DOUGLAS VINÍCIUS GONÇALVES ARAÚJO

Modelo de Regressão Logística Aplicada a Previsão de Inadimplência sobre Cartão de Crédito de uma Instituição Financeira

JI-PARANÁ

2022

DOUGLAS VINÍCIUS GONÇALVES ARAÚJO

Modelo de Regressão Logística Aplicada a Previsão de Inadimplência sobre Cartão de Crédito de uma Instituição Financeira

Relatório de Estágio Supervisionado apresentado como Trabalho de Pesquisa à Coordenação do Curso de Bacharelado em Estatística da Universidade Federal de Rondônia.

UNIVERSIDADE FEDERAL DE RONDÔNIA – UNIR
DEPARTAMENTO DE MATEMÁTICA E ESTATÍSTICA
RELATÓRIO DE PESQUISA

JI-PARANÁ
2022

"Os livros servem para nos lembrar quanto somos estúpidos e tolos. São o guarda pretoriano de César, cochichando enquanto o desfile ruge pela avenida: Lembre-se, César, tu és mortal. A maioria de nós não pode sair correndo por aí, falar com todo mundo, conhecer todas as cidades do mundo, não temos tempo, dinheiro ou tantos amigos assim. As coisas que você está procurando, Montag, estão no mundo, mas a única possibilidade que o sujeito comum terá de ver noventa e nove por cento delas está num livro".

- Fahrenheit 451 de Ray Douglas Bradbury

Resumo

O objetivo deste trabalho tem como aplicar uma análise de regressão logística a dados de cartões de crédito de uma instituição financeira do estado de Rondônia, de forma gerar um modelo logístico capaz de prever a probabilidade de inadimplência ou risco de o tomador não honrar com o crédito.

Palavras-chaves: Credit Scoring, Probabilidade de Default, KRegressão Logístico.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

ABNT	Associação Brasileira de Normas Técnicas
abnTeX	ABsurdas Normas para TeX

Lista de símbolos

Γ	Letra grega Gama
Λ	Lambda
ζ	Letra grega minúscula zeta
$f(x;\theta)$	Função de Densidade de Probabilidade
Π	Produtório

Sumário

1 Introdução

1.1 Objetivos

O objetivo deste trabalho é desenvolver um modelo de previsão de risco de inadimplência dos tomadores de cartões de créditos de uma instituição Financeira do Estado de Rondônia. Resumidamente, estamos interessados em construir um modelo preditivo que propõe efetivamente a decisão sobre o risco de crédito (ou modelo de Credit Scoring).

Neste contexto, vamos relacionar os seguintes objetivos específicos:

2 REFERENCIAL TEÓRICO

2.1 Credit Scoring

Segundo (SICSÚ, 2010), inúmeras tomadas de decisões precede a incertezas, com a concessão de crédito não se destingui disto, conceder crédito implica a possibilidade de perda. Razão está que o credor ao estimar a probabilidade de perda ajudará na sua tomada de decisão mais confiável. E este modelo de estimação chamado de credit scoring tem como objetivo de prever ou quantificar, na data da concessão de crédito, a probabilidade de perda em uma operação de crédito que denominanos **risco de crédito**.

2.2 Breve Introdução sobre Machine Learning

Uma definição básica sobre Machine Learning (Aprendizado de Máquina) é englobar um conjunto de regras com algoritmos e procedimentos que tem como objetivo de extrair informações apartir dos dados e dessas informações tomar uma decisão.

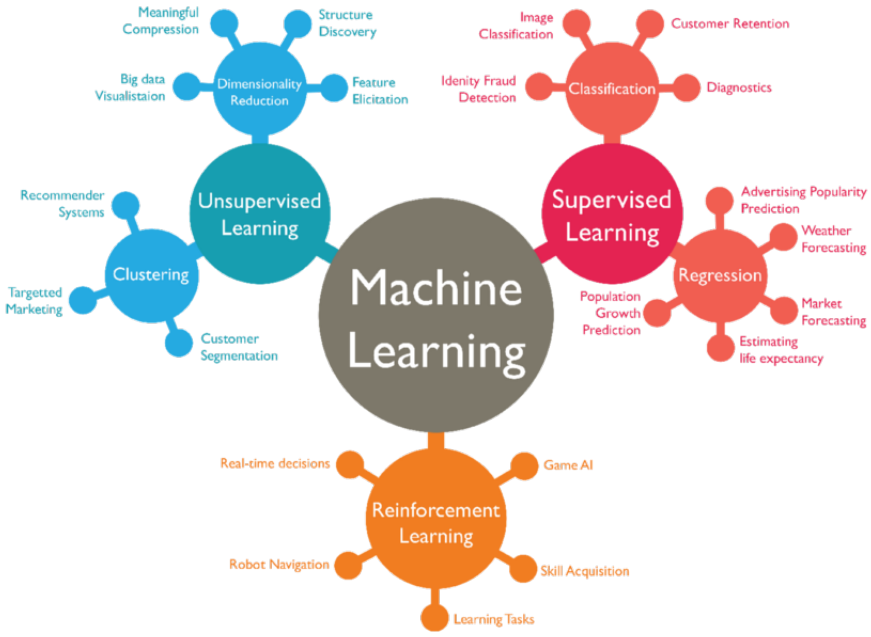
Segundo (GOODFELLOW; BENGIO; COURVILLE, 2016), os algoritmos de Machine Learning podem ser amplamente categorizados pelos tipos de aprendizagem, sitentizando essas diferenças no tipo de experiência durante o aprendizado do algoritmo.

- Supervisionado: O algoritmo procura relação entre as variáveis preditoras e a variável resposta de um *dataset*. Através dessas associação é possível realizar previsões quando o algoritmo é apresentado novos dados;
- Não-Supervisionado: aqui o algoritmo tem como objetivo agrupar os dados com base em características similares, descartando à apresentação da variavél resposta ao algoritmo;
- Aprendizagem por reforço: o algoritmo aprende com base nas interações com o ambiente. Não são apresentadas as ações que devem ser tomadas, apenas as consequências das ações.

2.3 Modelo de Regressão Logística

A regressão logística tem como principal uso modelar de uma variável binária $(0, 1)$, com base em mais variáveis, estas chamadas de variáveis explicativas ou preditoras. E comumentemente a variável resposta ou dependente, assim chama-se a variável binária

Figura 1 – Machine Learning e suas aplicações



Fonte: Learning (2022)

do modelo. Conforme (HILBE, 2016), o melhor modelo ajustado aos dados é assumido que:

- Não há correlação entre as variáveis preditoras;
- Estejam significativamente relacionados com a resposta;
- Que as observações dos dados não interferem entre si.

A resposta do modelo dito está conveniente a uma distribuição subjacente, ou seja, segue uma distribuição de Bernoulli. Concordantemente com (BOLFARINE; SANDOVAL, 2001), esta distribuição é um distribuição particular da distribuição Binomial que a função de probabilidade pode ser expressa:

$$f(x;\theta) = \theta_i^{x_i}(1-\theta_i)^{1-x_i}, \quad x_i = 0, 1, \tag{2.1}$$

em que $i = 1, \cdots, n$. Estes modelos são comumente empregados em situações que a resposta é dicotômica.

Porque não utilizar o modelo de regressão linear? Suponhamos uma situação, estamos tentando prever a condição médica de um paciente com três diagnósticos possíveis: acidente vascular cerebral (AVC), overdose de drogas e convulsões epilépticas. Podemos dar a essas condições valores como uma variável de resposta quantitativa:

$$Y = \begin{cases} 1, & \text{se AVC} \\ 2, & \text{se overdose} \\ 3, & \text{se convulsões} \end{cases}$$

Com essa converssão implica um ordenação dos resultados possíveis de Y, mas não há ordenação, pois se houvesse um ordenamento natural de leve, moderado e grave, da consideração a diferença de leve a moderado e entre moderado e grave seriam semelhantes os intervalos. Infelizmente, em geral, não há uma maneira de converter uma variável resposta qualitativa com mais de dois níveis em uma resposta quantitativa pronta para regressão linear.

Se tivermos uma resposta qualitativa binária (dois níveis), por exemplo, duas condições médicas do paciente e utilizando a variável *dummy* para codificar a resposta:

$$Y = \begin{cases} 1, & \text{se AVC} \\ 2, & \text{se overdose} \end{cases}$$

Mesmo usando a regressão linear para utilizar para obter uma estimativa de probabilidade do resultado, quebramos um pressuposto, pois algumas estimativas podem estar fora do intervalo $[0, 1]$.

E uma forma capaz de ter uma linha em forma de *"S"* para prever as probabilidades e descrever essa linha curva com os coeficientes da regressão linear.

Presuma que o modelo linear tradicional tenha a forma:

$$y_i = \mathbf{x'}_i\beta + \varepsilon_i \tag{2.2}$$

em que $\mathbf{x'}_i = [1, x_{i1}, x_{i2}, \cdots, x_{ik}]$, $\beta' = [\beta_0, \beta_1, \beta_2, \cdots, \beta_k]$ e a variável resposta tem valores entre o intervalo $[0, 1]$. Assumiremos qeu a variável resposta é uma variável aleatória com distribuição de Bernoulli com função de probabilidade dita anteriormente pela equação 2.1.

Uma vez que a $E(\varepsilon_i) = 0$, o valor esperado da variável resposta é:

$$E(y_i) = 1(\pi_i) + 0(1 - \pi_i) = \pi_i \tag{2.3}$$

o que implica em

$$E(y_i) = \mathbf{x'_i}\beta = \pi_i$$

2.4 Estimação dos Parâmetros

Para que se tenha um modelo ajustado, é imprescedível que seja feito a estimação dos parâmetros da regressão. Com isso utiliza-se o método de estimação de máxima de verossimilhança. Este método, a partir de um conjunto e um modelo estatístico, estima os valores dos parâmetros do modelo que mais máxima a probabilidade dos dados observados, ou seja, busca parâmetros que maximizem a função de verossimilhança. Condizente (Bolfarine), a definição da função de verossimilhança é:

Definição 2.4.1. Definição Sejam $X_1, \ldots, X_n$ uma amostra aleatória de tamanho n da variável aleatória X com função densidade $f(x|\theta)$, com $\theta \in \Theta$ é o espaço paramétrico. A função de verossimilhança de θ compatível com à amostra aleatória observada é dada por

$$L(\theta; x) = \prod_{i=1}^{n} f(x_i|\theta) \tag{2.4}$$

O estimador de máxima verossimilhança de θ é o valor $\theta \in \Theta$ que maximiza a função de verossimilhança $L(\theta; x)$.

Aplicando o logaritmo natural a função de verossimilhança

$$l(\theta; x) = \log L(\theta; x), \tag{2.5}$$

verificamos que o valor de θ

2.5 Interpretação dos Parâmetros

2.6 Testes de Significância

2.6.1 Teste da Razão de Verossimilhança (TRV)

2.6.2 Teste de Wald

2.7 Seleção de Variáveis

2.7.1 Desempenho dos Modelos

2.7.2 Curva ROC

3 Metodologia

4 Resultados e Discussões

5 Considerações Finais

Referências

BOLFARINE, H.; SANDOVAL, M. C. *Introdução à inferência estatística*. [S.l.]: SBM, 2001. v. 2. Citado na página 11.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep learning*. [S.l.]: MIT press, 2016. Citado na página 10.

HILBE, J. M. *Practical guide to logistic regression*. [S.l.]: crc Press, 2016. Citado na página 11.

LEARNING, A. I. T. M. 2022. Acesso em 26 de novembro de 2022. Disponível em: <https://becominghuman.ai/an-introduction-to-machine-learning-7db04da817c4>. Citado na página 11.

SICSÚ, A. L. *Credit Scoring: desenvolvimento, implantação, acompanhamento*. [S.l.]: Blucher, 2010. Citado na página 10.

Apêndices

APÊNDICE A – DESCRIÇÃO DAS VARIÁVEIS

Variável	Descrição da Variável	Tipo de Variável	Nº de Categorias	Categorias
Sexo	Sexo			
Estado Civl				
Escolaridade				
Idade				
Renda				
Patrimônio				
SM30				
SM60				
SM90				
SM180				
SM360				
Empréstimos				
Capital				
Aplicação				
Limite				
STATUS				

APÊNDICE B – Script em R

```
############################################################
############################################################
####                REGRESSÃO LOGISTICA                 ####
############################################################
############################################################
library()
library()
library()
library()
```

APÊNDICE C – Script em Python

```python
import numpy as np

def incmatrix(genl1,genl2):
    m = len(genl1)
    n = len(genl2)
    M = None #to become the incidence matrix
    VT = np.zeros((n*m,1), int)  #dummy variable

    #compute the bitwise xor matrix
    M1 = bitxormatrix(genl1)
    M2 = np.triu(bitxormatrix(genl2),1)

    for i in range(m-1):
        for j in range(i+1, m):
            [r,c] = np.where(M2 == M1[i,j])
            for k in range(len(r)):
                VT[(i)*n + r[k]] = 1;
                VT[(i)*n + c[k]] = 1;
                VT[(j)*n + r[k]] = 1;
                VT[(j)*n + c[k]] = 1;

                if M is None:
                    M = np.copy(VT)
                else:
                    M = np.concatenate((M, VT), 1)

                VT = np.zeros((n*m,1), int)

    return M
```

Listing C.1 – Python example