Universidade de Brasília
Departamento de Estatística

Interpretação de redes neurais

Davi Guerra Alves

Projeto apresentado para o Departamento de Estatística da Universidade de Brasília como parte dos requisitos necessários para obtenção do grau de Bacharel em Estatística.

Brasília
2023

Davi Guerra Alves

Interpretação de redes neurais

Orientador(a): Thais Carvalho Valadares Rodrigues

Projeto apresentado para o Departamento de Estatística da Universidade de Brasília como parte dos requisitos necessários para obtenção do grau de Bacharel em Estatística.

Brasília
2022

Sumário

1 Introdução . . . 5

2 Referencial teórico . . . 6

2.1 Regressão logística . . . 6

2.1.1 Interpretação . . . 7

2.2 Redes neurais artificiais . . . 7

2.2.1 Neurônio . . . 8

2.2.2 Arquitetura . . . 9

2.2.3 *Forward propagation* . . . 10

2.2.4 Função de perda . . . 11

2.2.5 *Backpropagation* . . . 12

2.3 SHAP . . . 13

2.3.1 Valores de Shapley . . . 14

2.3.2 Shapley Additive Explanations . . . 17

2.4 Medidas de associação . . . 19

2.4.1 Coeficiente de Pearson . . . 19

2.4.2 Coeficiente de contingência . . . 19

2.5 Métricas de avaliação . . . 19

2.5.1 Matriz de confusão . . . 20

2.5.2 Acurácia . . . 20

2.5.3 Precisão . . . 21

2.5.4 Recall . . . 21

2.5.5 F1-score . . . 21

3 Metodologia . . . 23

3.1 Conjunto de dados . . . 23

3.1.1 Variáveis . . . 23

3.1.2 Limpeza dos dados . . . 25

3.2 Métodos . . . 25

4 Resultados 27
4.1 Análise descritiva 27
4.1.1 Condição do empréstimo 27
4.1.2 Relação entre as covariáveis e a variável resposta 28
4.2 Regressão logística 33
4.3 Modelagem da rede neural 35
4.4 Interpretação da rede neural 36
4.5 Benchmark entre regressão logística e redes neurais 38
4.5.1 Complexidade da arquitetura 38
4.5.2 Resultado dos modelos 38
4.5.3 Tempo de execução 39
5 Conclusão 40
6 Anexo 41

1 Introdução

As redes neurais são modelos matemáticos que, unidos às técnicas computacionais, visam tentar reproduzir o funcionamento da estrutura neural presente no ser humano, buscando, assim, realizar tarefas complexas, como o reconhecimento de padrões, identificação de imagens, processamento de linguagem natural, etc. No entanto, apesar de sua eficácia em muitas aplicações, as redes neurais podem ficar muito complexas conforme sua arquitetura cresce, sendo consideradas como "caixas pretas", devido à sua complexidade e falta de transparência.

Por isso, a interpretação de redes neurais é uma área cada vez mais essencial, pois busca entender como esses modelos tomam decisões e quais fatores influenciam suas saídas. Entender o porquê uma rede neural tomou tal decisão é importante em diversas áreas, como a área da saúde, em diagnósticos médicos, e na área bancária, analisando um risco de crédito.

Uma das técnicas mais promissoras para a interpretação de redes neurais é o SHAP (Shapley Additive Explanations), que foi introduzido em 2017 "citeplundberg2017unified". O SHAP é uma técnica de interpretação que fornece explicações locais e globais para as saídas da rede neural. Ele é baseado no conceito matemático de valor de Shapley "citepshapley1953value", que atribui uma contribuição de importância para cada recurso de entrada na saída da rede neural.

Portanto, esse trabalho tem como objetivo explorar a técnica SHAP para a interpretação de redes neurais e sua aplicação em diversas áreas, pois, ao compreender como as redes neurais funcionam, e quais são os recursos mais importantes para suas decisões, será possível tornar o método mais confiável e transparentes para os usuários.

2 Referencial teórico

2.1 Regressão logística

A regressão logística é um método estatístico utilizado para modelar a probabilidade de uma variável dependente categórica. É comumente utilizada para problemas de classificação binária, onde a variável dependente possui apenas duas categorias, como sim/não, positivo/negativo, 0/1.

O cálculo da regressão logística é baseado na probabilidade da variável aleatória Y ser igual a 1, onde Y é uma variável aleatória com distribuição Bernouli, com parâmetro p de sucesso, cuja fórmula é dada por:

$$P(Y = 1|x_1, x_2, ..., x_k) = \frac{1}{1 + e^{-(\beta_0 + x_1\beta_1 + x_2\beta_2 + ... + x_k\beta_k)}} \tag{2.1.1}$$

onde cada variável explicativa $(x_1, x_2, ..., x_k)$ tem um parâmetro β correspondente, influenciando o resultado de Y.

A estimação dos coeficientes $(b_0, b_1, b_2, ..., b_k)$ na regressão logística é geralmente realizada por meio do método da máxima verossimilhança. O objetivo é encontrar os valores dos coeficientes que maximizam a função de verossimilhança, representando a probabilidade de observar os dados observados dado o modelo.

A função de verossimilhança (L) para a regressão logística é dada pelo produto das probabilidades condicionais de observar os eventos(valores da variável dependente) dados os valores das variáveis independentes. Para facilitar o cálculo, geralmente trabalhamos com o logaritmo natural da função de verossimilhança, conhecido como log-verossimilhança(l).

A log-verossimilhança para a regressão logística é:

$$l(\beta) = \sum_{i=1}^{N} [y_i \beta^T x_i - \log(1 + e^{\beta^T x_i})] \tag{2.1.2}$$

- Né o número total de observações.
- y_i é a variável dependente binária da i-ésima observação (0 ou 1).
- p_i é a probabilidade predita de $Y = 1$ para a i-ésima observação, dada pela função logística.

A ideia é encontrar os valores de $(b_0, b_1, b_2, ..., b_k)$ que maximizam essa função.

Isso geralmente é feito usando métodos computacionais, como o algoritmo de otimização Newton-Raphson ou o Gradiente Descendente.

2.1.1 Interpretação

Para se interpretar o modelo logístico é utilizado a Razão de Chances, que calcula a razão da probabilidade de um evento ocorrer em um grupo em relação à probabilidade de não ocorrer.

Esse resultado pode ser obtido quando se calcula a exponencial dos coeficientes do modelo logístico.

$$RC = \exp(\beta_k) \tag{2.1.3}$$

Um RC igual a 1 indica que a variável independente não tem efeito no resultado de Y(nenhuma associação). Um RC maior que 1 sugere uma associação positiva, enquanto um RC menor que 1 sugere uma associação negativa.

Outra medida interpretativa é a função log odds ou logíto. Ela é uma função que calcula o log da razão do evento acontecer e dele não acontecer, cuja formulação é dada por:

$$\text{logit}(P(Y=1)) = \log\left(\frac{P(Y=1)}{1-P(Y=1)}\right) \tag{2.1.4}$$

onde esse resultado nada mais é do que $\beta_0 + X_1\beta_1 + X_2\beta_2 + \ldots + X_k\beta_k$.

Portanto, a utilidade de se analisar o log-odds é justamente uma ponte entre olhar coeficiente e a probabilidade final, pois um coeficiente positivo indica que o aumento na variável está associado a um aumento nas log-odds (e, portanto, na probabilidade), enquanto um coeficiente negativo está associado a uma diminuição nas log-odds (e na probabilidade).

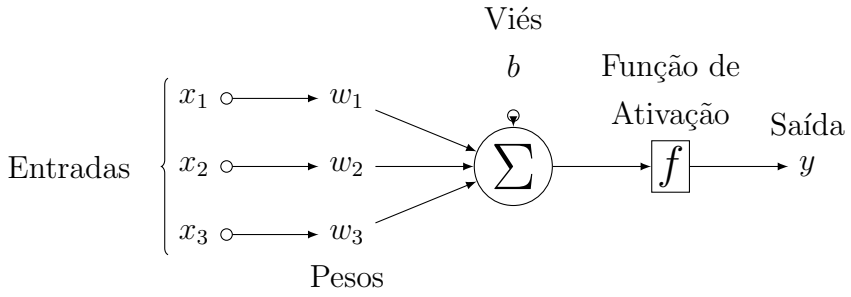
2.2 Redes neurais artificiais

Redes Neurais Artificiais (ou *Deep Learning*) é uma técnica preditiva presente no campo de Inteligência Artificial. As redes neurais tem sido amplamente utilizadas devido ao seu alto poder preditivo e também à flexibilidade de se aplicar esse método em diversos contextos, permitindo ser um modelo com menos restrições que os modelos tradicionais estatísticos.

2.2.1 Neurônio

Uma rede neural tem esse nome devido à tentativa de se reproduzir o comportamento do cérebro humano. Sua arquitetura é composta por um conjunto de unidades denominadas neurônios, e cada neurônio é responsável por receber informações, fazer o tratamento do que foi recebido, e repassar o resultado disso para frente. A Figura 2.2.1 ilustra a estrutura de 1 neurônio. Quando as informações x_i entram no neurônio, acontece primeiramente um processo onde é ponderada cada informação que foi recebida, os chamados **pesos**. Logo em seguida ocorre a soma dessa ponderação. Feito isso, é realizado mais um processo de soma, agora adicionando uma informação própria daquele neurônio nesse resultado. Essa informação é chamada de **bias** (ou Viés). Antes desse resultado ser repassado para outro neurônio, ele passa por uma função que vai definir a natureza daquela informação, chamada de **função de ativação**, retornando assim uma saída y.

Figura 1: Neurônio da Rede Neural

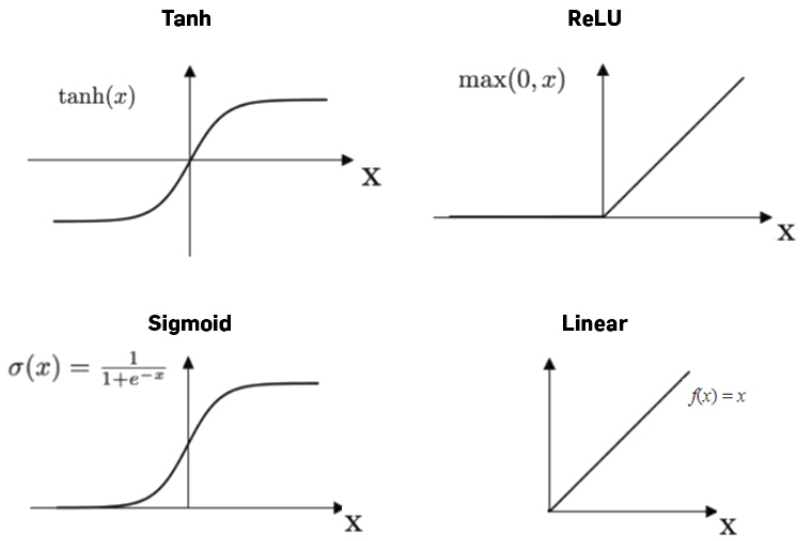


A Figura 1 pode ser representada matematicamente da seguinte maneira:

$$y = f(\beta + \sum_{i=1}^{d_x} w_i x_i) \tag{2.2.1}$$

onde d_x é o número de entradas.

Figura 2: Tipos de Função de Ativação.



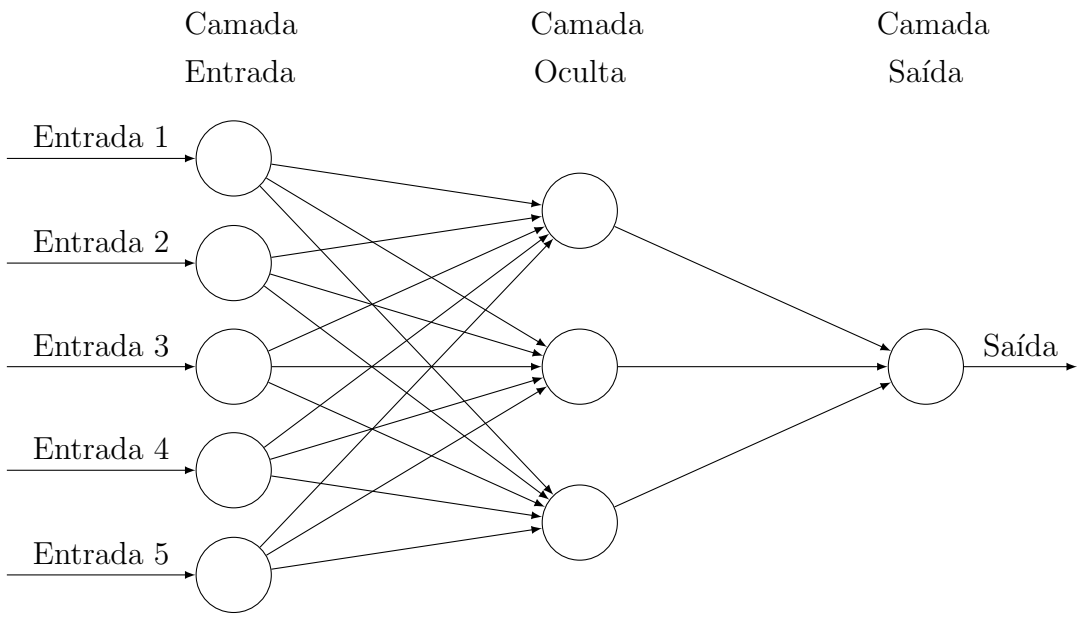
Fonte: https://machine-learning.paperspace.com/wiki/activation-function

A Figura 2 mostra alguns tipos de funções de ativação que um neurônio pode ser atribuído. Note como a maioria dessas funções restringe o valor de x (nesse caso o valor calculado no neurônio), no caso da função Sigmoide e da Tangente hiperbólica (tanh) limitando o valor do neurônio em um intervalo, a ReLU, que é bastante utilizada, desconsidera os valores negativos e existe a função Linear que basicamente só vai repassar a informação do neurônio para frente.

2.2.2 Arquitetura

Uma rede neural é estruturada em camadas formadas por um conjunto de neurônios. Conforme ilustrado na Figura 3, temos as camadas de entrada, as camadas ocultas e a camada de saída. A camada de entrada é o ponto de partida da rede neural, pois é onde as informações das variáveis entram. Logo em seguida encontram-se as camadas ocultas, que são as principais responsáveis por criar redes mais complexas, pois o número de camadas e o número de neurônio dentro dessas camadas podem ser moldados ou adicionados dependendo do objetivo empregado pela rede, conforme ilustrado na Figura 4. E por fim existe a camada de saída que contém o(s) valor(es) predito(s) pela rede.

Figura 3: Rede Neural com uma camada oculta



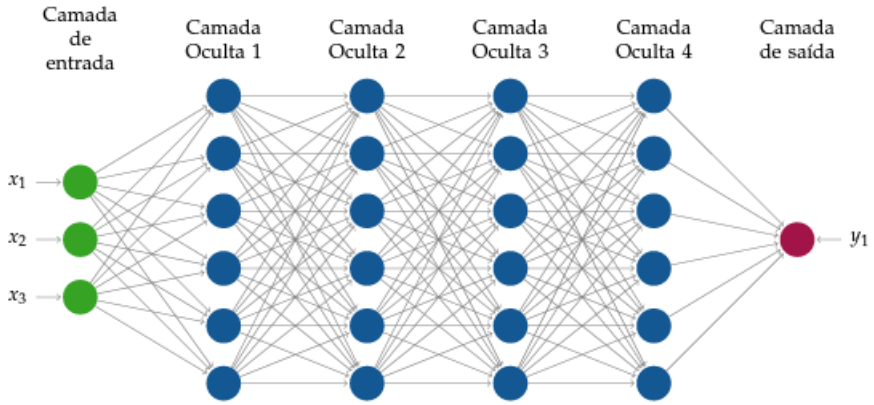


Figura 4: Arquitetura padrão de um rede neural *feedforward* (IZBICKI; SANTOS, 2020)

Note que as Figuras 3 e 4 evidenciam um potencial muito grande de crescimento da rede e naturalmente esse aumento pode acabar gerando um custo computacional elevado quando a rede estiver em treinamento.

2.2.3 *Forward propagation*

O processo *Forward propagation* (ou propagação direta) é o responsável por transmitir as informações, desde a camada de entrada, passando pelas camadas ocultas, até

chegar na camada de saída. O *Forward propagation* utiliza da generalização a Equação 2.2.1 para cada neurônio presente nas camadas internas da rede neural. Por isso temos que, para cada j-ésimo neurônio, da camada l:

$$z_j^{(l)} = b_j^{(l)} + \sum_{i=1}^{d_{(l-1)}} w_{ij} a_i^{(l-1)}$$

onde:

- w_{ij} é o peso associado à conexão entre o neurônio i na camada $l-1$ e o neurônio i na camada l;
- $a_{(i)}^{(l-1)}$ é a saída do neurônio i na camada anterior $(l-1)$;
- $b_j^{(l)}$ é o viés (bias) associado ao neurônio j na camada l.

Logo em seguida é aplicada uma função de ativação g em $z_i^{(l)}$ que vai ser a responsável por gerar o resultado final $a_i^{(l)}$, do o i-ésimo neurônio na l-ésima camada.

$$a_i^{(l)} = g(z_i^{(l)})$$

Esse processo vai ser realizado camada a camada, sequencialmente. Logo, supondo que uma rede neural tenha l camadas ocultas e, cada camada contendo d_l neurônios, considerando também w_{ij} como o peso presente no i-ésimo neurônio com a j-ésima saída na camada seguinte $(l+1)$, onde $l = 0, ..., H$. Temos que o resultado final da propagação é igual a:

$$f(\mathbf{x}) = \mathbf{a}^{H+1} = g(b_j^{(H+1)} + \sum_{i=1}^{d_H} w_{ij} a_i^H) \tag{2.2.2}$$

Note que a previsão da rede vem diretamente do resultado obtido da última camada oculta, e esse depende da camada que o antecede e assim sucessivamente até chegar na camada de entrada.

2.2.4 Função de perda

Para se obter informações sobre o desempenho do modelo, é escolhida uma função de perda. Uma função bastante utilizada é a do erro do quadrático médio:

$$EQM(f) = \frac{1}{n} \sum_{k=1}^{n} (f(\mathbf{x}_k) - y_k)^2$$

Essa função é uma indicadora do quão longe, em média, os valores preditos estão distantes dos valores reais. Note que o resultado da função f depende exclusivamente dos parâmetros da rede (viés e pesos), por isso, se essa função de perda tende a 0, significa que os parâmetros dessa rede alcançaram um ponto mínimo global. Entretanto, devido a complexidade desse modelo, acabam-se escolhidos pontos locais mínimos, que, dependendo do contexto, acabam satisfazendo o objetivo. A Figura 5 ilustra esse comportamento:

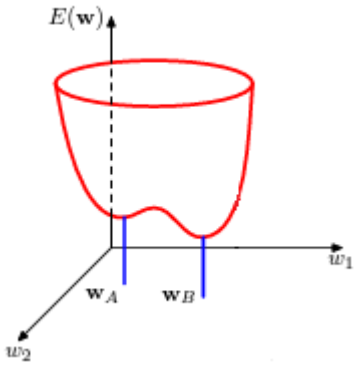


Figura 5: Comportamentos dos pesos em relação à função de perda. O ponto w_A representa um ponto local mínimo e w_B representa um ponto global mínimo. (BISHOP, 2006)

2.2.5 *Backpropagation*

Como a função de perda está relacionada com os parâmetros (θ) da rede, para se minimizar a função de perda R(θ), é necessário encontrar os valores de θ que resolvam esse problema de otimização. Para fazer isso, é necessário calcular o gradiente de R(θ) em relação à θ (JAMES et al., 2013),

$$\nabla R(\theta) = \frac{\partial R(\theta)}{\partial \theta} \tag{2.2.3}$$

A rede neural, durante todo o treinamento, aplica esse processo do cálculo do gradiente de R(θ) em relação à θ. Esse é um processo iterativo, com o objetivo de mudar o valor de θ afim de conseguir minimizar a função de perda. Com isso a Equação 2.2.3 pode ser descrita nesse processo iterativo como:

$$\nabla R(\theta^m) = \left.\frac{\partial R(\theta)}{\partial \theta}\right|_{\theta=\theta^m},$$

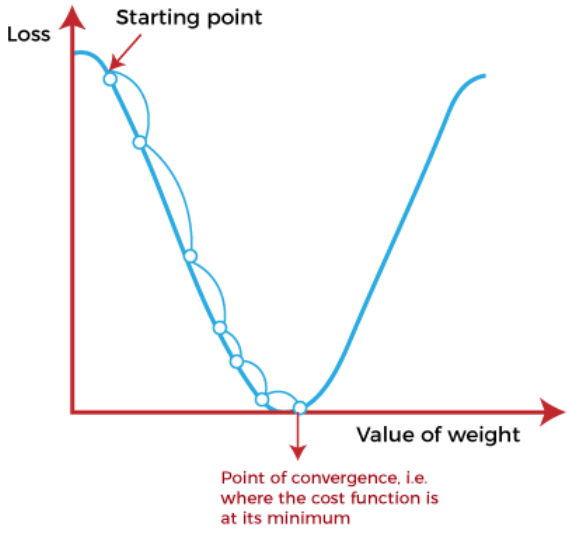
onde $\theta = \theta_m$ significa que o cálculo do gradiente está sendo realizado na iteração m.

E, para conseguir atualizar esse θ, conforme é calculado o gradiente durante as iterações, é utilizada a técnica de gradiente descendente, que pode ser descrita como:

$$\theta^{m+1} \leftarrow \theta^m - \lambda \frac{\partial R(\theta^m)}{\partial \theta^m}$$

sendo λ o parâmetro que vai definir a magnitude de influência da derivada $\frac{\partial R(\theta^m)}{\partial \theta^m}$ em θ^m.

Figura 6: Representação do método do gradiente descendente para a estimação de um parâmetro.



Fonte: https://www.javatpoint.com/gradient-descent-in-machine-learning

A Figura 6 demonstra o processo do gradiente descendente. Os parâmetros são iniciados com algum valor e, conforme ocorre os processos iterativos de aprendizado, o parâmetro converge para um mínimo da função de perda. Note que a distância entre cada ponto é definida pelo λ ou taxa de aprendizado.

Todo esse processo é realizado em cada parâmetro que existe na rede neural. Assim como as informações das variáveis são passadas camada a camada, saindo da camada de entrada, passando pelas camadas ocultas e chegando na camada de saída, visto anteriormente como *Forward propagation*, a informação do resultado da rede na função de perda é passada de forma contrária. O gradiente de cada parâmetro é calculado primeiro nas camadas mais próximas da saída, e essa informação é repassada para trás, chegando até os parâmetros próximos aos da camada de entrada. Esse processo é chamado de *Backpropagation* (WERBOS, 1974).

2.3 SHAP

A estrutura de uma rede neural, por mais que proporcione bons resultados, mostra uma deficiência na parte interpretativa. Conhecida por ser uma "caixa-preta"pelo fato de sua estrutura ser muito complexa, existe a necessidade de se entender as predições

feitas. Para isso, existem técnicas que abordam o tema de interpretação de modelos de redes neurais e dentro delas existe a técnica SHAP, que através dela é possível entender como as variáveis de entrada influenciam as previsões do modelo, fornecendo *insights* sobre sua lógica e permitindo uma explicação clara e confiável. Isso contribui para a transparência, confiabilidade e aceitação dos modelos, além de auxiliar na detecção de viéses e discriminação.

2.3.1 Valores de Shapley

Os valores de Shapley foram desenvolvidos por Lloyd Shapley (SHAPLEY, 1953) no contexto da teoria de jogos, e essa técnica ganhou força na área de inteligência artificial pela sua capacidade de conseguir interpretar modelos preditivos tidos como "caixa-preta". No método criado por Shapley, existiam uma quantidade de jogadores que exerciam juntos determinada atividade, e o intuito era observar o ganho que um jogador (ou um conjunto de jogadores), obtinha ao ser adicionado para realizar a mesma tarefa, sem a presença do restante do grupo.

Podemos definir $\mathbf{F}$ como o conjunto de jogadores (ou as variáveis explicativas) presentes na atividade, logo $\mathbf{F} = \{1, 2, ..., \mathbf{M}\}$, onde $\mathbf{M}$ é o número de variáveis. Definindo $\mathbf{S}$ como uma coligação do conjunto $\mathbf{F}$ ($\mathbf{S} \subseteq \mathbf{F}$), temos, por exemplo, as seguintes possibilidades de $\mathbf{S}$, quando $\mathbf{M}$ é igual a 3:

$$\{\{\emptyset\}, \{1\}, \{2\}, \{3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}, \{1, 2, 3\}\}$$

Podemos definir também ν como uma função que vai mapear um conjunto de valores e retornar um número real. Com isso, o retorno de $\nu(\mathbf{S})$ é um número real que pode ser definido como o "trabalho da coligação $\mathbf{S}$ ". Esse valor é equivalente ao total ganho que os jogadores podem obter caso trabalhem juntos em uma determinada coligação.

Para calcular o ganho ao adicionar uma variável i ou a importância do jogador em específico, pode-se calcular o ganho quando é adicionada aquela variável na coligação menos a coligação sem a adição daquela variável, ficando da seguinte maneira:

$$\nu(\mathbf{S} \cup \{i\}) - \nu(\mathbf{S})$$

No exemplo acima, caso queiramos calcular o efeito da coligação $\{3\}$, poderíamos fazer:

$$\text{Contribuição } de\ \{3\} = \nu(\{1, 2, 3\}) - \nu(\{1, 2\})$$

Mas suponha que as variáveis (ou jogadores) {2} e {3} sejam extremamente semelhantes. Quando é calculado o ganho após inserir {2} na coligação {1,2}, é possível notar um aumento substancial, mas quando é adicionado {3} na coligação {1,2,3}, o ganho obtido é muito pouco. Como as variáveis exercem um papel parecido, o ganho maior ficou sujeito à variável que foi adicionada primeiro na coligação, não necessariamente porque uma é mais importante que a outra. Por isso, para calcular o real ganho da variável {i}, é necessário testar todas as permutações de **F** (conjunto de jogadores) e obter a contribuição de {i} em cada uma delas, para então fazer a média dessas contribuições. Por exemplo, definindo **F** = {1,2,3,4}, suponha que estamos interessados em calcular a contribuição de {3}, logo, podemos obter a seguinte permutação de **F**:

$$[3, 1, 2, 4]$$

Calculando a contribuição de {3}, temos:

$$\nu(\{3\}) - \nu(\emptyset)$$

Outra permutação poderia ser :

$$[2, 4, 3, 1]$$

Calculando a contribuição de {3}, nessa permutação temos:

$$\nu(\text{coligação } de\ [2, 4, 3]) - \text{coligação } de\ [2, 4])$$

Uma observação deve ser feita: a função ν considera a coligação como argumento, não a permutação. A coligação é um conjunto, com isso a ordem dos elementos não importa, mas a permutação é uma coleção ordenada de elementos. Na permutação do tipo [3,1,2,4], 3 é a primeira variável adicionada e 4 é a última. Por isso, para cada permutação a ordem dos elementos pode mudar a contribuição do total ganho, contudo o total ganho da permutação somente depende dos elementos, não da ordem. Logo:

$$\nu(\text{coligação } de\ [3, 1, 2, 4]) = \nu(\{1, 4, 2, 3\})$$

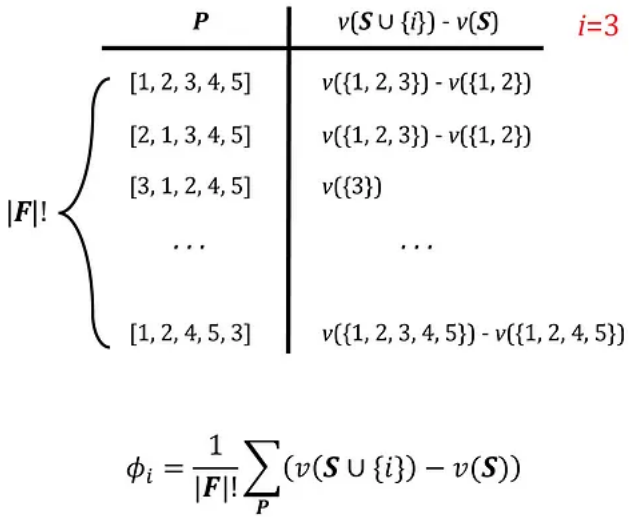
Sendo assim, para cada permutação **P**, é preciso primeiro calcular o ganho da coligação das variáveis que foram adicionadas antes de {i}, e esse conjunto pode ser chamado de coligação **S**. Feito isso, agora é preciso calcular o ganho das coligações que são formadas ao adicionar {i} em **S**, e podemos chamar isso de $\mathbf{S} \cup \{i\}$. Com isso, a

contribuição da variável {i}, denotada por ϕ_i, é:

$$\phi_i = \frac{1}{|\mathbf{F}|!} \sum_{\mathbf{P}} [\nu(\mathbf{S} \cup \{i\}) - \nu(\mathbf{S})] \tag{2.3.1}$$

O número total de permutações de **F** é $|\mathbf{F}|!$. Logo, podemos dividir a soma das contribuições por $|\mathbf{F}|!$ para encontrar o valor esperado de contribuição de {i}. A Figura 7 mostra como é feito esse calculo para um determinado jogador $\{i\}$.

Figura 7: Ganho do jogador 3 em relação à todas as permutações de jogadores.



$$\phi_i = \frac{1}{|F|!} \sum_{P} (v(S \cup \{i\}) - v(S))$$

Fonte: https://towardsdatascience.com/introduction-to-shap-values-and-their-application-in-machine-learning-8003718e6827

É possível perceber que algumas permutações possuem a mesma contribuição, desde que suas coligações $\mathbf{S} \cup \{i\}$ e **S** sejam as mesmas. Com isso, para reduzir o processo do cálculo de contribuição de cada permutação, pode-se identificar quantas vezes a permutação gerada vai resultar em uma contribuição que seja igual a outra.

Para fazer isso, é necessário descobrir quantas permutações podem ser formadas de cada coligação. Podemos definir $\mathbf{F} - \{i\}$ como o conjunto de todas as variáveis excluindo a variável {i}, e **S** como uma das coligações de $\mathbf{F} - \{i\} (\mathbf{S} \subseteq \mathbf{F} - \{i\})$.

Logo, para cada coligação **S** temos $|\mathbf{S}|!$ possíveis permutações, que corresponde às possibilidades de variáveis e suas respectivas ordens antes de adicionar a variável {i}.

Tendo os conjuntos $\mathbf{S} \cup \{i\}$ e **S** definidos, resta agora achar as possíveis permutações das variáveis restantes. E para saber o valor restante é preciso calcular o tamanho do conjunto gerado por: **F** - $(\mathbf{S} \cup \{i\} + 1)$ que basicamente é o que resta das variáveis para completar o conjunto **F**.

A Figura 8 mostra o que acontece quando se escolhe o jogador i, nesse caso $i = 3$. Note que, na linha das coligações, é definido as possíveis coligações de $\mathbf{S}$, que seria as permutações dos jogadores 1 e 2, temos a coligação de um único elemento na coluna $\{i\}$, que sempre vai ser o próprio elemento, em seguida a coligação dos jogadores restantes. Na linha das permutações é definida todas as possíveis permutações para $\mathbf{S}$, para $\{i\}$ e para $\mathbf{F} - \mathbf{S} - \{i\}$. E na última linha é representado o tamanho do conjunto formado pela permutação/coligação descrita anteriormente.

Figura 8: Relação entre permutações e coalizões.

	S	+	$\{i\}$	+	F-S-$\{i\}$	=	F
Coalitions	{1, 2}	+	{3}	+	{4, 5}	=	{1, 2, 3, 4, 5}
Permutations	[1, 2] [2, 1]	+	[3]	+	[4, 5] [5, 4] [4, 5] [5, 4]	= = = =	[1, 2, 3, 4, 5] [1, 2, 3, 5, 4] [2, 1, 3, 4, 5] [2, 1, 3, 5, 4]
Number of Permutations	$\|S\|!$	+	1	+	$(\|F\|-\|S\|-1)!$	=	$\|S\|!(\|F\|-\|S\|-1)!$

Fonte: https://towardsdatascience.com/introduction-to-shap-values-and-their-application-in-machine-learning-8003718e6827

Com isso, podemos reescrever a Equação 2.3.1 da seguinte maneira:

$$\phi_i = \sum_{\mathbf{S} \subseteq \mathbf{F} - \{i\}} \frac{|\mathbf{S}|!(|\mathbf{F}| - |\mathbf{S}| - 1)!}{|\mathbf{F}|!} [\nu(\mathbf{S} \cup \{i\}) - \nu(\mathbf{S})],$$

onde ϕ_i é o valor de shapley para a variável {i}.

2.3.2 Shapley Additive Explanations

Fazendo a relação do valor de Shapley para o SHAP (Shapley Additive Explanations), temos que a função característica ν é equivalente à função $f(x)$ responsável por fazer as predições. E os valores de SHAP são calculados a partir das observações que entram no modelo. Com isso, a fórmula do valor de SHAP, para cada conjunto de observação e variável especificada, se dá por:

$$\phi_i(f, \mathbf{x}) = \sum_{\mathbf{S} \subseteq \mathbf{F} - \{i\}} \frac{|\mathbf{S}|!(|\mathbf{F}| - |\mathbf{S}| - 1)!}{|\mathbf{F}|!} [f_{\mathbf{S} \cup \{i\}}(\mathbf{x}_{\mathbf{S} \cup \{i\}}) - f_{\mathbf{S}}(\mathbf{x}_{\mathbf{S}})]$$

Perceba que $f_{\mathbf{S}}(\mathbf{x}_S)$ representa o resultado do modelo com somente as variáveis que estão na coligação $\mathbf{S}$, algo que na realidade não é permitido na maioria dos modelos. Por isso, uma aproximação desse resultado é a seguinte:

$$f_S(\mathbf{x}_S)) \approx E[f(\mathbf{x}|\mathbf{x}_S)] \approx \frac{1}{k} \sum_{i=1}^{k} f(\mathbf{x}_{\bar{S}}^{(i)}, \mathbf{x}_S) \tag{2.3.2}$$

Figura 9: Cálculo de $f_{\mathbf{S}}$, sendo S o conjunto de variáveis X_1, X_3, X_4, dentre as observações de um conjunto de dados.

$\boldsymbol{x} = \{x_1, x_2, x_3, x_4, x_5\}$ $\boldsymbol{x_S} = \{x_1, x_3, x_4,\}$ $\boldsymbol{x_{\bar{s}}} = \{x_2, x_5\}$

X_1	X_2	X_3	X_4	X_5	$\frac{f\left(\boldsymbol{x}_{\bar{s}}^{(i)}, \boldsymbol{x}_s\right)}{}$
$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	$x_4^{(1)}$	$x_5^{(1)}$	$f\left(x_1, x_2^{(1)}, x_3, x_4, x_5^{(1)}\right)$
$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	$x_4^{(2)}$	$x_5^{(2)}$	$f\left(x_1, x_2^{(2)}, x_3, x_4, x_5^{(2)}\right)$
...	...	...	...	...	...
$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$x_4^{(k)}$	$x_5^{(k)}$	$+ f\left(x_1, x_2^{(k)}, x_3, x_4, x_5^{(k)}\right)$
					$\sum_{i=1}^{k} f\left(\boldsymbol{x}_{\bar{s}}^{(i)}, \boldsymbol{x}_s\right)$

$$f_S(x_S) \approx E[f(\boldsymbol{x})|\boldsymbol{x_S}] \approx \frac{1}{k} \sum_{i=1}^{k} f\left(\boldsymbol{x}_{\bar{s}}^{(i)}, \boldsymbol{x}_s\right)$$

Fonte: https://towardsdatascience.com/introduction-to-shap-values-and-their-application-in-machine-learning-8003718e6827

A Figura 9 representa, para cada observação do conjunto de dados, a Equação 2.3.2. Neste caso, as variáveis X_1, X_3 e X_4 representam o conjunto $\mathbf{S}$, e escolhendo um valor x_1, x_3 e x_4 dessas variáveis, respectivamente, calcula-se f para cada observação do conjunto de dados, travando x_1, x_3 e x_4 na função e utilizando o valor das variáveis complementares, que neste caso são os valores de X_2 e X_5, em suas respectivas observações. Feito isso, é calculada média desses valores que corresponde justamente com o resultado da função $f_{\mathbf{S}}(x_{\mathbf{S}})$.

2.4 Medidas de associação

2.4.1 Coeficiente de Pearson

O coeficiente de Pearson, também conhecido como correlação de Pearson, é uma medida estatística que avalia a relação linear entre duas variáveis contínuas.

$$\rho = \frac{cov(X,Y)}{\sqrt{var(X)var(Y)}} \tag{2.4.1}$$

Este coeficiente varia de -1 a 1, onde -1 indica uma relação linear negativa perfeita, 1 indica uma relação linear positiva perfeita, e 0 indica ausência de relação linear.

2.4.2 Coeficiente de contingência

O coeficiente de contingência é uma medida estatística utilizada para avaliar a associação entre duas variáveis categóricas em uma tabela de contingência. Ele é especialmente útil quando se trabalha com dados categóricos, fornecendo uma indicação da força e direção da associação entre as variáveis.

$$C = \sqrt{\frac{\chi^2}{N + \chi^2}} \tag{2.4.2}$$

O coeficiente de contingência varia de 0 a 1, onde 0 indica nenhuma associação e 1 indica uma associação perfeita. Valores mais próximos de 1 indicam uma forte relação entre as variáveis, enquanto valores mais próximos de 0 sugerem independência.

2.5 Métricas de avaliação

A avaliação do desempenho dos modelos é fundamental para obter insights sobre sua eficácia nas predições ou classificações. A utilização de estratégias que resumem o desempenho por meio de métricas específicas é crucial nesse processo. A análise dessas métricas proporciona uma compreensão mais aprofundada do modelo, permitindo identificar pontos fortes e áreas de melhoria. Essa avaliação não apenas valida a qualidade das previsões, mas também orienta os próximos passos na pesquisa, direcionando ajustes necessários no modelo ou indicando caminhos para refinamento. Dessa forma, a escolha e interpretação adequadas das métricas são passos essenciais para uma avaliação informada e um progresso significativo na pesquisa.

2.5.1 Matriz de confusão

Uma matriz de confusão é uma tabela usada para avaliar o desempenho de um modelo de classificação. Seu papel é de expor os resultados das predições do modelo quando comparadas com os valores reais.

A matriz de confusão organiza as previsões do modelo em quatro categorias, comumente chamadas de Verdadeiro Positivo (VP), Falso Positivo (FP), Verdadeiro Negativo (VN) e Falso Negativo (FN). Essas categorias são definidas da seguinte maneira:

- Verdadeiro Positivo (VP): Exemplos que foram corretamente classificadas como pertencentes à classe positiva.
- Falso Positivo (FP): Exemplos que foram erroneamente classificadas como pertencentes à classe positiva, quando na verdade pertencem à classe negativa.
- Verdadeiro Negativo (VN): Exemplos que foram corretamente classificadas como pertencentes à classe negativa.
- Falso Negativo (FN): Exemplos que foram erroneamente classificadas como pertencentes à classe negativa, quando na verdade pertencem à classe positiva.

		Previsão	
		Positivo	**Negativo**
Real	**Positivo**	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Negativo	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Tabela 1: Matriz de Confusão

2.5.2 Acurácia

A acurácia é a proporção de predições corretas feitas por um modelo em relação ao número total de predições. A fórmula básica para calcular a acurácia é dada por:

$$\text{Acurácia} = \frac{\text{Número de predições corretas}}{\text{Número total de predições}} = \frac{VP + VN}{VP + VN + FP + FN} \tag{2.5.1}$$

Essa métrica fornece uma visão geral do desempenho do modelo, indicando a porcentagem de instâncias corretamente classificadas. No entanto, a acurácia pode ser enganosa em casos onde as classes não estão balanceadas. Em situações desse tipo, um modelo que prevê sempre a classe majoritária pode ter uma acurácia alta, mesmo que não seja eficaz.

2.5.3 Precisão

A precisão é definida como a proporção de exemplos classificados corretamente como positivos em relação ao total de exemplos classificadas como positivas (verdadeiras positivas mais falsos positivos).

$$\text{Precisão} = \frac{VP}{VP + FP} \tag{2.5.2}$$

A precisão é particularmente útil quando os falsos positivos são mais problemáticos ou custosos em comparação com os falsos negativos. Por exemplo, em um sistema de detecção de spam, classificar erroneamente um e-mail legítimo como spam (falso positivo) pode ser mais prejudicial do que deixar passar um e-mail de spam (falso negativo).

2.5.4 Recall

O recall, também conhecido como sensibilidade, é outra métrica utilizada no contexto de classificação, focada em capturar a proporção de exemplos positivos que foram corretamente identificadas pelo modelo em relação ao total de exemplos positivos existentes.

$$\text{Recall} = \frac{VP}{VP + FN} \tag{2.5.3}$$

O recall é especialmente útil quando os falsos negativos (exemplos positivos não identificadas pelo modelo) são mais críticos ou custosos do que os falsos positivos. Por exemplo, em um sistema de detecção de fraudes, é crucial identificar todas as transações fraudulentas, mesmo que isso signifique aceitar algumas transações normais erroneamente classificadas como fraudulentas.

2.5.5 F1-score

O F1-score é uma métrica de avaliação que combina as métricas de precisão e recall em um único valor.

$$F1 = 2 \cdot \frac{\text{Precisão} \cdot \text{Recall}}{\text{Precisão} + \text{Recall}} \tag{2.5.4}$$

O F1-score é a média harmônica entre a precisão e o recall. A média harmônica é utilizada porque penaliza extremos, sendo particularmente sensível a baixos valores em qualquer uma das métricas.

O F1-score varia de 0 a 1, onde 1 indica o melhor desempenho possível, equilibrando tanto a precisão quanto o recall. Essa métrica é particularmente útil quando há um desequilíbrio significativo entre as classes, pois é menos sensível a grandes quantidades de verdadeiros negativos.

3 Metodologia

A metodologia adotada nesta pesquisa se fundamenta na análise do conjunto de dados *"Loan Data for Dummy"*, visando a compreensão e modelagem de padrões associados a operações de empréstimos. Dois métodos distintos, Regressão Logística e Redes Neurais, serão empregados para investigar as relações existentes nos dados e aprimorar as previsões. A implementação desses modelos será realizada utilizando tanto a linguagem de programação R quanto Python. Além disso, a técnica SHAP (*Shapley Additive Explanations*) será integrada para proporcionar uma interpretação aprofundada do modelo de redes neurais, ampliando a transparência nas decisões preditivas.

3.1 Conjunto de dados

O banco de dados *"Loan Data for Dummy"* é uma base de dados do Kaggle, projetada para simular informações relacionadas a operações de empréstimos. Desenvolvido para fins educacionais e de pesquisa, esse conjunto tem sua origem de um modelo de banco *"peer to peer"* sediado na Irlanda, no qual o banco disponibiliza recursos a potenciais clientes, obtendo lucros com base no risco que assume. Os dados disponíveis no Kaggle representam uma versão fictícia de uma situação real, com a maior parte dos dados manipulados ou criados sinteticamente para preservar as informações dos clientes originais.

A variável que será foco do estudo é a "Condição do empréstimo". Através dessa variável, é possível discernir se um empréstimo foi classificado como "bom"ou "ruim", proporcionando uma avaliação da qualidade e risco associados a cada transação. No contexto deste conjunto de dados, presume-se que a "Condição do empréstimo"seja uma variável binária, onde, por exemplo, "0"poderia indicar um empréstimo em boas condições e "1"indicaria o contrário. A compreensão aprofundada dessa variável é essencial para a construção e interpretação adequada dos modelos subsequentes, como a regressão logística e redes neurais, permitindo uma análise mais precisa e informada do risco associado aos empréstimos.

3.1.1 Variáveis

A base de dados é composta por 30 variáveis, incluindo a variável resposta, e existem 887379 observações. Para se avaliar a variável "Condição do empréstimo"foi utilizada algumas variáveis presentes na base de dados, como:

1. **Tempo de emprego**: Representa o tempo de emprego do solicitante expresso

numericamente. Um valor de 5 indicaria que o indivíduo está empregado há 5 anos.

2. **Tipo de residência**: Indica o status de moradia do solicitante, como proprietário, inquilino ou outra forma de ocupação residencial.

3. **Renda anual**: Reflete a renda anual do solicitante, uma medida crucial para avaliar a capacidade de pagamento do empréstimo. Pode ser expressa numericamente, por exemplo, 50,000.

4. **Valor do empréstimo**: Representa o valor do empréstimo solicitado pelo requerente, geralmente expresso em termos monetários, como 10,000.

5. **Prazo**: Indica o prazo do empréstimo, especificando o período de tempo durante o qual o empréstimo deve ser reembolsado. Pode ser, por exemplo, 36 meses.

6. **Tipo de aplicação**: Refere-se ao tipo de aplicação, indicando se é uma aplicação individual ou conjunta.

7. **Finalidade**: Descreve a finalidade do empréstimo, como consolidação de dívidas, compra de casa, educação, entre outros.

8. **Tipo do juros**: Indica a natureza dos pagamentos de juros, se são fixos ou variáveis.

9. **Taxa de juros**: Representa a taxa de juros associada ao empréstimo, geralmente expressa como uma porcentagem, como 10.

10. **Grau**: Refere-se à classificação de risco do tomador de empréstimo atribuída pela instituição financeira, como A, B, C, etc.

11. **DTI**: Significa "Debt-to-Income"(Dívida-para-Renda) e representa a proporção entre as dívidas mensais e a renda mensal do requerente, proporcionando uma medida da capacidade de pagamento.

12. **Valor bruto pago**: Representa o valor total pago, incluindo o principal e os juros, ao final do empréstimo.

13. **Valor líquido pago**: Indica o total de principal (quantia inicial do empréstimo) recuperado até o momento.

14. **Valor recuperado**: Representa o valor recuperado em caso de inadimplência ou perda.

15. **Parcelas**: Refere-se à parcela mensal que o requerente do empréstimo deve pagar, incluindo tanto o principal quanto os juros.

16. **Região**: Indica a região geográfica associada ao requerente do empréstimo.

3.1.2 Limpeza dos dados

Para diminuir a complexidade da base de dados, as variáveis passaram por 3 critérios de avaliação antes de serem utilizadas nos modelos:

1. Identificação de variáveis que não impactariam o resultado do modelo;
2. Comparação de variáveis que gerem a mesma informação;
3. Extração de variáveis presentes em apenas uma das categorias da variável resposta.

O item 1. destaca-se a variável "ID"como independente da variável resposta, atuando unicamente como identificador do cliente, sem exercer influência no resultado final do modelo.

O item 2. se refere à situação da base de dados em que o autor realizou uma rotulação numérica de variáveis já categorizadas, como por exemplo: "Tipo de juros"e "Tipo de juros Cat", onde na primeira variável tem as opções "Juros simples"e "Juro compostos"e na segunda variável o autor associa os números "1"e "2"respectivamente a essas variáveis.

O item 3. esclarece variáveis que desempenham funções em apenas uma das categorias da variável resposta. Como é o caso da variável "Recuperações totais"onde a mesma é presente apenas no caso do cliente ter sido inadimplente, se relacionando com a categoria "Empréstimo ruim"da variável resposta. Para o caso de "Empréstimo bom"os valores da variável estão zerados.

Com isso a base de dados ficou da seguinte maneira:

Base de dados	**Número de Colunas**
Antes da extração	30
Depois da extração	18

Tabela 2: Número de colunas antes e depois da preparação dos dados

3.2 Métodos

Falar da regressão logística aplicada na base de dados, como será feito a preparação do modelo, normalização etc

Falar do modelo de redes neurais utilizado, arquitetura base

Detalhar o porquê de se interpretar os modelos e as maneiras que é feito isso(testes, coefs, shap)

O intuito é ser a ponte entre referencial teório e resultados

4 Resultados

4.1 Análise descritiva

4.1.1 Condição do empréstimo

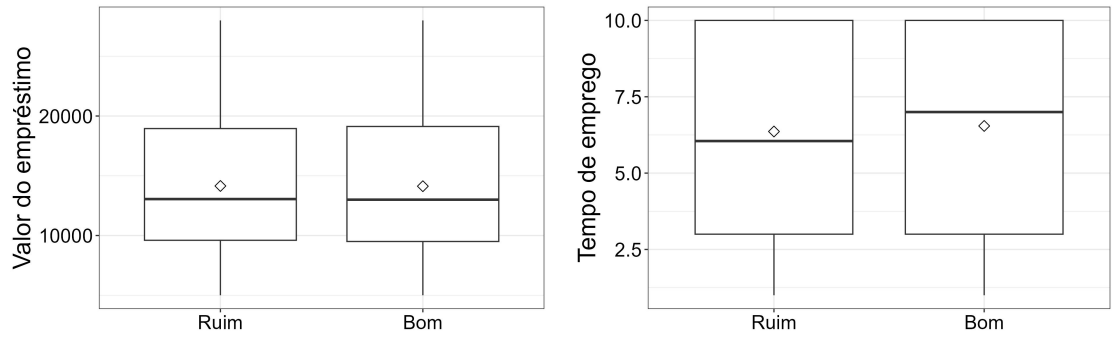
A variável "Condição do empréstimo"é a variável reposta desse estudo, como foi definido anteriormente. Com isso temos o seguinte comportamento dessa variável:

Condição do empréstimo	**Número de observações**	**Frequência relativa**
Empréstimo bom	819950	92,4%
Empréstimo ruim	67429	7,59%

Tabela 3: Número de observação em cada categoria da variável resposta

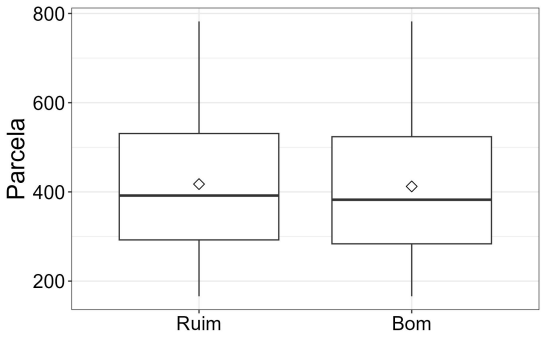
A Tabela 3 mostra a distribuição da variável "Condição do empréstimo". Uma variável composta majoritariamente por observações do tipo "Empréstimo bom", onde a mesma está presente em mais de 90% das observações na base de dados, mostrando que a cada 12 empréstimos rotulados como "bons", existe 1 rotulado como "ruim".

4.1.2 Relação entre as covariáveis e a variável resposta

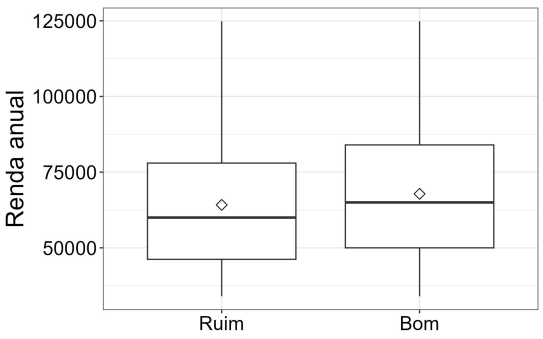


(a) Valor do empréstimo

(b) Tempo de emprego (em anos)



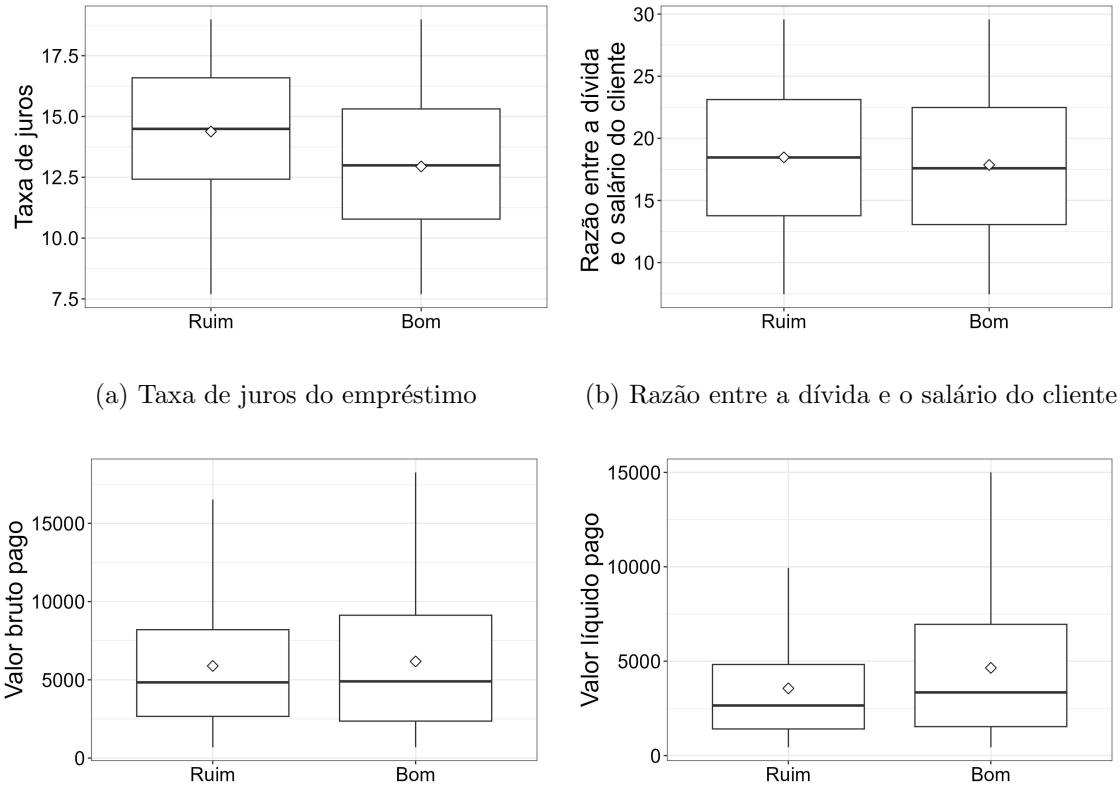
(c) Valor da parcela do emprésitmo



(d) Renda anual (em dólar)

Figura 10: Variáveis explicativas em relação à condição do empréstimo

O comportamento da variável resposta nas Figuras 10a e 10c demonstrou semelhanças, onde, em ambos os casos, não foi evidenciada uma clara diferença entre o valor do empréstimo e o valor da parcela em relação às categorias da variável resposta. A Figura 10b também apresenta um comportamento semelhante entre as classes "Empréstimo ruim"e "Empréstimo bom", mas com um detalhe: a mediana do tempo de trabalho dos clientes rotulados como "Empréstimo ruim"foi inferior em comparação ao outro caso. Por fim, a Figura 10d indica que clientes com uma renda anual elevada tendem a ser categorizados como "Empréstimo bom".



(a) Taxa de juros do empréstimo

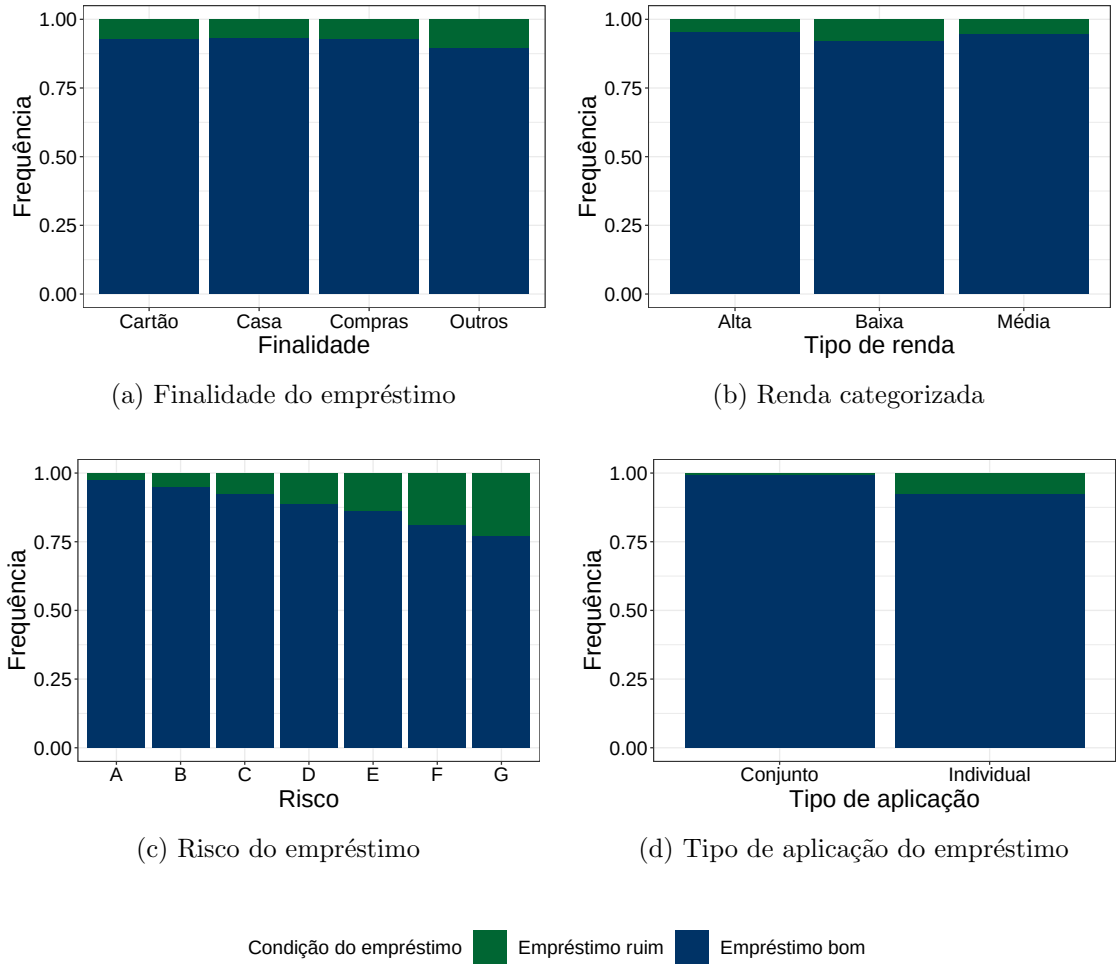
(b) Razão entre a dívida e o salário do cliente

(c) Valor bruto do emprétimo pago

(d) Valor líquido do emprétimo pago

Figura 11: Variáveis explicativas em relação à condição do empréstimo

A Figura 11a evidencia uma relação significativa entre taxas de juros elevadas e empréstimos considerados ruins. A Figura 11b complementa a informação fornecida pela Figura 10d, indicando que clientes com renda mais elevada tendem a cumprir adequadamente com seus pagamentos. As Figuras 11c e 11d seguem padrões semelhantes, sugerindo que clientes que quitaram a maior parte do empréstimo são frequentemente rotulados como bons pagadores.



(a) Finalidade do empréstimo

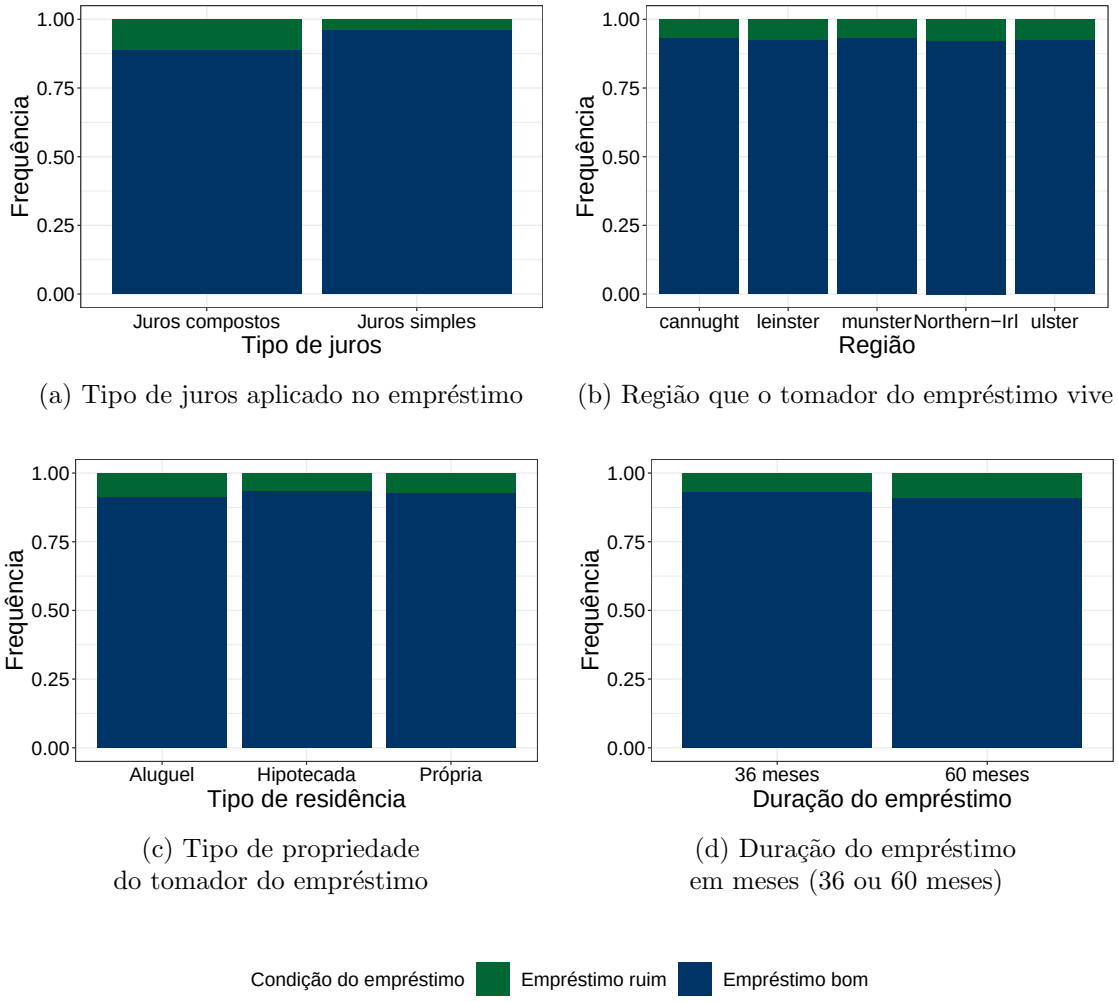
(b) Renda categorizada

(c) Risco do empréstimo

(d) Tipo de aplicação do empréstimo

Figura 12: Variáveis explicativas em relação à condição do empréstimo

A Figura 12a ilustra que as categorias da variável "Finalidade"seguem a proporção natural da condição do empréstimo, conforme indicado na Tabela de Condição do Empréstimo. Na Figura 12b, as categorias "Alta"e "Média"exibem proporções menores de empréstimos ruins em comparação com a categoria "Baixa", que apresenta uma proporção de quase 10% de empréstimos ruins. A Figura 12c revela um padrão de "cascata", indicando que à medida que o risco do empréstimo aumenta, a proporção de empréstimos ruins nas últimas categorias também aumenta, sendo a categoria G a mais afetada, com quase 25% de empréstimos classificados como ruins. Na Figura 12d, a categoria "Empréstimo conjunto"não registrou observações de empréstimos ruins, concentrando a maioria desses empréstimos na categoria "Empréstimo individual".



(a) Tipo de juros aplicado no empréstimo

(b) Região que o tomador do empréstimo vive

(c) Tipo de propriedade do tomador do empréstimo

(d) Duração do empréstimo em meses (36 ou 60 meses)

Figura 13: Variáveis explicativas em relação à condição do empréstimo

Na Figura 13, o gráfico 13a evidencia que empréstimos obtidos sob juros compostos possuem uma proporção mais elevada de rotulações ruins em comparação com empréstimos sob juros simples. As Figuras 13b e 13c destacam uma proporção natural refletida pela distribuição das categorias da variável resposta, conforme apresentado na Tabela 3. Já a Figura 13d revela uma proporção mais significativa de empréstimos ruins quando estes tendem a demorar mais para serem pagos.

Covariáveis	**Coeficiente de correlação**
Tempo de trabalho	-0.02
Renda anual	-0.03
Valor do empréstimo	0.00
Taxa de juros	0.18
DTI	0.01
Valor bruto pago	-0.04
Valor liquido pago	-0.10
Parcela	-0.01
Duração do empréstimo	0.01

Tabela 4: Valores do coeficiente de Pearson entre as covariáveis e a variável resposta

A partir da análise da Tabela 4, nota-se que as correlações entre as variáveis explicativas e a variável resposta são de baixa magnitude. Os coeficientes calculados indicam uma relação linear fraca ou inexistente entre essas variáveis. Esses resultados sugerem que outros fatores ou relações não lineares podem estar desempenhando um papel mais significativo na explicação da variabilidade na variável resposta.

Covariáveis	**Coeficiente de contingência**
Tipo de residência	0.04
Tipo de aplicação	0.01
Finalidade	0.03
Tipo de juros	0.14
Risco	0.15
Região	0.01
Prazo	0.04
Renda	0.04

Tabela 5: Valores do coeficiente de contingência entre as covariáveis e a variável resposta

Ao analisar a Tabela 5, nota-se que a maioria dos coeficientes de contingência entre as covariáveis e a variável resposta são próximos de zero. Destaca-se que a variável "Risco"exibe o maior valor de associação, atingindo 0.15. Entretanto, é importante ressaltar que esse valor ainda é relativamente baixo. Os coeficientes sugerem, em geral, uma falta de associação significativa entre as covariáveis mencionadas e a variável resposta.

4.2 Regressão logística

Falar do modelo utilizado, a normalização dos dados, os resultados métricas de avaliação e interpretação dos coeficientes

Covariáveis	**Coeficientes**	**Erro padrão**
Valor líquido pago	-4.733	0.034
Valor bruto pago	3.321	0.028
Tipo de aplicação	-1.848	0.106
Taxa de juros	1.412	1.412
Valor do empréstimo	-1.406	0.039
Risco	-1.049	0.014
Tipo de juros	-0.462	-0.462
Prazo	-0.203	0.028
Renda anual	-0.195	0.010
DTI	-0.151	0.011
Renda categorizada	-0.111	0.015
Tempo de trabalho	-0.061	0.009
Região	0.038	0.003
Duração do empréstimo	0.033	0.009
Tipo de residência	0.019	0.003
Finalidade	0.019	0.002
Parcela	0.005	0.000

Tabela 6: Estimativa dos coeficientes do modelo logístico e o erro padrão associado

Ao analisar os resultados apresentados na Tabela 6, fica evidente que as variáveis "Valor líquido pago"e "Valor bruto pago"exercem uma influência significativa no valor final de $P(Y = 1)$. Essas duas variáveis estão diretamente associadas à quantia do empréstimo que o cliente já quitou, indicando sua relevância na predição do resultado. Ao calcular a Razão de chances dessas duas variáveis, temos que:

- "Valor líquido pago": apresenta um RC de 0.008, o que sugere que, mantendo todas as outras variáveis constantes, a chance de o empréstimo ser classificado como bom é 125 vezes maior do que ser classificado como ruim.
- "Valor bruto pago": exibe um RC de 27.68, indicando que, ao manter todas as outras variáveis constantes, a chance de o empréstimo ser classificado como ruim é 27 vezes maior do que ser classificado como bom.

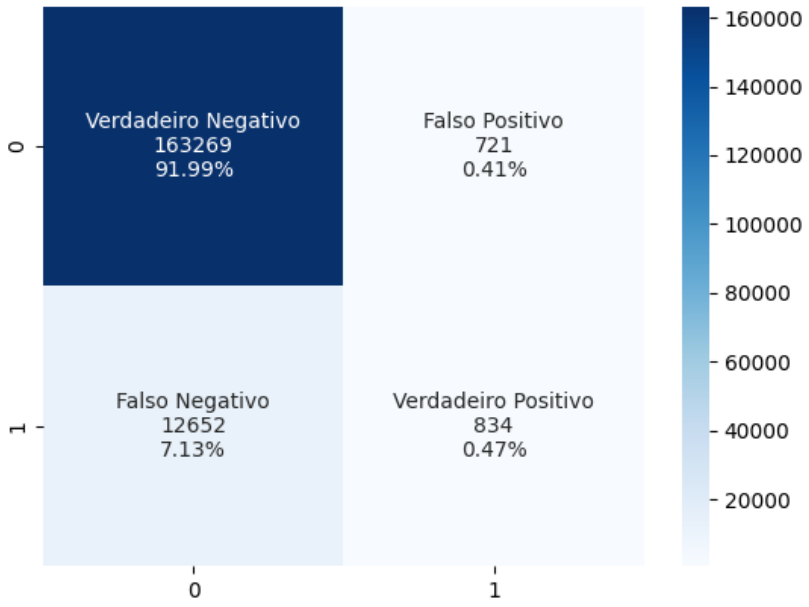


Figura 14: Matrix de confusão do modelo logístico

Visualizando os dados da Figura 14 é possível analisar os resultados do modelo logístico no conjunto de teste. O conjunto de teste apresenta uma distribuição da variável resposta de com mais de 92% dos casos como um empréstimo bom, e o restante como o empréstimo ruim.

	Precisão	**Recall**	**F1-Score**	**Tamanho da amostra**
0	0.928	0.995	0.961	163990
1	0.536	0.062	0.111	13486
Média macro	0.732	0.528	0.535	177476
Média ponderada	0.898	0.925	0.896	177476
Acurácia				0.924649

Tabela 7: Report do modelo logístico

Com base nos dados apresentados na Tabela 7 e na Figura 14, observamos que o modelo exibe uma acurácia elevada. Ele é capaz de fazer previsões precisas na maioria dos casos, alcançando uma taxa de 92,46% de classificações corretas no conjunto de teste. No entanto, é crucial destacar que essa elevada acurácia é influenciada pela proporção significativa de casos onde o empréstimo é rotulado como "bom", presente em mais de 92% dos dados de teste. Como resultado, o modelo tende a classificar uma parte considerável dos dados como "0", refletindo a influência dessa distribuição desigual na estimação dos parâmetros do modelo logístico.

Ao examinarmos a precisão do modelo, observamos uma taxa de acerto de 73% nas previsões em comparação com as rótulos reais do conjunto de teste. É importante ressaltar a notável precisão na categoria "Empréstimo bom", atingindo quase 92%. No entanto, vale destacar que esse valor elevado está correlacionado ao desequilíbrio nos dados, onde a classe "Empréstimo bom"é predominante.

Ao avaliar o recall do modelo logístico, observamos, em média, valores mais baixos em comparação com a precisão. O recall médio é de 52,8%, indicando que, ao analisar as porcentagens das rótulos reais, o modelo conseguiu acertar um pouco mais da metade delas. Esse desempenho é atribuído ao alto número de falsos negativos no modelo, visto que, ao considerar o total de "Empréstimos ruins"(13.486), o modelo acertou apenas 834 desses casos.

O F1-score acaba refletindo a real situação do modelo, pois ele balanceia os bons resultados apresentados pela precisão com os resultados ruins do recall. O F1-score médio apresentado foi de 52,57%.

A avaliação global do modelo logístico revela um viés significativo, amplificado pelo desequilíbrio nos dados. Embora o modelo tenha alcançado uma taxa geral de acerto de 92%, sua incapacidade de distinguir adequadamente entre "Empréstimos bons"e "Empréstimos ruins"é evidente. Este desempenho inferior sugere limitações na capacidade do modelo de generalizar e discriminar efetivamente entre as categorias, indicando a necessidade de refinamentos ou considerações adicionais para melhorar sua robustez.

4.3 Modelagem da rede neural

	Precisão	**Recall**	**F1-Score**	**Tamanho da amostra**
0	0.954	0.904	0.929	163990
1	0.291	0.478	0.362	13486
Média macro	0.622	0.691	0.645	177476
Média ponderada	0.904	0.872	0.886	177476
Acurácia				0.872

Tabela 8: Report da rede neural

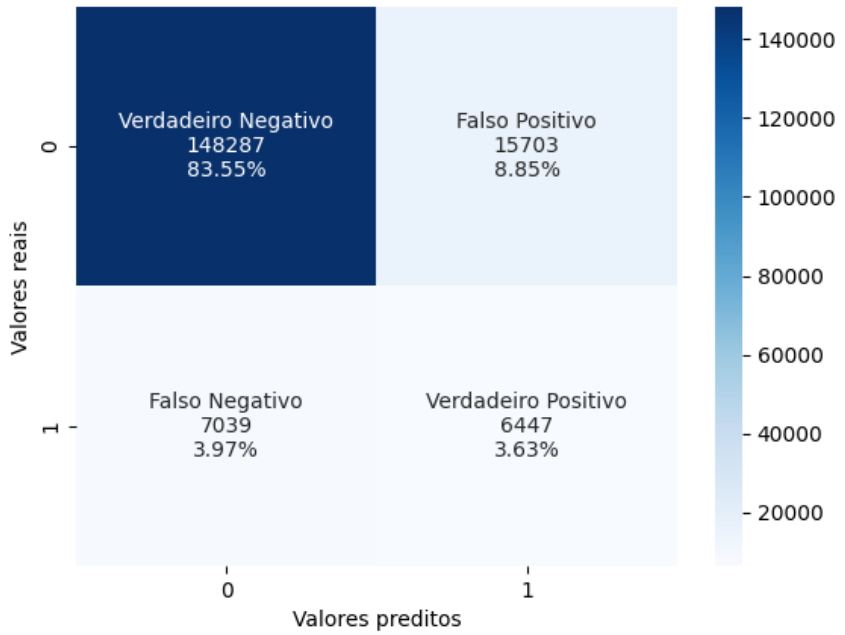


Figura 15: Matrix de confusão da rede neural

4.4 Interpretação da rede neural

- mostrar gráfico da média dos shap vs regressao logistica

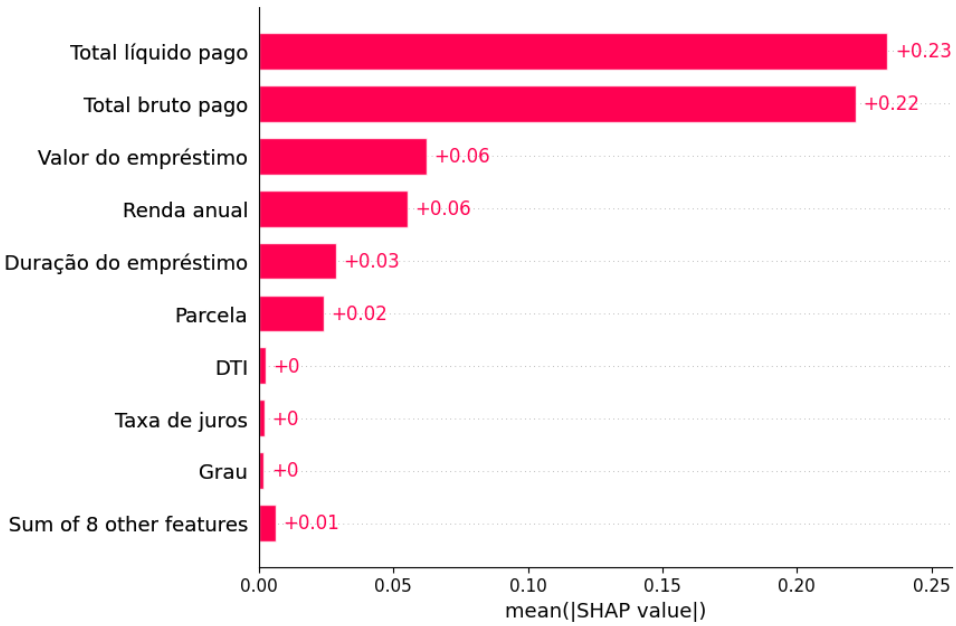
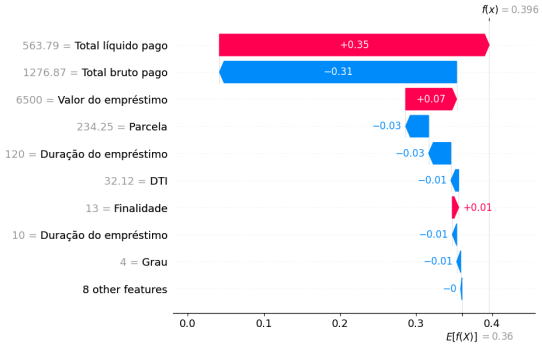


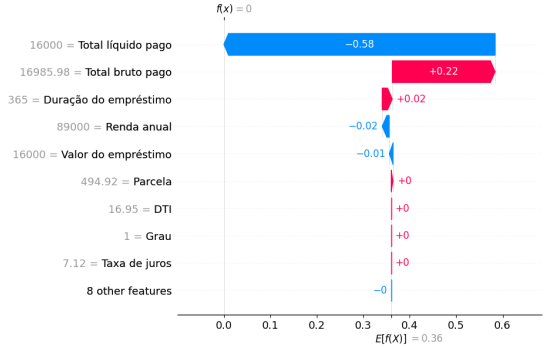
Figura 16: Média absoluta dos valores de shap

- mostrar 2 gráficos de shap especificos de 2 observações(pra mau pagador e pra

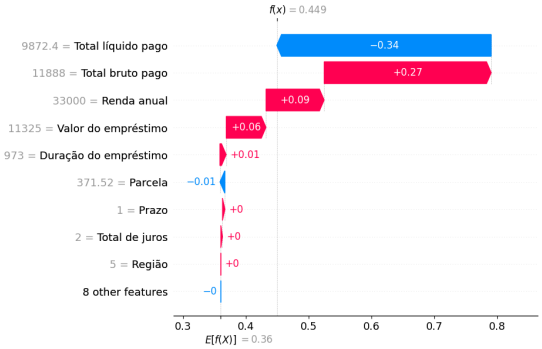
bom pagador)



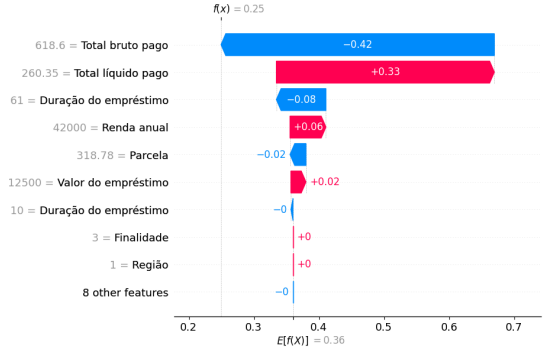
(a) Valor e classificação do emprestimo



(b) WSIR residuals.



(c) WSIR predictions in the covariate space.



(d) WSIR residuals

Figura 17: aloalo

- mostrar o gráfico com todos as amostras de shap(shap.plots.beeswarm)

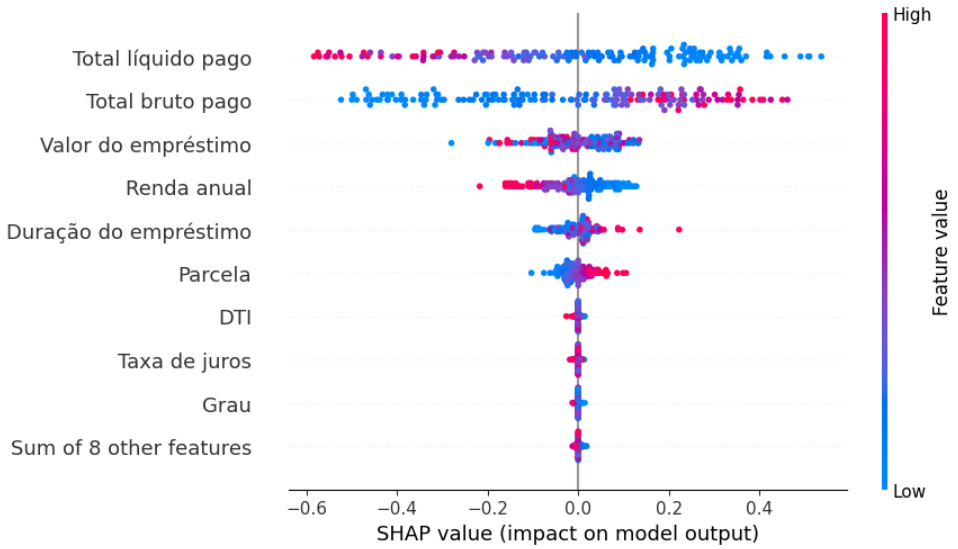


Figura 18: Valores de shap para as 80 observações utilizadas

- mostrar o gráfico de força pra apenas uma observaçã

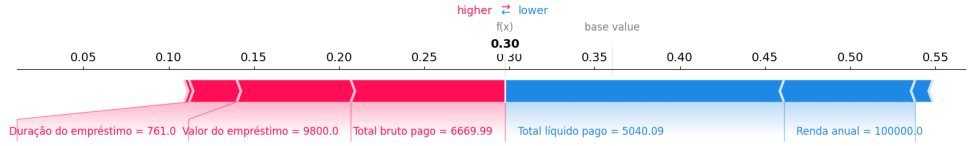


Figura 19: Gráfico de força em uma observação

- mostrar o gráfico de força para todas as observações

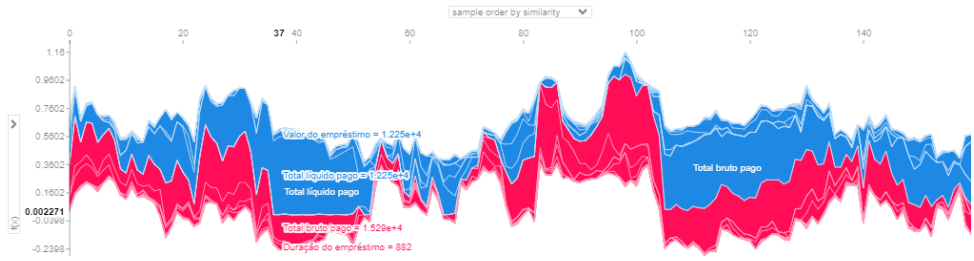


Figura 20: Gráfico de força para múltiplas observações

4.5 Benchmark entre regressão logística e redes neurais

4.5.1 Complexidade da arquitetura

Modelo	Número de parâmetros
Regressão Logística	18
Rede Neural	151233

Tabela 9: Número de parâmetros nos modelos

4.5.2 Resultado dos modelos

Métricas	Regressão logística	rede neural
Falsos Positivos	721	15703
Falsos Negativos	12652	7039

Tabela 10: Comparação dos resultados de Falsos Positivos e Falsos Negativos

Métricas	Regressão logística	Rede neural
Precisão (Classe 0)	0.928081	0.954
Precisão (Classe 1)	0.536334	0.291
Recall (Classe 0)	0.995603	0.904
Recall (Classe 1)	0.0618419	0.478
F1-Score (Classe 0)	0.960657	0.929
F1-Score (Classe 1)	0.110897	0.362
Acurácia	0.924649	0.872

Tabela 11: Comparação dos resultados da Regressão logística e Rede neural

4.5.3 Tempo de execução

Estatísticas	Regressão logística	Rede neural
Mínimo	0.0	52.067995
Quartil 25	0.0	53.327155
Média	0.3335619	59.446688
Mediana	0.0	56.855202
Quartil 75	0.88143349	66.278052
Máximo	1.50370598	92.03124
Variância	0.28385463	59.008917
Desvio padrão	0.5327801	7.681726

Tabela 12: Tempo de predição (em ms) de cada modelo, em uma amostra com 50 observações

5 Conclusão

Referências

BISHOP, C. M. *Pattern Recognition and Machine Learning*. [S.l.]: Springer, 2006.

IZBICKI, R.; SANTOS, T. M. dos. *Aprendizado de máquina: uma abordagem estatística*. [S.l.: s.n.], 2020. ISBN 978-65-00-02410-4.

JAMES, G. et al. *An Introduction to Statistical Learning with Applications in R*. New York: Springer, 2013.

SHAPLEY, L. S. A value for n-person games. *Contributions to the Theory of Games*, v. 2, p. 307–317, 1953.

WERBOS, P. J. *Beyond regression: New tools for prediction and analysis in the behavioral sciences*. Tese (Doutorado) — Harvard University, 1974.

6 Anexo