Análise Preditiva sobre Conversão de vendas no Varejo Online

UC: Inteligência Artificial

Everton Martins Simões Fernandes – RA 323132077

Introdução:

O objetivo deste trabalho é realizar uma análise preditiva para prever padrões de conversão de vendas, considerando sazonalidades, promoções e outros fatores que influenciam o comportamento do consumidor. Para isso, buscamos desenvolver e avaliar modelos de aprendizado de máquina capazes de prever a conversão de vendas de forma descentralizada, utilizando dados históricos fornecidos pelo **Retail Rocket Recommender Dataset**(https://www.kaggle.com/datasets/retailrocket/ecommerce-dataset).

Basicamente o objetivo do nosso projeto é explorar diferentes abordagens de aprendizado de máquina para prever a conversão de vendas em um cenário de varejo online. Os modelos avaliados incluem Regressão Linear e Random Forest, com foco em analisar:

1. Precisão das predições;
2. Comparação entre modelos;
3. Interpretação visual dos resultados, como matriz de confusão, gráficos de custo e importância de variáveis;

Dataset:

Como dito anteriormente o dataset utilizado para este projeto é o **Retail Rocket Recommender Dataset**, disponível no Kaggle, ele contém informações sobre interações de usuários com produtos e suas respectivas conversões.

Este conjunto de dados foi escolhido pela sua capacidade de representar o comportamento dos consumidores e informações valiosas para prever a conversão de vendas.

Link para o Dataset: Retail Rocket Recommender Dataset

Visão do DataFrame:

	timestamp	visitorid	event	itemid	transactionid
0	1433221332117	257597	view	355908	NaN
1	1433224214164	992329	view	248676	NaN
2	1433221999827	111016	view	318965	NaN
3	1433221955914	483717	view	253185	NaN
4	1433221337106	951259	view	367447	NaN
...	...	...	...	...	...
2756096	1438398785939	591435	view	261427	NaN
2756097	1438399813142	762376	view	115946	NaN
2756098	1438397820527	1251746	view	78144	NaN
2756099	1438398530703	1184451	view	283392	NaN
2756100	1438400163914	199536	view	152913	NaN

2756101 rows × 5 columns

Descrição de Colunas:

1. **timestamp**: Representa a data e hora da interação do usuário com o produto. Essa coluna é fundamental para analisar padrões temporais, como sazonalidades e picos de demanda.
2. **visitorid**: Identificador único do visitante. Permite mapear as interações de um mesmo usuário com diferentes produtos ao longo do tempo.

3. **event**: Tipo de evento realizado pelo usuário. Pode ser uma visualização de produto ("view"), clique ("click"), adição ao carrinho ("add-to-cart") ou compra ("purchase"). A análise deste campo nos permite entender o comportamento do consumidor em cada etapa da jornada de compra.

4. **itemid**: Identificador único do produto. Usado para associar os eventos aos produtos específicos que os usuários estão interagindo.
5. **transactionid**: Identificador único da transação, caso o evento seja uma compra. Quando o evento não resulta em uma compra, o valor é **NaN**. Esta coluna é crucial para verificar se o evento levou ou não à conversão (compra).

Pré-Processamento de Dados:

O pré-processamento dos dados constituiu uma fase crítica para assegurar que o modelo aprenderia de forma eficiente a prever a conversão das vendas. As ações principais realizadas foram a criação de novas variáveis, a filtragem e a transformação de dados. Detalho a seguir os passos que foram adotados:

1. **Criação da Coluna de Conversão**: Para realizar a análise criamos uma nova coluna partir da **coluna transactionid**, que indica se houve ou não uma transação. A logina utilizado foi:

    a. Se a **coluna transactionid** não for **NaN** (isto é, a transação ocorreu), a conversão foi realizada, e o valor de **converted** foi marcado como **1**.
    b. aso contrário, o valor de **converted** foi **0**, indicando que não houve compra.

2. **Filtragem e Agregação dos Dados para Treinamento**: : Após a criação da variável de conversão, os dados foram agrupados por **visitorid**, identificando os visitantes únicos. Para cada visitante, foram agregadas as seguintes informações:
    a. **Views**: Número total de visualizações realizadas;
    b. **First_view**: Hora da primeira visualização;
    c. **Last_view**: Hora da última visualização;
    d. **Converted**: Se houve conversão de vendas (considerando o valor máximo de converted, pois qualquer valor 1 indica que houve conversão).

    OBS: utilizamos a função groupby para agrupar as informações por visitante.

3. **Cálculo da Diferença de Tempo entre Primeira e Última Visualização**: A fim de entender melhor o comportamento dos visitantes, calculamos a diferença entre o tempo da primeira visualização e o tempo da última visualização,

obtendo assim uma medida do tempo, em segundos, que o usuário esteve navegando no site, caso tenha tomado a decisão de converter. Por tanto foi criado a coluna **view_time_diff** para análise.

Modelos de Aprendizado de Máquina Aplicados

Para prever a conversão de vendas no varejo online, foram utilizados dois modelos de aprendizado de máquina: **Regressão Linear** e **Random Forest**. A escolha desses modelos se deu pela sua robustez e aplicação em problemas de classificação, como a previsão de conversões, onde buscamos entender as relações entre variáveis de entrada (**views**, **view_time_diff**, etc...) e a variável alvo (**converted**).

Justificativa para a Escolha

- **Regressão Linear**: A regressão linear foi escolhida como uma base de comparação inicial. Ela pode ser útil para prever a probabilidade de conversão, tratando-a como uma variável contínua. Embora a saída de regressão linear não seja diretamente binária, podemos adaptá-la arredondando as previsões para 0 ou 1.
  Um modelo simples e fácil de entender. Apesar de ser uma escolha inicial, ele serve como benchmark para compararmos os resultados com modelos mais complexos.

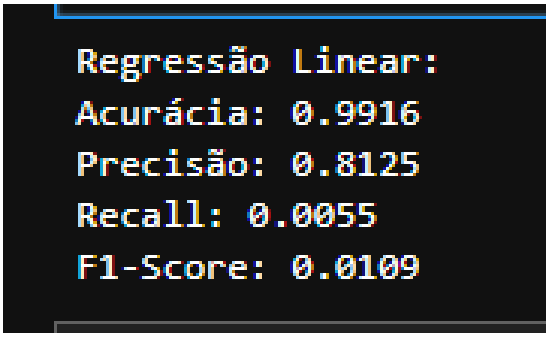
- **Random Forest**: O Random Forest foi escolhido por ser um modelo robusto para problemas de classificação binária. Ele lida bem com dados complexos e pode capturar relações não lineares entre as variáveis de entrada e a conversão. Além disso, ele pode fornecer uma avaliação precisa da importância das variáveis de entrada no modelo.

É um modelo poderoso para problemas de classificação binária e é capaz de capturar interações complexas e não lineares entre as variáveis, o que pode melhorar a precisão do modelo, especialmente em um problema como este, onde os padrões de comportamento do usuário podem ser não lineares.

Análise dos Resultados

Após o treinamento de ambos os modelos, foi possível avaliar o desempenho de cada um com base nas métricas de precisão, recall, F1-score e a matriz de confusão. A seguir, são apresentadas as observações detalhadas para cada modelo. Abaixo segue os valores observados:

Resultados - Regressão Linear



Apesar de sua alta acurácia, este fato pode ser explicado pela existência de um desbalanceamento entre classes no conjunto de dados, em que a maior parte dos casos pertence à classe "não convertido". A taxa de recall baixa (0,55%) do modelo indica que este apresenta dificuldades em classificar casos verdadeiros de conversão (classe positiva), o qual se revela uma performance restrita em termos de previsão realística de conversão.

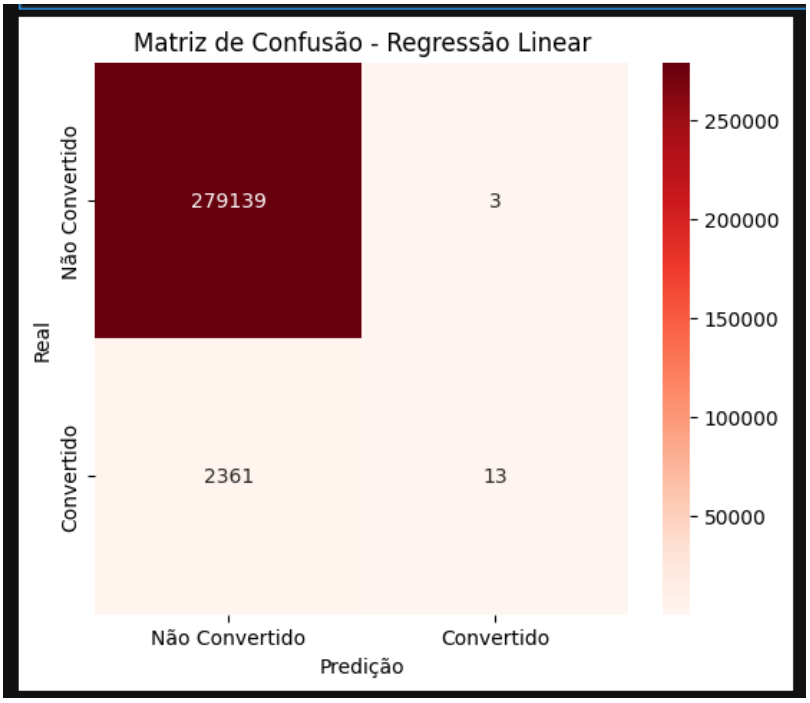
Resultados - Random Forest

```
Random Forest:
Acurácia: 0.9869
Precisão: 0.1220
Recall: 0.0885
F1-Score: 0.1025
```

O Random Forest me pareceu mais robusto ao lidar com relações complexas e não lineares entre as variáveis, alcançando um balanço mais adequado entre precisão e recall. No entanto, também foi impactado pelo desbalanceamento de classes, destacando a necessidade de técnicas adicionais, como oversampling (SMOTE) ou ajuste de pesos de classe, para melhorar a performance geral.

Matriz de Confusão:

Regressão Linear:



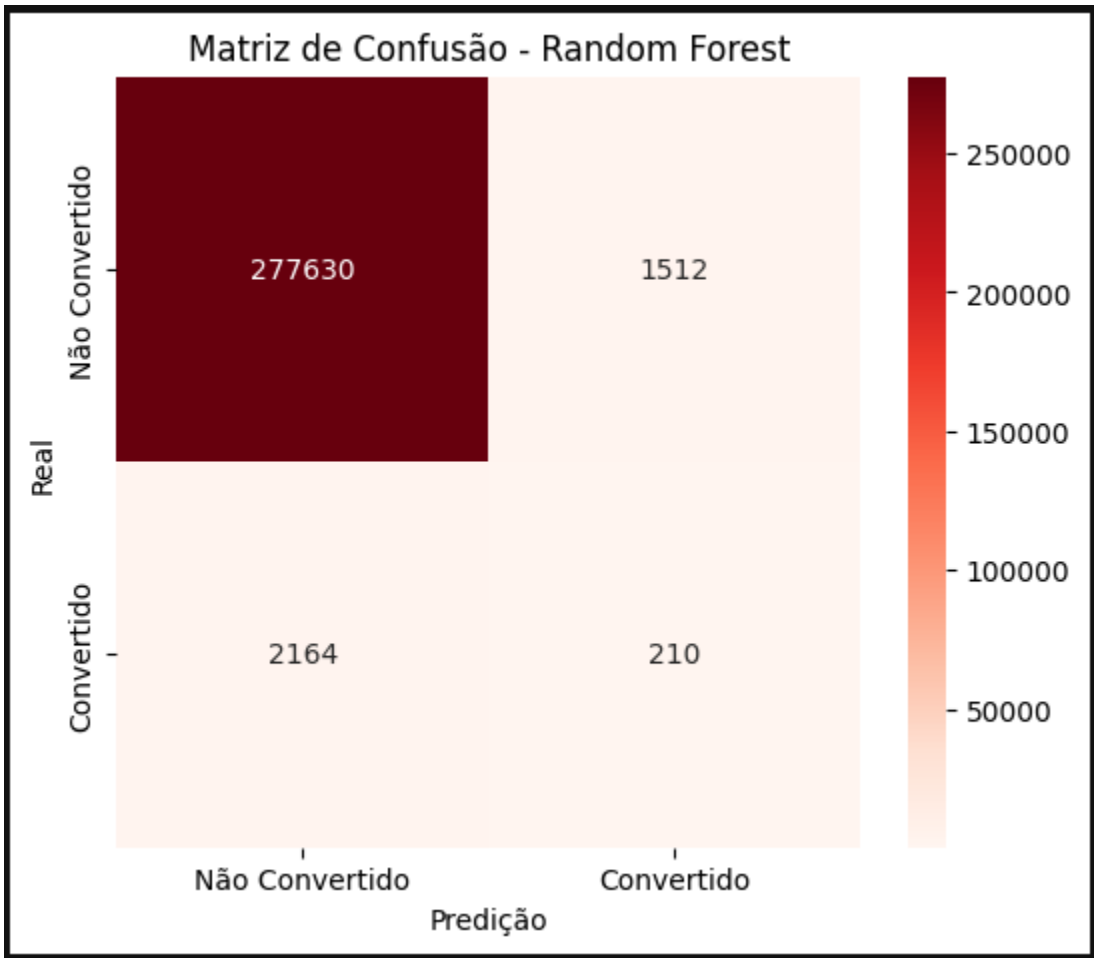
Considerações:

- **279139 (Verdadeiro Negativo - VN):** O modelo previu "Não Convertido" e a classe real era "Não Convertido". Este é o maior número, indicando que o modelo é bom em identificar casos "Não Convertidos"**.**
- **3 (Falso Positivo - FP):** O modelo previu "Convertido", mas a classe real era "Não Convertido". Este é um erro do tipo I (rejeitar a hipótese nula quando ela é verdadeira).
- **2361 (Falso Negativo - FN):** O modelo previu "Não Convertido", mas a classe real era "Convertido". Este é um erro do tipo II (aceitar a hipótese nula quando ela é falsa).
- **13 (Verdadeiro Positivo - VP):** O modelo previu "Convertido" e a classe real era "Convertido".

Análise da Matriz:

- **Desbalanceamento extremo:** Sua matriz mostra um desbalanceamento de classes severo. A classe "Não Convertido" tem muito mais exemplos do que a classe "Convertido". Isso pode levar a métricas de acurácia enganosas. Uma acurácia alta pode ser alcançada simplesmente prevendo "Não Convertido" para todos os casos.
- **Baixo desempenho na classe "Convertido":** O modelo quase nunca prevê "Convertido" corretamente. Isso sugere que a regressão linear pode não ser o melhor modelo para este problema, ou que os dados precisam de mais pré-processamento (ex: balanceamento de classes).
- **Métricas além da acurácia:** Devido ao desbalanceamento, você deve considerar outras métricas como precisão, recall, F1-score, e AUC para avaliar o modelo de forma mais completa.
- **Contexto é crucial:** A interpretação também depende do custo de cada tipo de erro. Em alguns casos, um falso negativo pode ser muito mais custoso do que um falso positivo (ex: diagnóstico médico). Você precisa considerar o contexto do seu problema para determinar se o desempenho do modelo é aceitável.

Random Forest:



Considerações:

- **277630 (Verdadeiro Negativo - VN):** O modelo previu "Não Convertido" e a classe real era "Não Convertido". Assim como na regressão linear, o modelo é bom em prever a classe majoritária.
- **1512 (Falso Positivo - FP):** O modelo previu "Convertido", mas a classe real era "Não Convertido". Este número é maior do que na regressão linear, indicando mais erros do tipo I.
- **2164 (Falso Negativo - FN):** O modelo previu "Não Convertido", mas a classe real era "Convertido". Este número é menor do que na regressão linear, o que é uma melhoria.
- **210 (Verdadeiro Positivo - VP):** O modelo previu "Convertido" e a classe real era "Convertido". Este número é significativamente maior do que na regressão

linear, mostrando uma melhora considerável na previsão da classe minoritária.

Análise da Matriz:

- **Desbalanceamento de classes:** A matriz demonstra um desbalanceamento significativo entre as classes "Não Convertido" e "Convertido", com a grande maioria dos exemplos pertencendo à classe "Não Convertido". Isso pode levar a uma acurácia enganosa, pois o modelo pode atingir alta acurácia simplesmente prevendo a classe majoritária na maioria das vezes.
- **Performance na classe "Convertido":** Embora o Random Forest geralmente lide melhor com classes desbalanceadas do que modelos mais simples, é importante avaliar especificamente seu desempenho na classe minoritária ("Convertido"). Observe os valores de Verdadeiros Positivos e Falsos Negativos para entender a capacidade do modelo de identificar corretamente os casos de conversão.
- **Métricas além da acurácia:** A acurácia, sozinha, não é suficiente para avaliar o desempenho em cenários com classes desbalanceadas. Métricas como Precisão, Recall, F1-score e AUC fornecem uma visão mais completa e precisa da performance do modelo, considerando sua capacidade de identificar corretamente tanto os casos "Convertidos" quanto os "Não Convertidos".
- **Custo dos erros e contexto da aplicação:** A interpretação da matriz de confusão deve levar em conta o custo associado a cada tipo de erro (falso positivo e falso negativo). O contexto da aplicação é crucial para determinar qual tipo de erro é mais prejudicial. Por exemplo, em campanhas de marketing, o custo de um falso positivo (classificar alguém como "Convertido" quando não é) pode ser diferente do custo de um falso negativo (perder uma oportunidade de conversão). Essa análise de custo deve guiar a escolha do modelo e a definição de seus parâmetros.

Conclusão:

A análise preditiva das conversões de vendas em varejo online, a partir da utilização da Regressão Linear e do Random Forest, indicou como o desbalanceamento das classes, com a preponderância dos casos "Não Convertido", influenciou a acurácia dos modelos. Apesar de ambos apresentarem alta acurácia, a Regressão Linear apresentou desempenho fraco para capturar as conversões, enquanto o Random Forest, mais robusto do ponto de vista da captura das relações não lineares, atendeu melhor a classe "Convertido". A avaliação feita pela matriz de confusão e por métricas como Precisão, Recall e F1-score demonstrou que uma análise mais detalhada é mais apropriada do que apenas a acurácia em si, por conta do contexto da aplicação e do custo de cada tipo de erro.

Para trabalhos futuros, sugere-se a integração de técnicas de balanceamento de dados (SMOTE, ajuste de pesos), a otimização dos hiperparâmetros e a inclusão de novas variáveis para melhorar a predição, especialmente com respeito a classe minoritária. Trabalho com modelos tais como Gradient Boosting ou Redes Neurais poderá também revelar interessante para captar padrões complexos e melhorar a performance na captura do cenário do varejo online com relação a classes desbalanceadas.

Link para o Repositório com o Repositório:

https://github.com/evertonmsimoes/PrevisaoConversaoVendas

Link par a pasta com o Código:

https://github.com/evertonmsimoes/PrevisaoConversaoVendas/tree/main/Notebooks

OBS: O notebook utilizado par a apresentação final é de nome AnaliseDeConversao.ipynb, os demais foram utilizando para testes e experimentos durante a elaboração do Relatório.