UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Modelo de Propensão: Como Identificar os Clientes com Maior Chance de Compra?

Leonardo Caresia Pires

Trabalho de Conclusão de Curso
MBA em Inteligência Artificial e Big Data

ICMC USP SÃO CARLOS

UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Modelo de Propensão: Como Identificar os Clientes com Maior Chance de Compra?

Leonardo Caresia Pires

USP - São Carlos

2023

Leonardo Caresia Pires

Modelo de Propensão: Como Identificar os Clientes com Maior Chance de Compra?

Trabalho de conclusão de curso apresentado ao Departamento de Ciências de Computação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo - ICMC/USP, como parte dos requisitos para obtenção do título de Especialista em Inteligência Artificial e Big Data.

Área de concentração: Inteligência Artificial

Orientador: Prof. Dr. Ricardo Rodrigues Ciferri

USP - São Carlos

2023

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados inseridos pelo(a) autor(a)

P667m

Pires, Leonardo
Modelo de Propensão: Como Identificar os Clientes com Maior Chance de Compra? / Leonardo Pires; orientador Ricardo Ciferri. -- São Carlos, 2023.
75 p.

Trabalho de conclusão de curso (MBA em Inteligência Artificial e Big Data) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2023.

1. Modelo de Propensão. 2. Classificação. 3. Aprendizado de Màquina. 4. Inteligência Artificial . 5. E-commerce. I. Ciferri, Ricardo , orient. II. Título.

Bibliotecários responsáveis pela estrutura de catalogação da publicação de acordo com a AACR2:
Gláucia Maria Saia Cristianini - CRB - 8/4938
Juliana de Souza Moraes - CRB - 8/6176

DEDICATÓRIA

Aos meus pais, meu primo e à minha namorada pela compreensão, carinho e apoio incansável.

AGRADECIMENTOS

Ao Prof. Dr. Ricardo Rodrigues Ciferri, pelo incentivo e dedicação em ajudar na construção dessa pesquisa. Sua participação foi de suma importância, sempre direcionando o andar dessa pesquisa da melhor maneira possível e sempre se apresentando disponível para o esclarecimento de dúvidas.

A Prof. Dra. Solange Oliveira Rezende, pela compreensão e orientação durante todo o curso de MBA em Big Data e Inteligência Artificial da Universidade de São Paulo. Sua ajuda foi fundamental para o direcionamento e esclarecimento de dúvidas durante as matérias de Metodologia e Projeto (MET).

Por fim, um agradecimento especial a todos os professores e tutores do MBA em Inteligência Artificial e Big Data pelo excelente material das aulas, pela mentoria e tutoria durante todo esse período de curso.

.

EPÍGRAFE

"A arte desafia a tecnologia, e a tecnologia inspira a arte."

John Lasseter (s.d) [1]

RESUMO

Pires, L. C. **Modelo de Propensão: Como Identificar os Clientes com Maior Chance de Compra?** 2023. 75f. Trabalho de conclusão de curso (MBA em Inteligência Artificial e Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

Conhecer o cliente alvo é fundamental para que uma campanha de marketing de certo. Agora, saber exatamente quais clientes tem maior chance de comprar um determinado produto, com uma probabilidade alta de assertividade, é um diferencial competitivo. Para chegar nesse resultado de alta assertividade, foi implementado nesse estudo um modelo de propensão, onde o resultado é uma lista de clientes mais propensos a compra. O processo para chegar nessa lista passou pela extração das bases de treinamento e teste (em arquivos csv), provenientes do site Kaggle. A base de teste foi utilizada apenas para a predição final. Já a base de treinamento foi subdividida em treinamento e validação, passando por 4 experimentos: Experimento 1 – divisão da base de dados (*test split)* de 80% treinamento e 20% validação com todos os atributos da base. Experimento 2 – *test split* de 80% treinamento e 20% validação com apenas os 10 atributos mais relevantes, de acordo com o modelo KNN. Experimento 3 - *test split* de 75% treinamento e 25% validação, com todos os atributos. Experimento 4 - *test split* de 75% treinamento e 25% validação, com apenas os 10 melhores atributos. Cada experimento utilizou 3 modelos de classificação, cada um representando uma determinada categoria. Representando a categoria dos ensembles, foi utilizado o modelo de *Random Forest*. Já para os algoritmos explicativos, a Regressão Logística foi utilizada. Por último, para representar os modelos de probabilidade, o Naïve Bayes foi selecionado. Esses 3 modelos foram utilizados em cada um dos 4 experimentos mencionados. Com o intuito de medir a eficácia de predição de cada modelo, foram utilizados indicadores como F-Score, acurácia, AUC Score e matriz de confusão. Esses indicadores ajudaram a definir qual modelo e experimento obteve melhor performance de predição. Com as melhores notas entre os indicadores, principalmente o F-Score, o experimento 2 com *test split* 80% treinamento e 20% validação e com seleção de variáveis, foi escolhido como modelo ideal para essa base de dados. Além disso, o algoritmo de Regressão Logística comprovou ser o melhor entre os 3 algoritmos analisados, com maiores notas de acurácia, AUC Score. Após a seleção do melhor algoritmo e experimento, a base de teste, que até então estava intacta, foi utilizada para a inserção das predições realizadas através da Regressão Logística. Em suma, o resultado foi uma lista de usuários, cada um com uma predição variando de 0 a 100%, sendo 100% a maior probabilidade de compra. Por ser um classificador binário, as predições realizadas pela Regressão Logística ficaram próximas dos extremos, entre 0-10% e 70-100%, com esse segundo grupo apresentando um total de 1.182 usuários com alta chance de compra.

Palavras-chave: Modelo de Propensão, Classificação, Aprendizado de Máquina, Inteligência Artificial, E-commerce, Compras Online.

ABSTRACT

Pires, L. C. **Propensity Model: How to Identify Customers with a Higher Likelihood of Purchase?** 2023. 75 p. Term Paper (MBA in Artificial Intelligence and Big Data) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2023.

Knowing the target customer is paramount for a marketing campaign to succeed. Now, knowing exactly which customers are most likely to buy a specific product with a high probability of accuracy is a competitive advantage. To achieve this high level of accuracy, a propensity model was implemented in this research, where the result is a list of customers more likely to make a purchase. The process to arrive at this list involved extracting training and test datasets (in csv files) from the Kaggle website. The test dataset was used only for the final prediction, while the training dataset was subdivided in training and validation going through four experiments: Experiment 1 - 80% training and 20% validation data split with all attributes from the dataset. Experiment 2 - 80% training and 20% validation data split with only the top 10 most relevant attributes according to the KNN model. Experiment 3 - 75% training and 25% validation data split with all attributes. Experiment 4 - 75% training and 25% validation data split with only the top 10 best attributes. Each experiment employed three classification models, each representing a specific category. The Random Forest model was used to represent the ensemble category. Logistic regression was used for explanatory algorithms, and Naïve Bayes was selected to represent the probability models. These three models were used in each of the four mentioned experiments. To measure the predictive effectiveness of each model, indicators such as F-Score, accuracy, AUC Score, and confusion matrix were used. These indicators helped determine which model and experiment achieved the best prediction performance. With the highest Scores among the indicators, especially F-Score, Experiment 2 with an 80% training and 20% validation data split and variable selection, was chosen as the ideal model for this dataset. Furthermore, the Logistic Regression algorithm proved to be the best among the three algorithms analyzed, with higher accuracy, AUC Score and F-Score. After selecting the best algorithm and experiment, the previously untouched test dataset was used to insert predictions made through logistic regression. In summary, the result was a list of users, each with a prediction ranging from 0 to 100%, where 100% represented the highest probability of purchase. Since it is a binary classifier, the predictions made by logistic regression were close to the extremes, ranging from 0-10% to 70-100%. The latter group included a total of 1,182 users with a high likelihood of purchase.

Keywords: Propensity Model, Classification, Machine Learning, Artificial Intelligence, E-commerce, Online Shopping.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Um dos maiores desafios do mundo corporativo nos dias de hoje é a falta de conhecimento que as empresas têm sobre o perfil de seus clientes. Geralmente, ações de marketing são definidas exclusivamente pela capacidade orçamentaria, sem levar em conta quem são as pessoas que serão afetadas por essas campanhas. Portanto, um grande diferencial competitivo é o conhecimento profundo sobre quem são os clientes desejados ou clientes alvo.

Nesse cenário, a utilização do aprendizado de máquina para a classificação e predição desses clientes alvo pode ajudar as empresas a conhecer melhor seus clientes e reduzir custos, uma vez que apenas os clientes mais propensos a compra deveriam receber a comunicação de marketing dessas empresas. Uma dessas técnicas de aprendizado de máquina, que envolve tanto classificação como predição, é conhecida como modelo de propensão e será analisada em mais detalhes durante essa pesquisa.

1.1 Contextualização

Um modelo de propensão é uma ferramenta de aprendizado de máquina que mede a probabilidade de um cliente realizar uma determinada ação futura, como comprar um determinado produto por exemplo, com base em acontecimentos passados. Na era atual do Big Data, onde grandes volumes de dados estão disponíveis a todo momento para serem capturados e estudados por empresas de diversos setores, se torna necessária uma filtragem e um agrupamento desse volume maçante de informações para que sejam excluídos os excessos, fazendo com que apenas os dados mais relevantes sejam utilizados [2]. Todas as etapas da jornada do cliente em um ambiente Online, como por exemplo compras em um site de comércio eletrônico (e-commerce), são armazenadas nos bancos de dados das empresas donas desses sites. O comércio eletrônico, do inglês e-commerce, pode ser definido como transações comerciais no ambiente online, que são realizadas através da ampla utilização da tecnologia e internet. Isso ocorre de maneira livre, com baixo custo e que pode ser realizada tanto entre empresas quanto entre empresas e pessoas [3]. Informações como por onde o cliente navegou enquanto esteve no site, onde clicou, quais itens selecionou, se conferiu o tempo de entrega do produto, se realizou ou não uma compra, são informações de suma importância para serem usadas em futuras interações com o usuário. Mesmo que o usuário não tenha comprado nada

em sua interação com o site, existe uma grande possibilidade de ele ou ela vir a comprar no futuro se o item de interesse estiver com boas condições de preço, frete etc.

Quando se trata de modelagem de dados em ambientes de Big Data, é muito comum o agrupamento dessas variáveis e até mesmo a seleção das mais representativas, aquelas que possuem maior impacto sobre a variável dependente. Um dos algoritmos mais utilizados para esse procedimento de agrupamento e classificação de variáveis é o KNN (*K Nearest Neighbor*), que será descrito em detalhes ao decorrer dessa monografia. Esse algoritmo possibilita a seleção das melhores variáveis de um *dataset* (base de dados), tornando-se muito interessante para uma base de dados com alto volume de atributos.

1.2 Motivação

Com o crescente volume de dados disponíveis para as empresas sobre seus clientes no ambiente de Big Data, é fundamental o uso correto dessas informações valiosas para a criação de campanhas de marketing mais segmentadas e personalizadas. Segundo o economista e guru do marketing Philip Kotler [4], conquistar um novo cliente pode custar de 5 a 7 vezes mais do que manter um cliente existente. Por esse motivo, cada vez mais é necessária a personalização de campanhas para retenção de clientes já existentes e aquisição de potenciais novos consumidores. Sobre os clientes já existentes, a revista Forbes [5] menciona que esse público tem uma probabilidade de 60-70% de comprar algo novamente, enquanto a chance de um novo cliente comprar algo fica entre 5-20%. Esses dados enfatizam a necessidade de as empresas focarem cada vez mais nos clientes que já estão presentes em suas bases de dados.

Em um cenário pós pandemia da COVID19, onde empresas do mundo inteiro sofrem com a escassez de recursos, com o aumento dos gastos alavancados pela alta inflação global e com a baixa expectativa de crescimento econômico, se torna fundamental o uso consciente e direcionado dos investimentos de captação de novos clientes por parte das empresas. Segundo relatório do IPEA (Instituto de Pesquisa Econômica Aplicada) [6], em 2022 a inflação global estava na casa dos 10% em outubro, mais de 3 vezes o valor pré pandemia, onde estava na casa dos 3% em fevereiro de 2020. Além disso, esse mesmo relatório [6] menciona uma diminuição da projeção do FMI (Fundo Monetário Internacional) sobre o crescimento do PIB (Produto Interno Bruto) global. Essa foi a terceira revisão negativa do PIB pelo FMI em 2022, onde em

abril a projeção era de 3,6%, sofrendo uma queda em julho para 2,9%, até chegar na casa dos 2,7% em outubro de 2022.

O cenário atual de alta inflação e baixa perspectiva de crescimento global no curto/médio prazo, pressiona ainda mais as empresas a serem mais eficientes com seus orçamentos de marketing e publicidade. Devido à alta acurácia dos modelos de aprendizado de máquina utilizados para seleção de clientes potenciais ou mapeamento de clientes existentes que tem alta chance de comprar novamente [21], a utilização de algoritmos de inteligência artificial se torna uma ótima alterativa para as empresas, fazendo com que o retorno financeiro dessas campanhas de captação e retenção seja muito positivo. Como o modelo de propensão utiliza comportamentos passados para prever ações futuras, ele se uma ótima opção para selecionar clientes que tem maior chance de compra, o que reduz significativamente os custos de comunicação e marketing, uma vez que apenas uma fração da base de clientes será impactada por campanhas e não a base inteira.

1.3 Objetivo

O objetivo geral desta pesquisa é apresentar, de maneira prática, a importância de um modelo de propensão. Ao longo dessa pesquisa serão detalhados os principais componentes, desde a explicação de uma análise exploratória dos dados no pré-processamento, até a comparação dos resultados obtidos pelos modelos de aprendizado de máquina. Sobre essa última etapa, as métricas de F-Score, acurácia, AUC Score e matriz de confusão, serão utilizadas para medir o desempenho dos modelos de Regressão Logística, *Random Forest* e Naïve Bayes, que vão ser utilizados após o pré-processamento dos dados e divisão de bases de teste e treinamento. Por fim, o modelo de propensão irá gerar uma predição de 0 até 100, que será aplicada para cada usuário, facilitando a seleção dos mais propensos, baseado nas predições mais altas.

1.4 Organização do Trabalho

Após a descrição deste capítulo 1, que apresentou a introdução, contextualização, motivação e objetivos desta monografia, serão vistos os seguintes capítulos na sequência:

- Capítulo 2 – Tem como objetivo descrever a fundamentação teórica sobre conceitos como aprendizado de máquina, pré-processamento, matriz de confusão, modelo KNN, Regressão Logística, *Random Forest*, Naïve Bayes, entre outros conceitos.
- Capítulo 3 – Exibe os trabalhos da literatura relacionados ao tema dessa monografia, modelo de propensão, mostrando suas aplicações e resultados.
- Capítulo 4 - Explora as potenciais contribuições desta monografia, apresenta a metodologia e proposta de solução.
- Capítulo 5 – Apresenta os resultados obtidos, comparando os 4 experimentos estudados.
- Capítulo 6 – Demonstra a conclusão da pesquisa e apresenta ideias para potenciais trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Aprendizado de Máquina

Aprendizado de máquina, do inglês *Machine Learning*, pode ser definido como um componente da área de ciências da computação e inteligência artificial, tendo como principal característica imitar a forma que humanos aprendem, melhorando continuamente sua precisão. Essa imitação da forma humana de aprendizagem é feita através do uso de dados e algoritmos. Esses algoritmos, por meio da utilização de métodos estatísticos, são treinados para realizar previsões ou classificações [7].

Graças aos avanços tecnológicos recentes em armazenamento e capacidade de processamento, foi possível a criação de modelos inovadores de aprendizado de máquina como o mecanismo de recomendação da Netflix e carros autônomos, por exemplo [7].

2.2 Pré-processamento

Parte fundamental no processo de preparação do modelo de aprendizado de máquina é a limpeza e a análise dos dados, conhecido como pré-processamento. Muitas vezes as bases de dados possuem “impurezas”, tais como valores nulos, colunas que não acrescentam nenhum valor para a análise, atributos desbalanceados, dados que precisam ser substituídos por outros valores etc.

Uma prática muito comum na limpeza dos dados é a função do pacote Pandas do Python chamada “Dropna”. Essa função remove os valores nulos de colunas (parâmetro axis = 1) e linhas (parâmetro axis = 0). Entretanto, essa prática não é a mais eficiente, pois remove uma coluna ou linha inteira apenas por ter um registro nulo, eliminando os demais registros que tinham informação. Por esse motivo, uma opção mais interessante é a função “fillna”, também da biblioteca Pandas do Python. Com essa função é possível substituir os valores nulos pela média, valor máximo, valor mínimo, mediana ou desvio padrão da coluna ou linha, mantendo assim os demais registros que possuem valores.

A normalização é uma das técnicas mais utilizadas para nivelar atributos que possuem escalas muito distintas. Segunda Pappa [8], normalização Min-Max transforma os valores dos atributos em um intervalo entre 0 e 1. Esse processo é feito através da seguinte fórmula:

Figura 1 – Fórmula de Normalização min-max

$$v' = \frac{v - min_A}{max_A - min_A}(nmax_A - nmin_A) + nmin_A$$

Fonte: Pappa [8]

A equação de normalização de min-max possui o símbolo *v´*, que representa a variável normalizada. Já as variáveis *min* e *max* representam os valores mínimos e máximos do conjunto do atributo, respectivamente. Com a finalidade de ilustrar como a normalização min-max funciona, se salário fosse um atributo do conjunto de dados, por exemplo, e variasse entre $12.000 e $98.000, o valor $73.600, após sofrer normalização min-max, corresponderia a 0.716 [8].

Figura 2 – Exemplo de Normalização min-max

$$v' = \frac{73.600 - 12.000}{98.000 - 12.000}(1 - 0) + 0 = 0.716$$

Fonte: Pappa [8]

A binarização é outra técnica de pré-processamento que ajuda na padronização da base de dados. Segundo Faria [9], a binarização é a transformação de atributos contínuos ou discretos em valores binários (0 ou 1). A codificação de 1-de-n representa 1 atributo binário para cada valor categórico. Se um atributo categórico tivesse 5 valores: péssimo, ruim, Ok, bom e ótimo; ele apresentaria 5 variáveis (colunas) binárias.

Tabela 1 – Exemplo de binarização com codificação 1-de-n

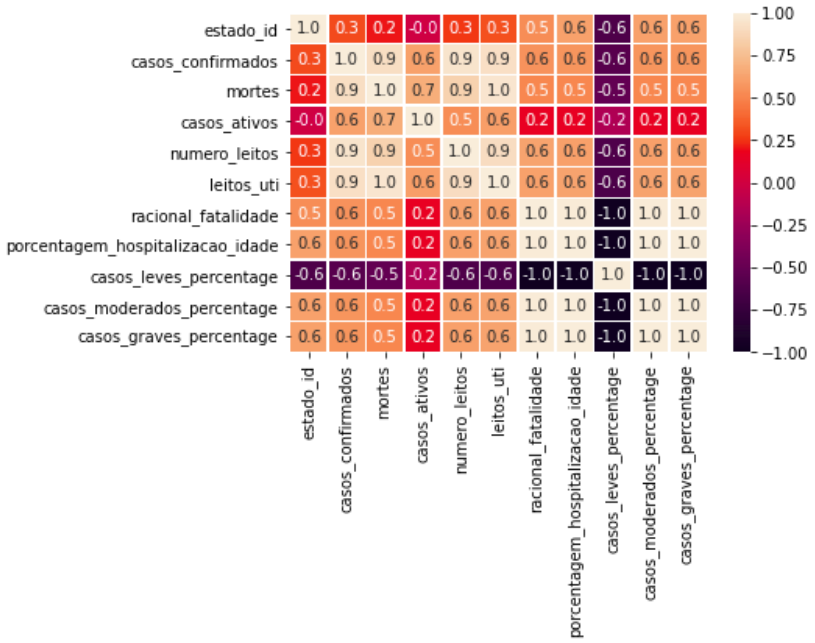
Valor Categórico	Valor Inteiro	x_1	x_2	x_3	x_4	x_5
Péssimo	0	1	0	0	0	0
Ruim	1	0	1	0	0	0
Ok	2	0	0	1	0	0
Bom	3	0	0	0	1	0
Ótimo	4	0	0	0	0	1

Fonte: Faria [9]

2.3 Correlação e Coeficiente de Correlação

Segundo Guimarães [10], a correlação é a relação linear existente entre duas ou mais variáveis distintas. Na estatística é muito utilizado o coeficiente de correlação, que mede o quão forte é a relação entre as variáveis. Os valores do coeficiente de correlação ficam entre -1 (opostamente relacionado) e 1 (altamente relacionado). Quando duas variáveis possuem correlação próxima a 1, significa que elas possuem uma correlação positiva. Ou seja, quando uma variável subir a outra também irá. Já uma correlação negativa, próxima de -1, significa que quando uma variável cresce a outra diminui [10].

Figura 3 – Matriz de coeficiente de correlação



Fonte: Guimarães [10]

Nesse exemplo de matriz de coeficiente de correlação, que utiliza uma base de dados fictícia de notificações de tuberculose no ano de 2019, o *heatmap* ajuda a evidenciar as variáveis com maiores correlações, sejam elas negativas ou positivas. Mortes, por exemplo, tem uma alta correlação positiva com leitos de UTI (1.0), número de leitos (0.9) e casos confirmados (0.9). Isso significa que o número de casos confirmados e leitos disponíveis tem alta correlação com o número de mortes nessa base fictícia. Em contrapartida, a porcentagem de hospitalização por

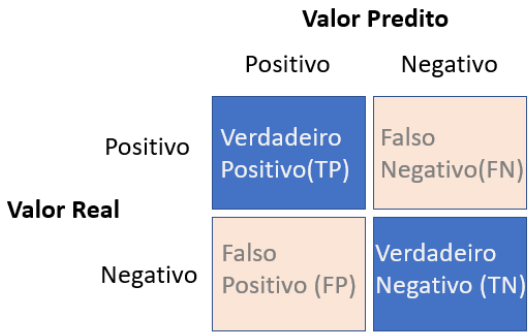
idade tem uma alta correlação negativa com porcentagem de casos leves (-1), o que significa que se um indicador subir o outro vai cair [10].

2.4 Matrix de Confusão

A matriz de confusão apresenta a frequência de classificação de cada classe do modelo, em formato de tabela. Se torna muito útil para demonstrar visualmente o resultado do desempenho do classificador [11]. Os possíveis resultados de classificação gerados pelo classificador são:

- Verdadeiro Positivo (True Positive TP) – Valor previsto e real são os mesmos. Valor previsto é positivo, assim como o valor real.
- Falso Positivo (False Positive FP) – Conhecido também como erro de tipo 1, falso positivo possui valores previstos e reais que não são os mesmos. O valor do modelo é falsamente previsto como positivo, enquanto o valor real é negativo.
- Verdadeiro Negativo (True Negative TN) – Valor previsto e real são os mesmos. Valor previsto é negativo, assim como o valor real.
- Falso Negativo (False Negative FN) – O erro de tipo 2, como também é conhecido o falso negativo, possui previsão incorreta do modelo. O valor do modelo é falsamente previsto como negativo, enquanto o valor real é positivo.

Figura 4 – Matriz de Confusão



Fonte: Próprio Autor

2.4.1 Acurácia

A acurácia é a porcentagem de acerto do modelo dentre as previsões realizadas. Segundo Mariano [12], a acurácia é um dos indicadores mais simples, mas, ao mesmo tempo, um dos mais importantes. Sua fórmula corresponde ao total de predições corretas dividido por total de predições. Ou seja, verdadeiro positivo (TP) somado a verdadeiro negativo (TN) dividido pelo total de predições.

Figura 5 – Fórmula de Acurácia

$$\text{Acurácia} = \frac{VP + VN}{VP + FP + FN + VN}$$

Fonte: Adaptado de Mariano [12]

2.4.2 Revocação

Conhecida como *Recall*, a revocação é a proporção de positivos corretamente identificados. São os verdadeiros positivos dividido pela soma de verdadeiro positivo (TP) e falso negativo (FN).

Figura 6 – Fórmula de Revocação

$$\text{Revocação} = \frac{VP}{VP + FN}$$

Fonte: Adaptado de Mariano [12]

2.4.3 Precisão

Da palavra *precision* em inglês, a precisão é a proporção de identificações positivas corretas, definindo o quão eficiente o modelo performou. A fórmula seria a divisão dos verdadeiros positivos pela soma de verdadeiro positivo e falso positivo.

Figura 7 – Fórmula de Precisão

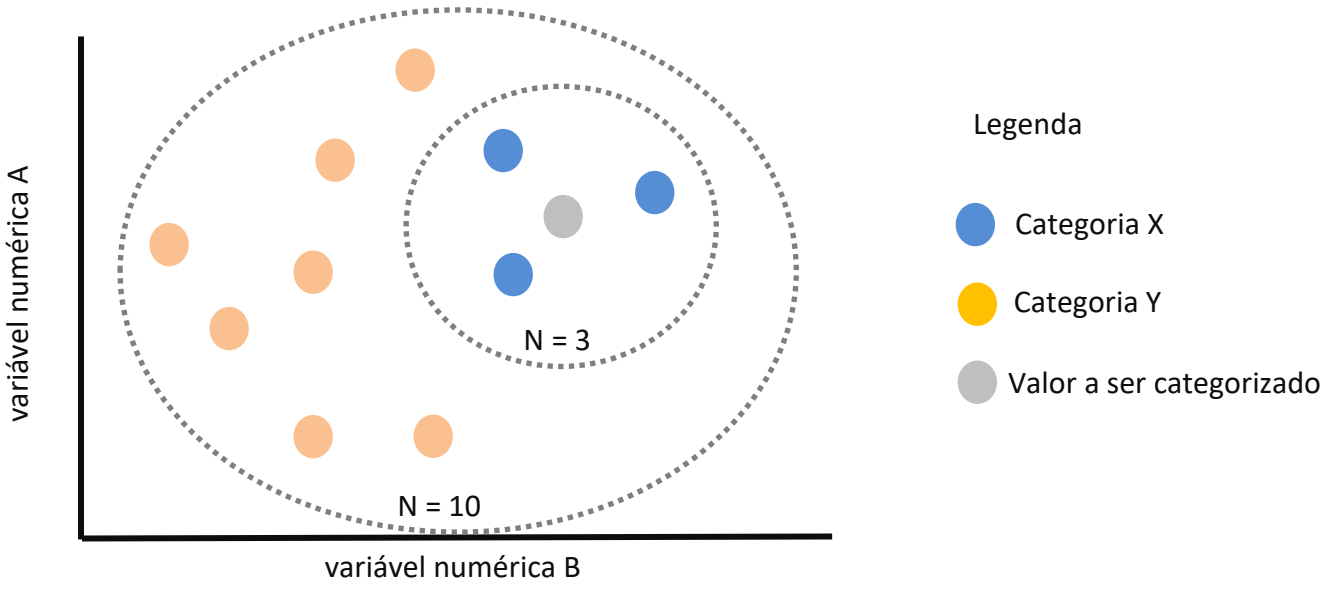
$$\textbf{Precisão} = \frac{VP}{VP + FP}$$

Fonte: Adaptado de Mariano [12]

2.5 Modelo KNN (K Nearest Neighbors)

O modelo KNN (K-Nearest Neighbors) é um algoritmo de classificação supervisionado que utiliza a proximidade entre os pontos para classificar um grupo de elementos [13]. Ou seja, quando um novo registro é inserido no modelo, ele será classificado em uma determinada categoria baseada na sua distância com os demais pontos. O KNN pode ser usado tanto para regressão como para classificação, assumindo que pontos similares podem ser encontrados perto um do outro [13]

Figura 8 – Racional de Agrupamento do K-Nearest Neighbors (KNN)



Fonte: Próprio Autor

O gráfico acima ilustra como um modelo de KNN funciona. Ambos os eixos do gráfico, x e y, são variáveis numéricas, representando um gráfico de dispersão onde os pontos coloridos são os pontos individuais de uma base de dados. Neste exemplo, é possível notar que existem dez pontos coloridos, entre azuis e laranjas, distribuídos no gráfico. Se o modelo KNN tivesse como parâmetro N (número de vizinhos mais próximos) o valor três, o ponto cinza seria categorizado como Categoria X, pois os três vizinhos mais próximos são da Categoria X. Porém, se o mesmo parâmetro N tivesse dez como valor, o ponto cinza seria classificado como Categoria Y, pois dos dez pontos mais próximos sete são da Categoria Y [13]. A distância entre pontos e a quantidade de vizinhos mais próximos são parâmetros ajustáveis e podem ser pré-definidos antes de rodar o modelo de KNN. As medidas de distância mais utilizadas são:

2.5.1 Distância Euclidiana

A distância euclidiana é a medida de distância mais utilizada. Ela mede o espaço em linha reta entre dois pontos. Se define como a raiz quadrada da soma dos quadrados das diferenças entre as coordenadas dos pontos (representados por p e q no exemplo abaixo) [13].

Figura 9 – Fórmula da distância euclidiana

$$dist = \sqrt{\sum_{k=1}^{m} (p_k - q_k)^2}$$

Fonte: IBM [13]

2.5.2 Distância Manhattan

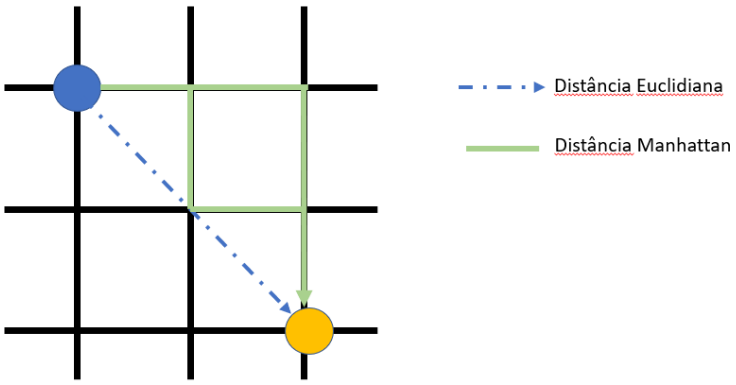
Distância Manhattan combina coordenadas e eixos, somando as diferenças absolutas entre dois pontos. Conhecida também como a distância do táxi ou distância L1 pelo seu formato em L, a distância Manhattan é frequentemente usada quando os dados não estão distribuídos uniformemente e possuem uma estrutura em forma de grade, como um tabuleiro de xadrez ou blocos de uma cidade [13].

Figura 10 – Fórmula da distância Manhattan

$$d(x,y) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2}$$

Fonte: IBM [13]

Figura 11 – Comparação entre distância Euclidiana e Manhattan



Fonte: Próprio Autor

2.5.3 Distância Minkowski

É considerada uma generalização da distância Euclidiana e de Manhattan. Ela considera as diferenças nas coordenadas e atribui um peso maior as diferenças maiores. A distância de Minkowski é tipicamente usada com o valor de p, número inteiro, sendo 1 ou 2, que correspondem à distância de Manhattan e à distância Euclidiana, respectivamente [13].

Figura 12 – Fórmula da Distância Minkowski

$$\textbf{Distância Minkowski} = \left(\sum_{i=1}^{n} \left|x_i - y_i\right|\right)^{1/p}$$

Fonte: Adaptado de IBM [13]

2.6 Regressão Logística

Considerada uma técnica de análise de dados, a Regressão Logística tem como objetivo encontrar relações entre dois fatores de dados. Depois que essa relação é encontrada, ela é utilizada para prever o valor de um dos fatores com base no outro. Além disso, a Regressão Logística procura estimar a probabilidade do valor da variável dependente em relação as demais variáveis, que podem ser categóricas ou não. Essa probabilidade é binaria (1 e 0), ou sim ou não [14].

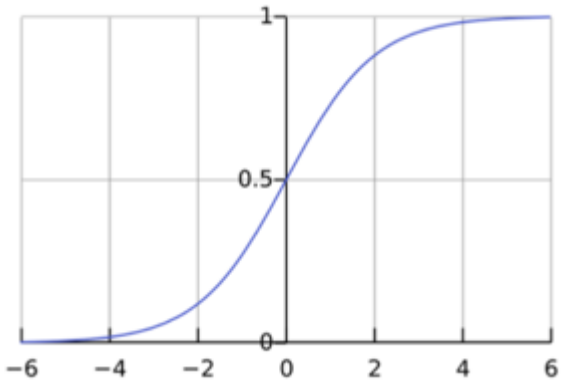
Figura 13 – Função Logística

$$f(x) = \frac{1}{1 + e^{-x}}$$

Fonte: Amazon [14]

A regressão logística pode ser calculada pela equação da função logística, onde o f(x) é a probabilidade de algo acontecer, como por exemplo um indivíduo ter uma determinada doença ou não. Já o x representa a função linear e a letra *e* corresponde a variável dependente [14].

Figura 14 – Representação gráfica da função Logística



Fonte: Amazon [14]

O gráfico acima mostra a relação entre x e y. Quando x cresce, a função tende a um. Já quando x decresce, a função tende a zero. A função Logística retorna apenas valores entre 0 e 1 para as variáveis dependentes, independentemente dos valores das variáveis independentes [14].

2.7 Naïve Bayes

O algoritmo usado em aprendizado de máquina Naïve Bayes é baseado no teorema de Bayes. Esse teorema permite, através de uma fórmula estatística, calcular a probabilidade de um evento acontecer, com base em eventos passados. Saritas e Yasar [15] definem Naïve Bayes como um classificador simples. Os autores também mencionam que esse classificador calcula a probabilidade através da contagem das frequências, além das contagens de combinações de valores de uma base de dados. Esse teorema assume que todos os atributos dos dados são independentes. A fórmula de cálculo para descobrir a probabilidade de um evento “A” acontecer considera a probabilidade de um evento “B” acontecer dado que evento “A” já tenha acontecido, vezes a probabilidade de “A” acontecer, dividido por probabilidade de “B” acontecer.

Figura 15 – Fórmula matemática do teorema de Bayes

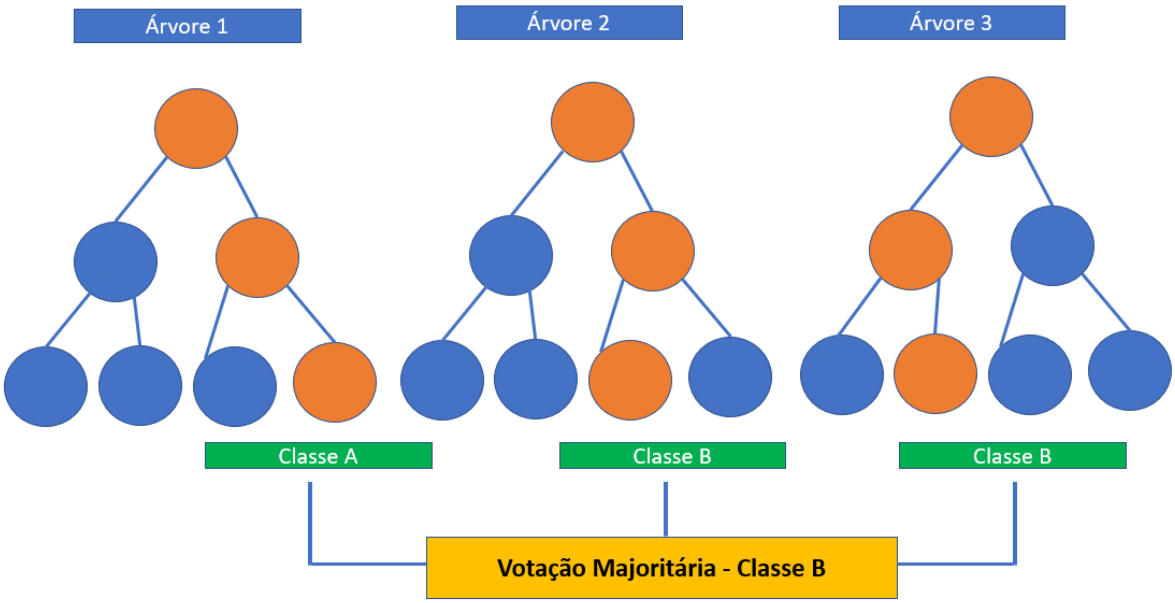
$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

Fonte: Saritas e Yasar [15]

2.8 *Random Forest*

O algoritmo usado em aprendizado de máquina *Random Forest* tem como principal objetivo predições de classificação ou regressão. Esse algoritmo cria várias árvores de decisão, com partes aleatórias da base de dados de treinamento em cada uma dessas árvores. Quando o objetivo é a classificação de um dado novo, cada árvore "manifesta" seu voto sobre a classe correta. Já na regressão, cada árvore cria a sua previsão e no final elas são combinadas. Esse algoritmo demanda 3 importantes parâmetros: Tamanho do nó, número de *estimators* (árvores) e número de atributos amostrados [16].

Figura 16 – *Random Forest* – Exemplo do Processo de Decisão

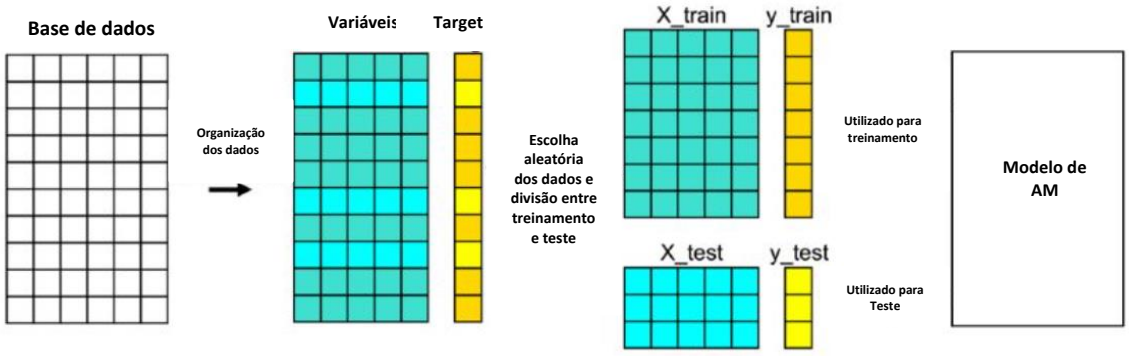


Fonte: Próprio Autor

2.9 Train *Test split*

Uma técnica de validação muito utilizada nos algoritmos de aprendizado de máquina é a divisão da base de dados entre treinamento e teste, conhecida como *test-split* ou *train test split*. Essa técnica consiste em dividir a base primeiro em treinamento, onde o algoritmo irá treinar a classificação ou predição para se familiarizar com as particularidades dos dados. Na sequência, novos registros são inseridos na etapa de teste, onde o modelo tentará prever os resultados. Segundo Galarnyk [17], train *test split* é um processo de validação de modelo que simula como o modelo performaria com novos dados. Geralmente esse processo consiste em dividir o *dataset* em 2 partes, treinamento e teste, aplicando uma porcentagem de divisão, como por exemplo 80% treinamento e 20% teste. Isso significa que, 80% dos dados são selecionados aleatoriamente sem reposição para serem treinados pelo modelo. Os 20% são "novos" dados que entrarão na parte de testagem e avaliação de performance do modelo.

Figura 17 – Composição de um Train *Test split*



Fonte: Adaptado de Galarnyk [17]

2.10 F-Score

A medida F-Score é um indicador de avaliação de modelo de aprendizado de máquina, sendo uma combinação das pontuações de precisão e revocação, itens abordados anteriormente em matriz de confusão. Segundo Rodrigues [18], a medida F-Score é uma média harmônica entre as duas métricas (precisão e revocação), estando mais próxima dos valores menores do que uma média aritmética simples. Se o valor de F-Score for baixo, significa que ou a precisão

ou a revocação estão baixos. A fórmula de cálculo seria a divisão dos verdadeiros positivos pelos verdadeiros positivos, somando com a metade dos falsos positivos mais os falsos negativos).

Figura 18 – Fórmula de cálculo da medida F-Score

$$\text{F-Score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)}$$

Fonte: Adaptado de Rodrigues [18]

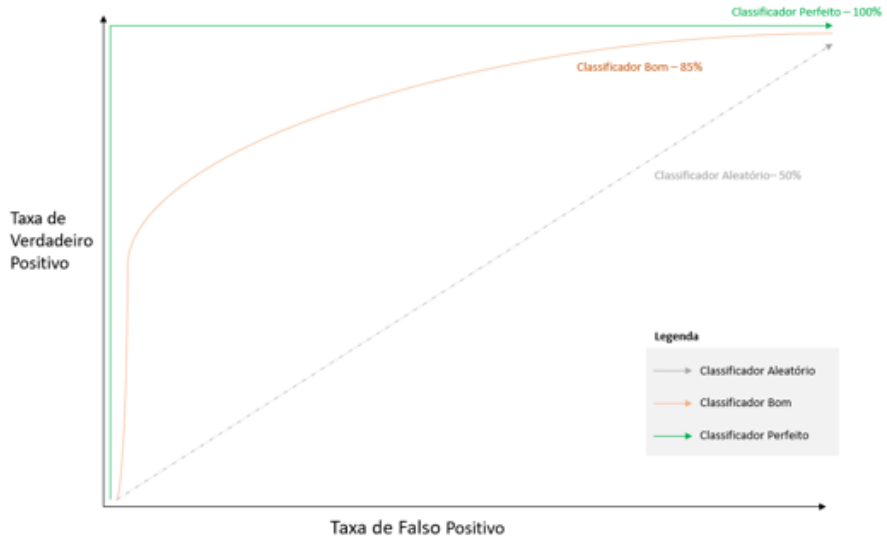
2.11 Score modelo de propensão

Após os modelos de aprendizado de máquina serem treinados e testados para classificação, existe a possibilidade de aplicar uma predição probabilística (*predict_proba)* para cada usuário da base de dados para saber, de 0 a 100, qual a propensão de uma determinada ação ser realizada. Segundo Khan [19], a função *predict_proba* retorna valores contínuos que representam a probabilidade de cada input fazer parte de cada categoria ou classe. No caso de o input fazer parte dos valores perto de 100 para uma categoria, significa que ele tem enormes chances de fazer parte dela. Mesma coisa no cenário contrário, quanto mais perto de 0 menor a chance de o input fazer parte daquela categoria.

2.12 Curva ROC e Curva AUC

A curva ROC (*Receiver Operating Characteristic*) ou Característica de Operação do Receptor, é uma ferramenta muito utilizada para medir o desempenho dos modelos de classificação. Ela traça a taxa de verdadeiros positivos no eixo Y do gráfico e falsos positivos no eixo X. Um modelo perfeito, com previsões 100% precisas, teria uma curva acentuada para o canto superior esquerdo do gráfico, o que indica perfeição nas classificações [20].

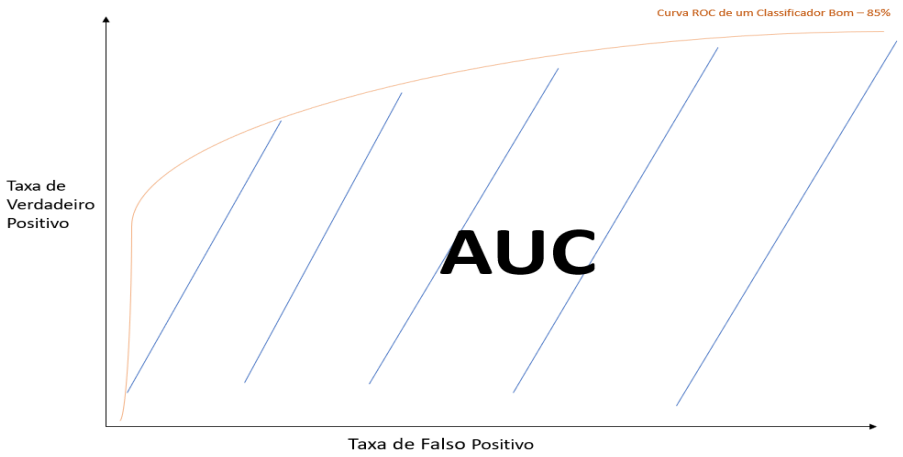
Figura 19 – Curva ROC



Fonte: Próprio Autor

A área AUC (*area under the curve*) também é uma métrica usada para medir o desempenho de um modelo de classificação e está fortemente relacionada com a curva ROC, sendo AUC a área sob a curva ROC. Os valores de AUC variam entre 0 e 1, onde um valor de 0,5 indica que o modelo tem desempenho fraco, quase que aleatório. Já um valor de 1 indica que o modelo é apto a classificar sem erros as duas classes [20].

Figura 20 – Área AUC



Fonte: Próprio Autor

3 TRABALHOS RELACIONADOS

Além dos trabalhos relacionados encontrados tradicionalmente em artigos, livros e periódicos, também foi selecionado para esse estudo de conclusão de curso exemplos práticos extraídos do site Kaggle. Esse site possui uma vasta e ativa comunidade de cientistas de dados, que constantemente resolvem desafios através de códigos de aprendizado de máquina. Cada participante desses desafios tem que resolver um problema pré-definido e a melhor resolução, a que obtém maior número de votos da comunidade, ganha o desafio. Tanto o problema pré-definido como a resolução ficam disponibilizados na íntegra. Para esse estudo de conclusão de curso, o desafio escolhido como referência foi o *"Customer propensity to purchase dataset"*. Esse desafio tem como objetivo identificar os clientes mais valiosos e com maiores tendencias de compra baseado em um *dataset* de e-commerce. Nesse *dataset* é possível encontrar variáveis como tipo de usuário (novo ou existente), se adicionou algum item ao checkout, fez ou não login etc.

3.1 *Strings* de busca

Com o intuito de buscar trabalhos relacionados a esta pesquisa, foram utilizadas algumas *strings* de busca para encontrar palavras-chave nos títulos de trabalho e pesquisas. O objetivo principal foi de encontrar trabalhos relacionados ao modelo de propensão, tendo como componentes o modelo de KNN para o agrupamento das variáveis e o modelo de Regressão Logística, Naïve Bayes e *Random Forest* para a classificação. Os operadores lógicos utilizados foram AND e OR que foram combinados para encontrar trabalhos entre 2000 e 2023 nas plataformas Springer Link, Scopus e ACM.

- Springer Link

"Propensity" AND "E-commerce"

("Score" OR "Model") AND "Propensity" AND ("E-commerce" OR "Shopping") AND
(("Logistic Regression" OR "Logistics Regression") OR
("KNN" OR "K Nearest Neighbor" OR "K Neighbors Nearest"))

- IEEE

(“Propensity Score” OR “Score Propensity” OR “Propensity model” OR “Propensity Model Score”) AND (“KNN” OR “K Nearest Neighbor” OR “K Neighbors Nearest”) AND (“Logistic Regression” OR “Logistics Regression” OR “Logistics Regression Model”) AND (“E-COMMERCE” OR “Online AND Shopping” OR “Purchase Online”)

- ACM

(([Title: "Propensity Model"] OR [Title: "Propensity Model Score"] OR [Title: "Propensity Score Model"] AND (([Title: “Logistic Regression”] OR [Title: “Logistic Regression Model”) AND ([Title: “KNN”] OR [Title: “K Nearest Neighbor”])) AND ([Title: “F-SCORE Score”] OR [Title: “F-SCORE Measure”]) AND ([Title: “E-Commerce”] OR [Title: “Online Shopping”] OR [“Title: Chance of Purchase”])

- Scopus

Title: “Propensity” AND “E-commerce”

- Google Scholar

Title: “Propensity” AND “E-commerce”

Os resultados de busca e os estudos selecionados podem ser encontrados na tabela a seguir. De um total de 40 resultados encontrados, 8 foram selecionados.

Tabela 2 – Resultados das buscas em bases de dados

Base de dados	**Resultado**	**Estudos Selecionados**
Springer Link	8	1
Scopus	6	2
Google Scholar	6	1
IEEE	13	2
Kaggle	7	2
Total	40	8

Fonte: Próprio Autor

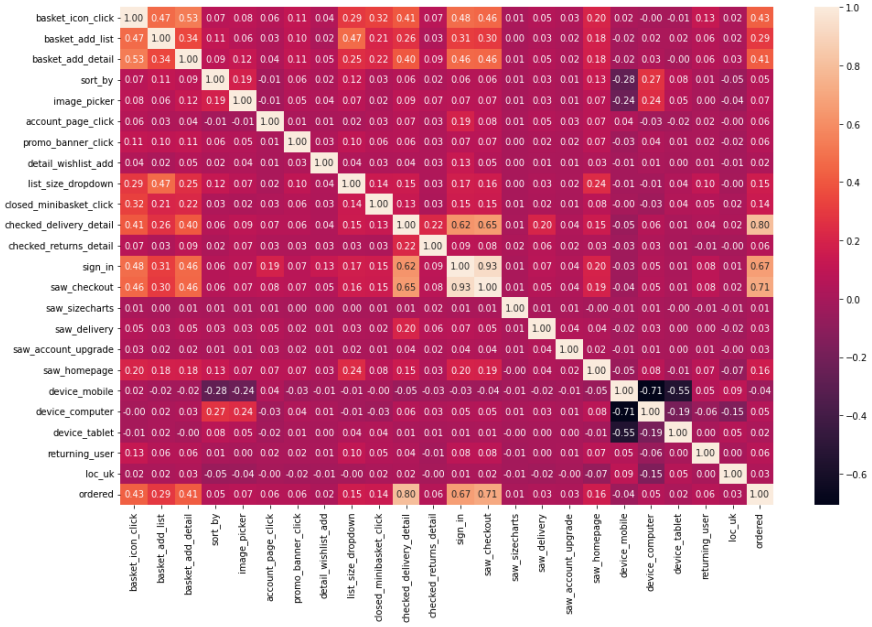
3.2 Descrição dos estudos selecionados

3.2.1 Chandrahasdhiraj (2022)

Em sua resolução para o desafio Kaggle (2018) de criar um modelo de propensão à compra, Chandrahasdhiraj [21] em 2022 utiliza a técnica de classificação de Gaussian Naïve Bayes para resolver o problema. O desafio consiste em desenvolver um modelo de aprendizado de máquina que possa prever qual a propensão de um cliente comprar algo de um determinado site, baseado em métricas de e-commerce como clicks, páginas visitadas etc.

Após análise rápida sobre a base de dados, Chandrahasdhiraj [21] nota que todos os atributos já estão com valores de 0 ou 1. Isso faz com que a normalização das variáveis não seja necessária, já que todos os atributos estão binarizados. Além disso, nota-se também que não há valores vazios ou nulos nessa base de dados, fazendo com que as técnicas do Pandas de fillna ou dropna, já abordadas anteriormente, sejam desnecessárias. O próximo passo adotado por Chandrahasdhiraj [21] após se assegurar que a base de dados está limpa e binarizada, é a criação de uma matriz de coeficiente de correlação para os atributos. Com a matriz fica visível que a variável "*Ordered*" (pedido realizado) é fortemente correlacionada com a variável "*checked_develivery_details*" (checagem dos detalhes de entrega) (0.80). Em contrapartida, fica evidente também que tanto comprar pelo telefone quanto pelo computador não tem correlação com a realização do pedido.

Figura 21 – Matriz de coeficiente de correlação Chandrahasdhiraj



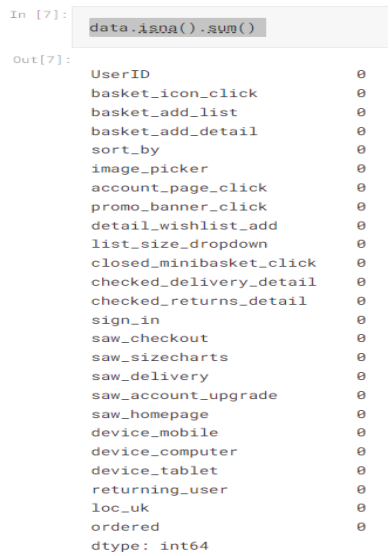
Fonte: Chandrahasdhiraj [21]

Na sequência, Chandrahasdhiraj [21] começa a separar a base de dados em teste (33%) e treinamento (67%). Para o algoritmo de classificação, ele utiliza o Gaussian NB, conhecido também como Gaussian Naïve Bayes. Chandrahasdhiraj [21] utiliza a matriz de confusão, medida F-Score e a acurácia como medidas de avaliação para o modelo. Para a acurácia Chandrahasdhiraj [21], obteve uma porcentagem alta de 98.8%, enquanto sua medida F-Score ficou em 87.4%. Para finalizar, Chandrahasdhiraj [21] utilizou a base de dados designadas para teste para realizar as predições finais e gerou uma lista de clientes mais propensos à compra online e seu respectivo Score de propensão.

3.2.2 Prakash (2022)

Diferentemente da abordagem adotada por Chandrahasdhiraj [21] que utilizou modelo de Gaussian NB, Prakash [22] adotou o algoritmo de Regressão Logística. Prakash [22] inicia importando as bibliotecas Python mais relevantes para esse projeto, como Pandas, Matplotlib, Seaborn e Numpy. Após uma breve análise dos dados utilizando o comando *data.isna().sum()* é possível notar a inexistência de valores nulos (N/A) no *dataset*.

Figura 22 – Lista de valores nulos por coluna utilizada por Prakash no desafio Kaggle



Fonte: Prakash [22]

A criação do modelo adotado por Prakash começa com a separação de 20% do *dataset* para teste e 80% para treinamento. Na sequência do *test split*, o modelo de Regressão Logística é aplicado, padronizando as variáveis através da função *StandardScaler*. Segundo a documentação oficial do scikit learn [23], a função *StandardScaler* padroniza os atributos removendo a média e dimensiona a variância unitária. Após esse processo, o modelo de Regressão Logística é avaliado utilizando a matriz de confusão. Com esse método, é possível calcular a precisão, acurácia, revocação e valor de F-Score. A acurácia obtida foi de 94.6%, ROC 96.9%, média ponderada da precisão em 98%, revocação em 95% e F-Score de 96%.

3.2.3 Kai-hong (2008)

Em seu estudo sobre a confiabilidade de compras online, Kai-hong [24] menciona que no mercado de e-commerce norte americano, em meados de 2008, 65% dos clientes desistiam das suas “cestas” de produtos na metade do fluxo de compra. Ele atribuiu essa desistência a falta de confiança, por conta da maior incerteza do e-commerce em relação a compras presenciais, e pela falta de conhecimento dos clientes de como realizar compras online.

Outro ponto relevante abordado por Kai-hong [24] é a alta percepção de risco que os chineses tinham em 2008 em relação a compras online. Para esse estudo, participando online e offline, foram considerados 102 indivíduos, 51% homens e 49% mulheres, com 85% dos participantes com menos de 35 anos. O questionário utilizado possuía 5 pontos de escala para testar as respostas dos consumidores sobre cada item, sendo 1 ponto uma alta discordância e 5 pontos uma alta concordância. As variáveis desse questionário tratavam de propensão ao risco, risco observado pelo consumidor, facilidade de uso identificado pelo usuário, utilidade e confiança inicial do consumir.

Tabela 3 - Variáveis utilizadas no questionário e modelo

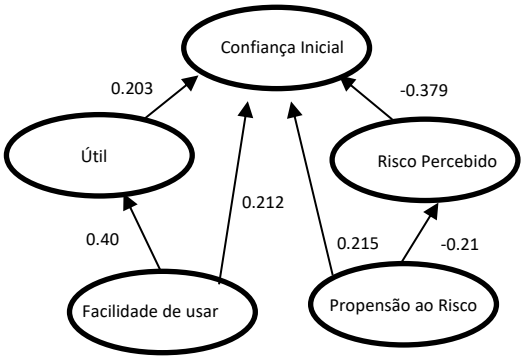
Variáveis	Itens	Descrição
Propensão ao Risco (RP)	RP1	As pessoas me disseram que pareço gostar de correr riscos
	RP2	Gosto de correr riscos
	RP3	Quando quero algo, estou disposto(a) a correr riscos para obtê-lo
Risco Percebido(PR)	PR1	Eu acredito que o risco de comprar online deste vendedor de comércio eletrônico é muito alto
	PR2	Existe uma grande probabilidade de perder muito ao comprar online deste vendedor de comércio eletrônico.
	PR3	Existe uma grande incerteza associada à compra online deste vendedor de comércio eletrônico.
Fácil de Usar(PEU)	PEU1	Aprender a usar este site seria fácil para mim.
	PEU2	Seria fácil me tornar habilidoso em usar este site.
	PEU3	Eu acho esse site fácil de usar
Útil(PU)	PU1	Usar este site pode melhorar meu desempenho de compras
	PU2	Usar este site pode aumentar minha efetividade de compras
	PU3	Eu acho útil usar este site
Confiança inicial do consumir (TRUST)	TRUST1	Este vendedor de comércio eletrônico é confiável.
	TRUST2	Este vendedor de comércio eletrônico mantém promessas e compromissos.
	TRUST3	O comportamento deste vendedor de comércio eletrônico atende às minhas expectativas.

Fonte: Adaptado de Kai-hong [24]

Após a realização da análise de confiabilidade SPSS, Kai-hong [24] comprovou que todas as medidas utilizadas possuíam um alpha maior que 0,7. Segundo Almeida [25], o SPSS, *Statistical Package for Social Science* (pacote estatístico para ciências sociais), é definido como pacote de análises estatísticas em ambientes altamente interativos, geralmente utilizado para auxiliar pesquisadores a analisar os resultados de questionários. Kai-hong[24] demonstra que indicador KMO (*Kaiser-Meyer-Olkin*) no valor de 0,64 se prova relevante em um nível de significância de p=0.000. Segundo IBM [26], o KMO é uma estatística que representa a proporção de variância em suas variáveis, podendo ser causada por fatores subjacentes. Quanto mais próximo de 1, mais uteis são os dados para uma análise de fator. Já valores abaixo de 0,5, significam que os resultados da análise de fatores não serão tão úteis.

Kai-hong [24] menciona que a variância total explicada por essas 5 variáveis (*Perceived ease of use*, *Perceived Usefulness, Risk Propensity*, *Perceived Risk e Trust*) é de 66,6%, o que significa que as medidas do questionário apresentam boa confiabilidade e validade. Essas 5 variáveis mais relevantes podem ser traduzidas para Facilidade de Usar, Útil, Propensão ao Risco, Risco Percebido e Confiança Inicial, respectivamente. De acordo com os resultados da pesquisa, a confiança inicial de comprar online tem uma correlação positiva com a percepção de facilidade de uso do e-commerce.

Figura 23 – Valores de Correlação entre diferentes percepções dos clientes sobre e-commerce



Fonte: Adaptado de Kai-hong [24]

Em sua conclusão, o autor sugere que as empresas de e-commerce estimulem os consumidores mais adeptos ao risco primeiro, que assim esses clientes mostrariam aos demais que comprar pela internet é seguro. Além disso, Kai-hong [24] mostra que clientes que achavam as plataformas de e-commerce fáceis de usar e de bom valor agregado, tinham maior confiança em comprar online do que clientes que não achavam plataformas de e-commerce úteis.

3.2.4 Rodríguez (2022)

Na literatura é possível encontrar também estudos relacionados a correlação entre o nível de educação do consumidor e a propensão a compra. Em seu estudo conduzido na Espanha entre 2015-2018, Rodríguez [27] testa duas hipóteses relevantes ao e-commerce:

A. Se há uma relação positiva entre o nível educacional e propensão ao uso do e-commerce
B. Se o nível de conhecimento dos clientes sobre computadores/informática tem alguma relação a inversão ao risco em compras online.

Com a ajuda do modelo de propensão utilizado no estudo, foi possível observar que indivíduos com níveis mais altos de educação tinham maior probabilidade de comprar algo na internet. Rodriguez [27] apresenta que indivíduos com nível universitário mostraram uma probabilidade maior que indivíduos com apenas o ensino fundamental a usar o e-commerce.

Outro descobrimento relevante desse estudo foi a comprovação que indivíduos com maior conhecimento de informática tem maior propensão de uso de e-commerce e menor percepção de risco durante as compras online.

Esses resultados de correlação entre nível educacional e propensão ao uso do e-commerce são derivados de um modelo de regressão bivariada. Porém, antes do modelo ser aplicado, houve um "balanceamento" das respostas dos participantes da pesquisa, onde as respostas positivas para uso da internet tiveram peso maior que as respostas negativas. Além disso, respostas de baixa frequência de uso da internet também foram penalizadas. Após esses ajustes, a variável dependente "Já realizou em algum momento uma compra de produtos ou serviços na internet?" foi selecionada. Assim como conhecimento de informática, nível educacional, idade e ocupação profissional foram selecionados como variáveis independentes. O modelo de Regressão Logística bivariada foi aplicado nas variáveis independentes, que explicaram 79.5% da alteração da variável dependente, para medir a propensão ao consumo online. Com a comprovação da influência do nível educacional sobre a propensão a compra online, Rodriguez confirma sua primeira hipótese.

Tabela 4 – Tabela de parâmetros do modelo de propensão para E-commerce

Variáveis	B	S.E	Wald	df	Significância	Exp(B)
Indicador ponderado de conhecimento de Informática	5.537	0.111	2,488.764	1	0	253.993
Tipo de Profissão	-0.192	0.018	117.155	1	0	0.826
Faixa Etária	-0.245	0.023	112.731	1	0	0.783
Renda Familiar	0.063	0.013	22.031	1	0	1.065
Constante	-0.930	0.102	83.129	1	0	0.395

Fonte: Adaptado de Rodríguez [27]

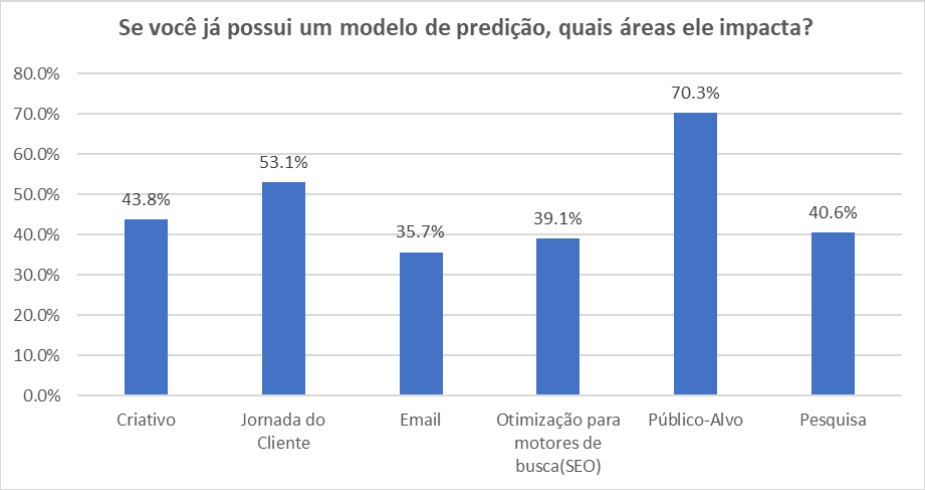
Já para a segunda hipótese, se os níveis de conhecimento computacional/informática têm relação a aversão ao risco em compras online, o autor do estudo utiliza um modelo de Regressão Logística multivariada para saber a influência das variáveis independentes (renda familiar, idade, gênero e ocupação profissional) sobre a variável dependente (aversão ao risco).

Os resultados proporcionados pelo modelo mostram que quanto maior é o conhecimento de informática do indivíduo, menor é a aversão em comprar algo online. Outro dado interessante foi que quanto mais velho é o indivíduo, maior é a aversão ao risco, mostrando altas taxas de aversão ao risco na faixa etária de 56-70. Além disso, o modelo também mostrou que quanto maior a renda do indivíduo menor é a sua aversão ao risco com compras online. O cenário oposto também foi comprovado, onde os participantes do estudo com menor renda apresentaram maiores aversões ao risco.

3.2.5 Gupta e Joshi (2022)

Em seu estudo "Técnicas Analíticas Preditivas para melhorar o desempenho de marketing e a Experiência Personalizada do Cliente", Gupta e Joshi [28] ressaltam a importância do monitoramento ativo e constante durante cada etapa da construção de um modelo de predição. Todos os passos, desde o levantamento de requisitos antes do desenvolvimento até o *deploy* e monitoramento do modelo devem ser trabalhados passo a passo para garantir que o sistema seja usado da forma correta, evitando falhas durante o processo.

Figura 24 – Áreas impactadas por modelos preditivos



Fonte: Adaptado de Gupta e Joshi [28]

Como apontado pela figura acima, a área mais impactada por modelos preditivos é a de público-alvo. Modelos preditivos tem como característica analisar dados e acontecimentos do passado para determinar a probabilidade de um certo acontecimento futuro ocorrer. Para selecionar os indivíduos mais apropriados (público-alvo), técnicas de clusterização, que agrupam indivíduos com características similares, são muito utilizadas para agrupar clientes parecidos. Uma dessas técnicas é a de KNN (K-nearest neighbors), já descrita anteriormente nesse TCC, onde a classificação categórica de um novo dado é baseada na distância em relação aos demais dados ou pontos. O intuito dessa técnica é de agrupar dados similares em grupos (clusters), baseado em características parecidas.

Gupta e Joshi [28] também abordam em seu estudo a relevância de modelos de propensão. Esse tipo de modelo tende a prever a probabilidade de uma determinada ação do cliente acontecer. As autoras mencionam um caso de uso concreto, onde a petshop online Fin & Fur adotou um modelo de propensão, que ajudou a alavancar as vendas da empresa substancialmente. A personalização de anúncios foi feita com base no produto mais provável a ser vendido para cada cliente específico. Com base nesse ranking de produtos, Fin&Fur pôde determinar quais abordagens resultariam nas melhores repostas de cada consumidor. Além disso, o ranking gerado pelo modelo de propensão ajudou a empresa a otimizar a aquisição de clientes usando procedimentos de e-mail automatizados para indivíduos previamente identificados como propensos a comprar.

3.3 Considerações Finais do Capítulo

As soluções propostas por Chandrahasdhiraj [21] e Prakash [22] para resolver o mesmo desafio de E-commerce do Kaggle seguiram etapas diferentes, o que acabou afetando o resultado dos modelos. Após a limpeza do *dataset*, Prakash [22] cria uma função para correlação de variáveis, chamada de *correlation* (correlação). Porém, o resultado dessa função apresenta um problema. Ele aponta que as variáveis *saw_checkout* (verificar detalhes da entrega), juntamente com *device_computer* (Aparelho de Computador), devem ser retiradas da base de dados. Contudo, como apresentado na resolução proposta por Chandrahasdhiraj [21], o coeficiente de correlação entre as variáveis *saw_checkout* e *ordered* é uma das mais altas do *dataset*, com 71%. Logo, a função utilizada por Prakash acaba eliminando uma importante variável do *dataset*, impactando diretamente o resultado do modelo. Outro lado negativo do

estudo realizado por Prakash [22] que vale a pena ser mencionado é o *output,* o resultado de final da sua pesquisa. No arquivo final, Prakash [22] entrega uma base de clientes com apenas valores em 0 para a coluna de *result* ou resultado. Seu modelo não conseguiu prever nenhuma situação de compra para nenhum cliente. Ou seja, não conseguiu realizar previsões com o novo conjunto de dados (teste).

Como apresentado por Kai-hong [24], 65% dos clientes norte-americanos desistiam das suas "cestas" de produtos na metade do fluxo de compra online. Esse dado corrobora com os resultados da matriz de coeficiente de correlação apresentada por Chandrahasdhiraj [21], onde altas taxas de correlação entre "pedido realizado" (*ordered*) e "verificar detalhes da entrega (*saw_checkout)* são apresentadas. Ao passar da metade do fluxo de compra, a chance do pedido ser concretizado é muito grande. Nessa etapa do fluxo de compra, o cliente já apresenta maior confiança no site e sabe como utilizar a internet, fatores fundamentais segundo Kai-hong [24] para o uso adequado do e-commerce.

4 PROPOSTA DE PESQUISA

4.1 Considerações Iniciais

Esse capítulo é dedicado ao detalhamento da proposta desta pesquisa, cujo objetivo é selecionar os clientes mais propensos a comprar algo de uma empresa de e-commerce fictícia. A proposta que será abordada envolve o carregamento dos dados de treinamento e teste, que já foram separados previamente pela plataforma Kaggle. Será realizada também uma análise exploratória e identificação de correlações entre a variável dependente e as variáveis independentes. Na sequência, 4 experimentos distintos de *test-split* e eleição de variáveis serão testados. Dentro de cada experimento, 3 algoritmos de aprendizado de máquina (Naive Bayes, Regressão Logística e *Random Forest*) serão testados e suas eficiências de predição serão medidos. Por fim, a melhor combinação de experimento e algoritmo será utilizada para realizar as predições de melhores clientes.

Sobre as bases disponibilizadas pelo site Kaggle, é importante ressaltar que a base de teste contém apenas valores para serem preditos. Ou seja, nenhum usuário efetuou nenhuma compra nessa base. Ela será usada apenas para a inclusão das predições finais. Esse comportamento na disponibilização das bases do Kaggle é algo comum. Segundo Súniga [29], isso é feito para que o Kaggle divida as bases de teste em duas partes, uma designada para o cálculo da pontuação pública e a outra para pontuação privada, que é apenas revelada no final da competição. Se o foco for apenas a pontuação pública, existe grandes chances de ocorrer um *overfitting*. Sendo assim, a base de treinamento será subdividida em base de treinamento e base de validação, para as 4 abordagens distintas. Além da seção de considerações iniciais, esse capítulo é divido em mais 3 seções. Na seção 4.2 será abordada a metodologia utilizada para o desenvolvimento das tarefas da pesquisa. Em seguida, na seção 4.3, a proposta de solução será descrita, detalhando o passo a passo de como será executada a resolução desta pesquisa.

4.2 Metodologia

O experimento será conduzido utilizando o Jupyter Notebook do Python, uma plataforma altamente eficiente para todas as fases desta pesquisa. Desde a análise dos dados até a implementação e avaliação dos resultados dos modelos de aprendizado de máquina. A base

de dados que será utilizada é fictícia e foi obtida do site Kaggle, contendo informações sobre os comportamentos dos clientes de uma página de e-commerce. A base consiste em 25 colunas e em 607.056 usuários, somando as bases de treinamento (455.401) e teste (151.655). Porém, como mencionado anteriormente, a base de teste (151.655) vai ser utilizada apenas para a alocação das predições, já que os valores de *ordered* estão zerados. A lista completa de variáveis presentes nas bases está presente na tabela abaixo.

Tabela 5 – Lista de variáveis da base de dados Kaggle

Variáveis Originais	Descrição	Descrição Traduzida
UserID	A unique identifier for the visitor	Um identificador único para o visitante?
basket_icon_click	Did the visitor click on the shopping basket icon?	O visitante clicou no ícone do carrinho de compras?
basket_add_list	Did the visitor add a product to their shopping cart on the 'list' page?	O visitante adicionou um produto ao carrinho de compras na página 'listagem'?
basket_add_detail	Did the visitor add a product to their shopping cart on the 'detail' page?	O visitante adicionou um produto ao carrinho de compras na página 'detalhes'?
sort_by	Did the visitor sort products on a page?	O visitante ordenou produtos em uma página?
image_picker	Did the visitor use the image picker?	O visitante usou o seletor de imagens?
account_page_click	Did the visitor visit their account page?	O visitante visitou a página da sua conta?
promo_banner_click	Did the visitor click on a promo banner?	O visitante clicou em um banner promocional?
detail_wishlist_add	Did the visitor add a product to their wishlist from the 'detail' page?	O visitante adicionou um produto à sua lista de desejos a partir da página de 'detalhes'?
list_size_dropdown	Did the visitor interact with a product dropdown?	O visitante interagiu com um menu suspenso de produtos?
closed_minibasket_click	Did the visitor close their mini shopping basket?	O visitante fechou a cesta de compras mini?
checked_delivery_detail	Did the visitor view the delivery FAQ area on a product page?	O visitante visualizou a área de perguntas frequentes sobre entrega em uma página de produto?
checked_returns_detail	Did the visitor check the returns FAQ area on a product page?	O visitante verificou a área de perguntas frequentes sobre devoluções em uma página de produto?
sign_in	Did the visitor sign in to the website?	O visitante fez login no site?
saw_checkout	Did the visitor view the checkout?	O visitante visualizou a página de checkout?
saw_sizecharts	Did the visitor view a product size chart?	O visitante visualizou uma tabela de tamanhos de produto?
saw_delivery	Did the visitor view the delivery FAQ page?	O visitante visualizou a página de perguntas frequentes sobre entrega?
saw_account_upgrade	Did the visitor view the account upgrade page?	O visitante visualizou a página de upgrade de conta?
saw_homepage	Did the visitor view the website homepage?	O visitante visualizou a página inicial do site?
device_mobile	Was the visitor on a mobile device?	O visitante estava em um dispositivo móvel?
device_computer	Was the visitor on a desktop device?	O visitante estava em um dispositivo de mesa?
device_tablet	Was the visitor on a table device?	O visitante estava em um dispositivo de desktop?
returning_user	Was the visitor new or returning?	O visitante era novo ou retornando?
loc_uk	Was the visitor located in the UK, based on their IP address?	O visitante estava localizado no Reino Unido, com base no endereço IP?
ordered	Did the customer place an order?	O cliente fez um pedido?

Fonte: Próprio Autor

Devido à limpeza prévia dos dados, a etapa de pré-processamento não é necessária para essa base de dados. Todos os valores desses atributos já foram binarizados, alternando entre 0 e 1, e sem a existência de dados faltantes ou nulos. Com o intuito de explorar a melhor combinação de divisão dos dados de teste, treinamento e seleção de variáveis, serão realizados 4 experimentos ao todo. Experimento 1 terá a base de dados dividida aleatoriamente em 80% para treinamento e 20% para teste (validação), com todas as variáveis do *dataset*. O experimento 2 terá a mesma divisão, mas com apenas as 10 variáveis mais relevantes, de acordo com o modelo KNN. No experimento 3 haverá uma divisão de 75% treinamento e 25% validação, com todas as variáveis do *dataset*. Por fim, o experimento 4 terá a mesma divisão de validação e treinamento do experimento 3, mas considerando apenas as 10 variáveis mais relevantes segundo o modelo KNN.

Tabela 6 – Divisão dos experimentos

Experimento	**Divisão treinamento/validação**	**Seleção de variáveis**
Experimento 1	80% treinamento & 20% validação	Sem seleção
Experimento 2	80% treinamento & 20% validação	Com seleção
Experimento 3	75% treinamento & 25% validação	Sem seleção
Experimento 4	75% treinamento & 25% validação	Com seleção

Fonte: Próprio Autor

Antes da divisão das bases em treinamento e validação para os experimentos mencionados acima, uma análise de coeficiente de correlação será realizada, utilizando a função *corr* da biblioteca pandas. Além disso, através das bibliotecas *matplotlib.pyplot* e *Seaborn*, um gráfico de *heatmap* será apresentado para analisar quais variáveis independentes tem maior correlação com a variável dependente, nesse caso *ordered.* Após essa análise de coeficiente de correlação, será feita a divisão da base de dados em treinamento e validação com a biblioteca Python *sklearn train_test_split*. Essa biblioteca será utilizada para separar aleatoriamente os usuários de treinamento e teste, através do parâmetro *test_size*. Os modelos adotados, como Regressão Logística, Gaussian Naïve Bayes e *Random Forest*, serão utilizados através das bibliotecas do Python *sklearn* LogisticRegression, GausssianNB e RandomForestClassifier, respectivamente. Adicionalmente, o modelo que será adotado para a seleção de variáveis, o K-Nearest Neighbors (KNN), vem também da biblioteca *sklearn* do Python, com a especificação de KNeighborsClassifier e o pré-requisito de selecionar o número de vizinhos (k) a serem considerados. Para a utilização do KNN nesse estudo, será adotada a distância Euclidiana.

Os 3 modelos de classificação mencionados durante a monografia foram selecionados para representar categorias diferente de algoritmos. O *Random Forest* representa os algoritmos de Ensemble, que combinam previsões de vários submodelos para chegar na previsão final. Já

o algoritmo de Regressão Logística faz parte dos explicativos. Esse grupo de algoritmos permite identificar e entender melhor as variáveis que têm relação estatisticamente significativa com o resultado. Por fim, o algoritmo Naïve Bayes faz parte dos algoritmos probabilísticos, estimando a probabilidade de uma classe com base em acontecimentos passados.

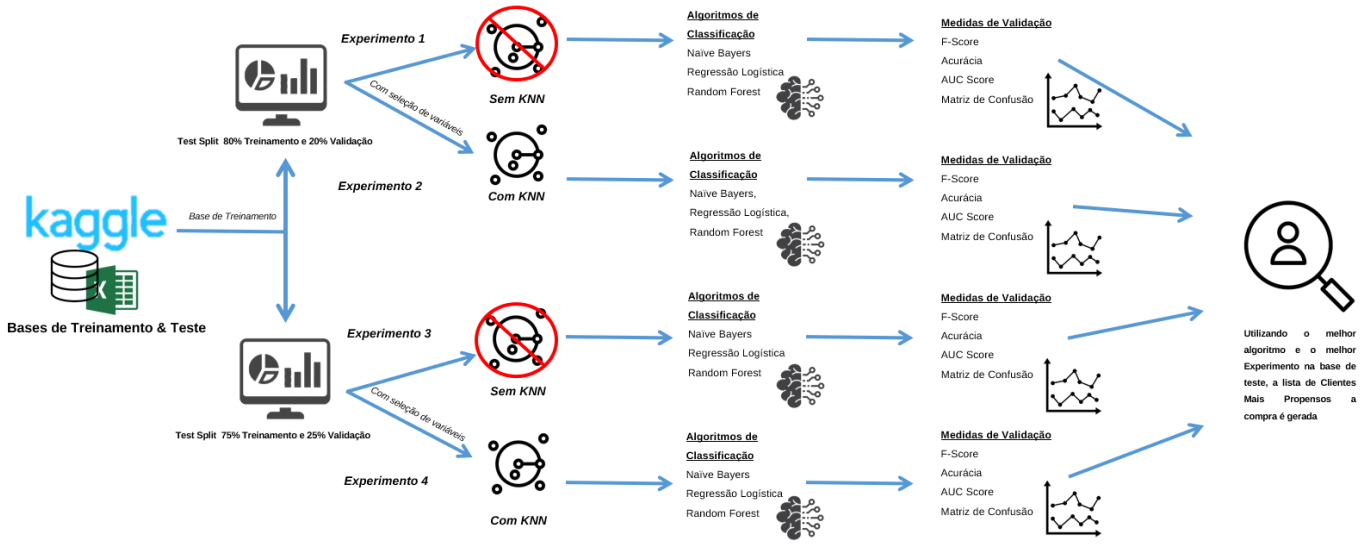
Uma vez que os modelos foram treinados e preditos, é necessária uma comparação entre as medidas de Acurácia, Medida F-Score, matriz de confusão e AUC Score, para que seja possível a seleção do melhor modelo. Essas medidas de apuração e avaliação dos modelos são provenientes da biblioteca *sklearn* metrics, com os nomes de *accuracy_Score, F-Score_Score, confusion_matrix,* e *roc_auc_Score*, respectivamente. Utilizando essas métricas de avaliação, é possível selecionar o modelo com melhor desempenho e maior chance de conversão. Após a comparação dessas métricas entre os 4 experimentos, o de melhor resultado geral será utilizado para realizar as predições na base de teste. Além da predição, uma nota de 0 - 100 será aplicada para cada usuário da base de teste, com 100 sendo a maior probabilidade de compra.

4.3 Proposta de solução

O primeiro passo para a resolução desse problema será a importação dos dados de treinamento e teste em csv, previamente já disponibilizadas pelo site Kaggle, no Python através da biblioteca Pandas. Em seguida uma análise exploratória será necessária para identificar tendências e correlações entre variáveis. Como a base de dados já está binarizada, variáveis apresentam atributos 0 ou 1 e sem valores nulos, a etapa de limpeza de dados não se torna necessária. Na sequência a análise exploratória é feita através do pacote Matplotlib e Seaborn, que resultará no *heatmap* com os coeficientes de correlação.

Como mencionado na seção de metodologia, a abordagem de resolução será dividida em 4 experimentos. Experimentos 1 e 2 serão realizados em um notebook do Python, onde as abordagens com todas as variáveis e com apenas as top 10 melhores serão realizadas com um *test split* de 80% treinamento e 20% validação. Em outro notebook os mesmos estudos serão realizados para os experimentos 3 e 4, mas com um *test split* de 75% treinamento e 25% validação. Cada experimento utilizará os 3 algoritmos de classificação, *Random Forest*, Naïve Bayes e Regressão Logística.

Figura 25 – Fluxo das Abordagens para a Resolução do Problema Proposto



Fonte: Próprio Autor

Após a etapa de análise de dados e separação das bases em treinamento e validação através da biblioteca *train_test_split* do *sklearn*, a seleção das variáveis principais será realizada através do algoritmo de KNN KNeighborsClassifier para os experimentos 2 e 4. Com as variáveis mais relevantes já selecionadas, o seguinte passo será treinar e aplicar os modelos de classificação *Random Forest* (RandomForestClassifier), Regressão Logística (LogisticRegression) e Gaussian NB (GausssianNB) na base de treinamento e validação. Por fim, medidas como Acurácia, Medida F-Score, matriz de confusão e pontuação AUC, serão utilizadas para medir a eficiência dos modelos e ajudar eleger o melhor com base nas melhores predições. Após a seleção dos melhores resultados de experimento e algoritmo, a lista de clientes mais propensos a compra será gerada com a utilização da função *predict_proba*, do conjunto *sklearn*. Uma pontuação de 0 a 100, sendo 100 mais propenso a compra, será aplicada para cada usuário da base de teste, que até então estava intacta. Os usuários que apresentarem maiores pontuações, serão selecionados. Por fim, vale ressaltar que tanto os dois *scripts* Python, com experimentos 1 e 2 e o outro com experimentos 3 e 4, assim como o Excel com a lista final de clientes mais propensos, estão disponíveis no Gitlab (caminho completo no Apêndice A).

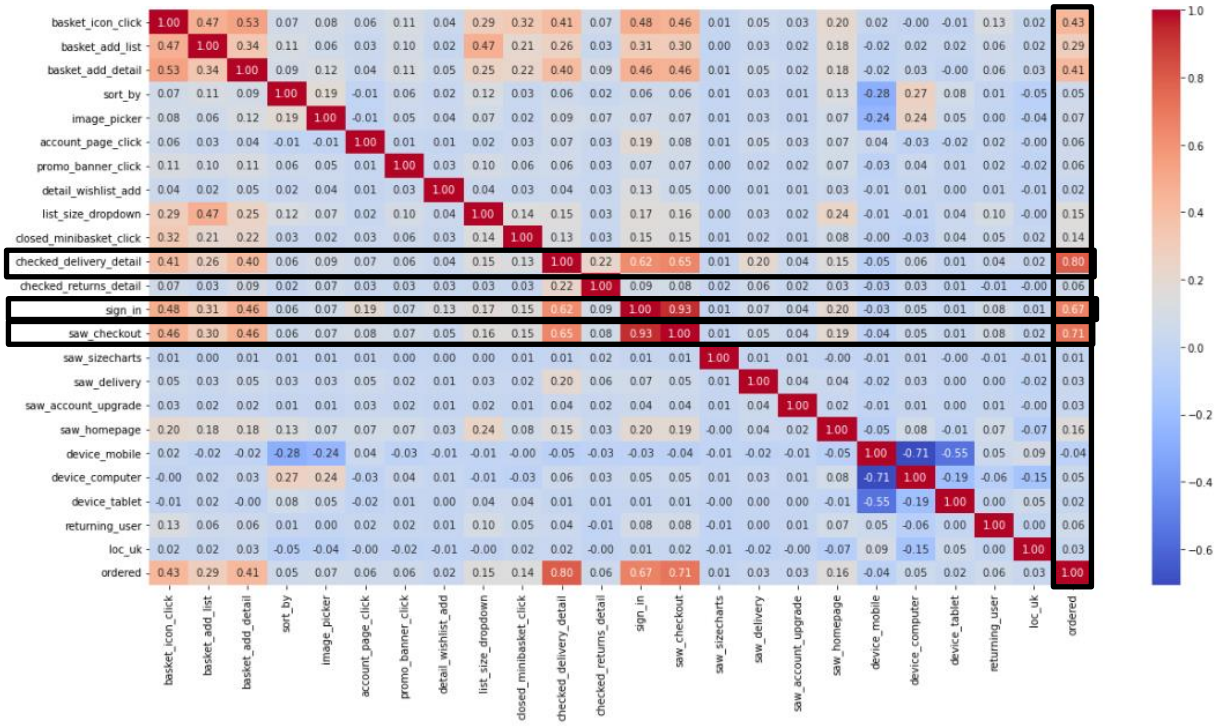
5 ANÁLISE DOS RESULTADOS

5.1 Análise dos Resultados

O primeiro passo adotado nessa pesquisa foi a conferência do estado da base de dados, através da função *info* da biblioteca pandas. Com ela foi possível observar que todos os 455.401 registros da base de treinamento estavam presentes, sem nenhum dado faltante. Como mencionado anteriormente, essa base já foi disponibilizada pelo site Kaggle de maneira "limpa", com todos os valores já binarizados, sem a necessidade de qualquer outro tipo de pré-processamento.

O segundo passo foi a análise das variáveis mais correlacionadas com a variável dependente *ordered*. Como demonstrado na figura abaixo, as variáveis mais correlacionadas positivamente foram *checked_delivery_detail (80%), saw_checkout (71%)* e *sign_in (67%).*

Figura 26 – Matriz de Coeficiente de Correlação – Base Treinamento



Fonte: Próprio Autor

Após a análise de correlação, a base de treinamento disponibilizada pelo site Kaggle foi dividida em duas partes: 80% treinamento e 20% validação e 75% treinamento e 25% validação. Com essa divisão, os grupos de experimentos de 1 a 4 foram montados. Os resultados foram os seguintes:

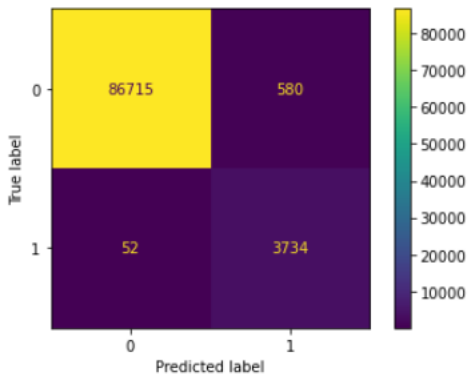
5.1.1 Experimento 1

Dentre as 25 colunas presentes no *dataset*, 23 foram incluídas em treinamento e teste de X e uma incluída no treinamento e validação de y. As colunas *'UserID'* e *'ordered'* foram removidas das variáveis de X, enquanto apenas a variável *'ordered'* foi considerada para as variáveis de treinamento e validação de y. Oitenta por cento da base foi utilizada para o *fit* e predição dos 3 modelos de classificação, *Random Forest*, Regressão Logística e Naïve Bayes. Isso representa 364.320 clientes alocados para as variáveis X_treinamento e y_treinamento e 91.081 para X_teste e y_teste.

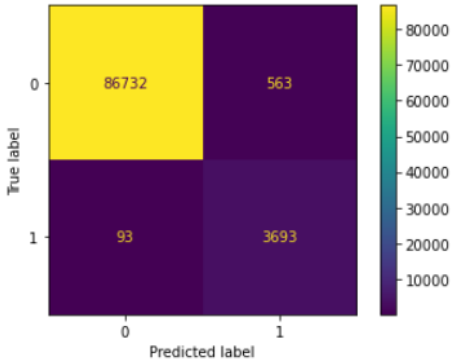
O loop *FOR* foi inserido nessa parte do script para que o processo se torne mais automático, não necessitando a inclusão manual de cada um dos modelos de classificação. Após o *fit* e a predição, as métricas de avaliação foram aplicadas, como F-Score, acurácia, AUC Score, revocação e Matriz de confusão. Essa última métrica apresentou os seguintes resultados:

Figura 27 – Matrizes de Confusão Experimento 1

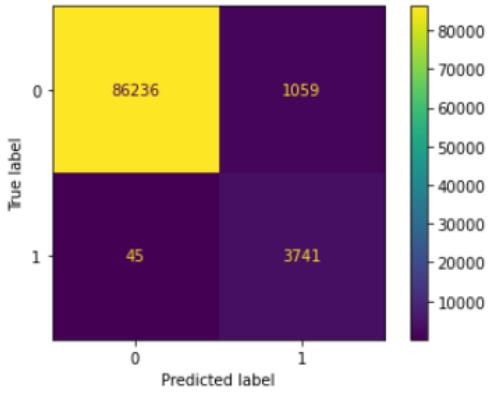
(a) Matriz de Confusão utilizando Regressão Logística



(b) Matriz de Confusão utilizando *Random Forest*



(c) Matriz de confusão utilizando Naïve Bayes

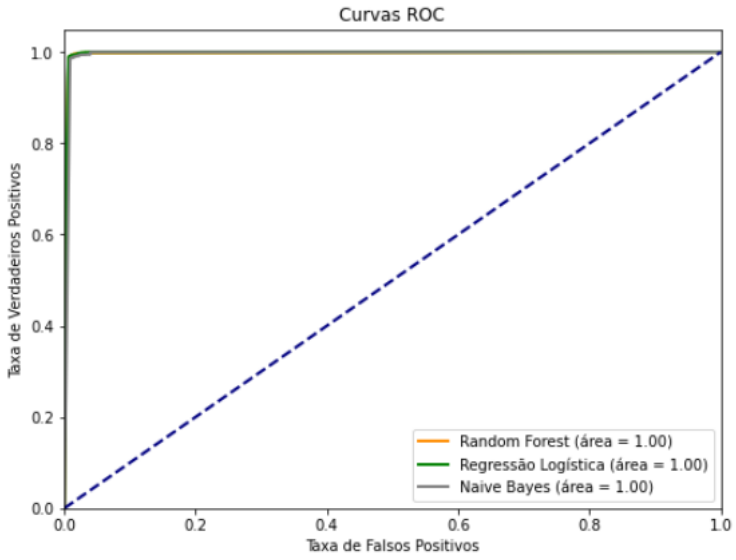


Fonte: Próprio autor

Nos gráficos apresentados acima, o eixo de *true label* representa os rótulos verdadeiros, enquanto o eixo de *predicted label* representa os rótulos preditos. Os valores que se encontram no quadrante 0 de ambos os rótulos representam os verdadeiros positivos, assim como os valores que se encontram no quadrante 1 de ambos os rótulos representam os verdadeiros negativos. Entre os 3 modelos estudados para o experimento 1, o que melhor performou de acordo com a matriz de confusão, foi o modelo de Regressão Logística. Com uma acurácia de 99,31% e um valor de F-Score de 92,20%, esse modelo obteve resultados melhores dos que os obtidos por *Random Forest* (Acurácia – 99,28%/ F-Score - 91,84%) e Naïve Bayes (Acurácia – 98,79%/ F-Score – 87,14%). Os valores de falsos negativos se mantiveram baixos nos 3 modelos, mantendo a revocação alta. Isso significa que, das amostras positivas existentes, os modelos conseguiram classificar corretamente grande parte das predições realizadas. O modelo de Naïve Bayes apresentou o melhor resultado de revocação, com 98,81%, ligeiramente

superior à Regressão Logística e *Random Forest*, que ficaram com 98,63% e 97,54%, respectivamente. Já que as taxas de verdadeiros positivos foram elevadas para os 3 modelos. A curva ROC apresentada na figura abaixo, se mostrou muito perto da perfeição, com as 3 linhas (cada uma representando um modelo) bem próximas.

Figura 28 – Curva ROC



Fonte: Próprio Autor

5.1.2 Experimento 2

A sequência de desenvolvimento do experimento 2 segue quase o mesmo formato do experimento 1, com o mesmo pré-processamento e *test split* (80% treinamento / 20% validação), mas com a diferença na seleção das variáveis, utilizando dessa vez apenas os 10 atributos mais relevantes. Essa seleção é feita com base nas maiores pontuações de KNN, utilizando os 5 vizinhos mais próximos e a distância Euclidiana para a mensuração da separação dos pontos. As variáveis que apresentaram os maiores *Scores* de KNN estão presentes na figura abaixo.

Figura 29 – Top 10 variáveis mais relevantes Experimento 2

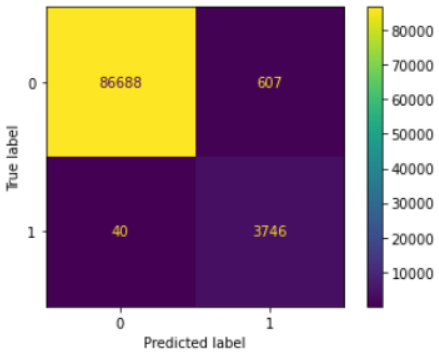
	Feature	Score
1	checked_delivery_detail	644609.494400
2	saw_checkout	370670.445728
3	sign_in	291524.271374
4	basket_icon_click	82038.292181
5	basket_add_detail	75667.880477
6	basket_add_list	32856.707438
7	saw_homepage	9236.523129
8	list_size_dropdown	9017.488357
9	closed_minibasket_click	7498.602916
10	image_picker	1857.119342

Fonte: Próprio Autor

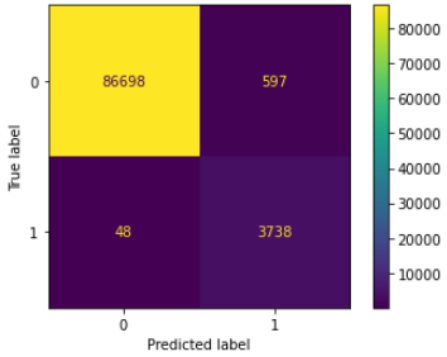
Após a seleção das variáveis apresentadas na figura acima, as bases de treinamento e teste foram reduzidas para apenas as 10 colunas correspondentes e foram salvas nas variáveis X_treinamento_knn e X_teste_knn. O passo seguinte foi o *fit* e predição dos 3 modelos de classificação, utilizando a base de X_treinamento_knn e y_treinamento. Já a predição foi realizada através da base X_teste_knn. Os modelos de Regressão Logística, *Random Forest* e Naïve Bayes obtiveram resultados muito próximos de acurácia, F-Score e AUC.

Figura 30 – Matrizes de Confusão Experimento 2

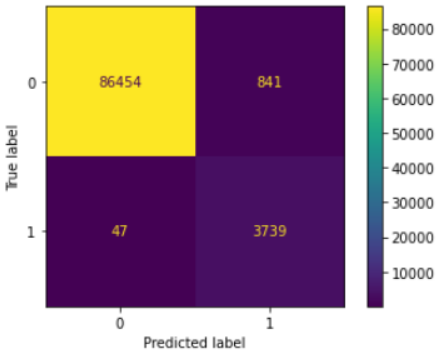
(a) Matriz de confusão utilizando Regressão Logística



(b) Matriz de confusão utilizando *Random Forest*



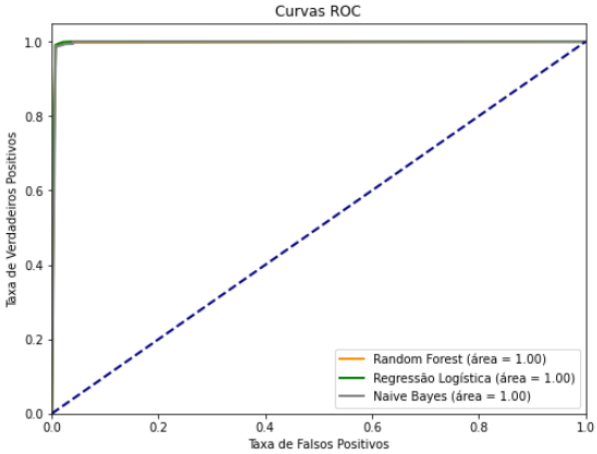
(c) Matriz de confusão utilizando Naïve Bayes



Fonte: Próprio Autor

A Regressão Logística apresentou valores de acurácia de 99,29%, F-Score de 99,31% e AUC de 99,12. Já *Random Forest* obteve acurácia de 99,29%, F-Score de 99,31% e AUC de 99,02%. Por fim, Naïve Bayes ficou com acurácia de 99,02%, F-Score de 99,07% e AUC de 98,90%. O indicador mais destacado de Regressão Logística foi a revocação, que obteve um valor de 98,94%, enquanto *Random Forest* obteve 98,73% e Naïve Bayes 98,76%. Em linhas gerais, os resultados apresentados pelos 3 modelos foram ligeiramente melhores do que no experimento 1, quando todas as variáveis foram utilizadas. Isso mostra que a seleção das variáveis mais relevantes melhorou o desempenho de predição dos modelos, fazendo com que os resultados fossem mais assertivos.

Figura 31 – Curva ROC Experimento 2



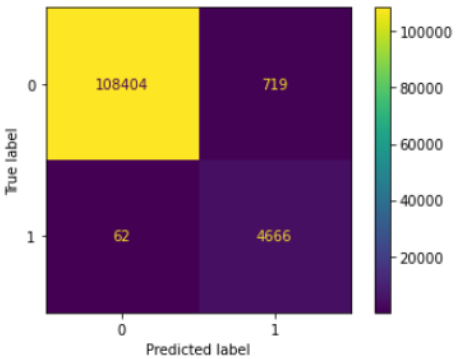
Fonte: Próprio Autor

5.1.3 Experimento 3

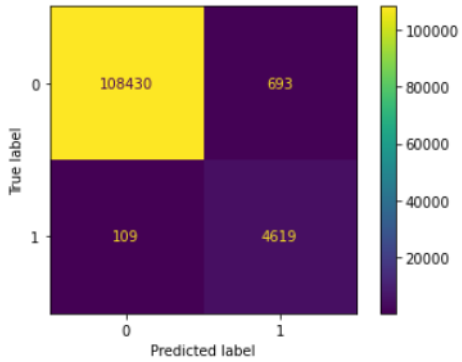
No experimento 3 a base foi dividida em 75% treinamento e 25% validação. Os demais procedimentos foram idênticos aos adotados pelo experimento 1, onde todas as 23 variáveis independentes da base de treinamento mais a variável y_treinamento foram utilizadas para o *fit* do modelo. Já a predição utilizou a variável X_teste. Para essa divisão de *test split*, 341.550 clientes foram alocados para as variáveis X_treinamento e y_treinamento, enquanto 113.851 foram designados para as variáveis X_teste e y_teste.

Figura 32 – Matrizes de Confusão Experimento 3

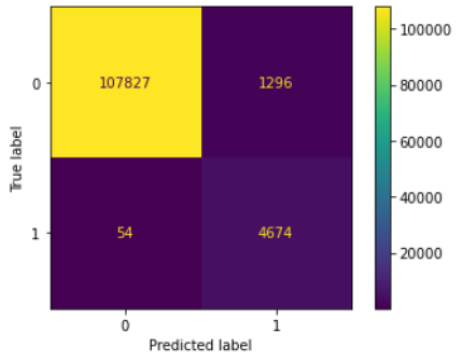
(a) Matriz de confusão utilizando Regressão Logística



(b) Matriz de confusão utilizando *Random Forest*



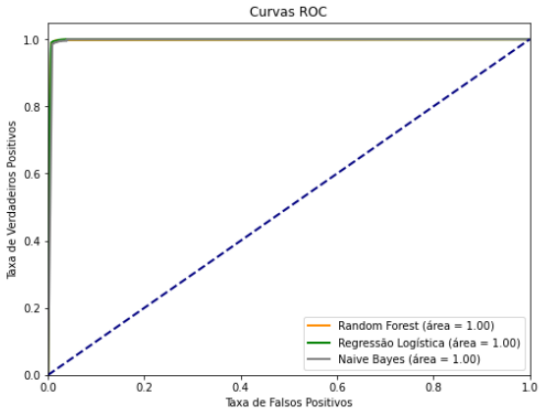
(c) Matriz de confusão utilizando Naïve Bayes



Fonte: Próprio Autor

Assim como visto nos experimentos 1 e 2, os resultados dos 3 modelos de classificação ficaram bem próximos. Regressão Logística teve acurácia de 99,31%, F-Score de 92,28% e AUC de 99,01%. *Random Forest* obteve acurácia de 99,30%, F-Score de 92,01% e AUC de 98,53%. Por fim, Naïve Bayes com acurácia de 98,81%, F-Score de 87,38% e AUC de 98,83%.

Figura 33 – Curva ROC Experimento 3



Fonte: Próprio Autor

5.1.4 Experimento 4

No experimento 4, foi adotado o *test split* de 75% treinamento e 25% validação, assim como no experimento 3. Porém, nesse experimento as variáveis X_treinamento e X_teste foram reduzidas para apenas 10 colunas, com base nas variáveis que obtiveram maiores *Scores* de KNN. A lista dessas variáveis mais influentes seguiu a mesma ordem da figura 29, quando o mesmo procedimento foi realizado para o experimento 2. Entretanto, os *Scores* apresentaram ligeira queda em relação ao *test split* de 80% treinamento e 20% validação, apresentados no experimento 2.

Figura 34 - Top 10 variáveis mais relevantes Experimento 4

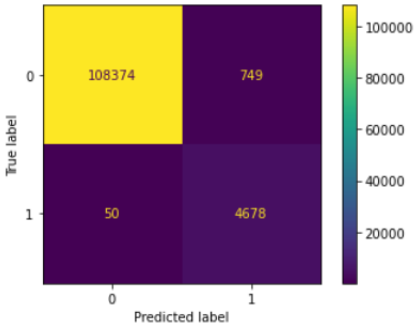
	Feature	Score
1	checked_delivery_detail	605128.349169
2	saw_checkout	347367.170881
3	sign_in	273179.584634
4	basket_icon_click	77245.046473
5	basket_add_detail	70740.932670
6	basket_add_list	31010.617237
7	saw_homepage	8725.601965
8	list_size_dropdown	8501.463262
9	closed_minibasket_click	7089.406832
10	image_picker	1739.394698

Fonte: Próprio Autor

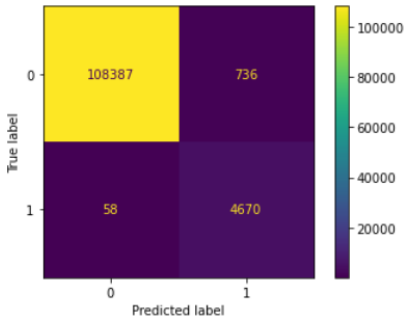
Após a seleção das 10 melhores variáveis, os grupos de X_treinamento_knn e X_teste_knn foram criados, onde apenas as 10 colunas mais influentes foram armazenadas. Na sequência desse procedimento, os 3 modelos de classificação foram implementados. Os resultados das matrizes de confusão para o experimento 4 foram:

Figura 35 – Matrizes de Confusão Experimento 4

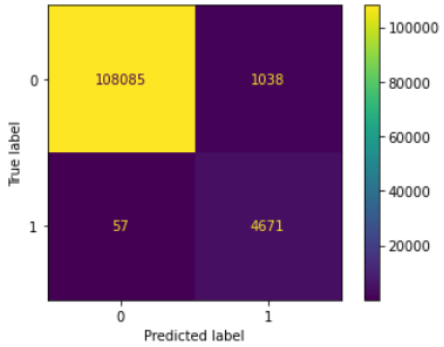
(a) Matriz de confusão utilizando Regressão Logística



(b) Matriz de confusão utilizando *Random Forest*



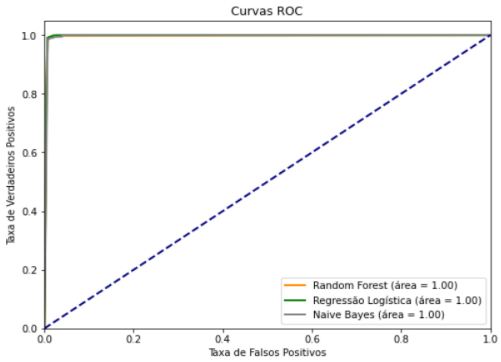
(c) Matriz de confusão utilizando Naïve Bayes



Fonte: Próprio Autor

Seguindo o comportamento dos resultados vistos nos experimentos 1,2 e 3, os indicadores de acurácia, F-Score e AUC ficaram bem próximos para os modelos de Regressão Logística, *Random Forest* e Naïve Bayes. Regressão Logística obteve uma acurácia de 99,30%, F-Score de 92,13% e AUC Score de 99,13%. *Random Forest* mostrou uma acurácia de 99,30%, F-Score de 92,16% e AUC de 99,05%. Por fim, Naïve Bayes apresentou acurácia de 99,04%, F-Score de 89,51% e AUC Score de 98,92%.

Figura 36 – Curva ROC Experimento 4



Fonte: Próprio Autor

Consolidando todos os resultados apresentados pelos 4 experimentos em uma tabela, os números de cada indicador e modelo ficaram da seguinte forma:

Tabela 7– Tabela com resultados de Validação

Experimento	***Test split***	**Com KNN / Sem KNN**	**Métricas de Avaliação**	**Regressão Logística**	***Random Forest***	**Naïve Bayes**
Experimento 1	80%/20%	Sem KNN	Acurácia	99,31%	99,28%	98,79%
Experimento 1	80%/20%	Sem KNN	AUC	98,98%	98,44%	98,80%
Experimento 1	80%/20%	Sem KNN	F-Score	92,20%	91,84%	87,14%
Experimento 2	80%/20%	Com KNN	Acurácia	99,29%	99,29%	99,02%
Experimento 2	80%/20%	Com KNN	AUC	99,12%	99,02%	98,90%
Experimento 2	80%/20%	Com KNN	F-Score	99,31%	99,31%	99,07%
Experimento 3	75%/25%	Sem KNN	Acurácia	99,31 %	99,30%	98,81%
Experimento 3	75%/25%	Sem KNN	AUC	99,01%	98,53%	98,83%
Experimento 3	75%/25%	Sem KNN	F-Score	92,28%	92,01%	87,38%
Experimento 4	75%/25%	Com KNN	Acurácia	99,30%	99,30%	99,04%
Experimento 4	75%/25%	Com KNN	AUC	99,13%	99,05%	98,92%
Experimento 4	75%/25%	Com KNN	F-Score	92,13%	92,16%	89,51%

Fonte: Próprio Autor

6 CONCLUSÃO E TRABALHOS FUTUROS

6.1 Conclusão

Como apresentado na tabela acima, os resultados foram muito similares, tanto no quesito divisão de treinamento e validação, como na utilização ou não do KNN para a seleção de variáveis. Contudo, é possível notar um ligeira melhora dos 3 indicadores (Acurácia, AUC e F-Score) nos experimentos que envolveram a seleção das melhores variáveis. O experimento 2, por exemplo, obteve resultados melhores de acurácia no modelo de Naïve Bayes, de AUC para os 3 modelos e de F-Score para *Random Forest* e Naïve Bayes, em relação aos mesmos cenários do experimento 1. Algo muito parecido também ocorreu quando os experimentos 3 e 4 foram comparados, com o experimento 4 performando ligeiramente melhor que o experimento 3. Entre os experimentos que utilizaram o KNN para a seleção das variáveis, experimentos 2 e 4, o experimento 2 apresentou os melhores resultados de F-Score nos 3 algoritmos analisados. Como os demais indicadores ficaram muito próximos nos experimentos 2 e 4, o fato do F-score do experimento 2 ser o mais alto entre todos os experimentos, fez com que ele fosse escolhido como melhor experimento entre os 4 apresentados.

Na questão de eficiência de modelo, o que apresentou constantemente o melhor desempenho entre os diferentes subgrupos foi o modelo de Regressão Logística, com altos valores de AUC, F-Score e acurácia. Utilizando esse modelo para fazer as predições, é possível gerar a lista final de usuários com maior chance de comprar algo da loja fictícia de e-commerce. Utilizando a base de teste disponibilizada pelo site Kaggle, que permaneceu intacta durante todos os experimentos, foi possível aplicar as predições de probabilidade do modelo de Regressão Logística através da função *predict_proba*. Essa função retorna uma probabilidade de 0 a 100, baseado no modelo selecionado e treinado. Como o experimento 2 apresentou os melhores resultados entre os 4 experimentos analisados, a base de validação foi reduzida também para 10 colunas, assim como foi feito durante o experimento 2.

Dentre os 151.655 clientes presentes na base de teste do Kaggle, 1.048 apresentaram uma propensão maior que 80% de compra, enquanto 182 apresentaram uma propensão acima de 90% de compra. Dessa forma, atribuindo uma probabilidade de compra a cada usuário, possibilita que uma empresa, por exemplo, tenha a flexibilidade de limitar o percentual de propensão de acordo com a sua capacidade operacional e financeira para uma determinada campanha de comunicação. Se a empresa possuir alta capacidade operacional e alto *budget* para

campanhas de marketing, ela pode selecionar até as propensões de 70% ou mais, já que a quantidade de clientes será mais abrangente.

Figura 37 – Quantidade de clientes por Intervalo de Propensão (%)

	Intervalo de Propensão (%)	Quantidade de Clientes
0	0-10	150473
1	10-20	0
2	20-30	0
3	30-40	0
4	40-50	0
5	50-60	0
6	60-70	0
7	70-80	134
8	80-90	866
9	90-100	182

Fonte: Próprio Autor

A figura acima demonstra exatamente o resultado esperado pelas predições de um modelo de Regressão Logística. Como esse modelo tem previsões binárias, 0 ou 1, os valores de predição acompanham esse comportamento. Os clientes estão concentrados nos dois extremos, ou perto de zero, dentro do intervalo de propensão entre 0-10%, ou no espectro oposto, perto de uma propensão de 100%. Isso faz com que não exista nenhum valor dentro dos intervalos intermediários, entre 10-60%.

Sendo assim, a lista de clientes mais propensos a compra está definida para esse e-commerce fictício. Essa lista possui uma abrangência de até 1.182 clientes altamente propensos a compra, com uma acurácia muito elevada. A empresa fictícia de e-commerce poderia utilizar para companhas de e-mail marketing ou até mesmo campanhas nas redes sociais, a lista desses 1.182 clientes mais propensos, ao invés de utilizar a base inteira de 151.655. Esse movimento traria uma grande redução de custos com campanha e uma melhora significativa no retorno sobre o investimento, já que a chance de conversão desses 1.182 clientes é muito elevada.

6.2 Trabalhos Futuros

Apesar dos resultados interessantes e promissores apresentados pelos modelos de Regressão Logística, *Random Forest* e Naïve Bayes, vale a pena ressaltar as dificuldades e desafios encontrados durante a realização deste trabalho, incentivando assim a produção de trabalhos futuros sobre o tema.

Um desses desafios foi a seleção e implementação das melhores variáveis para a realização do modelo. Como as bases disponibilizadas pelo site Kaggle são relativamente pequenas em comparação com bases do mundo real e de empresas de grande porte, a seleção de variáveis através do KNN não se mostrou tão eficaz como poderia ter sido caso as bases fossem de maior volume. Como visto durante a realização deste estudo, os resultados apresentados pelos experimentos com KNN foram muito similares aos apresentados pelos experimentos sem KNN. Logo, caso um estudo futuro fosse realizado utilizando também a seleção de variáveis para uma base de dados maior, com milhões de linhas e centenas de atributos, muito provavelmente a seleção de variáveis apresente resultados significativamente melhores do que a mesma análise com todas as variáveis de uma vez. Além disso, o *deploy* do modelo de propensão para produção e o acompanhamento dos resultados reais, analisando na prática se os clientes sugeridos pelo modelo de fato converteram mais, é também uma abordagem relevante para estudos futuros.

Outro desafio e possível melhoria para futuros estudos seria a utilização de *fine-tunning.* Ajustando os hiperparâmetros de cada modelo e testando quais valores podem trazer melhores resultados de acurácia, AUC e F-Score, pode significar resultados finais ainda melhores do que os apresentados durante esse trabalho. Hiperparametros como C (alpha) e *max_iter* (máximo número de interações) para Regressão Logística, *n_estimators* (número de árvores) e *max_depth* (profundidade máxima das árvores na floresta) para *Random Forest* e por fim alpha (suavização) e distribuição de probabilidade para Naïve Bayes, são exemplos de hiperparametros ajustáveis que podem ser testados com diferentes valores e que podem trazer melhores resultados para os modelos selecionados em trabalhos futuros.

REFERÊNCIAS

[1] SZCZERBA, Robert. 20 Great Technology Quotes To Inspire, Amaze, And Amuse. **Forbes**, 09 de fevereiro de 2015 – Citação de Arthur Schlesinger. Disponível em: <https://www.forbes.com/sites/robertszczerba/2015/02/09/20-great-technology-quotes-to-inspire-amaze-and-amuse/>. Acesso em: 16 de setembro de 2023.

[2] **UNIVERSIDADE FEDERAL DO OESTE DO PARÁ (UFOPA)**. Disponível em: www.ufopa.edu.br/. Acesso em: 19 de setembro de 2023.

[3] ALBERTIN, A. L. Comércio Eletrônico: Modelo, Aspectos e Contribuições de sua Aplicação. 2 ed. São Paulo: **Atlas**, 2000a. Acesso em: 5 de maio de 2023.

[4] KOTLER, P. Marketing Management. 15. Ed. **Pearson**, 2015. Acesso em: 8 de maio de 2023.

[5] HULL, P. Tools for Entrepreneurs to Retain Clients. **Forbes**, 06 de dezembro de 2013. Disponível em: https://www.forbes.com/sites/patrickhull/2013/12/06/tools-for-entrepreneurs-to-retain-clients/?sh=15a0f8992443. Acesso em: 5 de março de 2023.

[6] **IPEA - Instituto de Pesquisa Econômica Aplicada**. Nota de Conjuntura nº 16: Economia Mundial. 2022. Disponível em: https://www.ipea.gov.br/cartadeconjuntura/wp-content/uploads/2022/11/221123_nota16_economia_mundial.pdf. Acesso em: 8 de março de 2023.

[7] **IBM**. Machine Learning. Disponível em: https://www.ibm.com/br-pt/topics/machine-learning. Acesso em: 12 de março de 2023.

[8] PAPPA, G.L. Introdução a Algoritmos de Computação Natural para Mineração de Dados. **Universidade Federal de Minas Gerais (UFMG).** Sem Data. Apresentação do Power Point. Disponível em: https://homepages.dcc.ufmg.br/~glpappa/slides/Curso-Parte1.pdf. Acesso em: 20 de setembro de 2023.

[9] FARIA, E. Medidas de Distância.pdf. **Universidade Federal de Uberlândia (UFU)**. 2018. Apresentação do Power Point. Disponível em: https://www.facom.ufu.br/~elaine/disc/MFCD2018/Aula7-MedidasDistancia.pdf. Acesso em: 20 de setembro de 2023.

[10] GUIMARÃES, A.M. Estatística: análise de correlação usando Python e R. **Medium**, 03 de fevereiro de 2021. Disponível em: https://medium.com/omixdata/estat%C3%ADstica-an%C3%A1lise-de-correla%C3%A7%C3%A3o-usando-python-e-r-d68611511b5a. Acesso em 6 de março de 2023.

[11] OLIVEIRA, T., A., S. Comparação de Algoritmos de Aprendizado de Máquina na Classificação de Neoplasias Mamárias. **Universidade Federal de Uberlândia - UFU**, MG, Brasil, 2021. Disponível em: https://repositorio.ufu.br/handle/123456789/32251. Acesso em: 18 de março de 2023.

[12] MARIANO, D., C., B. Métricas de avaliação em machine learning. **Revista Brasileira de Bioinformática,** junho de 2021. Disponível em: https://bioinfo.com.br/metricas-de-avaliacao-em-machine-learning-acuracia-sensibilidade-precisao-especificidade-e-F-Score/. Acesso em: 18 de março de 2023.

[13] **IBM**. K-nearest neighbors (KNN). Disponível em: https://www.ibm.com/topics/knn. Acesso em: 6 de março de 2023.

[14] **AMAZON WEB SERVICES**, **INC**.O que é regressão logística? Disponível em: https://aws.amazon.com/pt/what-is/logistic-regression/. Acesso em: 20 de março de 2023.

[15] SARITAS, M., M., YASAR, A. Performance Analysis of ANN and Naive Bayes Classification Algorithm for Data Classification. **International Journal of Intelligent Systems and Applications in Engineering**, ISSN:2147-6799, IJISAE, 2019, 7(2), 88–91, maio de 2019. Disponível em: https://ijisae.org/IJISAE/article/view/934/585. Acesso em: 22 de março de 2023.

[16] GÓMEZ, S., N. *Random Forest*s Estocástico. **Pontifícia Universidade Católica do Rio Grande do Sul**, Faculdade de Informática, Programa de Pós-Graduação em Ciência da Computação Porto Alegre, RS, Brasil, agosto de 2012. Disponível em: http://tede2.pucrs.br/tede2/handle/tede/5226. Acesso em: 29 de março de 2023.

[17] **BUILTIN**, INC.Train-*Test split* in Machine Learning: What It Is and How to Use It. Disponível em: https://builtin.com/data-science/train-test-split. Acesso em: 29 de março de 2023.

[18] RODRIGUES.V.B. Métricas de Avaliação: Acurácia, Precisão, Recall - Quais as Diferenças? **Medium**. Disponível em: https://vitorborbarodrigues.medium.com/m%C3%A9tricas-de-avalia%C3%A7%C3%A3o-acur%C3%A1cia-precis%C3%A3o-recall-quais-as-diferen%C3%A7as-c8f05e0a513c. Acesso em: 6 de março de 2023.

[19] EDUCATIVE, INC. Difference between predict () and predict_proba() in scikit-learn. Disponível em: https://www.educative.io/answers/difference-between-predict-and-predictproba-in-*sklearn*. Acesso em: 23 de março de 2023.

[20] Rodrigues, V. Entenda o que é AUC e ROC nos modelos de Machine Learning. **Medium**. Disponível em: https://medium.com/bio-data-blog/entenda-o-que-%C3%A9-auc-e-roc-nos-modelos-de-machine-learning-8191fb4df772. Acesso em: 31 de março de 2023.

[21] CHANDRAHASDHIRAJ. Customer propensity to Buy. *In:* **Kaggle**. [S.I],1 de junho de 2018. Disponível em: https://www.kaggle.com/code/chandrahasdhiraj/customer-propensity-to-buy. Acesso em: 18 de março de 2023.

[22] PRAKASH.N.J. Customer propensity to Buy. *In:* **Kaggle**. [S.I],1 de junho de 2018. Disponível em: https://www.kaggle.com/code/nikhiljothiprakash/propensity-using-logistic-regression. Acesso em: 18 de março de 2023.

[23] **SCIKIT-LEARN**. StandardScaler - scikit-learn 0.24.2 documentation. Disponível em: https://scikit-learn.org/stable/modules/generated/*sklearn*.preprocessing.StandardScaler.html. Acesso em: 24 de setembro de 2023.

[24] KAI-HONG, X. An Empirical Study of Custumer`s Online Initial Trust Production Model Based on Consumer`s Perception. **Henan University of Technology**. 2008. Disponível em: https://ieeexplore.ieee.org/document/4680305. Acesso em: 8 de abril de 2023.

[25] ALMEIDA, C. Aplicação do pacote computacional spss em pesquisa de opinião utilizando escala de likert. 2016. 66f. Dissertação - Mestrado em Engenharia Mecânica do departamento de pós-graduação da **Universidade de Taubaté**, 2016. Acesso em 24 de setembro de 2023.

[26] **IBM**. Teste de KMO e Bartlett. Disponível em: https://www.ibm.com/docs/pt-br/spss-statistics/29.0.0?topic=detection-kmo-bartletts-test. Acesso em: 24 de setembro de 2023.

[27] RODRÍGUEZ, E.J; CANO, V.E; HERNANDEZ, A.C; MENESES, E.L. Influence of Computer Knowledge and Level of Education on Spanish Citizens' Propensity to Use E-Commerce. Vol 40. **Social Science Computer Review**. 2022. Disponível em: https://dl.acm.org/doi/abs/10.1177/08944393211007313. Acesso em: 10 de abril de 2023.

[28] GUPTA,S; JOSHI,S. Predictive Analytic Techniques for enhancing marketing performance and Personalized Customer Experience. 2022. Disponível em: https://ieeexplore.ieee.org/abstract/document/10060286. Acesso em: 08 de abril de 2023.

[29] SÚNIGA, A. Conjuntos de Treino, Teste e Validação em Machine Learning (Fast.ai). **Medium**. Disponível em: https://medium.com/@abnersuniga7/conjuntos-de-treino-teste-e-valida%C3%A7%C3%A3o-em-machine-learning-fast-ai-5da612dcb0ed. Acesso em: 25 de setembro de 2023.

Apêndice A – Pasta Gitlab Com Scripts e Base Final

Com o intuito de prover todos os arquivos utilizados durante a pesquisa desse estudo, foi criado um projeto novo no Gitlab chamado Modelo de Propensão, onde os seguintes arquivos podem ser encontrados:

- *Training_sample* - Base original de treinamento providenciada pelo site Kaggle em formato csv.
- *Test_sample* - Base original de teste providenciada pelo site Kaggle em formato csv.
- *Modelo_De_Propensao_-_Experimentos_1_e_2_-_80_20_*- Script Python com todo o desenvolvimento dos experimentos 1 e 2.
- *Modelo_De_Propensao_-_Experimentos_3_e_4_-__75_25* - Script Python com todo o desenvolvimento dos experimentos 3 e 4 e aplicação da predição.
- *Clientes_Mais_Propensos* - Base excel com os clientes mais propensos.

https://gitlab.com/machine-learning-and-artificial-intelligence/modelo-de-propensao/-/tree/main