# Trabalho Final do programa intensivo Data Science da Awari

Aluno: Adriano Marçal Nogueira Neto

## Apresentação detalhada do problema, de sua relevância e do porquê de ser um problema resolvível por um projeto de ciência de dados.

Para o setor de serviços, muitas empresas apresentam o problema do cliente churn. Churn rate (ou taxa de rotatividade) é um índice que calcula a porcentagem de clientes que deixaram de ser consumidores ativos dos serviços ou produtos de determinada empresa. Esse indivíduo, portanto, é classificado como cliente churn. Para tanto, as empresas se especializaram em buscar a retenção de clientes ou a identificação de potenciais clientes churners, haja visto que o custo de se manter um cliente, no geral, é menor do que o custo de aquisição de novos clientes. Em alguns mercados, como o de telecomunicações móveis, o custo de retenção é cinco vezes menor do que o custo de aquisição de um novo cliente (MOZER et al., 2000).

O trabalho tem por objetivo analisar os dados de clientes de uma suposta empresa de telecomunicação, identificar padrões nos dados e criar um modelo de previsão, através de técnicas de aprendizado de máquina, para detectar clientes prestes a sair de um determinado serviço de telecom.

## Apresentação e justificativa da escolha dos procedimentos de coleta de dados utilizados

Por não ter contato direto com nenhuma empresa do setor de serviços comerciais, foi necessário recorrer a uma base de dados que simulasse uma empresa do tipo. Portanto, após buscas, foi encontrado no Kaggle (site especializado em ciência de dados, com repositório de diversas bases de dados, de competições e conteúdos acadêmicos para a área) dados que serviam para a identificação de clientes churners de uma suposta empresa de telecomunicação. Portanto, o passo seguinte foi baixar os dados no formato .csv e manipulá-los através de um DataFrame do Pandas.

## Apresentação e justificativa da escolha dos procedimentos de manipulação de dados utilizados

Após a importação dos dados, alguns procedimentos foram adotados para o melhor aproveitamento das informações coletadas. Os dados importados possuem 5.986 linhas e 21 colunas, dentre esses atributos temos: customerID; gender (gênero do cliente, male / female); SeniorCitizen – o cliente é aposentado (1, 0); Partner (o cliente é casado, Yes, No); tenure (há quantos meses a pessoa é cliente da empresa); PhoneService (existe o serviço de

telefone conectado, Yes, No); MultipleLines (existem múltiplas linhas de telefone conectadas, Yes, No, No phone service); InternetService (o provedor de internet do cliente, DSL, Fiber optic, No); OnlineSecurity (o serviço online de segurança está conectado, Yes, No, No internet service); OnlineBackup - is the online backup service activated (Yes, No, No internet service); DeviceProtection (o cliente tem equipamento de seguro, Yes, No, No internet service); TechSupport (o serviço de suporte técnico está conectado, Yes, No, No internet service); StreamingTV (o serviço de TV de streaming está conectado, Yes, No, No internet service); StreamingMovies (o serviço de streaming de filmes está conectado, Yes, No, No internet service); Contract (o tipo de contrato do consumidor, Month-to-month, One year, Two year), PaperlessBilling (se o cliente utiliza fatura sem papel, Yes, No); PaymentMethod (método de pagamento, Electronic check, Mailed check, Bank transfer (automatic), Credit card (automatic); MonthlyCharges (pagamento mensal atual); TotalCharges (montante total de pagamento do cliente pelos serviços prestados no tempo total; Churn (se o cliente é churn, Yes or No).

Inicialmente, foi retirada a coluna "Unnamed:0" que continha uma espécie de índice de cada linha do DataFrame. Após algumas análises, foi decidido também por retirar a coluna "customerID", pois foi observado que nenhum cliente aparecia mais de uma vez nos dados, e, portanto, foi decidido que essa identificação de cada cliente não era necessária. Finalmente, a coluna "TotalCharges" não era uma coluna do tipo float, mas do tipo object, apesar de apresentar informações numéricas. Dessa forma, existiam informações ausentas na coluna, que não foram identificadas inicialmente, por se tratar de uma string vazia. Com isso, essas 10 linhas que continham informações ausentes foram excluídas do DataFrame.

Exemplo do Dataframe Inicial

| | Unnamed: 0 | customerID | gender | SeniorCitizen | Partner | Dependents | tenure | PhoneService | MultipleLines | InternetService | ... | DeviceProtection | TechSupport |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1869 | 7010-BRBUU | Male | 0 | Yes | Yes | 72 | Yes | Yes | No | ... | No internet service | No internet service |
| 1 | 4528 | 9688-YGXVR | Female | 0 | No | No | 44 | Yes | No | Fiber optic | ... | Yes | No |
| 2 | 6344 | 9286-DOJGF | Female | 1 | Yes | No | 38 | Yes | Yes | Fiber optic | ... | No | No |
| 3 | 6739 | 6994-KERXL | Male | 0 | No | No | 4 | Yes | No | DSL | ... | No | No |
| 4 | 432 | 2181-UAESM | Male | 0 | No | No | 2 | Yes | No | DSL | ... | Yes | No |

Além disso, haviam muito dados categóricos, como "Yes", "No", além de "No internet service" e "No phone service". Nestes casos, foram atribuídos o valor 1 (um) para o caso de "Yes", o valor de 0 (zero) para "No phone service" e 2 (dois) para "No internet service".

Na coluna "gender", foram criadas dummies, substituindo a coluna por uma nova chamada "Male", que caso o cliente seja homem, o valor será um, e caso seja mulher, o valor será zero.

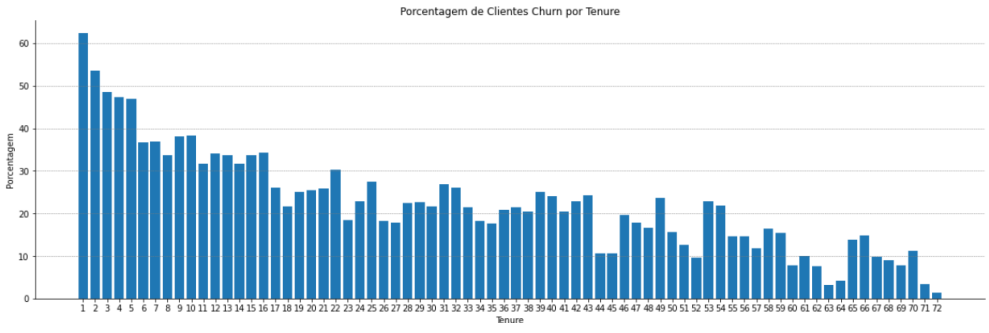
Foram também usadas as técnicas de Label Encoding e OneHot Encoding para os atributos categóricos (não numéricos) das colunas que possuíam mais de duas opções, como as colunas "InternetService", "Contract" e "PaymentMethod". Para o Label Encoding, cada dado categórico recebe um número de código correspondente, enquanto no OneHot Encoding são criadas novas colunas e assinalado o valor um apenas para a coluna que corresponda ao dado categórico.

Finalmente, foram testados dois tipos de escalonamento dos dados. Como haviam dados com escalas muito variadas (como os valores de "tenure", "MonthlyCharges" e "TotalCharges"), foram utilizados o Standard Scaler e o MinMax Scaler para que os dados ficassem balanceados. No primeiro método, os dados são transformados, com média zero e desvio-padrão igual a um. Enquanto o segundo método transforma os dados para que eles fiquem distribuídos numa escala entre zero e um, de acordo com o máximo e mínimo valores encontrados.

Os quatro tipos de dados (Label e OneHot Encoding com Standard e MinMax Scaler) foram usados nos modelos de aprendizado de máquina. O modelo final que alcançou os melhores resultados na métrica utilizada usou os dados com Label Encoding e Standard Scaler.

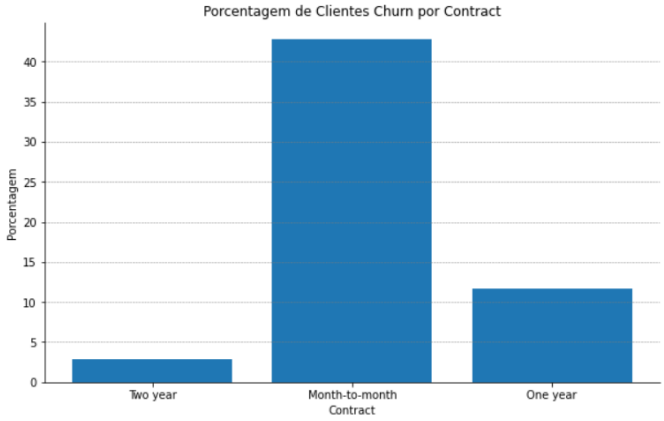
**Apresentação de *insights* retirados da análise exploratória de dados**

- A maior parte dos clientes do tipo churn foram consumidores do serviço por poucos meses (a maior frequência, foi cliente por apenas 1 mês). Clientes com mais de 71 e 72 meses de consumo do serviço eram majoritariamente clientes não churn. Quanto menos meses de 'Tenure', maior a chance de ser um cliente churn.

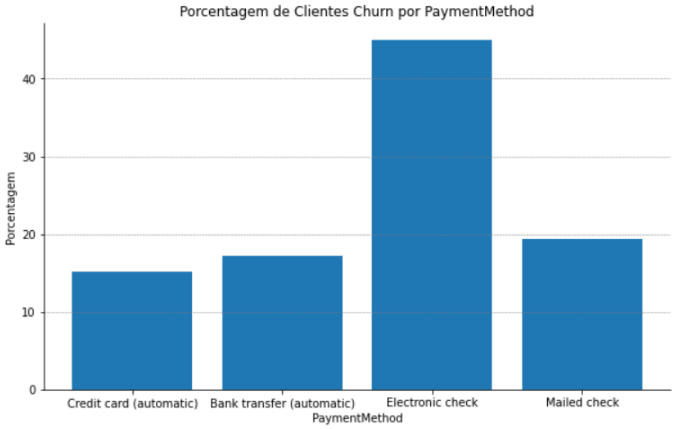



- A homens e mulheres tem a mesma proporção de clientes churn e não churn (não é um atributo determinante).

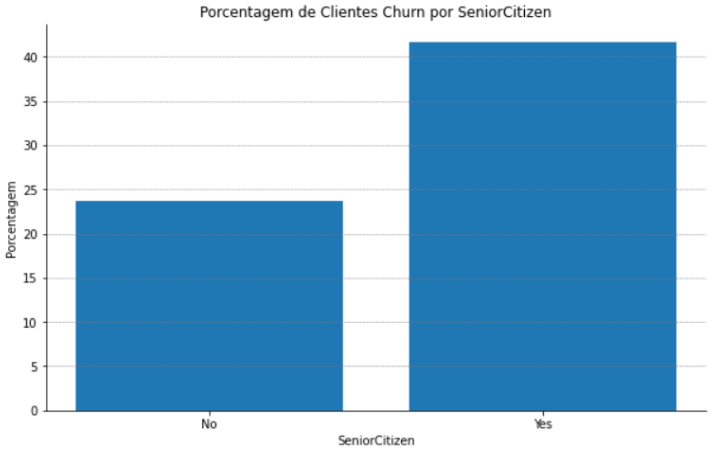
- A forma de contrato com o serviço mostra uma tendência de o cliente ser churn. Clientes com o contrato do tipo "Month-to-month contract" estão mais propensos a ser churn.



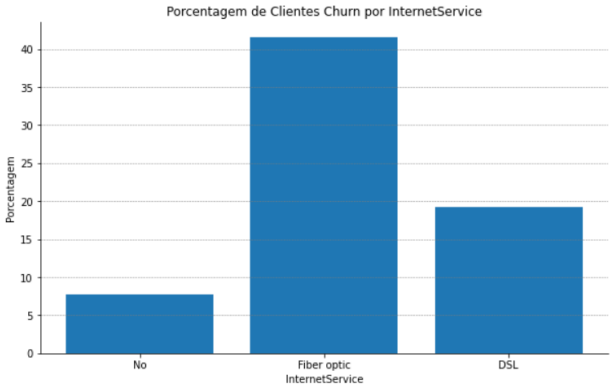

- A forma do método de pagamento do serviço mostra uma tendência de o cliente ser churn. Clientes com o método de pagamento do tipo “Eletronic check” estão mais propensos a ser churn.




- Percentualmente, clientes aposentados (‘SeniorCitizen’) estão mais propensos a serem clientes ‘churn’.




- O tipo de serviço de internet mostra uma tendência de o cliente ser churn. Clientes com serviço de internet do tipo “Fiber Optic” estão mais propensos a ser churn.

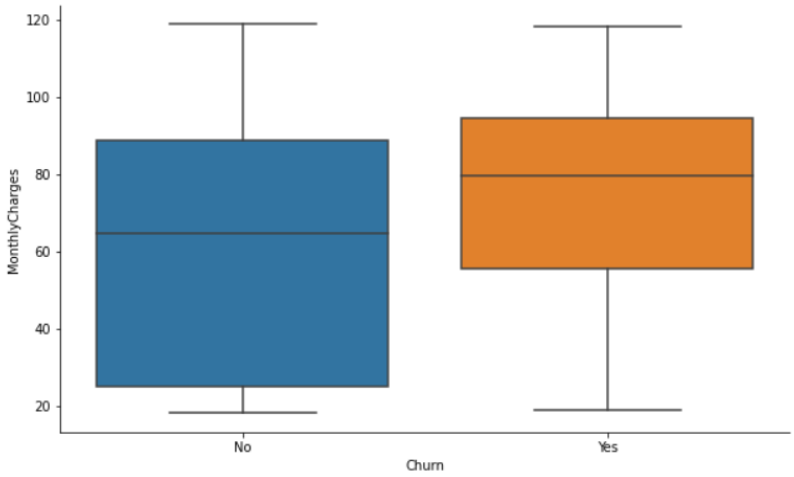



- Percentualmente, um cliente que não tem 'Online Security', 'Online Backup', 'Device Protection' ou 'Tech Support' tem mais chances ser do tipo churn, do que aqueles que tem o serviço.

- Em relação ao 'Paperless Billing', o cliente que opta por essa modalidade é mais provável de ser um cliente churn.

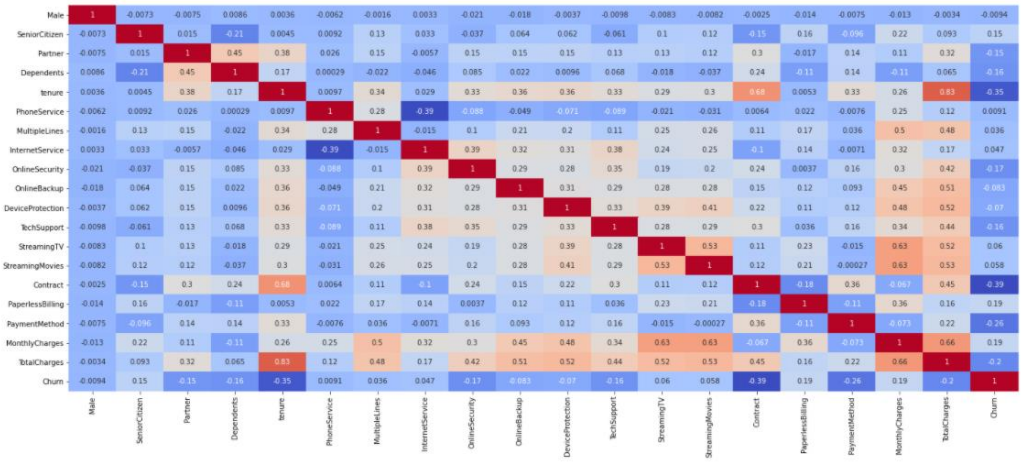
- Clientes que não tem dependentes e não são casados tem uma leve indicação de serem clientes churn.

- Finalmente em relação aos valores, 'Monthly Charges' para clientes do tipo churn tendem a ser mais elevados do que clientes não churn.




- Já para os valores total ('Total Charges'), o oposto ocorre. Isso deve ocorrer, porque clientes churn tem menos meses de serviços, resultando em valores totais menores.

Diagrama de correlações dos atributos




## Apresentação e justificativa da escolha dos procedimentos de modelagem utilizados

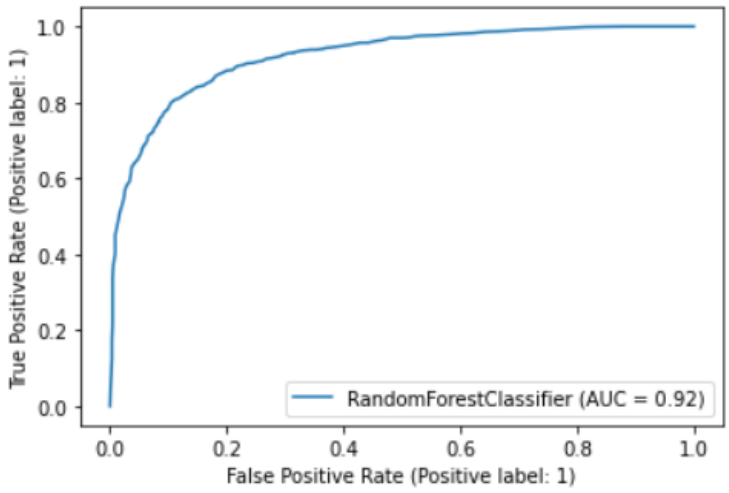
Para a modelagem, foi-se utilizada a biblioteca de Sklearn para fazer os split dos dados de treino e teste, na proporção de 80% para treino e 20% para teste.

Por se tratar de um problema de classificação de aprendizado supervisionado, os algoritmos selecionados foram os relacionados aos problemas de classificação. Os algoritmos utilizados para modelar os dados foram os principais modelos de classificação da biblioteca: LogisticRegression; KNeighborsClassifier; SVC (support vector machines); DecisionTreeClassifier; RandomForestClassifier; GaussianNB; XGBClassifier; SGDClassifier; LGBMClassifier; MLPClassifier.

## Avaliação do modelo final e comparação com o *benchmark* utilizado

A escolha do modelo final se baseia na decisão de quais métricas utilizar para a avaliação do mesmo. Para um problema de classificação, além da acurácia, também é fundamental a avaliação das métricas de precisão e revocação (recall). Além dessas, também foram analisadas a matriz de confusão (confusion matrix) dos modelos, que retornam os resultados de falsos positivos e negativos, o relatório de classificação (classification report), com os resultados de F1-score, e a curva ROC AUC.

Curva ROC AUC



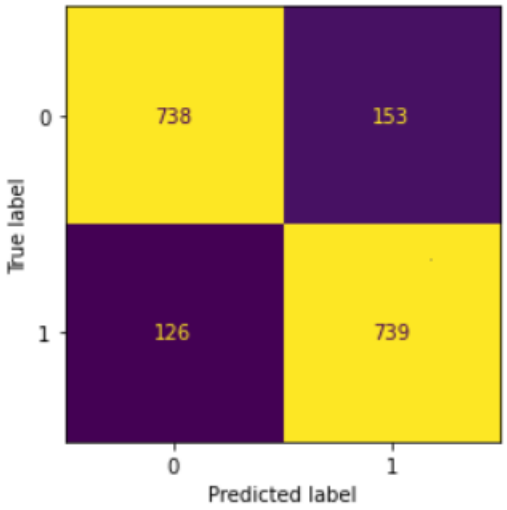

Como estamos trabalhando um modelo de classificação de clientes churn, queremos otimizar o nosso modelo para o recall, ou seja, preferimos ter falsos positivos do que falsos negativos, já que é melhor gastar recursos num cliente, mesmo que ele não seja churn, do que perder o cliente por não identificar que ele é churn.

Relatório de Classificação

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.85 | 0.83 | 0.84 | 891 |
| 1 | 0.83 | 0.85 | 0.84 | 865 |
| accuracy | | | 0.84 | 1756 |
| macro avg | 0.84 | 0.84 | 0.84 | 1756 |
| weighted avg | 0.84 | 0.84 | 0.84 | 1756 |

No entanto, os resultados encontrados para os modelos não foram considerados satisfatórios incialmente. A análise dos dados identificou que os mesmos estavam muito desbalanceados, isto é, existiam muito mais dados de clientes não churn do que clientes do tipo churn: 1587 clientes churn (26.6%) e 4389 clientes não churn (73.4%). Desta forma, o modelo tinha dificuldade de aprender sobre os dois tipos de clientes. Portanto, foi-se decidido utilizar a técnica de SMOTE: método de oversampling que replica as observações com menor quantidade para se equalizar ao número de classificações de maior quantidade. A técnica de SMOTE encontra vizinhos próximos para as classes em minoria para cada amostra das classificações. Em seguida, traça uma reta entre o ponto original e o vizinho para definir a localização da observação genérica.

Matriz de Confusão




Finalmente, pode-se utilizar os novos dados balanceados para o modelo final, que foi o modelo com os melhores resultados nas métricas escolhidas: Random Forest Classifier.

**Reflexões sobre o quão eficaz foi todo o processo para a resolução do problema proposto**

O projeto seguiu uma linha de construção que contribuiu para os resultados finais. A coleta dos dados através de uma plataforma confiável e estabelecida com o Kaggle foi muito importante por facilitar a coleta e permitir analisar dados que são difíceis de se encontrar através de empresas privadas.

A análise geral dos dados para identificar possíveis colunas redundantes ou sem informações importantes, além de linhas nulas, foram importantes para a remoção dessas colunas e linhas que não acrescentavam para a análise do modelo e poderiam prejudicar a performance final.

A análise exploratória permitiu um melhor entendimento do negócio da área de Telecom, podendo ser possível entender o comportamento dos clientes e as variáveis mais importantes para o modelo final.

O escalonamento dos dados, devido à grande variância dos valores encontrados, e principalmente a técnica SMOTE de oversampling permitiram que a performance do modelo final aumentasse consideravelmente.

Finalmente, a escolha do algoritmo utilizado se deu pela avaliação das métricas certas para se alcançar o objetivo final considerado no projeto, ou seja, a identificação de clientes do tipo churn.

**Apresentação de melhorias possíveis em seu projeto (da aquisição de dados à modelagem)**

Para uma melhor performance do modelo final, seria necessário a coleta de um maior número de dados, visto que a quantidade de entradas de dados estava abaixo de dez mil. Além disso, seria melhor indicado dados mais balanceados, que não precisassem a utilização de uma técnica de oversampling. Finalmente, uma melhor capacidade de processamento computacional poderia permitir a utilização de modelos mais complexos, como redes neurais, além de um grande volume de dados para se treinar.