8337 -0.8334 -0.8346
## SE           0.0514  0.0514  0.0514  0.0514  0.0515  0.0515  0.0515  0.0515
##
## Indice_char  1.3021  1.3096  1.3088  1.3079  1.3165  1.3156  1.3148  1.3225
## SE           0.0765  0.0770  0.0769  0.0769  0.0774  0.0773  0.0773  0.0778
##
## Indice_capital 0.9590  0.9556  0.9557  0.9558  0.9524  0.9524  0.9525  0.9491
## SE           0.0578  0.0578  0.0578  0.0578  0.0578  0.0578  0.0578  0.0578
##
```

Conseguimos perceber que os coeficientes de regressão dos modelos propostos, quando se retiraram os pontos identificados na análise de diagnóstico não mudaram em relação ao modelo com todos os pontos (`model1:fit.modelcr`), e as interpretações são mantidas. Assim, mantemos o modelo reduzido como modelo final.

Comparando o AIC dos modelos com retiradas dos pontos

Vamos analisar os valores de AIC de cada modelo.

```
data.frame(  
  Modelo= c("Completo", "Removendo 3239", "Removendo 4394",  
            "Removendo 4396", "Removendo 3239 e 4394", "Removendo 3239 e 4396",  
            "Removendo 4394 e 4396", "Removendo 3239, 4394 e 4396"),  
  AIC = c(AIC(fit.modelcr), AIC(ajuste2), AIC(ajuste3), AIC(ajuste4),  
          AIC(ajuste5), AIC(ajuste6), AIC(ajuste7),  
          AIC(ajuste8)))  
  
##           Modelo      AIC  
## 1           Completo 4654.747  
## 2           Removendo 3239 4652.708  
## 3           Removendo 4394 4652.916  
## 4           Removendo 4396 4653.185  
## 5   Removendo 3239 e 4394 4650.844  
## 6   Removendo 3239 e 4396 4651.116  
## 7   Removendo 4394 e 4396 4651.326  
## 8 Removendo 3239, 4394 e 4396 4649.222
```

Ao comparar os AICs, observamos que sempre que removemos algum ponto detectado na análise diagnóstica, obtemos um menor AIC, indicando um melhor modelo, embora a diminuição seja pequena. Nós detectamos que o modelo com menor AIC é aquele que retira todos os três pontos influentes. O AIC do modelo com todos os pontos é 4654.747 e o AIC do modelo removendo os três pontos influentes é 4649.222.

Como a retirada dos pontos é uma questão delicada, não podemos retirar os pontos sem antes conhecer a fundo o problema e sem a devida permissão do pesquisador. Então, podemos sugerir dois modelos, o com os dados completos e um alternativo removendo 3239, 4394 e 4396.

## 8. Modelo final e interpretação de parâmetros

Anteriormente, dizemos que o ganho de AIC quando retiramos os 3 pontos problemáticos foi mínimo, dessa forma, escolhemos o modelo reduzido como o mais apropriado para os dados de spam.

- Modelo final

Assim o modelo final é

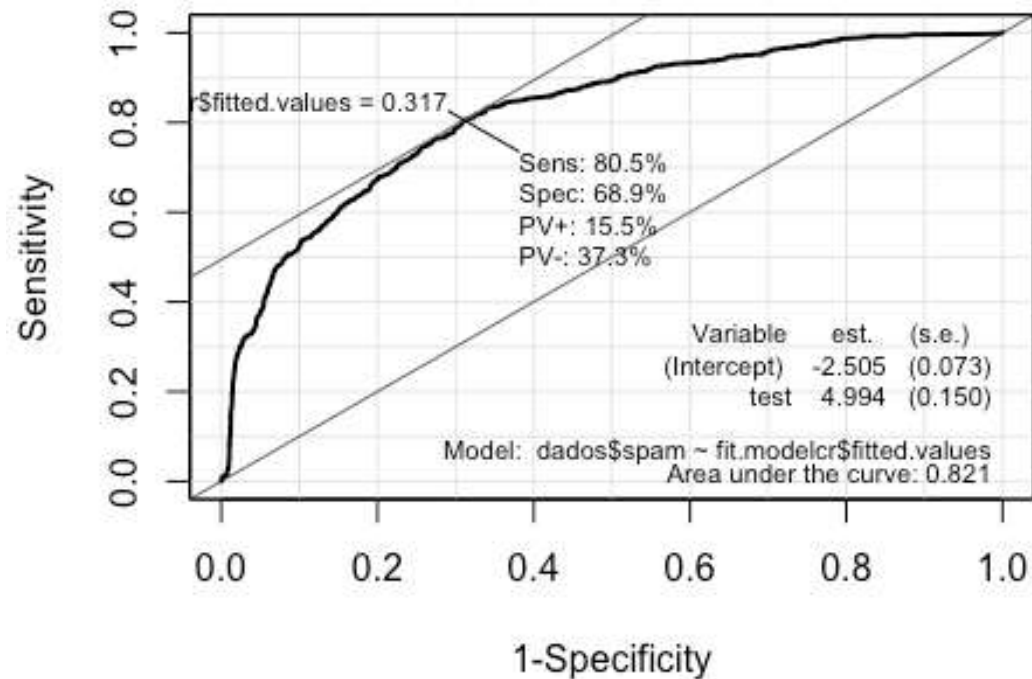
- $y_i|x \stackrel{\text{iid}}{\sim} \text{Bernoulli}(\hat{\mu}_i)$
- $\tan(\pi(\hat{\mu}_i - 0.5)) = -0.83214 + 1.30210 \times \text{Indice.char} + 0.95901 \times \text{Indice.capital}$   
ou
- $\hat{\mu}_i = 0.5 + \frac{1}{\pi} \arctan(-0.83214 + 1.30210 \times \text{Indice.char} + 0.95901 \times \text{Indice.capital})$

Notamos que os coeficientes são positivos. Assim, a cada aumento de uma unidade na variável preditiva `Indice_char` para um efeito zero do `Indice_capital`, há um aumento de  $0.5 + \arctan(-0.83214 + 1.30210)/\pi = 0.6398425$  na média (probabilidade) da variável resposta (spam). Por outro lado, a cada aumento de uma unidade na variável preditiva `Indice_capital` para um efeito zero do `Indice_char`, há um aumento de  $0.5 + \arctan(-0.83214 + 0.95901)/\pi = 0.5401694$  na média (probabilidade) da variável resposta (spam). Também, a cada aumento de uma unidade em ambas variáveis preditivas `Indice_char` e `Indice_capital`, há um aumento de  $0.5 + \arctan(-0.83214 + 1.3021 + 0.95901)/\pi = 0.8056416$  na média (probabilidade) da variável resposta (spam).

Em outras palavras podemos dizer que a) isolando o índice de caracteres, se este se incrementa em uma unidade há uma probabilidade de 64% de obter spam e também, b) isolando o índice de letras maiúsculas, há uma probabilidade do 54% de obter spam e c) se os índices de caracteres e de letras maiúsculas se incrementam em uma unidade então a probabilidade de obter spam é de 81%. Assim, decidir se um e-mail será spam ou não depende destes índices e de um umbral o ponto de corte para decidir se ultrapassado essa probabilidade, o e-mail pode ser considerado spam.

Uma forma bastante utilizada para determinar o ponto de corte é através da Curva ROC que para o modelo final é

```
library(Epi)
ROC(fit.modelcr$fitted.values, dados$spam, plot= "ROC")
```

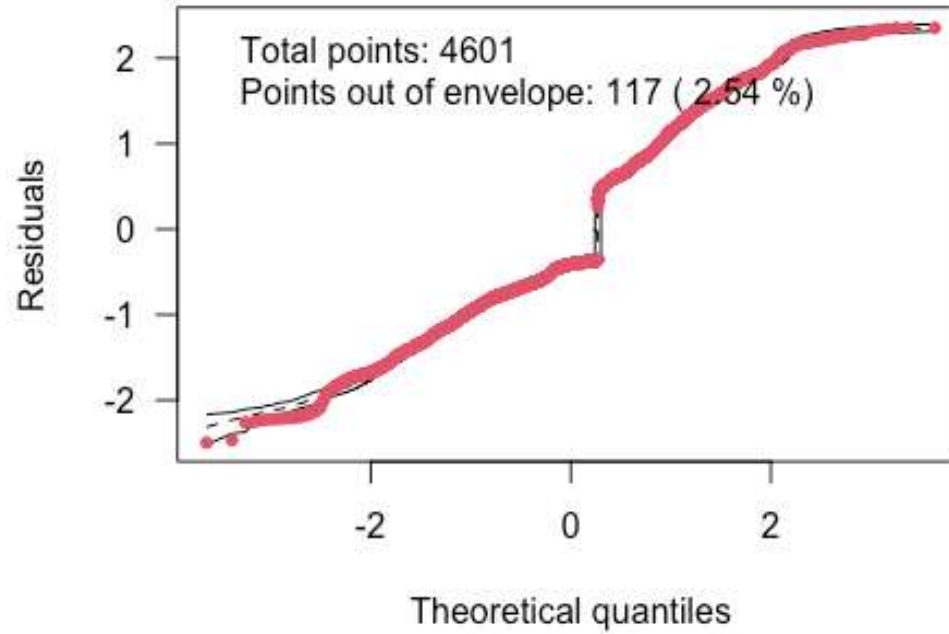


A escolha do ponto de corte deve ser baseada em uma combinação ótima tanto da sensibilidade (proporção de verdadeiros positivos) quanto da especificidade do modelo. É a (proporção de verdadeiros negativos), pois partimos do suposto que classificar o e-mail como sendo spam dado que ele não é spam (falso positivo) e classificar o e-mail como não sendo spam dado que ele é spam (falso negativo) traz prejuízos equivalentes para o usuário. Pela análise da curva ROC, escolhemos o ponto de corte referente a combinação da sensibilidade e 1-especificidade que mais se aproxima do canto superior esquerdo do gráfico que neste caso é aproximadamente 0.75.

Temos encontrado que área baixo a curva ROC do modelo de 82%, a sensibilidade do modelo é 81% (capacidade do modelo classificar um indivíduo como spam ( $\hat{Y} = 1$ ) dado que realmente ele é spam ( $Y = 1$ )) e especificidade de 69% (capacidade do modelo predizer um indivíduo como não spam ( $\hat{Y} = 0$ ) dado que ele realmente não é spam ( $Y=0$ )). Tendo em consideração estes resultados, é possível obter um outro modelo para classificar os e-mails como sendo spam ou não que seja melhor do que este modelo?

Adicionalmente a Análise de Envelope do modelo é

```
hnp.glm.cauchit <- hnp(fit.modelcr, print.on=TRUE, plot=FALSE, halfnormal=F)
## Binomial model
plot(hnp.glm.cauchit, las=1, pch=20, cex=1, col=c(1,1,1,2))
```



Ao ver parece que o modelo está bem ajustado, somente encontramos um 2.54% fora do envelope o que indica que ainda este modelo pode ser melhorado.

## 9. Conclusões acerca da metodologia

### 9.1 Conclusões do modelo

- Concluímos que o melhor modelo de classificação para os dados entre os considerados neste reporte é o modelo de regressão binária utiliza função de ligação cauchito sem desconsiderar nenhuma observação.
- Além disso, as variáveis que melhor preveem se um e-mail é considerado spam ou não são os índices de caracteres (Indice\_char) e índice de letras capitais (Indice\_capital) ambos influenciando positivamente na probabilidade de um e-mail ser classificado como spam.
- Três observações foram identificadas como problemáticas (os e-mails 239, 4394 e 4396) e as análises mostraram que elas podem ser desconsideradas na formulação do modelo porém o ganho em termos de ajuste não foi relevante. Assim, é importante salientar que a retirada de pontos é delicada, portanto, é necessário consultar o pesquisador e entender bem sobre o contexto da situação que está sendo analisada.

### 9.2 Conclusões da metodologia

Tomando em consideração todos os dados fornecidos, fizemos uma análise descritiva e formulamos um modelo de regressão binária logística, com isso estudamos modelos alternativos usando várias funções de ligação e usando diferentes critérios escolhemos o modelo com o menor critério de comparação de modelos. Logo após, encontramos o modelo reduzido selecionado as variáveis significativas no modelo escolhido e para esse modelo desenvolvemos uma análise diagnóstica, onde identificamos três pontos problemáticos, porém mesmo assim o modelo final coincide com o modelo reduzido. Visto isso, interpretamos os coeficientes de regressão e verificamos que, em média, há efeitos na probabilidade de um e-mail ser classificado como spam considerando as duas covariáveis consideradas. Finalmente encontramos a curva ROC e concluímos que o modelo está bem ajustado embora pode ainda ser melhorado em termos de previsão.



### 9.3 Melhoras do modelo

- Estudar e acrescentar alguma covariável que possa ser significativa para o modelo. Por exemplo identificar variáveis mais específicas baseadas em algumas palavras chaves ou ainda se o e-mail possui alguma imagem ou não.
- Incorporar metodologias de previsão.

## 10. Outras considerações do problema

### 10.1 Previsão versus Explicação

Uma proposta de análise que pode ser desenvolvida é a análise de previsão. Neste caso a ideia é prever se o modelo classifica para outro conjunto de amostras. Com isso devemos pensar em amostras de teste e amostras de treino. Assim, podemos selecionar uma nova amostra aleatória sem reposição, dividir a amostra em conjuntos chamados de treinamento e teste, por exemplo de 70% e 30% respectivamente (metodologia de validação cruzada), e então reproduzir a metodologia proposta.

A partir disso criar a chamada matriz de confusão e encontrar algumas medidas como a sensibilidade, especificidade e a acurácia do modelo (taxa de acertos). Com base a estes resultados assim, pode-se comparar com o modelo proposto atualmente, e assim se pode verificar o desempenho do modelo final.

O seguinte código apresenta um exemplo disso

```
set.seed(6)
dados1<-dados_spamf[sample(nrow(dados_spamf),0.7*4601),]
#install.packages("dplyr")
library(dplyr)
dados2<-(dados_spamf %>% anti_join(dados1))

## Joining, by = c("e-mail", "spam", "Indice_word", "Indice_char", "Indice_capital")

dados3<-dados2[sample(nrow(dados2),0.3*4601),]
```

Podemos considerar uma amostra aleatória sem reposição de 30% observações que não contém o 70% das observações anteriores como um novo banco de dados de análise de previsão futura. Dessa forma, usaremos o modelo final, e com ele faremos predições usando a função predict, e por tanto será possível criar uma matrix de confusão dada por:

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction  0    1
##           0 699 217
##           1 123 341
```

```
##
##           Accuracy : 0.7536
##           95% CI   : (0.73, 0.7762)
## No Information Rate : 0.5957
## P-Value [Acc > NIR] : < 2.2e-16
##
##           Kappa   : 0.4743
##
## Mcnemar's Test P-Value : 4.568e-07
##
##           Sensitivity : 0.8504
##           Specificity : 0.6111
##           Pos Pred Value : 0.7631
##           Neg Pred Value : 0.7349
##           Prevalence : 0.5957
##           Detection Rate : 0.5065
## Detection Prevalence : 0.6638
##           Balanced Accuracy : 0.7307
##
##           'Positive' Class : 0
##
```

Utilizando o mesmo modelo anterior, isto é, o modelo reduzido, definimos um ponto de corte  $> 0.5$  para spam da variável resposta (de fato o ponto de corte ou cutoff pode ser determinado usando diferentes critérios). Assim será possível obter a Sensibilidade, Especificidade e Acurácia do modelo com novos dados.

## 11.2 Outros métodos de classificação binária

- Naive Bayes,
- Support Vector Machine,
- Voting Classifier
- Neuronal Network.
- K-Nearest Neighbours,
- Decision Tree

- Bagging Decision Tree.
- Binary regression using asymmetrical link function

### **11.3 Métodos baseados em Amostragem**

- Sobre Amostragem (Over-sampling)
- Sub amostragem (Under-Sampling)
- Sobre amostragem seguida por sub amostragem
- Ensemble Classifiers com amostragem interna

### **11.4 Outros métodos para comparar modelos**

- F1-Score
- g-mean
- Weighted Accuracy (WA)
- Jaccard index
- Índice de Sokal e Sneath (SSI)
- Índice de Faith (FAITH)
- Pattern Difference (PDIF)
- Gilbert skill score (GS)

## 12. Referencias

### 12.1 Referencias metodológicas

- Bazán, J. (2020). Análise de Dados Categorizados com auxílio computacional. a4. Regressão dicotômica.
- Hosmer, D. W. e Lemeshow, S. (2000). Applied Logistic Regression. John Wiley & Sons, New York.
- Paula, G. A. (2013). Modelos de Regressão com apoio computacional. IME-USP, São Paulo. [Não publicado, disponível em [https://www.ime.usp.br/~giapaula/texto\\_2013.pdf](https://www.ime.usp.br/~giapaula/texto_2013.pdf)]

### 12.2 Acerca do modelamento de dados de Spam

- Dada, E. G., Bassi, J. S., Chiroma, H., Abdulhamid, S. M., Adetunmbi, A. O. and Ajibuwa, O. E. (2019). Machine learning for e-mail spam filtering: review, approaches and open research problems. Heliyon, 5(6) e01802.
- Mohammad, R. M. A. (2020). A lifelong spam e-mails classification model. Applied Computing and Informatics. Article in Press.
- Trivedi, S.K. and Panigrahi, P.K. (2018). Spam classification: a comparative analysis of different boosted decision tree approaches. Journal of Systems and Information Technology, 20(3), 298-105.



## Machine learning for email spam filtering: review, approaches and open research problems



Emmanuel Gbenga Dada <sup>a,\*</sup>, Joseph Stephen Bassi <sup>a</sup>, Haruna Chiroma <sup>b</sup>,  
Shafi i Muhammad Abdulhamid <sup>c</sup>, Adebayo Oluola Adetunmbi <sup>d</sup>, Opeyemi Emmanuel Ajibuwa <sup>e</sup>

<sup>a</sup> Department of Computer Engineering, University of Maiduguri, Maiduguri, Nigeria

<sup>b</sup> Department of Computer Science, Federal College of Education (Technical), Garki, Nigeria

<sup>c</sup> Department of Cyber Security Science, Federal University of Technology Minna, Minna, Nigeria

<sup>d</sup> Department of Computer Science, Federal University of Technology Akure, Akure, Nigeria

<sup>e</sup> Department of Electrical Engineering, University of Ibadan, Ibadan, Nigeria

JSIT  
20,3

298

Received 2 November 2017  
Revised 19 March 2018  
15 March 2018  
21 March 2018  
24 May 2018  
Accepted 26 July 2018

## Spam classification: a comparative analysis of different boosted decision tree approaches

Shrawan Kumar Trivedi

*Indian Institute of Management Sirmour, Sirmour, India, and*

Prabin Kumar Panigrahi

*Indian Institute of Management Indore, Indore, India*



## A lifelong spam emails classification model

Rami Mustafa A. Mohammad

Department of Computer Information Systems, College of Computer Science and Information Technology, Imam Abdulrahman Bin Faisal University, P.O. Box 1982, Dammam, Saudi Arabia





<https://uspdigital.usp.br/jupiterweb/obterDisciplina?sgldis=SME0823&codcur=55060&codhab=1>



The screenshot shows a web browser window displaying the USP Jupiter system. The browser's address bar shows the URL: <https://uspdigital.usp.br/jupiterweb/obterDisciplina?sgldis=SME0823&codcur=55060&codhab=1>. The page header includes the USP logo and the text "Universidade de São Paulo Brasil". A navigation menu on the left lists "Pública" (with sub-items: Calendário Escolar, Cursos de ingresso, Disciplina, Editais, FAQ) and "Acesso Restrito" (with sub-items: Entrar, Esqueci a Senha, Primeiro Acesso). The main content area is titled "Informações da Disciplina" and features a "Preparar para impressão" button. The central text reads: "Júpiter - Sistema de Gestão Acadêmica da Pró-Reitoria de Graduação", "Instituto de Ciências Matemáticas e de Computação", "Matemática Aplicada e Estatística", and "Disciplina: SME0823 - Modelos de Regressão e Aprendizado Supervisionado II" (with the English translation "Regression Models and Supervised Learning II" below). A table of course details is provided:

<b>Créditos Aula:</b>	4
<b>Créditos Trabalho:</b>	1
<b>Carga Horária Total:</b>	90 h
<b>Tipo:</b>	Semestral
<b>Ativação:</b>	01/01/2020
<b>Desativação:</b>	

**Objetivos**  
Apresentar conceitos introdutórios de modelos lineares generalizados em um enfoque computacional.  
*Present introductory concepts of generalized linear models in a computational approach.*

## Objetivos gerais da disciplina

- Introduzir a ideia de Aprendizado Supervisionado
- Apresentar conceitos introdutórios dos modelos lineares generalizados considerando um enfoque computacional.
- Introduzir Enfoque Bayesiano (Opcional)
- Promover a apresentação de relatórios e discussão de resultados com dados aplicações em dados reais

## Tópicos parte I

- O que é Aprendizado estatístico
- Aprendizado Supervisionado versus Aprendizado Não-Supervisionado
- Regressão versus Classificação
- Avaliando a Acurácia de um Modelo
- Uso do R

## Tópicos parte II

- Família Exponencial com um parâmetro
- Modelo Linear Generalizado
- Técnicas de Diagnóstico/ Verificação de Ajuste do Modelo

## Tópicos parte III

- Modelos para Dados Positivos Assimétricos
- Modelos para Dados Binários
- Modelos para Dados de Contagem

**HABILIDADES A SEREM DESENVOLVIDAS**

Habilidades	Estatística	Ciência de Dados
1. Entendimento e discussão da lógica do Conceito de Aprendizado Supervisionado e Modelos Lineares Generalizados	X	X
2. Cálculo e derivação das propriedades dos MLG e suas formas de estimação	X	
4. Uso da metodologia da análise de Aprendizado e Modelos de Regressão II em situações reais usando dados de textos (Modelamento)	X	
6. Uso de ferramentas computacionais, especialmente uso do R e elaboração de relatórios de análise	X	X
5. Aplicação das técnicas de Aprendizado e Modelos de Regressão II em situações reais	X	X
7. Identificação de tópicos adicionais associados	X	X

## O curso Bacharelado em Estatística e Ciência de Dados

Mais informações

<https://www.icmc.usp.br/graduacao/estatistica-bacharelado>

The screenshot shows the website for the Bachelor's degree in Statistics and Data Science at ICMC USP. The page features a dark blue header with the ICMC USP logo and navigation links for Institutional, People, Partnerships, and Contact. A search bar is also present. Below the header, the main content area is titled 'Estatística e Ciência de Dados (bacharelado)'. It provides key information: 40 vacancies (28 via FUVEST and 12 via SISU/ENEM), a 4-year duration, and details on how to apply via FUVEST. A sidebar on the left offers links to 'Veja também', 'Cursos', 'Estude conosco', 'Ingresso', 'Bolsas e auxílios', 'Oportunidades na graduação', 'Secretarias acadêmicas', 'FAQ e depoimentos', and 'Venha pro ICMC'. The main text under 'SOBRE O CURSO' describes the program as a rigorous course in exact sciences with a strong foundation in mathematics and computer science. An image of a calculator and a pen is visible on the right side of the page.

## **Como citar este documento**

Bazán, J. L. (2021). Modelos de regressão para classificar e-mails como spam. Uma introdução à disciplina SME0823 Modelos de Regressão e Aprendizado Supervisionado II. Programa Aulas Abertas - Departamento de Matemática Aplicada e Estatística - ICMC – USP. 16 de agosto de 2021. Disponível em [Download](#).