**Proposta:**

**Churn e seu Impacto**

- O termo "churn" refere-se à taxa de rotatividade de clientes, ou seja, o número de clientes que deixam uma empresa em determinado período. Em empresas de telecomunicações, o churn é um dos principais indicadores de saúde do negócio, pois a retenção de clientes é um fator chave para o crescimento e sustentabilidade.

- A capacidade de prever quais clientes estão propensos a deixar a empresa permite a implementação de estratégias preventivas, como a oferta de promoções ou melhorias nos serviços para evitar a perda desses clientes. O churn também reflete a satisfação do cliente e pode ser influenciado por fatores como qualidade do serviço, atendimento ao cliente e preço.

Por isso então decidimos fazer o estudo sobre quem está saindo e por quê utilizando a ciência de dados

**Dados**

Informações disponíveis no Dataset:

Clientes que saíram no último mês - a coluna é chamada Churn, esse é o valor a ser previsto no modelo

Serviços para os quais cada cliente se inscreveu:  telefone, várias linhas, internet, segurança online, backup online, proteção de dispositivo, suporte técnico, streaming de TV e filmes.

Informações da conta do cliente: Informações da conta do cliente: há quanto tempo eles são clientes (tenure), contrato, forma de pagamento, faturamento sem papel, cobranças mensais e cobranças totais.

Informações demográficas sobre clientes: sexo, faixa etária e se têm parceiros e dependentes

Etapa 1: Primeira visualização

**Análise exploratória de dados (EDA)**: visa descobrir padrões, identificar anomalias e testar hipóteses usando estatísticas descritivas e ferramentas de visualização. No contexto deste projeto, a EDA é essencial para entender como as variáveis independentes (serviços, informações de conta, características demográficas) se relacionam com a variável dependente (churn).

**Visualizações Utilizadas**: Durante a EDA, gráficos como histogramas, boxplots e countplots foram usados para visualizar a distribuição dos dados e as correlações entre os recursos. Essa etapa também ajuda a identificar a necessidade de tratamentos como remoção de valores ausentes e transformação de variáveis categóricas.

**Os insights adquiridos podem ser visualizados na divisão de feature selection do notebook jupiter**

Etapa 2: Limpeza e Tratamento dos Dados

**Imputação de Dados**: Em datasets reais, é comum encontrar dados ausentes. Para evitar que essas lacunas afetem o desempenho do modelo preditivo, utilizamos técnicas de imputação. A imputação simples com a média, mediana ou moda, dependendo do tipo da variável, é uma das abordagens adotadas.

**Impacto dos Valores Ausentes**: Variáveis como *TotalCharges* podem conter valores ausentes devido a erros de entrada de dados ou novas contas com pouco histórico de cobrança. Substituímos esses valores ausentes com base na média dos dados numéricos, minimizando o impacto no modelo.

**Transformação de Variáveis Categóricas**: Para que os algoritmos de machine learning funcionem adequadamente, é necessário converter variáveis categóricas em numéricas. Técnicas como **One-Hot Encoding** ou **Target Encoding** são utilizadas para transformar colunas como *InternetService*, *Contract* e *PaymentMethod* em valores binários ou numéricos.

**Normalização de Variáveis Numéricas**: Variáveis numéricas com escalas diferentes, como *MonthlyCharges* e *tenure*, são padronizadas para melhorar o desempenho do modelo. A padronização coloca todas as variáveis em uma escala comparável, o que facilita o processo de aprendizagem.

Etapa 3: Arvore de Decisão

As árvores de decisão são modelos de machine learning que funcionam como diagramas de decisão em árvore. Cada nó representa uma condição baseada em uma variável do conjunto de dados, e cada ramo da árvore representa uma possível decisão ou caminho a seguir. No final de cada caminho, o modelo faz uma previsão.

**Vantagens**: Árvores de decisão são simples de entender e interpretar. Elas podem lidar com dados categóricos e numéricos, e são menos sensíveis a dados faltantes. Além disso, possibilitam a visualização clara do processo de tomada de decisão.

**Validação**: O modelo de árvore de decisão foi avaliado usando validação cruzada para garantir sua generalização, e ajustado com o GridSearch para identificar os melhores hiper parâmetros, como profundidade máxima e critério de divisão (gini ou entropia).

**Calculando a eficácia do modelo: ROC AUC**

A curva ROC é uma representação visual do desempenho do modelo em todos os limites. A versão longa do nome, característica de operação do receptor, é uma retenção da detecção de radar da Segunda Guerra Mundial.

A curva ROC é desenhada pelo cálculo da taxa de verdadeiro positivo (TPR, na sigla em inglês) e a taxa de falsos positivos (FPR, na sigla em inglês) em todos os limiares possíveis (na prática, intervalos selecionados), e fazer um gráfico de TPR sobre FPR. Um modelo perfeito, que em algum limite tem um TPR de 1,0 e um FPR de 0,0, pode ser representada por um ponto (0, 1) se todos os outros limites forem ignorados ou pelo seguinte:

Figura 1. Um gráfico de TPR (eixo y) em relação ao FPR (eixo X) mostrando os
            desempenho de um modelo perfeito: uma linha de (0,1) a (1,1).

A área sob a curva ROC (AUC) representa a probabilidade de que o modelo, se receber um exemplo positivo e negativo aleatoriamente, classificará positivo maior que negativo.

O modelo perfeito acima, que contém um quadrado com lados de comprimento 1, tem sob a curva (AUC) de 1,0. Isso significa que há 100% de probabilidade de que o modelo vai classificar corretamente um exemplo positivo escolhido aleatoriamente um exemplo negativo aleatoriamente escolhido.

**Prós:**

* **Independência do threshold**: O ROC AUC avalia o desempenho do modelo em todos os possíveis valores de threshold, fornecendo uma visão mais geral do desempenho em comparação a métricas como acurácia, que dependem de um threshold específico. Isso é útil, pois diferentes aplicações podem exigir diferentes thresholds.
* **Desempenho balanceado entre classes**: O ROC AUC é robusto em cenários com classes desbalanceadas, uma vez que ele considera tanto a taxa de verdadeiros positivos quanto a taxa de falsos positivos. Outras métricas como acurácia podem ser enganosas quando as classes estão desbalanceadas.
* **Comparação entre modelos**: O ROC AUC permite uma fácil comparação entre diferentes modelos, pois fornece uma única métrica que resume a capacidade discriminativa do modelo. Um valor mais próximo de 1 indica um melhor desempenho.
* **Aplicabilidade em diferentes domínios**: Como a ROC AUC foca em como o modelo diferencia entre as classes, ela é útil em uma variedade de domínios, como medicina (diagnóstico de doenças), finanças (classificação de risco) e muitos outros.

**Contras:**

* **Pode ser enganoso em cenários de classes extremamente desbalanceadas:** Embora o ROC AUC funcione bem em dados levemente desbalanceados, em cenários com classes extremamente desbalanceadas a taxa de falsos positivos pode não refletir adequadamente a importância dos positivos verdadeiros.
* **Insensível à erros de classificação em thresholds específicos**: Como a métrica mede a performance global, ela pode mascarar problemas em regiões específicas do threshold. Por exemplo, mesmo que o modelo performe mal em um threshold crítico, a pontuação AUC ainda pode ser alta.

**Por que escolhemos esse método de avaliação?:**

O ROC AUC é escolhido frequentemente porque fornece uma visão global da capacidade do modelo em distinguir entre as classes positivas e negativas, sem depender de um threshold específico. Ele é particularmente útil em cenários onde a escolha do threshold será feita posteriormente ou onde queremos comparar diferentes modelos de forma mais objetiva. Além disso, como ele considera tanto a sensibilidade quanto a especificidade, ele pode fornecer uma avaliação mais balanceada do desempenho do modelo, especialmente quando as classes não estão perfeitamente balanceadas.

Em resumo, o ROC AUC é uma métrica poderosa e versátil, mas deve ser utilizada com cautela em cenários de desbalanceamento extremo ou quando é necessário otimizar diretamente um threshold de decisão.

**Trajetória e métodos escolhidos:**

Primeiro decidimos utilizar o DecisionTree para ter uma ideia de como o modelo receberia os dados, tanto quanto para aprofundar a nossa análise exploratória através da biblioteca de visualização de árvore ***dtreeviz***

Após isso procuramos aprimorar os resultados utilizando um modelo de ensemble learning (Random forest) pois ao combinar múltiplas árvores de decisão, geralmente oferece um desempenho superior. Ele faz isso criando várias árvores a partir de diferentes amostras do dataset e agregando seus resultados (usando uma média ou votação). Isso reduz a chance de um modelo overfittar os dados, o que é um problema comum nas árvores de decisão individuais, apesar do custo computacional mais elevado e sua interpretabilidade ser geralmente mais baixa do que a de uma árvore de decisão

Houve um aumento do roc auc de 0.82 para 0.84

Após isso experimentamos um modelo de bagging (extratreesclassifier) para tentar melhorar o desempenho da previsão através da redução de variância

**Bagging** (Bootstrap Aggregating) é uma técnica de ensemble learning que combina as previsões de múltiplos modelos base (geralmente do mesmo tipo) para melhorar a precisão e a robustez do modelo final.

O bagging começa gerando várias amostras bootstrap a partir do conjunto de dados original. Isso significa que, para cada amostra, um subconjunto aleatório do conjunto de dados é criado com reposição. Isso pode resultar em algumas observações sendo selecionadas várias vezes, enquanto outras podem não ser selecionadas.

Cada modelo (geralmente uma árvore de decisão) é treinado de forma independente em uma das amostras bootstrap.

Para fazer a previsão, as previsões de todos os modelos são combinadas. Para problemas de classificação, isso geralmente é feito por votação majoritária (o rótulo mais frequentemente previsto é escolhido).

O resultado foi insatisfatório, melhoria quase imperceptível (0.84 -> 0.84) e o modelo demorou uma hora para se ajustar aos dados.

Por fim, aplicamos uma técnica de **boosting**, optando pelo **CatBoost** devido à sua capacidade de lidar nativamente com variáveis categóricas, o que é particularmente relevante considerando a abundância dessas variáveis neste conjunto de dados.

O **boosting** consiste em treinar modelos base de forma sequencial, onde cada modelo subsequente se concentra em corrigir os erros do modelo anterior. Essa abordagem permite que os modelos seguintes atribuam maior atenção aos dados que foram classificados incorretamente nas etapas anteriores.

Embora tenha ocorrido uma variação mínima de 0,02 no score ROC AUC, o CatBoost demonstrou uma capacidade de ajuste rápida e eficaz dos dados. Esse desempenho, aliado à praticidade de não exigir a codificação das variáveis categóricas, consolidou o CatBoost como a melhor escolha para este caso específico.

**Conclusão:**

Ao longo deste relatório, identificamos a relevância estratégica de compreender e mitigar o churn dentro do contexto das empresas de telecomunicações. A utilização de métodos analíticos permitiu uma investigação detalhada sobre os fatores que contribuem para a rotatividade de clientes, possibilitando o desenvolvimento de estratégias preventivas mais eficazes baseadas em insights de dados.

Dentre os modelos de machine learning utilizados, o CatBoost demonstrou a maior capacidade de prever churn com alta precisão, conforme evidenciado pela métrica ROC AUC. Este resultado não apenas sublinha a eficiência do CatBoost em manipular variáveis categóricas sem a necessidade de pré-processamento extensivo, mas também destaca sua superioridade em relação a outros modelos testados.

Por fim, a análise realizada e os modelos desenvolvidos oferecem um caminho promissor para reduzir a perda de clientes e aumentar a rentabilidade da empresa. Este estudo reforça o valor da ciência de dados na formulação de estratégias que podem levar a decisões empresariais mais informadas e, por consequência, a uma vantagem competitiva sustentável no mercado. Recomendamos que a abordagem adotada neste estudo seja continuada e aprimorada, utilizando-se a análise de dados como um instrumento fundamental na gestão estratégica de relacionamentos com clientes.

REFERENCIAS

1. Churn e seu Impacto

- KOTLER, Philip; KELLER, Kevin Lane. \*Administração de Marketing\*. 14. ed. São Paulo: Pearson Prentice Hall, 2012.

- Esse livro clássico em marketing traz conceitos sobre retenção de clientes e estratégias para minimizar o churn.

- RICHARDSON, Alan. \*Retention Strategies for Telecom Customers\*. Telecommunications Policy, v. 34, n. 11, p. 666-679, 2010.

- Um estudo específico sobre o impacto do churn em empresas de telecomunicações, abordando métodos para identificar e reduzir a perda de clientes.

2. Análise Exploratória de Dados (EDA)

- TUKEY, John W. \*Exploratory Data Analysis\*. Reading: Addison-Wesley, 1977.

- Tukey foi o pioneiro da análise exploratória de dados, e este livro oferece uma compreensão fundamental da EDA e seus métodos.

- AGARWAL, Ruchi; AGGARWAL, Ajay. \*Data Analytics: Principles, Tools and Practices\*. 1. ed. Nova York: Apress, 2019.

- Esse livro aborda técnicas de análise exploratória de dados, oferecendo exemplos práticos e aplicações no campo de machine learning.

3. Substituição de Valores Ausentes

- LITTLE, Roderick J. A.; RUBIN, Donald B. \*Statistical Analysis with Missing Data\*. 2. ed. Nova York: John Wiley & Sons, 2019.

- Uma das referências mais respeitadas sobre imputação de valores ausentes, cobrindo métodos estatísticos para tratar dados incompletos.

- ALLISON, Paul D. \*Handling Missing Data by Maximum Likelihood\*. Sociological Methods & Research, v. 28, n. 3, p. 301-309, 2000.

- Este artigo oferece uma introdução detalhada às técnicas de substituição de valores ausentes por estimativas de máxima verossimilhança.

4. Pré-processamento de Dados

- GÉRON, Aurélien. \*Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems\*. 2. ed. Sebastopol: O'Reilly Media, 2019.

- Este livro fornece exemplos práticos sobre a conversão de variáveis categóricas e normalização de dados em problemas de machine learning.

- WITTEN, Ian H.; FRANK, Eibe; HALL, Mark A. \*Data Mining: Practical Machine Learning Tools and Techniques\*. 4. ed. Burlington: Morgan Kaufmann, 2016.

- Contém uma seção detalhada sobre preparação e pré-processamento de dados antes da aplicação de algoritmos de machine learning.

5. Árvore de Decisão

- QUINLAN, J. R. \*C4.5: Programs for Machine Learning\*. San Mateo: Morgan Kaufmann, 1993.

- Quinlan desenvolveu um dos algoritmos de árvore de decisão mais influentes, o C4.5. Esta referência explica em profundidade o funcionamento de árvores de decisão.

- BREIMAN, Leo et al. \*Classification and Regression Trees\*. Belmont: Wadsworth International Group, 1984.

- Um dos trabalhos clássicos que introduz árvores de decisão e suas aplicações em machine learning e estatística.

6. Curva ROC AUC

- <https://developers.google.com/machine-learning/crash-course/classification/roc-and-auc?hl=pt-br>

- Um dos trabalhos clássicos que introduz árvores de decisão e suas aplicações em machine learning e estatística.