Universidade de Brasília
Departamento de Estatística

Interpretação de redes neurais utilizando a técnica SHAP

Davi Guerra Alves

Brasília
2023

Davi Guerra Alves

Interpretação de redes neurais utilizando a técnica SHAP

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
2023

Dedicatória

Dedico esse trabalho primeiramente à minha família. Aos meus pais e minha irmã, que com certeza, sem a luta a diária que vivenciamos, hoje não teria chegado aonde cheguei. Aos meus professores do ensino médio, que sem os conselhos dos mesmos e orientação da existência de um curso chamado "Estatística", hoje não teria tomado uma das melhores decisões da minha vida.

Aos amigos que fiz durante o curso. Aos que me acompanharam do início ao fim de toda a graduação, em especial a Laurinha, o Marcelo, o Lucas, o Hermes e o Kevyn e também aqueles que fui encontrando pelo caminho, como a Jéssica, o João e o Gabriel (aqui representado todos da Tukey). Obrigado por todos os momentos vividos, nervosismos pré provas, alegrias de finais de semestre. Vocês foram responsáveis por enviesar a Estatística que eu amo tanto.

E por fim, aos professores que passaram pelo meu caminho. Professor Guilherme, presente do começo ao fim da minha estadia no curso, com toda certeza um dos grandes responsáveis pelo caminho que trilho hoje no curso. Professora Thais, que aceitou a missão de ser a orientadora do autor desse projeto, seus conselhos e orientações me tornaram um estatístico melhor. E a todos os outros professores no qual fui feliz no curso, Leandro, Ana Maria, Antônio, Juliana. Espero ser 10% dos que vocês foram pra mim nesse curso.

Agradecimentos

O presente trabalho foi financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico(CNPq), baseado no Programa de Pesquisa em Tecnologia da Informação - CGCEX, com bolsa na Iniciação Tecnológica em TICS - ITC-A.

Resumo

O objetivo central deste trabalho é buscar métodos para interpretar modelos de redes neurais. Concomitantemente, realizou-se uma comparação dos resultados desse modelo com um modelo estatístico convencional, a regressão logística. Foram utilizados dados relacionados a empréstimos, obtidos da plataforma *Kaggle*. A variável de estudo analisada foi a "Condição do empréstimo", que classifica os empréstimos dos clientes como "Bom"ou "Ruim".

A técnica adotada para interpretar os modelos de redes neurais foi o SHAP(*SHapley Additive exPlanations*). Os resultados comparativos entre os modelos revelam que o modelo de redes neurais produziu resultados melhores que o modelo logístico.

Palavras-chaves: SHAP, redes neurais, regressão logística, interpretação, empréstimos.

Lista de Tabelas

Lista de Figuras

Sumário

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. O propósito desses modelos é executar tarefas complexas, tais como reconhecimento de padrões, identificação de imagens, processamento de linguagem natural, entre outras. 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 (LUNDBERG; LEE, 2017). 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 (SHAPLEY, 1953), que atribui uma contribuição de importância para cada recurso de entrada na saída da rede neural.

Portanto, esse trabalho tem como objetivo principal 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.

Uma alternativa aos modelos de rede neurais é escolher modelos tradicionais da estatística, como a regressão logística, que já possui um conjunto de ferramentas interpretativas consolidadas. Dessa forma, o estudo também se concentra em uma análise comparativa entre o modelo de redes neurais e o modelo logístico. Essa abordagem visa avaliar tanto os resultados quanto as interpretações geradas por ambos os modelos, proporcionando uma compreensão mais profunda de como essas abordagens se comportam em um cenário comum. Essa comparação não apenas lança luz sobre as diferenças de desempenho, mas também destaca as diferenças interpretativas distintas entre os dois modelos, enriquecendo assim a compreensão sobre a escolha adequada de modelos em diferentes contextos.

As seções subsequentes abordarão os métodos empregados no projeto, detalhando a aplicação e os resultados obtidos. A Seção 2 introduzirá a metodologia utilizada nos modelos de Regressão Logística e Redes Neurais, além de explicar as métricas de avaliação

e o funcionamento da técnica SHAP. Já na Seção 3, o foco será o processo de modelagem de dados específico para os modelos de Regressão Logística e Redes Neurais, incluindo a seleção do modelo mais eficaz e a interpretação dos resultados alcançados. Por fim, na Seção 4, serão apresentados os resultados obtidos com os modelos, com ênfase no uso do método SHAP para a interpretação do modelo de Redes Neurais.

2 Referencial teórico

2.1 Regressão logística

A regressão logística é um método estatístico utilizado para modelar a probabilidade condicional de uma variável, dado um conjunto de covariáveis. É comumente utilizada para problemas de classificação binária, onde a variável dependente possui apenas duas categorias, como sim/não, positivo/negativo, 0/1 (HOSMER, 2013).

O modelo de regressão logística é construído considerando a probabilidade da variável aleatória Y ser igual a 1, onde Y é uma variável aleatória com distribuição Bernoulli, com probabilidade 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 parâmetros $(\beta_0, \beta_1, \beta_2, ..., \beta_k)$ 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 $L(\beta)$ 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 $l(\beta)$.

A log-verossimilhança para a regressão logística é:

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

onde,

- Né o número total de observações;
- y_i é a variável dependente binária da i-ésima observação (0 ou 1);
- x_i é o vetor de covariáveis da i-ésima observação;
- β é o vetor de parâmetros do modelo modelo logístico.

A ideia é encontrar os valores de $(\beta_0, \beta_1, \beta_2, ..., \beta_k)$ que maximizam essa função. Isso geralmente é feito usando métodos computacionais, como o algoritmo de otimização Newton-Raphson ou o Gradiente Descendente (CURRY, 1944).

2.1.1 Interpretação

Para a interpretação do modelo logístico é utilizada a Razão de Chances, que calcula a razão entre duas chances, sendo a chance (ou, do inglês, *odds*) de um evento definida como a probabilidade do sucesso do evento sobre a probabilidade do fracasso (SZUMILAS, 2010),

$$\text{Chance} = \frac{P(Y = 1|X = x)}{P(Y = 0|X = x)}. \tag{2.1.3}$$

Para calcular a Razão de Chances, inicialmente, é determinada a chance de ocorrência de um Y dado um conjunto de covariáveis x. Em seguida, é calculada a chance desse mesmo evento ocorrer quando, por exemplo, a k-ésima covariável correspondente é incrementada em uma unidade. Dessa forma, a fórmula para a Razão de Chances (RC) é expressa por:

$$\text{Razão de Chances} = \frac{\frac{P(Y=1|X\text{`` e } X_j=x_j+1)}{P(Y=0|X\text{`` e } X_j=x_j+1)}}{\frac{P(Y=1|X\text{`` e } X_j=x_j)}{P(Y=0|X\text{`` e } X_j=x_j)}}, \tag{2.1.4}$$

onde, X`` é um vetor com todas as covariáveis exceto a covariável X_j.

Considerando a Equação 2.1.1, temos que a Razão de Chances pode ser calculada como:

$$RC = \exp(\beta_j). \tag{2.1.5}$$

Logo, para o crescimento de 1 unidade em X_k, variável associada ao β_j, a chance do evento ocorrer é multiplicada por $\exp(\beta_j)$ unidades, considerando as demais variáveis constantes. Ou seja, uma RC igual a 1 indica que a variável independente não tem efeito no resultado de Y (nenhuma associação). Uma RC maior que 1 sugere uma associação positiva, enquanto uma 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 chance, ou seja, o log da razão das probabilidades do evento acontecer e dele não acontecer, cuja formulação é dada por:

$$\text{logit}(P(Y=1)) = \log\left(\frac{P(Y=1)}{1-P(Y=1)}\right), \tag{2.1.6}$$

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

Portanto, um coeficiente β_j positivo indica que o aumento na respectiva covariável está associado a um aumento na log-odds (e, portanto, na probabilidade de sucesso), enquanto um coeficiente negativo está associado a uma diminuição nas log-odds (e na probabilidade de sucesso).

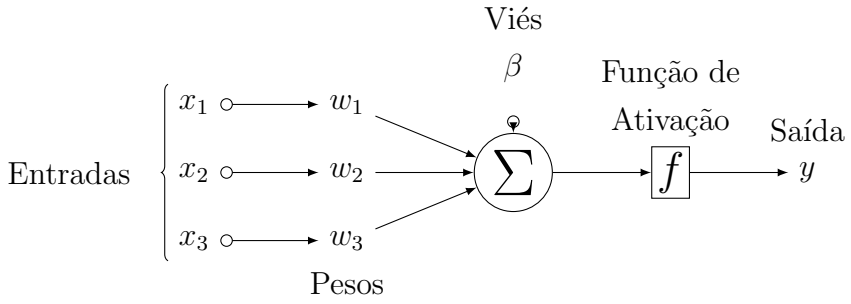
2.2 Redes neurais artificiais

Redes Neurais Artificiais (ou *Deep Learning*) é uma técnica de modelagem de dados 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 (HARDESTY, 2017).

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 para frente. A Figura 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, pelos chamados **pesos**. Logo em seguida ocorre a soma dessa combinação linear. 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

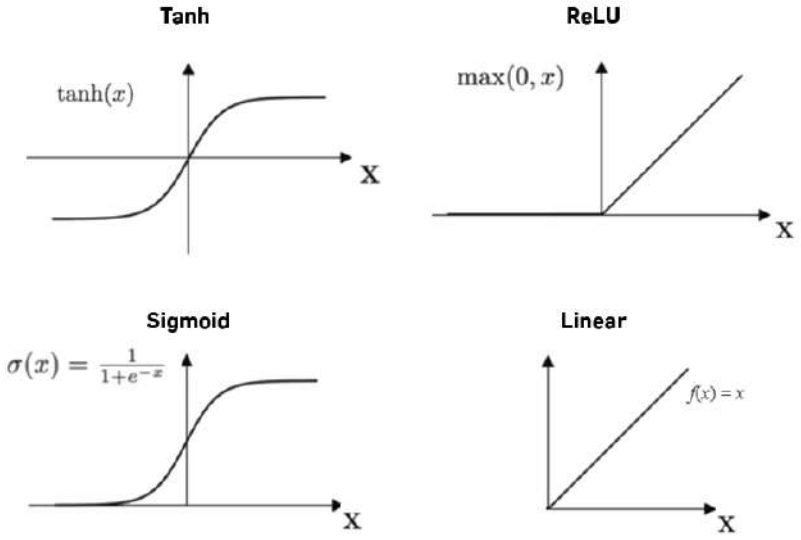


A Figura 1 pode ser representada matematicamente da seguinte maneira:

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

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 utilizado. Note como a maioria dessas funções restringe o valor de sua entrada. No caso da função Sigmoide e da Tangente hiperbólica (tanh), o valor da saída é limitado em um intervalo. Já a ReLU, desconsidera os valores negativos, e por fim existe a função Linear que apenas repassa a mesma informação 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 neurônios 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, contendo o(s) valor(es) predito(s) pela rede (ZELL, 1994).

Figura 3: Rede Neural com uma camada oculta

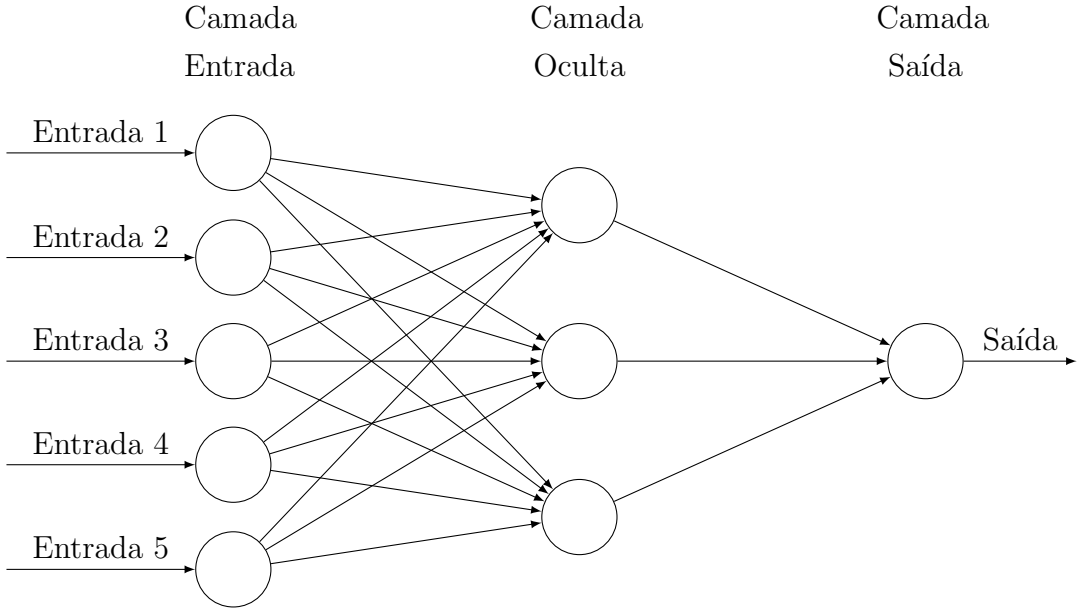
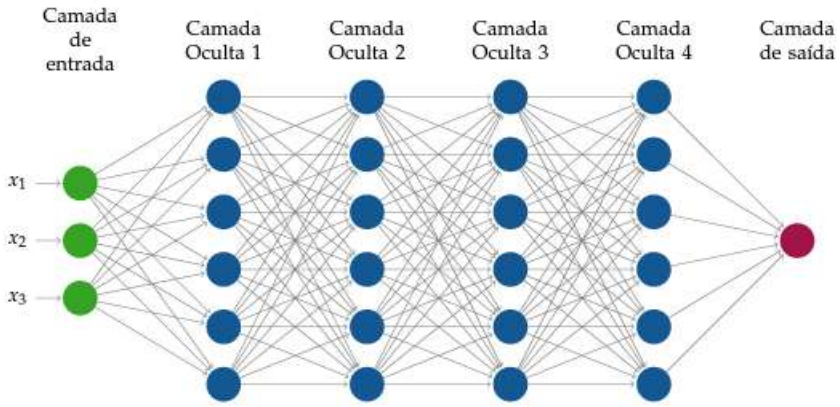


Figura 4: Arquitetura padrão de um rede neural *feedforward*



Fonte: (IZBICKI; SANTOS, 2020)

Note que as Figuras 3 e 4 evidenciam um potencial muito grande de crescimento da rede e, naturalmente, esse aumento pode gerar 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* aplica a Equação 2.2.1 para cada neurônio presente nas camadas internas da rede neural. Por isso temos, para cada j-ésimo neurônio, da camada l, a seguinte combinação linear:

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

onde:

- w_{ij} é o peso associado à conexão entre o neurônio i na camada $l-1$ e o neurônio j 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 $z_j^{(l)}$, sendo esta responsável por gerar o resultado final $a_j^{(l)}$, do j-ésimo neurônio na l-ésima camada.

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

Esse processo vai ser realizado camada a camada, sequencialmente. Logo, considerando uma rede com H camadas ocultas, o resultado da camada de saída $H + 1$ é dado por:

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

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

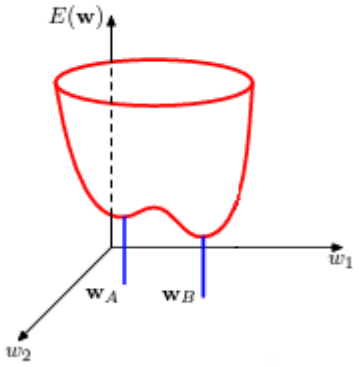
2.2.4 Função de perda

Para otimizar o desempenho do modelo, é escolhida uma função de perda R. Uma função bastante utilizada para estimação de médias 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, os valores preditos $f(x_k)$ estão distantes dos valores reais y_k. Note que o resultado da função f depende exclusivamente dos parâmetros da rede (viés e pesos), por isso, a derivada da função de perda, em relação aos parâmetros, sendo igual a 0, significa que os parâmetros dessa rede minimizaram a perda. Entretanto, devido à complexidade desse modelo, é possível parar em pontos locais mínimos, que, dependendo do contexto, são suficiente ótimos. 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.



Fonte: (BISHOP, 2006)

2.2.5 *Backpropagation*

Como a função de perda $R(\theta)$ depende dos parâmetros (θ) da rede, para se minimizar a função de perda, 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, que provoca alterações no 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:

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

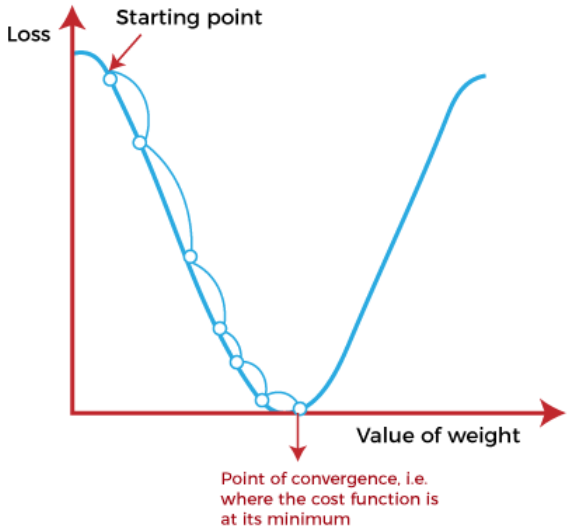
onde $\theta = \theta^m$ significa que o cálculo do gradiente está sendo realizado na iteração m.

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, iniciando na camada de entrada, passando pelas camadas ocultas e chegando na camada de saída (*Forward propagation*), a informação do resultado da rede na função de perda é passada de forma contrária. Ou seja, 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 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 interpretar as predições feitas. Para isso, existem técnicas que visam a interpretação de modelos de redes neurais, como por exemplo a técnica SHAP. Através dela é possível entender como as variáveis de entrada influenciam as previsões do modelo, fornecendo explicações sobre sua lógica. 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 propostos 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, havia 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 (HART, 1989).

Podemos definir $\mathbf{F}$ como o conjunto dos jogadores disponíveis para a realização da tarefa, logo $\mathbf{F} = \{1, 2, ..., \mathbf{M}\}$, onde $\mathbf{M}$ é o número total de jogadores. 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:

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

Existe também uma função que vai mapear um conjunto de valores e retornar um número real, chamada de ν. Com isso, o retorno de $\nu(\mathbf{S})$ é um número real que pode ser definido como o "trabalho da coligação $\mathbf{S}$"ou o "trabalho dos jogadores presentes no conjunto $\mathbf{S}$". 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 i-ésimo jogador em uma tarefa, pode-se calcular o ganho quando é adicionado aquelo jogador na coligação menos o ganho da coligação sem a adição daquele jogador, ficando da seguinte maneira:

$$\nu(\mathbf{S} \cup \{i\}) - \nu(\mathbf{S}) \tag{2.3.1}$$

No exemplo acima, para calcular o efeito da coligação $\{3\}$, pode-se realizar o seguinte processo:

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

Entretanto, considere que as variáveis (ou jogadores) $\{2\}$ e $\{3\}$ apresentam uma semelhança extraordinária. Ao calcular o ganho ao inserir $\{2\}$ na coligação $\{1,2\}$, observamos um aumento substancial. No entanto, ao adicionar $\{3\}$ à coligação $\{1,2,3\}$, o

ganho é significativamente menor. Devido à semelhança no papel desempenhado por essas variáveis, o maior ganho está vinculado à variável adicionada primeiro à coligação, não necessariamente indicando que uma seja 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, considerando 4 jogadores, **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:

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

É importante ressaltar que a função ν aceita a coligação como argumento, não a permutação. Uma coligação é um conjunto, onde a ordem dos elementos não tem importância, ao passo que uma permutação é uma coleção ordenada de elementos. Na permutação [3,1,2,4], por exemplo, 3 é a primeira variável adicionada, enquanto 4 é a última. Portanto, para cada permutação, a ordem dos elementos pode influenciar na contribuição total, mas o ganho total da permutação depende apenas dos elementos, não da ordem. Assim:

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

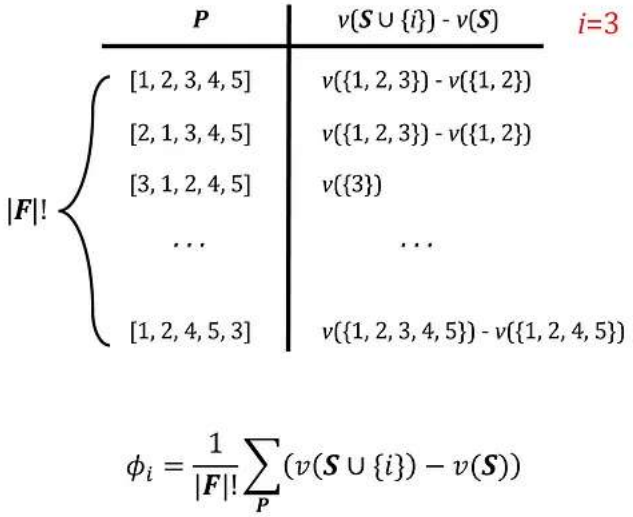
Sendo assim, para cada permutação **P**, é preciso primeiro calcular o ganho da coligação das variáveis que foram adicionadas antes de $\{i\}$, e esse conjunto pode ser chamado de coligação **S**. Feito isso, agora é preciso calcular o ganho das coligações que são formadas ao adicionar $\{i\}$ em **S**,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

Portanto, para cada permutação **P**, é necessário inicialmente calcular o ganho da coligação das variáveis que foram adicionadas antes de i, e esse conjunto pode ser referido como coligação **S**. Em seguida, é necessário calcular o ganho das coligações formadas ao adicionar i a **S**, e podemos representar isso como $\mathbf{S} \cup i$. Dessa forma, a contribuição da variável i, denotada por ϕ_i, é:

$$\phi_i = \frac{1}{|\mathbf{F}|!} \sum_{\mathbf{P}} [\nu(\mathbf{S} \cup \{i\}) - \nu(\mathbf{S})]. \tag{2.3.2}$$

O número total de permutações de **F** é $|\mathbf{F}|!$, sendo $|\mathbf{F}|$ a cardinalidade do conjunto 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 a todas as permutações de jogadores.



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

É evidente que algumas permutações apresentam a mesma contribuição, contanto que suas coligações $\mathbf{S} \cup i$ e **S** sejam idênticas. Para otimizar o cálculo da contribuição de cada permutação, é possível identificar quantas vezes a geração de permutações resultará em uma contribuição 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 $\mathbf{S}$ 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: $\mathbf{F}$ - $(\mathbf{S} \cup \{i\} + 1)$ que basicamente é o que resta das variáveis para completar o conjunto $\mathbf{F}$.

A Figura 8 ilustra o procedimento ao escolher o jogador $i = 3$. Na linha das coligações, são especificadas as possíveis coligações de $\mathbf{S}$, que consistem nas permutações dos jogadores 1 e 2, e a coligação $\{i\}$ cujo único elemento na coluna i, onde o único elemento será o próprio i. Em seguida, são listadas as coligações dos jogadores restantes $\{4, 5\}$. Na linha das permutações, são apresentadas todas as permutações possíveis para $\mathbf{S}$, $\{i\}$ e $\mathbf{F} - \mathbf{S} - \{i\}$. A última linha representa o tamanho do conjunto formado pela permutação/coligação descrita anteriormente.

Figura 8: Relação entre permutações e coalisões.

Coalitions	***S***	+	$\{i\}$	+	***F*-*S*-$\{i\}$**	=	***F***
	$\{1, 2\}$	+	$\{3\}$	+	$\{4, 5\}$	=	$\{1, 2, 3, 4, 5\}$
Permutations	$[1, 2]$ $[2, 1]$	+	$[3]$	+	$[4, 5]$ $[5, 4]$	= =	$[1, 2, 3, 4, 5]$ $[1, 2, 3, 5, 4]$
					$[4, 5]$ $[5, 4]$	= =	$[2, 1, 3, 4, 5]$ $[2, 1, 3, 5, 4]$
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 2.3.2 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 o paralelo do valor de Shapley para o SHAP (*Shapley Additive Explanations*), temos que os jogadores são as covariáveis do modelo e a função característica ν é equivalente à função $f(x)$ responsável por fazer as predições. Assim, valores de SHAP

são calculados para cada observação. Com isso, a fórmula do valor de SHAP, para cada a i-ésima covariável de determinada observação x é dada por:

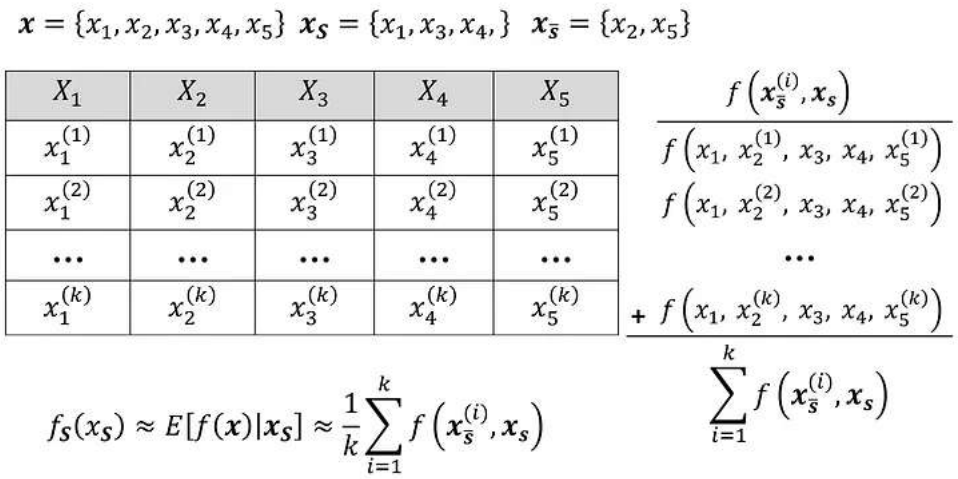
$$\phi_i(f, \mathbf{x}) = \sum_{\mathbf{S} \subseteq \mathbf{F} - \{i\}} \frac{|\mathbf{S}|!(|\mathbf{F}| - |\mathbf{S}| - 1)!}{|\mathbf{F}|!} [f_{\mathbf{S} \cup \{i\}}(\mathbf{x}_{\mathbf{S} \cup \{i\}}) - f_{\mathbf{S}}(\mathbf{x}_{\mathbf{S}})] \tag{2.3.3}$$

Perceba que $f_{\mathbf{S}}(\mathbf{x}_S)$ representa o resultado do modelo com somente as covariáveis 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:

$$f_S(\mathbf{x}_S) \approx E[f(\mathbf{x}|\mathbf{x}_S)] \approx \frac{1}{k} \sum_{i=1}^{k} f(\mathbf{x}_{\bar{S}}^{(i)}, \mathbf{x}_S) \tag{2.3.4}$$

Ou seja, para x_s fixo, toma-se variações do conjunto de variáveis que não pertencem ao conjunto $\mathbf{x}_S$, e calcula-se a média, conforme a Equação 2.3.4. Isso resulta na criação de uma estimativa para $f(\mathbf{x})$, levando em consideração apenas as variáveis presentes em $\mathbf{S}$. A Figura 9 ilustra o cálculo de 2, 3, 4, considerando $x = \{x_1, x_2, x_3, x_4, x_5\}$, $x_S = \{x_1, x_3, x_4\}$ e $x_{\bar{S}} = \{x_2, x_5\}$.

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.



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

Ao analisar a Figura 9, tem-se que as covariáveis X_1, X_3 e X_4 representam o conjunto $\mathbf{S}$. Ao fixar x_1, x_3 e x_4 dessas variáveis, respectivamente, é definido o conjunto $\mathbf{x}_S$. Para obter $f_{\mathbf{S}}(x_{\mathbf{S}})$, que seria resultado do modelo somente com as covariáveis presentes em $\mathbf{x}_S$, calcula-se f para cada observação do conjunto de dados, fixando 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, a média desses valores é a estimativa $f_{\mathbf{S}}(x_{\mathbf{S}})$.

Com $f_{\mathbf{S}}(x_{\mathbf{S}})$ estimado, é possível calcular a Equação 2.3.3. Os dados que a técnica SHAP utiliza são divididos em 2 tipos: as observações necessárias para a estimação de $f_{\mathbf{S}}(x_{\mathbf{S}})$ e as observações em que se deseja calcular o valor de SHAP.

A avaliação do desempenho dos modelos é fundamental para mensurar 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 (GERON, 2019).

2.3.3 Matriz de confusão

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

Tabela 1: Matriz de confusão

2.3.4 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:

$$\text{Acurácia} = \frac{\text{Número de predições corretas}}{\text{Número total de predições}} = \frac{VP + VN}{VP + VN + FP + FN} \tag{2.3.5}$$

Essa métrica fornece uma visão geral do desempenho do modelo, indicando a porcentagem de instâncias corretamente classificadas. No entanto, a acurácia pode ser uma medida inacurada 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.

2.3.5 Precisão

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

$$\text{Precisão} = \frac{VP}{VP + FP} \tag{2.3.6}$$

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).

2.3.6 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 identificados pelo modelo, em relação ao total de exemplos positivos

existentes.

$$\text{Recall} = \frac{VP}{VP + FN} \tag{2.3.7}$$

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.

2.3.7 F1-score

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

$$F1 = 2 \cdot \frac{\text{Precisão} \cdot \text{Recall}}{\text{Precisão} + \text{Recall}}. \tag{2.3.8}$$

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 envolve a modelagem do conjunto de dados *"Loan Data for Dummy"*, retirado da plataforma *Kaggle*, 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 resposta que desejamos modelar 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, a "Condição do empréstimo"é uma variável binária, onde "0"indica um empréstimo em boas condições e "1"indica o contrário.

3.1.1 Variáveis

A base de dados é composta por 30 variáveis, incluindo a variável resposta, e existem 887379 observações. A fim de estudar "Condição do empréstimo"foram utilizadas algumas variáveis presentes na base de dados, como:

1. **Tempo de emprego**: Representa o tempo (em anos) de emprego do solicitante expresso numericamente.

2. **Tipo de residência**: Indica o status de moradia do solicitante(por exemplo: pro-

prietário, inquilino ou outra forma de ocupação residencial).

3. **Renda anual**: Indica a renda anual do solicitante, uma medida crucial para avaliar a capacidade de pagamento do empréstimo.

4. **Valor do empréstimo**: Representa o valor do empréstimo solicitado pelo requerente.

5. **Prazo**: Indica o prazo do empréstimo, especificando o período de tempo durante o qual o empréstimo deve ser reembolsado (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(por exemplo: como consolidação de dívidas, compra de casa, educação, entre outros).

8. **Tipo do juros**: Indica a natureza dos pagamentos de juros(por exemplo: se são fixos ou variáveis).

9. **Taxa de juros**: Representa a taxa de juros associada ao empréstimo.

10. **Grau**: Refere-se à classificação de risco do tomador de empréstimo atribuída pela instituição financeira (por exemplo: 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**: Indica o valor da 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.

17. **Condição do empréstimo**: Representa a situação do empréstimo.

18.

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. Remoção de variáveis irrelevantes para o estudo;
2. Exclusão de variáveis que geram a mesma informação;
3. Extração de variáveis presentes em apenas uma das categorias da variável resposta.

No 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. exclui variáveis que desempenham funções em apenas uma das categorias da variável resposta. Como é o caso da variável "Recuperações totais", visto que esta 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 são sempre zero.

Com isso a base de dados ficou da seguinte maneira:

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

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

Com o tratamento dos dados realizado, foi feita a separação dos dados para a modelagem dos dois modelos.

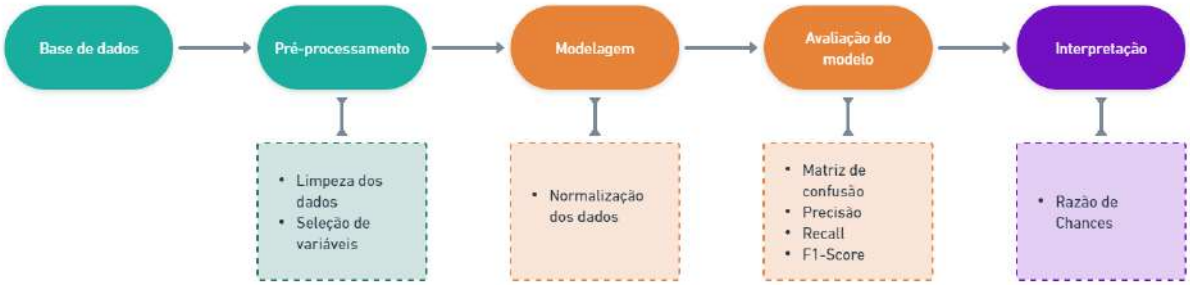
Tipo de dado	**% dos dados originais**
Treino	70%
Validação	10%
Teste	20%

Tabela 3: Divisão dos dados para a modelagem dos modelos de regressão logística e redes neurais

3.2 Modelagem dos dados

3.2.1 Regressão logística

Figura 10: Fluxograma da implementação do modelo logístico



Fonte: Autoria própria

A metodologia para a implementação da regressão logística incluiu várias etapas essenciais para garantir a robustez e eficácia do modelo. Inicialmente, foi realizada uma cuidadosa etapa de pré-processamento, que envolveu a limpeza e tratamento das variáveis. Durante essa fase, foram identificados e tratados possíveis valores ausentes, outliers e erros nos dados, contribuindo para a qualidade do conjunto de dados.

Outro aspecto crítico foi a categorização adequada das variáveis, quando aplicável. Isso incluiu a transformação de variáveis categóricas em formatos adequados para análise estatística, garantindo que todas as variáveis estivessem em uma forma consistente para o modelo de regressão logística.

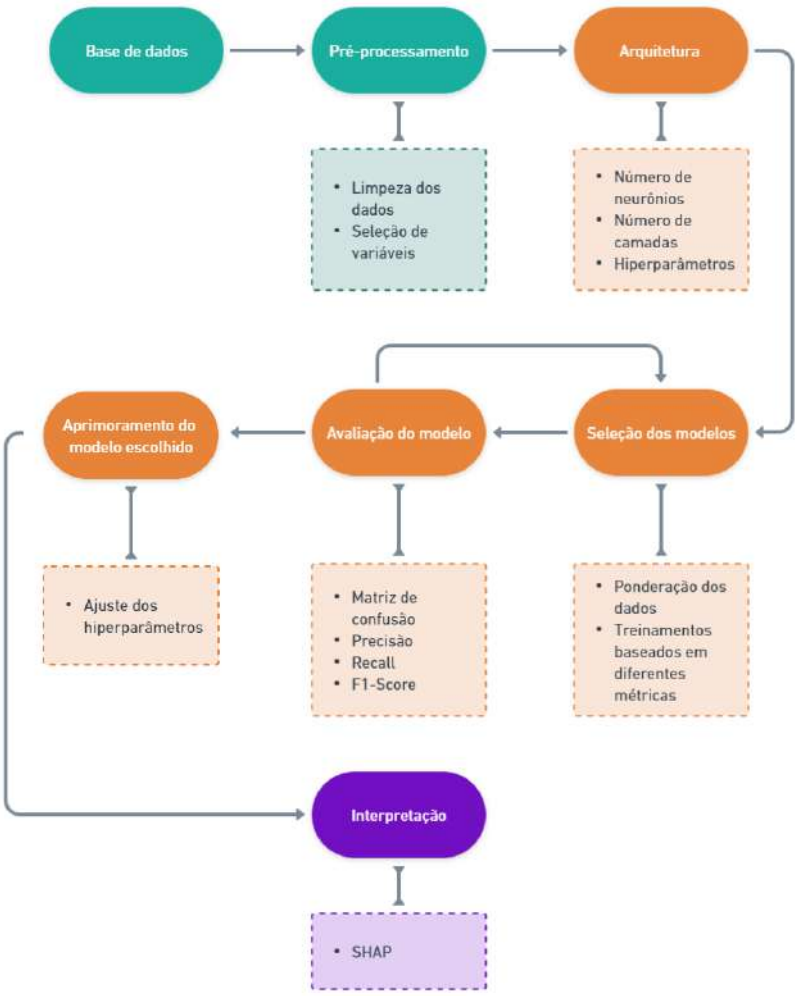
Para facilitar a interpretação do modelo, os valores das variáveis normalizados, usando uma padronização que retorna dados em uma escala com média zero e desvio padrão unitário.

Dada a natureza desbalanceada do conjunto de dados, onde as classes "Empréstimo bom"e "Empréstimo ruim"têm proporções significativamente diferentes, foi aplicado um corte diferente de 0.5 no momento de dividir as predições do modelo. Foi utilizada a proporção da categoria minoritária em relação à classe majoritária como ponto de corte (AGRESTI, 2002).

Tendo o modelo definido, foi feita uma avaliação das predições do modelo no conjunto de teste, usando métricas apropriadas para problemas de classificação, como precisão, recall, F1-score e matriz de confusão. Essas métricas permitiram uma compreensão abrangente do desempenho do modelo, especialmente no que diz respeito à capacidade de prever corretamente os casos de "Empréstimo ruim"e "Empréstimo bom".

3.2.2 Rede neural

Figura 11: Fluxograma da implementação da rede neural



Fonte: Autoria própria

A construção e ajuste das redes neurais envolveu diversas etapas cruciais para garantir a eficácia do modelo. Inicialmente, foi realizada uma fase de pré-processamento, que consistiu na limpeza e tratamento das variáveis. Durante esse estágio, foram tratados valores ausentes, outliers e possíveis erros nos dados, contribuindo para a qualidade do conjunto de dados. Essa etapa foi a mesma apresentada pelo modelo logístico.

A escolha da arquitetura inicial foi um passo importante. Foi necessário definir o número de camadas ocultas, a quantidade de neurônios em cada camada e a função de ativação a ser utilizada. Essas escolhas iniciais foram baseadas tanto em conhecimentos prévios do problema quanto em experimentações para encontrar a configuração que melhor

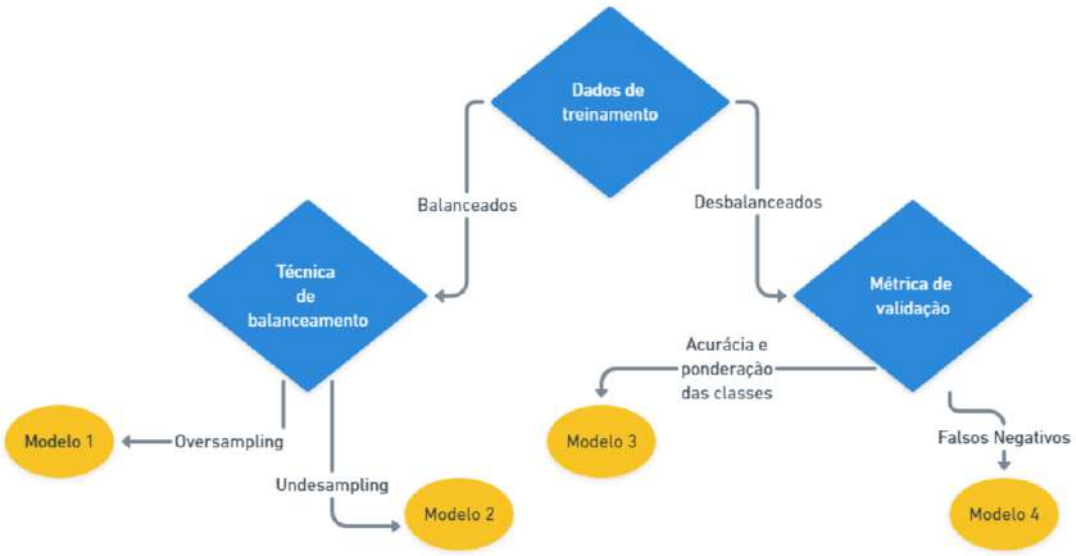
se adequava aos dados. E para isso foi utilizado uma rede neural baseada na Figura 12:

Figura 12: Arquitetura de rede neural inicial

Diversos modelos de redes neurais foram desenvolvidos e treinados em contextos variados, considerando abordagens distintas, tais como:

- Tratamento para dados desbalanceados;
- Ajuste dos modelos otimizando métricas de treinamento variadas;
- Ponderação das classes da variável resposta.

Figura 13: Fluxograma da escolha dos modelos de redes neurais estimados



Fonte: Autoria própria

Conforme ilustrado na Figura 13, o treinamento dos modelos com dados equilibrados, foram empregados dois processos distintos: *undersampling* e *oversampling*. No âmbito da modelagem, em que se priorizaram métricas específicas de treinamento, além da acurácia, métrica padrão, foram considerados o Recall e o número de Falsos Negativos. Este último é particularmente crucial em cenários de empréstimos, sendo considerado o erro mais relevante. Por fim, foi utilizado também um modelo com ponderação nas classes, buscando assim equilibrar a classe majoritária. Essas abordagens visaram explorar

diferentes aspectos do treinamento para encontrar a configuração mais eficaz, adaptada às situações do problema em análise.

Cada modelo foi treinado utilizando o conjunto de treinamento e validado para avaliar seu desempenho, no conjunto de validação. As métrica de desempenho, geralmente relacionadas à precisão, recall e F1-score, foi usada para comparar e selecionar os melhores modelos.

O modelo mais promissor foi então submetido a uma etapa de ajuste de hiperparâmetros. Ajustes finos nos hiperparâmetros, como taxa de aprendizado, número de épocas de treinamento e tamanho do lote, foram realizados para otimizar o desempenho do modelo, utilizando o conjunto de validação. A arquitetura final do modelo está ilustrada na Figura 14.

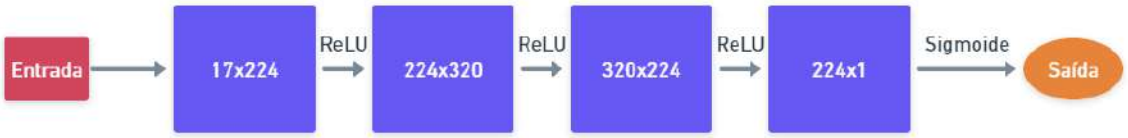


Figura 14: Arquitetura final da rede neural

O modelo que apresentou a melhor performance foi utilizando as classes ponderadas e a acurácia como métrica de desempenho.

A avaliação final do modelo ocorreu no conjunto de validação e teste. Essa etapa foi crucial para garantir que o modelo não apenas se ajustasse bem aos dados de treinamento, mas também generalizasse de maneira eficaz para novos dados. As métricas de desempenho foram novamente utilizadas para avaliar a capacidade do modelo de fazer previsões precisas e úteis, no conjunto de teste.

3.3 Interpretação dos modelos

Definindo os modelos logístico e o de redes neurais, foi feita a interpretação de ambos. E realizar a interpretações dos modelos é uma etapa crucial na análise de dados, mostrando o funcionamento e as relações das variáveis no contexto do problema em questão.

No caso do modelo logístico, a interpretação se concentrou nos parâmetros estimados para cada variável. Esses coeficientes fornecem uma medida da magnitude e direção da influência de cada variável na predição da variável resposta. Além disso, a interpretação envolveu a análise da Razão de Chances (RC), que expressa como a chance de o evento ocorrer se torna multiplicativamente maior ou menor com a mudança em uma

unidade na variável explicativa.

Para a interpretação da rede neural, utilizou-se a técnica SHAP (*SHapley Additive exPlanations*). Essa abordagem proporciona uma compreensão mais profunda ao atribuir a contribuição de cada variável para a saída do modelo em nível individual. Com o auxílio de gráficos SHAP, pôde-se observar como cada variável influencia as predições e identificar padrões de comportamento em diferentes cenários.

4 Resultados

Esta seção inicia explorando a relação entre as variáveis explicativas e a variável resposta. Em seguida, resultados derivados tanto do modelo logístico quanto da rede neural serão detalhadamente apresentados, destacando a ênfase na interpretação de ambos os modelos. Além disso, será conduzido um benchmark comparativo entre as duas abordagens, proporcionando uma análise crítica de suas performances.

4.1 Análise descritiva

4.1.1 Condição do empréstimo

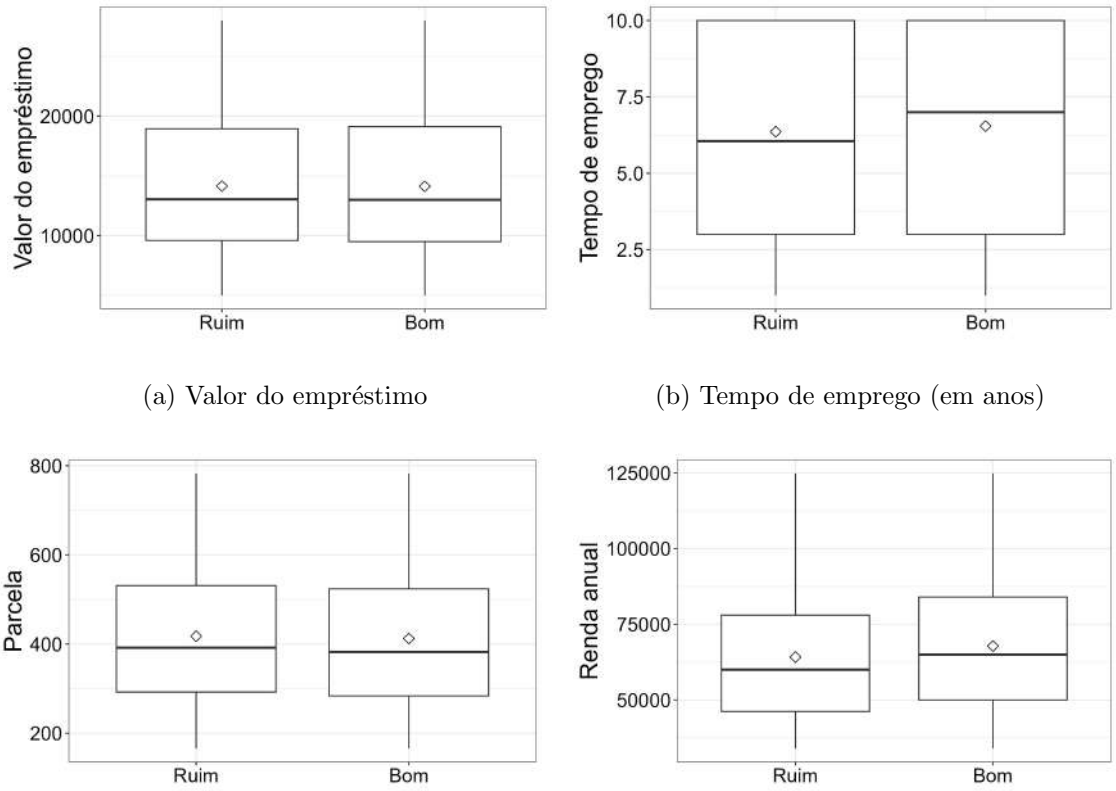
A variável "Condição do empréstimo"é a variável resposta desse estudo, como foi definido anteriormente. Com isso temos o seguinte comportamento dessa variável:

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

Tabela 4: Descrição da variável resposta

A Tabela 4 mostra a distribuição da variável "Condição do empréstimo". Uma variável composta majoritariamente por observações do tipo "Empréstimo bom", 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

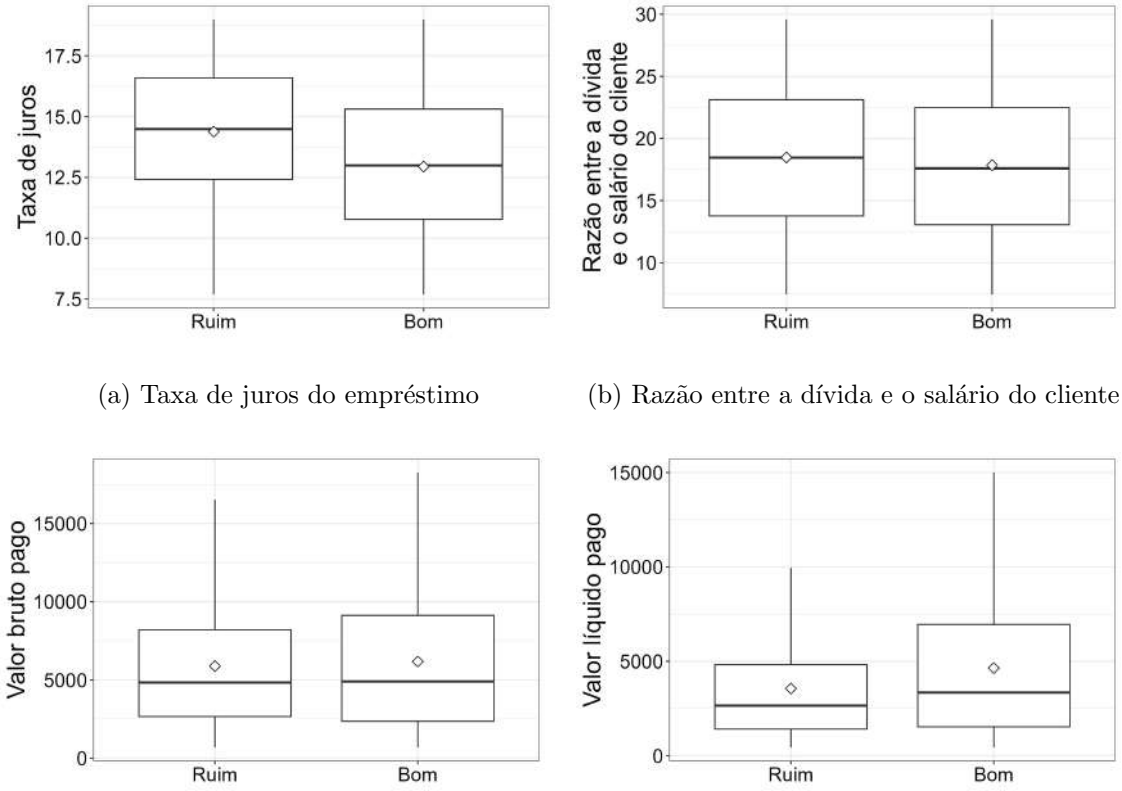
(b) Tempo de emprego (em anos)

(c) Valor da parcela do emprésitmo

(d) Renda anual (em dólar)

Figura 15: Boxplot das covariáveis com relação a condição do empréstimo - Parte 1

O comportamento da variável resposta nas Figuras 15a e 15c demonstrou semelhanças, visto que, 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 15b também apresenta um comportamento semelhante entre as classes "Empréstimo ruim"e "Empréstimo bom", sendo que a mediana do tempo de trabalho dos clientes rotulados como "Empréstimo ruim"foi inferior em comparação ao outro caso. Por fim, a Figura 15d indica que clientes com uma renda anual maior tendem a ser categorizados como "Empréstimo bom".



(a) Taxa de juros do empréstimo

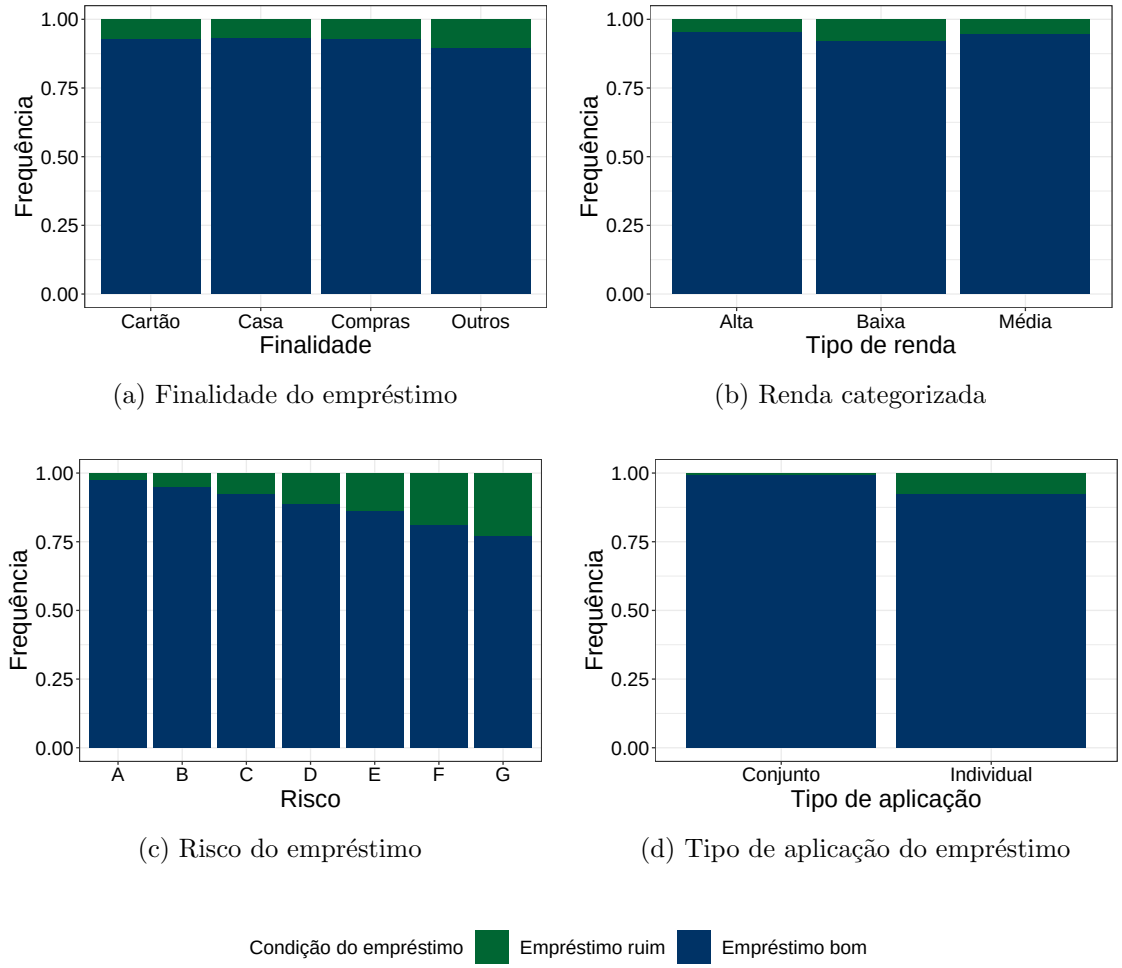
(b) Razão entre a dívida e o salário do cliente

(c) Valor bruto do emprétimo pago

(d) Valor líquido do emprétimo pago

Figura 16: Boxplot das covariáveis com relação a condição do empréstimo - Parte 2

A Figura 16a evidencia uma relação entre taxas de juros elevadas e empréstimos considerados ruins. A Figura 16b complementa a informação fornecida pela Figura 15d, indicando que clientes com renda mais elevada tem ligeira propensão a cumprir adequadamente com seus pagamentos. As Figuras 16c e 16d seguem padrões semelhantes, sugerindo que clientes que quitaram o empréstimo têm uma leve tendência de serem rotulados como bons pagadores.



(a) Finalidade do empréstimo

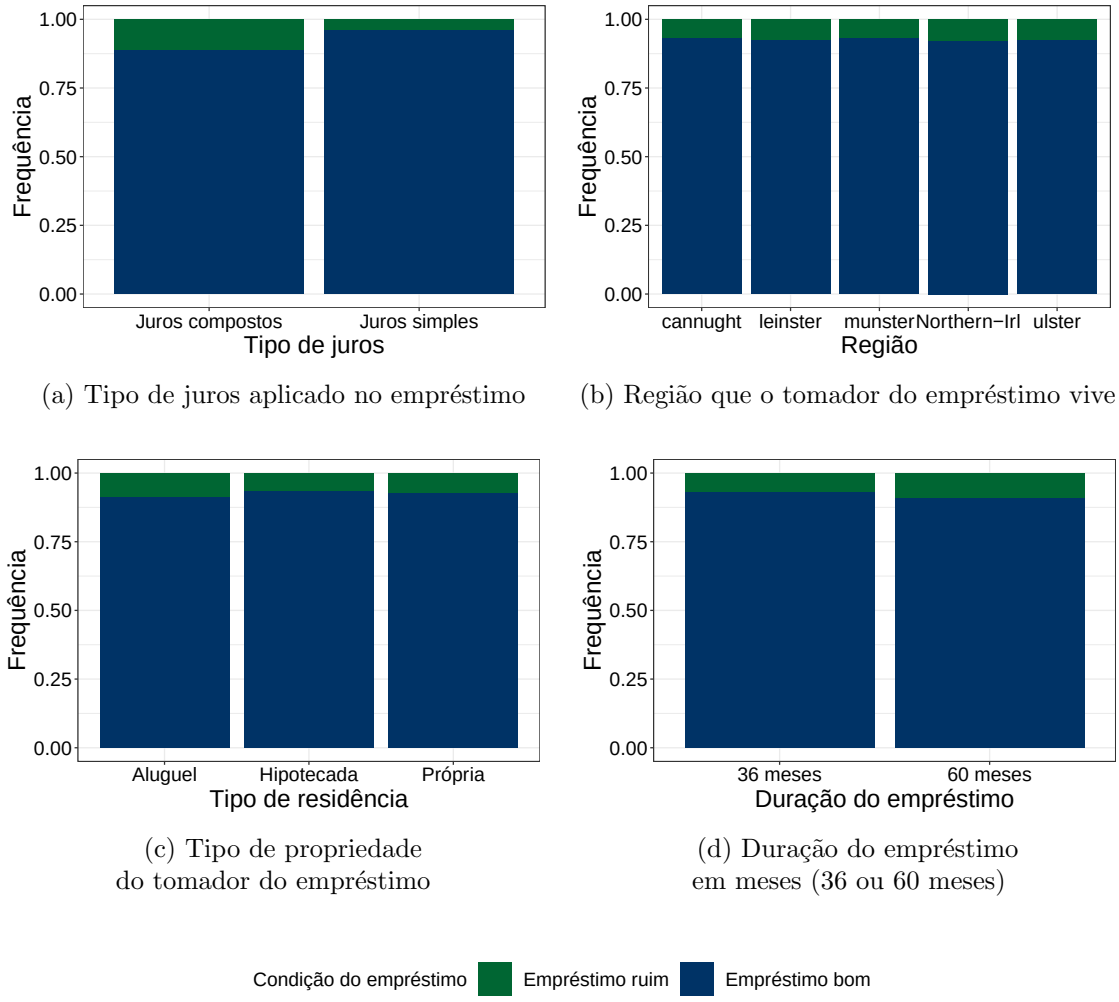
(b) Renda categorizada

(c) Risco do empréstimo

(d) Tipo de aplicação do empréstimo

Figura 17: Gráfico de barras das covariáveis com relação a condição do empréstimo - Parte 1

A Figura 17a ilustra que as categorias da variável "Finalidade"seguem a proporção natural da condição do empréstimo, indicado na Tabela 4. Na Figura 17b, as categorias de renda "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 17c revela um padrão de crescente na proporção dos empréstimos ruins, 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 17d, 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 18: Gráfico de barras das covariáveis com relação a condição do empréstimo - Parte 2

Na Figura 18, o gráfico 18a evidencia que empréstimos obtidos sob juros compostos possuem uma proporção mais elevada de rotulações ruins em comparação com empréstimos sob juros simples. As Figuras 18b e 18c apresentam uma proporção natural refletida pela distribuição das categorias da variável resposta, conforme apresentado na Tabela 4. Já a Figura 18d revela uma proporção ligeiramente maior de empréstimos ruins quando estes tem uma duração maior.

4.2 Regressão logística

Como visto na Equação 2.1.1, a fórmula do modelo logístico é dada por:

$$P(Y = 1|x_1, x_2, ..., x_k) = \frac{1}{1 + e^{-(\beta_0 + x_1\beta_1 + x_2\beta_2 + ... + x_k\beta_k)}} \tag{4.2.1}$$

Sendo Y=1 a classificação de clientes com “Empréstimos ruins”e Y=0, os clientes

com “Empréstimos bons”. Dessa forma, os seguintes resultados foram encontrados:

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 5: Estimativa dos coeficientes do modelo logístico e o erro padrão associado (todas as variáveis foram significativas ao nível de significância de 5%)

Considerando que a escala das covariáveis foi padronizada para ter média 0 e desvio padrão unitário, é possível perceber que, com os resultados apresentados na Tabela 5, fica evidente que as variáveis “Valor líquido pago”e “Valor bruto pago”exercem uma influência na maior probabilidade 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”: $\exp(-4.733) = 0.008$, o que sugere que, mantendo todas as outras variáveis constantes, aumentar em 1 unidade o valor líquido pago, torna 125 vezes a chance do empréstimo ser classificado como ruim.
- “Valor bruto pago”: $\exp(3.321) = 27.68$, indicando que, ao manter todas as outras variáveis constantes, aumentar em 1 unidade o valor bruto pago, torna 27 vezes a chance do empréstimo ser classificado como bom.

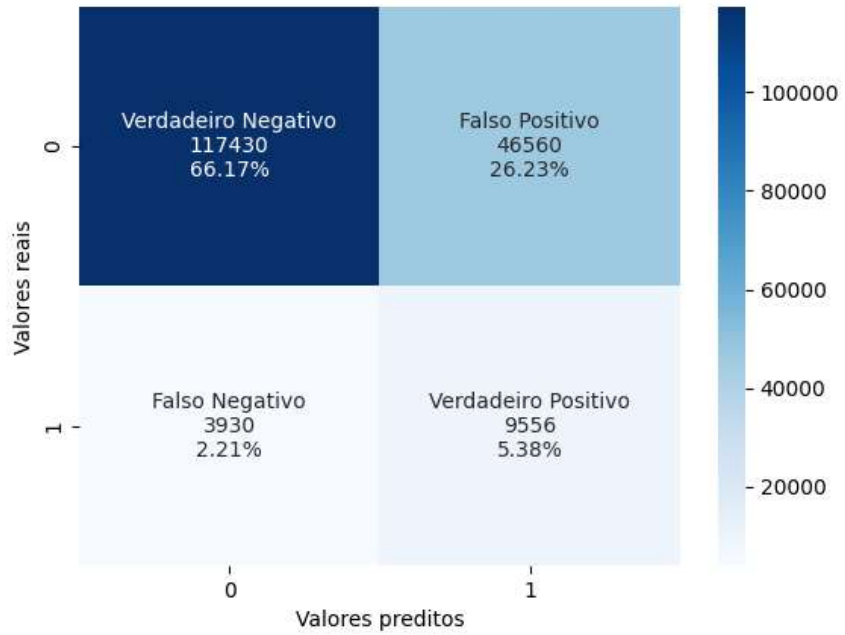


Figura 19: Matrix de confusão do modelo logístico

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.

O modelo logístico fez um pouco mais de 30% de previsões da classe "1". Esse cenário resultou em um alto número de Falsos Positivos. Em contrapartida, o número de Falsos Negativos ficou menor, resultando em menos de 3% das previsões da amostra de teste sendo incorretamente rotuladas como classe "0".

	Precisão	**Recall**	**F1-Score**	**Tamanho da amostra**
0	0.928	0.716	0.823	163990
1	0.170	0.708	0.274	13486
Média macro	0.569	0.712	0.548	177476
Média ponderada	0.907	0.715	0.781	177476
Acurácia				0.715

Tabela 6: Métricas de avaliação do modelo logístico

Com base nos dados apresentados na Tabela 6 e na Figura 19, observamos que o modelo exibe uma acurácia elevada. Ele é capaz de fazer previsões precisas em um número considerável dos casos, alcançando uma taxa de 71,5% de classificações corretas no conjunto de teste.

Ao examinarmos a precisão do modelo, observamos uma taxa de acerto de 56,9%

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, pois ao observarmos as precisão da classe 1, é possível notar um valor menor que 20%.

Ao avaliar o Recall do modelo logístico, observamos, em média, valores maiores em comparação com a precisão. O recall médio é de 71,2%, indicando que, ao analisar as porcentagens das rótulos reais, o modelo conseguiu acertar um valor considerável. Esse desempenho é atribuído ao baixo número de falsos negativos no modelo, visto que, ao considerar o total de "Empréstimos ruins"(13.486), o modelo acertou apenas 9554 desses casos.

O F1-score acaba refletindo a real situação do modelo, pois ele balanceia os resultados de ambas as métricas. O F1-score médio apresentado foi de 54,8%.

O modelo apresentou resultados interessantes, dado a sua arquitetura não tão robusta, uma taxa geral de acerto de 71,5%. Na classe minoritária, a classe "1", o modelo foi pouco preciso, acertando menos de 20% dos casos. Este desempenho inferior sugere limitações na capacidade do modelo, indicando a necessidade de refinamentos ou considerações adicionais para melhorar seu melhor desempenho preditivo.

4.3 Rede neural

Esse seção vai abordar os resultados obtidos pelo modelo de rede neural.

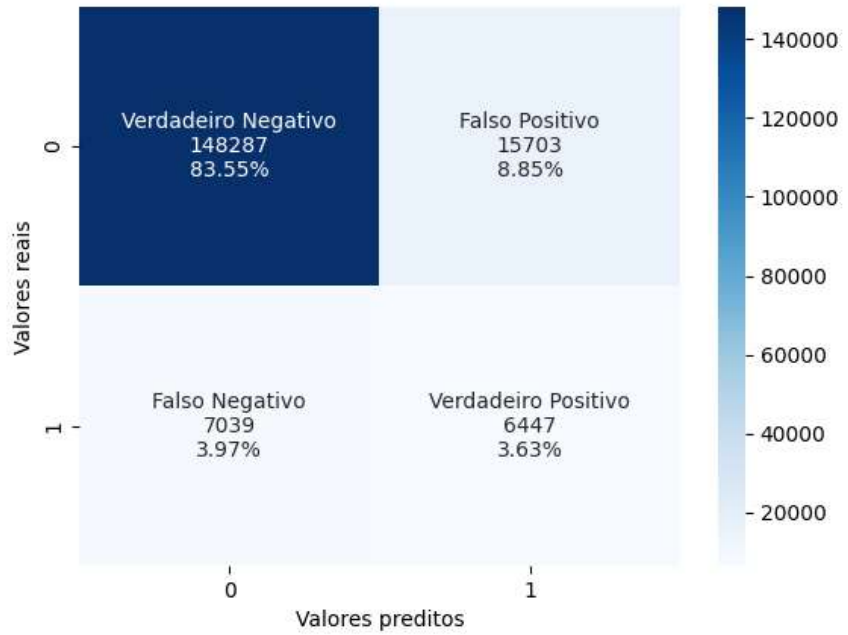


Figura 20: Matrix de confusão da rede neural

A Figura 20 indica que aproximadamente 12% das previsões do modelo na amostra de teste consistem em empréstimos ruins, uma estimativa ligeiramente distante da distribuição real apresentada na Tabela 4. Por outro lado, o número de empréstimos classificados como "Empréstimos bons", diminuiu, representando mais de 84% da amostra de teste.

	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 7: Métricas de avaliação da rede neural

Os resultados apresentados na Tabela 7 indicam um desempenho satisfatório do modelo, especialmente em termos de acurácia e métricas avaliadas para a classe "0". Contudo, ao comparar esses resultados com as métricas da classe "1", percebe-se que o modelo ainda é influenciado pelo elevado número de observações na classe "0". Por outro lado, uma análise mais detalhada do Recall da classe "1"revela que o modelo conseguiu reduzir significativamente o número de Falsos Negativos, identificando corretamente quase metade dos empréstimos considerados ruins na base de dados. Por outro lado, a precisão da

classe "1"diminuiu, refletindo que menos de 30% das previsões do modelo foram corretas nesse contexto, como evidenciado na Tabela 7.

Em termos gerais, as decisões relacionadas à arquitetura do modelo, seus hiperparâmetros, estratégia de treinamento e outros fatores contribuíram para que o modelo de redes neurais realizasse previsões de boa qualidade, lidando melhor com o desbalanceamento dos dados. Portanto, buscar aprimorar ainda mais essa arquitetura pode ser uma abordagem promissora na busca por resultados ainda melhores.

4.4 Interpretação da rede neural

Após definir o modelo de rede neural e examinar seus resultados, esta seção aborda a interpretação do modelo. Para isso, foram construídos gráficos com base nos resultados do SHAP, destacando as variáveis de maior importância no resultado final. Utilizando uma amostra de 80 observações, o valor de SHAP foi calculado para cada observação, permitindo uma análise tanto individual quanto conjunta dessa amostra.

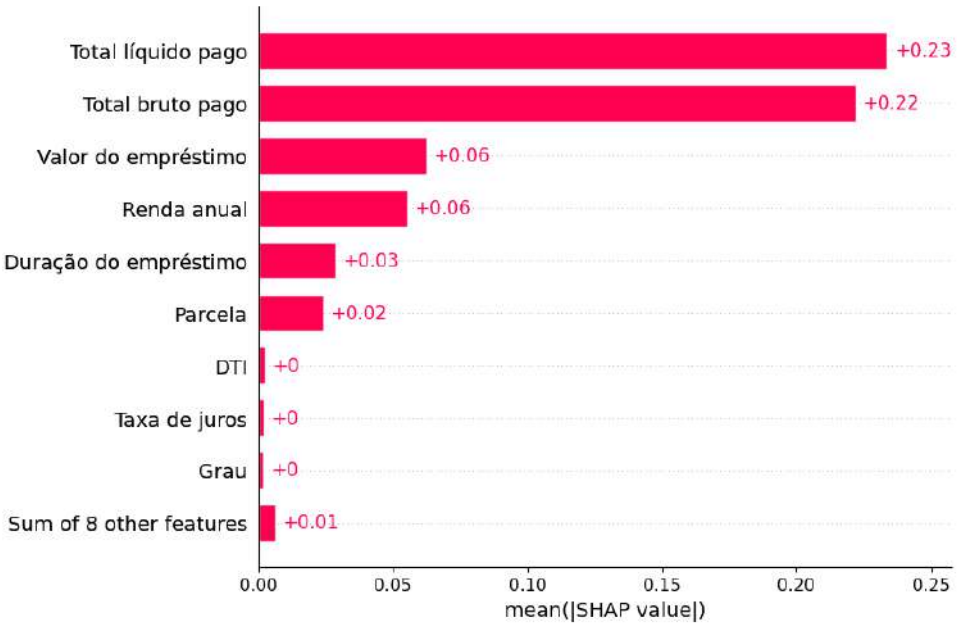


Figura 21: Média absoluta dos valores de SHAP de cada covariável

A Figura 21 exibe a média absoluta dos valores de SHAP para cada covariável nas 80 observações consideradas. Essa representação oferece uma visão sobre quais variáveis o modelo considerou mais relevantes durante as predições, destacando a magnitude da contribuição de cada variável. Vale notar que, por se tratar de valores absolutos, o gráfico não proporciona informações sobre se a contribuição de cada variável é positiva ou negativa. Entretanto, os próximos gráficos irão elucidar essa questão ao detalhar a contribuição específica de cada variável.

Ao analisar cada variável no gráfico, destaca-se que "Total líquido pago"e "Total bruto pago"são as que mais contribuíram para o resultado final do modelo. O gráfico enfatiza a relevância de nove variáveis, omitindo o restante devido à sua baixa contribuição. Ao observar as variáveis omitidas, percebe-se que, somadas, suas contribuições aproximam-se de 0.01, evidenciando a sua baixa influência no resultado final do modelo de rede neural.

Os próximos gráficos proporcionam a visualização dos valores SHAP para cada variável em observações individuais, ilustrando cenários distintos que podem aparecer nos problemas de classificação binária. Os gráficos também revelam os valores específicos observados para cada covariável de uma dada observação.

As Figuras 22 e 23 mostram casos onde o modelo acertou suas predições, tanto para casos de "Empréstimos ruins"(Figura 22) quanto para "Empréstimos bons"(Figura 23).

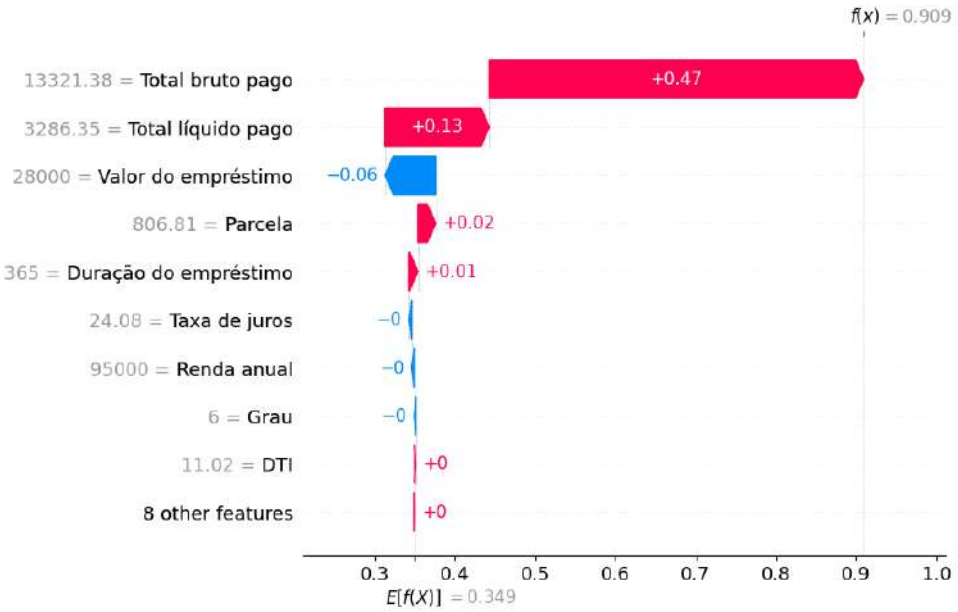


Figura 22: Valor de SHAP para uma observação do tipo Verdadeiro Positivo

Na Figura 22, temos que, no eixo y, os valores observados para as covariáveis dessa observação específica do banco de dados, enquanto que nas barras coloridas do gráfico, temos indicado o valor de SHAP para cada covariável (contribuições positivas, em vermelho, e contribuições negativas, em azul). A soma de todos os valores SHAP retorna a predição dessa observação, nesse exemplo, $f(x) = 0,909$, conforme indicado no eixo x.

No cenário acima, em que o modelo acertou a classificação de um empréstimo como ruim, observa-se que as variáveis que mais contribuíram foram as mesmas observadas na Figura 21, comportamento esse que vai prevalecer nos demais casos. Ao analisar os

valores observados dessas duas covariáveis, nota-se que, para esse cliente específico, ainda resta um montante significativo do empréstimo a ser pago, totalizando quase 90% do valor líquido pendente e quase 50% do valor bruto pendente. Uma diferença muito grande entre o valor do empréstimo e o que falta a ser pago pode ser um dos fatores que está ocasionando a rotulação dessa observação como "Empréstimos ruins".

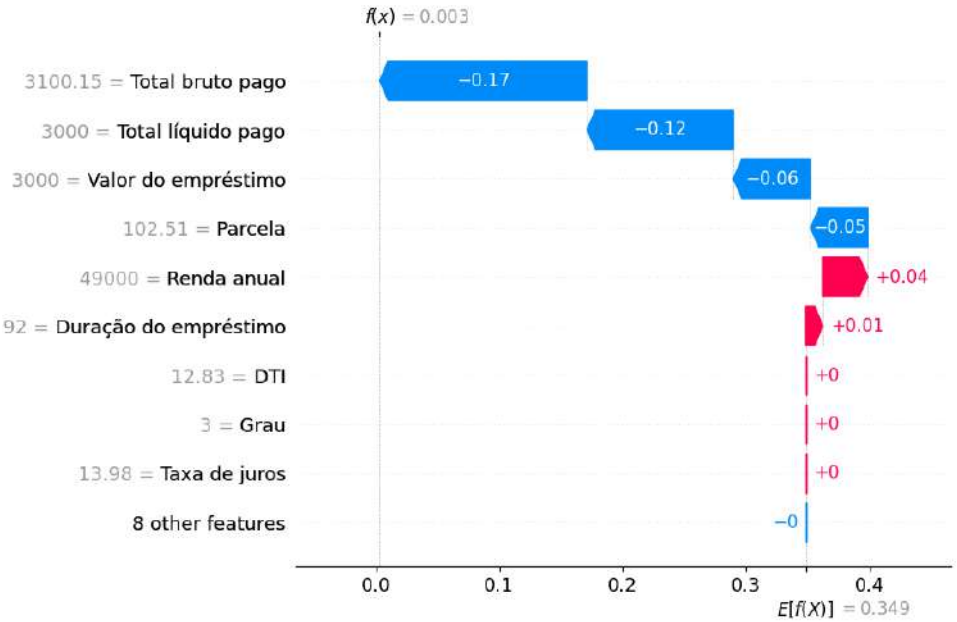


Figura 23: Valor de SHAP para uma observação do tipo Verdadeiro Negativo

Ao analisar a Figura 23, que representa o cenário em que o modelo acertou a rotulação de um empréstimo bom, nota-se um valor final bem próximo de 0 (f(x) = 0.003). Indícios de que o modelo teve mais confiança ao realizar essa predição. Ao analisar os valores de cada variável, é possível perceber que o cliente já está finalizando ou finalizou o empréstimo, dado que as variáveis de pagamento do empréstimo chegaram no valor real do empréstimo.

Analisando as Figuras 22 e 23, é possível observar a flexibilidade do modelo em lidar com diferentes valores da mesma covariável. Quando o modelo encontra valores que indicam um empréstimo ruim, atribui valores positivos para a contribuição das variáveis "Total líquido pago"e "Total bruto pago", aproximando-as de 1. Da mesma forma, para empréstimos bons, o modelo adiciona uma contribuição negativa, direcionando o resultado para 0. Isso demonstra como o modelo responde de forma dinâmica e não linear às variações nos valores das variáveis.

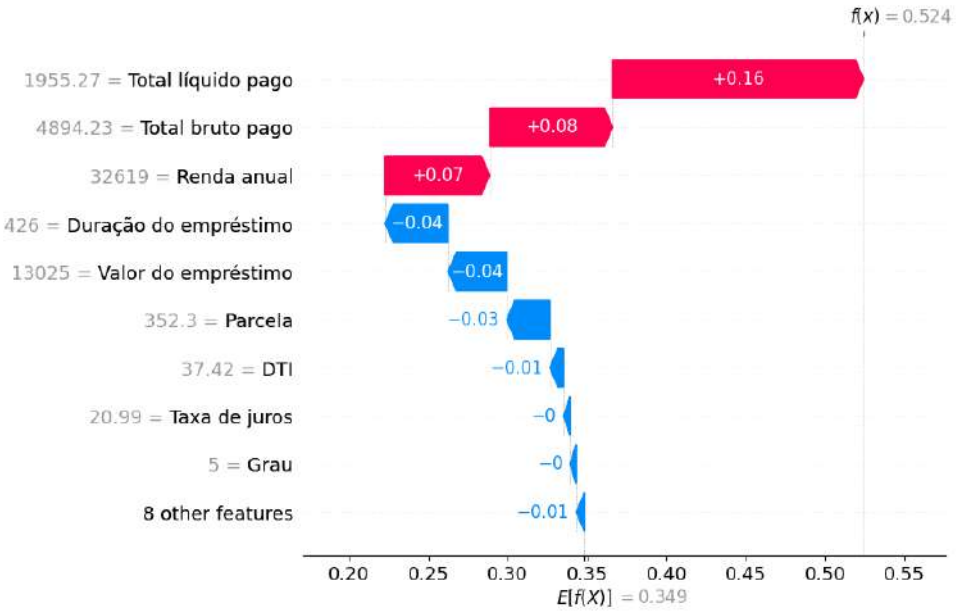


Figura 24: Valor de SHAP para uma observação do tipo Falso Positivo

Ao analisar um exemplo em que o modelo erroneamente classifica como "Empréstimo ruim", conforme ilustrado na Figura 24, destacam-se as características já discutidas nas interpretações anteriores. O resultado do modelo foi muito próximo do limiar de decisão, que é 0,5, indicando que o modelo teve uma maior indecisão ao realizar a predição desse caso.

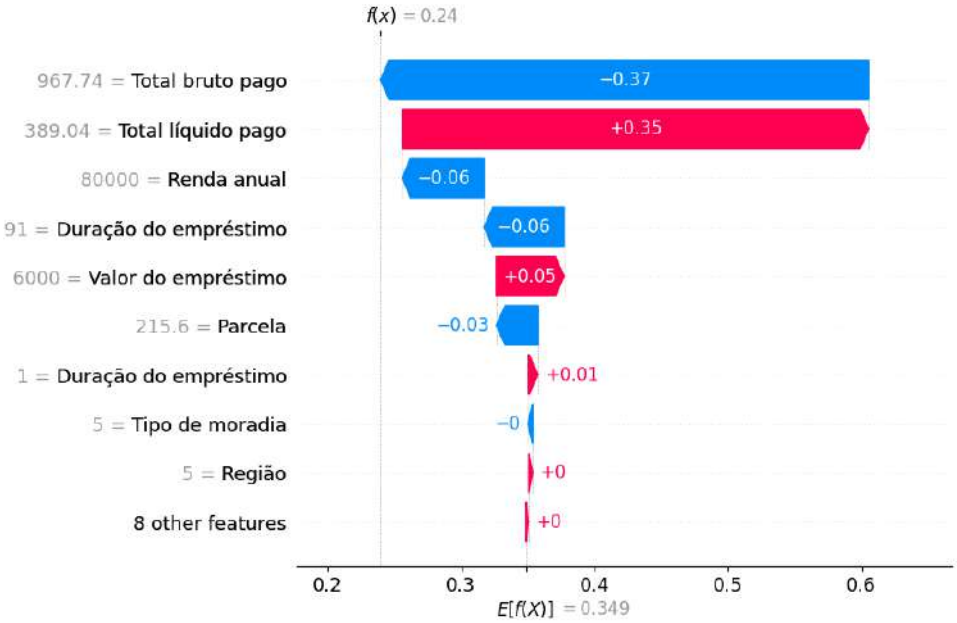


Figura 25: Valor de SHAP para uma observação do tipo Falso Negativo

Observando a Figura 25, que retrata um cenário em que o modelo classifica de forma equivocada como "Empréstimo bom", é possível perceber um comportamento de

divergência entre as duas covariáveis que mais contribuem com o resultado do modelo. Esse padrão de divergência não foi observado nos casos anteriores, e pode indicar uma incerteza na predição.

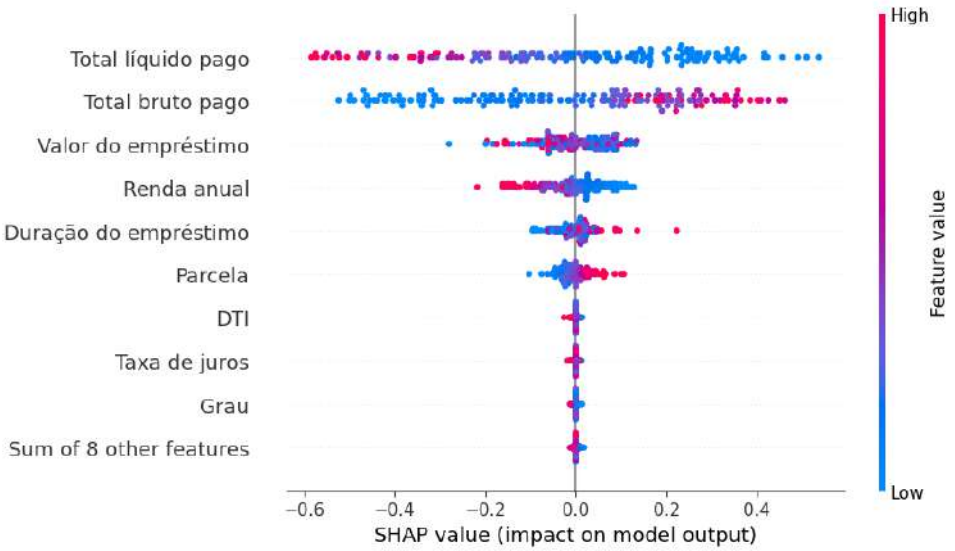


Figura 26: Valores de SHAP para as 80 observações utilizadas

A Figura 26 apresenta o comportamento do valor de SHAP para cada covariável entre as 80 observações utilizadas no experimento. No gráfico, o eixo x refere-se à distribuição do valor de SHAP, enquanto o eixo y representa cada variável. O eixo das cores ilustra a distribuição dos valores observados da covariável, sendo que cores mais quentes indicam valores maiores e cores mais frias indicam valores menores.

Ao analisar as variáveis que mais influenciam o resultado do modelo, observa-se uma distribuição mais dispersa para o valor de SHAP. Para a primeira variável, elevados valores da covariável "Total líquido pago"tendem a impactar negativamente no resultado do modelo, ou seja, diminuir a probabilidade de classificação como um "Empréstimo Ruim"enquanto a segunda variável apresenta o comportamento oposto, com valores mais baixos do "Total bruto pago", contribuindo negativamente no resultado do modelo.

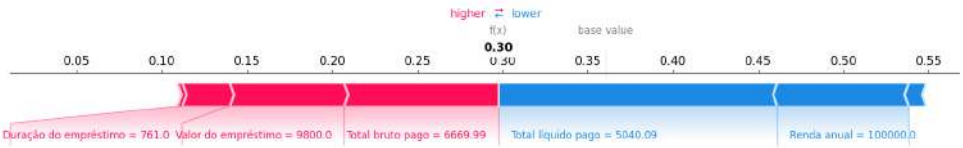


Figura 27: Gráfico de força do SHAP para uma observação

A Figura 27 ilustra a "força"de contribuição de cada variável no modelo. Este gráfico proporciona uma visão clara das variáveis que tiveram impacto positivo e negativo

no resultado dessa observação específica. Cada barra representa a intensidade da contribuição de uma variável específica, sendo que aquelas com maior influência concentram-se no centro, enquanto as de menor influência ficam nas extremidades. A figura é centrada no valor final predito pelo modelo, que, neste caso, é observado como $f(x) = 0.3$, indicando um empréstimo classificado como bom.

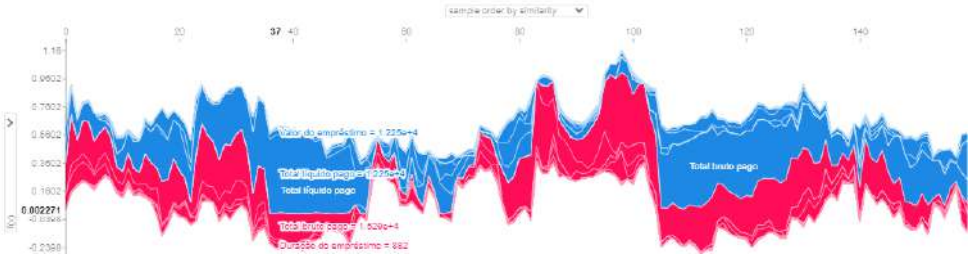


Figura 28: Gráfico de força para múltiplas observações

A Figura 28 é uma generalização da Figura 27, considerando 160 observações. Este gráfico é um recorte obtido de uma visualização dinâmica gerada pelo pacote SHAP. Devido à natureza dinâmica do gráfico original, ao transformá-lo em uma imagem estática, parte de sua capacidade de representação foi limitada. No entanto, é possível observar alguns padrões nesse recorte. Por exemplo, a predominância do vermelho em determinado recorte vertical, indica a previsão de um "Empréstimo ruim". Da mesma forma, que a predominância da cor azul indica predições próximas de 0.

4.5 Benchmark entre regressão logística e redes neurais

Ao obter interpretações do modelo de redes neurais, abre-se a possibilidade de realizar comparações entre esses modelos com os regressão logística no âmbito interpretativo. Até recentemente, essa capacidade de interpretação estava exclusivamente associada ao modelo logístico em comparação com o modelo de redes neurais. Essa análise comparativa se soma às comparações já realizadas entre os modelos, englobando aspectos como arquitetura, tempo de treinamento, tempo de predição e os resultados gerados por ambos os modelos. Os resultados abaixo evidenciam essas comparações.

4.5.1 Complexidade da arquitetura

Modelo	Número de parâmetros
Regressão Logística	18
Rede Neural	151233

Tabela 8: Número de parâmetros

Ao examinar a Tabela 8, destaca-se a significativa disparidade na complexidade entre os modelos. Enquanto o modelo logístico possui apenas um parâmetro para cada covariável, a rede neural apresenta mais de 8400 parâmetros para cada parâmetro do modelo logístico, sendo esses distribuídos nos neurônios e camadas da rede.

4.5.2 Resultado dos modelos

Métricas	Regressão logística	Rede neural
Falsos Positivos	46560	15703
Falsos Negativos	3930	7039

Tabela 9: Comparação dos resultados de Falsos Positivos e Falsos Negativos

Ao analisar os casos em que os modelos cometeram erros, conforme apresentado na Tabela 9, é evidente que, em geral, o modelo logístico cometeu mais erros do que a rede neural. O modelo logístico registrou um total de mais de 50 mil classificações incorretas, enquanto a rede neural teve pouco mais de 18 mil erros.

Os resultados a seguir vão identificar melhor como foi o processo preditivo dos dois modelos no conjunto de teste.

Métricas	Regressão logística	Rede neural
Precisão (Classe 0)	0.928	0.954
Precisão (Classe 1)	0.170	0.291
Recall (Classe 0)	0.716	0.904
Recall (Classe 1)	0.708	0.478
F1-Score (Classe 0)	0.823	0.929
F1-Score (Classe 1)	0.274	0.362
Acurácia	0.715	0.872

Tabela 10: Comparação dos resultados da Regressão logística e Rede neural. Em verde, o modelo que obteve o melhor resultado na respectiva métrica.

Ao examinar a Tabela 10, é possível observar que a rede neural apresentou melhores resultados em quase todas as métricas. Esse desempenho destacado da regressão logística pode ter ocorrido devido a sua arquitetura. Este modelo, ao ser treinado com dados desbalanceados, enfrenta limitações devido à sua arquitetura menos complexa.

Contudo, ambos os modelo apresentaram um resultado ruim nos Falsos Positivos, métrica importante no contexto financeiro, pois permite identificar os empréstimos ruins classificados de maneira errônea.

Covariáveis	**Coeficientes da regressão logística**	**Médias absoluta dos valores de SHAP**
Total líquido pago	-4.733	0.23363
Total bruto pago	3.321	0.22619
Valor do empréstimo	-1.406	0.06298
Renda anual	-0.195	0.06132
Tempo de trabalho	-0.061	0.03105
Parcela	0.005	0.02586
DTI	-0.151	0.00242
Taxa de juros	1.412	0.00232
Risco	-1.049	0.00169
Duração do empréstimo	0.033	0.00166
Tipo de moradia	0.019	0.00117
Finalidade	0.019	0.00094
Tipo de juros	-0.462	0.00077
Região	0.038	0.00071
Prazo	-0.203	0.00068
Renda categorizada	-0.111	0.00058
Tipo da aplicação	1.848	0

Tabela 11: Relação entre os coeficientes da regressão logística com os valores absolutos de SHAP

A Tabela 11 realiza uma comparação entre os coeficientes do modelo logístico e a média dos valores absolutos de SHAP, sendo esse último calculado com base em uma amostra composta por 169 observações. A média dos valores de SHAP proporciona uma medida da magnitude das contribuições das variáveis, apresentando uma ordem de importância que pode ser comparada com a dos coeficientes da regressão logística.

Ambos os modelos exibiram semelhança nas duas variáveis que mais impactam o resultado final. No entanto, ao classificar as demais variáveis em ordem de importância, observa-se uma divergência significativa no grau de influência que esses valores exercem.

4.5.3 Tempo de execução

Examinar o tempo de predição é importante na avaliação da utilidade prática do modelo, quase tão significativo quanto as métricas de desempenho do modelo.

Estatísticas	**Regressão logística**	**Rede neural**
Mínimo	0.0	52.06
Quartil 25	0.0	53.32
Média	0.33	59.44
Mediana	0.0	56.85
Quartil 75	0.88	66.27
Máximo	1.50	92.03
Desvio padrão	0.53	7.68

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

A Tabela 12 destaca a extrema eficiência do modelo logístico em comparação com o modelo de redes neurais. O modelo logístico demonstrou um tempo de predição mais curto, com a predição mais demorada levando apenas 1,5 milissegundos, sendo mais de 60 vezes mais rápido do que a predição mais demorada do modelo de redes neurais.

Nº de observações	**Tempo de execução**
1	317s
80	7.07hrs

Tabela 13: Tempo de execução para realizar a interpretação das variáveis

Ao contrário do modelo logístico, onde não há um tempo de execução associado à interpretação, pois a interpretação é derivada dos coeficientes estimados, a situação é diferente no modelo de redes neurais. A interpretação de uma única observação demandou mais de 5 minutos, e o cálculo das interpretações para as 80 observações utilizadas nos resultados anteriores exigiu mais de 7 horas. Essa diferença substancial de tempo destaca a complexidade computacional envolvida na interpretação de modelos mais elaborados, como redes neurais.

5 Conclusão

A presente pesquisa buscou ampliar o entendimento sobre a interpretabilidade de modelos de redes neurais, explorando métodos que permitam elucidar as predições desses modelos complexos. A comparação sistemática entre um modelo de redes neurais e a tradicional regressão logística foi central para a análise, devido à natureza interpretativa já conhecida do modelo logístico.

A utilização da técnica SHAP (SHapley Additive exPlanations) para a interpretação do modelo de redes neurais revelou-se uma ferramenta poderosa, permitindo uma compreensão mais profunda das contribuições de cada variável nas predições do modelo. Este método proporcionou uma visão holística, destacando as características individuais que mais influenciaram nas decisões do modelo.

Ao confrontar os resultados preditivos da rede neural com a regressão logística, observou-se que o modelo de redes neurais apresentou resultados melhores na maioria das métricas, como Precisão, F1-score e tendo um desempenho abaixo apenas no Recall da classe "1". Esses resultados indicam que devido à arquitetura menos robusta do modelo logístico, o mesmo não conseguiu lidar com o desbalanceamento dos dados.

Essa pesquisa contribui para a discussão em torno da interpretabilidade em inteligência artificial e fornecendo resultados valiosos para a aplicação prática desses modelos em contextos onde a transparência é essencial. A busca contínua por métodos interpretativos robustos é vital para a implementação responsável e eficaz de modelos de aprendizado de máquina em diversos domínios.

Referências

AGRESTI, A. Summarizing predictive power: Classification tables. In: *An Introduction to Categorical Data Analysis*. 2. ed. [S.l.]: Wiley, 2002. cap. 5.1.6.

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

CURRY, H. B. The method of steepest descent for non-linear minimization problems. *Quart. Appl. Math.*, v. 2, n. 3, p. 258–261, 1944.

GERON, A. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. [S.l.]: O'Reilly Media, 2019. ISBN 978-1492032649.

HARDESTY, L. *Explained: Neural Networks*. 2017. ⟨http://news.mit.edu/2017/explained-neural-networks-deep-learning-0414⟩.

HART, S. Shapley value. In: EATWELL, J.; MILGATE, M.; NEWMAN, P. (Ed.). *The New Palgrave: Game Theory*. [S.l.]: Norton, 1989. p. 210–216. ISBN 978-0-333-49537-7.

HOSMER, D. W. *Applied Logistic Regression*. [S.l.]: John Wiley, 2013.

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.

LUNDBERG, S.; LEE, S.-I. A unified approach to interpreting model predictions. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2017. v. 30, p. 4765–4774.

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

SZUMILAS, M. Explaining odds ratios. *Journal of the Canadian Academy of Child and Adolescent Psychiatry*, v. 19, n. 3, p. 227–229, August 2010. ISSN 1719-8429.

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

ZELL, A. *Simulation Neuronaler Netze [Simulation of Neural Networks]*. 1st. ed. [S.l.]: Addison-Wesley, 1994. 73 p. ISBN 3-89319-554-8.

Anexo

A Repositório GitHub com os códigos utilizados

https://github.com/davialvesguerra/tcc/tree/master