



**UNIVERSIDADE FEDERAL DO MARANHÃO**  
**Programa de Pós-Graduação em Ciência da Computação**

**Lucas Gabriel Rezende de Jesus**

***Aplicação de Rede Neural Colaborativa à Classificação de  
Consumidores de Serviços Públicos***

**São Luís**  
**2021**

Lucas Gabriel Rezende de Jesus

# **Aplicação de Rede Neural Colaborativa à Classificação de Consumidores de Serviços Públicos**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Prof. Dr. Mário Antonio Meireles Teixeira

Coorientador: Prof. Dr. Areolino de Almeida Neto

São Luís - MA

2021

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).  
Diretoria Integrada de Bibliotecas/UFMA

Rezende de Jesus, Lucas Gabriel.

Aplicação de Rede Neural Colaborativa à Classificação  
de Consumidores de Serviços Públicos / Lucas Gabriel  
Rezende de Jesus. - 2021.

56 f.

Coorientador(a): Areolino de Almeida Neto.

Orientador(a): Mário Antonio Meireles Teixeira.

Dissertação (Mestrado) - Programa de Pós-graduação em  
Ciência da Computação/ccet, Universidade Federal do  
Maranhão, São Luís, 2021.

1. Aprendizagem profunda. 2. Classificação de risco.  
3. CollabNet. 4. Concessionárias de serviço público. I.  
Almeida Neto, Areolino de. II. Meireles Teixeira, Mário  
Antonio. III. Título.

Lucas Gabriel Rezende de Jesus

# **Aplicação de Rede Neural Colaborativa à Classificação de Consumidores de Serviços Públicos**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, como parte dos requisitos necessários para obtenção do grau de Mestre em Ciência da Computação.

Trabalho Aprovado. São Luís - MA, 26 de Outubro de 2021:

---

**Prof. Dr. Mário Antonio Meireles  
Teixeira**

Orientador

Universidade Federal do Maranhão

---

**Prof. Dr. Areolino de Almeida Neto**

Coorientador

Universidade Federal do Maranhão

---

**Prof. Dr. Omar Andrés Carmona  
Cortés**

Membro da Banca Examinadora

Instituto Federal do Maranhão

---

**Prof. Dr. Renato Porfirio Ishii**

Membro da Banca Examinadora

Universidade Federal de Mato Grosso do Sul

São Luís - MA

2021

Aos meus pais,  
que sobre todas as circunstâncias,  
me apoiaram na busca da minha verdade.

# Agradecimentos

Vivemos um momento delicado da história. A era pós-modernidade nunca esteve tão vigilante sobre o presente e incerta sobre o futuro. A sociedade foi obrigada a ressignificar suas relações e a saber lidar com o novo. Ser resiliente tornou-se imprescindível. Por outro lado, a sensação de confiança nas pessoas que caminham ao nosso lado e o estreitamento das relações fizeram da pandemia um período de muito aprendizado.

Portanto, primeiramente quero agradecer a Deus, que com todo seu amor e misericórdia intercedeu por mim e me guiou até o fim dessa jornada.

Aos meus pais, Marcos e Rejane, que não desistiram em nenhum momento de me apoiar. Serei eternamente grato pelos valores ensinados. Hoje, graças a eles, humildade e honestidade regem a minha vida.

Ao meu orientador, Professor Mário Meireles, pelos direcionamentos durante as adversidades encontradas e pela confiança plena em meu trabalho. Agradeço a parceria firmada durante esses anos.

Ao meu coorientador, Professor Areolino Neto, pelo admirável acompanhamento do trabalho e pelos constantes incentivos de melhoria na pesquisa. Agradeço por ter aceitado firmemente o desafio da coorientação.

Aos professores e colegas do Programa de Pós-Graduação em Ciência da Computação (PPGCC) da Universidade Federal do Maranhão (UFMA), que generosamente compartilharam seus conhecimentos e auxiliaram na idealização deste trabalho.

À UFMA, que por meio de sua estrutura me proporcionou uma vida de aprendizado e possibilitou minhas formações acadêmicas, dais quais me orgulharei a todo momento.

Por fim, agradeço a todos os familiares, amigos e tutores que de alguma forma contribuíram e auxiliaram na conclusão desta jornada.

*"É graça divina começar bem. Graça maior persistir na caminhada certa. Mas graça das graças é não desistir nunca."*

Dom Hélder Câmara

# Resumo

Nos anos 20, o estado brasileiro iniciou um processo de descentralização política e econômica que trouxe como uma das principais consequências a concessão a empresas privadas do direito de desenvolver e explorar economicamente um serviço público, responsabilidade até então única e exclusiva do Estado. Desde então, essas concessionárias buscam equilibrar a prestação do serviço público equânime e de qualidade com o processo de arrecadação inerente a toda companhia. No intuito de oferecer parâmetros que tragam mais objetividade nesse processo, neste trabalho é proposta uma metodologia para classificação do perfil de risco de inadimplência dos consumidores. É apresentado o uso da rede neural profunda colaborativa, *CollabNet*, em uma base de dados de clientes de uma concessionária de serviços públicos. A metodologia apresentou resultados promissores como acurácia de 88,1%, sensibilidade de 93,9% e valor preditivo negativo de 93,1%. Ainda é aventada a incorporação de novas características acerca dos consumidores como aspectos geográficos e renda familiar no intuito de melhorar os resultados obtidos.

**Palavras-chave:** concessionárias de serviço público; classificação de risco; CollabNet; aprendizagem profunda.

# Abstract

In the 1920s the Brazilian state instituted a political and economic decentralization process, one of the main consequences of which was a concession to private companies of the right to economically develop and explore a public service, until then was the sole and exclusive responsibility of the State. Since then, these concessionaires have searched to balance the provision of equitable and quality public service with the collection process inherent to every company. In order to offer parameters that bring more objectivity to this process, in this work we propose a methodology for classifying the default risk profile of consumers. The use of the deep collaborative neural network, CollabNet is presented in a customer database of a utility company. The methodology presents promising results such as an accuracy of 88.1 %, a sensitivity of 93.9 % and a negative predictive value of 93.1 %. It is still suggested an incorporation of new characteristics about consumers such as geographic aspects and family income in order to improve the results obtained.

**Keywords:** public service concessionaires; credit scoring; CollabNet; deep learning.

# Lista de ilustrações

Figura 1 – Processo de descoberta de conhecimento . . . . .	19
Figura 2 – Arquitetura de uma Rede Estritamente <i>feedforward</i> . . . . .	21
Figura 3 – Representação de um neurônio artificial com duas entradas. . . . .	22
Figura 4 – Exemplo de rede neural <i>MLP</i> . . . . .	24
Figura 5 – Representação da estrutura da CollabNet. . . . .	26
Figura 6 – Representação da máscara <i>D</i> . . . . .	27
Figura 7 – Metodologia. . . . .	30
Figura 8 – Exemplo de tratamento dos dados históricos. . . . .	32
Figura 9 – Exemplo de <i>label encoded</i> . . . . .	33
Figura 10 – Validação. . . . .	35
Figura 11 – Método <i>k-fold</i> com $k=10$ . . . . .	36
Figura 12 – Matriz de confusão durante o treinamento do modelo final. . . . .	44
Figura 13 – Curva ROC. . . . .	45
Figura 14 – Comportamento do erro quadrático médio nos dados de treino e validação com uma camada . . . . .	47
Figura 15 – Comportamento do erro quadrático médio nos dados de treino e validação com duas camadas ocultas . . . . .	48
Figura 16 – Comportamento do erro quadrático médio nos dados de treino e validação com três camadas ocultas . . . . .	49

# Lista de tabelas

Tabela 1 – Tabela descritiva das variáveis independentes. . . . .	33
Tabela 2 – Distribuição da base pré-processada. . . . .	35
Tabela 3 – Hiperparâmetros da CollabNet. . . . .	38
Tabela 4 – Espaço de busca de hiperparâmetros. . . . .	42
Tabela 5 – Distribuição dos dados de treino, validação e teste. . . . .	42
Tabela 6 – Resultados por inserção de cada camada . . . . .	42
Tabela 7 – Melhores Hiperparâmetros. . . . .	47
Tabela 8 – Distribuição dos dados treino e teste. . . . .	49
Tabela 9 – Tabela de Resultado da validação cruzada. . . . .	50

# Lista de abreviaturas e siglas

SNIS	Sistema Nacional de Informações sobre Saneamento.
COELBA	Companhia de Eletricidade do Estado da Bahia.
UFMA	Universidade Federal do Maranhão.
PPGCC	Programa de Pós-Graduação em Ciência da Computação.
AUC	Area Under the Curve.
KDD	Knowledge Discovery in Databases
RNA	Rede Neural Artificial.
LM	Levenberg Marquardt.
BP	Backpropagation.
DL	Deep Learning.
MLP	Multilayer Perceptron.
CNN	Convolutional Neural Network.
RNN	Recurrent Neural Network.
AE	Autoencoders.
GAN	Generative Adversarial Network.
RBM	Restricted Boltzmann Machine.
CSV	Comma-separated values.
ROC	Receiver Operating Characteristic.
EQM	Erro quadrático médio.
SMOTE	Synthetic Minority Over-sampling Technique.

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>1.1</b>	<b>Objetivos</b>	<b>15</b>
<b>1.2</b>	<b>Trabalhos Relacionados</b>	<b>15</b>
<b>1.3</b>	<b>Estrutura do Trabalho</b>	<b>17</b>
<b>2</b>	<b>FUNDAMENTAÇÃO TEÓRICA</b>	<b>18</b>
<b>2.1</b>	<b>Classificação de Risco Financeiro</b>	<b>18</b>
<b>2.2</b>	<b>Processo de Descoberta de Conhecimento</b>	<b>18</b>
<b>2.3</b>	<b>Redes Neurais</b>	<b>20</b>
2.3.1	Breve Histórico	20
2.3.2	Conceito	21
2.3.3	Treinamento	23
<b>2.4</b>	<b>Deep Learning</b>	<b>24</b>
2.4.1	Tipos de Aprendizado	25
<b>2.5</b>	<b>CollabNet</b>	<b>26</b>
2.5.1	Máscara	27
2.5.2	Fator $c$	28
2.5.3	Função de Ativação Mutante	28
<b>3</b>	<b>CLASSIFICAÇÃO DE INADIMPLÊNCIA</b>	<b>30</b>
<b>3.1</b>	<b>Extração e Seleção</b>	<b>31</b>
<b>3.2</b>	<b>Pré-processamento</b>	<b>31</b>
<b>3.3</b>	<b>Transformação</b>	<b>34</b>
3.3.1	Segmentação da base e Balanceamento de classes	34
<b>3.4</b>	<b>Treinamento</b>	<b>37</b>
3.4.1	Busca de Hiperparâmetros	37
3.4.2	Adição de Camadas	37
<b>3.5</b>	<b>Avaliação e Interpretação</b>	<b>38</b>
<b>4</b>	<b>RESULTADOS E DISCUSSÃO</b>	<b>41</b>
<b>5</b>	<b>CONCLUSÃO</b>	<b>51</b>
<b>5.1</b>	<b>Trabalhos Futuros</b>	<b>52</b>
	<b>REFERÊNCIAS</b>	<b>53</b>

# 1 Introdução

O fornecimento de serviços públicos é previsto por lei pela Constituição de 1988, sendo o Estado responsável por zelar pela prestação adequada desses serviços e garantir o acesso destes à população. É previsto em lei a concessão e permissão da prestação desses serviços. No caso da primeira, é estabelecido um acordo em que a administração pública transfere a responsabilidade da execução de um serviço de utilidade pública para uma empresa ou um consórcio de empresas.

Dessa forma, a concessão permite uma descentralização da administração de serviços essenciais à população, o que reduz o aparelhamento do estado e podendo levar a uma melhor prestação desses serviços (GROTTI, 2007). Nesse contexto, além de prestar um serviço de qualidade e de forma equânime, as empresas necessitam fundamentalmente da geração de lucros e da conquista de alguns importantes indicadores como os índices de satisfação. Estes inferem-se diretamente na administração eficiente e efetiva da organização, demonstrando ótimos indicativos para possíveis investimentos.

No entanto, algumas peculiaridades no fornecimento de serviços essenciais podem prejudicar o processo de arrecadação dessas empresas. No setor elétrico por exemplo, (ARAÚJO, 2016) cita a regulação incentivada e o próprio fato de ser um serviço universalizado. O primeiro aspecto refere-se ao fato dessas empresas operarem sobre metas estabelecidas por um regime de regulação tarifária. Já o segundo aspecto denota a obrigatoriedade do consórcio em fornecer o serviço a todos os consumidores, sem distinções de qualquer natureza.

Os fatores citados implicam que durante o planejamento estratégico dessas empresas é comum considerar um alto risco de inadimplência. Dados do Sistema Nacional de Informações sobre Saneamento (SNIS) informam que em 2018, das 26 prestadoras de serviço analisadas, somente metade delas terminaram o ano em *superávit*, ou seja, receita operacional maior que a despesa total com os serviços. Essas taxas chegaram a 25% em 2020, durante o período da pandemia.

Grande parte das concessionárias brasileiras sofrem com a problemática da inadimplência e destinam uma parcela considerável do seu capital no processo de recuperação desses clientes (RIBEIRO, 2015). Essas estratégias resumem-se aos métodos de negociação, onde são oferecidas condições de pagamento especiais para clientes já em condição de inadimplência. A outra parcela são estratégias para impedir que propensos consumidores à inadimplência entrem em condição de devedores e gerem *déficit* na arrecadação da empresa. Comumente são aplicadas ferramentas de cobrança que obedecem critérios estabelecidos pelos setores de cobrança e arrecadação das organizações.

Uma grande lacuna nessas estratégias é o fato de que grande parte das análises realizadas por esses setores são manuais. Nesse sentido, abordagens computacionais podem ser aplicadas com o intuito de fornecer parâmetros para avaliação dos clientes de forma contínua e automatizada, ao reduzir o esforço gasto em uma tarefa muitas vezes onerosa, proporcionando um processo de abordagem aos clientes mais justo e eficaz.

Tal cenário é possível pois essas empresas, em sua grande maioria, possuem informações de consumo e pagamento dos clientes e podem utilizá-las em algoritmos de aprendizado de máquina. Notavelmente, por se tratarem de serviços essenciais destinados à toda população, esses dados são críticos e apresentam extremo valor. Portanto, as concessionárias são guardiãs dessas informações, ao zelar pela sua segurança e sigilo.

Por conseguinte, Lemos, Steiner e Nievola (2005) inauguram o tema de classificação de inadimplência no Brasil por meio da aplicação de redes neurais e árvores de decisão. Os autores comprovam que, de posse de dados históricos de 339 pessoas jurídicas de um banco, é concebível chegar a resultados satisfatórios na determinação de provável inadimplência daquele cliente. Pode-se explicar boa parte da variação do comportamento do cliente por meio de suas informações históricas na relação com a concessionária, tornando realizável um mapeamento entre os dados desses clientes e sua propensão à inadimplência.

Nessa perspectiva, os algoritmos de aprendizado de máquina costumam ser mais eficientes. Esses algoritmos fornecem melhores parâmetros para o ajuste entre viés e variação, dentre eles, as redes neurais profundas vêm ganhando destaque na comunidade acadêmica. Diversas arquiteturas dessas redes estão sendo exploradas na resolução de problemas complexos como reconhecimento de padrões e processamento de linguagem natural (GOODFELLOW et al., 2014).

Nesse espectro, Freitas Jr. et al. (2018) aborda um problema decorrente da inicialização estocástica das arquiteturas de redes neurais profundas. O autor revela a importância de tratar a inserção de novas camadas em redes neurais. Por meio do método proposto, *CollabNet*, pode-se controlar a natural perturbação do erro no processo de aprendizado supervisionado de redes *feed-forward*, comumente causada pela aleatoriedade do processo de inicialização.

Tendo em vista os bons resultados dessa recente técnica, o presente trabalho propõe validá-lo em um caso de uso no problema de classificação de risco de inadimplência, propondo uma metodologia no contexto de concessionárias de serviços públicos. Por fim, ainda é propósito encontrar parâmetros que auxiliem a análise do cumprimento do contrato do serviço provido pela concessionária.

## 1.1 Objetivos

Diante do cenário apresentado, o objetivo principal deste trabalho é propor uma metodologia para classificação de risco de inadimplência no contexto de concessionárias de serviços públicos, por meio da aplicação da *CollabNet* (FREITAS JR. et al., 2018). Pretende-se assim auxiliar as concessionárias no processo de análise de inadimplência e tomadas de decisão por meio de ações de cobrança.

Além disso, este trabalho busca os seguintes objetivos específicos:

- Estudar os principais fatores que influenciam a inadimplência em concessionárias de serviços públicos.
- Investigar a utilização da *CollabNet* na construção de um método para a classificação do problema exposto nesta metodologia.
- Contribuir com o a literatura sobre uso de métodos computacionais para análise de risco, trazendo um caso de uso voltado para o contexto de inadimplência em concessionárias de serviço público.
- Compreender os fatores decorrentes da inserção colaborativa de camadas ocultas em uma rede neural profunda.

## 1.2 Trabalhos Relacionados

Grande parte da literatura aborda o problema da inadimplência pelo viés do risco da concessão de crédito. Araújo (2016) apresentou uma vasta revisão bibliográfica acerca de métodos estatísticos e de otimização na solução de problemas envolvendo inadimplência e caracterização de perfis de consumidores. A autora justificou a seleção de 12 variáveis relacionadas ao atraso de pagamento e sua reincidência de consumidores da Companhia de Eletricidade do Estado da Bahia (COELBA). Em seu estudo, foi aplicado o teste *Kolmogorov-Smirnov* (KS) aos modelos de regressão logística e de redes neurais desenvolvidos onde foi obtido valores KS de 52% e 69% respectivamente.

No estudo realizado por Lopes (2017) foi verificado que aspectos como perfil de renda e época do ano influenciam no comportamento de pagamento do cliente. O estudo foi realizado em cima de uma base de dados de um banco brasileiro com 12 variáveis numéricas e 16 categóricas. Foram aplicados 3 modelos de aprendizado de máquina sobre os dados onde destacaram-se as técnicas de *deep learning* e *gradient boosting*, este ultimo com valor de recall 92,86%.

Em Lessmann et al. (2015) é apresentado uma pesquisa acerca de algoritmos de classificação utilizados em problemas de *credit scoring* e análise de inadimplência. O estudo

comparou 41 classificadores em termos de seis medidas de desempenho, agrupados em 3 categorias: classificadores individuais, multiclassificadores homogêneos e multiclassificadores heterogêneos.

Seguindo a mesma linha de pesquisa, Forti (2018) realizou uma comparação de técnicas de aprendizado de máquina no cenário de recuperação de crédito no mercado brasileiro. O estudo apontou o *gradient boosting* como promissor em processos de predição para *collection score*. O trabalho, porém, não abordou técnicas de aprendizado profundo como uma das alternativas.

Uma combinação do método *K-Means* com uma medida de sensibilidade estocástica e uma rede neural com função de base radial é apresentada em Chen, Zhang e Ng (2018). O objetivo da aplicação dos métodos computacionais é melhorar a previsão de inadimplência de empréstimos. O trabalho utiliza um conjunto de dados de 25504 instâncias de uma plataforma financeira.

Uma abordagem com redes neurais profundas para classificação de clientes em propensos a pagamento ou não foi apresentada em Chishti e Awan (2019). Os autores utilizaram dados do ano de 2005 de empresas de cartões de crédito orientais. A base contava com 30.000 instâncias e com 29 variáveis, sendo uma dependente e 28 independentes. Em uma rede *multilayer perceptron* com 3 camadas ocultas e 2465 pesos, os autores obtiveram uma acurácia de 82%, especificidade de 67% e sensibilidade de 95%.

No mesmo âmbito financeiro, redes *autoencoder* demonstraram-se bem eficientes na detecção de fraudes de cartão crédito, como apresentado em Zamini e Montazer (2018). Na abordagem, a rede é responsável por realizar a função de mapeamento para redução de dimensionalidade das entradas e por fim é aplicado o algoritmo *K-means* para realizar o agrupamento das transações fraudulentas e não fraudulentas. O modelo atingiu métricas de 0,989 de acurácia e 0,961 de *Area Under Curve* (AUC). De forma análoga, Chen et al. (2018) aplica *autoencoder* convolucional para detecção de anomalia.

A detecção de características latentes por meio de *autoencoders* é apresentada em Zhuang et al. (2015) onde é proposta uma estrutura de aprendizado de representação de recursos, combinando os auto-codificadores em uma abordagem multitarefa afim de aumentar o poder de generalização dos modelos. A abordagem permitiu o compartilhamento de todos os pesos de codificação e decodificação juntamente com os parâmetro do modelo.

Por fim, Hassan e Abraham (2013) chamam atenção ao usar um *Ensemble* de redes neurais. Os autores aplicaram filtros de redução em um com 1000 casos e 24 atributos, resultando em grupos com 24, 20 e 9 atributos. Nestes são aplicadas 3 redes neurais treinadas com diferentes algoritmos de treinamento, gradiente conjugado escalonado, algoritmo de *Levenberg-Marquardt* e *Backpropagation* secante em uma etapa. Ao final as 3

redes são combinadas e o resultado final é a classe dominante e o método alcançou uma acurácia de 96% na base de dados com 20 atributos.

## 1.3 Estrutura do Trabalho

Além da Introdução, este trabalho está organizado em 6 capítulos apresentados a seguir.

O Capítulo 2 apresenta a fundamentação teórica, na qual estão presentes os principais conceitos inerentes à área de *deep learning* e ao processo de descoberta de conhecimento, além de um aprofundamento acerca da principal técnica utilizada neste trabalho, a *CollabNet*.

O Capítulo 3 descreve detalhadamente o processo de desenvolvimento da metodologia de classificação de inadimplência. As etapas da metodologia e como cada técnica se encaixa em cada uma delas são discriminadas neste capítulo.

Em seguida, no Capítulo 4, são apresentados os resultados da aplicação da metodologia proposta para classificação de risco de inadimplência. Por meio da análise numérica e gráfica dos resultados, é realizada uma contextualização sobre a aplicabilidade do método na resolução dos problemas descritos.

Por último, com base em toda essa discussão, no Capítulo 6 são apresentadas as considerações finais sobre o trabalho e as propostas de trabalhos futuros, com foco nos pontos em que a metodologia apresentada é aplicável no contexto real.

## 2 Fundamentação Teórica

### 2.1 Classificação de Risco Financeiro

Este trabalho é motivado pela área de classificação de risco financeiro, frequentemente conhecido na literatura como *credit scoring*. A origem desse conceito deu-se com Fisher (1936) que apresentou a ideia de discriminar diferentes grupos entre uma população por meio de características que se relacionassem com cada grupo. Posteriormente, Durand (1941) aplicou estas técnicas em contextos financeiros e provou que podem ser utilizadas para discriminar bons e maus empréstimos. Ao mesmo tempo, era um momento que as grandes empresas do mercado financeiro tinham dificuldade em gerenciar seu risco de crédito. Com o crescimento na acessibilidade dos recursos computacionais somado ao cenário de *Big Data*, os métodos de análise manuais deram espaço para técnicas estatísticas e de inteligência computacional (TRIVEDI, 2020).

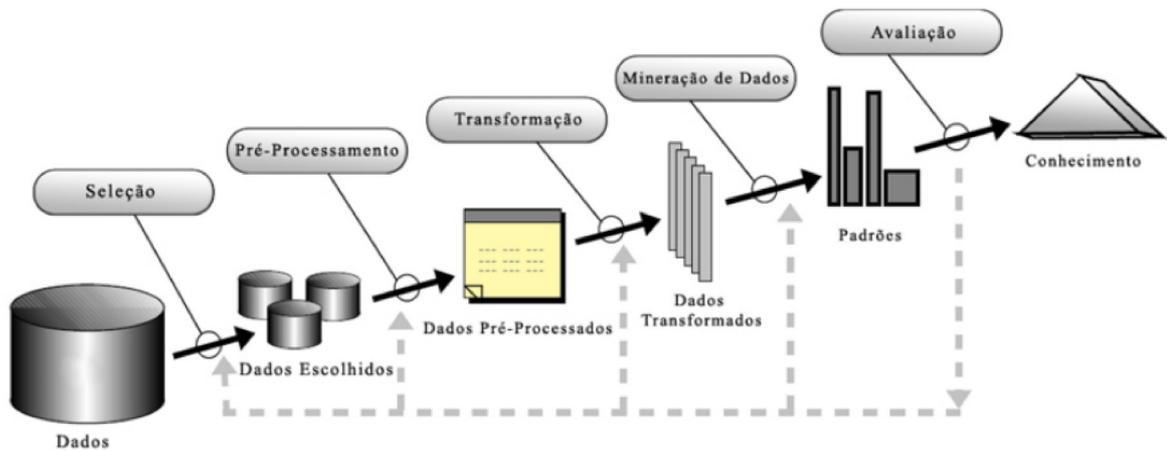
Dessa forma, o *credit scoring* pode ser interpretado como um problema de classificação binária que, dadas as informações do comportamento do usuário coletadas dentro de um determinado período de tempo, atribui-se a este cliente um rótulo de bom ou mau pagador (THOMAS, 2000). Essas informações habitualmente são constituídas de histórico de pagamento, formulários de avaliação e cadastro. Em modelos computacionais, essas características dos clientes são aplicadas como dados de entrada para o modelo e, então, geradas saídas que resultam na classificação entre bom e mau pagador.

### 2.2 Processo de Descoberta de Conhecimento

O nome refere-se a um conjunto de processos que busca gerar conhecimento confiável, compreensível e potencialmente útil. Portanto, o *Knowledge Discovery in Databases (KDD)* almeja mapear um conjunto de dados de baixo nível em formatos mais compactos e úteis, como um relatório ou um modelo preditivo (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996). Os padrões extraídos devem permitir o uso do conhecimento de forma a tirar proveito de alguma vantagem, seja científica ou comercial, por meio da otimização de processos e reduções de custos (DANTAS et al., 2008).

Fayyad et al. (1996) apontam o processo como interativo e iterativo. Interativo pois o usuário pode controlar o curso das atividades realizando intervenções e, iterativo, por tratar-se de sucessão finita de operações onde o resultado de cada uma é dependente dos resultados das etapas que a precedem. A Figura 1 mostra uma visão geral de cada fase do processo.

Figura 1 – Processo de descoberta de conhecimento



Fonte: (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996).

- Seleção: Etapa que engloba o aprendizado do domínio em questão. São identificadas quais informações presentes na base de dados devem ser consideradas no restante do processo. Escolher um conjunto de características ou atributos condizentes para análise é um trabalho que deve ser feito com apoio de um especialista da área em questão. Esse estágio ainda pode ser bastante complexo, pois os dados podem ser oriundos de diversas fontes diferentes como *Data Warehouse* e Planilhas (PRASS, 2016);
- Pré-processamento: Esta etapa corrige as inconsistências encontradas nos dados para garantir a confiabilidade dos resultados da mineração (COSTA et al., 2016). Algumas tarefas como eliminação de dados redundantes, recuperação de dados incompletos e execução de estratégias para campos faltantes podem estar presentes nesta etapa (PRASS, 2016). Outro exemplo, como será apresentando neste trabalho, é a preparação dos registros para sequências temporais;
- Transformação: Fase em que são aplicadas técnicas para codificar os dados de forma que eles possam ser utilizados eficientemente na etapa de mineração. Alguns exemplos de métodos que podem ser aplicados são: a normalização, a padronização e as técnicas de redução de dimensionalidade;
- Mineração de Dados (*Data Mining*): Estágio principal do processo de descoberta de conhecimento, sendo o responsável pela produção de informação por meio de inferências matemáticas ou pela simples análise dos dados. Nesta etapa, são selecionadas e aplicadas técnicas de aprendizagem computacional bem como realizada a otimização de hiper-parâmetros. Alguns exemplos de técnicas utilizadas: redes neurais, algoritmos genéticos, modelos estatísticos e probabilísticos (GOLDSCHMIDT; PASSOS, 2005);

- Avaliação: Etapa final em que o conhecimento adquirido do processo deve ser interpretado e avaliado, por meio de métricas que façam sentido para o objetivo final, de forma a gerar um conhecimento útil e confiável. Neste ponto, é comum que o ciclo retorne a algum dos estágios anteriores até que se alcance um resultado satisfatório (PRASS, 2016).

A metodologia proposta neste trabalho é baseada no *KDD*. Cada etapa da metodologia é acomodada por uma ou mais etapas do processo de descoberta do conhecimento, de modo a obedecer a sequência original discutida acima. Mais especificamente, é proposta na etapa de mineração a utilização de uma técnica baseada em *Deep Learning*.

Redes neurais de aprendizado profundo são usualmente exploradas em tarefas de previsão de inadimplência e classificação de risco de inadimplência como apresentado, respectivamente, em Guimarães (2006) e Guilhon et al. (2020). Ainda estimulado pela recente técnica de inserção de novas camadas em redes neurais *Deep Learning* proposta por Freitas Jr. et al. (2018), este trabalho faz a aplicação dessa técnica no contexto da classificação de inadimplência em concessionárias de serviços públicos.

## 2.3 Redes Neurais

### 2.3.1 Breve Histórico

O conceito de rede neural como um modelo computacional foi primordialmente introduzido por McCulloch e Pitts (1943). Na ocasião, os autores, por meio de circuitos elétricos e uso de portas lógicas, descreveram a utilização do que seria um neurônio artificial. Este trabalho abriu portas para diversos estudos tanto pelo viés biológico quanto matemático. No que diz respeito a este último, Rosenblatt (1957) apresenta o *Perceptron*, a forma mais simples de rede neural. Por meio de operações matemáticas, Rosenblatt demonstrou que há possibilidade de classificar padrões considerados linearmente separáveis por meio de um neurônio artificial com pesos e polarização sinápticas ajustáveis. Nesse mesmo artigo, o autor já sinaliza o uso de neurônios encadeados que pudessem formar mais de uma camada.

O método de otimização utilizando decida do gradiente foi proposto na década de 1970. Porém, foi em Rumelhart, Hinton e Williams (1986) que o método ganhou utilidade no processo de aprendizado das redes neurais. O posteriormente denominado *backpropagation* despertou o interesse de diversos cientistas pois a técnica permitiu que o cálculo dos gradientes percorresse várias camadas de uma rede neural, por meio de um fluxo inverso da composição da rede. Porém, em pouco tempo de pesquisa, foi constatado que a limitante desse algoritmo era a capacidade computacional da maioria dos computadores da época.

O processo de treinamento por *backpropagation* de algumas redes neurais multicamadas podiam demorar até dias, sendo, na maioria dos cenários, de uso impraticável.

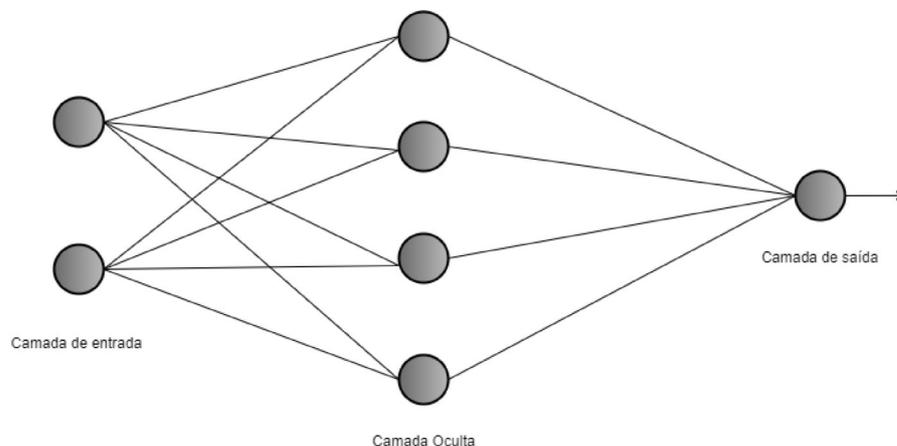
Após um grande período de hiato, Geoffrey Hinton e Ruslan Salakhutdinov publicaram um trabalho denominado *Reducing the Dimensionality of Data with Neural Networks*. Nele, é abordado a possibilidade de utilizar uma rede de *autoencoders* para a redução da dimensionalidade de dados de entrada, bem como o treino individualizado de cada camada em uma rede profunda (HINTON; SALAKHUTDINOV, 2006). O trabalho é considerado o grande responsável pela popularização do termo *deep learning* e pelo grande despertar de interesse da comunidade acadêmica pela área.

Desde então, a aprendizagem profunda ganhou destaque tanto no mundo científico quanto na própria indústria. Corporações como Google (*Google Brain*), Facebook (*Facebook AI*), Microsoft (*Microsoft Research*) possuem subdivisões dedicadas aos estudos e aplicações de inteligência artificial por meio de *Deep Learning*. Hoje, essas soluções estão cada vez mais presentes não só na construção de sistemas e aplicações inteligentes, mas também no processo de tomada de decisão das empresas.

### 2.3.2 Conceito

Uma rede neural artificial (RNA) padrão consiste na composição de muitos processadores simples e conectados chamados neurônios, cada um produzindo uma sequência de ativações com valor real (SCHMIDHUBER, 2015). Sobre uma visão algorítmica é possível entender o processo de aprendizado da rede e como adquirir uma relação entrada-saída por meio dos dados (ALMEIDA NETO, 2003). Portanto, produzir esse mapeamento para diferentes dados e pelo mesmo processo e estrutura é o objetivo da rede neural.

Figura 2 – Arquitetura de uma Rede Estritamente *feedforward*.



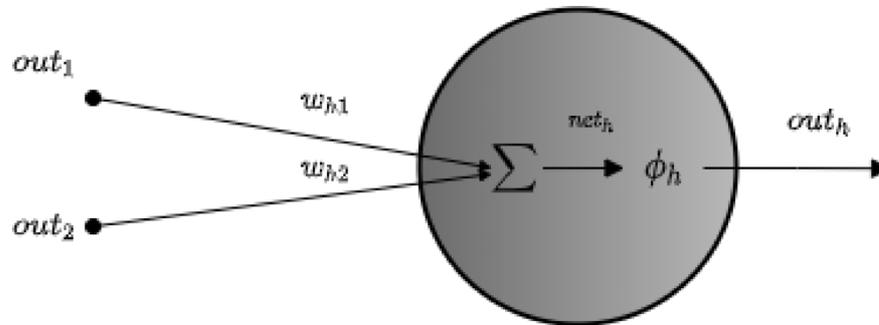
Fonte: Autor

Na Figura 2 é possível visualizar o formato básico de uma rede neural artificial estritamente *feedforward*. A camada de entrada representada é composta por unidades de armazenamento de dados e não realiza nenhum processamento dentro do processo de aprendizagem. Por esse fato, alguns autores não a consideram efetivamente como uma camada. Na camada oculta é onde ocorre o primeiro processamento composto por operações matemáticas, em que as suas saídas serão insumos para a camada consequente. A camada de saída reúne as entradas e determina o valor de saída da rede neural.

A passagem de valores entre as camadas se dá por meio das conexões. Estas possuem um peso associado, o que biologicamente pode ser interpretado como a intensidade da conexão entre os neurônios por meio do axônio. Matematicamente esse conjunto representa uma composição, geralmente uma soma, ponderada dos valores de saída da camada imediatamente anterior, formando a chamada entrada líquida do neurônio.

Este valor é submetido a um função de ativação. Este operador é normalmente responsável pela não-linearidade desse tipo de arquitetura, portanto sua escolha é uma das tarefas crucias ao projetar redes neurais. Por fim, por se tratar de uma rede neural simples, o neurônio de saída apresenta o valor como resultado final, caso contrário a rede de conexões se estenderia até a última camada.

Figura 3 – Representação de um neurônio artificial com duas entradas.



Fonte: Autor

Tendo como base o cenário da Figura 3, a entrada líquida de um neurônio,  $net_h$ , pode ser dada pela Equação 2.1, onde  $out_i$  corresponde a saída do neurônio  $i$  da camada anterior conectado ao neurônio  $h$  e  $w_{hi}$ , ao peso entre os neurônios  $h$  e  $i$ .

$$net_h = \sum_{i=1}^n out_i \cdot w_{hi} \quad (2.1)$$

Dessa forma, os neurônios das camadas intermediárias e de saída processam esse valor via uma função de ativação definida pela Equação 2.2, para assim produzir a saída.

$$out_h = \phi^h(net_h) \quad (2.2)$$

Essa etapa é conhecida como *forward* e permite, por meio da composição de operadores, o mapeamento dos dados de entrada em uma ou várias saídas. Porém, o aprendizado das redes neurais está totalmente voltado as quais valores dos pesos das conexões entre os neurônios irão obter ao final do processo de treino. Desse modo a saída dessa etapa deve ser utilizada em um cálculo de erro em relação à saída desejada. Esse cálculo é utilizado como parâmetro dentro do processo de treinamento da rede. Iterativamente a rede adapta seus pesos em direção a minimizar o erro. Nesse processo, o algoritmo *backpropagation* é o mais amplamente utilizado.

### 2.3.3 Treinamento

Como abordado anteriormente, o treinamento de uma rede neural ocorre por meio da otimização dos seus pesos em busca do mapeamento correto entre as entradas para as saídas. Nesta etapa, podem ser utilizados alguns algoritmos, dentre os quais se destacam: *Levenberg-Marquardt* (LM) (LEVENBERG, 1944), filtro de Kalman estendido (FKE) (MCGEE; SCHMIDT; SMITH, 1962) e o amplamente utilizado *backpropagation* (BP) (RUMELHART; HINTON; WILLIAMS, 1986).

Em conformidade com grande parte da literatura, neste trabalho é aplicado o *backpropagation* na etapa de treinamento da rede neural. O BP propõe-se a realizar a otimização dos pesos da rede por meio do gradiente descendente, o que leva a condição de que as funções de ativação precisam ser contínuas e diferenciáveis. O BP é classificado como um aprendizado de correção de erro (HAYKIN; NETWORK, 2004).

Frequentemente o erro quadrático médio é utilizado para fornecer no cálculo da variação que os pesos devem sofrer. O fator preponderante para sua popularização do BP é a possibilidade dos ajustes de pesos serem realizados em várias camadas ocultas. Esse processo ocorre de forma inversa a fase *forward*. Portanto, o cálculo do gradiente começa pela camada de saída, estendendo-se até a primeira camada oculta. A Equação 2.3 apresenta o cálculo proposto pelo algoritmo para a variação dos pesos entre as camadas de saída e escondida:

$$\Delta W_{ij} = \eta \cdot e_i \cdot f'_i \cdot out_j \quad (2.3)$$

onde  $\Delta W_{ij}$  representa a variação dos pesos entre o neurônio  $i$  da camada de saída e o neurônio  $j$  da camada escondida,  $\eta$  é a taxa de aprendizado,  $e_i$  é o erro de saída do neurônio  $i$ , ou seja, o valor da saída do neurônio menos o valor desejado,  $f'_i$  a derivada da função de ativação do neurônio  $i$  em relação a entrada  $net$  e  $out$  o valor de saída do neurônio  $j$  da camada oculta.

A taxa de aprendizado é um dos hiperparâmetros escolhidos de forma empírica, sendo uma situação de compensação (*tradeo-ff*). Um alto valor proporciona um aprendizado

mais rápido, porém pode levar a não diminuição do erro de saída e à instabilidade da rede. Por outro lado, um valor baixo tende a diminuir o erro de forma mais estável porém incrementa o tempo do aprendizado.

A Equação 2.4 apresenta o cálculo do erro retro propagado para a camada oculta:

$$e_j = \sum_{i=1}^n e_i \cdot f'_i \cdot w_{ij} \quad (2.4)$$

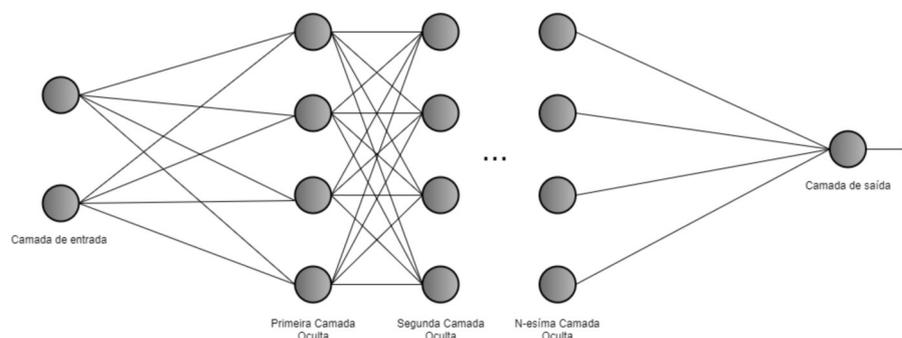
onde  $e_i$  representa o erro de saída do neurônio  $i$ ,  $w_{ij}$  o peso entre o neurônio  $i$  da camada de saída e o neurônio  $j$  da camada oculta, e finalmente  $f'_i$  a derivada da função de ativação do neurônio  $i$ .

Além da taxa de aprendizado, outro fator que deve ser observado ao se projetar as redes neurais são os critérios de parada. Eles determinam em que momento a rede atingiu um desempenho satisfatório e pode encerrar o processo de aprendizado. Algumas critérios mais utilizados são as quantidades de ciclos e alcançar um limiar de erro (HAYKIN; NETWORK, 2004).

## 2.4 Deep Learning

*Deep Learning* (DL) é uma classe de algoritmos amplamente desenvolvidos a partir de 2006 que em geral possui como base as redes neurais artificiais (ALOM et al., 2019). Uma importante peculiaridade do aprendizado profundo é a possibilidade de construção de estruturas complexas pelo uso de inúmeras camadas. Estas são formadas por unidades de processamento de informação chamadas de neurônios, fazendo alusão ao cérebro humano. O sequenciamento dessas estruturas permite a geração de representações de dados em níveis cada vez maiores, formando estruturas hierárquicas e conectadas. Estas podem ser exploradas para aprendizado de recursos e classificação de padrões, por exemplo (LAROCHELLE et al., 2009).

Figura 4 – Exemplo de rede neural *MLP*.



Fonte: Autor

A Figura 4 é a representação de uma *multilayer perceptron* (MLP). Os aspectos de uma rede *Deep Learning* como números de neurônios, quantidade de camadas e até métodos de treinamento abrem espaço para o desenvolvimento contínuo de novas arquiteturas.

### 2.4.1 Tipos de Aprendizado

Segundo Alom et al. (2019), as abordagens de aprendizado podem ser categorizadas em: supervisionada, semi-supervisionada e sem supervisão. Por se tratar de aprendizagem de máquina, os algoritmos de *Deep Learning* seguem a mesma classificação.

#### **Aprendizado supervisionado**

Na aprendizagem supervisionada, o algoritmo recebe um conjunto de dados rotulados e partir dele computa uma função. Nas abordagens *deep learning* a chamada etapa de treinamento utiliza os dados rotulados como entrada para estimar os melhores parâmetros da rede. Dentro do processo, uma função de perda é responsável por fornecer os valores de espaçamento da saída da rede com a saída correta previamente rotulada. Combinado com um método de otimização, as redes neurais permitem a aproximação de qualquer função contínua (CYBENKO, 1989). Existem diferentes arquiteturas de aprendizagem supervisionado para *deep learning* tais como *stacked autoencoder*, redes neurais convolucionais (CNN) e redes neurais recorrentes (RNN's).

#### **Aprendizado não supervisionado**

Em ambientes com dados não rotulados, as redes de aprendizagem profunda podem exercer o papel de agentes discriminadores de características fornecendo representações latentes sobre os dados de origem. Outra importante função é o agrupamento dos dados não rotulados por meio da similaridades entre eles. Consequentemente existem diversas técnicas aplicadas às redes neurais nesse tipo de abordagem, como a introdução de uma função de perda de reconstrução. Abordagens como *autoencoders* (AE), máquinas restritas de Boltzmann (RBM) e as recentes desenvolvidas redes adversárias generativas (GAN's) são bastante utilizadas em tarefas de redução de dimensionalidade e agrupamento.

#### **Aprendizado semi-supervisionado**

A aprendizagem semi-supervisionada é inserida em uma variedade de problemas em que os dados rotulados são escassos ou o processo de categorização é custoso. Este tipo de aprendizagem utiliza algoritmos capazes de aprender a partir de dados rotulados e não rotulados. Deste modo, em um problema de classificação, por exemplo, uma parcela dos dados não rotulados são submetidos a uma estrutura, que por sua vez estima seus rótulos que são utilizados posteriormente no processo de treinamento supervisionado. Quando utilizados em tarefas de agrupamento, os dados rotulados são utilizados na formação de

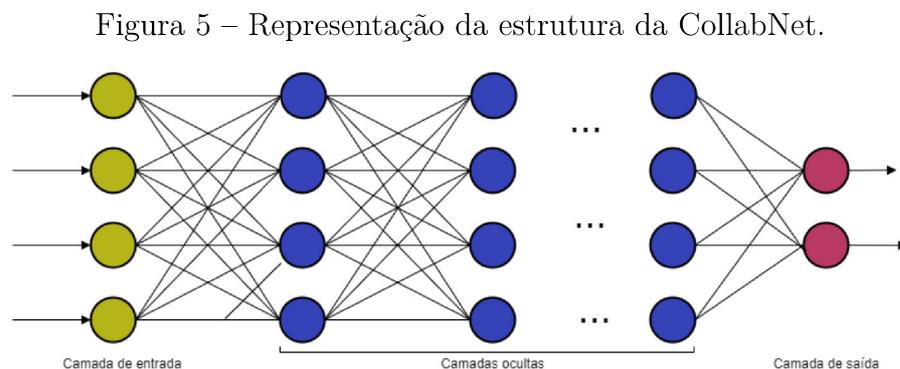
grupos. Em alguns casos, abordagens de *deep reinforcement learning* são utilizadas de forma semi-supervisionada.

## 2.5 CollabNet

Como visto, redes *deep learning* são caracterizadas principalmente pela presença de mais de uma camada oculta. Sabe-se também que a topologia da rede sempre foi um dos principais pontos para o sucesso desses algoritmos (BRAGA, 2000). Contudo, nesse processo de inserção de novas camadas ocultas, grande parte das arquiteturas utilizam métodos estocásticos na inicialização dos pesos dos neurônios dessa camada, o que acarreta em uma perturbação no erro de treinamento.

Dessa forma, visando ao processo de inserção de novas camadas em redes escondidas e baseadas nos conceitos de *AutoEncoders* e redes *Deep Stacked*, Freitas Jr. et al. (2018) apresentou uma proposta de uma rede batizada pelos autores de *CollabNet*. A grande inovação dessa técnica refere-se às estratégias adotadas que permitem inserir uma nova camada na rede de modo que não haja a natural perturbação no erro de saída de forma que o valor convirja de forma mais suave e estável a um ponto ótimo.

A estrutura da *CollabNet* consiste no empilhamento de camadas escondidas a uma rede pré-treinada. Essa nova camada deve possuir o mesmo número de neurônios da camada anterior bem como preservar o aprendizado obtido por ela. Por consequência, a camada anterior é treinada previamente e só posteriormente adicionada uma nova camada à estrutura. Após a inserção, técnicas são utilizadas durante o aprendizado para gradativamente o aprendizado da nova camada ser incorporado à rede, atenuando ou até anulando o impactado negativo dessa operação. O resultado é uma estabilidade no erro de saída. A figura abaixo apresenta a disposição das camadas e neurônios na *CollabNet*.



Fonte: Autor

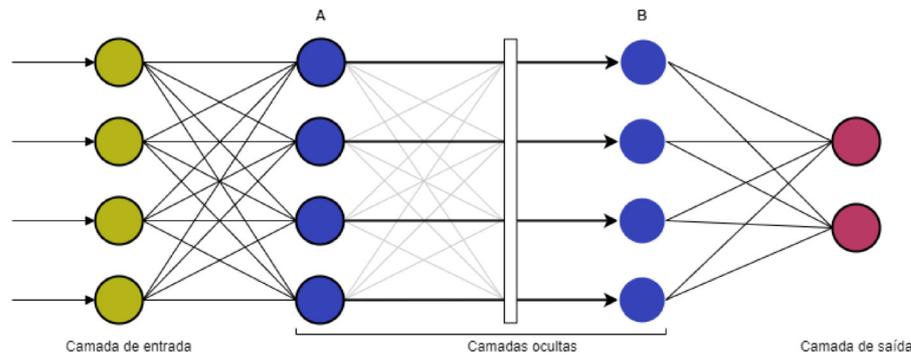
Nota-se que na estrutura as camadas devem obedecer os números de neurônios estabelecidos, com exceção da camada de entrada e de saída. Esse princípio é primordial para o funcionamento da técnica, especialmente para o treinamento.

### 2.5.1 Máscara

Como já citado, a inserção da nova camada na *Collabnet* é mediada por alguns procedimentos que visam à estabilizar o erro da rede durante a fase de treinamento. Um parâmetro comum para nortear a inserção de uma nova camada é a pausa no decaimento do erro da rede. Ao alcançar-se esse estado é necessário a inclusão de uma nova camada para aumentar a complexidade da rede.

Na proposta, a inserção da nova camada permite com que a saída de cada neurônio dessa camada seja igual ao neurônio correspondente da camada anterior, refletindo naturalmente na não perturbação no erro de saída. Esse fato é dado por meio da presença de uma máscara  $D$ , inclusa entre a última camada oculta treinada e a recém camada oculta inserida, como mostra a Figura 6

Figura 6 – Representação da máscara  $D$ .



Fonte: Autor

Durante o treinamento, os pesos entre a recém treinada camada (A) e a nova camada (B) são filtrados pela máscara. Os valores de saída dos neurônios correspondentes, representados pela seta mais grossa, são preservados enquanto os outros pesos recentemente inicializados são atenuados ou até retidos pela máscara. Os pesos entre a camada de saída e a penúltima camada oculta são preservados e tornam-se os pesos entre a nova camada e camada de saída, que serão alterados durante o treinamento.

Matematicamente, esse elemento é representado por uma matriz quadrada de ordem  $N$ , onde  $N$  é o número de neurônios em cada camada oculta. Desse modo, é válido afirmar que a diagonal principal dessa matriz deve representar a conexão entre os neurônios correspondentes entre as camadas e os outros valores as conexões adjacentes.

No treinamento, para que o produto do peso entre os neurônios e o valor de saída seja anulado, é necessário que na diagonal principal da máscara, o valor seja o inverso do peso. Fora dessa diagonal, para que o aprendizado seja incorporado pela rede, o valor deve ser um fator que corresponda ao quanto de influência dos outros pesos deseja-se incorporar no treinamento, um  $\Delta D$ .

Em uma rede neural com dois neurônios em cada camada oculta, pode-se considerar a máscara  $D$  no formato  $\begin{pmatrix} 1/W_{hi} & \Delta D \\ \Delta D & 1/W_{hi} \end{pmatrix}$ .  $W_{hi}$  são os pesos entre a nova camada oculta e a anterior a ela e  $\Delta D$ , o fator de influência dos outros pesos na nova camada, variando entre zero e um.

Desta maneira, a operação de entrada da nova camada  $net_h$  pode ser dada pela Equação 2.5, onde  $Y$  é a saída da camada anterior e o operador  $.*$  representa uma multiplicação elemento por elemento.

$$net_h = (W_{hi} .* D) * Y \quad (2.5)$$

O intuito é que, ao final do treinamento, toda matriz seja composta por elementos unitários e assim os pesos entre ambas as camadas estejam estabilizados, a ponto de suas variações não influenciarem negativamente no erro de saída da rede. Os valores iniciais do  $\Delta D$  são definidos manualmente no planejamento de arquitetura da rede e serão alterados gradativamente conforme outros hiperparâmetros.

## 2.5.2 Fator $c$

A ideia por trás do fator  $c$  é permitir, mesmo com a inclusão da máscara e alteração na função de ativação dos neurônios (será apresentada em seguida), que o aprendizado da rede ainda seja obtido por meio da variação dos pesos de forma gradual e sucessiva.

Além do próprio valor  $c$ , a *Collabnet* ainda possui duas outras variáveis de controle, o  $\Delta c$  e o salto  $c$ . Este último refere-se a que passo das épocas pretende-se incrementar o valor inicial de  $c$  e o primeiro o quanto deste valor será incrementado. A escolha desses hiperparâmetros é fundamental para o sucesso da técnica, pois quanto maior o valor de  $c$ , maior será a influência da nova camada no aprendizado, o que impacta diretamente no erro de saída da rede. Logo, direcionar o aprendizado da rede, tendo em vista a relação entre esses parâmetros e o aprendizado da rede, é a principal motivação na orientação dos experimentos apresentados neste trabalho.

## 2.5.3 Função de Ativação Mutante

Na *Collabnet*, a função de ativação também deve receber um tratamento. A ideia proposta pelo autor é que ao início do treinamento a função de ativação seja a função identidade, e com o passar do tempo, esta convirja para a função sigmoide. Portanto, a partir da inclusão da segunda camada oculta, a função de ativação dos neurônios das camadas ocultas pode ser dada pela Equação 2.6:

$$\phi_A(net_h) = \frac{1}{1 + e^{(-net_h)}} * c + net_h(1 - c) \quad (2.6)$$

onde  $\phi_A(net_h)$  é a saída do neurônio com a sigmoide  $A$ ,  $net_h$  é a soma ponderada de todas as entradas sinápticas dos neurônios e  $c$  representa o fator de ponderação já apresentado.

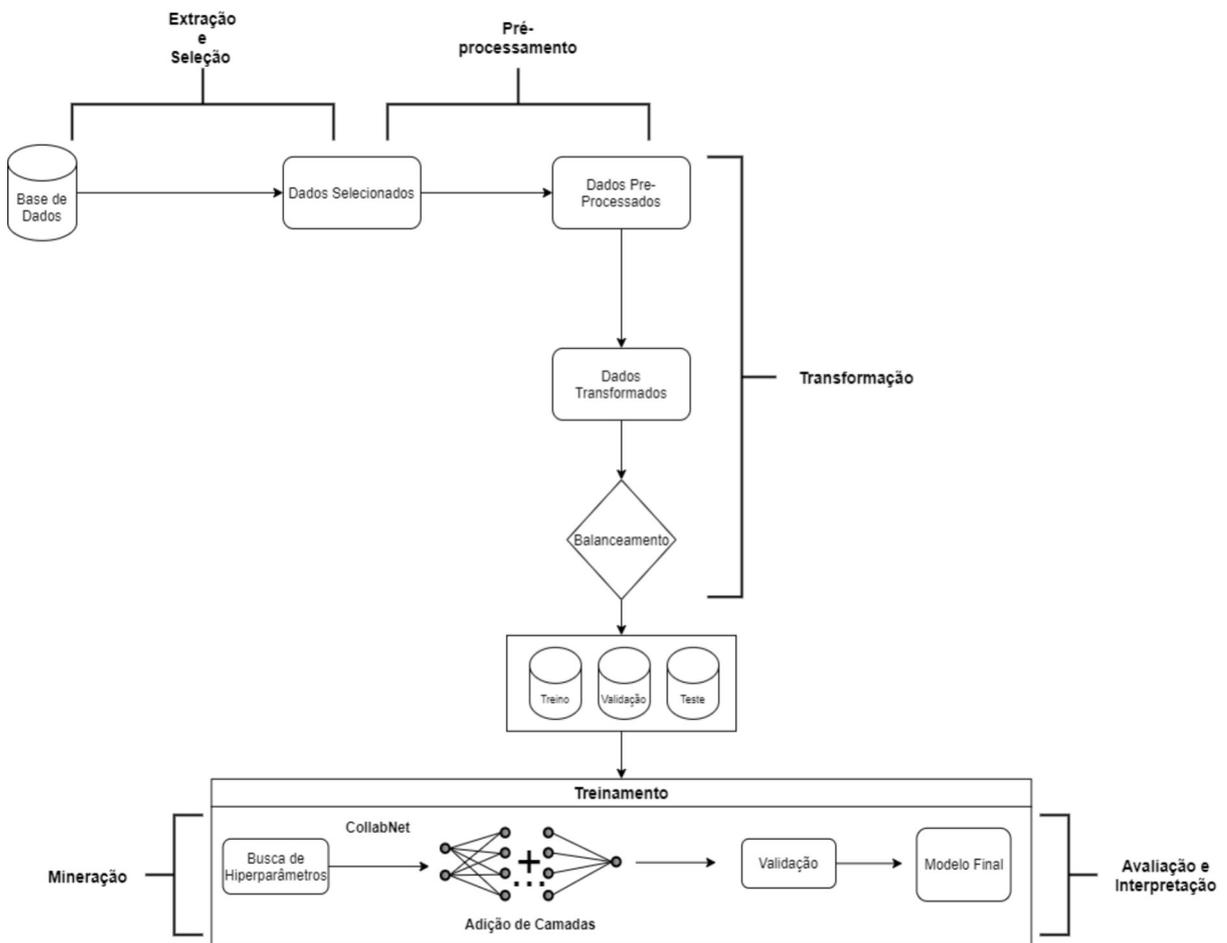
Conseqüentemente, a derivada da função de ativação deve ser alterada para o uso no *backpropagation*, como apresenta a Equação 2.7, em que  $y$  é a função sigmoide e  $c$  é o fator de ponderação.

$$\frac{d\phi_A}{dn} = c \cdot (y) \cdot (1 - y) + (1 - c) \quad (2.7)$$

### 3 Classificação de Inadimplência

Os métodos que compõem a metodologia deste trabalho seguem as etapas de seleção, pré-processamento, transformação, mineração e avaliação propostas pelo KDD. Em contrapartida, os materiais são compostos sobretudo pela base de dados fornecida por uma concessionária de serviços públicos. Adaptações foram realizadas a fim de acoplar a utilização da *CollabNet* com a resolução do problema desta pesquisa, o que resulta no fluxo apresentado na Figura 7.

Figura 7 – Metodologia.



Fonte: Autor.

Cada etapa do KDD pode incorporar um ou mais estágios da metodologia. Cada uma produz um componente que é utilizado no estágio subsequente. Assim, é intenção deste trabalho uma estratégia de classificação de risco de inadimplência com base em uma rede neural treinada de forma colaborativa.

## 3.1 Extração e Seleção

A etapa inicial do método diz respeito à extração dos dados. A base utilizada neste trabalho é proveniente da colaboração de uma concessionária de serviço público do setor de saneamento. A fim de compreender os dados disponibilizados, inicialmente foi realizado um estudo sobre o modelo de negócio adotado pela companhia. Este estudo foi acompanhado por um especialista que auxiliou na interpretação dos dados e na identificação das informações necessárias para a abordagem do problema, de acordo a literatura sobre *credit scoring*.

Os dados foram disponibilizados em arquivos no formato CSV (do inglês: *Comma-separated values*, valores separados por vírgula) e representam, de forma geral, dois tipos de informações acerca dos clientes: histórico de pagamentos e cadastro. No total, a base fornecida contou com 600.000 registros de pagamentos distribuídos igualmente no meses de julho de 2019 a maio de 2020. A importação dos dados foi realizada por meio da linguagem Python 3.8 e da biblioteca Pandas. Por tratarem-se de dados originados de um banco de dados relacional, foi necessário realizar a desnormalização dos dados nessa etapa. O intuito é mesclar e manter uma estrutura única e manipulável para a próxima etapa de pré-processamento e transformação.

## 3.2 Pré-processamento

Nesta etapa, procurou-se tratar os dados disponibilizados a fim de gerar informações úteis e confiáveis. Os dados brutos foram submetidos a procedimentos que visam encontrar representações adequadas para o uso nas etapas posteriores, em especial a etapa de treinamento do modelo. Por isso, os procedimentos aqui executados são de suma importância para o sucesso da metodologia e serão detalhados a seguir.

Procurou-se verificar a qualidade dos dados, tanto de formatação quanto de informação. Na base original foram encontrados alguns valores negativos e nulos, que foram desconsiderados e retirados. Para o processo de formatação, foram realizadas tratativas nos campos de data e valores de fatura. No primeiro, foi realizado um tratamento para adequação do formato de data para o formato aceito pela linguagem *Python*. No segundo, os valores foram convertidos para ponto flutuante.

Conforme apontado na seção anterior, neste trabalho adota-se a previsão de risco de inadimplência em concessionárias de serviços públicos como um problema de *credit scoring*. Procurou-se gerar características que correspondessem as abordagens utilizadas nessa natureza de problemas (THOMAS, 2000). Portanto, foram considerados os seguintes dados para a criação de características:

- Valor da Fatura;

- Data de Vencimento;
- Data de Pagamento;
- Mês de Referência;
- Grupo de Faturamento;
- Categoria Principal.

Com a posse desses dados, verificou-se a possibilidade de obter-se os valores necessários para a modelagem do problema. Problemas dessa natureza são abordados por meio da incorporação do histórico de pagamento do cliente e informações de cadastro, como apontado anteriormente. Em relação ao histórico, a intenção é reconhecer um padrão no pagamento dos clientes ao longo de um período de tempo, representado neste trabalho pelos meses do ano. As informações de cadastro traduzem uma possível diferença de comportamento de consumo entre os diferentes clientes de grupo de faturamento e de diferentes categorias.

No intuito de considerar o histórico de pagamento do cliente, procurou-se tratar os dados a fim de que cada registro tivesse o atraso em relação a um determinado período de tempo. Como mostra a Figura 8, é realizada sobre a tabela de pagamentos uma agregação a nível do cliente.

Figura 8 – Exemplo de tratamento dos dados históricos.

Cliente	Mês de Referência	Valor da Fatura
X	02-2019	R\$ 30,05
Y	02-2019	R\$ 120,32
X	03-2019	R\$ 80,45
X	04-2019	R\$ 10,02
Y	04-2019	R\$ 75,98
Y	03-2019	R\$ 87,52



Cliente	Pay3	Pay2	Pay1
X	30.05	80.45	10.02
Y	120.32	87.52	75.98

Fonte: Autor.

O resultado final é uma tabela com detalhamento em nível de cliente e com atributos dos seis meses anteriores ao mês analisado. Para a base, foram disponibilizados dados de julho de 2019 a maio de 2020. Portanto, ao aplicar essa transformação, restaram dados

de janeiro a maio de 2020. Similarmente, foram calculados outros valores para serem utilizados na etapa de treinamento.

Para as características que não possuem valores numéricos, como é o caso do grupo de faturamento e categoria principal, foi utilizada a técnica de *label encoded*. Esta técnica consiste no mapeamento de cada valor distinto de uma coluna em um número. Como apresentado na Figura 9, em um cenário onde têm-se três categorias de perfil associado ao cliente são obtido três valores numéricos representantes desta informação.

Figura 9 – Exemplo de *label encoded*.



Cliente	Perfil (Valor Categórico)
X	Normal
Y	Alto
Z	Baixo
R	Alto
T	Baixo
F	Normal

Cliente	Perfil (Valor Categórico)
X	0
Y	2
Z	1
R	2
T	1
F	0

Fonte: Autor.

Foram geradas as informações apresentadas na Tabela 1.

Tabela 1 – Tabela descritiva das variáveis independentes.

N	Variável	Descrição
1-6	PAY(1-6)	Valor pago nos 6 meses anteriores.
7-12	BILL(1-6)	Valor em débito nos 6 meses anteriores.
13-18	QFA(1-6)	Quantidade de faturas atrasadas.
19-24	VFA(1-6)	Valor das faturas em atraso.
25-30	DA (1-6)	Dias em atraso em relação a fatura do mês.
36	Perfil	Perfil de consumo.
37	CP	Categoria Principal.

Fonte: Autor.

O problema apresentado neste trabalho é a classificação de risco de inadimplência de um cliente. Por conseguinte, o conceito de inadimplência passa por diversas interpretações de acordo com o contexto em que é empregado, o que se torna uma dificuldade ao estudar

sobre os trabalhos da área, pois a interpretações de risco de inadimplência ou risco de crédito é um fator subjetivo e passível de avaliação.

Neste trabalho em que o cenário é prestação de serviço publico, foi definido, junto aos especialistas da área de negócio, inadimplente como todo cliente que atrasa a fatura por mais de 30 dias. A variável resposta ou variável dependente foi calculada segundo essa informação. Conseqüentemente, o valor da variável será 1 caso o cliente esteja com os pagamentos dentro do intervalo de tolerância de atraso (adimplente), e 0 caso contrário (inadimplente).

### 3.3 Transformação

Diante dos dados pré-processados, observou-se que os valores estão em escalas diferentes, a exemplo da quantidade de faturas atrasadas e valor pago. Deste modo, foi necessária a utilização de uma escala comum para que não haja distorções de valores e a distribuição original seja mantida. Optou-se pela utilização da normalização Min-Max (SINGH; SINGH, 2020), com os parâmetros de  $min = 0$  e  $max = 1$ . A Fórmula 3.1 apresenta a transformação linear aplicada pela técnica.

$$\hat{x}_i = \frac{x_i - \min(x_i)}{\max(x_i) - \min(x_i)} (\max x_{new} - \min x_{new}) + \min x_{new} \quad (3.1)$$

em que  $\hat{x}_i$  é o valor mapeado de um atributo  $x_i$  de um intervalo  $[\min(x_i), \max(x_i)]$  para um novo intervalo  $[\min x_{new}, \max x_{new}]$ . Ao aplicar essa normalização, as relações entre os valores dos dados originais são preservados, não havendo perda de informação. Logo, é possível atenuar as diferentes escalas entre os valores e modelar corretamente os dados para aplicação da técnica apresentada neste trabalho.

Ainda sobre a normalização, há um detalhe que deve ser observado: ao realizar a segmentação da base, a independência dos dados de treino, teste e validação devem ser mantidas. Ao ser treinado, com a base de treino, o algoritmo pode ser generalista a ponto de obter métricas próximas das obtidas durante o treinamento, na base de teste e validação. Por isso, é esperado que a aplicação da normalização Min-Max seja com base nos intervalos  $[\min x_{new}, \max x_{new}]$  dos dados de treino e então aplicada no mapeamento de valores  $x_i$  para valores normalizados  $\hat{x}_i$  de todos os 3 segmentos da base.

#### 3.3.1 Segmentação da base e Balanceamento de classes

De posse da base tratada, foi verificada uma distribuição acentuada para os casos negativos, ou seja, de clientes classificados como inadimplentes, como mostra a Tabela 2.

Tabela 2 – Distribuição da base pré-processada.

Classe	Registros
Negativo	299.997
Positivo	200.003

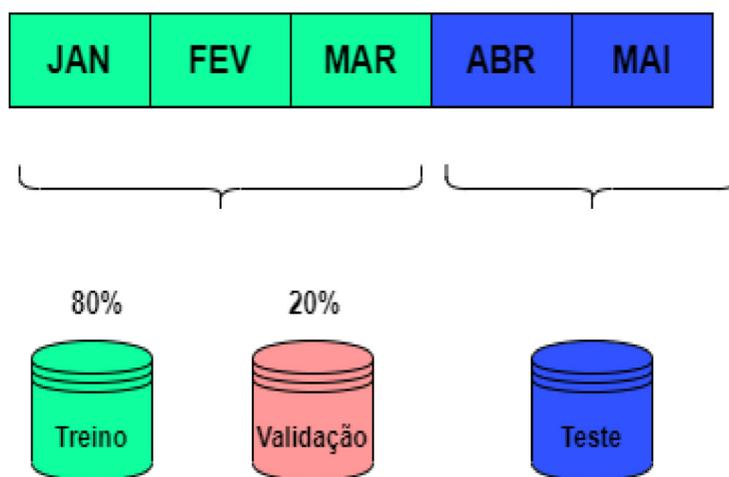
Fonte: Autor.

O balanceamento de classes foi realizado por subamostragem. É esperado que, devido a grande quantidade de registros, o uso desta técnica não impacte na diminuição na perda de informações na classe da maioria.

Neste trabalho, para a validação da metodologia empregada, é proposta a segmentação da base conforme as datas dos dados processados. Pretende-se demonstrar a geração correta das características de modo que estas possam ser aplicadas independentemente do período em questão. Ainda é proposto avaliar a capacidade de generalização da *Collabnet* por meio de uma validação cruzada.

O método de segmentação dos dados é com base no período de tempo das informações fornecidas, janeiro, fevereiro, março, abril e maio. O intuito é que se comprove que a geração das características propostas na metodologia incorporam valores que de fato correspondam ao histórico do cliente. Então, o modelo gerado com o treinamento de um período anterior possa ser aplicado a outro período, demonstrando também a independência da base. A Figura 10 apresenta a divisão realizada.

Figura 10 – Validação.



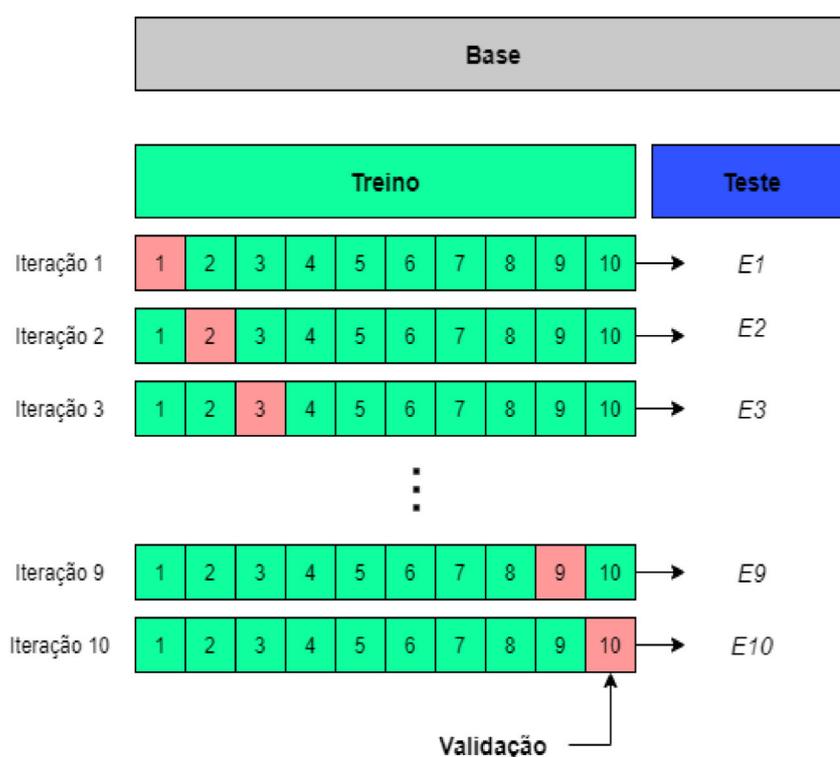
Fonte: Autor.

Os dados referentes aos meses de janeiro, fevereiro e março são utilizados para o treino e validação (80%/20%). Os referentes aos meses de abril e maio constituem a base de teste.

## Validação Cruzada

Para comprovar a capacidade generalista do modelo, é proposta a validação cruzada por meio do esquema *k-fold* de Kohavi et al. (1995). Como apresenta a Figura 11, o método consiste em segmentar a base em *k* parcelas, de modo que o treino seja realizado com *k-1* segmentos e a validação com o segmento restante. Para completar as combinações, devem ser feitas *k* iterações, de modo que cada iteração gere um modelo avaliado com uma base de teste. Ao final são calculadas as métricas de média e desvio padrão sobre os erros dos modelos gerados.

Figura 11 – Método *k-fold* com *k=10*.



Fonte: Autor.

Neste trabalho, é utilizado o método *k-fold* com 10 segmentos, onde nove segmentos são utilizados para o treino e *ofold* restante para a validação dos hiperparâmetros da *CollabNet*. Ao final das inserções das camadas, é realizada a avaliação do modelo com a base de teste. Este processo é então repetido 10 vezes.

De posse da base tratada, é realizada a divisão inicial em treino e teste, 75% e 25% respectivamente, e então aplicado o balanceamento de classes por sobreamostragem. Foi utilizado sobre a base de treino a técnica *k-fold* com *k=10*, e ao final de cada iteração, o resultado é registrado e obtida a média,  $\bar{x} = \sum \frac{x_i}{n}$ , e o desvio padrão,  $\sigma = \sqrt{\sum \frac{(x_i - \bar{x})^2}{n}}$ , das métricas dos experimentos.

## 3.4 Treinamento

Neste trabalho, é proposta a utilização da *CollabNet* no contexto da classificação do risco de inadimplência. Até então, todo trabalho de manipulação e segmentação dos dados é realizado por meio da linguagem Python e de bibliotecas acopladas. Desta etapa em diante, o ambiente de treinamento e avaliação foi o Matlab, sem bibliotecas adicionais.

### 3.4.1 Busca de Hiperparâmetros

Em redes neurais, os pesos incorporados durante o processo de aprendizagem são considerados os parâmetros do algoritmo. Logo, dada uma função  $y = F(x, \theta)$ , em que  $x$  é o vetor de entrada e  $y$  a saída,  $\theta$  seriam os pesos da rede neural. Porém, alguns aspectos controlam o comportamento do algoritmo durante o processo de aprendizagem, estes valores são chamados de hiperparâmetros (CLAESEN; MOOR, 2015). A escolha dos hiperparâmetros pode ser considerada um processo crítico em problemas de aprendizado de máquina, pois valores diferentes geram modelos diferentes que por consequência impactam nos resultados (WANG; GONG, 2018).

O algoritmo *CollabNet*, utilizado neste trabalho, propõe o uso de alguns valores adicionais que são explanados na próxima subseção. De modo geral, os hiperparâmetros tradicionais ajustados nessa metodologia são:

- $\eta$ : Taxa de aprendizado.
- $\epsilon$ : Número de épocas de treinamento de cada camada.
- $N$ : Quantidade de neurônios em cada camada.
- $NC$ : Número de camadas ocultas.

Por definição da técnica, a função de ativação utilizada é a sigmoide e o aprendizado ocorre por lote, ou seja, a cada iteração os pesos são atualizados com base em todos os dados de treino.

A escolha dos hiperparâmetros foi realizada de forma empírica, por meio de sucessivos experimentos. Dessa maneira, após o treinamento com os valores escolhidos, a amostra de dados de validação é utilizada para a avaliação do ajuste do modelo. Por ser uma base diferente da usada no treino, é esperada uma avaliação imparcial e que possa certificar a seleção dos valores.

### 3.4.2 Adição de Camadas

No contexto da *CollabNet*, alguns hiperparâmetros adicionais são necessários para que seja possível o controle da influência da nova camada no aprendizado global da rede.

Nesta técnica, como já descrito no capítulo anterior, a adição de camadas ocorre após o treinamento completo da camada anterior de maneira que a adição desta nova ocorra de forma gradual e colaborativa, a ponto de não ocorrer elevações bruscas no erro de saída da rede.

A técnica fornece ao projetista alguns valores que podem ser controlados durante o processo de treinamento da rede. Estes valores compõem o cálculo do gradiente e da entrada líquida do neurônio e são controladas de forma empírica. São eles:

Tabela 3 – Hiperparâmetros da CollabNet.

Hiperparâmetro	Domínio
$\Delta C$	[0, 1.0]
$saltoC$	[1, $\epsilon$ ]
$C$	[0,1]

Fonte: Autor.

$\Delta C$  corresponde ao valor que deve ser incrementado a cada atualização do valor  $C$ . O  $saltoC$  é o passo da iteração que deve ser atualizado o  $C$ . O  $C$  neste caso é o valor inicial da variável  $c$  ao começar o treinamento da nova camada. Partindo de  $C$ , a variável  $c$  é incrementada em  $\Delta C$  a cada  $saltoC$ . Cada um desses valores influencia diretamente no processo de adição de uma nova camada e deve ser cuidadosamente escolhido pelo projetista. Portanto, a escolha errônea dos parâmetros pode causar um comportamento inesperado no erro global da rede.

Selecionando corretamente os hiperparâmetros, pretende-se adicionar novas camadas a uma rede neural de modo que não haja perturbação no erro da rede e o aprendizado da nova camada seja incorporado. Para ratificar tais feitos, é proposta a comparação dos resultados ao final do treinamento de cada camada. As métricas de avaliação serão apresentadas na próxima subseção.

### 3.5 Avaliação e Interpretação

Para avaliação dos resultados dos modelos, é proposta a utilização das métricas de acurácia, precisão, sensibilidade, especificidade e  $f$ -score. Além de um método que busque classificar corretamente os clientes inadimplentes e adimplentes, deve ser possível verificar a não perturbação do erro global e a incorporação do aprendizado de cada nova camada. Para tal, é realizada a comparação do erro global e das métricas descritas acima ao final do treinamento de cada camada.

Optou-se por definir o cliente inadimplente como classificação negativa e o adimplente como positiva, pois é considerado como evento de interesse os clientes que irão cumprir com o contrato da prestação de serviço, é tratado como inadimplente o cliente que atrasa

por mais de 30 dias o pagamento. Também é esperado que, ao considerar o uso do balanceamento de classes proposto na etapa anterior, as métricas calculadas não sejam enviesadas pela distribuição original da base.

Podemos interpretar as medidas Falso Positivo, Falso Negativo, Verdadeiros Negativos e Verdadeiros Positivos da seguinte forma:

- Falsos Positivos (FP): casos em que o cliente se tornou inadimplente, caso negativo, porém o modelo classificou como positivo, ou seja, adimplente.
- Verdadeiros Negativos (VN): casos em que a classe é negativa e o modelo classificou corretamente como negativa.
- Falsos Negativos (FN): casos em que o cliente é sinalizado como adimplente, classe positiva, e o modelo classificou como negativa.
- Verdadeiros Positivos (VP): casos de clientes adimplentes corretamente classificados pelo modelo.

De posse dessas medidas, há possibilidade de calcular as métricas responsáveis pela avaliação dos resultados obtidos e validação dos modelos. A métrica de acurácia, é normalmente conhecida por fornecer uma avaliação geral do experimento, e pode ser definida como a proporção de todos os resultados corretos em relação a população do experimento (FAWCETT, 2006). Então, o cálculo do ACU (Acurácia), pode ser definido segundo a Equação 3.2.

$$ACU = \frac{VP + VN}{VP + VN + FP + FN} \quad (3.2)$$

A especificidade (ESP) representa a taxa de verdadeiros negativos, ou seja, dentre todas os clientes inadimplentes, quantos o modelo corretamente classificou (VN). Quanto maior este valor, melhor o modelo capta os casos de inadimplência. Seu cálculo é apresentado pela Equação 3.3.

$$ESP = \frac{VN}{VN + FP} \quad (3.3)$$

De forma análoga, considerando todos os casos de classe positiva, a sensibilidade (SEN) pode ser considerada como taxa de verdadeiros positivos. Logo, seu cálculo pode ser inferido como segue a Equação 3.4:

$$SEN = \frac{VP}{VP + FN} \quad (3.4)$$

No contexto deste trabalho, a precisão (PRE), ou valor preditivo positivo, pode ser considerada como a taxa de acerto do modelo em relação as classificações de clientes adimplentes. Esta métrica indica a incompatibilidade entre todas as classificações da classe positivo que o modelo gerou e destas quantas de fato correspondem ao rótulo positivo. Pode-se calcular essa taxa por meio da Equação 3.5:

$$PRE = \frac{VP}{VP + FP} \quad (3.5)$$

De modo oposto, o valor preditivo negativo (VPN) pode ser considerado como a taxa de acerto do modelo em relação as classificações de clientes inadimplentes. Essa taxa é calculada por meio da Equação 3.6:

$$VPN = \frac{VN}{VN + FN} \quad (3.6)$$

Por fim, o *f-score* considera tanto o cálculo precisão quanto o da sensibilidade, sendo composto pela média harmônica de ambas as métricas. Seu calculo portanto, é definido pela Equação 3.7

$$f\text{-score} = 2 \cdot \frac{PRE \cdot SEN}{PRE + SEN} \quad (3.7)$$

## 4 Resultados e Discussão

Esta seção apresenta os resultados obtidos por meio da metodologia proposta, que averigua a aplicação de uma rede neural colaborativa, a *CollabNet*, na classificação de risco de inadimplência em concessionárias de serviços públicos. Como apresentado no Capítulo 3, no intuito de comprovar tanto a geração das características quanto a técnica, são propostas duas validações: uma referente ao período de tempo da base disponibilizada e uma validação cruzada.

Além de resultados satisfatórios em relação as métricas apresentadas na Seção 3.5, a metodologia deste trabalho propõe o uso da *CollabNet* no processo de inserção de novas camadas em uma rede neural. Para demonstrar a eficiência da técnica, será exposto o comportamento do erro global e das métricas da rede neural ao longo das inserções das camadas. Igualmente será apresentado um comparativo entre modelos treinados com diferentes quantidades de camadas ocultas.

Como descrito na seção anterior, é utilizado o balanceamento por sobreamostragem para contornar a distribuição original da base. Esta distribuição é acentuada para os casos negativos, referenciando os clientes inadimplentes. Logo, com o objetivo de avaliar de forma imparcial as métricas dos resultados, foi optado por realizar o balanceamento de classes. Dessa forma, é esperado que o modelo final seja eficiente tanto na classificação dos clientes adimplentes quanto dos inadimplentes.

No que diz respeito a parametrização, a busca dos hiperparâmetros dos modelos foi realizada de forma empírica. Ressalta-se que, apesar de não ter sido utilizada nenhuma técnica de otimização nessa etapa, os experimentos foram realizados de forma exaustiva a fim de trazer as melhores configurações de aprendizagem do modelo e consequentemente de resultados.

Portanto, foram testadas diversas combinações de parâmetros, tanto as que compõem a arquitetura da rede neural, como o número de neurônios e de camadas, quanto as relacionadas à *CollabNet*:  $\Delta C$  como o valor de incremento da variável  $C$ , o *saltoC* como o passo da iteração que deve ser atualizado o  $C$  e o valor inicial de  $C$ . A Tabela 4 apresenta o espaço de busca de hiperparâmetros utilizado durante os experimentos.

Como apresentado na Seção 3.3.1, é proposta uma divisão baseada nos meses dos dados disponibilizados, em que os dados de janeiro, fevereiro e março são segmentados em 80% e 20% para treino e validação, respectivamente. Os dados referentes aos meses de abril e maio constituem a base de teste. É importante recapitular que os casos negativos são referentes aos clientes inadimplentes e os casos positivos aos clientes adimplentes. Após

Tabela 4 – Espaço de busca de hiperparâmetros.

Hiperparâmetro	Domínio
Numero de Neurônios	[6-30]
Quantidade de Camadas	[1,4]
Taxa de Aprendizagem	[0,001, 0,00001]
Delta C	[0,1, 0,0001]
Salto C	[2,50]
Início de C	[0,0,5]

Fonte: Autor.

o balanceamento da base, a distribuição dos dados, em valores absolutos, seguiu conforme a Tabela 5.

Tabela 5 – Distribuição dos dados de treino, validação e teste.

Dados	Quantidade
Treino	99.243
Validação	24.793
Teste	75.967

Fonte: Autor.

De posse da base balanceada, os experimentos foram realizados a fim de buscar a melhor configuração da rede. Seguindo a técnica *CollabNet*, as inserções das camadas foram realizadas sequencialmente. Portanto, ao final do treinamento de uma camada é inserida outra camada com a mesma quantidade de neurônio. A Tabela 6 apresenta os valores das principais métricas por inserção de cada camada. Os valores são resultantes da aplicação do método sobre a base de teste:

Tabela 6 – Resultados por inserção de cada camada

Camada	ACU	PRE	VPN	ESP	SEN	F-Score
1°	0,84258	0,81433	0,87641	0,79765	0,88752	0,84935
2°	0,87438	0,83214	0,92893	0,81079	0,93797	0,88189
3°	<b>0,88149</b>	<b>0,84210</b>	<b>0,93115</b>	<b>0,82391</b>	<b>0,93908</b>	<b>0,88795</b>

Fonte: Autor.

É de interesse deste trabalho observar que a cada inserção houve uma melhoria em todas as métricas avaliadas. Constatou-se que durante os experimentos que a partir da quarta camada não houve um incremento significativo nas métricas que justificasse a inserção de uma nova camada, portanto o modelo final possui três camadas.

De imediato, observa-se que o método alcançou bons resultados, principalmente nas métricas de sensibilidade (SEN) e valor predito negativo (VPN). Ao observar o valor da

métrica de desempenho geral do modelo, a acurácia (ACU), percebe-se que, da primeira para a segunda camada, houve uma melhoria de aproximadamente 0,032 no valor da métrica, enquanto da segunda para a terceira camada houve um acréscimo menor, de 0,007. Isso representa uma melhoria na classificação tanto dos clientes adimplentes, quanto inadimplentes.

A precisão (PRE) revela o quanto o modelo acertou dentre todas as classificações geradas da classe positiva, ou seja, qual a probabilidade de acerto do modelo ao gerar classificações para clientes adimplentes. A inclusão da segunda camada forneceu uma melhoria de aproximadamente 0,02 no valor da precisão, enquanto a inclusão da terceira camada permitiu um acréscimo de aproximadamente 0,01. A boa precisão demonstra o baixo erro do modelo ao gerar as classificações positivas. De modo inverso, o valor preditivo negativo (VPN) revela a probabilidade de acerto do modelo ao gerar classificações para clientes inadimplentes. O valor dessa métrica está maior que o da precisão, o que revela que o modelo é mais assertivo ao prever classes inadimplentes. A evolução no valor da métrica também foi mais expressiva da primeira para a segunda camada.

A taxa de especificidade corresponde a capacidade do modelo em captar os clientes inadimplentes. Esta taxa apresentou uma evolução mais tímida durante a inserção das camadas porém apresentou um salto de 0,025 da primeira para a terceira inserção de camada. De forma análoga, a métrica que mais destacou-se no método foi a sensibilidade. Ela traduz a capacidade do modelo em capturar corretamente classificações de clientes adimplentes. A inclusão da primeira camada provocou um acréscimo de 0,05 no valor da métrica.

Seguindo o comportamento das outras métricas, o f-score apresentou melhor evolução na inserção da segunda camada, com um ganho de aproximadamente 0,03 nesta inserção. Seu alto valor de 0,88 é um bom indicativo para a resolução do problema proposto neste trabalho.

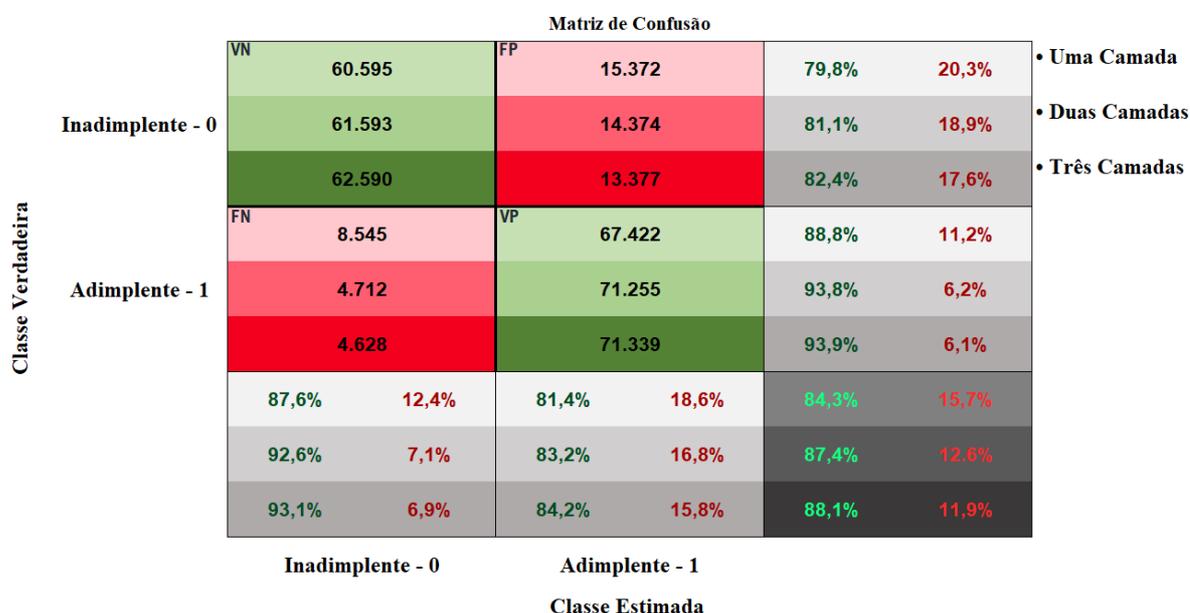
Ao analisar as métricas expostas, em especial a de especificidade e sensibilidade, e tendo em vista o balanceamento de classes presente na base, constata-se que o método detecta com mais facilidade os casos de clientes adimplente, caso positivo, em relação aos casos de clientes inadimplentes. De certo modo, é esperado que a tarefa de classificação para os casos de clientes inadimplentes seja mais complexa. Tal fato deve-se, em grande parte, por características inerentes ao domínio apresentado neste trabalho. Porém, ao serem observadas as métricas de precisão, 82,3% e valor preditivo negativo, 93,9%, verifica-se que a probabilidade de acerto do método ao selecionar uma classe inadimplente é maior que ao selecionar uma classe como adimplente.

Desse modo, as métricas indicam eficiência na resolução do problema aqui exposto. O método dispõe de um acerto médio de 88% ao ser confrontado por uma base totalmente independente da qual foi utilizada para seu aprendizado. Além disso, no contexto da

classificação de risco, a sobreposição do valor preditivo negativo em relação ao valor da precisão configura um ótimo indicativo para tomada de decisão que venha a ser baseada nesta metodologia.

Ainda é proposta deste trabalho evidenciar a melhoria de desempenho do método por meio da inclusão de novas camadas. Para tal, é apropriada uma visualização que demonstre a evolução no valor das métricas de forma clara e objetiva. Sob a ótica da matriz de confusão apresentada na Figura 12, é possível observar o progresso tanto no valor das métricas quanto dos casos de classificação.

Figura 12 – Matriz de confusão durante o treinamento do modelo final.



Fonte: Autor.

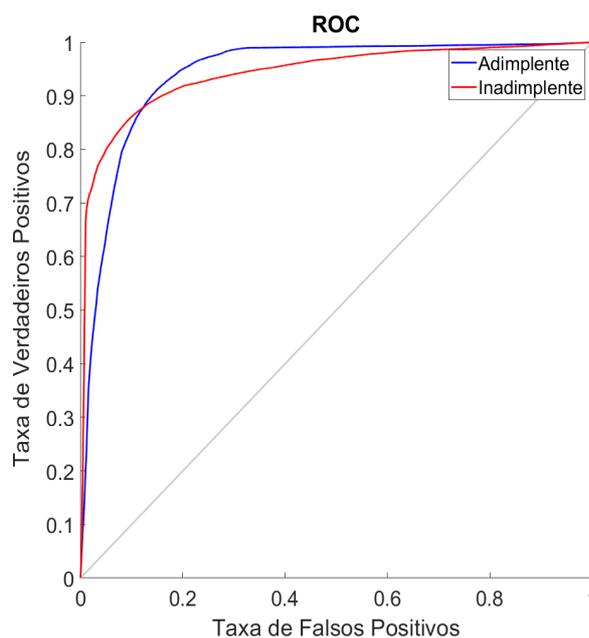
Na figura, os quadrantes em verde representam as classificações corretas, os quadrantes em vermelho as classificações incorretas e os quadrantes em cinza o valor das métricas em percentual. Cada quadrante é segmentado pela quantidade de camadas inseridas. A lacuna gerada é preenchida pelo valor correspondente da matriz de confusão e por uma das cores, em que sua intensidade obedece a ordem da quantidade de camadas. Assim, a cor mais amena representa a primeira camada e a cor mais forte, a última camada.

Sob esse ângulo, constata-se com clareza o comportamento desejado de evolução nos valores calculados, apresentados agora em percentual. Nos quadrantes verdes, o incremento das classificações corretas ao decorrer das inserções é espelhado na melhoria nos valores das métricas de acerto. Estes valores são visualizados, na cor verde, tanto verticalmente como horizontalmente. O fenômeno inverso é observado com as classificações incorretas, que são decrementadas ao decorrer das inserções, refletindo assim na diminuição dos valores das métricas de erro observados em vermelho.

Na diagonal principal, observa-se o aprimoramento no valor da acurácia, que partiu do valor de 84,3% ao final do treinamento da primeira camada e chegou a 88,1% ao final do treinamento da terceira camada. Isso representa uma melhoria de 3,8 % na performance do método. O mesmo ocorre com as métricas de especificidade e sensibilidade observadas horizontalmente e as de precisão e valor preditivo negativo observadas verticalmente. Tal fato ainda é validado pela movimentação presente dos casos de classificação dos quadrantes vermelhos para os quadrantes verdes.

Outra forma de diagnóstico do modelo é por meio da curva ROC (do inglês: *Receiver operating characteristic*). Ela tem por objetivo avaliar a capacidade discriminativa de um modelo. Para tal, é plotado para cada classe do problema uma curva que mede a taxa de verdadeiros positivos, sensibilidade, pela taxa de falsos positivos, especificidade. Esta curva é composta por pontos de corte que representam a saída do classificador para cada classe. Portanto, a curva acompanha o intervalo contínuo de  $[0,1]$ . A Figura 13 apresenta a curva ROC do modelo com três camadas, onde a linha vermelha é a curva da classe de clientes inadimplentes e a curva azul, a de clientes adimplentes.

Figura 13 – Curva ROC.



Fonte: Autor.

Observa-se que ambas curvas estão próximas ao canto superior esquerdo, indicando um bom classificador. No intervalo de saída entre 0,5 e 1, a classificação para a inadimplência tem taxas de verdadeiros positivos maiores que a classificação de adimplência. De forma análoga, no intervalo de 0 e 0,5, a classificação para a rejeição da inadimplência funciona melhor, representada pela curva vermelha. Esses fatores montam um intervalo de incerteza de classificação maior para os casos de clientes inadimplentes.

O cenário apresentado pela curva ROC é corroborado pelas métricas já apresentadas, das quais a sensibilidade, com valor de 93,9%, apresentou-se superior à especificidade, com valor de 82,3%. Diversos podem ser os fatores que influenciam nesse comportamento, em grande parte são inerentes ao domínio estudado e naturalmente devem ser objeto de estudo em trabalhos futuros que visam o aperfeiçoamento das métricas aqui expostas.

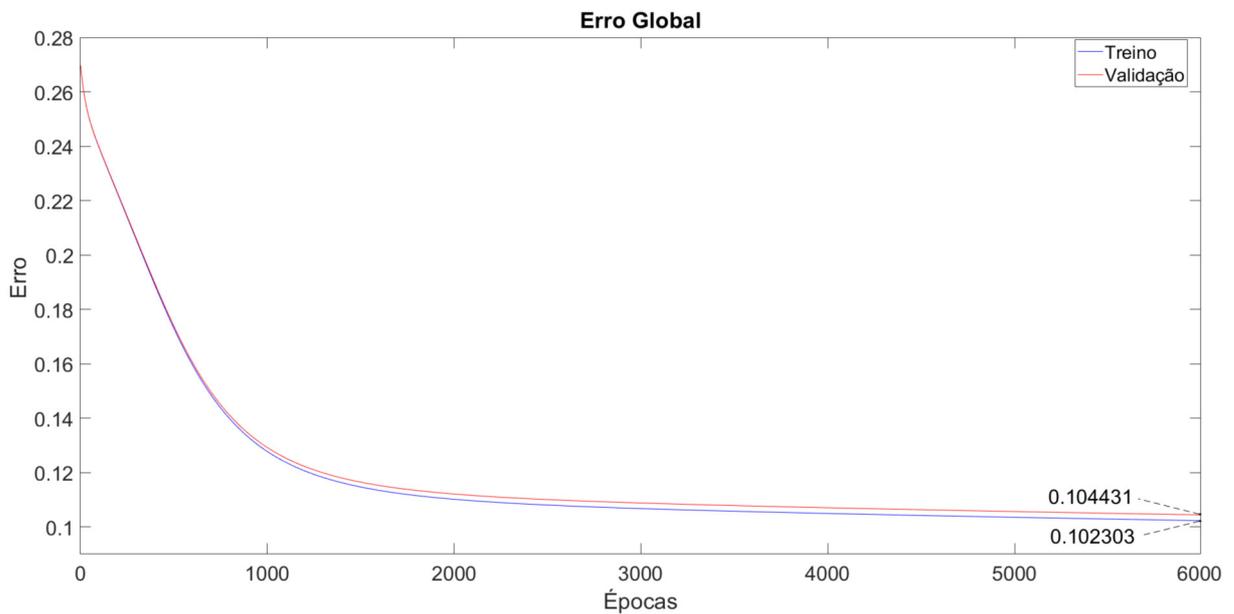
### Comparativo das Curvas de Aprendizado

No contexto do aprendizado da *CollabNet*, um aspecto importante a ser ponderado é o momento em que as inserções das camadas ocorrem durante o treinamento global. Foi considerado que a inserção deve ser realizada após o esgotamento do aprendizado de cada camada, ou seja, quando não for mais observável o comportamento inerente do aprendizado caracterizado pelo decaimento do erro global da rede. Desse modo, é importante realizar a inserção da nova camada em um momento que ela contribua para o aprendizado da rede e que essa inclusão não ocasione a perturbação do erro global do modelo.

Esse fenômeno foi observado ao realizar a comparação da rede com e sem as inserções da camadas. O experimento consistiu em treinar a rede fixando-se as quantidade de épocas de treinamento e os hiperparâmetros, alterando somente a inclusão ou não inclusão de uma nova camada. Para tal procedimento foram armazenados os pesos das ligações entre os neurônios da rede em cada momento da inclusão da nova camada e posteriormente treinado o modelo sem a inserção. A rede foi treinada durante 6.000 épocas, com taxa de aprendizagem 0,00005 e com 12 neurônios em cada camada. A Figura 14 apresenta o comportamento do erro global ao realizar o treinamento somente com uma camada.

Na Figura 14 observam-se duas curvas de aprendizado. A linha azul representa a curva de aprendizado do treino, calculada a partir do conjunto de dados de treinamento. Esta curva é considerada um indicativo de como o modelo está aprendendo. Já a linha vermelha representa a curva de aprendizado de validação, que expressa o quão bem o modelo está generalizando. Ao acompanhar ambas as curvas é possível observar que o treinamento encontra um ponto de estabilidade, que é estendido até a época 6.000, onde sobre os dados de treino apresentou um EQM de 0,102303 e sobre os dados de validação um EQM de 0,104431. A lacuna entre as curvas indica um bom ajuste do modelo de modo a conferir a rede uma capacidade de generalização. Pela consistência das curvas é possível inferir também que o conjunto de dados de treino é representativo diante dos conjunto de dados de validação. Ainda sob esse aspecto, demonstra-se que o modelo não sofreu de *overfitting* (sobreajuste) ou *underfitting* (sub-ajuste).

Figura 14 – Comportamento do erro quadrático médio nos dados de treino e validação com uma camada



Fonte: Autor.

Como mencionado, os experimentos basearam-se na comparação dos modelos sem e com a inserção de novas camadas. É então inserido na época 2.000 uma nova camada partindo dos pesos previamente armazenados. Neste ponto, a inserção proposta pela *CollabNet* exige ao projetista o ajuste de alguns hiperparâmetros. Como citado na Seção 2.5, esses valores controlam o comportamento da variável  $C$ , que por sua vez influencia diretamente em como o aprendizado será incorporado durante o treinamento das camadas e conseqüentemente no erro global do modelo. Portanto, a combinação do  $\Delta C$  como o valor de incremento da variável a cada *saltoC*, partindo de uma valor inicial, deve ser ajustada pelo projetista. A Tabela 7 apresenta o melhor conjunto de hiperparâmetros alcançados durante os experimentos.

Tabela 7 – Melhores Hiperparâmetros.

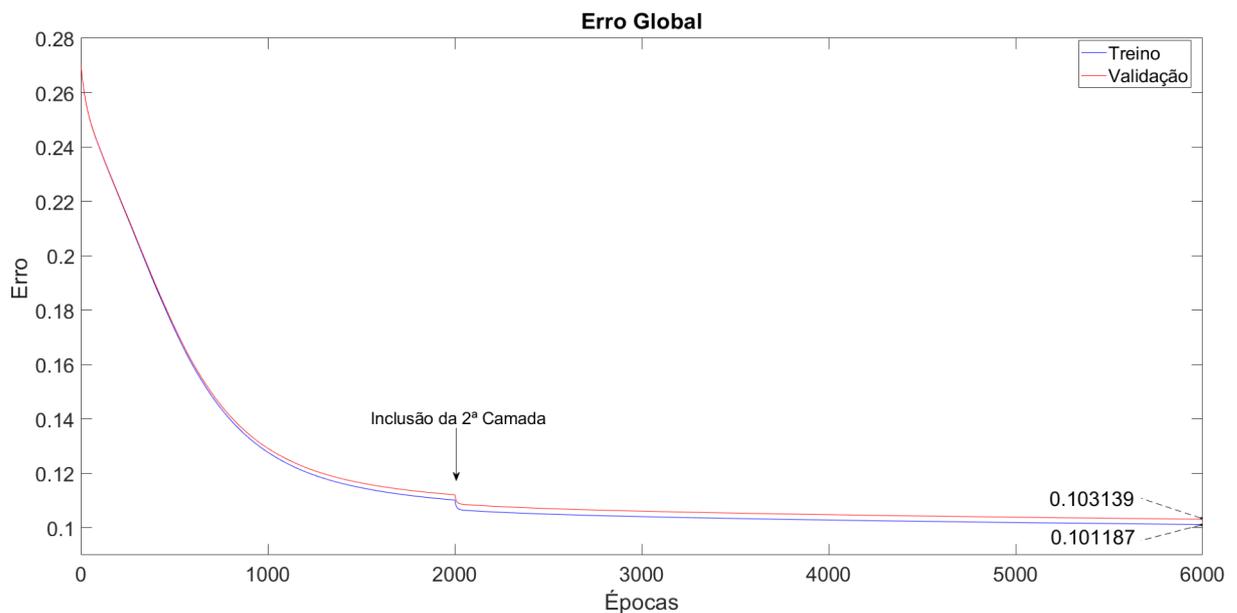
Hiperparâmetros	Valor
Numero de Neurônios	12
Quantidade de Camadas	3
Taxa de Aprendizagem	0,00005
Delta C	[0,0005, 0,01]
Salto C	10,25
Início de C	0

Fonte: Autor.

A parametrização da variável  $C$  ocorreu de forma empírica. Nos experimentos, semelhante a Freitas Jr. et al. (2018), observou-se que valores de incrementos elevados

podem comprometer o erro global da rede, bem como se aplicados em curtos intervalos entre épocas. Os melhores resultados foram obtidos partindo do valor inicial zero. A cada 10 e 25 épocas, o valor inicial de C foi incrementado com valores entre 0,005 e 0,01. Esses valores permitiram com que a inclusão da nova camada ocorresse de forma gradual e completa. A Figura 15 apresenta o comportamento do erro global da rede com duas camadas ocultas.

Figura 15 – Comportamento do erro quadrático médio nos dados de treino e validação com duas camadas ocultas

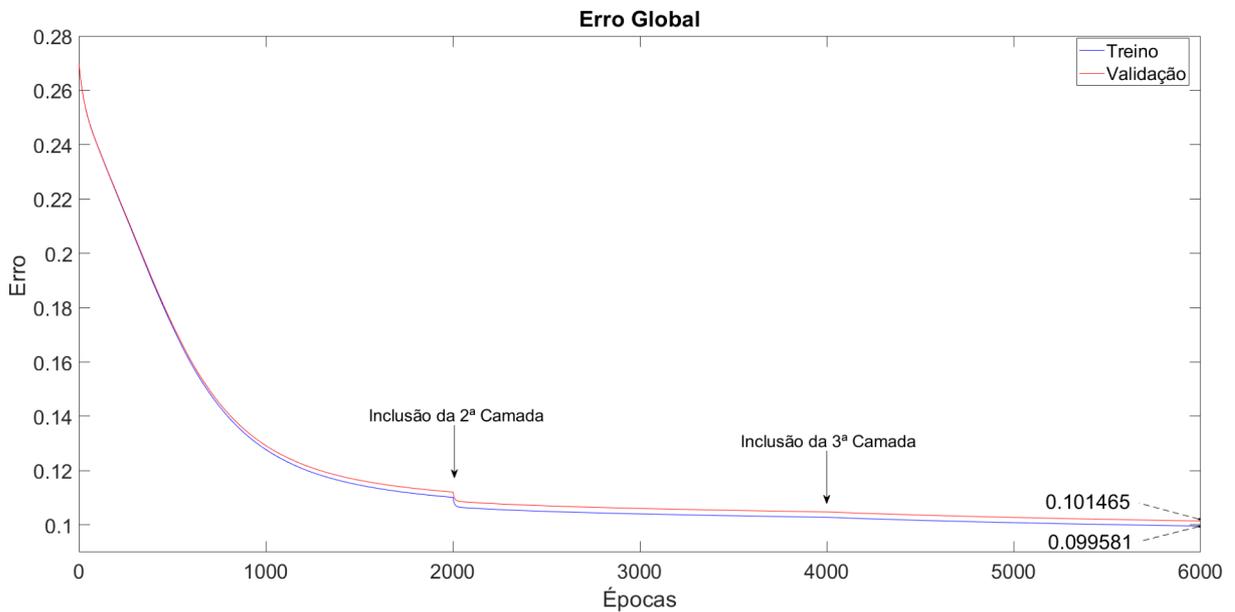


Fonte: Autor.

No momento da inclusão da segunda camada, o erro global da rede começa a seguir uma trajetória decrescente, de modo a colaborar para o aprendizado da rede e consequentemente contribuir para a diminuição do EQM ao final do treinamento, 0,101187 sobre os dados de treino e 0,103139 sobre os dados de validação. Outro comportamento verificado é a não perturbação do erro global. Por conta da aleatoriedade da inicialização dos pesos da nova camada, o comportamento inverso seria o esperado ao anexar-se essa nova camada, porém por meio da parametrização da *CollabNet* foi possível atingir este comportamento.

Seguindo os mesmos parâmetros abordados anteriormente, foi aplicada a inclusão da terceira camada oculta à rede neural. Então, é verificado um decaimento do EQM sobre os dados de treino para 0,099581 e sobre os dados de validação para 0,101465. Após experimentos com inserções de mais camadas, notou-se que o ganho no erro global não era mais significativo para a resolução do problema e portanto o modelo final é composto por três camadas ocultas. A Figura 16 apresenta as curvas de aprendizado de treino e validação do modelo final deste trabalho.

Figura 16 – Comportamento do erro quadrático médio nos dados de treino e validação com três camadas ocultas



Fonte: Autor.

### Validação Cruzada

Para avaliação mais efetiva da técnica e de sua capacidade de generalização, é utilizada uma validação cruzada por meio do esquema *k-fold*, com  $k=10$ . Logo, de posse da base tratada, realizou-se a divisão inicial em treino e teste, 75% e 25% respectivamente, e então aplicado o balanceamento de classes por sobreamostragem. São obtidas bases com a seguinte distribuição:

Tabela 8 – Distribuição dos dados treino e teste.

Dados	Quantidade
Treino	149.889
Teste	50.114

Fonte: Autor.

A base de treino é então segmentada em 10 parcelas, onde nove delas são utilizadas para treino e uma para a validação. Neste experimento, a validação serviu para comprovar a escolha dos hiperparâmetros. Os parâmetros referentes a *CollabNet* foram replicados enquanto nos outros tiveram alteração somente da taxa de aprendizagem, 0,00001, e o número de neurônios, 14. O número de camadas ocultas também permaneceu o mesmo. A Tabela 9 apresenta aplicações dos modelos sobre a base de teste utilizando validação cruzada.

Tabela 9 – Tabela de Resultado da validação cruzada.

<b>Métrica</b>	<b>Média</b>	<b>Desvio Padrão</b>
ACU	0,85263	0,00037
PRE	0,81643	0,00222
VPN	0,89421	0,00259
ESP	0,80932	0,00352
SEN	0,89853	0,00318
F-Score	0,85551	0,00033

Fonte: Autor.

Ao confrontar os resultados da validação cruzada com os resultados dos experimentos anteriores (Tabela 6), verifica-se que a ordem de melhores métricas é mantida, com destaque para sensibilidade e valor predito negativo. Os valores absolutos das métricas tiveram um leve decréscimo. O fenômeno pode ser associado ao sucesso da geração de características que permitiu incorporar fatores sazonais ao modelo. Ao contrário dos experimentos anteriores, a aleatoriedade proposta pelo esquema de validação cruzada *k-fold* desorganiza a temporalidade da base de dados. Portanto, a base de treino é trocada a cada iteração. Nessa conjuntura, a metodologia foi capaz de apresentar métricas aceitáveis para a resolução do problema.

Ainda sobre os resultados, o baixo desvio padrão demonstra que os resultados dos experimentos estão próximos da média. Consequentemente, essa baixa dispersão em um contexto de diferentes bases pode demonstrar a capacidade do método em encontrar padrões nas características geradas e assim efetivamente gerar classificações corretas.

Ao final comprovou-se a eficiência do método tanto no que diz respeito a inserção colaborativa de camadas ocultas em uma rede neural, quanto a classificação da base de risco de inadimplência no contexto de uma concessionária de serviço público. Há carência de estudos e de base de dados que compartilhem o domínio estudado neste trabalho. Por outro lado, os resultados aqui expostos pretendem instigar outros trabalhos à investigar a mesma problemática.

## 5 Conclusão

A prestação indireta de serviços públicos por meio de concessões a empresas privadas celebra uma parceria entre os poderes público e privado que visa a preencher a lacuna existente entre os riscos econômicos da exploração industrial e a eficiência no fornecimento de um serviço essencial. Nesse cenário, a regulamentação do Estado ocorre por meio da ação de órgãos e instituições reguladoras que estabelecem tetos tarifários e diretrizes de prestação de serviço. A redução dos custos operacionais e o aprimoramento dos processos de cobrança e de comunicação devem ser parte do planejamento estratégico dessas empresas.

Manter o equilíbrio entre a continuidade de um serviço público e essencial com a necessidade de arrecadação por partes das empresas é uma complexa tarefa. A utilização de técnicas computacionais que classifiquem o comportamento dos contribuintes diminui a subjetividade no processo de análise de inadimplência. Nesse aspecto, a metodologia apresentada neste trabalho apresentou resultados que indicam a viabilidade da inclusão dessas técnicas no processo de tomada de decisão das empresas e conseqüentemente no ciclo de relacionamento com o cliente.

A proposta apresentada neste trabalho refere-se à utilização da *CollabNet* na classificação de risco de inadimplência no contexto de concessionárias de serviços públicos. Os resultados obtidos por meio dos experimentos demonstraram um desempenho satisfatório da metodologia, tanto na resolução do problema de classificação quanto nas inserções de camadas ocultas na rede neural. As métricas de acurácia, valor preditivo negativo, sensibilidade e *f-score*, calculadas sobre os dados balanceados e após a busca de hiperparâmetros, expuseram notáveis valores.

Entende-se que o método apresentou resultados promissores tanto no que diz respeito à detecção de clientes adimplentes (alto valor de sensibilidade), quanto à seleção de clientes inadimplentes (alto valor preditivo negativo). A consistência dos valores denotam a possibilidade do método em auxiliar no processo decisório de arrecadação das concessionárias. No entanto, é incumbência da empresa a definição de um limiar de decisão que traga segurança na utilização do sistema.

Ainda sobre os experimentos, constatou-se a eficácia da *CollabNet* na inserção de camadas ocultas em uma rede neural profunda. Ainda que não se tenha utilizado nenhum método de busca, realizando o processo manual de parametrização, os hiperparâmetros dispostos pela técnica foram suficientes para o aprendizado colaborativo entre as camadas. Esse entendimento foi obtido através do comparativo entre as curvas de aprendizado. Nessa etapa, foi mostrada a não interferência da inclusão de uma nova camada no erro global da rede.

Investigou-se, por meio do contraste com a validação cruzada, a influência de fatores como a sazonalidade no comportamento de pagamento e consumo dos clientes. Além disso, é considerada contribuição deste trabalho a aplicação de um método computacional no contexto da inadimplência em concessionárias de serviço público.

Por fim, é considerada de grande valia a aplicação de novas abordagens de aprendizado de máquina em domínios ainda pouco explorados pela literatura. Da mesma forma, é de suma importância aproximar as problemáticas da indústria com as técnicas desenvolvidas no âmbito acadêmico. Os resultados provenientes dessa união deverão ser capazes de resolver diversos problemas ainda recorrentes na economia global.

## 5.1 Trabalhos Futuros

Tendo em vista os resultados satisfatórios da aplicação da *CollabNet* no contexto da classificação de risco de inadimplência em concessionária de serviço público, encoraja-se a exploração de algumas oportunidades de melhorias tanto à nível da técnica aplicada, quanto do domínio explorado. Os pontos de melhoria são listados abaixo:

- Desenvolver um *software* que permita a utilização da metodologia em ambiente de produção. Além de permitir analistas utilizarem o resultado do modelo na tomada de decisão, é necessário que o *software* forneça a capacidade de treinamento contínuo. Como demonstrado, o comportamento do cliente possui forte sazonalidade o que pode afetar a performance de um modelo treinado com dados ultrapassados.
- Avaliar a metodologia em outras base de dados que compartilhem o mesmo contexto de serviços públicos ou de risco de crédito (*credit scoring*).
- Explorar outras características presentes em base de dados semelhantes, como aspectos geográficos, renda familiar e respostas às iniciativas de cobrança.
- Criar métodos para a aplicação de técnicas de otimização de hiperparâmetros nos parâmetros nativos da *CollabNet*, como o  $C$ ,  $\Delta C$  e *saltoC* além de propor algum método computacional para encontrar o momento ideal de inserção de uma nova camada oculta.
- Aplicar outras técnicas de aprendizagem computacional, como redes neurais recorrentes (RNN's) e *XGBoost*, bem como outras técnicas de amostragem de dados como o SMOTE (*Synthetic Minority Over-sampling Technique*).

# Referências

- ALMEIDA NETO, A. *Aplicações de Múltiplas Redes Neurais em Sistemas Mecatrônicos*. Tese (Doutorado) — Instituto Tecnológico de Aeronáutica, 2003. Citado na página 21.
- ALOM, M. Z. et al. A state-of-the-art survey on deep learning theory and architectures. *Electronics*, Multidisciplinary Digital Publishing Institute, v. 8, n. 3, p. 292, 2019. Citado 2 vezes nas páginas 24 e 25.
- ARAÚJO, R. V. d. Proposta de melhoria de efetividade de ações de cobrança através da aplicação de métodos quantitativos. Escola Politécnica, 2016. Citado 2 vezes nas páginas 13 e 15.
- BRAGA, A. d. P. *Redes neurais artificiais: teoria e aplicações*. [S.l.]: Livros Técnicos e Científicos, 2000. 59–61 p. Citado na página 26.
- CHEN, Y.-Q.; ZHANG, J.; NG, W. W. Loan default prediction using diversified sensitivity undersampling. In: IEEE. *2018 International Conference on Machine Learning and Cybernetics (ICMLC)*. [S.l.], 2018. v. 1, p. 240–245. Citado na página 16.
- CHEN, Z. et al. Autoencoder-based network anomaly detection. In: IEEE. *2018 Wireless Telecommunications Symposium (WTS)*. [S.l.], 2018. p. 1–5. Citado na página 16.
- CHISHTI, W. A.; AWAN, S. M. Deep neural network a step by step approach to classify credit card default customer. In: IEEE. *2019 International Conference on Innovative Computing (ICIC)*. [S.l.], 2019. p. 1–8. Citado na página 16.
- CLAESEN, M.; MOOR, B. D. Hyperparameter search in machine learning. *arXiv preprint arXiv:1502.02127*, 2015. Citado na página 37.
- COSTA, C. N. et al. Descoberta de conhecimento em bases de dados. *Revista Eletrônica: Faculdade Santos Dumont*, v. 2, 2016. Citado na página 19.
- CYBENKO, G. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, Springer, v. 2, n. 4, p. 303–314, 1989. Citado na página 25.
- DANTAS, E. R. G. et al. O uso da descoberta de conhecimento em base de dados para apoiar a tomada de decisões. *V Simpósio de Excelencia em Gestão e Tecnologia*, p. 1–10, 2008. Citado na página 18.
- DURAND, D. *Risk elements in consumer installment financing*. [S.l.]: National Bureau of Economic Research, New York, 1941. Citado na página 18.
- FAWCETT, T. An introduction to roc analysis. *Pattern recognition letters*, Elsevier, v. 27, n. 8, p. 861–874, 2006. Citado na página 39.
- FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. *AI magazine*, v. 17, n. 3, p. 37–37, 1996. Citado 2 vezes nas páginas 18 e 19.

- FAYYAD, U. M. et al. *Advances in knowledge discovery and data mining*. the MIT Press, 1996. Citado na página 18.
- FISHER, R. A. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, Wiley Online Library, v. 7, n. 2, p. 179–188, 1936. Citado na página 18.
- FORTI, M. *Técnicas de machine learning aplicadas na recuperação de crédito do mercado brasileiro*. Tese (Doutorado) — FGV, 2018. Citado na página 16.
- FREITAS JR., M. L. et al. Deep collabnet: Rede deep learning colaborativa. Universidade Federal do Maranhão, 2018. Citado 5 vezes nas páginas 14, 15, 20, 26 e 47.
- GOLDSCHMIDT, R.; PASSOS, E. *Data mining: um guia prático*. [S.l.]: Gulf Professional Publishing, 2005. Citado na página 19.
- GOODFELLOW, I. et al. Generative adversarial nets in advances in neural information processing systems (nips). 2014. Citado na página 14.
- GROTTI, D. A. M. A experiência brasileira nas concessões de serviço público. *Notadez*, p. 3, 2007. Citado na página 13.
- GUILHON, D. M. et al. Classificação de risco em transferências voluntárias federais utilizando xgboost. Universidade Federal do Maranhão, p. 22–28, 2020. Citado na página 20.
- GUIMARÃES, L. *Previsão de Inadimplência e Redes Neurais Artificiais*. Tese (Doutorado) — Dissertação de Mestrado, Programa de Pós-Graduação em Modelagem . . . , 2006. Citado na página 20.
- HASSAN, A. K. I.; ABRAHAM, A. Modeling consumer loan default prediction using ensemble neural networks. In: IEEE. *2013 INTERNATIONAL CONFERENCE ON COMPUTING, ELECTRICAL AND ELECTRONIC ENGINEERING (ICCEEE)*. [S.l.], 2013. p. 719–724. Citado na página 16.
- HAYKIN, S.; NETWORK, N. A comprehensive foundation. *Neural networks*, v. 2, n. 2004, p. 41, 2004. Citado 2 vezes nas páginas 23 e 24.
- HINTON, G. E.; SALAKHUTDINOV, R. R. Reducing the dimensionality of data with neural networks. *science*, American Association for the Advancement of Science, v. 313, n. 5786, p. 504–507, 2006. Citado na página 21.
- KOHAVI, R. et al. A study of cross-validation and bootstrap for accuracy estimation and model selection. In: MONTREAL, CANADA. *Ijcai*. [S.l.], 1995. v. 14, n. 2, p. 1137–1145. Citado na página 36.
- LAROCHELLE, H. et al. Exploring strategies for training deep neural networks. *Journal of machine learning research*, v. 10, n. Jan, p. 1–40, 2009. Citado na página 24.
- LEMO, E. P.; STEINER, M. T. A.; NIEVOLA, J. C. Análise de crédito bancário por meio de redes neurais e árvores de decisão: uma aplicação simples de data mining. *Revista de Administração-RAUSP*, Universidade de São Paulo, v. 40, n. 3, p. 225–234, 2005. Citado na página 14.

- LESSMANN, S. et al. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, Elsevier, v. 247, n. 1, p. 124–136, 2015. Citado na página 15.
- LEVENBERG, K. A method for the solution of certain non-linear problems in least squares. *Quarterly of applied mathematics*, v. 2, n. 2, p. 164–168, 1944. Citado na página 23.
- LOPES, R. G. Predição da recuperação da inadimplência em operações de crédito. 2017. Citado na página 15.
- MCCULLOCH, W. S.; PITTS, W. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, Springer, v. 5, n. 4, p. 115–133, 1943. Citado na página 20.
- MCGEE, L.; SCHMIDT, S.; SMITH, G. Application of statistical filter theory to the optimal estimation of position and velocity on board a circumlunar vehicle. *NASA Technical Report R-135, Tech. Rep*, 1962. Citado na página 23.
- PRASS, F. S. Kdd—uma visão geral do processo. *Recuperado em*, v. 15, 2016. Citado 2 vezes nas páginas 19 e 20.
- RIBEIRO, A. dos S. Recuperação dos clientes no sistema de cobrança de uma empresa de saneamento. In: SIMPEP. *XXII Simpósio de Engenharia de Produção*. [S.l.], 2015. Citado na página 13.
- ROSENBLATT, F. *The perceptron, a perceiving and recognizing automaton Project Para*. [S.l.]: Cornell Aeronautical Laboratory, 1957. Citado na página 20.
- RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. *nature*, Nature Publishing Group, v. 323, n. 6088, p. 533–536, 1986. Citado 2 vezes nas páginas 20 e 23.
- SCHMIDHUBER, J. Deep learning in neural networks: An overview. *Neural networks*, Elsevier, v. 61, p. 85–117, 2015. Citado na página 21.
- SINGH, D.; SINGH, B. Investigating the impact of data normalization on classification performance. *Applied Soft Computing*, Elsevier, v. 97, p. 105524, 2020. Citado na página 34.
- THOMAS, L. C. A survey of credit and behavioural scoring: forecasting financial risk of lending to consumers. *International journal of forecasting*, Elsevier, v. 16, n. 2, p. 149–172, 2000. Citado 2 vezes nas páginas 18 e 31.
- TRIVEDI, S. K. A study on credit scoring modeling with different feature selection and machine learning approaches. *Technology in Society*, Elsevier, v. 63, p. 101413, 2020. Citado na página 18.
- WANG, B.; GONG, N. Z. Stealing hyperparameters in machine learning. In: IEEE. *2018 IEEE Symposium on Security and Privacy (SP)*. [S.l.], 2018. p. 36–52. Citado na página 37.

---

ZAMINI, M.; MONTAZER, G. Credit card fraud detection using autoencoder based clustering. In: IEEE. *2018 9th International Symposium on Telecommunications (IST)*. [S.l.], 2018. p. 486–491. Citado na página 16.

ZHUANG, F. et al. Representation learning via semi-supervised autoencoder for multi-task learning. In: IEEE. *2015 IEEE International Conference on Data Mining*. [S.l.], 2015. p. 1141–1146. Citado na página 16.