Universidade de São Paulo
Escola Superior de Agricultura "Luiz de Queiroz"

Fraude de cartão de crédito: como a estatística e o machine learning se conversam

Gabriel Ferreira dos Santos Silva

Dissertação apresentada para obtenção do título de Mestre em Ciências. Área de concentração: Estatística e Experimentação Agronômica

Piracicaba
2020

Gabriel Ferreira dos Santos Silva
Bacharel em Ciências Econômicas

Fraude de cartão de crédito: como a estatística e o machine learning se conversam

Orientador:
Prof. Dr. **GABRIEL ADRIÁN SARRIÉS**

Dissertação apresentada para obtenção do título de Mestre em Ciências. Área de concentração: Estatística e Experimentação Agronômica

Piracicaba
2020

Dados Internacionais de Catalogação na Publicação
DIVISÃO DE BIBLIOTECA - DIBD/ESALQ/USP

Silva, Gabriel Ferreira dos Santos
Fraude de cartão de crédito: como a estatística e o machine learning se conversam/ Gabriel Ferreira dos Santos Silva. – – Piracicaba, 2020.
96 p.

Dissertação (Mestrado) – – USP / Escola Superior de Agricultura "Luiz de Queiroz".

1. Algoritmos 2. Estatística 3. Revisão bibliográfica sistemática 4. Fraude I. Título

AGRADECIMENTOS

Agradeço a Deus pela vida, por toda a força durante a jornada. Por, mesmo diante de minhas fraquezas, estar comigo a todo momento.

Aos professores e funcionários do Departamento de Ciências Exatas da Escola Superior de Agricultura "Luiz de Queiroz", pelos ensinamentos, orientações e paciência. Ao Professor César Gonçalves de Lima, por todo suporte burocrático no início do mestrado. Ao Serviço de Pós-Graduação da ESALQ, pelas constantes prontidão e solicitude no atendimento.

Ao Professor Gabriel Adrián Sarriés, por orientar-me academicamente, desde a graduação. Agradeço pela amizade, preocupação e confiança. Ao Gustavo Furlan e Yuniel Tejada, pelo suporte nas análises de dados e no desenho metodológico.

À minha família: meus pais, Adriana e José Mauro, pelo apoio e incentivo de sempre. Aos meus avós, Maria e José, pelos conselhos, cuidados e presença constante. Aos tios, Paula e André, pela preocupação e afeto. À minha irmã, Manuela, que, mesmo sem entender ao certo o que tudo isso se trata, é fonte de inspiração. Agradeço por compreender a distância e por sempre me receber com um abraço cheio de carinho quando nos encontramos. Ao Luciano Fernandes pelas conversas e preocupação.

À Janaina, minha namorada, por segurar a barra tantas vezes. Por compreender as ausências necessárias, por estar ao meu lado nos momentos difíceis, por não ter abandonado o barco. Obrigado pelas incentivos, pelo amor, carinho, cuidado, companheirismo, risadas. Agradeço por estar comigo durante todo este processo de crescimento e amadurecimento. A seus pais, Sueli e Antônio, pelo suporte e cuidado durante todos estes anos.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela concessão da bolsa de estudos.

A todos que, direta ou indiretamente, contribuíram para mais esta realização, registro o meu "Muito Obrigado".

EPÍGRAFE

Senhor, dai-me força para mudar o que pode ser mudado.
Resignação para aceitar o que não pode ser mudado.
E sabedoria para distinguir uma coisa da outra

São Francisco de Assis

SUMÁRIO

RESUMO

Fraude de cartão de crédito: como a estatística e o machine learning se conversam

A fraude de cartão de crédito é um mecanismo financeiro responsável por movimentar ilegalmente bilhões de dólares todos os anos, com implicações onerosas para clientes, redes de cartão de crédito e seguradoras. Diante disso, buscar mecanismos que permitam captar situações fraudulentas é um importante movimento para que se possa reduzir os prejuízos tangíveis e intangíveis causados no setor. Dois destes mecanismos são a Estatística e o Machine Learning, áreas que, embora muitas vezes consideradas excludentes, apresentam um potencial sinérgico capaz de oferecer resultados mais consistentes. Nesse sentido, este trabalho buscou explorar tal potencial a partir da utilização de um conjunto de modelos estatísticas e algoritmos de Machine Learning. Para tanto, foi realizada uma Revisão Bibliográfica Sistemática acerca da utilização de algoritmos de Machine Learning em situações de fraude de cartão de crédito, com base nos artigos disponíveis no portal Web of Science e publicados no período de 2008 a 2018. Em seguida, a partir de um conjunto de dados sobre fraude de cartão de crédito disponível na plataforma Kaggle, foram sorteados sete grupos amostrais, por meio de amostragem progressiva, respeitando o balanceamento dos dados, com 30, 60, 120, 180, 360, 540 e 984 observações. Para as análises estatísticas, os conjuntos foram submetidos à NP-ANOVA, NP-MANOVA, Regressão Logarítmica, Análises Discriminantes Linear e Quadrática, Mínimos Quadrados Parciais e à Regressão Logística. Para as classificações por meio do Machine Learning, foram utilizados os algoritmos Naive Bayes, Random Forest, Multilayer Perceptron, CART, K-NN e Support Vector Machine. Os resultados revelaram dois movimentos principais: com base na revisão bibliográfica realizada, foi observado que os principais algoritmos utilizados na detecção de fraude de cartão de crédito são a Regressão Logística, Random Forest, Multilayer Perceptron, Naive Bayes, C4.5 e Redes Neurais. Os melhores desempenhos, considerando o levantamento bibliográfico, foram obtidos através dos algoritmos Naive Bayes, Random Forest e Random Tree. Além disso, a aplicação dos algoritmos e dos métodos estatísticos neste trabalho confirmou que o Random Forest é uma das ferramentas mais eficazes para este tipo de classificação, com bons índices de taxa de acerto e taxa de falsos positivos. Para a taxa de falsos negativos, no entanto, a Regressão Logística, método tradicionalmente estatístico, apresentou melhor desempenho, de modo que, para a busca de um cenário mais assertivo, recomenda-se a utilização de ambos os métodos.

Palavras-chave: Algoritmos, Estatística, Revisão bibliográfica sistemática, Fraude

ABSTRACT

Credit card fraud: how statistics and machine learning talk

The credit card fraud is a illegal financial mechanism responsible for moving billions of dollars each year, with costly implications for customers, credit card companies and insurance companies. That said, looking for mechanisms that allow fraudulent situations to be captured is an important move to reduce the tangible and intangible caused by this illegal action. Two of these mechanisms are Statistics and Machine Learning, areas that, although often considered exclusive, have a synergistic potential capable of offering more consistent results. In this sense, this work sought to explore this potential from the use of a set of statistical models and Machine Learning algorithms. For that, a Systematic Bibliographic Review was carried out, considering the use of Machine Learning algorithms in situations of credit card fraud, based on the articles available on the Web of Science portal and published in the period from 2008 to 2018. Then, from a set of data on credit card fraud available on the Kaggle platform, seven sample groups were drawn by means of progressive sampling, respecting the balance of the data, with 30, 60, 120, 180, 360, 540 and 984 observations. For statistical analysis, the subsets were submitted to NP-ANOVA, NP-MANOVA, Logarithmic Regression, Discriminating Linear and Quadratic Analysis, Partial Least Squares and Logistic Regression. For classifications through Machine Learning, the following algorithms were used: Naive Bayes, Random Forest, Multilayer Perceptron, CART, K-NN and Support Vector Machine. The results showed two main movements: based on the bibliographic review, it was observed that the main algorithms used to detect credit card fraud are Logistic Regression, Random Forest, Multilayer Perceptron, Naive Bayes, C4.5 and Neural Networks. The best performances, considering the bibliographic review, were obtained through the Naive Bayes, Random Forest and Random Tree algorithms. In addition, the application of algorithms and statistical methods in this work confirmed that Random Forest is one of the most effective tools for this type of classification, with good rates of accuracy and false positive rates. For false negatives, however, Logistic Regression, traditionally a statistical method, performed better. This way, based on this work, a guide for a more assertive performance is to use both methods together.

Keywords: Algorithms, Statistics, Systematic review, Fraud

Capítulo 1

MACHINE LEARNING E FRAUDE DE CARTÃO DE CRÉDITO: UMA REVISÃO BIBLIOGRÁFICA SISTEMÁTICA

RESUMO

A Inteligência Artificial (IA) está cada vez mais presente na vida humana, indicando um futuro integrado entre o real e o virtual. Inserido como um campo da IA, o Machine Learning opera com um conjunto de ferramentas que permitem a execução de atividades nas mais diversas áreas do conhecimento. Uma das aplicações dos algoritmos é na detecção de fraudes de cartão de crédito, que causam perdas bilionárias para instituições financeiras e para a sociedade. Neste sentido, este trabalho se propôs a identificar de que forma o Machine Learning tem contribuído para este problema, que causa danos financeiros e não financeiros. Para tanto, foi utilizada a Revisão Bibliográfica Sistemática, por meio da qual obteve-se diagnósticos primários sobre as áreas do Machine Learning, Fraude e Fraude de Cartão de Crédito. Como fonte de pesquisa, foi selecionado o portal Web of Science, com buscas centradas no período de 2008 a 2018. Os resultados apontam que, tanto no tema de Machine Learning, quanto no de fraude, grande parte dos trabalhos referem-se às áreas de Medicina & Biomedicina, existindo, nesse último, aplicações na área financeira. Na integração entre Machine Learning e fraude de cartão de crédito, foram encontrados cinco trabalhos que, ao serem avaliados integralmente, revelaram-se com um bom poder de detecção de fraudes, ainda que não tenham incorporado discussões como sobreajuste dos classificadores.

Palavras-chave: Inteligência Artificial, Algoritmos, Detecção.

1.1 Introdução

O Machine Learning ou Aprendizado de Máquina é um campo da Inteligência Artificial que estuda o reconhecimento de padrões, com o objetivo de classificação e/ou predição de determinados comportamentos, com base em um conjunto de dados. Em meados da década de 2000, o Machine Learning passou a ser cada vez mais difundido, de modo a se tornar, na década seguinte, uma ferramenta capaz de auxiliar desde em atividades mais simples, como a filtragem AntiSpam de e-mails, até em questões de reconhecimento facial (Bartlett *et al.*, 2005), diagnóstico de doenças (Kourou *et al.*, 2015), segurança de dados na internet (Domingos, 2012), Ecologia (Trivellato *et al.*, 2020), rastreabilidade de alimentos (Fernandes *et al.*, 2020) e (Mazola *et al.*, 2019), entre outros.

O Machine Learning trabalha com duas formas básicas de aprendizagem: a supervisionada e a não supervisionada. Na aprendizagem supervisionada, é dada uma predefinição para o programa, ou

seja, o cientista indica que determinado padrão se refere a determinado comportamento, de modo que o algoritmo saiba reconhecer os novos dados e classificá-los de acordo com o que foi supervisionado. Na aprendizagem não supervisionada, o programa não possui conhecimento inicial sobre os dados, podendo encontrar padrões por si só, ou seja, não há uma supervisão prévia que ligue um padrão a um determinado comportamento (Alpaydin, 2009).

Um campo de aplicação em que o aprendizado supervisionado tem sido utilizado é o da detecção de fraudes, situações em que pessoas, cenários e/ou organizações são manipuladas, com o intuito de se obter vantagem ilícita, em prol de benefício próprio, seja ele individual ou corporativo. Existe uma extensa aplicabilidade do tema fraude: na ciência, por exemplo, pode ocorrer por meio de plágio ou manipulação de resultados; no ambiente empresarial, se configura através de alteração de balanços contábeis e apropriações indevidas; no governo, a partir desvios de verba e propinas; e na sociedade, desde simples ações, como falsas pirâmides e golpes financeiros.

Um tipo de fraude que tem crescido nos últimos anos é o golpe de cartão de crédito. Até as décadas passadas, os cartões eram clonados através de leitores óticos adulterados, que transmitiam informações aos fraudadores. Nos dias atuais, no entanto, a exposição maior está no ambiente virtual, devido ao intenso volume de transações comerciais e pagamentos online. Segundo estudo realizado pela consultoria Nilson Report (2016), no ano de 2015, os valores perdidos com a fraude de cartão de crédito atingiram, mundialmente, 21,84 bilhões de dólares, o que representa, em Reais, R$ 85,26 bilhões, valor superior ao Produto Interno Bruto de cerca de metade dos países do mundo (Fundo Monetário Internacional, 2019).

Por esta razão, o desenvolvimento de mecanismos que permitam a identificação destes cenários fraudulentos é essencial, ao passo que possibilita o direcionamento dos esforços para o combate destes golpes. No entanto, para que se possa avançar, é fundamental entender quais são os mecanismos que a literatura já dispõe, ou seja, o que se sabe sobre o papel do Machine Learning na detecção de fraude de cartão de crédito. Além disso, avanços nesta área de estudo são importantes tanto para a sociedade, que é diretamente atingida pelos golpes, quanto para as empresas de cartão de crédito, ao passo que desejam oferecer aos seus clientes uma maior segurança nas transações comerciais e reduzir as perdas ocasionadas pelas fraudes.

Nesse sentido, o objetivo deste trabalho é investigar quais são as ferramentas atuais utilizadas na detecção de fraude de cartão de crédito, avaliando suas assertividades por meio de análise bibliográfica sistemática dos trabalhos publicados no Brasil e no mundo, entre os anos de 2008 e 2018.

1.2 Material e Métodos

1.2.1 Revisão Bibliográfica Sistemática

A metodologia utilizada neste estudo foi a Revisão Bibliográfica Sistemática (RBS), ferramenta que permite uma análise criteriosa da literatura, caracterizando quantitativa e qualitativamente o desenvolvimento científico de determinada área, e identificando, conforme Conforto *et al.* (2011), seu "estado da arte", ou seja, aquilo que se tem de mais recente, inovador e eficaz em relação ao assunto destacado.

A RBS carrega o caráter sistemático pelo fato de exigir um roteiro de execução. Não se trata de uma revisão bibliográfica narrativa, onde se apresenta uma descrição, geralmente histórica, sobre determinado tema, destacando-se artigos relevantes ao longo do tempo. Ao adotar-se um método criterioso de revisão bibliográfica, é possível reduzir vieses, aumentar a confiabilidade, analisar o comportamento e identificar tendências do tema investigado (de Medeiros *et al.*, 2015).

Neste sentido, para a execução da RBS foi adotado o procedimento sugerido por Conforto *et al.* (2011), a partir do qual foram realizadas algumas adaptações para o cumprimento dos objetivos deste trabalho. A Tabela 1.1 apresenta as etapas seguidas durante a execução do processo metodológico.

1.2.1.1 Etapa de Entrada

Com base no conjunto de etapas proposto por Conforto *et al.* (2011), a estrutura metodológica foi definida da seguinte forma:

- Definição do Problema: O que tem sido discutido academicamente na área de intersecção entre o Machine Learning e a detecção de fraude de cartão crédito, no período de 2008 a 2018?
- Levantamento de fontes primárias: As buscas foram realizadas na base Web of Science (2019), com artigos de livre acesso, brasileiros ou internacionais, no período de 2008 a 2018. A seleção de trabalhos exclusivamente de livre acesso se deu não só pela limitação da quantidade de artigos, mas também pela necessidade ocasional de consulta da íntegra dos trabalhos.
- Definição das *strings* de busca: as *strings* de busca foram definidas de acordo com o diagrama representado na Figura 1.1.

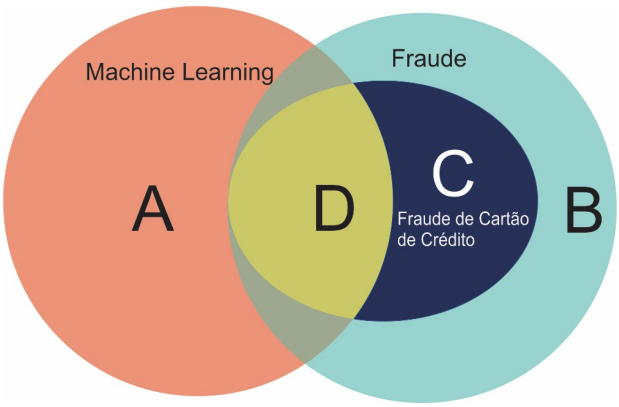


Figura 1.1: Regiões de buscas utilizadas como base para as *strings* de busca.

Tabela 1.1: Etapas do processo de Revisão Bibliográfica Sistemática.

Etapa	Descrição
1. Entrada	
1.1. Definição do Problema	Elaboração das questões que se pretende responder com a RBS.
1.2. Levantamento de fontes primárias	Materiais em que os trabalhos podem ser consultados.
1.3. Definição das strings de busca	As strings são os termos que dão base para a realização das buscas, ou seja, nesta etapa são definidas as palavras ou os conjuntos de palavras a serem utilizados para a captação dos materiais.
1.4. Definição dos critérios de inclusão	Os critérios de inclusão oferecem insumos para a filtragem prévia dos materiais que serão ou não analisados. A sua definição deve levar em conta os problemas a serem respondidos.
1.5. Definição dos critérios de qualificação	Critérios de qualificação são os componentes a serem avaliados dentro do material, como título, resumo, método de pesquisa, citações, ou seja, é uma forma de classificar a importância do trabalho para a literatura.
1.6. Definição dos métodos e ferramentas de análise	Nesta etapa, é definido de que forma as buscas são realizadas, quais são os filtros de busca e de leitura e como os resultados são armazenados.
2. Processamento	
2.1. Realização das buscas	As buscas dos materiais são realizadas nos portais selecionados.
2.2. Leitura e Análise dos Resultados	Etapa realizada de acordo com os filtros de leitura estabelecidos na etapa 1.6.
2.3. Documentação e arquivamento dos artigos selecionados	Na documentação, apresenta-se as quantidades de artigos analisados, quantos destes foram selecionados e quantos foram excluídos. No arquivamento, os artigos selecionados são salvos e registrados, com descrições sintéticas de seu conteúdo.
3. Saída	
3.1. Síntese e Resultados	Elaboração de um relatório sintético que apresentará o conteúdo estudado, onde são apresentadas as percepções e as conclusões sobre o nível de desenvolvimento científico do assunto em questão. Conforme Conforto *et al.* (2011), é importante destacar aspectos como principais autores, quantidade de artigos, métodos teóricos utilizados, evolução da conceituação, entre outros.

Desta forma, para atender os campos destacados no diagrama, as *strings* foram estabelecidas conforme a Tabela 1.2.

- Definição dos critérios de inclusão: para a análise das regiões A e B, foram avaliados artigos com livre acesso e no mínimo cinco citações, com o intuito de caracterizar os respectivos campos de estudos. Para a região C, devido à restrição numérica de trabalhos encontrados, avaliaram-se

Tabela 1.2: Definição das *strings* de buscas.

Região	String
A	"Machine Learning"
B	"Fraud" OR "Fraude"
C	"Fraude de Cartão de Crédito" OR "Credit Card Fraud"
D	"Machine Learning" AND "Fraude de Cartão de Crédito" OR "Machine Learning" AND "Credit Card Fraud"

todos, independentemente do número de citações. Na região D, foram incluídos na análise apenas os artigos que apresentaram a aplicação de pelo menos um algoritmo de Machine Learning para a detecção de cenários de fraude de cartão de crédito.

- Definição dos critérios de qualificação: nas regiões A, B e C, foram avaliados os títulos, os resumos e as palavras chaves dos artigos. Na região D, os artigos foram avaliados integralmente.

- Definição dos métodos e ferramentas de análise: as buscas foram realizadas na base Web of Science, no período definido de janeiro de 2008 a dezembro de 2018. As *strings* de pesquisa foram utilizadas seguindo os critérios da Tabela 1.2. Para as regiões A, B e C, avaliaram-se parâmetros como área de aplicação, número de citações (exceto na região C), quantidade de artigos por área e evolução do número de artigos por ano. Para a região D, foram avaliadas questões metodológicas e os resultados de cada artigo. Os filtros realizados a cada região estão dispostos na Figura 1.3.

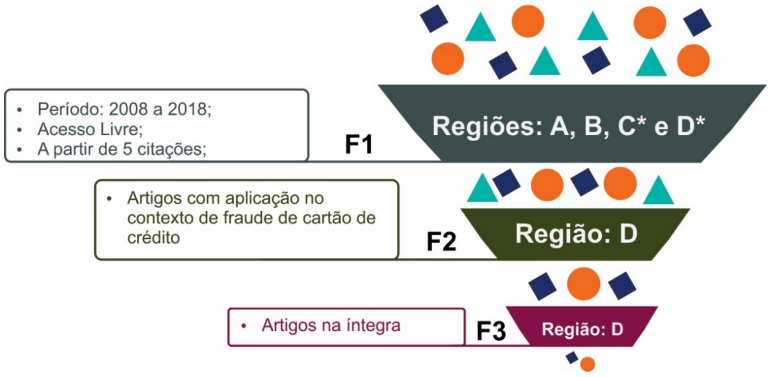


Figura 1.2: Filtros estabelecidos para a seleção dos artigos.
(*) Devido ao baixo volume de trabalhos encontrados, não foram estabelecidos filtros de citações nas regiões C e D.

1.2.1.2 Etapa de Processamento

Com base na composição da Tabela 1.1, a etapa de processamento foi elaborada de acordo com os seguintes critérios:

- Realização das buscas: as buscas foram realizadas no portal da base de dados Web of Science (Figura 1.3), levando-se em conta os atributos definidos, como as strings e o período de análise.

Ressalta-se que a busca considerou apenas artigos com acesso aberto, diante da disponibilidade de consulta da íntegra dos materiais.

Figura 1.3: Interface do portal de buscas Web of Science.

- Leitura e Análise dos Resultados: a partir dos filtros estabelecidos previamente, foram realizadas as leituras. Parcialmente, 9.346 artigos foram analisados, distribuídos nas regiões A, B, C e D. Integralmente, analisaram-se cinco artigos, que utilizaram algoritmos de Machine Learning para a detecção de fraude de cartão de crédito.
- Documentação e arquivamento dos artigos selecionados: os artigos foram todos documentados, em planilha de Excel, conforme extração do Web of Science. Os artigos analisados integralmente foram arquivados e salvos em pasta específica.

1.2.1.3 Etapa de Saída

Por fim, a etapa de saída foi estruturada da seguinte maneira:

- Síntese e Resultados: a síntese de resultados foi realizada, contemplando a interpretação da leitura dos artigos e dos gráficos elaborados. Nas regiões A, B e C, as informações analisadas referem-se ao conteúdo dos títulos, por meio de mineração de texto, expressa a partir de "wordclouds" elaboradas com auxílio do software R/RStudio e do pacote adicional "wordcloud". Os campos de aplicação foram separados, incialmente, entre Ciências Naturais, Exatas e Humanas. Em seguida, as informações foram desagregadas por áreas temáticas, como Medicina & Biomedicina, Química & Biologia e Engenharia & Robótica. Analisaram-se, a partir disso, a evolução das publicações por área durante o período entre 2008 e 2018 e os principais meios de publicação dos trabalhos, como revistas, jornais, periódicos, em geral. Na região D, foram avaliadas as publicações por ano, os algoritmos e as abordagens utilizadas, a descrição dos bancos de dados, as variáveis disponíveis e os algoritmos de melhor desempenho entre os cinco artigos. As análises foram dispostas de forma condensada na seção de Resultados e de forma individual no Apêndice, seguindo a estrutura da Tabela 1.3.

1.3 Resultados e Discussão

A estrutura metodológica proposta permitiu a obtenção dos resultados em diferentes níveis, seguindo as regiões apresentadas na Figura 1.1. Desta forma, esta seção divide-se em quatro subseções,

Tabela 1.3: Estrutura da disposição individual dos resultados.

Critérios	Descrição
Título; Autores; Ano de Publicação	
Meio de Publicação	
Objetivos	
Banco de Dados Utilizados	
Variáveis Disponíveis	
Metodologia Proposta	
Assertividade dos Algoritmos	

que possibilitaram traçar um panorama geral das áreas de Machine Learning, Fraude, Fraude de Cartão de Crédito e Machine Learning & Fraude de Cartão de Crédito. As buscas foram realizadas seguindo-se as strings estabelecidas na Tabela 1.3.

1.3.1 Machine Learning

Ao se pesquisar pelo tema Machine Learning no portal Web of Science, obteve-se, no período de 2008 a 2018, um retorno de 95.304 resultados, distribuídos em diversos campos de conhecimento. Passando-se pelo processo de filtragem F1, do total de trabalhos encontrados, 20.102 são de acesso aberto. Destes, 16.011 são classificados como artigos científicos. Aplicando-se o filtro de cinco ou mais citações, restaram 8.943 artigos. A primeira classificação destes trabalhos foi realizada com base na grande área de conhecimento. Conforme a Figura 1.4, é possível perceber que, para o período estabelecido, o tema Machine Learning foi amplamente aplicado no campo das Ciências Naturais, com cerca de 75% das publicações. Em seguida, aparecem as Ciências Exatas, com 22% e as Ciências Humanas, com 3%.

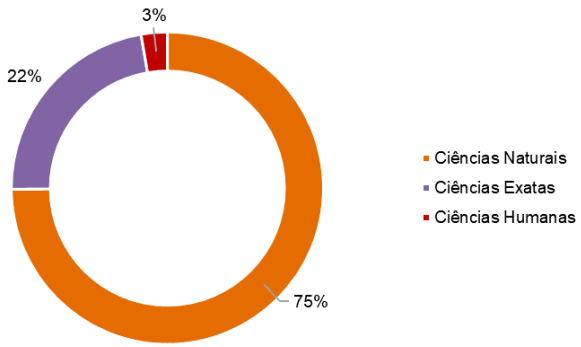


Figura 1.4: Distribuição das aplicações do tema Machine Learning por campo de conhecimento, de 2008 a 2018, com base no portal Web of Science.

Ao avaliar-se a evolução de cada campo ao longo do período estabelecido, é possível inferir, a partir da Figura 1.5, que as aplicações em Ciências Humanas se mantiveram relativamente estáveis, enquanto nas Ciências Exatas e Naturais houve aumento considerável no número publicações. Destaca-se, no entanto, que os três campos apresentaram uma queda no ano de 2018, fato que está associado à filtragem realizada, que selecionou artigos com cinco ou mais citações. Muitos artigos do ano de 2018

não atingiram o índice mínimo para sua inclusão na análise, porém isso se deve ao aspecto temporal, e não a uma questão de queda de publicações.

Outra inferência é o distanciamento que o campo das Ciências Naturais tem apresentado em relação aos demais. No ano de 2008, os três campos estavam relativamente próximos, pelo fato do Machine Learning ser um tema novo, à época. Já no ano de 2017, com os algoritmos mais popularizados, observa-se uma disparidade entre os campos, demonstrando como o Machine Learning se tornou familiar com maior velocidade nas Ciências Naturais.

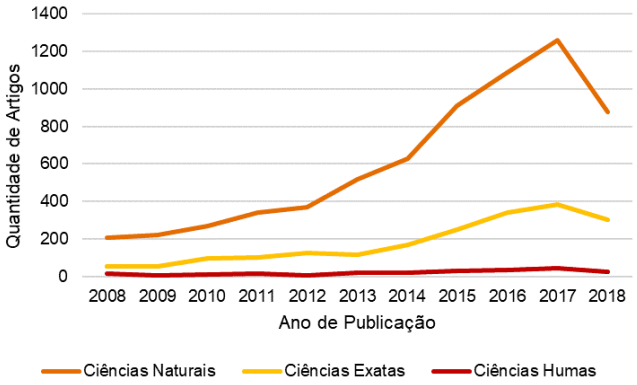


Figura 1.5: Evolução da quantidade de publicações com aplicação de Machine Learning por campo de conhecimento, de 2008 a 2018, com base no portal Web of Science.

A partir da Tabela 1.4, é possível obter o desmembramento dos três campos apresentados. Cerca de 38% do total dos artigos estão aplicados na área de Medicina & Biomedicina, enquanto que 30% estão concentrados na área Química & Biologia, demonstrando como o Machine Learning tem sido explorado nessas duas áreas. Em seguida, com cerca de 5,6%, aparecem Engenharia & Robótica. Ressalta-se, no entanto, a discrepância entre o segundo e o terceiro colocado, em torno de 24 pontos percentuais. Física & Astronomia aparecem em quarto lugar, com 4% de participação, seguidas por Data Science, Tecnologia, Agricultura e Psicologia & Comportamento Humano. A área com menor participação foi a Gestão Pública, com apenas 4 artigos publicados, equivalente a 0,04%. Além disso, outros 4 artigos não apresentaram informações suficientes para sua classificação.

Com base na ferramenta wordcloud (Figura 1.6), é possível observar que palavras como “cancer”, “disease”, “human”, “clinical” e “diagnosis” são as mais frequentes nos títulos dos artigos, demonstrando, novamente, a supremacia das áreas da Medicina & Biomedicina. Verifica-se, também, que o Machine Learning tem sido utilizado como ferramenta para a detecção de doenças, como o câncer, além de outras identificadas nos trabalhos, como Parkinson, Alzheimer e Demência. Por meio das palavras “gene” e “protein”, há indícios de que outro uso aplicado do Machine Learning é na área da genética e microbiologia.

Por fim, a partir dos quinze meios com mais publicações (Tabela 1.5), verifica-se, novamente, a prevalência de revistas voltadas para área de Medicina, Ciências e Biologia. Destaca-se, em contrapartida, a presença de periódicos como Sensors, com publicações nas áreas de Engenharia, Física e Tecnologia, Remote Sensing, que trabalha com Agricultura, Geografia, Biologia e Engenharia, e IEEE Access, com foco principal na área de Engenharia.

Tabela 1.4: Quantidade de artigos publicados com aplicações de Machine Learning por área de conhecimento, de 2008 a 2018, com base no portal Web of Science.

Área de Conhecimento	Quantidade de Artigos	Participação (%)
Medicina & Biomedicina	3380	37,79%
Química & Biologia	2650	29,63%
Engenharia & Robótica	500	5,59%
Física & Astronomia	355	3,97%
Data Science	354	3,96%
Tecnologia	339	3,79%
Agricultura & Pecuária	331	3,70%
Psicologia & Comportamento Humano	192	2,15%
Matemática & Estatística	169	1,89%
Geografia & Meteorologia	136	1,52%
Tecnologia da Informação	129	1,44%
Linguística	100	1,12%
Farmácia	95	1,06%
Economia & Administração	83	0,93%
Educação Física & Esporte	47	0,53%
Transporte & Logística	20	0,22%
Educação & Ciência	19	0,21%
Música	16	0,18%
Direito	13	0,15%
Comunicação & Jornalismo	7	0,08%
Gestão Pública	4	0,04%
Não Classificado	4	0,04%
Total	8943	100,00%

Tabela 1.5: Quantidade de artigos publicados com aplicações de Machine Learning por meio de publicação, de 2008 a 2018, com base no portal Web of Science.

Meio de Publicação	Quantidade de Artigos
Plos One	514
Bmc Bioinformatics	336
Bioinformatics	282
Scientific Reports	226
Sensors	159
Journal Of Biomedical Informatics	144
Remote Sensing	133
Ieee Access	129
Proceedings Of The National Academy Of Sciences Of The USA	112
Plos Computational Biology	110
Journal Of The American Medical Informatics Association	107
Neuroimage	92
Bmc Genomics	84
Machine Learning	82
Nucleic Acids Research	77

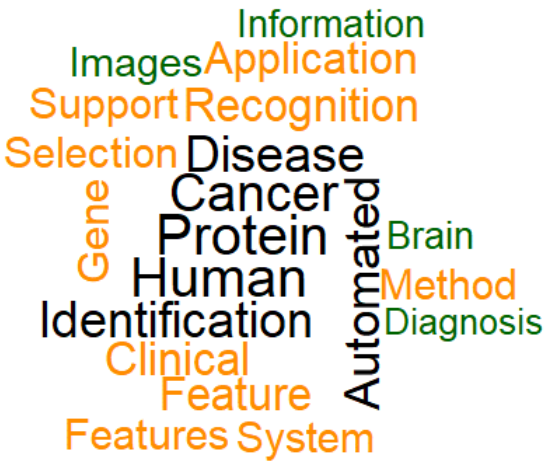


Figura 1.6: Wordcloud dos títulos dos artigos com aplicação de Machine Learning, no período de 2008 a 2018, com base no portal Web of Science.

1.3.2 Fraude

Em termos de fraude, a pesquisa, realizada nos parâmetros estabelecidos, apresentou 8.953 resultados, dos quais 1.506 são de acesso aberto. Deste total, 1.101 são classificados como artigos científicos. Ao realizar a filtragem das citações, restaram 362 artigos para análise.

A análise revelou que, mesmo quando o tema é "fraude", as Ciências Naturais ainda levam vantagem na quantidade de publicações, sendo contempladas com 43% do total de artigos. Em relação ao Machine Learning, há uma troca de posição nos demais campos. As Ciências Humanas aparecem em segundo lugar, com 33%, enquanto as Ciências Exatas foram responsáveis por 24% das publicações (Figura 1.7).

Em termos da evolução das publicações, os três campos apresentaram comportamento oscilante, porém crescente ao longo do tempo. Assim como no caso do Machine Learning, no ano de 2008 os campos estavam próximos em número de publicações. No decorrer do período, é possível observar vantagem das Ciências Naturais, porém em menor grau de discrepância. Por fim, nos anos de 2017 e 2018, observa-se um comportamento de queda, mas fruto de dois fatores: a oscilação natural presente na série (2017) e o filtro de citações que retirou considerável quantidade de artigos, pelo curto tempo para que o trabalho seja citado (Figura 1.8).

De modo similar ao comportamento observado no tema Machine Learning, as áreas Medicina & Biomedicina e Química & Biologia lideram o número de publicações relacionadas à fraude, sendo responsáveis, juntas, por cerca de 38,4% do total de artigos. Em seguida, Tecnologia da Informação e Economia & Administração aparecem, ambas, com 12,15%.

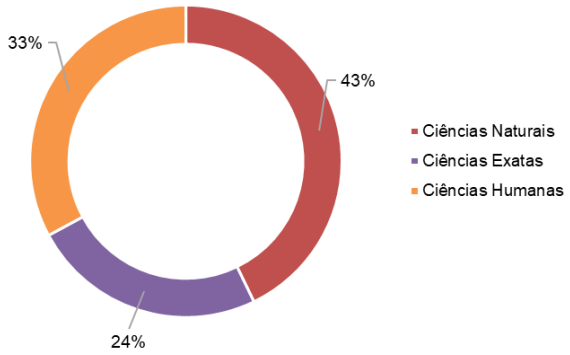


Figura 1.7: Distribuição das aplicações do tema Fraude por campo de conhecimento, de 2008 a 2018, com base no portal Web of Science.

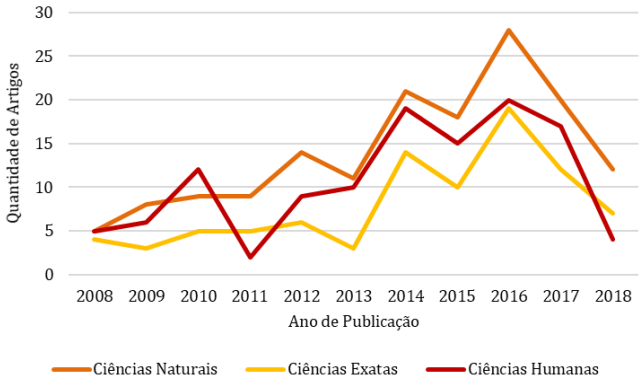


Figura 1.8: Evolução da quantidade de publicações com abordagens relativas a fraudes, por campo de conhecimento, de 2008 a 2018, com base no portal Web of Science.

Por meio da wordcloud apresentada na Figura 1.9, destaca-se, novamente, a presença de palavras relativas à Medicina & Biomedicina, como “health”, “medical”. Há, no entanto, palavras como “food”, “meat” e “olive”, relativos ao eixo da Agricultura & Pecuária. Ressalta-se, também, a presença da palavra “financial”, remetendo a questões de fraude financeira, grande grupo da fraude de cartão de crédito.

Ao avaliar-se os meios de publicação dos artigos, observa-se, por meio da Tabela 1.7, que assim como no tema Machine Learning, a revista Plos One é a que mais possui materiais disponíveis, com 20 publicações. Em seguida, destaca-se a presença da Food Chemistry, com 9 artigos. De modo geral, ressalta-se que as cinco primeiras revistas com mais publicações estão relacionadas à Medicina & Biomedicina, Agricultura & Pecuária e Química & Biologia, reforçando que o estudo de fraudes pode ser aplicado em diversas áreas do conhecimento.

1.3.3 Fraude de Cartão de Crédito

Em relação ao tema “fraude de cartão de crédito”, a pesquisa inicial apresentou 381 resultados, dos quais apenas 29 eram de acesso aberto. Diante do número restrito de trabalhos, não foi realizada a filtragem subsequente. Portanto, os 29 artigos foram analisados. Conforme a Tabela 1.8, destes 29 trabalhos, 23 referem-se ao tema “Cartão de Crédito”, o que representa cerca de 79% do total de publi-

Tabela 1.6: Quantidade de artigos publicados com abordagens relativas a fraudes por área de conhecimento, de 2008 a 2018, com base no portal Web of Science.

Área de Conhecimento	Quantidade de Artigos	Participação (%)
Medicina & Biomedicina	75	20,72%
Química & Biologia	64	17,68%
Tecnologia da Informação	44	12,15%
Economia & Administração	44	12,15%
Engenharia & Robótica	30	8,29%
Direito	23	6,35%
Gestão Pública	18	4,97%
Agricultura & Pecuária	13	3,59%
Sociologia	12	3,31%
Educação & Ciência	12	3,31%
Matemática & Estatística	9	2,49%
História & Filosofia	4	1,10%
Física & Astronomia	3	0,83%
Geografia	3	0,83%
Outros	8	2,21%
Total	362	100,00%

Tabela 1.7: Quantidade de artigos publicados com a temática de Fraude por meio de publicação, de 2008 a 2018, com base no portal Web of Science.

Meio de Publicação	Quantidade de Artigos
Plos One	20
Food Chemistry	9
Nature	8
Proceedings Of The National Academy Of Sciences Of The USA	7
Food Control	7
Journal Of Dairy Science	5
Ieee Access	5
Peerj	4
Scientific Reports	4
Political Analysis	4
Anaesthesia	4
Sensors	4
Journal Of Medical Internet Research	4
Journal Of Business Ethics	4
Plos Medicine	3

Figura 1.9: Wordcloud dos títulos dos artigos com aplicações do tema Fraude, no período de 2008 a 2018, com base no portal Web of Science.

Tabela 1.8: Quantidade de artigos publicados com a temática de Fraude de Cartão de Crédito, por assunto de aplicação, de 2008 a 2018, com base no portal Web of Science.

Tópico	Quantidade de Artigos	Participação (%)
Cartão de Crédito	23	79,31%
Cartão de Presente	1	3,45%
Cheques	1	3,45%
Dados Espaço-temporais	1	3,45%
Finanças	1	3,45%
Passagens Aéreas	1	3,45%
Cigarro	1	3,45%
Total	29	100,00%

cações. Além disso, os temas “Cartão de Presente”, “Cheques”, “Dados Espaço-Temporais”, “Finanças”, “Passagens Aéreas” e “Cigarro” completam os tópicos abordados, com uma publicação cada.

Considerando apenas os trabalhos publicados com a temática “cartão de crédito”, observa-se, a partir do gráfico apresentado na Figura 1.10 , a crescente produção de materiais publicados. No ano de 2008, apenas um trabalho foi publicado, enquanto que, em 2018, o número chegou a sete. Neste caso, diferentemente do comportamento observado com as strings “Machine Learning” e “Fraude/Fraud”, não se observa a queda no último ano do período, pelo fato do filtro de citações não ter sido aplicado.

Ao se avaliar as revistas em que os trabalhos foram publicados, não foi identificada qualquer convergência. As publicações foram feitas em diversos canais, conforme exposto pela Tabela 1.9, ou seja,

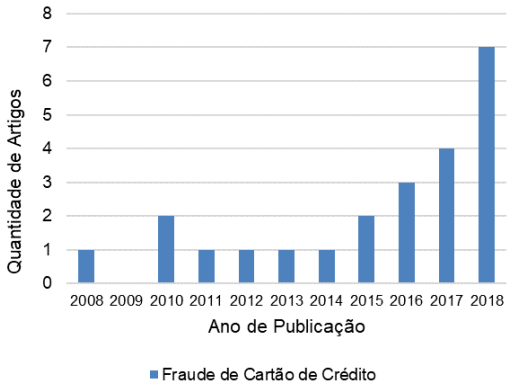


Figura 1.10: Evolução da quantidade de publicações com abordagens exclusivamente relativas ao tema Fraude de Cartão de Crédito, por campo de conhecimento, de 2008 a 2018, com base no portal Web of Science.

no período de análise, não houve uma revista específica em que os artigos de Fraude de Cartão de Crédito tenham sido concentrados.

1.3.4 Machine Learning & Fraude de Cartão de Crédito

Por fim, foi avaliada a intersecção entre os temas "Machine Learning" e "Fraude de Cartão de Crédito". As strings de busca foram tomadas com base na região D, da Tabela 1.2. Em uma pesquisa inicial, foram encontrados 149 trabalhos, sendo 12 de acesso aberto. Deste total, segundo a Tabela 1.10, 5 são relacionados à detecção de fraude de cartão de crédito, enquanto os demais trabalham com Data Science e Detecção de Outliers.

Após a filtragem destes 5 artigos, realizou-se uma análise de seus respectivos conteúdos. Para tanto, levou-se em consideração a estrutura proposta na Tabela 1.3, identificando o título, o(s) autor(es), ano e meio de publicação, objetivos, banco de dados utilizado, variáveis disponíveis, metodologia proposta e assertividade dos algoritmos. As tabelas com os conteúdos completos encontram-se no Apêndice.

De maneira sintética, observa-se, a partir da Figura 1.11, que os cinco artigos avaliados utilizaram, ao todo, dezessete algoritmos distintos, sendo que Logistic Regression e Random Forest apresentaram maior reincidência, com abordagem em três trabalhos. Em seguida, aparecem MultiLayer Perceptron, Naive Bayes, C4.5 e Neural Network, com duas aplicações cada. Com apenas uma utilização, aparecem Linear Regression, K-Nearest Neighbor, Support Vector Machine, Gradient Boost Tree, Random Tree, Parenclitic Network, Deep Learning, Bagging e Decision Stump.

A Tabela 1.11 apresenta dados condensados dos cinco artigos selecionados. O primeiro trabalho foi publicado apenas em 2013. Nos demais anos, houve uma publicação no ano de 2015 e três em 2018, relacionados à detecção de fraude de cartão de crédito com base no Machine Learning, apresentando indícios de que o tema tem ganhado um pouco mais de destaque a partir deste ano.

Além disso, observa-se que os trabalhos operam com abordagens distintas. Apesar de todos apresentarem resultados individuais dos algoritmos, foram utilizadas algumas técnicas com o objetivo de incrementar a qualidade dos resultados, como a Análise Combinada, o Majority Voting, Adaboost, Thresold Optimization, Bayesin Minimum Risk Classifier e Bayesian Minimum Risk Classifier com Probabilidade Ajustada. Em geral, os resultados apresentam melhoria de desempenho, como observado em

Tabela 1.9: Quantidade de artigos publicados exclusivamente com a temática de Fraude de Cartão de Crédito por meio de publicação, de 2008 a 2018, com base no portal Web of Science.

Meio de Publicação	Quantidade de Artigos
12th International Conference On Machine Learning And Applications (Icmla 2013), Vol 1	1
Big Data & Society	1
Complexity	1
Decision Support Systems	1
Fourth International Symposium On Information Assurance And Security, Proceedings	1
IEEE Access	1
IEEE Security & Privacy	1
Information	1
Information Visualization	1
International Conference On Computer, Communication And Convergence (ICCC 2015)	1
International Conference On Identification, Information And Knowledge In The Internet Of Things	1
International Conference On Emerging Trends In Engineering, Science And Technology (Icetest - 2015)	1
International Journal Of Computational Intelligence Systems	1
Journal Of Engineering Science And Technology	1
Journal Of Politics And Law	1
Knowledge-Based And Intelligent Information & Engineering Systems: Proceedings Of The 20Th International Conference Kes-2016	1
Mathematical Problems In Engineering	1
Pattern Recognition	1
Plos One	1
Romanian Statistical Review	1
Scientific World Journal	1
Theory Of Computing Systems	1
Visualization And Data Analysis 2010	1
Total	23

Tabela 1.10: Quantidade de artigos publicados envolvendo Machine Learning e Fraude de Cartão de Crédito, por tópico de aplicação, de 2008 a 2018, com base no portal Web of Science.

Tópico	Quantidade de Artigos	Participação (%)
Fraude de Cartão de Crédito	5	41,67%
Data Science	4	33,33%
Detecção de Outliers	3	25,00%
Total	12	100,00%

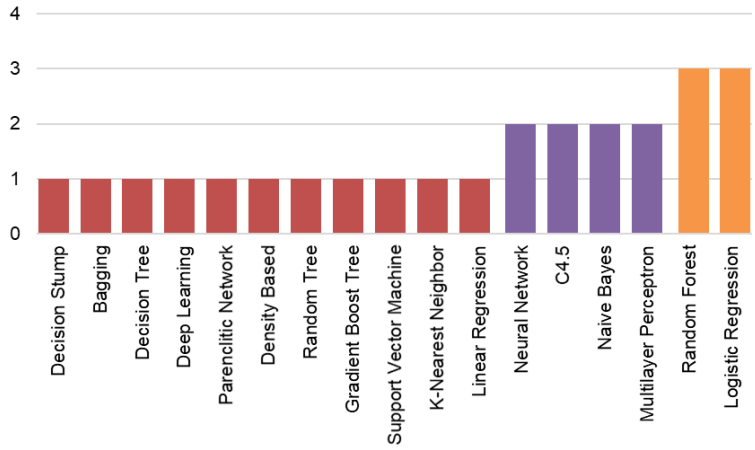


Figura 1.11: Algoritmos utilizados para a detecção de cenários de fraude de cartão de crédito, de 2008 a 2018, com base no portal Web of Science.

Zanin *et al.* (2018) e Randhawa *et al.* (2018). Há casos, no entanto, em que ferramentas mais complexas tendem a piorar o desempenho dos algoritmos individuais, como em Bahnsen *et al.* (2013).

Em relação aos bancos de dados utilizados para as análises, cada trabalho operou com um conjunto distinto. Zanin *et al.* (2018), Bahnsen *et al.* (2013) e Randhawa *et al.* (2018) operaram com dados provenientes de Instituições Financeiras. Choi and Lee (2018) obtiveram as informações junto ao mercado de Internet of Things, enquanto Zareapoor and Shamsolmoali (2015) trabalharam com dados viabilizados a partir de uma competição na Universidade da Califórnia.

Em análise das principais variáveis componentes dos bancos de dados, observa-se que a informação sobre o montante da transação é a variável mais utilizada, presente em quatro trabalhos. Em seguida, a variável de maior incidência foi data da transação, abordada em três trabalhos. A bandeira do cartão, o horário da transação, o número do cartão, a identificação do vendedor e a variável binária indicativa de fraude foram utilizadas em dois trabalhos, cada uma. Destaca-se, no entanto, que a presença da variável no banco de dados não implica que ela seja relevante para o desempenho do algoritmo. A relevância de cada variável está associada ao seu respectivo nível de significância, cuja métrica é definida pelos autores.

Por fim, dos cinco trabalhos analisados, o melhor desempenho de detecção de fraude foi observado em Randhawa *et al.* (2018), que obtiveram 100% de assertividade com os algoritmos Naive Bayes, Random Forest e Random Tree, a partir da abordagem de Adaboost, e aplicação de Decision Stump + Gradient Boost Tree, Decision Tree + Desision Stump, Decision Tree + Gradient Boost Tree, com o incremento do Majority Boosting.

Há, no entanto, de se destacar que, pelo fato dos trabalhos utilizarem bancos de dados distintos, não é possível firmar uma base comparativa em que se possa dizer quais algoritmos e quais métodos possuem melhores poder de detecção de fraudes financeiras. Além disso, deve-se considerar a possibilidade dos classificadores estarem sobreajustados, ou seja, possuírem uma parametrização ótima apenas para aquele conjunto de dados, de modo que, ao se inserir novas transações, os algoritmos se tornem menos eficazes.

Tabela 1.11: Informações condensadas dos artigos que aplicaram Machine Learning para fins de detecção de fraude de cartão de crédito, no período de 2008 a 2018, com base no portal Web of Science.

Componente	Resultados Observados
Anos de Publicação	2013 (1), 2015 (1) e 2018 (3)
Abordagens Utilizadas	Algoritmos Individuais (5); Análise Combinada (1); Majority Voting (1); Adaboost (1); Thresold Optimization (1); Bayesin Minimum Risk Classifier (1); Bayesian Minimum Risk Classifier com Probabilidade Ajustada (1)
Bancos de Dados	Companhia de Cartão de Crédito Europeia (1); Registros de Transações disponibilizados em competição da Universidade da Califórnia (1); Instituição Financeira da Malásia (1); Transações de cartão de débito e crédito do banco espanhol BBVA (1); Dados de Pagamento no ambiente da Internet of Things, no mercado da Coreia do Sul (1)
Variáveis	Montante da Transação (4); Data da transação (3); Bandeira do cartão (2); Horário da transação (2); Identificação do número do cartão (2); Identificação do Vendedor (2); Presença de fraude (variável binária, 0 ou 1) (2); Tipo de Mercadoria (2); Tipo de transação (internet, cartão presente, etc) (2); Banco emissor do cartão (1); Código da moeda da fatura do titular (1); Código da moeda da transação (1); Código de Autenticação do Cliente (1); Código do grupo do vendedor (1); Companhia de Telecomunicação (1); Fatura do titular do cartão (1)
Melhores Desempenhos	Randhawa *et al.* (2018). Aplicação de Naive Bayes, Random Forest e Random Tree, sob a abordagem de Adaboost; Aplicação de Decision Stump + Gradient Boost Tree, Decision Tree + Desision Stump, Decision Tree + Gradient Boost Tree, sob a abordagem do Majority Boosting

1.4 Considerações Finais

O Machine Learning engloba um conjunto de ferramentas utilizadas em diversas áreas do conhecimento, com o objetivo de potencializar a tomada de decisão humana, reduzindo os riscos de erros e suas eventuais consequências. As Ciências Naturais têm sido as principais utilizadoras dos recursos, no entanto, muito tem se trabalhado em demais campos do conhecimento aplicado, como a detecção de fraude de cartão de crédito. Nesta área, alguns algoritmos já apresentam desempenho satisfatório. Contudo, os resultados estão associados à qualidade dos bancos de dados. É notório, portanto, que o Machine Learning tem cooperado na detecção de transações de cartão de crédito fraudulentas, mas que existem caminhos abertos para a evolução e melhoria, como a comparação dos diversos métodos utilizados na literatura em dados semelhantes e a exploração do potencial sinérgico entre a Estatística e o Machine Learning, que pode ampliar o leque de ferramentas utilizadas na detecção de fraude de cartão de crédito.

Capítulo 2

ESTATÍSTICA E MACHINE LEARNING PARA DETECÇÃO DE FRAUDE DE CARTÃO DE CRÉDITO

RESUMO

A fraude de cartão de crédito é uma prática ilegal que movimenta bilhões de dólares todos os anos, implicando prejuízos para empresas e consumidores. Diante disso, é fundamental buscar mecanismos que permitam a detecção de tal prática, de modo a atenuar seu impacto negativo. Dois destes mecanismos são a Estatística e o Machine Learning, que oferecem, a partir de um conjunto de dados, a possibilidade de classificar novas transações, indicado a natureza legítima ou fraudulenta da operação. Neste sentido, este estudo buscou compreender de que forma a Estatística e o Machine Learning contribuem para a detecção de fraude de cartão de crédito. A partir de um conjunto de dados obtido na plataforma Kaggle, foram selecionados sete grupos amostrais balanceados, com 30, 60, 120, 180, 360, 540 e 984 observações. Os dados foram submetidos a análises estatísticas, representadas pelas NP-ANOVA, NP-MANOVA, Regressão Logarítmica, Análises Discriminantes Linear e Quadrática, Mínimos Quadrados Parciais e pela Regressão Logística, enquanto os algoritmos de Machine Learning utilizados foram o Naive Bayes, Random Forest, Multilayer Perceptron, CART, K-NN e Support Vector Machine. Os resultados revelaram que, para este conjunto de dados, a Estatística apresenta desempenho superior em amostras menores. Nos grupos com maior número de observações, observou-se um comportamento semelhante entre os algoritmos e os métodos estatísticos. Para implementações práticas, o indicado seria a combinação de duas metodologias: Random Forest e Regressão Logística. Desta forma, não se pode dizer, com precisão, qual método foi o mais efetivo. Nesse sentido, deve-se explorar o potencial sinérgico de ambos conjuntos de ferramentas, em vista de se obter melhores resultados.

Palavras-chave: Inteligência Artificial, Algoritmos, Detecção

2.1 Introdução

A fraude financeira é o ato de manipular pessoas, cenários e/ou organizações, com o objetivo de se obter vantagem ilícita, em prol de benefício próprio, seja ele individual ou corporativo. Pode ocorrer por meio de diferentes formas: apropriação, extorsão, maquiagem das práticas contábeis, falta de documento comprobatório de transações financeiras, entre outros. Um dos diversos tipos de fraude financeira é a de cartão de crédito, que, de modo geral, ocorre de duas maneiras: a partir do detentor do

cartão, que prática o ato ilícito em busca de benefício próprio, ou a partir de terceiros, situação na qual o detentor do cartão é a vítima Gadi (2008).

De acordo com estudo realizado pela consultoria Nilson Report (2016), no ano de 2015, os valores perdidos com a fraude de cartão de crédito atingiram a marca de 21,84 bilhões de dólares, valor superior ao Produto Interno Bruto dos 100 menores países (em renda) (Fundo Monetário Internacional, 2019).

Desta forma, buscar mecanismos que auxiliem na captação de fraude em operações de cartão de crédito é fundamental para mitigar os riscos e os impactos causados pela mesma. Ciente desta importância, a partir da década de 80 a Estatística surgiu como uma ferramenta para a identificação de possíveis cenários fraudulentos. Nos anos 90, novos modelos estatísticos emergiram, utilizando uma abordagem Bayesiana para a identificação de lavagem de dinheiro. Outros mecanismos, contendo identificação de padrões, análise de regressão, lógica difusa (*fuzzy logic*) e mineração de texto também foram progressivamente inseridos no contexto de fraude financeira como um todo (Phua *et al.*, 2012).

A partir da década de 2000, ganhou força o Machine Learning (ML), que, a princípio, dividiu os campos da Estatística e Inteligência Artificial. Para os cientistas conservadores, trata-se de pura Estatística. Para os mais progressistas, no entanto, é um braço que integra a Estatística e a Inteligência Artificial (IA). De fato, a Estatística per se, é fortemente amparada por equações e pressupostos, ainda que, grande parte deles, exija certo grau de abstração. O ML, por sua vez, também possui bases teóricas, mas permite uma maior flexibilização conceitual, pois não se restringe aos pressupostos matemáticos e estatísticos (Bzdok *et al.*, 2018).

Com diversos campos de aplicação, o ML tem sido utilizado em tópicos reconhecimento facial Bartlett *et al.* (2005), diagnóstico de doenças Kourou *et al.* (2015), segurança de dados na internet Domingos (2012), rastreabilidade de alimentos Fernandes *et al.* (2020), veículos autônomos Bzdok *et al.* (2017), entre outros. Diante da multiplicidade de áreas de aplicação e dos desempenhos satisfatórios que os algoritmos frequentemente apresentam, o ML é o principal componente da Inteligência Artificial, liderando o direcionamento de investimentos. No primeiro trimestre de 2019, por exemplo, foram investidos em torno de 42,9 bilhões de dólares em ML, considerando suas aplicações diretas e suas plataformas de suporte, liderando as inversões em Inteligência Artificial. No mesmo período, o gasto com robôs inteligentes, segundo colocado na lista, atingiu 7,5 bilhões de dólares (Columbus, 2020).

No entanto, mesmo diante do destaque recente vivenciado pelo ML, a discussão sobre a superioridade entre as áreas precisa ser substituída: Machine Learning e Estatística, aliados ao conhecimento prático de determinado contexto, compõem uma nova área de conhecimento, a Data Science. Desta forma, deve-se explorar o potencial (sinergia) que ambas resguardam entre si, ao invés de se insistir na tese de exclusividade e superioridade. Neste sentido, o objetivo que aqui se insere é explorar tal potencial sinérgico, por meio da utilização de métodos estatísticos e de algoritmos de ML, para a detecção de fraude de cartão de crédito.

2.2 Material e Métodos

2.2.1 Fraude de Cartão de Crédito

De acordo com o Cambridge Dictionary (2020)[1], fraude é "o crime de se tomar dinheiro ou propriedade trapaceando ou enganando as pessoas", ou seja, trata-se de um ato ilícito em prol de vantagem pessoal, que pode ocorrer tanto por meio de personalidade física, quanto jurídica.

Segundo a Serasa (2020)[2] , os principais tipos de fraude estão associados a boletos falsos, roubo de dados em sites falsos, compra de linhas telefônicas, abertura de empresas e pedido de empréstimo com documentos falsos. Em parte destes casos, o objetivo do fraudador é obter dados relacionados a cartões de crédito, utilizados para compras e saques ilícitos.

Configurado, portanto, como um dos diversos tipos de fraude, a fraude de cartão de crédito apresenta diferentes modalidades, como perda ou roubo, extravio de cartão, fraude de proposta, invasão da conta, falsificação manual, falsificação eletrônica e *Mail Order or Telephone Order* (MOTO), implicando custos financeiros e não financeiros, tanto para as vítimas diretas (titulares dos cartões), quanto para as indiretas (bandeiras e seguradoras) (Gadi, 2008). Dessa forma, para evitar tais perdas, esforços têm sido dirigidos para a prevenção e controle de fraudes de cartões de crédito, como em trabalhos de Gadi (2008), De Moraes (2008), Zanin *et al.* (2018), Randhawa *et al.* (2018), Bahnsen *et al.* (2013), Zareapoor and Shamsolmoali (2015) e Choi and Lee (2018).

Utilizando-se do benchmark dos trabalhos anteriores, bem como do potencial da Estatística e do Machine Learning, foi realizada uma seleção de métodos estatísticos e de algoritmos para a avaliação dos respectivos desempenhos na detecção de cenários de fraude de cartão de crédito. As próximas subseções apresentam brevemente as ferramentas selecionadas, com objetivo de oferecer a base teórica mínima para a compreensão de seus contextos de aplicação. Em casos de maior robustez matemática e estatística, o formulário não foi apresentado, sendo sugeridas leituras para o aprofundamento teórico.

2.2.2 Métodos Estatísticos

2.2.2.1 Non-Parametric MANOVA

A Non-Parametric MANOVA (NP MANOVA), também conhecida por Permutational Multivariate Analysis of Variance (PERMANOVA) é uma variação da tradicional Multivariate Analysis of Variance (MANOVA), porém orientada para estruturas que não atendem os pressupostos da MANOVA, dados por:

- Homogeneidade da matriz de variâncias e covariâncias;
- Normalidade Multivariada;
- Ausência de *Outliers*

Desta forma, na violação de um ou mais destes pressupostos, recomenda-se a utilização da NP MANOVA, cujo objetivo é a comparação da igualdade entre grupos, com no mínimo duas variáveis dependentes (Kelly *et al.*, 2015).

[1] https://dictionary.cambridge.org/pt/dicionario/ingles/fraud
[2] https://www.serasa.com.br/ensina/seu-cpf-protegido/o-que-e-fraude/

Seja **Y** o conjunto de variáveis dependentes, divididas em p grupos com n observações. Seja, também, **D** uma matriz de distância pareada, tal que d_{ij}, representa a distância entre as observações i e j. A soma de quadrados é decomposta por Soma de Quadrados Total (SQ_T), Soma de Quadrados Dentro dos grupos (SQ_D) e Soma de Quadrados Entre os grupos (SQ_E), de tal forma que $SQ_T = SQ_D + SQ_E$.

A soma total de quadrados é dada por:

$$SQ_T = \frac{1}{N} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (d_{ij})^2 \tag{2.1}$$

Para a soma de quadrados dentro dos grupos, tem-se que:

$$SQ_D^p = \frac{1}{n^p} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} (d_{ij}^p)^2 \epsilon_{ij}^p \tag{2.2}$$

De forma que ϵ_{ij} assume valor unitário quando as variáveis pertencem ao mesmo grupo. Sabe-se, portanto, que $SQ_T = SQ_D + SQ_E$. Assim, a estatística do teste, denominada *pseudo-F* é dada por:

$$F = \frac{\frac{SQ_A^p}{\alpha - 1}}{\frac{SQ_D^p}{N - \alpha}} \tag{2.3}$$

O termo *pseudo-F* vem do fato de que, no caso não paramétrico, o valor de F pode não ter as mesmas propriedades da distribuição F tradicional. Nesse sentido, a significância do teste é obtida através do processo de permutação[3].

2.2.2.2 Teste U de Mann-Whitney

O teste U de Mann-Whitney é método de comparação não paramétrica entre duas populações[4]. Similar ao Teste t de Student para amostras independentes, é recomendado quando as pressuposições paramétricas não são atendidas, sendo necessário o relaxamento de tais condições (Nachar, 2008).

Sejam X e Y duas amostras independentes. As hipóteses a serem testadas são:

$$\begin{cases} H_0 : \text{as populações são iguais} \\ H_1 : \text{as populações são diferentes} \end{cases}$$

As observações de ambos os grupos são ranqueadas em ordem crescente, sendo atribuído um valor r_{x_i} e r_{y_j}, denominados de postos, às respectivas observações. A estatística do teste é dada por:

$$U_x = n_x n_y + \frac{n_x(n_x - 1)}{2} - R_X \tag{2.4}$$

$$U_y = n_x n_y + \frac{n_y(n_y - 1)}{2} - R_Y \tag{2.5}$$

Em que n_x e n_y são, respectivamente, os tamanhos amostrais de X e Y e R_X e R_Y a soma dos postos de $r_x i$ e $r_y j$.

Escolhe-se, entre U_x e U_y aquela de menor valor. Para amostras com menos de dezesseis observações ($n_x + n_y$), utiliza-se uma aproximação de normalidade, tal que:

[3] Ver mais em Anderson (2001)

[4] Quando se tem um conjunto com três ou mais populações, utiliza-se o Teste de Kruskal Wallis.

$$z_{calculado} = \frac{U - \mu_R}{\sigma_R} \tag{2.6}$$

Onde:

$$\mu_R = \frac{n_x n_y}{2} \tag{2.7}$$

$$\sigma_R = \frac{\sqrt{n_x n_y (n_x + n_y + 1)}}{12} \tag{2.8}$$

Assim, consultando a tabela z e considerando as hipóteses apresentadas, que implicam a utilização de um teste bicaudal, a hipótese nula é rejeita quando $|z_{calculado}| \geq |z_{tabelado}|$.

2.2.2.3 Regressão Logística

A Regressão Logística é uma ferramenta que possibilita a estimação da probabilidade de ocorrência de determinando evento, representado por duas ou mais categorias, a partir de um conjunto de variáveis, binárias ou não. É diversamente utilizada nos diferentes campos de conhecimento, como na determinação de diagnósticos, na área da saúde Langer *et al.* (2009), análises de risco, na área financeira (Bayaga, 2010) ou na avaliação dos principais fatores determinantes da pobreza Achia *et al.* (2010).

Conforme exposto por Figueira (2006), seja $\mathbf{X}$ um conjunto de variáveis independentes com n observações e $\mathbf{Y}$ um vetor de variável dependente binária, tal que $0 \leq E\ (\mathbf{Y}|\mathbf{x}) \leq 1$. É composta uma função linear $g(\cdot)$, denominada *logit*, dada por:

$$g(\mathbf{X}) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \beta_n x_n \tag{2.9}$$

Apresentando um modelo probabilístico tal que $P(Y=1|\boldsymbol{x})=\pi(\boldsymbol{x})$, tem-se que:

$$\pi(\mathbf{x}) = \frac{exp[g(\mathbf{x})]}{1 + exp[g(\mathbf{x})]} \tag{2.10}$$

Generalizando a função *logit* para p variáveis independentes, obtém-se a seguinte estrutura:

$$g(\mathbf{X}) = \frac{exp[g(\mathbf{X})]}{1 + exp[g(\mathbf{X})]} = ln(\frac{\pi(\mathbf{X})}{1 - \pi(\mathbf{X})}) = \beta_0 + \beta_1 \mathbf{x}_1 + \beta_2 \mathbf{x}_2 + \ldots + \beta_p \mathbf{x}_p \tag{2.11}$$

Sendo que:

$$\begin{aligned}
g_1 &= g_1(\mathbf{x}_1) = \beta_0 + \beta_1 x_{11} + \beta_2 x_{12} + \ldots + \beta_p x_{1p} + \varepsilon_1 \\
g_2 &= g_2(\mathbf{x}_2) = \beta_0 + \beta_1 x_{21} + \beta_2 x_{22} + \ldots + \beta_p x_{2p} + \varepsilon_2 \\
g_3 &= g_3(\mathbf{x}_3) = \beta_0 + \beta_1 x_{31} + \beta_2 x_{32} + \ldots + \beta_p x_{3p} + \varepsilon_3 \\
&\ldots \\
g_n &= g_n(\mathbf{x}_n) = \beta_0 + \beta_1 x_{n1} + \beta_2 x_{n2} + \ldots + \beta_p x_{np} + \varepsilon_n
\end{aligned} \tag{2.12}$$

Onde ε_i representam os i erros aleatórios, tal que:

1. $E(\varepsilon_i|\mathbf{x}_i) = 0$
2. $Var(\varepsilon_i|\mathbf{x}_i) = \pi(\mathbf{x}_i)[1 - \pi(\mathbf{x}_1)]$
3. $Cov(\varepsilon_i, \varepsilon_l) = 0 \forall (i, l) \in (1, \ldots, n) | i \neq l$

A partir das estruturas apresentadas, o modelo logístico é dado por:

$$y_i = \frac{exp(g_i)}{1 + exp(g_i)} + \varepsilon_i \tag{2.13}$$

Os parâmetros de tal modelo são obtidos por meio do método da máxima verossimilhança. As demonstrações estão disponíveis em Figueira (2006).

2.2.2.4 Principal Component Anaylisis

A Principal Component Analysis (PCA) é uma técnica multivariada de combinação linear entre as variáveis de um banco de dados, com o objetivo de extrair as informações mais importantes, redimensionar o tamanho dos dados com base nessas informações e retirar problemas de multicolinearidade entre as variáveis (Abdi and Williams, 2010).

Seja $\mathbf{X}$ uma matriz de variáveis independentes de um conjunto de dados, com n observações e p variáveis, tal que:

$$\boldsymbol{X} = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{p1} & x_{p2} & \dots & x_{np} \end{bmatrix}_{n \times p}$$

A matriz de variâncias e covariâncias $\mathbf{\Sigma}$ é dada por:

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_{11}^2 & \sigma_{12}^2 & \dots & \sigma_{1p}^2 \\ \sigma_{21}^2 & \sigma_{22}^2 & \dots & \sigma_{2p}^2 \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{p1}^2 & \sigma_{p2}^2 & \dots & \sigma_{pp}^2 \end{bmatrix}_{p \times p}$$

De forma que $\text{tr}(\mathbf{\Sigma}) = \sum_{i=1}^{p} Var(\boldsymbol{X_i})$. Sendo $\boldsymbol{\lambda_i}$ os autovetores e e_i os autovalores associados à matriz $\mathbf{\Sigma}$, a matriz de componentes principais $\mathbf{Z}$, inferida a partir da combinação de $\mathbf{X}$, é dada por **Z=e'X**. Para o i-ésimo caso, tem-se que:

$$\boldsymbol{Z_i} = \boldsymbol{e_i'X} = e_{i1}X_1 + e_{i2}X_2 + \ldots + e_{ip}X_p \tag{2.14}$$

Com $i = (1, \ldots, n)$.
A $Var(\boldsymbol{Z_i})$ é dada por:

$$Var(\boldsymbol{Z_i}) = Var(\boldsymbol{e_i'X}) = \boldsymbol{e_i'Var(X)e_i} = \boldsymbol{e_i'\Sigma e_i} \tag{2.15}$$

A partir da composição espectral (ver mais em Abdi and Williams (2010); Poole (2014)), tem-se que:

$$\mathbf{\Sigma} = tr(\mathbf{P\Lambda P'}) = tr(\mathbf{\Lambda PP'}) = tr(\mathbf{\Lambda I}) = tr(\mathbf{\Lambda}) = \sum_{i=1}^{p} \lambda_i = \sum_{i=1}^{p} var(\boldsymbol{X_i}) \tag{2.16}$$

Assim, é possível obter a participação de cada componente principal $\boldsymbol{Z}_i$, dada por:

$$C_k = \frac{Var(Z_i)}{\sum_{i=1}^{p} Var(Z_i)} \times 100 = \frac{\lambda_i}{\sum_{i=1}^{p} \lambda_i} \times 100 = \frac{\lambda_i}{tr(\boldsymbol{\Sigma})} \times 100 = \frac{\lambda_i}{\sum_{i=1}^{p} Var(\boldsymbol{X_i})} \times 100 \tag{2.17}$$

A partir do cálculo da participação, obtém-se o número de componentes a serem utilizadas no modelo, com base no critério de $\sum_{k=1}^{i} C_k \geq 80\%$ (Hongyu *et al.* (2016);Johnson and Wichern (2002)).

2.2.2.5 Mínimos Quadrados Parciais

O método de Mínimos Quadrados Parciais trata-se de um estimador linear orientado para situações em que o pressuposto da independência entre as variáveis é violado (Geladi and Kowalski, 1986). Conforme Ng (2013), seja X a matriz de variáveis preditoras e Y a matriz de variável resposta. A partir destas matrizes, é a realizada a decomposição em componentes principais, de modo que:

$$\boldsymbol{X = TP'} \tag{2.18}$$

$$\boldsymbol{Y = UQ'} \tag{2.19}$$

Em que P e T são matrizes de componentes principais e score de X, respectivamente; U é uma matriz de autovetores de Y e Q é uma matriz ortogonalizada dos vetores de U.

Em visão análoga, Pirouz (2006) define o modelo matemático dos Mínimos Quadrados Parciais por:

$$N = W'Y \tag{2.20}$$

$$Y = PN + E \tag{2.21}$$

Em que N é a componente principal, W é a composição dos pesos, E é a variância residual , Y representam os scores dos valores observados e P é um conjunto de carregamento das componentes principais.

Desta forma, substituindo 2.20 em 2.21, obtém-se o estimador de mínimos quadrados parciais[5], dado por:

$$Y = PW'Y + E = PW'Y + (I - PW')Y \tag{2.22}$$

O modelo de Mínimos Quadrados Ponderados permite a criação de um classificador, de onde será obtida uma matriz de confusão, que permitirá a avaliação do desempenho e a comparação com os algoritmos de Machine Learning.

2.2.3 Algoritmos de Machine Learning

2.2.3.1 Redes Neurais Artificiais

Inspirada no neurônio humano, a Rede Neural Artificial é um sistema capaz de aprender e reconhecer padrões, por meio de funções matemáticas, normalmente não lineares. É composta por unidades

[5] Ver mais em Geladi and Kowalski (1986).

de processamento, denominadas nodos (ou neurônios), que se interligam através de conexões geralmente unidirecionais. Conforme a Figura 2.1, os nodos são agrupados em três camadas: Camada de Entrada, onde os padrões são introduzidos, Camada Oculta [6], onde os padrões são processados e Camada de Saída, onde o resultado final é apresentado (Braga *et al.*, 2000).

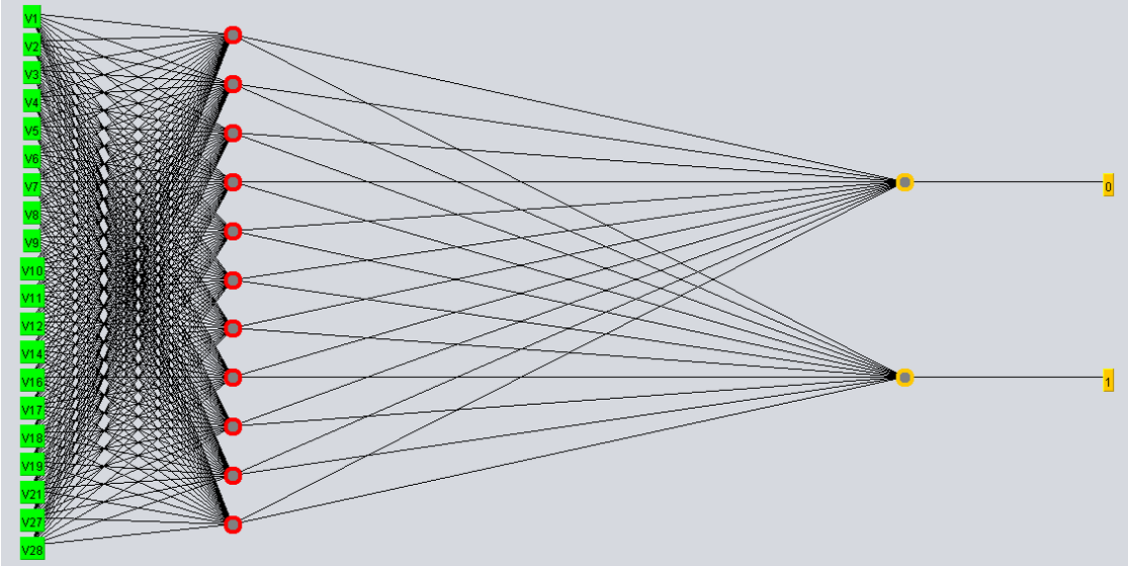


Figura 2.1: Esquema estrutural de uma rede neural.
Fonte: Elaborado pelo autor a partir do software Weka.

Matematicamente, tem-se a seguinte estrutura (Moreira et al, 2006):

$$y^{ij} = f(\sum_{i'}^{n} y^{i'(j-1)} w_{i'}^{ij} + b^{ij}) \tag{2.23}$$

Em que:

- y^{ij} é o valor de saída de cada neurônio i da camada j;
- n é o número de neurônios da camada anterior;
- $y^{i'(j-1)}$ é valor de saída do neurônio i' da camada anterior;
- $w_{i'}^{ij}$ é o valor do peso sináptico do neurônio i da camada j, ativado pelo neurônio i' da camada anterior;
- b^{ij} é o valor de compensação do neurônio i da camada j;
- f é a função de ativação do neurônio i.

[6] Quando se aumenta o número de camadas ocultas, associando as redes neurais a grande volumes de dados, passa-se a operar na esfera do deep learning.

2.2.3.2 Random Forest

O Random Forest configura-se como uma ferramenta de classificação baseada na integração entre diversas árvores de decisão (decision trees), geradas aleatoriamente a partir de um conjunto de dados. É utilizado em diversas situações, como no diagnóstico de doenças (Statnikov *et al.* (2008), Nguyen *et al.* (2013)), finanças (Kumar and Thenmozhi (2006)), identificação da cobertura do solo (Gislason *et al.* (2006)), mostrando-se um potente algoritmo de classificação.

Conforme Breiman (2001), a partir da criação de k árvores aleatórias, tem-se k vetores aleatórios $_k$, caracterizados também pela independência. Sendo x um vetor de entrada, é obtido um classificador h, tal que $h = f(x, _k)$. Após a geração de diversas árvores, que configurarão a floresta, cada árvore apresenta um voto para a classificação da classe do vetor x.

De forma geral, a estrutura do Random Forest é dada pela Figura 2.2. No exemplo, tem-se uma matriz de dados $\mathbf{X}$, que irá gerar k distintas árvores (A1, A2, ..., Ak). Cada árvore apresenta um output, que irá definir a classificação final do algoritmo do vetor x.

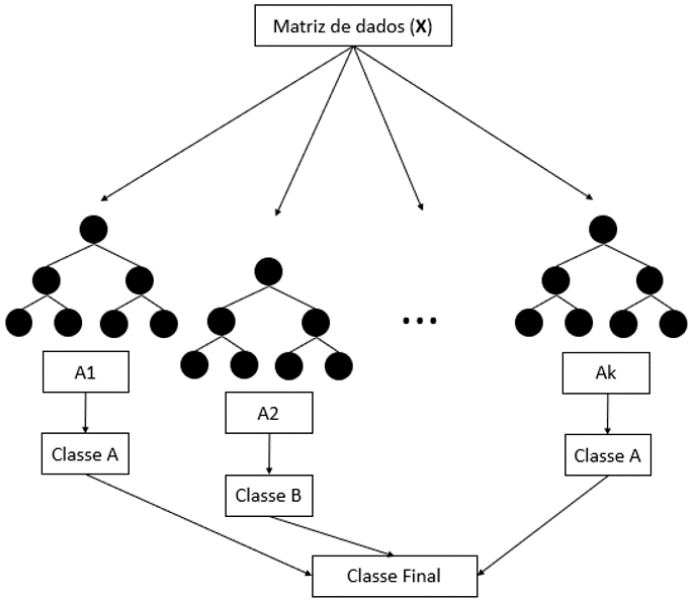


Figura 2.2: Esquema estrutural do Random Forest.

2.2.3.3 Support Vector Machine

O Support Vector Machine é um algoritmo que mapeia os dados em um espaço dimensional, de onde se obtém um hiperplano de separação ótima, que funcionará como uma fronteira de decisão (classificação) (Lorena and De Carvalho, 2007). Capaz de operar tanto para dados lineares, quanto para não lineares, configura-se como um dos algoritmos de Machine Learning mais robustos.

A separação ótima é obtida através do processo de otimização, formando margens de confiança, conforme expresso na Figura 2.3.

Seja $\mathbf{X}$ uma matriz de dados e $\mathbf{y}$ um vetor de rótulos, de forma que os pontos relacionados sejam $(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)$, com y assumindo valores entre -1 e +1. A equação do espaço hiperplano é definida por:

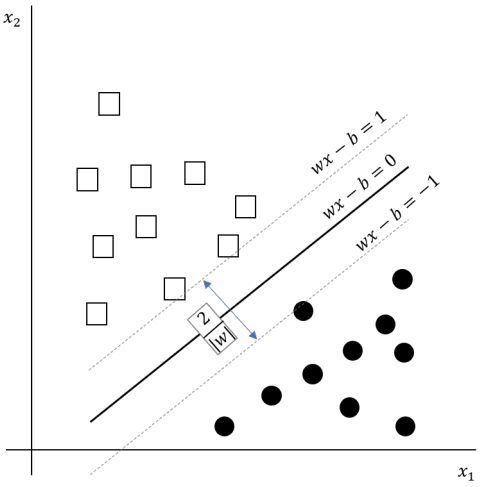


Figura 2.3: Esquema estrutural do Support Vector Machine.

$$\mathbf{wx\text{-}b} = 0 \tag{2.24}$$

Em que **w** é um vetor normal, perpendicular à reta **wx-b** = 0 e b uma constante real. As margens são definidas por uma função sinal $g(x)$, de forma que:

$$g(x) = sgn(f(x)) = \begin{cases} \mathbf{wx}_i + b \geq +1, \forall y = +1 \\ \mathbf{wx}_i + b \leq -1, \forall y = -1 \end{cases} \tag{2.25}$$

Desta forma, tem-se dois hiperplanos, delimitados por $\mathbf{wx}_i + b \geq +1$ e $\mathbf{wx}_i + b \leq -1$, cuja amplitude é dada por:

$$\frac{2}{|\mathbf{w}|} \tag{2.26}$$

No Weka, software utilizado para os algoritmos de Machine Learning, o Support Vector Machine apresenta a descrição Sequential Minimal Optimization (SMO), que substitui valores não atribuídos, transforma as variáveis nominais em binárias e normaliza os atributos (Bouckaert *et al.*, 2013).

2.2.3.4 Naive Bayes

O Naive-Bayes configura-se como um algoritmo de aprendizado supervisionado que utilizado o aparato estatístico bayesiano para realizar as classificações. Nesse sentido, é fundamental resgatar de forma elementar o Teorema de Bayes, expresso por:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)} \tag{2.27}$$

Em que:

- $P(A|B)$: é a probabilidade condicional de determinado evento A ocorrer dado que B já tenha ocorrido;
- $P(B|A)$: é a probabilidade condicional de determinado evento B ocorrer dado que A já tenha ocorrido;
- $P(A)$: é a probabilidade do evento A ocorrer;
- $P(B)$: é a probabilidade do evento B ocorrer.

Para o algoritmo de Naive Bayes, conforme Bramer (2007), seja c o vetor de k classes, com as probabilidades a priori $P(c_1), P(c_2), \ldots, P(c_k)$, associadas a n atributos, dados por $a_1, a_2, a_3, \ldots, a_n$. A cada atributo, associa-se um valor (v) respectivo , de forma que $v = v_1, v_2, v_3, \ldots, v_n$. Assim, a probabilidade da classe c_i ocorrer para o valor específico de v é tida por:

$$P(c_i) \times P(a_1 = v_1; a_2 = v_2; \ \ldots \ , a_n = v_n \mid c_i) \tag{2.28}$$

Assumindo que os atributos são independentes entre si, a expressão acima pode ser reescrita por:

$$P(c_i) \times P(a_1 = v_1 \mid c_i) \times P(a_2 = v_2 \mid c_i) \times \ \ldots \ \times P(a_n = v_n \mid c_i) \tag{2.29}$$

Por fim, calcula-se o valor de cada observação i, de 1 a k, e selecionando-se aquela de maior dimensão (Bramer, 2007).

Um dos pressupostos do Naive Bayes é a independência entre as variáveis. Para tanto, uma das alternativas é utilizar a transformação por meio da Principal Component Analysis, método que foi implementado neste trabalho.

2.2.3.5 CART

O Classification and Regression Trees (CART) é um algoritmo composto por árvores de decisão, que são estruturas de aprendizado supervisionado semelhantes a uma árvore invertida, possuindo quatro elementos principais: nó, ramo, raiz e folhas (Figura 2.4). O nó (ou nó de decisão) apresenta os testes para algum atributo da análise. Os ramos referem-se a um possível valor para estes atributos. Raiz é o questionamento central da análise, enquanto as folhas são as categorias para classificação (Rutkowski *et al.*, 2014).

O CART é aplicado tanto em situações de predição, quanto classificação. Com abordagem não paramétrica, utiliza atributos explicativos para classificar ou predizer outro atributo de interesse (Cabete and Cardoso (2006)). Nesse sentido, seja X o conjunto de variáveis explicativas e Y uma variável categórica a ser explicada. A partir dos conjuntos, o algoritmo gera uma árvore binária que realiza a divisão de um subconjunto, denominado folha, em duas partes, denominadas sub-folhas. Tal divisão toma por base o critério de minimização da heterogeneidade.

Desta forma, seja T uma árvore de decisão e t uma de suas folhas. Por meio do mapeamento T, para cada amostra $(X_i^1, \ldots, X_i^p)$ é associada uma folha t. Para o cálculo da heterogeneidade, conforme Bel *et al.* (2009), os critérios mais populares são a entropia e o Índice de Gini.

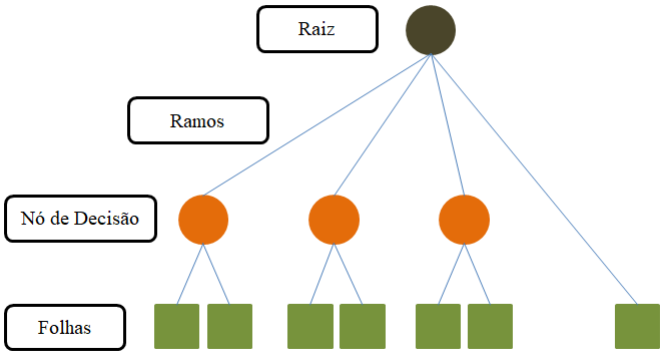


Figura 2.4: Esquema estrutural de uma Árvore de Decisão.

A entropia é dada por:

$$E_t = \sum_{j}^{max} p(j|t) \times \log p(j|t) \tag{2.30}$$

Para o Índice de Gini, tem-se que:

$$D_t = \sum_{i \neq j} p(i \mid t) p(j|t) = 1 - \sum_{i} p(i \mid t)^2 \tag{2.31}$$

Em que $P(j|t)$ é a proporção da classe j na folha t.

Quando se tem apenas uma classe em cada folha, o valor para ambos os índices equivale a 0. Por outro lado, apresentam valor máximo quando todas as classes estão presentes com a mesma probabilidade (Bel *et al.*, 2009). O algoritmo mantem as subdivisões de folhas até que a diferença entre a heterogeneidade de uma folha e suas sub-folhas seja máxima e positiva, encerrando o processo de divisão até a condição ser atingida. Em seguida, cada folha é associada a uma atributo, de onde se terá a classificação.

2.2.3.6 K-NN

O K-Nearest Neighbors (K-NN) é um classificador que toma por base a semelhança entre objetos rotulados para a classificação de objetos não rotulados. Existem, nesse sentido, dois conjuntos de dados: um de treinamento e outro de teste. O de treinamento é utilizado para ensinar a máquina a reconhecer os padrões, enquanto que o de teste é direcionado à avaliação da assertividade do algoritmo (Zhang, 2016).

A ideia principal reside na distância com que um objeto não rotulado se encontra de um objeto rotulado. Tal distância é medida, no KNN, pela Distância Euclidiana, obtida por:

$$D(p,q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2} \tag{2.32}$$

Em que:

- p e q são os objetos a serem classificados, com n características.

2.2.3.7 Dados de Cartão de Crédito

Os dados analisados foram obtidos junto à plataforma Kaggle[7], uma comunidade virtual de Machine Learning e Data Science. O banco de dados, denominado *Credit Card Fraud Detection*, é composto por trinta e uma variáveis, sendo vinte e oito anonimizadas e três nomeadas, relativas, respectivamente, ao instante de captação do vetor de dados, ao volume da operação e à classificação de transação fraudulenta ou não. Desta forma, obtém-se o seguinte conjunto de variáveis:

- *Time*: instante de captação do vetor de dados;
- V1 a V28: variáveis anonimizadas;
- *Amount*: volume da transação;
- *Class*: variável binária indicativa de fraude (1 para fraude, 0 para não-fraude).

O banco completo é composto por 284.807 linhas, das quais 492 são relativas a operações fraudulentas e 284.315 referem-se a transações legítimas. Como forma de balancear a análise e explorar o desempenho dos algoritmos, foram realizadas sub-amostragens aleatórias. Para tanto, foi utilizada a função *sample_n*, disponível no pacote *dplyr*, do software R (script disponível no Apêndice). Os tamanhos amostrais foram definidos segundo o conceito de amostragem progressiva (Díaz (2004), Provost *et al.* (1999)), conforme a Tabela 2.1:

Tabela 2.1: Tamanhos amostrais definidos.

0 (Não Fraudulenta)	1 (Fraudulenta)	Total de Observações
15	15	30
30	30	60
60	60	120
90	90	180
180	180	360
270	270	540
492	492	984

2.2.3.8 Softwares Utilizados

Ao todo, foram utilizados cinco softwares para as execuções das análises: Weka, RStudio, SAS, Action Stat e JMP.

Weka

O Weka é um software gratuito criado pela Universidade de Waikato, da Nova Zelândia, e amplamente utilizado na área de inteligência artificial. Com um desempenho satisfatório em bancos de dados médios, apresenta uma interface estabelecida em menus, conforme Figura 2.5. Com diversas ferramentas incluídas, permite a realização de análises de classificação, clusterização e regressão.

[7] https://www.kaggle.com/

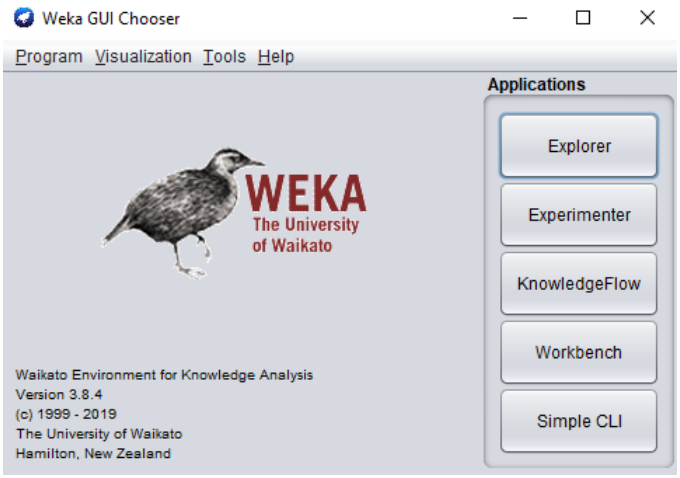


Figura 2.5: Menu inicial do software Weka (3.8.4).

Neste sentido, o Weka (3.8.4) foi utilizado para a execução das seguintes análises: Redes Neurais Artificiais[8], Random Forest, Support Vector Machine[9], Naive-Bayes, CART[10] e K-NN[11].

RStudio

O RStudio é um software de programação gratuito, orientado para análises matemáticas e estatísticas. Baseado na linguagem C++, o RStudio integra o software R em uma plataforma visual, com quatro quadrantes principais, conforme a Figura 2.6.

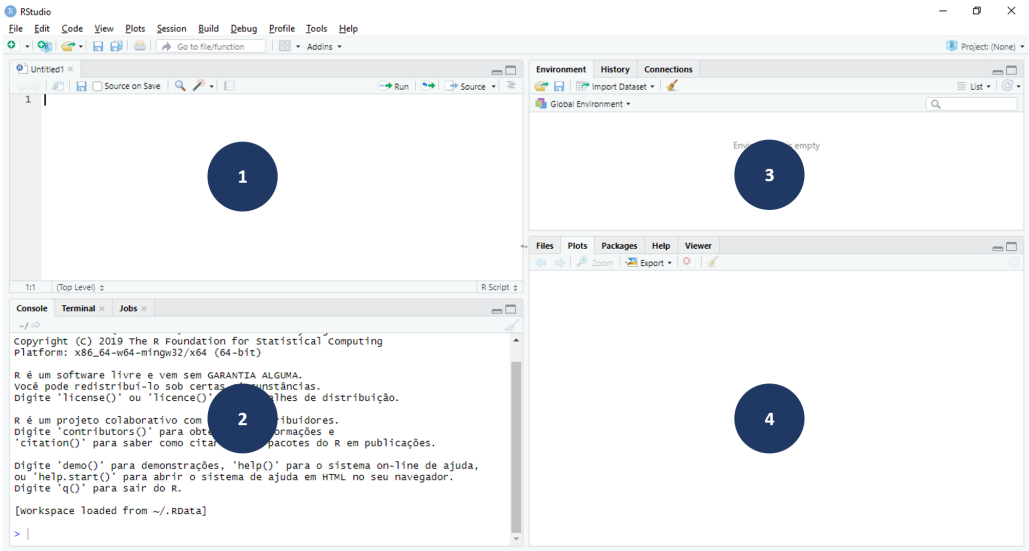


Figura 2.6: Telas de operação do software RStudio (Version 1.2.5033).

No primeiro quadrante (1), encontra-se a Janela de Scripts, onde os códigos são escritos e

[8] No Weka: Multilayer Perceptron
[9] No Weka: SMO
[10] No Weka: Classification via Regression
[11] No Weka: IBk

executados. No quadrante 2 apresenta-se o Console, onde os códigos são transcritos e os resultados são obtidos. No terceiro quadrante, está o Ambiente, local em que são registradas as variáveis e bancos criados na análise. Por fim, o quarto quadrante apresenta a saída gráfica, a tela de ajuda e a biblioteca de pacotes.

Diante da aplicabilidade de métodos estatísticos e matemáticos, o RStudio (Version 1.2.5033) foi utilizado para a execução do processo de amostragem, além das análises de Regressão Logística e NP MANOVA.

SAS

Utilizado inicialmente como um sistema para pesquisa agrícola, o SAS se tornou uma das principais ferramentas de análise estatística e Business Intelligence. Diferentemente do R, trata-se de um software pago, mas que possui uma versão limitada e gratuita, o *SAS on Demand* (Figura 2.7). O diferencial do SAS é a completude de seus resultados a partir de uma linguagem de quarta geração, que permite efetuar grande número de processos em poucas linhas de códigos. Os comandos são executáveis a partir de *procedures*, que indicam o tipo de análise a ser realizada.

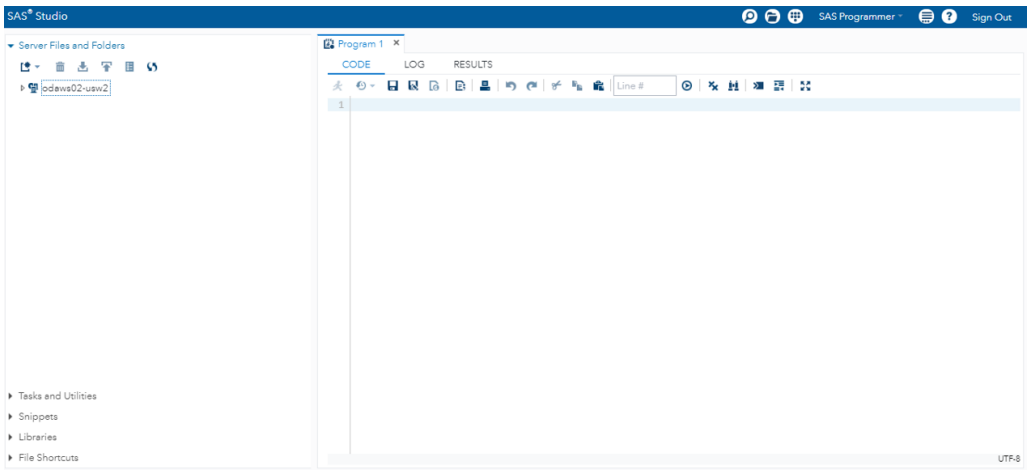


Figura 2.7: Interface do Software SAS on Demand.

As análises executadas no SAS foram: Teste U de Mann-Whitney, Mínimos Quadrados Ponderados e Principal Component Analysis.

Action Stat

Com lastro brasileiro, o Action Stat (Figura 2.8) é um software de análise estatística com interface em Excel. Por meio da versão 3.6.331.450, foram realizadas as Análises Discriminantes Linear e Quadrática.

JMP

O JMP é um software que apresenta um conjunto de pacotes gráficos com interface em SAS. Sob a versão 15.1.0, foram executadas os gráficos da Principal Component Analysis, das Análises Discriminantes Linear e Quadrática e dos gráficos de variáveis canônicas.

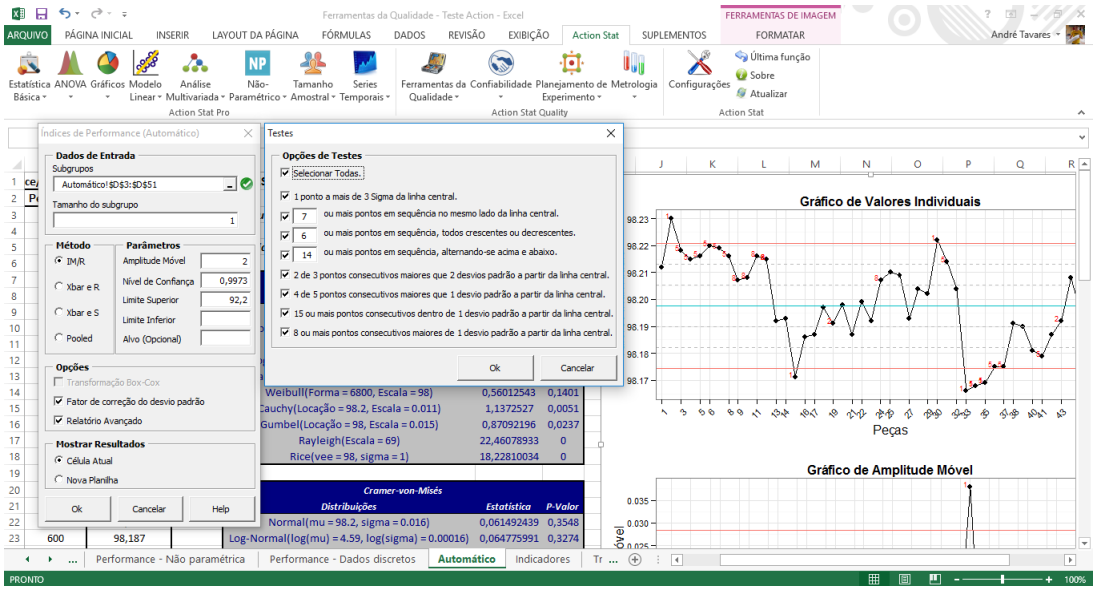


Figura 2.8: Interface do Software Action Stat (3.6.331.450). Fonte: Portal Action (2020).

2.2.3.9 Teste U de Mann-Whitney e Seleção de Variáveis

Conduzido no SAS, o Teste U de Mann-Whitney foi utilizado como critério de inclusão para a elaboração dos cenários. Ao todo, foram elaborados quatro cenários, tomando-se por base o *p-valor* de cada variável. Foi estipulado um número inicial de 540 observações, sendo 270 fraudulentas e 270 não fraudulentas. Os resultados estão apresentados na Tabela 2.2.

Tabela 2.2: Resultados para o Teste U de Mann-Whitney.

Variável	Z	Pr > Z	Pr > |Z|
V1	11,022	<0,0001	<0,0001
V2	-13,522	<0,0001	<0,0001
V3	16,337	<0,0001	<0,0001
V4	-17,609	<0,0001	<0,0001
V5	8,448	<0,0001	<0,0001
V6	10,751	<0,0001	<0,0001
V7	13,913	<0,0001	<0,0001
V8	-5,758	<0,0001	<0,0001
V9	13,013	<0,0001	<0,0001
V10	16,114	<0,0001	<0,0001
V11	-16,965	<0,0001	<0,0001
V12	17,646	<0,0001	<0,0001
V13	1,992	0,0232	0,0464
V14	17,808	<0,0001	<0,0001
V15	0,972	0,1656	0,3312
V16	13,868	<0,0001	<0,0001
V17	12,289	<0,0001	<0,0001
V18	10,011	<0,0001	<0,0001
V19	4,219	<0,0001	<0,0001
V20	-3,618	0,0001	0,0003
V21	10,061	<0,0001	<0,0001
V22	-1,260	0,1039	0,2078
V23	0,445	0,3280	0,6560
V24	3,458	0,0003	0,0005
V25	0,759	0,2240	0,4480
V26	-2,809	0,0025	0,0050
V27	6,548	<0,0001	<0,0001
V28	-4,852	<0,0001	<0,0001
Amount	1,954	0,0254	0,0507

Observa-se, portanto, que a maior parte das variáveis apresenta significância para $\alpha = 0,05$. Das variáveis listadas, apenas V15, V22, V23, V25 e Amount não obtiveram $Pr > |Z| < 0,05$.

Neste sentido, o primeiro cenário foi elaborado utilizando as sete variáveis de menor significância, sendo duas significativas e quatro não-significativas. Também foram inseridas três variáveis aleatórias, com o intuito de testar o desempenho dos modelos e algoritmos. Assim, o Cenário I foi estruturado conforme a Tabela 2.3.

Tabela 2.3: Variáveis selecionadas para a construção do Cenário I.

Variáveis	Pr > |Z|
V13	0,0464
V15	0,3312
V22	0,2078
V23	0,6560
V25	0,4480
V26	0,0050
Amount	0,0507
A1	-
A2	-
A3	-

As variáveis aleatórias simuladas sem efeito do fator fraude foram geradas em Excel, a partir da fórmula ALEATÓRIOENTRE(). A1 tem amplitude de 0 a 1, A2 vai de 0 a 5 e A3 de 0 a 10, todas com seis casas decimais, acompanhando as variáveis reais do conjunto de dados.

O Cenário II foi construído a partir do Cenário I, adicionando-se as seguintes variáveis: V1, V5, V6, V8, V18, V19, V20, V21, V24, V27 e V28, selecionado variáveis significativas, mas não incluindo as mais significativas, mantendo-se, no entanto, as variáveis menos significativas. Desta forma, o Cenário II foi estruturado conforme a Tabela 2.4.

Tabela 2.4: Variáveis selecionadas para a construção do Cenário II.

Variáveis	Pr > |Z|
V1	<0,0001
V5	<0,0001
V6	<0,0001
V8	<0,0001
V13	0,0464
V15	0,3312
V18	<0,0001
V19	<0,0001
V20	0,0001
V21	<0,0001
V22	0,2078
V23	0,6560
V24	0,0003
V25	0,4480
V26	0,0050
V27	<0,0001
V28	<0,0001
Amount	0,0507
A1	-
A2	-
A3	-

O Cenário III foi composto por todas as variáveis disponíveis na Tabela 2.2, além das variáveis aleatórias A1, A2 e A3. Assim, foram incluídas as variáveis altamente significativas (V2, V3, V4, V7,

V9, V10, V11, V12, V14, V16, V17),

Por fim, o Cenário IV apresenta apenas as variáveis altamente significativas segundo o Teste U de Mann-Whintey, com $\alpha = 0,01$, sendo obtida a disposição expressa na Tabela 2.5. Para este cenário, foram retiradas as variáveis aleatórias (A1, A2 e A3).

Tabela 2.5: Variáveis selecionadas para a construção do Cenário IV

Variáveis	Pr > |Z|
V1	<0,0001
V2	<0,0001
V3	<0,0001
V4	<0,0001
V5	<0,0001
V6	<0,0001
V7	<0,0001
V8	<0,0001
V9	<0,0001
V10	<0,0001
V11	<0,0001
V12	<0,0001
V14	<0,0001
V16	<0,0001
V17	<0,0001
V18	<0,0001
V19	<0,0001
V21	<0,0001
V27	<0,0001
V28	<0,0001

Desta forma, todas as análises foram realizadas para os quatro cenários. Os resultados foram agrupados por cenário nas quatro primeiras sessões. Ao final, é apresentado um comparativo entre os cenários para as análises estatísticas e os algoritmos de Machine Learning.

2.2.3.10 Métodos de Avaliação

Como forma de avaliação do aprendizado do Machine Learning, são utilizados diversos critérios de validação, a partir dos quais será expressa a taxa de acerto (TA) do algoritmo. Dentre tais critérios, destacam-se dois: separação entre treino e teste e Validação Cruzada (*Cross Validation*).

Treino e Teste

A abordagem de treino e teste consiste na separação amostral entre dois grupos. Um será utilizado para o treinamento do algoritmo e o outro para a testagem e avaliação de tal aprendizado. Para seleção dos dados de treino e teste, é aplicado um split, ou seja, uma divisão do conjunto original, geralmente expressa em termos percentuais. Os splits frequentemente utilizados são os de 60% treino, 40% teste e 75% treino, 25% teste.

Desta forma, a partir de um banco de dados original, são extraídas aleatoriamente duas subamostras, conforme a Figura 2.9. Após o processo de seperação, os dados de treino são submetidos a um algoritmo, que irá passar pela aprendizagem, com o intuito de compreender possíveis padrões e características determinísticas do conjunto de dados. Em seguida, os dados de teste passam pelo processo de classificação, com base no que o algoritmo aprendeu com o conjunto de treino. Por fim, é realizada a estimativa da performance do algoritmo, em que o principal elemento avaliado é a TA de classificação (*accuracy*).

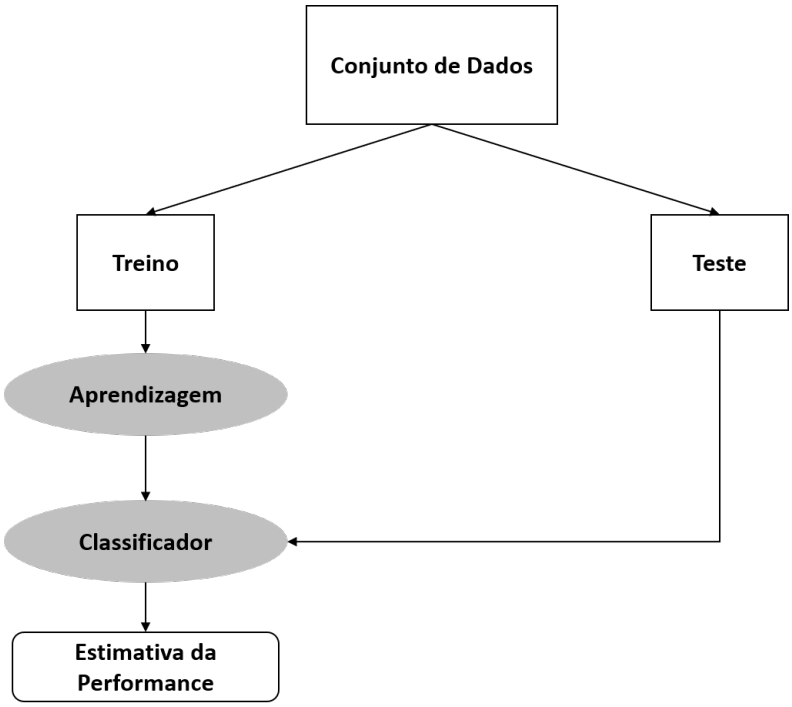


Figura 2.9: Esquema estrutural de treino e teste de dados.

Neste formato de avaliação, o algoritmo tem seu desempenho embasado somente pela classificação do conjunto de teste, ou seja, os dados separados para treino não são avaliados diretamente.

Para os algoritmos utilizados, foi aplicado um split de 60/40, que não é indicado para amostras pequenas. Em caso de 30 observações, por exemplo, 18 serão utilizadas para treino e 12 para teste, o

que é um número considerado pequeno para qualquer conclusão. Por esta razão, também foi utilizado o método da validação cruzada, que utiliza todas as observações para a testagem do algoritmo.

k-fold Cross Validation

A validação cruzada trata-se de um método de testagem de algoritmo em que todas as observações são consideradas para a análise de desempenho do aprendizado. Para tanto, os dados são separados em k grupos (*folds*), dos quais $k-1$ serão utilizados para treino. O procedimento é repetido por k vezes, conforme apresentado na Figura 2.10.

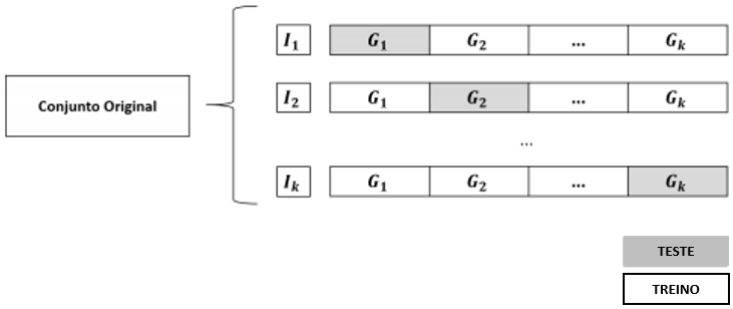


Figura 2.10: Esquema estrutural da Validação Cruzada.

2.2.3.11 Critérios de Desempenho

Os resultados da classificação são dispostos em uma estrutura denominada matriz de confusão, que apresenta um cruzamento entre os dados classificados pelo algoritmo e a classe de fato observada, conforme expresso na Figura 2.11.

		Classificação	
		Sim	Não
Classe Observada	Sim	VP	FN
	Não	FP	VN

Figura 2.11: Exemplo de uma matriz de confusão.

Internamente à matriz, tem-se quatro elementos de estruturais: os verdadeiros positivos (VP), os verdadeiros negativos (VN), os Falsos Positivos (FP) e os Falsos Negativos (FN).

Seja "Sim" a classificação para transações fraudulentas e "Não" para legítimas. Para os elementos estruturais da matriz de confusão, tem-se a seguinte relação:

- Verdadeiro Positivo: transações fraudulentas classificadas como fraudulentas;
- Falso Positivo: transações legítimas classificadas como fraudulentas;
- Verdadeiro Negativo: transações legítimas classificadas como legítimas;
- Falso Negativo: transações fraudulentas classificadas como legítimas.

A partir da matriz de confusão, são obtidos critérios de desempenho do algoritmo. Entre eles, destacam-se:

- Taxa de Acerto (TA): (VP + VN) / (VP + VN + FP + FN);
- Taxa de Falso Positivo: (FP) / (FP + VN);
- Taxa de Falso Negativo: (FN) / (FN + VP);
- Taxa de Verdadeiro Positivo (ou Sensitividade): (VP) / (VP + FN);
- Taxa de Verdadeiro Negativo (ou Especificidade): (VN) / (VN + FP).

A TA traduz-se pela capacidade global de classificação do algoritmo, independente da classe da transação. A taxa de falso positivo representa o percentual de instâncias classificadas como positivas, que deveriam ser rotuladas como negativas. Por outro lado, a taxa de falso negativo representa o percentual de observações classificadas como negativas, quando se tratam de instâncias com a presença da característica de interesse. A taxa de verdadeiro positivo representa a capacidade do classificador em rotular operações que apresentam a característica de interesse, enquanto a taxa de verdadeiro negativo evidencia a eficácia do algoritmo em predizer operações sem a característica de interesse (Bramer, 2007).

No conjunto de dados utilizado, a característica de interesse é a fraude. Desta forma, termo positivo refere-se a transações fraudulentas, enquanto o negativo remete a operações legítimas.

2.3 Resultados e Discussão

2.3.1 Cenário I

No Cenário I, a NP-MANOVA apresentou significância ($\alpha = 5\%$) a partir da amostra 270/270 (Tabela 2.6), revelando indícios de que as variáveis não apresentam uma boa capacidade de discriminação necessária para implementar métodos de classificação. Em geral, a significância estatística para NP-MANOVA é observada em amostras menores, como no trabalho de Fernandes *et al.* (2020), premiado em 2019 pela Editora Elsevier, em que foi observado um p-valor $< 0{,}01$ para o número de 25 repetições, contrastando com as 270 necessárias no Cenário I, valor aproximadamente onze vezes maior. No entanto, diante da presença de significância, ainda que em amostras superiores, a capacidade de diferenciação da NP-MANOVA demonstra ser viável proceder com a classificação através dos modelos estatísticos e dos algoritmos de ML.

Tabela 2.6: p-valor por tamanho amostral - NP-MANOVA (Cenário I).

Amostra	**p-valor**
15/15	0,6619
30/30	0,4880
60/60	0,1775
90/90	0,0799
180/180	0,1386
270/270	0,0011
492/492	0,0130

Nos cenários seguintes (II, III, IV), foi possível observar que a significância da NP-MANOVA aconteceu com tamanhos de amostras menores. Por exemplo, no Cenário IV, com 15 repetições por nível do fator fraude, obteve-se um p-valor (α) de 0,003.

Os métodos estatísticos utilizados para a classificação dos eventos foram a Regressão Logística, Análise Discriminante Linear, Análise Discriminante Quadrática e Mínimos Quadrados Parciais. Conforme a Tabela 2.7, observa-se que a Análise Discriminante Quadrática apresentou o melhor desempenho, atingindo uma TA de 62,50% para a amostra 492/492. Destaca-se também que o mesmo método obteve uma TA de 96,67%, porém para o menor tamanho amostral (15/15). O segundo melhor desempenho foi obtido por meio da Análise Discriminante Linear, seguida dos Mínimos Quadrados Parciais e Regressão Logística.

Tabela 2.7: Taxa de acerto da classificação dos algoritmos de ML (Cenário I).

Método	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Análise Discriminante Linear	76,67%	58,33%	65,00%	60,00%	61,11%	58,52%	57,01%
Análise Discriminante Quadrática	96,67%	83,33%	71,67%	67,22%	67,78%	62,41%	62,50%
Regressão Logística	63,33%	46,67%	50,00%	55,56%	55,56%	55,93%	54,17%
Mínimos Quadrados Parciais	73,33%	61,67%	65,00%	58,89%	60,28%	57,96%	56,50%

Com o auxílio da Figura 2.12, observa-se que a TA dos quatro métodos diminuiu ao passo em que se aumentou o tamanho amostral, apresentando estabilidade a partir da amostra 90/90. Além disso, também é perceptível a convergência entre os três métodos, em torno de uma TA de 60%. No caso da Análise Discriminante Quadrática, observou-se que a TA diminuiu conforme o aumento do tamanho amostral. Ainda assim, este foi o melhor método em todas as amostras.

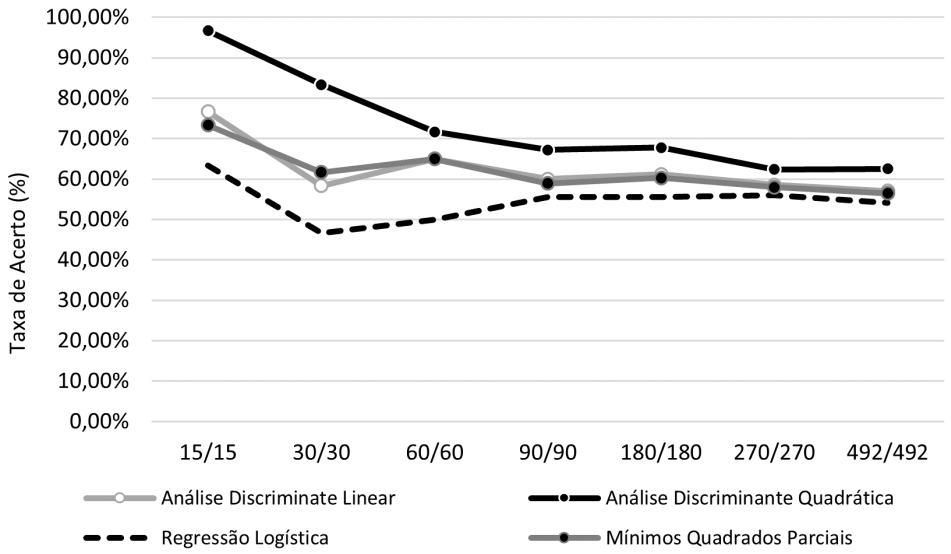


Figura 2.12: Taxa de acerto dos métodos estatísticos - Cenário I.

Em relação ao ML, segundo a Tabela 2.8, os algoritmos apresentaram melhora de desempenho, conforme aumento do tamanho amostral para a validação cruzada (100% dos dados). Random Forest

saiu de uma TA de 60% (4-fold Cross Validation) para 74,19%, na amostra com 984 observações. No split 60/40, a TA atingiu 75,13%, maior valor dentre os algoritmos de ML utilizados.

A TA do 4-fold Cross-Validation diminuiu pouco em relação ao split 60/40, o que implica uma boa possibilidade de generalização, o que diminui a chance de ocorrência de sobre-ajuste (overfitting)

Tabela 2.8: Taxa de acerto da classificação dos algoritmos de ML (Cenário I).

Critério de Teste	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Naive Bayes							
60/40	58,33%	62,5%	70,83%	51,39%	59,72%	52,78%	65,74%
10-fold	46,67%	53,33%	61,67%	57,78%	63,06%	57,04%	57,93%
4-fold	58,33%	48,33%	62,5%	52,78%	61,94%	57,04%	59,35%
Multilayer Perceptron							
60/40	75%	54,17%	66,67%	59,72%	59,72%	63,89%	58,38%
10-fold	53,33%	40%	47,5%	54,44%	61,94%	59,07%	60,67%
4-fold	46,67%	46,67%	55%	55%	65,83%	58,7%	57,42%
Support Vector Machine							
60/40	50%	45,83%	47,92%	51,39%	56,25%	57,87%	45,18%
10-fold	46,67%	38,33%	47,5%	52,22%	51,67%	54,44%	52,54%
4-fold	36,67%	53,33%	60,83%	50,56%	52,75%	56,11%	54,78%
K-NN							
60/40	58,33%	45,83%	58,33%	51,39%	64,58%	57,41%	52,33%
10-fold	53,33%	50%	37,74%	47,22%	62,22%	58,15%	52,34%
4-fold	53,33%	56,67%	42,5%	48,33%	58,89%	57,04%	52,54%
Random Forest							
60/40	58,33%	58,33%	70,83%	69,44%	78,47%	71,3%	75,13%
10-fold	63,33%	61,67%	75%	70,56%	70,56%	73,89%	72,87%
4-fold	60%	75%	69,17%	73,89%	73,89%	71,85%	74,19%
CART							
60/40	41,67%	41,67%	64,58%	65,28%	73,61%	64,81%	71,32%
10-fold	50%	51,67%	68,33%	64,44%	70%	67,78%	69,72%
4-fold	43,33%	63,33%	67,5%	62,78%	71,39%	67,59%	69,92%

Os algoritmos Naive Bayes, Multilayer-Perceptron, Support Vector Machine e K-NN apresentaram menor efeito no aumento do tamanho amostral. Considerando a amostra de 984 observações, a maior TA observada para estes algoritmos foi de 65,74% (Naive Bayes - split 60/40).

Em termos comparativos entre os métodos de validação, os algoritmos K-NN, Random Forest, CART e Multilayer Perceptron foram menos sensíveis entre as alternações de split 60/40, 10-fold Cross Validation e 4-fold Cross Validation, o que implica uma boa capacidade de generalização, ou seja, classificação de novas amostras, ainda que a TA seja baixa para o Cenário I. Por outro lado, Naive Bayes e Support Vector Machine apresentaram variações maiores, com base na amostra 492/492. A amplitude da TA foi de 7,81 pontos percentuais para Naive Bayes e 9,6 para Support Vector Machine, conforme a Tabela 2.9.

Tabela 2.9: Amplitude do taxa de acerto entre os métodos de avaliação (Cenário I, amostra 492/492).

Maior Taxa de Acerto	Menor Taxa de Acerto	Amplitude (p.p.)	Algoritmo
54,78%	45,18%	9,60	Support Vector Machine
65,74%	57,93%	7,81	Naive Bayes
60,67%	57,42%	3,25	Multilayer Perceptron
75,13%	72,87%	2,26	Random Forest
71,32%	69,72%	1,60	CART
52,54%	52,33%	0,21	K-NN

Por meio da Figura 2.13, é possível observar uma convergência entre os métodos de avaliação, conforme se aumenta o tamanho das amostras, ou seja, para o conjunto de dados do Cenário I, o método de avaliação tem maior influência em menores amostras.

Em termos comparativos, ao se analisar conjuntamente os melhores desempenhos dos métodos estatísticos e dos algoritmos de ML, obtém-se, para o Cenário I, que a Estatística obtém uma melhor performance em menores amostras, conforme observado na Figura 2.15. O algoritmo Random Forest, por sua vez, apresentou melhor desempenho para amostras maiores, com aparente estabilização em torno de 75%, tanto para o split 60/40, quanto para a 4-fold Cross Validation.

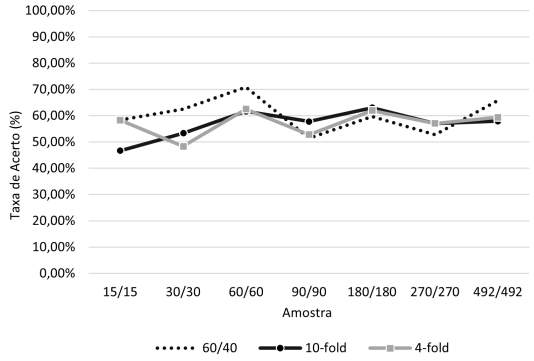
Por fim, em relação aos demais critérios de desempenho, o algoritmo Random Forest apresentou semelhança entre as taxas de Falsos Positivos e Falsos Negativos, que se mantiveram convergentes em 75% e 80% (Figura 2.15). Na Análise Discriminante Quadrática, embora os Falsos Positivos tenham sido menos frequentes do que no Random Forest, não foi observado um bom desempenho em relação aos Falsos Negativos.

Considerando o fato de a característica positiva representar a presença de fraude, casos Falsos Positivos estão associados a situações em que o algoritmo/método classificou uma transação como fraudulenta quando a mesma é legítima. De forma simétrica, os Falsos Negativos ocorrem quando o algoritmo/método classifica uma transação como legítima, sendo a mesma fraudulenta.

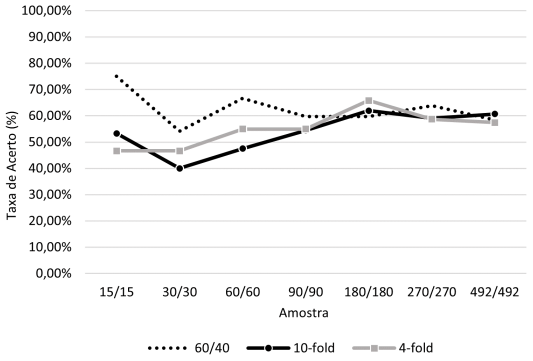
O resultado da (Figura 2.15) gera preocupação em relação ao índice de 1-Tx. Falsos Negativos para a Análise Discriminante. Com valores entre 30% e 40%, considerados de baixo desempenho, a probabilidade de se classificar uma operação fraudulenta como legítima, o que configura o erro mais crítico para este tipo de fraude, se estabelece em torno de 60% a 70%.

2.3.2 Cenário II

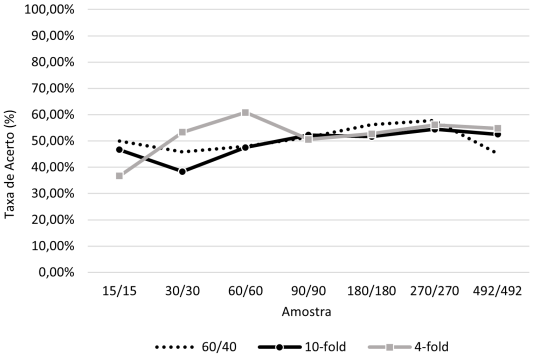
No segundo cenário, que representa o Cenário I com a inclusão de mais variáveis significativas, a NP-MANOVA apresentou diagnóstico de viabilidade a partir da amostra 30/30, com um p-valor <0,001 (Tabela 2.10), confirmando a possibilidade de prosseguimento com a aplicação das técnicas de classificação, mostrando semelhança ao trabalho de Fernandes *et al.* (2020), que obteve significância a partir de 25 repetições. No Cenário II, a significância foi observada a partir de 30 repetições.



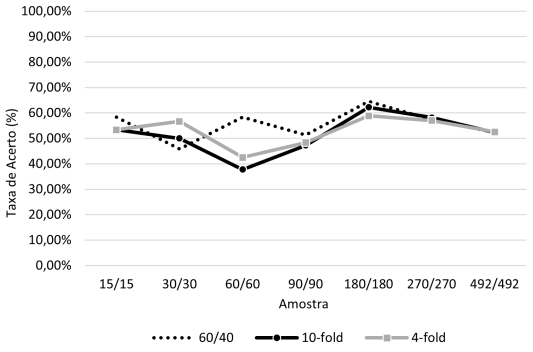
(a) Naive Bayes



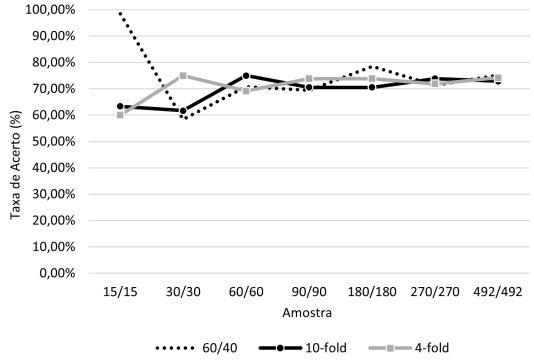
(b) Multilayer Perceptron



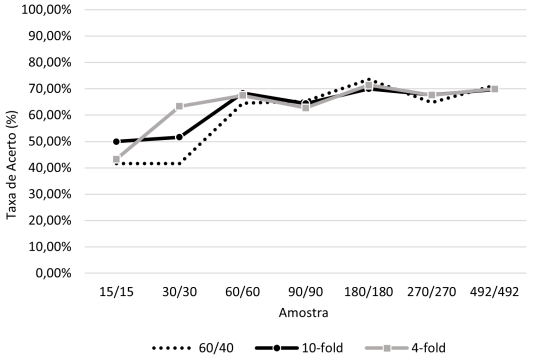
(c) Support Vector Machine



(d) K-NN



(e) Random Forest



(f) CART

Figura 2.13: Taxa de Acerto dos Algoritmos de ML por tamanho amostral (Cenário I).

Tabela 2.10: p-valor por tamanho amostral - NP-MANOVA (Cenário II).

Amostra	**p-valor**
15/15	0,6252
30/30	$<$ 0,0001
60/60	$<$ 0,0001
90/90	$<$ 0,0001
180/180	$<$ 0,0001
270/270	$<$ 0,0001
492/492	$<$ 0,0001

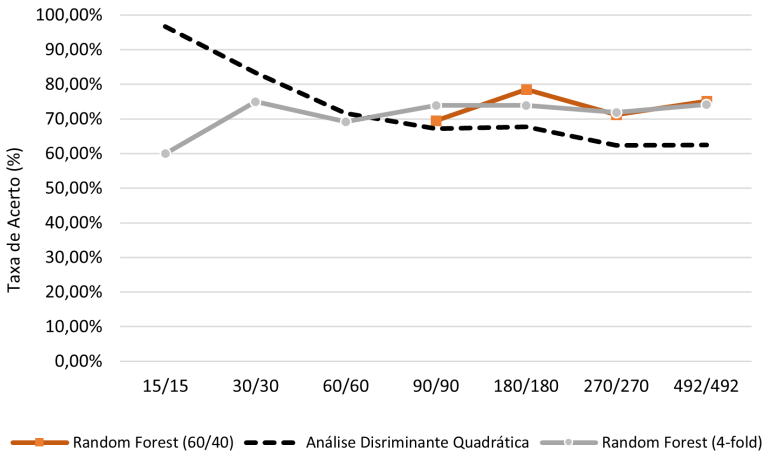


Figura 2.14: Comparativo das melhores Taxas de Acerto dos algoritmos de ML e dos métodos estatísticos (Cenário I).

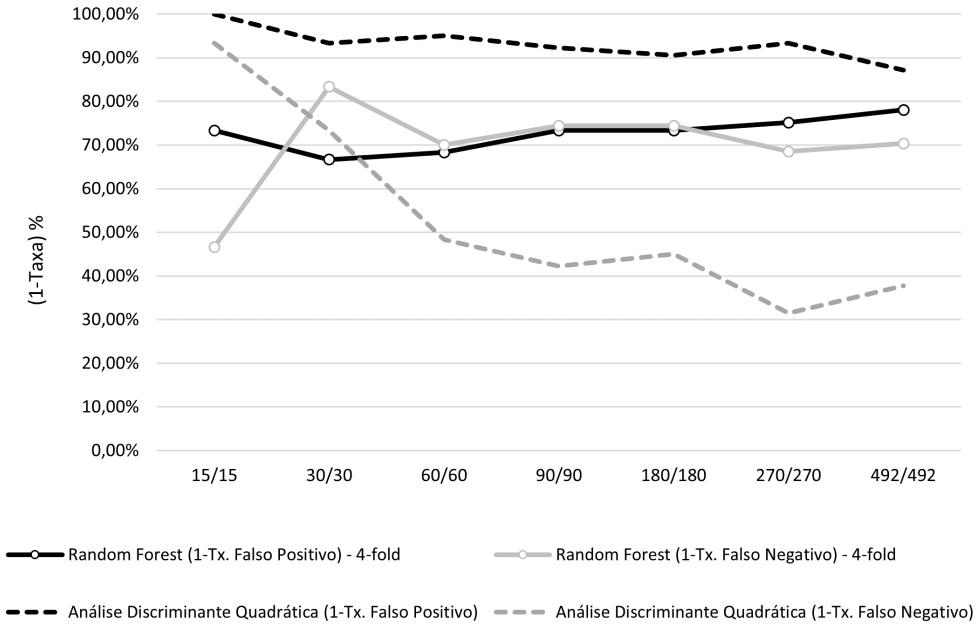


Figura 2.15: Taxas de Falso Positivo e Falso Negativo - Melhor método estatístico e melhor algoritmo de ML (Cenário I).

Neste sentido, conforme a Tabela 2.11, dentre os métodos estatísticos, a Regressão Logística apresentou maior TA, considerando a amostra 492/492. Em seguida, aparecem Mínimos Quadrados Parciais, Análise Discriminante Linear e Quadrática.

Tabela 2.11: Taxa de acerto da classificação dos algoritmos de ML (Cenário II).

Método	15/15	30/30	60/60	90/90	180/180	270/270	492/492
Análise Discriminante Linear	86,67%	93,33%	86,67%	83,89%	83,61%	81,30%	82,11%
Análise Discriminante Quadrática	93,33%	100,00%	90,83%	87,78%	88,06%	80,74%	81,40%
Regressão Logística	53,33%	80,00%	80,00%	79,44%	81,11%	81,85%	83,33%
Mínimos Quadrados Parciais	86,67%	85,00%	85,00%	85,29%	83,06%	81,30%	82,52%

Por meio da Figura 2.16, é possível observar que as Análises Discriminantes apresentaram um melhor comportamento em amostras menores, enquanto a Regressão Logística obteve um efeito positivo em relação ao aumento do tamanho amostral. Os Mínimos Quadrados Parciais apresentaram maior estabilidade em relação aos demais métodos. No entanto, é possível observar uma convergência das taxas de acerto em torno de 82%.

Em relação aos algoritmos de ML, ao se considerar a amostra 492/492, obtém-se a maior TA para o Random Forest, que atingiu 89,85%. CART, Multilayer Perceptron e Support Vector Machine também tiveram desempenhos superior a 80%. Naive Bayes e K-NN, no entanto, apresentaram taxas de acerto entre 68,29% e 75,63% (Tabela 2.12)

Destaca-se, também, que o Random Forest apresentou o melhor desempenho nos três métodos de avaliação: split 60/40, 10-fold Cross Validation e 4-fold Cross Validation.

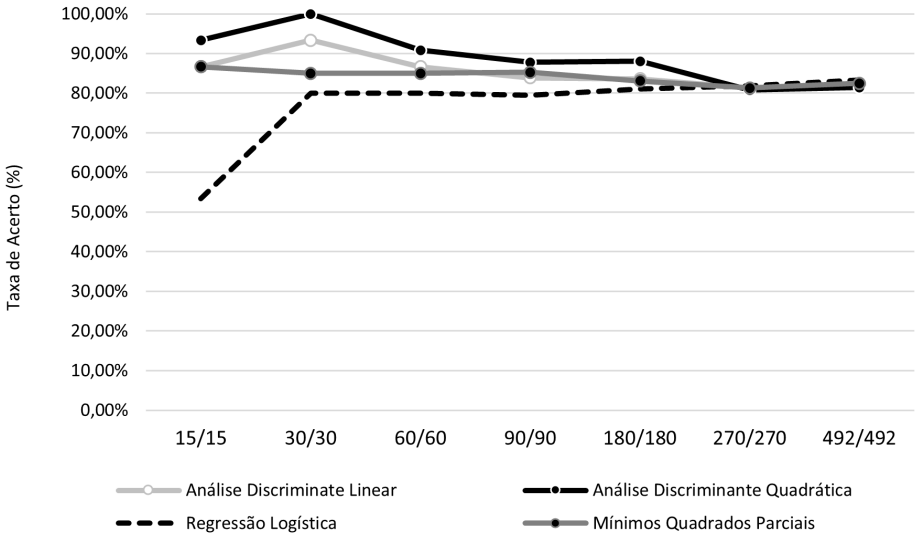


Figura 2.16: Taxa de acerto dos métodos estatísticos - Cenário II.

Tabela 2.12: Taxa de acerto da classificação dos algoritmos de ML (Cenário II).

Critério de Teste	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Naive Bayes							
60/40	75,00%	66,67%	72,92%	59,72%	73,61%	64,81%	72,34%
10 FOLD	56,67%	53,33%	65,00%	72,22%	69,44%	67,96%	68,29%
4 FOLD	73,33%	55,00%	65,00%	70,56%	71,11%	67,78%	68,29%
Multilayer Perceptron							
60/40	58,33%	58,33%	79,17%	76,39%	81,25%	85,19%	84,26%
10 FOLD	53,33%	45,00%	82,50%	81,67%	83,06%	82,59%	84,25%
4 FOLD	43,33%	76,67%	85,00%	81,11%	83,06%	81,48%	85,37%
Support Vector Machine							
60/40	75,00%	33,33%	77,08%	72,22%	79,17%	79,63%	84,26%
10 FOLD	63,33%	83,33%	80,00%	78,33%	80,83%	80,37%	82,62%
4 FOLD	53,33%	76,67%	82,50%	66,15%	80,83%	79,63%	82,32%
K-NN							
60/40	75,00%	58,33%	70,83%	63,89%	78,47%	68,98%	75,63%
10 FOLD	60,00%	75,00%	68,33%	67,22%	75,83%	72,78%	72,46%
4 FOLD	60,00%	76,67%	67,50%	68,89%	74,17%	72,04%	72,56%
Random Forest							
60/40	83,33%	83,33%	83,33%	76,39%	91,67%	87,50%	89,85%
10 FOLD	73,33%	90,00%	86,67%	80,56%	87,22%	89,07%	89,53%
4 FOLD	76,67%	88,33%	87,50%	76,11%	88,06%	88,33%	88,72%
CART							
60/40	83,33%	66,67%	83,33%	80,56%	84,72%	86,11%	86,04%
10 FOLD	76,67%	80,00%	82,50%	75,00%	78,61%	88,41%	83,64%
4 FOLD	70,00%	85,00%	78,33%	78,89%	80,83%	86,11%	84,65%

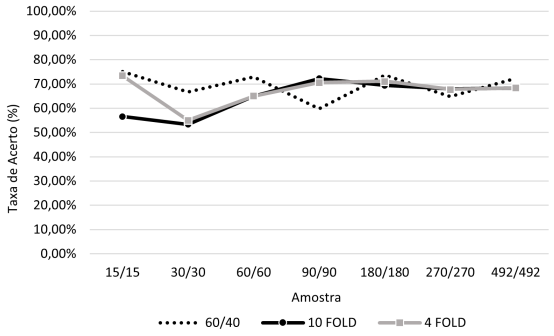
Considerando a amplitude da TA, os algoritmos de ML mais sensíveis ao método de avaliação foram Naive Bayes, K-NN e CART, conforme a Tabela 2.13. Nota-se, no entanto, que a sensibilidade foi menor do que observado no Cenário I, mostrando uma boa capacidade de generalização dos algoritmos de ML.

Tabela 2.13: Amplitude do taxa de acerto entre os métodos de avaliação (Cenário II, amostra 492/492).

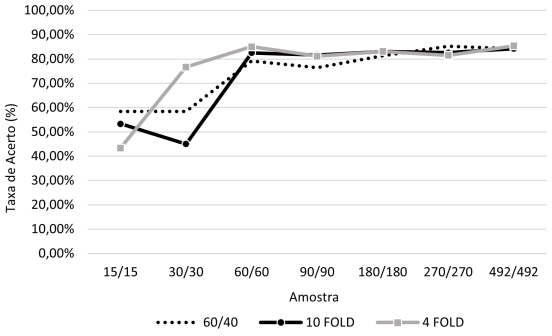
Maior Taxa de Acerto	**Menor Taxa de Acerto**	**Amplitude (p.p.)**	**Algoritmo**
72,34%	68,29%	4,05	Naive Bayes
75,63%	72,46%	3,17	K-NN
86,04%	83,64%	2,40	CART
84,26%	82,32%	1,94	Support Vector Machine
85,37%	84,23%	1,14	Multilayer Perceptron
89,85%	88,72%	1,13	Random Forest

Com base na Figura 2.17, observa-se que algoritmos como Random Forest e o CART tiveram taxas de acerto mais estáveis, considerando tanto os métodos de avaliação, quanto o tamanho amostral. Os algoritmos Naive Bayes, K-NN e Multilayer Perceptron apresentaram maior diferença entre os métodos de avaliação para amostras menores.

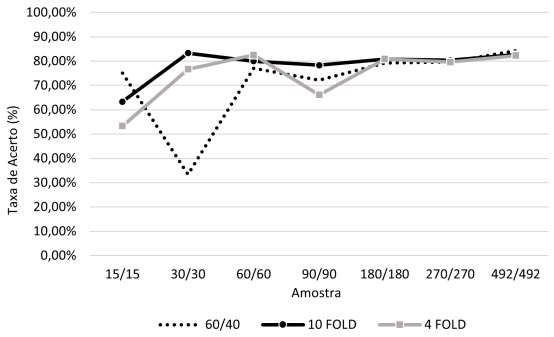
Em geral, diferentemente do que foi observado no Cenário I, os algoritmos de ML apresentaram maior estabilidade no Cenário II, o que está associado à melhor qualidade dos dados. Com dados mais significativos, é necessário um tamanho amostral menor para obter melhores classificações.



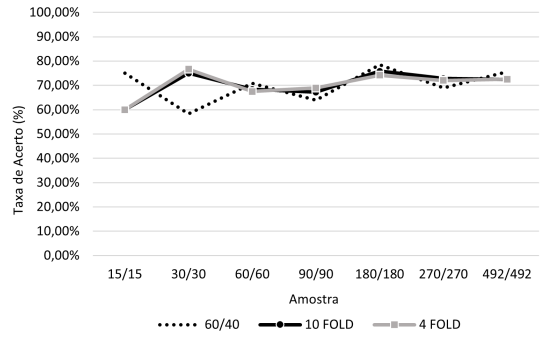
(a) Naive Bayes



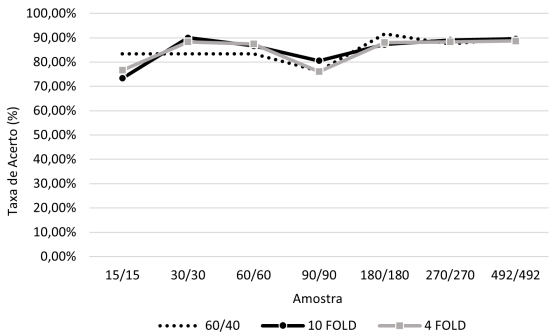
(b) Multilayer Perceptron



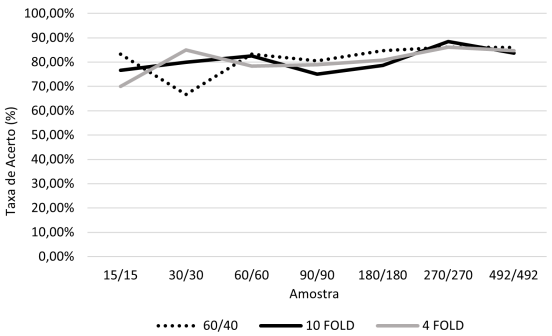
(c) Support Vector Machine



(d) K-NN



(e) Random Forest



(f) CART

Figura 2.17: Taxa de Acerto dos Algoritmos de ML por tamanho amostral (Cenário II).

Ao se analisar os desempenhos da Estatística e do ML, foram tomados como base o Random Forest, a Análise Discriminante Linear e a Regressão Logística. Conforme a Figura 2.18, observa-se, assim como no Cenário I, que a Estatística obteve melhor desempenho em amostras menores, por meio da Análise Discriminante Linear, enquanto o ML apresenta melhor TA em amostras maiores.

Outro ponto de destaque é a Regressão Logística, que apresentou comportamento mais semelhante aos algoritmos de ML, embora seja um método classificado como estatístico.

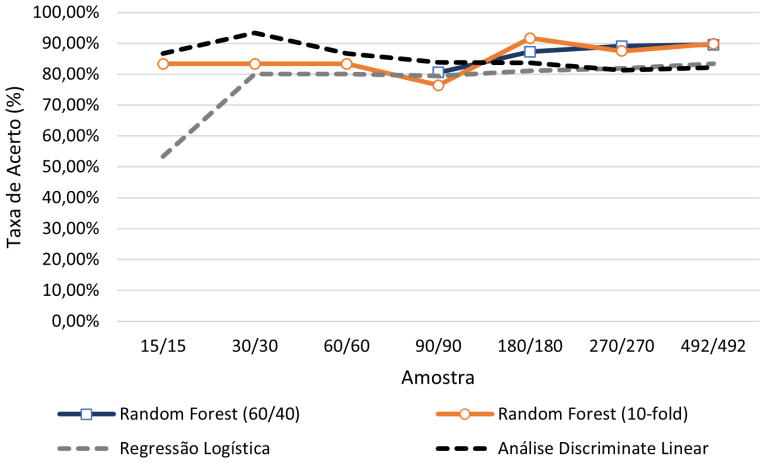


Figura 2.18: Comparativo das melhores Taxas de Acerto dos algoritmos de ML e dos métodos estatísticos (Cenário II).

Por fim, considerando as taxas de Falsos Positivos e Falsos Negativos, observa-se que o Random Forest obteve um melhor desempenho quando se considera os dois erros conjuntamente (Figura 2.19). Diante da criticidade da ocorrência de Falsos Negativos, observou-se que o Regressão Logística obteve desempenho muito baixo, com taxas inferiores a 70%, implicando em se cometer o pior erro para esta pesquisa (aceitar um transação fraudulenta como legítima, causando perdas financeiras e grande insatisfação dos clientes).

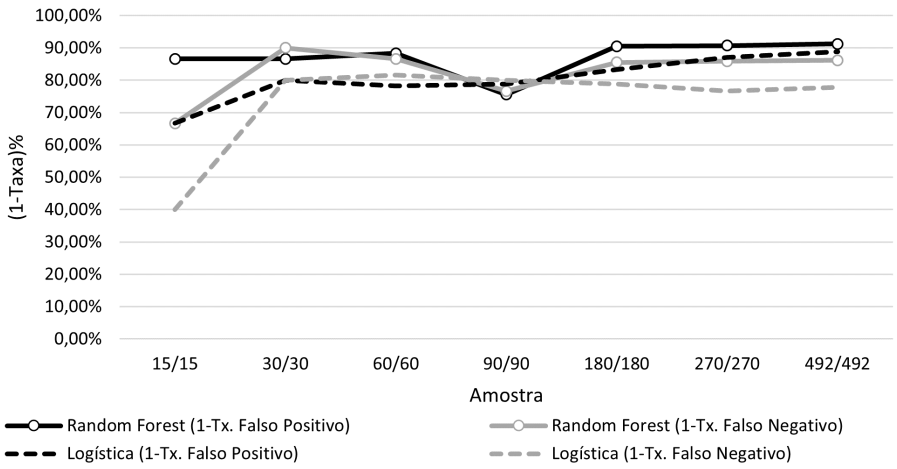


Figura 2.19: Taxas de Falso Positivo e Falso Negativo - Melhor método estatístico e melhor algoritmo de ML (Cenário II).

2.3.3 Cenário III

No Cenário III, composto por todas as variáveis, independente da significância, também observou-se, por meio da NP-MANOVA, viabilidade para a classificação a partir de 30 repetições, como no Cenário II (p-valores abaixo de 0,01). Na amostra 15/15, devido à quantidade de variáveis e ao baixo volume de observações (insuficientes graus de liberdade do resíduo), não foi possível obter as estatísticas da NP-MANOVA. As demais amostras, no entanto, revelaram-se altamente significativas (Tabela 2.14).

Tabela 2.14: p-valor por tamanho amostral - NP-MANOVA (Cenário III).

Amostra	**p-valor**
15/15	NA
30/30	$< 0{,}0001$
60/60	$< 0{,}0001$
90/90	$< 0{,}0001$
180/180	$< 0{,}0001$
270/270	$< 0{,}0001$
492/492	$< 0{,}0001$

Os quatro métodos estatísticos utilizados obtiveram taxas de acerto semelhantes, considerando a amostra com 984 observações. A Análise Discriminante Quadrática obteve 92,89% de classificações corretas, seguida da Análise Discriminante Linear e Regressão Logística, ambas com 92,78%, além dos Mínimos Quadrados Parciais, com 89,43%, conforme a Tabela 2.15.

Tabela 2.15: Taxa de acerto da classificação dos algoritmos de ML (Cenário III).

Método	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Análise Discriminante Linear	100,00%	100,00%	95,00%	95,00%	90,28%	91,67%	92,78%
Análise Discriminante Quadrática	90,00%	100,00%	97,50%	96,11%	92,22%	93,33%	92,89%
Regressão Logística	63,33%	80,00%	81,67%	87,22%	90,00%	92,78%	92,78%
Mínimos Quadrados Parciais	90,00%	95,00%	91,67%	90,56%	89,17%	89,63%	89,43%

Nas amostras menores, as Análises Discriminantes obtiveram o melhor desempenho, com desempenhos semelhantes (Figura 2.20). Com o aumento do tamanho amostral, no entanto, observou-se, assim como nos cenários anteriores, uma convergência entre os três métodos, cujas taxas de acerto se estabilizaram em torno de 93%.

Em relação aos algoritmos de ML, todos apresentaram taxas de acerto superiores a 90% em pelo menos uma das instâncias de avaliação. Os melhores desempenhos foram observados no Random Forest (94,92% - split 60/40), Random Forest (94,11% - 10-fold cross validation) e Support Vector Machine (93,61% - split 60/40). Para o método 4-fold cross validation, a maior TA também foi obtida pelo Random Forest, com 93,50%. Destaca-se, também, que o algoritmo Naive Bayes não obteve convergência para a amostra 15/15, o que impossibilitou a elaboração do classificador (Tabela 2.16).

Com o auxílio da Tabela 2.17, observou-se que os algoritmos de ML mais sensíveis ao método de avaliação foram K-NN, com uma amplitude de TA equivalente a 3,51%, CART, que atingiu 1,82% e Naive Bayes, com 1,74%.

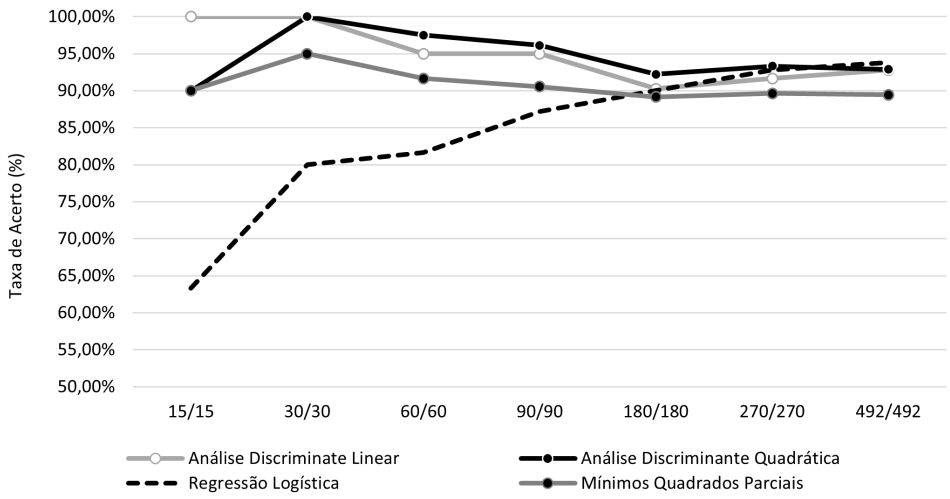


Figura 2.20: Taxa de acerto dos métodos estatísticos - Cenário III.

Tabela 2.17: Amplitude do taxa de acerto entre os métodos de avaliação (Cenário III, amostra 492/492).

Maior Taxa de Acerto	**Menor Taxa de Acerto**	**Amplitude (p.p.)**	**Algoritmo**
92,64%	89,13%	3,51%	K-NN
92,17%	90,36%	1,82%	CART
90,36%	88,62%	1,74%	Naive Bayes
94,92%	93,50%	1,43%	Random Forest
93,91%	92,68%	1,23%	Support Vector Machine
93,65%	92,68%	0,97%	Multilayer Perceptron

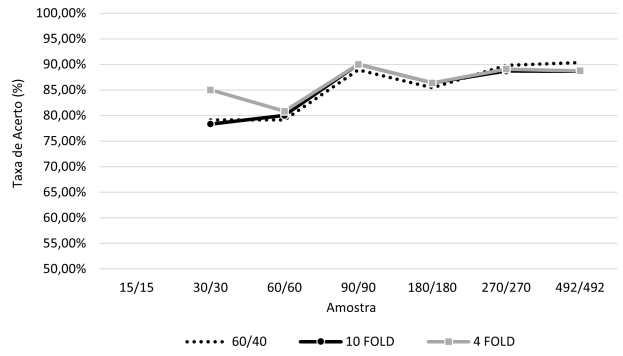
A Figura 2.21 permite observar que os algoritmos Support Vector Machine, Multilayer Perceptron e Random Forest apresentaram maior estabilidade dentre os tamanhos amostrais e os métodos de avaliação. Por outro lado, Naive Bayes, K-NN e CART obtiveram melhores desempenhos em amostras maiores. De forma geral, no entanto, os seis algoritmos de ML apresentaram um comportamento de convergência conforme aumento do número de observações.

Em termos comparativos, a Figura 2.22 apresenta os melhores algoritmos de ML e método estatístico, com base na TA. Observa-se que os desempenhos foram semelhantes, com destaque para a Análise Discriminante Quadrática nos conjuntos menores, obtendo 100%, 97,50% e 96,11% nas amostras 30/30, 60/60 e 90/90, respectivamente. Com o aumento do tamanho amostral, no entanto, a Análise Discriminante apresentou queda em sua TA, ao passo que os algoritmos de ML melhoraram seus desempenhos. Apesar de estarem próximos, observa-se que o Random Forest apresentou TA sensivelmente maior, atingindo 94,92%, ante 92,89% da Análise Discriminante Quadrática.

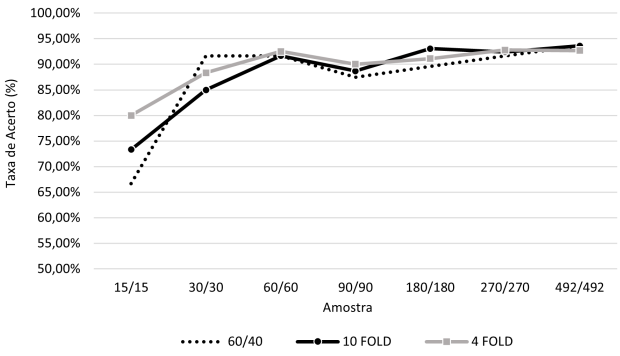
Por fim, ao se considerar as taxas de Falsos Positivos e Falsos Negativos da Análise Discriminante Quadrática e do Random Forest, observou-se comportamento semelhante para ambos os métodos. Com auxílio da Figura 2.23, a taxa de Falso Positivo, expressa indiretamente pelas linhas pretas caminham lado a lado em ambos os métodos. Os Falsos Negativos, no entanto, apresentam diferença considerável em amostras menores, indicando melhor classificação da Análise Discriminante Quadrática até o conjunto 180/180. A partir da amostra 270/270, o erro foi menor no algoritmo Random Forest.

Tabela 2.16: Taxa de acerto da classificação dos algoritmos de ML (Cenário III).

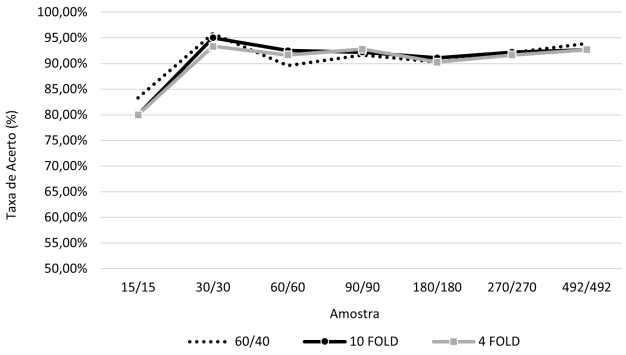
Critério de Teste	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Naive Bayes							
60/40	-	79,17%	79,17%	88,89%	85,42%	89,81%	90,36%
10 FOLD	-	78,33%	80,00%	90,00%	86,39%	88,70%	88,62%
4 FOLD	-	85,00%	80,83%	90,00%	86,39%	89,07%	88,72%
Multilayer Perceptron							
60/40	66,67%	91,67%	91,67%	87,50%	89,58%	91,67%	93,65%
10 FOLD	73,33%	85,00%	91,67%	88,67%	93,06%	92,41%	93,60%
4 FOLD	80,00%	88,33%	92,50%	90,00%	91,11%	92,78%	92,68%
Support Vector Machine							
60/40	83,33%	95,83%	89,58%	91,67%	90,28%	92,13%	93,91%
10 FOLD	80,00%	95,00%	92,50%	92,22%	91,11%	92,22%	92,68%
4 FOLD	80,00%	93,33%	91,67%	92,78%	90,28%	91,67%	92,68%
K-NN							
60/40	91,67%	79,17%	89,58%	81,94%	86,81%	90,74%	92,64%
10 FOLD	76,67%	88,33%	85,00%	88,33%	90,00%	89,26%	89,13%
4 FOLD	70,00%	85,00%	87,50%	88,33%	89,17%	89,26%	89,63%
Random Forest							
60/40	100,00%	95,83%	89,58%	91,67%	90,28%	92,13%	94,92%
10 FOLD	90,00%	95,00%	92,50%	94,44%	91,94%	94,44%	94,11%
4 FOLD	90,00%	95,00%	94,17%	94,44%	92,22%	94,07%	93,50%
CART							
60/40	91,67%	95,83%	85,42%	83,33%	86,11%	91,12%	90,36%
10 FOLD	70,00%	93,33%	88,33%	90,00%	91,11%	91,48%	92,17%
4 FOLD	76,67%	95,00%	90,00%	89,44%	90,56%	90,19%	91,87%



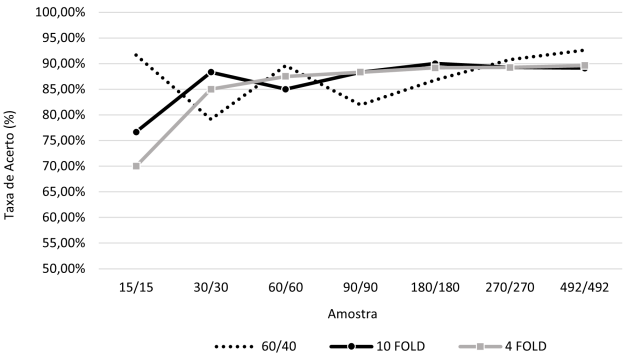
(a) Naive Bayes



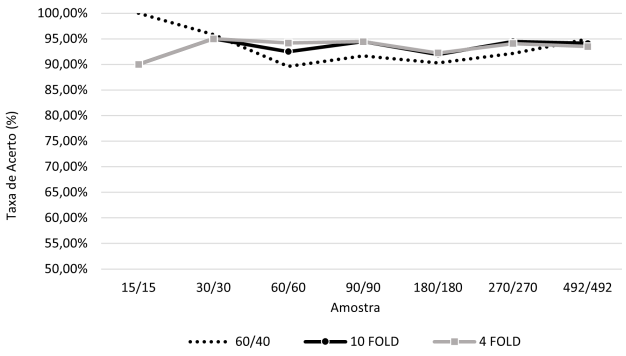
(b) Multilayer Perceptron



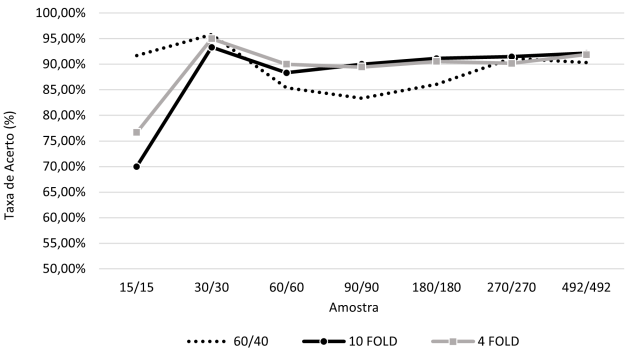
(c) Support Vector Machine



(d) K-NN



(e) Random Forest



(f) CART

Figura 2.21: Taxa de Acerto dos Algoritmos de ML por tamanho amostral (Cenário III).

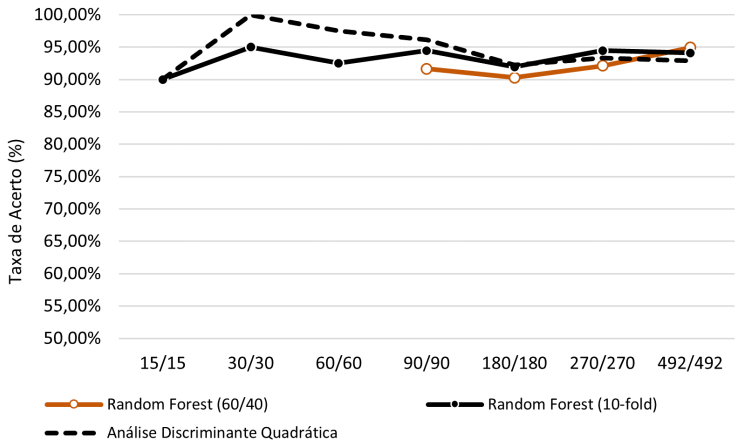


Figura 2.22: Comparativo das melhores Taxas de Acerto dos algoritmos de ML e dos métodos estatísticos (Cenário III).

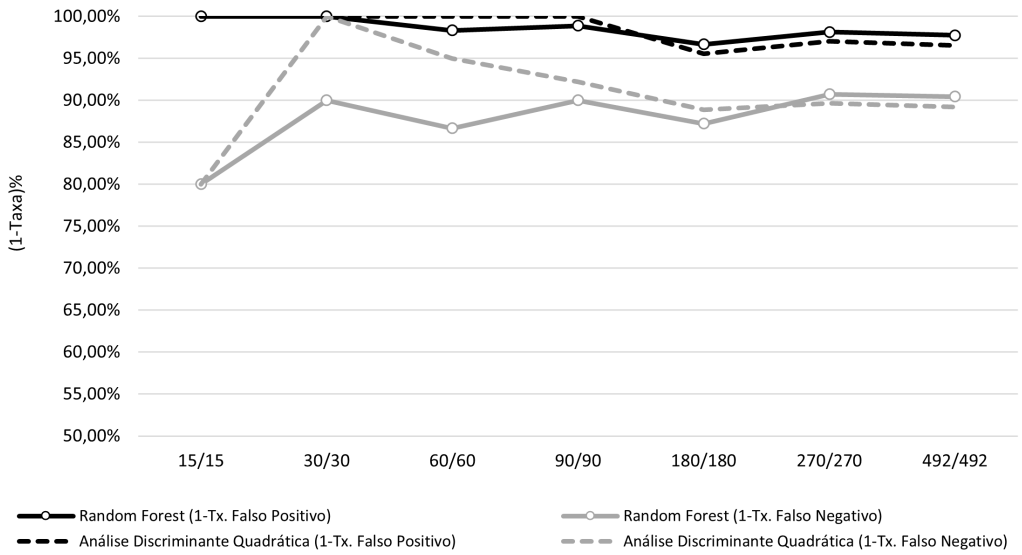


Figura 2.23: Taxas de Falso Positivo e Falso Negativo - Melhor método estatístico e melhor algoritmo de ML (Cenário III).

2.3.4 Cenário IV

O Cenário IV, composto apenas por variáveis significativas, apresentou, por meio da NP-MANOVA, viabilidade de análise, já a partir da amostra 15/15, que obteve p-valor de 0,0033 ($< \alpha = 1\%$). Nas demais amostras, os valores foram altamente significativos, com p-valor $<$ 0,0001, indicando viabilidade de estimação de algoritmos classificatórios para este conjunto de dados (Tabela 2.18).

Tabela 2.18: p-valor por tamanho amostral - NP-MANOVA (Cenário IV).

Amostra	p-valor
15/15	0,0033
30/30	$<$ 0,0001
60/60	$<$ 0,0001
90/90	$<$ 0,0001
180/180	$<$ 0,0001
270/270	$<$ 0,0001
492/492	$<$ 0,0001

Em relação aos métodos estatísticos, a Regressão Logística, com 94,21% de TA, apresentou o melhor desempenho na amostra 492/492. Em seguida, aparece o método dos Mínimos Quadrados Parciais, com 88,92%. As Análises Discriminantes tiveram desempenho inferior àquele observado no Cenário III, com taxas de acerto de 82,11% e 81,40%, para a Linear e Quadrática, respectivamente, conforme a Tabela 2.19.

Tabela 2.19: Taxa de acerto da classificação dos algoritmos de ML (Cenário IV).

Método	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Análise Discriminante Linear	100,00%	95,00%	86,67%	83,89%	83,61%	81,30%	82,11%
Análise Discriminante Quadrática	96,67%	100,00%	90,83%	87,78%	88,06%	80,74%	81,40%
Regressão Logística	83,33%	76,67%	90,83%	89,44%	90,56%	93,33%	94,21%
Mínimos Quadrados Parciais	90,00%	93,33%	87,50%	90,56%	89,17%	88,70%	88,92%

Por meio da Figura 2.24, é possível observar que as Análises Discriminantes e os Mínimos Quadrados Parciais apresentaram melhores taxas de acerto em amostras menores, pareando com a Regressão Logística em amostras intermediárias. A partir da amostra 180/180, a Regressão Logística obteve desempenho superior.

Os algoritmos de ML apresentaram, de modo geral, bons desempenhos. Apenas Naive Bayes obteve TA inferior a 90% em pelo menos um dos métodos de avaliação. Ainda assim, na 4-fold Cross Validation, 90,24% dos eventos foram classificados de forma correta.

Os três melhores desempenhos foram obtidos com o algoritmo Random Forest, que apresentou taxas de acerto de 95,18% no split 60/40, 94,51% na 10-fold cross validation e 93,70% na 4-fold. Em seguida, Multilayer Peceptron, com 93,65% (split 60/40) e K-NN, com 93,40%, também no split 60/40.

A Tabela 2.21 permite observar que os algoritmos de ML foram menos sensíveis ao método de avaliação, com menores amplitudes observadas no grupo amostral 492/492. Naive Bayes foi o que mais variou, com 1,92 pontos percentuais entre a maior e menor TA, seguido de Random Forest e K-NN, com 1,48% e 0,92%.

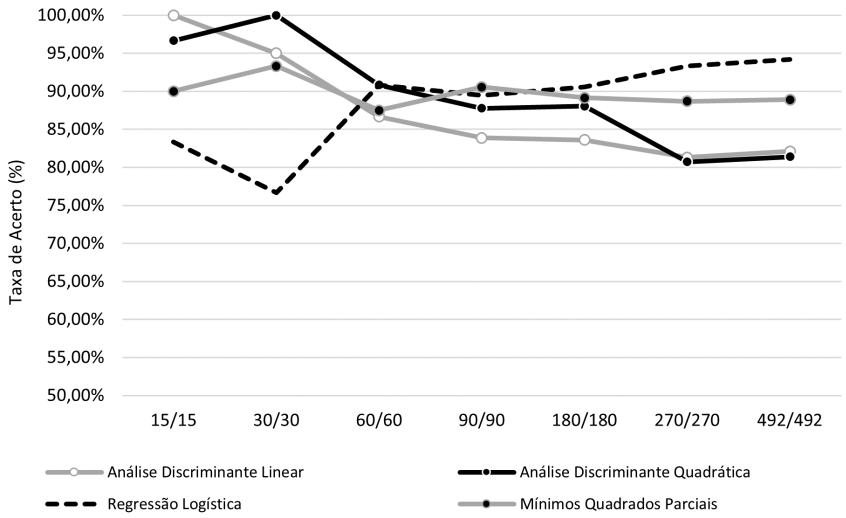


Figura 2.24: Taxa de acerto dos métodos estatísticos - Cenário IV.

Tabela 2.21: Amplitude do taxa de acerto entre os métodos de avaliação (Cenário IV, amostra 492/492)

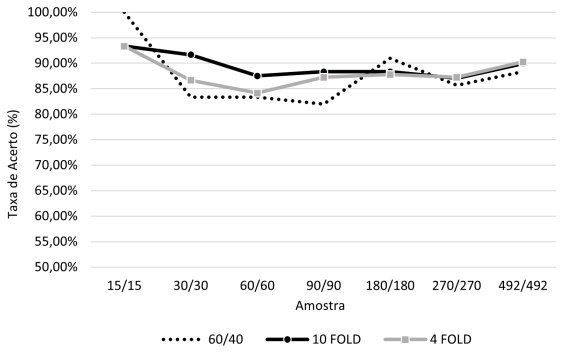
Maior Taxa de Acerto	**Menor Taxa de Acerto**	**Amplitude (p.p.)**	**Algoritmo**
90,24%	88,32%	1,92%	Naive Bayes
95,18%	93,70%	1,48%	Random Forest
93,40%	92,48%	0,92%	KNN
93,65%	92,89%	0,76%	Support Vector Machine
92,89%	92,17%	0,72%	CART
93,39%	92,89%	0,50%	Multilayer Perceptron

Os gráficos das taxas de acerto dos algoritmos de ML, expressos na Figura 2.25, permitem observar poucas oscilações no Naive Bayes, Multilayer Perceptron, Support Vector Machine e Random Forest. K-NN e CART apresentaram variações nas amostras menores, mas se estabilizaram a partir do grupo 60/60. De modo geral, os algoritmos de ML convergiram para uma TA acima de 90%.

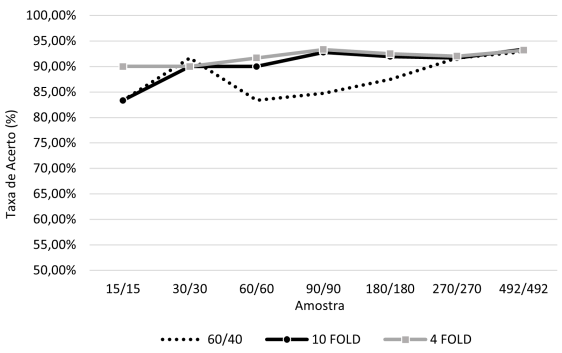
No comparativo entre o melhores métodos, a Estatística, representada pela Regressão Logística, apresentou desempenho próximo do ML, expresso através do Random Forest. Na Figura 2.26, observa-se que, diferentemente dos cenários anteriores, o Random Forest obteve o melhor desempenho nas amostras menores. No entanto, conforme se aumentou o tamanho amostral, os métodos convergiram para uma TA próxima, entre 94% e 95%.

Por fim, ao se analisar as taxas de Falso Positivo e Falso Negativo dos melhores classificadores (Figura 2.27), também se obteve convergência entre os comportamentos da Estatística e do ML. Nas amostras maiores, embora os valores estejam próximos, o Random Forest apresentou desempenho sensivelmente superior em relação aos Falsos Positivos, enquanto a Regressão Logística apresentou melhor aproveitamento em Falsos Negativos, comportamento que se pretende privilegiar. Ainda assim, destaca-se que ambos os métodos apresentaram melhor desempenho em falsos positivos, em detrimento de falsos negativos, demonstrando problemas em relação ao pior tipo de erro.

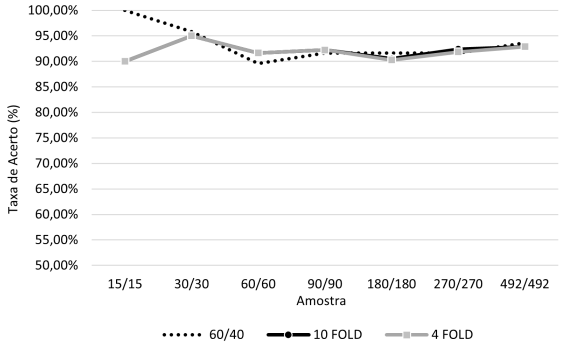
Considerando o algoritmo Random Forest, no split 60/40, em que se obteve uma TA de 95,18%, a matriz de confusão apresenta a validação de 394 observações, que representam 40% da amostra 492/492,



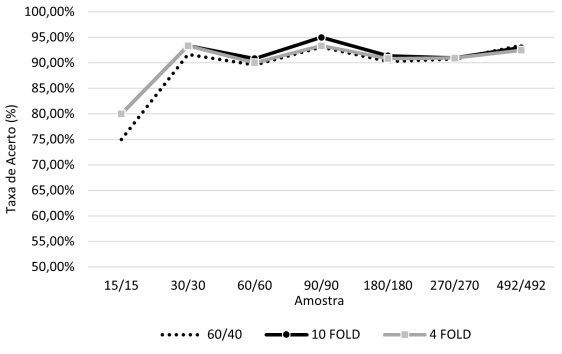
(a) Naive Bayes



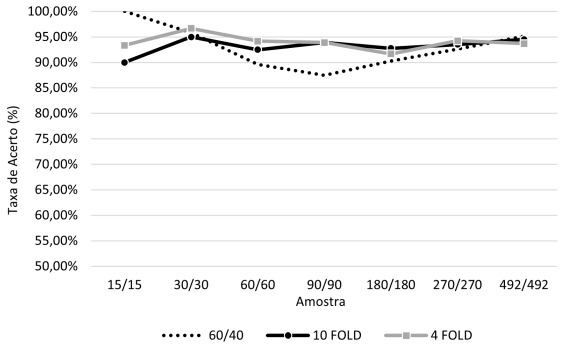
(b) Multilayer Perceptron



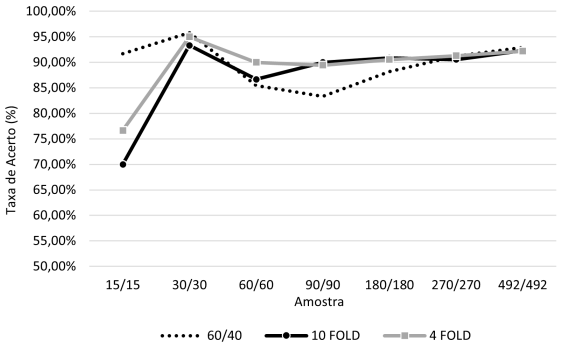
(c) Support Vector Machine



(d) K-NN



(e) Random Forest



(f) CART

Figura 2.25: Taxa de Acerto dos Algoritmos de ML por tamanho amostral (Cenário IV).

utilizados para teste. Desta forma, conforme a Figura 2.28, das 394 observações, 161 foram corretamente classificadas como fraudes (Verdadeiro Positivo), enquanto 17 foram incorretamente classificadas como legítimas (Falso Negativo). Por outro lado, 214 observações foram corretamente classificadas como legítimas (Verdadeiro Negativo) e 2 apresentaram-se como Falsos Positivos, ou seja, foram classificadas como fraude, porém eram transações legítimas.

Desta forma, considerando os itens descritos na seção 2.2.3.11, obtém-se os seguintes resultados:

- Taxa de Acerto = (VP + VN) / (VP + VN + FP + FN) = (214+161)/(214+161+17+2) = 95, 18%;

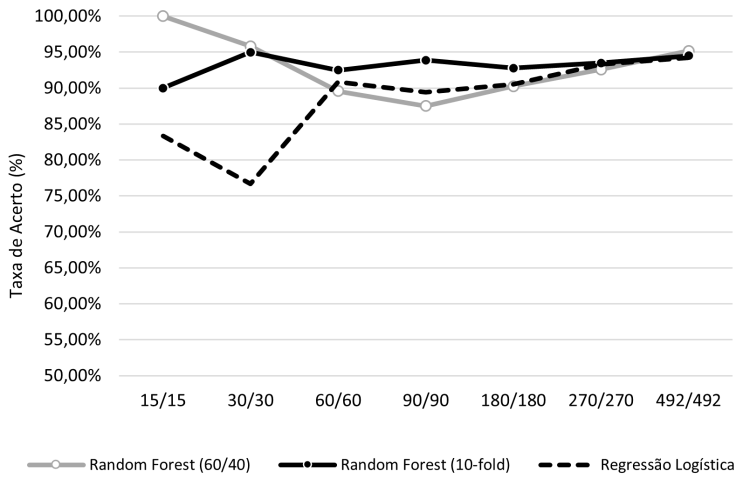


Figura 2.26: Comparativo das melhores Taxas de Acerto dos algoritmos de ML e dos métodos estatísticos (Cenário IV).

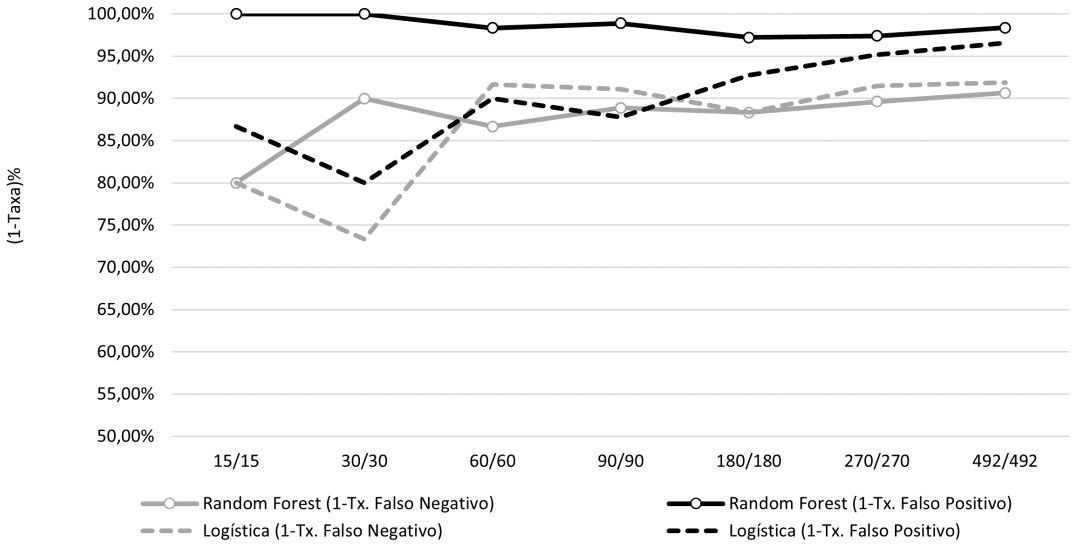


Figura 2.27: Comparativo das melhores Taxas de Acerto dos algoritmos de ML e dos métodos estatísticos (Cenário IV).

- Taxa de Falso Positivo = (FP) / (FP + VN) = $(2)/(2+214) = 0,93\%$;
- Taxa de Falso Negativo = (FN) / (FN + VP) = $(17)/(17+161) = 9,55\%$;
- Taxa de Verdadeiro Positivo (ou Sensitividade) = (VP) / (VP + FN) = $(161)/(161+17) = 90,45\%$;
- Taxa de Verdadeiro Negativo (ou Especificidade): (VN) / (VN + FP) = $(214)/(214+2) = 99,07\%$

Na taxa de acerto, tem-se a proporção de classificações corretas, independente de fraude ou legitimidade. Desta forma, o Random Forest obteve, em geral, um desempenho de 95,18%. A taxa de

		Classificação	
		Sim	Não
Classe Observada	Sim	161	17
	Não	2	214

Figura 2.28: Matriz de Confusão do Algoritmo Random Forest para Split 60/40 - Cenário IV.

falso positivo, equivalente a 0,93%, representa o percentual de instâncias legítimas classificadas como fraudulentas, enquanto a taxa de falso negativo (9,55%) indica o percentual de transações fraudulentas classificadas como legítimas.

Os 99,07% de especificidade indicam que o algoritmo possui uma boa capacidade em predizer as operações legítimas, enquanto a sensitividade, no total de 90,45%, demonstra que o algoritmo é menos eficaz em predizer situações de fraude.

Nesse sentido, com base nos critérios calculados, observa-se que o Random Forest apresentou melhor desempenho ao classificar operações legítimas, com menores taxas de erro. A taxa de verdadeiro positivo indica que a cada 100 transações fraudulentas, o algoritmo seria capaz, com base neste conjunto de dados, de classificar corretamente 90. Por outro lado, para as transações legítimas, a cada 100 operações, 1 seria classificada de forma incorreta.

Resultados Complementares

Além dos resultados obtidos no Cenário IV, foram realizadas análises complementares, em que se destacam: redes neurais com duas camadas de neurônios, gráfico de variáveis canônicas e dimensionamento de tamanho amostral para ML.

Ao parametrizar a rede neural, por meio do Multilayer Perceptron, para a operação em duas camadas ocultas (Figura 2.29), o que configura-se como o caminho inicial para o aprendizado profundo (deep learning), o algoritmo apresenta sensível melhora de desempenho. No split 60/40, a TA obtida para uma camada de neurônios foi de 92,89%. Ao se inserir uma nova camada, a TA atingiu 93,65%. Para a 4-fold Cross-Validation, a TA foi equivalente a 92,89% com duas camadas de neurônios. Evidencia-se, portanto, que a inclusão da segunda camada de neurônios, para o split 60/40, incrementa o desempenho do classificador. Ao se comparar com a 4-fold Cross Validation, observa-se que, devido à proximidade entre as taxas de acerto, o método apresenta uma boa capacidade de generalização[12].

Por fim, o gráfico de análise canônica com biplot (Figura 2.30) representa, por meio de vetores, as variáveis com maior poder de discriminação por classe. Desta forma, é possível observar que as variáveis V7, V18, V4 e V28 apresentam maior valor absoluto para observações fraudulentas, enquanto as variáveis V14, V8, V1 e V17 possuem maior valor absoluto para operações legítimas. As demais variáveis se aproximam de um ângulo de 90°, de modo a terem pouco poder de discriminação.

Frequentemente, dois dos problemas enfrentados pelas organizações são as dificuldades operacionais e os custos envolvidos no processo de coleta de dados. Desta forma, é de grande importância buscar modelos e cenários que permitam tomadas de decisão assertivas com a menor quantidade de dados necessária. Considerando o Random Forest, algoritmo que obteve o melhor desempenho geral em relação

[12]Quando se aumenta a quantidade de camadas de neurônios, frequentemente se observa problemas de sobreajuste (overfitting)

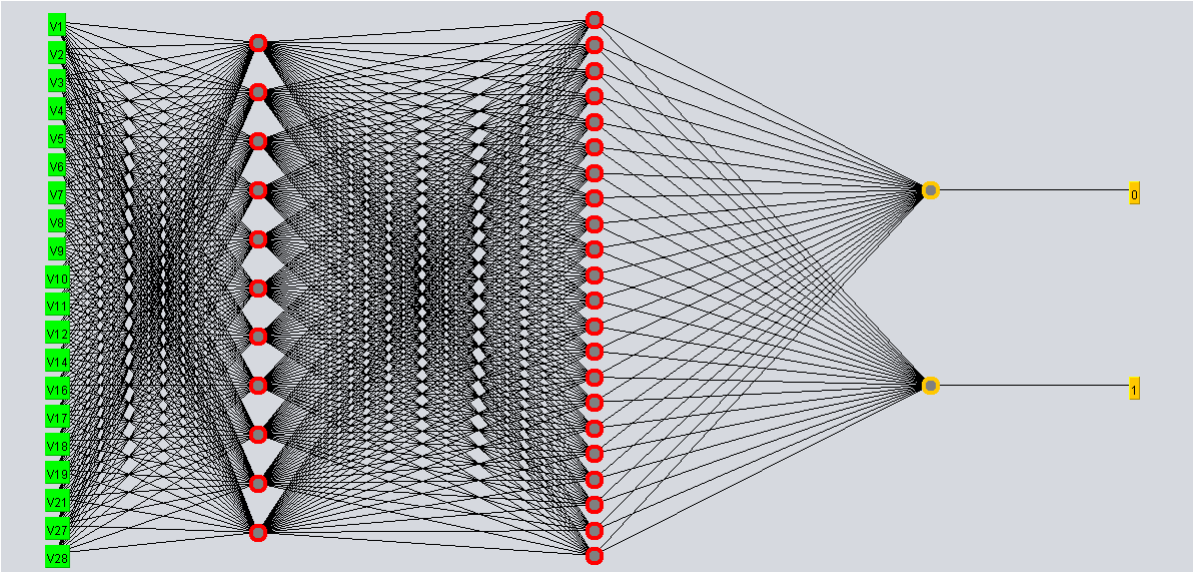


Figura 2.29: Rede Neural com duas camadas de neurônios (Multilayer Perceptron - Split 60/40 - amostra 492/492) - Cenário IV.

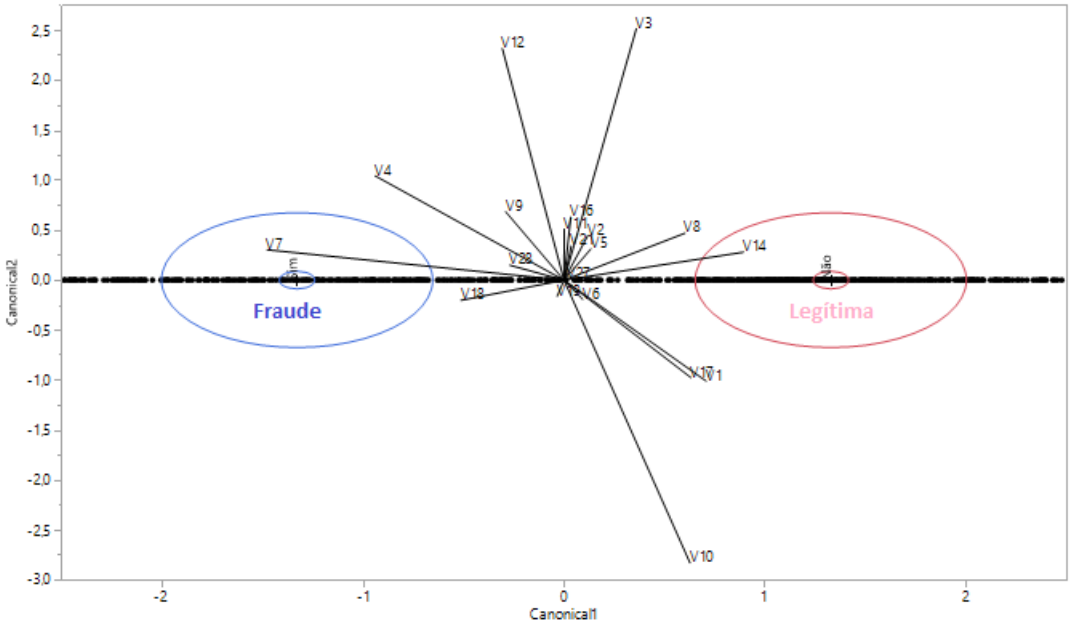


Figura 2.30: Gráfico de Variáveis Canônicas - Cenário IV.

à TA, foram simulados os tamanhos amostrais necessários para se atingir uma TA de 99,9%, com base nos cenários I e IV.

No primeiro cenário, conforme expresso na Figura 2.31, a TA se estabilizou em torno de 70%, em um comportamento relativamente assintótico. Desta forma, ao se aplicar um modelo de regressão logarítmica, obtém-se, com base nos sete conjuntos amostrais iniciais, que seriam necessárias cerca de 2,38 milhões de observações para se atingir uma TA de 99,9%.

Por outro lado, no Cenário IV, observa-se que a inclusão de variáveis significativas reduz a

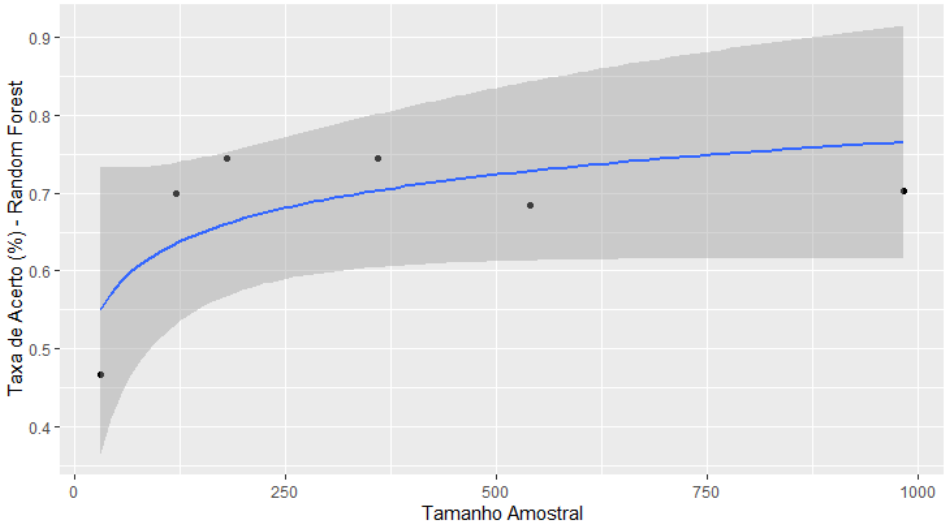


Figura 2.31: Curva de regressão não linear ajustada para as amostras e taxas de acerto - modelo logarítmico (Cenário I).

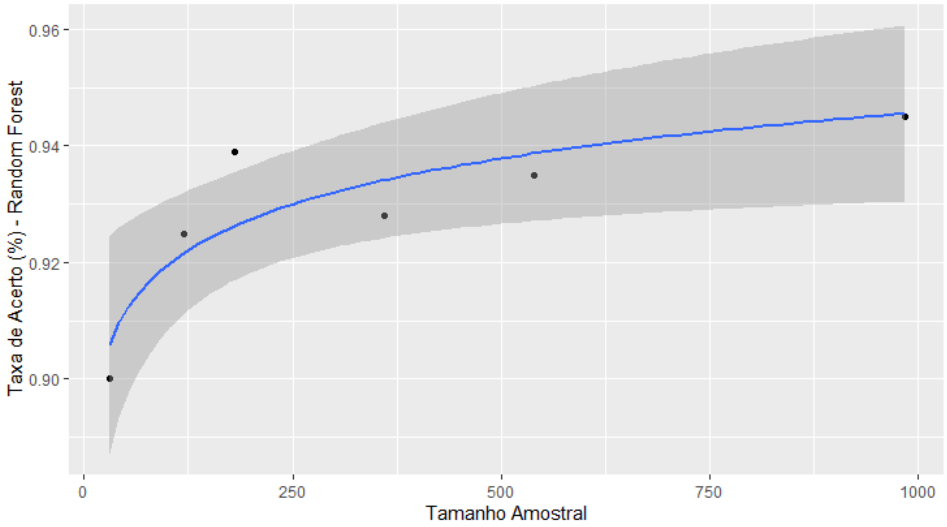


Figura 2.32: Curva de regressão não linear ajustada para as amostras e taxas de acerto - modelo logarítmico (Cenário I).

necessidade de maiores amostras. Conforme a Figura 2.32, a TA se estabilizou em torno de 94% para as sete amostras iniciais. Desta forma, por meio do modelo logarítmico, tem-se que o Random Forest atingiria uma TA de 99,9% em torno de 109 mil observações.

Neste sentido, no Cenário I, é necessário contar com uma grande quantidade de dados, diante da baixa de significância das variáveis preditoras. Já no Cenário IV, o número de observações se reduz para 8,2% do Cenário I por conta da maior significância das variáveis preditoras. Assim, considerando o custo da coleta de informações e a limitação técnica ao se analisar grandes conjuntos de dados, que podem inviabilizar a aplicação da tecnologia no dia a dia, são fatores preponderantes a seleção de variáveis em algoritmos de ML e nos métodos estatísticos convencionais. Neste caso, a utilização da NP-ANOVA

(Teste U de Mann-Whitney, para duas categorias, ou Kruskal-Wallis, para mais categorias) apresentou-se como um método eficaz para a seleção de variáveis significativas, permitindo a redução de custos de coleta de dados e processamento.

Além disso, a Regressão Logarítmica permitiu estimar o tamanho amostral necessário para se trabalhar com implementações práticas, a partir de uma TA maior ou igual a 99,9% ou erro menor que 1/1000. Este método também pode ser utilizado para minimizar falsos positivos e falsos negativos, o que implicaria três recomendações de tamanho amostral.

Aplicou-se a Regressão Logarítmica a subconjuntos aleatórios da amostra considerada para se recomendar o dimensionamento do tamanho amostral. Não foram encontradas referências bibliográficas nesse sentido. A metodologia desenvolvida poderia ser combinada com técnicas de reamostragem *Bootstrapping* ou *Jacknife* para se obterem maiores informações.

Tabela 2.20: Taxa de acerto da classificação dos algoritmos de ML (Cenário IV).

Critério de Teste	**15/15**	**30/30**	**60/60**	**90/90**	**180/180**	**270/270**	**492/492**
Naive Bayes							
60/40	100,00%	83,33%	83,33%	81,94%	90,97%	85,65%	88,32%
10 FOLD	93,33%	91,67%	87,50%	88,33%	88,33%	87,04%	89,94%
4 FOLD	93,33%	86,67%	84,17%	87,22%	87,78%	87,22%	90,24%
Multilayer Perceptron							
60/40	83,33%	91,67%	83,33%	84,72%	87,50%	91,67%	92,89%
10 FOLD	83,33%	90,00%	90,00%	92,78%	91,94%	91,67%	93,39%
4 FOLD	90,00%	90,00%	91,67%	93,33%	92,50%	92,04%	93,19%
Support Vector Machine							
60/40	100,00%	95,83%	89,58%	91,67%	91,67%	91,67%	93,65%
10 FOLD	90,00%	95,00%	91,67%	92,22%	90,56%	92,41%	92,89%
4 FOLD	90,00%	95,00%	91,67%	92,22%	90,28%	91,85%	92,89%
K-NN							
60/40	75,00%	91,67%	89,58%	93,06%	90,28%	90,74%	93,40%
10 FOLD	80,00%	93,33%	90,83%	95,00%	91,39%	90,93%	92,99%
4 FOLD	80,00%	93,33%	90,00%	93,33%	90,83%	90,93%	92,48%
Random Forest							
60/40	100,00%	95,83%	89,58%	87,50%	90,28%	92,59%	95,18%
10 FOLD	90,00%	95,00%	92,50%	93,89%	92,78%	93,52%	94,51%
4 FOLD	93,33%	96,67%	94,17%	93,89%	91,67%	94,26%	93,70%
CART							
60/40	91,67%	95,83%	85,42%	83,33%	88,19%	91,20%	92,89%
10 FOLD	70,00%	93,33%	86,67%	90,00%	90,83%	90,56%	92,28%
4 FOLD	76,67%	95,00%	90,00%	89,44%	90,56%	91,30%	92,17%

2.4 Discussão Final

Ao se analisar os quatro cenários conjuntamente, observou-se que a aplicação da NP-MANOVA para o diagnóstico de viabilidade de utilização de métodos de classificação supervisionada é fundamental, pois permite identificar a significância das informações previamente, em tamanhos amostrais menores. Por esta razão, o Cenário I apresentou menores taxas de acerto em todas as amostras, com um valor máximo de 75,13% (Random Forest). Do Cenário I para o Cenário II, tem-se um efeito positivo na TA em torno de 15%. O melhor classificador (Random Forest) atingiu uma taxa de 89,85%. Do Cenário II para o Cenário III, o ganho marginal se reduziu. Com uma TA de 94,92%, o Random Forest apresentou um incremento de 5% com todos as variáveis incluídas na análise. Por fim, ao se retirar as variáveis não significativas (Cenário IV), o aumento da TA foi de 0,26%. Desta forma, observa-se que a inclusão de variáveis não significativas (Cenário IV para Cenário III) pouco altera a TA, porém inflaciona o custo de armazenagem e processamento, destacando importância de seleção de variáveis para os classificadores.

O algoritmo de ML com melhor desempenho em todos os cenários foi o mesmo: Random Forest. No entanto, o segundo melhor melhor classificador alterou entre os cenários, conforme abaixo:

- Cenário I: Random Forest (75,13%) e CART (71,32%);
- Cenário II: Random Forest (89,85%) e CART (86,04%);
- Cenário III: Random Forest (94,92%) e Support Vector Machine (93,91%);
- Cenário IV: Random Forest (95,18%) e Support Vector Machine (93,65%)

Por outro lado, os piores desempenhos apresentaram a seguinte relação:

- Cenário I: K-NN (52,54%) e Support Vector Machine (54,78%);
- Cenário II: Naive Bayes (72,34%) e K-NN (75,63%);
- Cenário III: Naive Bayes (90,36%) e CART (92,17%);
- Cenário IV: Naive Bayes (90,24%) e CART (92,89%)

Destacam-se, também, dois pontos: o algoritmo CART apresentou, para os cenários I e II, a segunda melhor TA na amostra 492/492. Por outro lado, nos Cenário III e IV, com variáveis mais significativas, o algoritmo apresentou a segunda pior TA. Desta forma, a inclusão de melhores variáveis preditoras, embora tenha afetado positivamente o desempenho do CART, o faz em menor grau do que em outros algoritmos de ML, perdendo desempenho relativo (Cenário III e Cenário IV). De modo contrário, o Support Vector Machine obteve o segundo pior desempenho no Cenário I, mas recuperou-se nos cenários III e IV, em que apresentou a segunda maior TA.

Em relação aos métodos estatísticos, em amostras menores, as análises discriminantes e os Mínimos Quadrados Parciais apresentaram taxas de acerto superiores aos obtidos na Regressão Logística, indicando que, para este conjunto de dados, tratam-se de métodos mais adequados em situações com menores observações. Em amostras maiores, no entanto, os quatro métodos convergiram nos cenários I (TA: $\approx$ 60%), II (TA: $\approx$ 83%) e III (TA: $\approx$ 92,5%). Apenas no Cenário IV (94,2%) foi observado um desempenho consideravelmente melhor da Regressão Logística.

De modo geral, ao se analisar conjuntamente os métodos estatísticos e os algoritmos de ML, observa-se que as análises discriminantes apresentaram melhores desempenhos em amostras menores, em todos os quatro cenários. A Regressão Logística, apesar de ser um método tradicionalmente estatístico, muito embora seja frequentemente confundida com o ML, apresentou comportamento semelhante aos algoritmos de ML, com uma melhora de desempenho em função do tamanho amostral. Desta forma, em conjuntos com maiores observações, os algoritmos de ML obtiveram melhores taxas de acerto. No entanto, os métodos estatísticos foram superiores a pelo menos um algoritmo de ML nas amostras maiores.

No Cenário I, por exemplo, considerando a amostra 492/492, a Análise Discriminante Quadrática obteve uma TA de 62,5%, superior ao desempenho dos algoritmos Multilayer Perceptron (60,7%), Support Vector Machine (54,8%) e K-NN (52,6%). No Cenário II, o melhor método estatístico para a referida amostra foi a Regressão Logística, que, com 83,3% de TA, foi superior ao Naive Bayes (72,3%) e K-NN (75,6%). No Cenário III, obteve-se, para a Análise Discriminante Quadrática, uma taxa de 92,9%, maior do que aquelas observas nos algoritmos Naive Bayes (90,4%), K-NN (92,6%) e CART (92,2%), porém menor que o Random Forest (94,9%). Por fim, no Cenário IV, a Regressão Logística (94,2%) apresentou um dos melhores desempenhos, considerando tanto os métodos estatísticos, quanto o ML, ficando atrás apenas do algoritmo Random Forest, por uma diferença 0,97 pontos percentuais.

Ao se considerar todos os conjuntos, observou-se que, de modo geral, a estatística obteve melhor desempenho em amostras menores. Em amostras superiores, o ML apresentou pequena vantagem por meio do algoritmo Random Forest.

As taxas de Falsos Positivos e Falsos Negativos apresentaram comportamento variado entre os quatro cenários. Falsos Positivos, neste contexto de fraude, traduzem-se pelas situações em que os classificadores atribuem a determinada transação o rótulo de fraude, quando a mesma é legítima, gerando insatisfação no usuário de cartão de crédito (cliente). No Cenário I, a estatística, representada pela Análise Discriminante, apresentou melhor desempenho em todas as amostras, superando o algoritmo Random Forest. Por outro lado, nos Falsos Negativos, em que o classificador atribui a rotulagem de legitimidade a uma transação fraudulenta, situação em que se cria, além da insatisfação do cliente, um prejuízo financeiro para o cliente e/ou companhia de cartão de crédito/seguradora das operações de cartão de crédito, o algoritmo Random Forest apresentou uma menor taxa.

A inserção de variáveis significativas, no entanto, fez com que os comportamentos da estatística e do ML se aproximassem. No Cenário III, por exemplo, as respectivas taxas de Falsos Positivos foram semelhantes, com valores próximos de zero. Já a taxa de Falsos Negativos apresentou um melhor desempenho da estatística em amostras menores, mas se aproximou do ML conforme aumento do tamanho amostral .

Por fim, no Cenário IV, o algoritmo Random Forest apresenta vantagem em relação à taxa de Falsos Positivos, de modo mais evidente nas amostras menores. Nas amostras maiores, a vantagem se manteve, porém em menor escala (1,6% contra 3,5% da Regressão Logística). Por outro lado, na ocorrência de Falsos Negativos, que é o erro mais crítico para este caso de fraude, observou-se, na amostra 492/492 que a estatística, por meio da Regressão Logística (8,1%), apresentou uma taxa de erro menor do que o ML (Random Forest: 9,6%).

Desta forma, não foi possível observar um método unânime, que tenha apresentado desempenho superior em todos os critérios de avaliação, de modo que sua respectiva indicação fica condicionada ao objetivo da análise. Quando se busca, por exemplo, maximizar a TA, Random Forest (95,2%) e a Regressão Logística (94,2%) apresentaram desempenhos superiores e semelhantes. Quando o objetivo é

minimizar a taxa de falsos positivos, o Random Forest foi sensivelmente superior à Regressão Logística. No entanto, quando intuito é reduzir a ocorrência de Falsos Negativos, traduzida por uma situação mais grave no caso de fraude de cartão de crédito, a Regressão Logística apresentou melhor comportamento (8,1%), comparativamente ao Random Forest (9,6%).

Assim, em uma implantação prática (comercial), é possível que seja necessária utilização de Random Forest para a minimização de falsos positivos e de Regressão Logística, para uma menor quantidade de falsos negativos.

2.5 Conclusão

Pelos aspectos apresentados, seria imprudente afirmar, entre Estatística e ML, qual método é o mais efetivo. Tampouco se deve. Tratam-se de duas áreas que oferecem ferramentas com grande potencial sinérgico, de modo que, quando utilizadas conjuntamente, permitem resultados assertivos e mais consistentes. Ademais, não se deve desprezar o fato de que os algoritmos de ML são estruturados sob métodos estatísticos, ou seja, configura-se não só uma relação de sinergia, mas também de indissociabilidade.

Desenvolveu-se um algoritmo de otimização de tamanho amostral que funciona para técnicas de aprendizado de máquina utilizando ML ou métodos estatísticos multivariados. O dimensionamento pode ser efetuado para margem de erro global, margem de erro para falsos positivos ou margem de erro para falsos negativos.

A representação gráfica das funções canônicas com bi-plot, assim como o resultado de NP-ANOVA, permitiram selecionar variáveis preditoras significativas e diminuir a dimensão dos problemas.

Os métodos adotados e desenvolvidos se mostraram adequados para se detectar fraudes de cartão de crédito e podem ser facilmente expandidos para outras áreas do conhecimento, praticamente para qualquer problema de classificação com base numérica. NP-MANOVA, NP-ANOVA e a maior parte dos algoritmos de ML são considerados métodos robustos para a Ciência de Dados, diminuindo a preocupação com distribuições probabilísticas. A NP-MANOVA funcionou corretamente para o diagnóstico de viabilidade da aplicação de técnicas de classificação supervisionadas.

REFERÊNCIAS

Abdi, H. and L. J. Williams, 2010 Principal component analysis. Wiley interdisciplinary reviews: computational statistics **2**: 433–459.

Achia, T. N., A. Wangombe, and N. Khadioli, 2010 A logistic regression model to identify key determinants of poverty using demographic and health survey data. European Journal of Social Sciences **13**: 38–45.

Alpaydin, E., 2009 *Introduction to Machine Learning*, volume 1. The MIT Press, Boston, MA, second edition.

Anderson, M. J., 2001 A new method for non-parametric multivariate analysis of variance. Austral ecology **26**: 32–46.

Bahnsen, A. C., A. Stojanovic, D. Aouada, and B. Ottersten, 2013 Cost sensitive credit card fraud detection using bayes minimum risk. IEEE **1**: 333–338, In 2013 12th international conference on "Machine Learning" and applications.

Bartlett, M. S., G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, 2005 Recognizing facial expression: machine learning and application to spontaneous behavior. In 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) **2**: 568–573.

Bayaga, A., 2010 Multinomial logistic regression: Usage and application in risk analysis. Journal of applied quantitative methods **5**.

Bel, L., D. Allard, J. M. Lauren, R. Cheddadi, and A. Bar-Hen, 2009 Cart algorithm for spatial data: Application to environmental and ecological data. Computational Statistics & Data Analysis **53**: 3082–3093.

Bouckaert, R. R., E. Frank, M. Hall, R. Kirkby, P. Reutemann, A. Seewald, and D. Scuse, 2013 Weka manual for version 3-7-8. The University of Waikato .

Braga, J. P., M. B. de Almeida, A. P. Braga, and J. C. Belchior, 2000 Hopfield neural network model for calculating the potential energy function from second virial data. Chemical Physics **260**: 347–352.

Bramer, M., 2007 *Principles of data mining*, volume 180. Springer, London, 8th edition.

Breiman, L., 2001 Random forests. Machine Learning **45**: 5–32.

Bzdok, D., N. Altman, and M. Krzywinski, 2017 Computer vision for autonomous vehicles: Problems, datasets and state of the art. Arxiv .

Bzdok, D., N. Altman, and M. Krzywinski, 2018 Points of significance: statistics versus machine learning. Nature **333**.

Cabete, N. P. and M. G. M. S. Cardoso, 2006 Algoritmo cart: previsão do desempenho na matemática do secundário. Revista de Ciências da Computação **1**: 11–27.

Cambridge Dictionary, 2020 Fraud. Technical report, Disponível em: `https://dictionary.cambridge.org/pt/dicionario/ingles/fraud`.

Choi, D. and K. Lee, 2018 An artificial intelligence approach to financial fraud detection under iot environment: A survey and implementation. Security and Communication Networks **1**.

Columbus, L., 2020 Roundup of machine learning forecasts and market estimate. Forbes .

Conforto, E. C., D. C. Amaral, and S. D. Silva, 2011 Roteiro para revisão bibliográfica sistemática: aplicação no desenvolvimento de produtos e gerenciamento de projetos. 8° Congresso Brasileiro de Gestão de Desenvolvimento de Produto Trabalho apresentado (8).

Díaz, A. E., 2004 Muestreo progresivo. NSC .

de Medeiros, I. L., A. Vieira, G. Braviano, and B. S. Gonçalves, 2015 Revisão sistemática e bibliometria facilitadas por um canvas para visualização de informação. InfoDesign-Revista Brasileira de Design da Informação **12(1)**: 93–110.

De Moraes, D., 2008 Modelagem de fraude em cartão de crédito. Repositório Institucional UFSCAR **1**.

Domingos, P. M., 2012 A few useful things to know about machine learning. Commun. acm **55(10)**: 78–87.

Fernandes, E. A. D. N., G. A. Sarriés, M. A. Bachhi, Y. T. Mazola, C. Gonzaga, and S. R. Sarriés, 2020 Trace elements and machine learning for brazilian beef traceability. Food Chemistry **333**.

Figueira, C. V., 2006 Modelos de regressão logística. LUME: Repositório Digital UFRGS .

Fundo Monetário Internacional, 2019 Gdp, current prices (2015). Technical report, Fundo Monetário Internacional, Disponível em: `https://www.imf.org/external/datamapper/NGDPD@WEO/OEMDC/ADVEC/WEOWORLD`.

Gadi, M. F. A., 2008 Uma comparação de métodos de classificação aplicados à detecção de fraude em cartões de crédito. Universidade de São Paulo **1**, Tese de Doutorado.

Geladi, P. and B. R. Kowalski, 1986 Partial least-squares regression: a tutorial. Analytica chimica acta pp. 1–17.

Gislason, P. O., J. A. Benediktsson, and J. R. Sveinsson, 2006 Random forests for land cover classification. Pattern Recognition Letters **27**: 294–300.

Hongyu, K., V. L. M. Sandanielo, and G. J. De Oliveira Junior, 2016 Análise de componentes principais: resumo teórico, aplicação e interpretação. E&S Engineering and Science **5**: 83–90.

Johnson, R. A. and D. W. Wichern, 2002 *Applied multivariate statistical analysis*, volume 5. Prentice hall, Upper Saddle River, NJ, 8th edition.

Kelly, B. J., R. Gross, K. Bitinger, S. Sherrill-Mix, J. D. Collman, R. Collman, and H. Li, 2015 Power and sample-size estimation for microbiome studies using pairwise distances and permanova. Bioinformatics **31**: 2461–2468.

Kourou, K., T. P. Exarchos, M. V. Karamouzis, and D. I. Fotiadis, 2015 Machine learning applications in cancer prognosis and prediction. Computational and structural biotechnology journal pp. 8–17.

Kumar, M. and M. Thenmozhi, 2006 Forecasting stock index movement: A comparison of support vector machines and random forest. In: Indian institute of capital markets 9th capital markets conference paper .

Langer, D. L., T. H. Van der Kwast, A. J. Evans, J. Trachtenberg, B. C. Wilson, and M. A. Haider, 2009 Prostate cancer detection with multi-parametric mri: Logistic regression analysis of quantitative t2, diffusion-weighted imaging, and dynamic contrast-enhanced mri. Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine **30**: 327–334.

Lorena, A. C. and A. C. De Carvalho, 2007 Uma introdução às support vector machines. Revista de Informática Teórica e Aplicada **14**: 43–67.

Mazola, Y. T., E. A. D. N. Fernandes, G. A. Sarriés, M. A. Bacchi, and C. L. Gonzaga, 2019 Neutron activation analysis and data mining techniques to discriminate between beef cattle diets. Journal of Radioanalytical and Nuclear Chemistry pp. 1571–1578.

Nachar, N., 2008 The mann-whitney u: A test for assessing whether two independent samples come from the same distribution. Tutorials in quantitative Methods for Psychology **4**: 13–20.

Ng, K. S., 2013 A simple explanation of partial least squares. The Australian National University .

Nguyen, C., Y. Wang, and N. H. N, 2013 Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic. Journal of Biomedical Science and Engineering **6**: 10.

Nilson Report, 2016 The nilson report. Technical report, Disponível em: `https://www.nilsonreport.com/upload/content_promo/The_Nilson_Report_10-17-2016.pdf`.

Phua, C., E. Y. Cheu, Y. G. E, S. K, and M. N. Nguyen, 2012 Feature engineering for click fraud detection. In ACML Workshop on Fraud Detection in Mobile Advertising .

Pirouz, D. M., 2006 An overview of partial least squares. Analytica chimica acta pp. 1–17, Available at SSRN 1631359.

Poole, D., 2014 *Linear algebra: A modern introduction*, volume 1. Cengage Learning, Stamford, CT, fourth edition.

Provost, F., D. Jensen, and T. Oates, 1999 Efficient progressive sampling. In: Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining pp. 23–32.

Randhawa, K., C. K. Loo, M. Seera, C. P. Lim, and A. K. Nandi, 2018 Credit card fraud detection using adaboost and majority voting. IEEE access **6**: 14277–14284.

Rutkowski, L., M. Jaworski, L. Pietruczuk, and P. Duda, 2014 The cart decision tree for mining data streams. Information Sciences pp. 1–15.

Serasa, 2020 O que é fraude. Technical report, Disponível em: `https://www.serasa.com.br/ensina/seu-cpf-protegido/o-que-e-fraude/`.

Statnikov, A., L. Wang, and C. F. Aliferis, 2008 A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. BMC bioinformatics **9**: 319.

Trivellato, G. M. L., G. A. Sarriés, and G. N. Furlan, 2020 Machine learning para análise do banco de dados do sistema de avaliação ponderada da multifuncionalidade da agricultura (mfa). In: Workshop on Probabilistic and Statistical Methods **8**: 33–34, São Carlos: ICMC/USP - DEs/UFSCar.

Web of Science, 2019 Web of knowledge. Technical report, Web of Science, Disponível em: `http://apps.webofknowledge.com`.

Zanin, M., M. Romance, S. Moral, and R. Criado, 2018 Credit card fraud detection through parenclitic network analysis. Complexity .

Zareapoor, M. and P. Shamsolmoali, 2015 Application of credit card fraud detection: Based on bagging ensemble classifier. Procedia computer science **48**: 679–685.

Zhang, Z., 2016 Introduction to machine learning: k-nearest neighbors. Annals of translational medicine **4**.

Apêndice

Capítulo 1

Tabela A.22: Descrição estrutural do artigo 1. Bahnsen *et al.* (2013).

Critérios	Descrição
Título; Autores; Ano de Publicação	Cost Sensitive Credit Card Fraud Detection using Bayes Minimum Risk; Alejandro Bahnsen, Aleksander Stojanovic, Djamila Aouada & Bjorn Ottersten; 2013
Meio de Publicação	12th International Conference on Machine Learning and Applications (anais)
Objetivos	Avaliar os custos monetários de má classificação de um caso não fraudulento como fraudulento, além de apresentar um classificador bayesiano de risco mínimo, de modo a se obter um sistema que seja sensível ao custo das detecções de fraude de cartão de crédito.
Banco de Dados Utilizados	Dados provenientes de uma companhia de cartão de crédito europeia. O banco possui um conjunto de 80 milhões de registros, divididos entre as modalidades crédito e débito, com 27 variáveis disponíveis. Para o estudo, foram selecionadas apenas as transações de cartão de crédito. Deste total, utilizaram-se 750.000 registros como amostragem, dos quais 3.500 são fraudulentos.
Variáveis Disponíveis	Do total de 27 variáveis, foram selecionadas 14 para comporem a análise. São elas: Data da transação; Identificação do número da conta do titular; Identificação do número do cartão; Tipo de transação (internet, cartão presente, etc); Volume da transação, em Euros; Identificação do vendedor; Código do grupo do vendedor; País de origem da transação; País de origem do titular do cartão; Bandeira do cartão; Sexo do titular do cartão; Idade do titular do cartão; Banco emissor do cartão; Variável indicativa de fraude ou não.
Metodologia Proposta	Os autores dividiram os dados em grupos de treino e de teste, com, respectivamente, 625.000 e 125.00 observações. Em seguida, selecionaram cinco subamostras, todas com 2.900 observações fraudulentas, mas diferentes volumes totais de transações. Desta forma, obtiveram conjuntos em que a representatividade da fraude totalizou 1%, 5%, 10%, 20% e 50%. Além disso, aplicaram uma abordagem de precificação dos erros de classificação, buscando quantificar quanto se perde com os falsos positivos (operações classificadas como fraude, porém normais) e falsos negativos (operações classificadas como normais, porém fraudulentas). Para avaliar a detecção de fraude, utilizaram três algoritmos: Random Forest, C4.5 e Logistic Regression. Além disso, avaliaram os resultados utilizando três outras abordagens: a Thresholding Optimization, o Bayes Minimum Risk Classifier e Bayes Minimum Risk Classifier com probabilidade ajustada. A forma de avaliação dos resultados foi obtida através de valores monetários, em Euros.
Assertividade dos Algoritmos	Os resultados foram apresentados graficamente pelos autores, portanto não são trazidas suas tabulações. Na aplicação individual dos algoritmos, os autores revelam um fato curioso: ao considerar a curva de custo, o melhor resultado está quando se utiliza uma amostra com 50% de dados fraudulentos, para os três algoritmos testados. No entanto, ao levar-se em consideração apenas o desempenho dos algoritmos, o melhor resultado foi encontrado para uma amostra com 5% de dados fraudulentos. Após a aplicação da Thresold Optimization, houve melhora de desempenho apenas para o algoritmo Random Forest. Com a utilização do Bayes Minimum Risk Classifier, não houve qualquer melhoria de desempenho dos algoritmos. Pelo contrário, grande parte apresentou perda de desempenho, tanto em valores monetários, quanto em capacidade de detecção. No entanto, após aplicação do Bayes Minimum Risk Classifier com probabilidade ajustada, os resultados com o Random Forest voltaram a se destacar. De modo geral, o melhor resultado obtido pelos autores foi com a utilização do Random Forest na abordagem Bayes Minimum Risk Classifier com probabilidade ajustada, tanto no aspecto monetário, quanto no desempenho.

Tabela A.23: Descrição estrutural do artigo 2. Zareapoor and Shamsolmoali (2015).

Critérios	Descrição
Título; Autores; Ano de Publicação	Application of Credit Card Fraud Detection: Based on Bagging Ensemble Classifier; Masoumeh Zareapoor & Pourya Shamsolmoali; 2015
Meio de Publicação	Procedia Computer Science (journal)
Objetivos	Elaborar uma revisão de literatura e discutir os algoritmos mais inovadores na detecção de fraude de cartão de crédito.
Banco de Dados Utilizados	Dados obtidos de uma competição da Universidade da Califórnia em São Diego (UCSD), com 100.000 registros de transações, dos quais 2.293 referem-se a situações de fraude. O conjunto é relativo a um período de 98 dias, porém a data não é especificada.
Variáveis Disponíveis	O banco é composto por dezessete variáveis, cujas descrições não estão disponíveis.
Metodologia Proposta	Os autores dividiram o conjunto de dados em quatro grupos, com base na taxa de registros fraudulentos. Desta forma, obtiveram amostras com 20%, 15%, 10% e 3% de transações com fraude. A partir da amostragem, aplicaram quatro algoritmos: Bagging, Naive Bayes, K-Nearest Neighbors.
Assertividade dos Algoritmos	Os valores de acurácia são apresentados apenas graficamente, o que inviabiliza sua tabulação. No entanto, segundo os autores, o classificador Bagging apresentou a melhor performance na detecção de registros fraudulentos.

Tabela A.24: Descrição estrutural do artigo 3. Choi and Lee (2018).

Critérios	Descrição
Título; Autores; Ano de Publicação	An Artificial Intelligence Approach to Financial Fraud Detection under IoT Environment: A Survey and Implementation; Dahee Choi & Kyungho Lee; 2018.
Meio de Publicação	Security And Communication Networks (Journal)
Objetivos	(1) Revisar e Avaliar as técnicas de detecção de fraudes utilizadas em trabalhos de 2016 a 2018; (2) Analisar as vantagens e as limitações de tais técnicas; (3) Construir um modelo baseado na revisão de literatura e no banco de dados disponível; (4) Comparar os resultados do Machine Learning tradicional com o deep learning.
Banco de Dados Utilizados	Dados de Pagamento no ambiente da Internet of Things, no mercado da Coreia do Sul (2016). Os dados não se encontram disponíveis online.
Variáveis Disponíveis	O banco é composto por 21 variáveis. No entanto, apenas 14 são apresentadas pelos autores: Número de Série da Transação; Tipo da Transação; Data da Transação; Tempo de Autenticação; Status da Transação; Companhia de Telecomunicação; Número de Telefone; Montante da Transação; Identificação da Bandeira; Identificação do Vendedor; Identificação do Serviço; Hash de E-mail; Informação do IP; Código de Autenticação do Cliente
Metodologia Proposta	Inicialmente, foi realizada uma revisão de Literatura de trabalhos anteriores que contemplaram a detecção de fraudes financeiras em geral e o Machine Learning, entre 2016 e 2017. Em seguida, os autores utilizaram aprendizado não supervisionado para captar as ameaças aos pagamentos e aprendizado supervisionado para avaliar a classificação de cenários fraudulentos. No ensino não supervisionado, para fins de clusterização, foram utilizados os algoritmos EM e DensityBased. No ensino supervisionado, para fins de classificação, os autores utilizaram Random Forest, Regressão Logística e C4.5. Além disso, também foi utilizada uma Rede Neural Artificial, enquadrada como deep learning.
Assertividade dos Algoritmos	EM: 0,99862; DensityBased: 0,98788; Regressão logística: 0,9997; Random Forest: 0,99969; C4.5 0,99943; Rede Neural Artificial 0,914

Tabela A.25: Descrição estrutural do artigo 4. Zanin *et al.* (2018).

Critérios	Descrição
Título; Autores; Ano de Publicação	Credit Card Fraud Detection through Parenclitic Network Analysis; Massimiliano Zanin, Miguel Romance, Santiago Moral & Regino Criado; 2018.
Meio de Publicação	Complexity (Journal)
Objetivos	Avaliar a utilização de redes complexas para o aumento de desempenho da detecção de fraude de cartão de crédito.
Banco de Dados Utilizados	Transações de cartão de debito e crédito dos clientes do banco espanhol BBVA, de janeiro de 2011 a dezembro de 2012.
Variáveis Disponíveis	- Volume da transação, em Euro; - Tempo desde a última transação; - Volume da última transação, em Euro; - Volume médio das últimas transações, em Euro; - Tempo médio entre as transações; - Variável indicativa de compra idêntica, em valores booleanos; - Hora do dia; - Taxa de fraude das últimas 50.000 transações de cartão (geral); - Suspeita de fraude (probabilidade entre 0 e 1); - Presença de fraude (variável binária, 0 ou 1)
Metodologia Proposta	Os autores utilizaram a Parenclitic Networks, uma rede neural de maior complexidade, no intuito de avaliar se sua aplicação implicaria maior assertividade na detecção de fraudes de cartões. Para tanto, compararam os valores de acurácia com os algoritmos Rede Neural Artificial e Multi-Layer Perceptron [MLP]. No entanto, para a disposição dos resultados, foram apresentadas três formas de análise: análise combinada, com a presença de data mining e a rede neural complexa; apenas Parenclitic Network; e Multi-Layer Perceptron.
Assertividade dos Algoritmos	- Análise combinada: 0,8365 - Parenclitic Network: 0,8392 - Multi-Layer Perceptron: 0,8388

Tabela A.26: Descrição estrutural do artigo 5. RANDHAWA *ET AL.* (2018).

Critérios	Descrição
Título; Autores; Ano de Publicação	Credit Card Fraud Detection Using AdaBoost and Majority Voting; Kuldeep Randhawa, Chu Kiong Loo, Manjeevan Seera, Chee Peng Lim & Asoke Nandi; 2018
Meio de Publicação	IEEE Access (journal)
Objetivos	Avaliar a assertividade de diferentes algoritmos na detecção de fraude de cartão de crédito, a partir de um conjunto de dados real.
Banco de Dados Utilizados	Dados de transações de cartão de crédito de uma instituição financeira da Malásia, de fevereiro de 2017 a abril de 2017, com 287.224 linhas de registro, das quais 102 são classificadas como fraudulentas.
Variáveis Disponíveis	Número do cartão de crédito; Volume da transação; Fatura do titular do cartão; Número sistêmico de auditoria; Horário da transação; Data da transação; Tipo de Mercadoria; Modo de operação da transação; Identificação do responsável; Código da moeda da transação; Código da moeda da fatura do titular.
Metodologia Proposta	Os autores avaliaram o desempenho de doze algoritmos, a partir de três perspectivas distintas: a primeira, por meio dos algoritmos individuais; a segunda, através da majority voting, método que trabalha com pelo menos dois algoritmos classificadores simultaneamente, selecionando, os melhores, de acordo com a classe das variáveis; e a terceira, por meio do Adaboost, ferramenta que permite a reclassificação de um algoritmo até que as más classificações sejam minimizadas. Desta forma, os doze algoritmos selecionados foram: Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), Gradient Boost Tree (GBT), Decision Stump (DS), Random Tree (RT), Neural Network (NN), Deep Learning (DL), Multi-Layer Perceptron (MLP), Linear Regression (LIR), Logistic Regression (LOR) e Support Vector Machine (SVM)
Assertividade dos Algoritmos	**Algoritmos individuais**: NB: 0,99999; DT: 0,99999; RF: 0,99999; GBT: 0,99999; DS: 0,99999; RT: 0,99992; DL: 0,99985; NN: 0,99997; MLP: 0,99997; LIR: 0,99965; LOR: 0,99999; SVM: 0,95564. **Algoritmos com AdaBoost**: NB: 1,00000; DT: 0,99999; RF: 1,00000; GBT: 0,99999; DS: 0,99999; RT: 1,00000; DL: 0,99994; NN: 0,99998; MLP: 0,99996; LIR: 0,99992; LOR: 0,99999; SVM: 0,99959. **Algoritmos com Majority Voting**: DS+GBT: 1,00000; DT+DS: 1,00000; DT+GBT: 1,00000; DT+NB: 0,99999; NB+GBT: 0,99999; NN+NB: 0,99998; RF+GBT: 0,99999

Capítulo 2

Exemplo de execução de Random Forest no software Weka

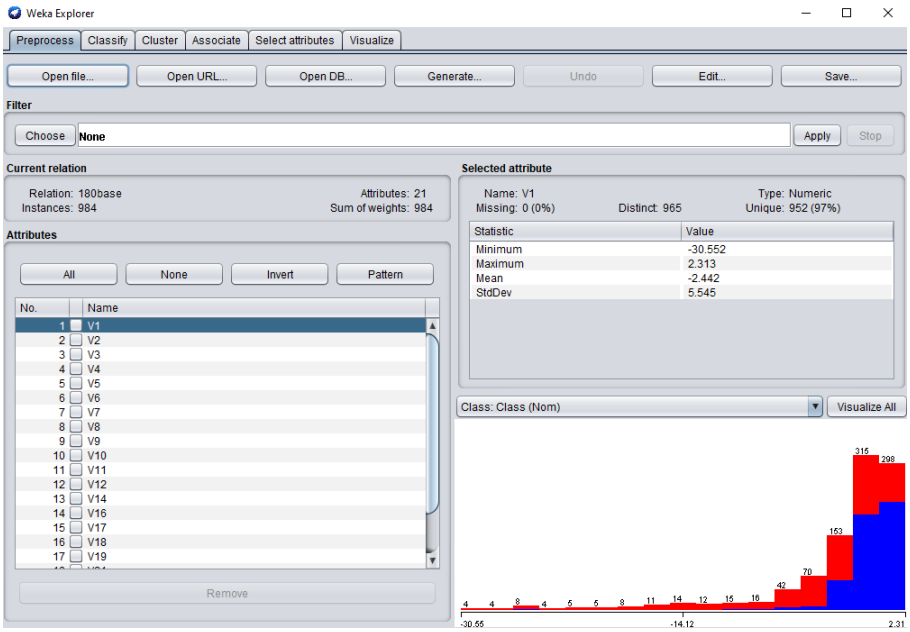


Figura A.33: Passo 1 para a execução do algoritmo Random Forest no Weka

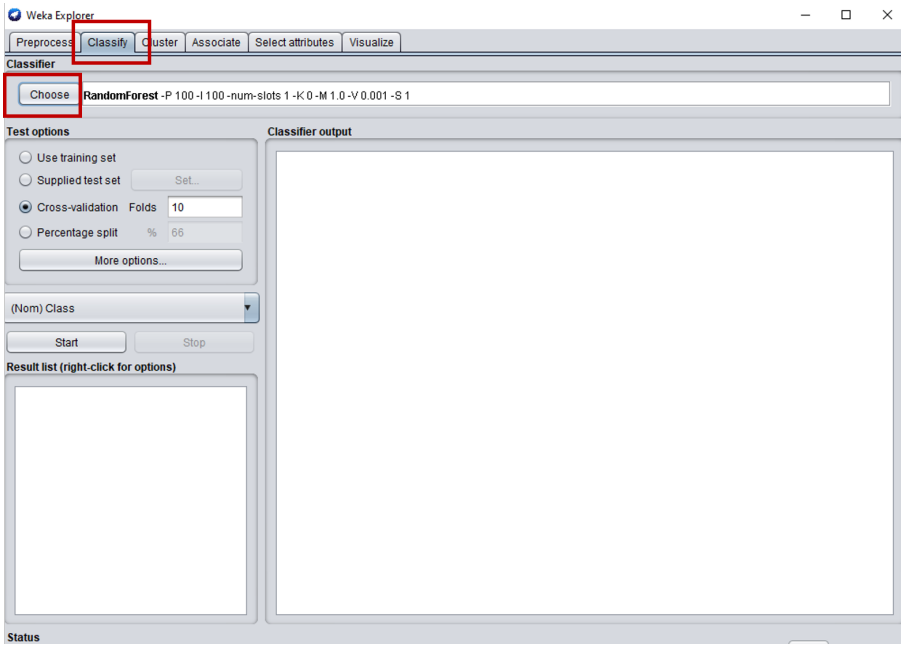


Figura A.34: Passo 2 para a execução do algoritmo Random Forest no Weka

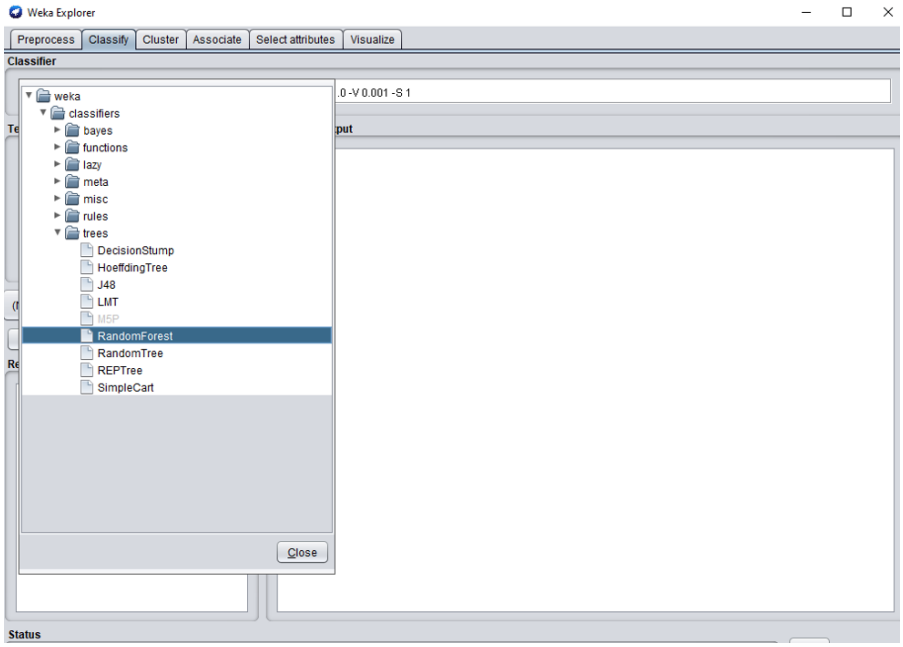


Figura A.35: Passo 3 para a execução do algoritmo Random Forest no Weka

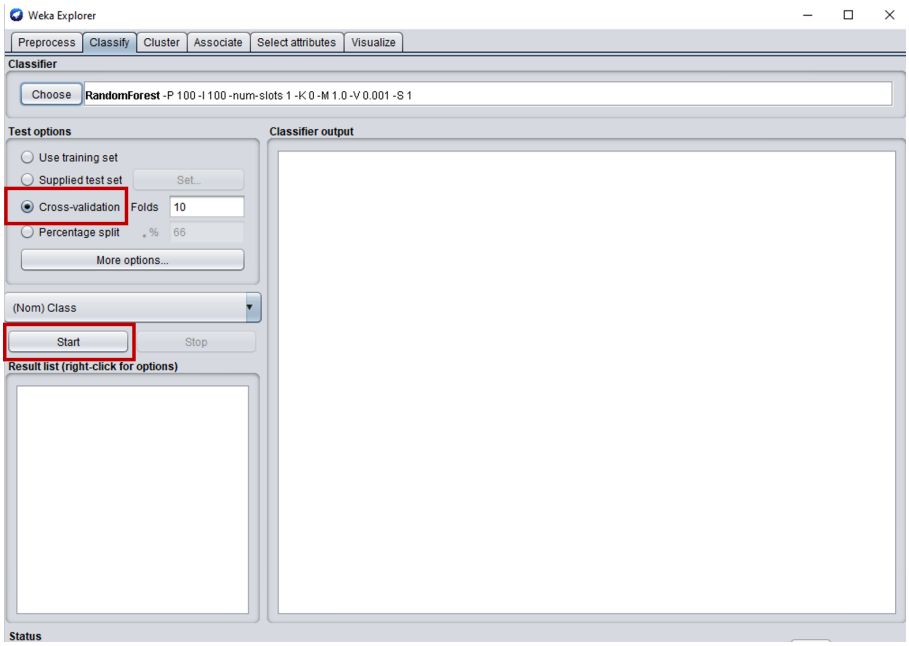


Figura A.36: Passo 4 para a execução do algoritmo Random Forest no Weka

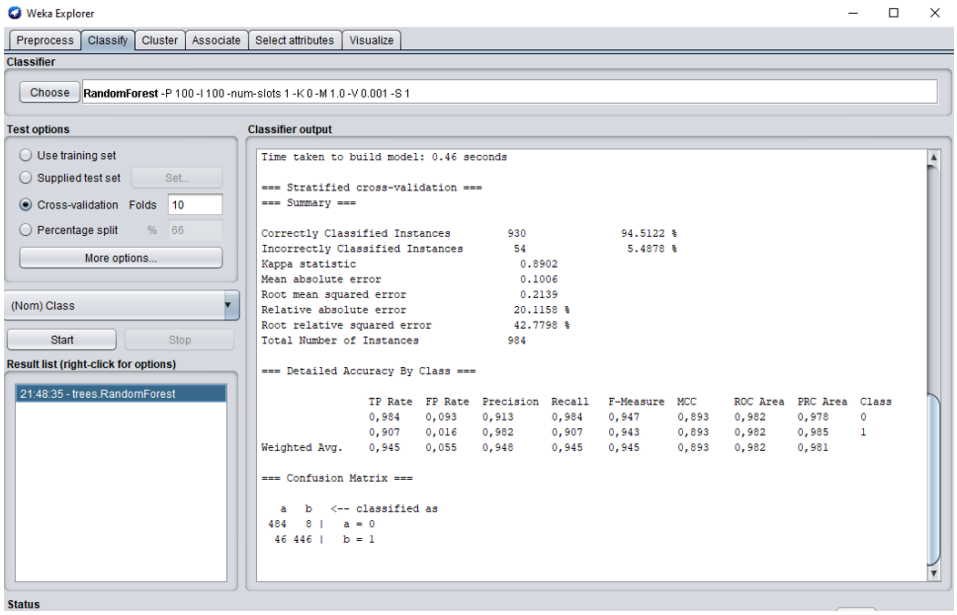


Figura A.37: Passo 5 para a execução do algoritmo Random Forest no Weka

Script em R utilizado para amostragem

```
library(dplyr)

completo = read.csv(''creditcard.csv", h=T)

str(completo)
fraude_01 = filter(completo, completo$Class == 1)
fraude_00 = filter(completo, completo$Class == 0)

#30
base30_00 = sample_n(fraude_00,15,F)
base30_01 = sample_n(fraude_01,15,F)
base_30 = rbind(base30_00,base30_01)
write.csv(base_30,file=''base_30.csv")

#60
base60_00 = sample_n(fraude_00,30,F)
base60_01 = sample_n(fraude_01,30,F)
base_60 = rbind(base60_00,base60_01)
write.csv(base_60,file=''base_60.csv")

#120
base120_00 = sample_n(fraude_00,60,F)
base120_01 = sample_n(fraude_01,60,F)
base_120 = rbind(base120_00,base120_01)
write.csv(base_120,file=''base_120.csv")

#180
base180_00 = sample_n(fraude_00,90,F)
base180_01 = sample_n(fraude_01,90,F)
base_180 = rbind(base180_00,base180_01)
write.csv(base_180,file=''base_180.csv")

#360
base360_00 = sample_n(fraude_00,180,F)
base360_01 = sample_n(fraude_01,180,F)
base_360 = rbind(base360_00,base360_01)
write.csv(base_360,file=''base_360.csv")

#540
base540_00 = sample_n(fraude_00,270,F)
base540_01 = sample_n(fraude_01,270,F)
```

```
base_540 = rbind(base540_00,base540_01)
write.csv(base_540,file=``base_540.csv")

#984
base984_00 = sample_n(fraude_00,492,F)
base984_01 = sample_n(fraude_01,492,F)
base_984=rbind(base984_00,base984_01)
write.csv(base_984,file=``base_984.csv")
```