Machine Learning Engineer Nanodegree

Projeto Final

Gabriel Ramos Uaquim
26 de dezembro de 2018

I. Definição

Visão Geral do Projeto

Em 2014 com a recessão do país, a arrecadação caiu. Isso fez a Prefeitura Municipal de Salvador (PMS) voltar sua atenção para ações de cobrança. Dentre elas, uma reorganização das coordenações e setores dentro da Secretaria da Fazenda da PMS (SEFAZ-PMS), com isso foi criada a Coordenação de Cobrança, específica para pensar e executar a cobrança dos débitos dos contribuintes.

Alinhado a este raciocínio o presente trabalho busca soluções que visem otimizar o problema da cobrança de débitos. O uso de machine learning para tratar este problema não é novo ou recente. Diversas técnicas foram utilizadas [1] porém sempre foi dado preferência a técnicas lineares, devido ao seu alto poder de interpretação. Em [4] os autores tentam combinar estes métodos com técnicas com mais variância (árvores) para aumentar o poder de predição.

Em [3] e [4] se ressalva a importância da modelagem do problema, das variáveis escolhidas e do resultado a ser alcançado, pois aumentando essa especificidade ajuda a manter a estabilidade de modelos com maior variância como deep learning.

Porém a maioria desses técnicas e estudos são voltados para análise de risco e crédito no setor privado. Foi no início dos anos 2000 que uma técnica de cobrança no setor público ganhou notoriedade. Em Nova York, foi construída um sistema que utilizava um agente de aprendizagem por reforço para mapear a ação de cobrança do governo com o contribuinte e débito que deveria ser cobrado. O trabalho é descrito em [2].

O presente trabalho tenta adequar as técnicas usadas no setor privado para a análise da possibilidade de retorno de um débito do contribuinte. Para tanto, serão utilizados os dados referentes à 2016 e 2017.

Para modelar a variável alvo será feita a seguinte metodologia:

1. Para determinar se uma dívida é cobrável ou não será selecionado como “Cobrável” todos os débitos pagos no intervalo de 30 a 360 dias.

2. Os débitos não pagos nesse intervalo serão selecionados como “Não Cobrável”

Descrição do Problema

Os débitos que o contribuinte pode ter possuem diversas naturezas: oriundos de uma ação fiscal (ex: auto de infração), do não pagamento de um tributo devido, do não pagamento de um parcelamento e etc.

Dentre as fases do ciclo de crédito apresentados em [1], este trabalho tratará da Collection Score. Ou seja, daqueles débitos compreendidos no intervalo de atraso de

15 a 180 dias. Porém, como estamos tratando de governo, onde as pessoas tendem a atrasar mais as suas obrigações, e com a indicação da área de negócio, este prazo será aumentado para 30 e 360 dias.

Assim, o problema é entender a relação entre as diversas características dos dados que são coletados como por exemplo (data de constituição do débito, valor, origem, se está em dívida ativa e etc) e uma chance de retorno desse débito. Além de entender essa relação, este projeto também tem como meta dar visibilidade a estes dados e propor soluções que otimizem o trabalho de cobrança.

Métricas

Os analistas e auditores do setor de cobrança utilizam um método ad-hoc para definir o que será cobrado. Uma lista é enviada à cobrança a cada dois meses. Dessa lista os débitos são priorizados conforme intuição.

Assim, este trabalho irá buscar classificar a saúde do débito. O modelo utilizado para se determinar a saúde do débito é chamado de Collection Score e faz referência aos débitos pagos entre 15 e 180 dias (neste trabalho, por ser governo, 30 a 360 dias). Como modelo de referência temos:

Acurácia Básica= Total Pagos / Total Débito

No conjunto inicial, com 2.481.582 de registros a acurácia básica é de 34%. Usando o modelo básico GaussianNB, do sklearn, os valores de acurácia e fbeta (beta = 2) foram de 0.52 e 0.77 respectivamente.

A aferição da eficiência do modelo se dará a partir de quatro métricas:

1. Acurácia: métrica de fácil avaliação e principalmente fácil comunicação aos gestores da área de cobrança. Ela não será avaliada unicamente, pois as classes da variável dependente são desbalanceadas (76% e 34%), foi escolhida mais pela facilidade de comunica-la e explica-la.

2. Fbeta: para o presente trabalho temos recall e precision conforme descrito abaixo:

    a. Recall: um recall alto implica que poucos contribuintes que poderiam pagar foram deixados de fora. Em outras palavras, a Secretaria arrecada mais.

    b. Precision: uma boa precisão implica pouco desperdício no esforço da cobrança, aquilo que é cobrado é recebido.

    c. Beta = 2, pois o recall é uma métrica mais importante para a área de cobrança.

3. Curvas AUC: principalmente para comparar os diversos modelos, bem como também para comunicar graficamente o modelo escolhido.

4. Tempo de treino: o conjunto de dados é grande, e este trabalho deve ser repetível, a ponto de se construir uma funcionalidade para o próprio gestor treinar se for preciso. Tempos muito longos implicam em baixa usabilidade.

II. Análise

Exploração dos dados

O conjunto de dados usado neste projeto foi obtido a partir do DataMart de cobrança da prefeitura. Foram extraídos os dados relativos aos anos de 2016 e 2017. Este conjunto de dados inclui os seguintes parâmetros:

COD_DEB: é o código único do débito. Um único contribuinte pode possuir diversos débitos. A base usada para este trabalho possui 3.338.995 débitos.

COD_SSIS_ORIG: é o código de identificação do sistema que originou aquele débito. A maioria dos débitos são originados no sistema de administração tributária, SAT. Esse sistema é o principal sistema da Secretaria. O PAT cuida dos débitos relativos ao parcelamento, esses débitos são poucos (50) e possuem um tratamento bem específico. Esses débitos estão fora do escopo do trabalho. Abaixo segue uma tabela com os números de cada sistema:

Sistema	Quantidade de registros
SAT	3.331.921
SALUS	4604
AINFL	2420
PAT	50

COD_TIP_ATLZ_DEB: são três os valores que podem ser assumidos. “I” é o débito incluso no CDM (Cadastro de dívidas do munícipio), “A” é o débito que sofreu alguma atualização, seja por ação da prefeitura (aplicação de multas e juros) ou por ação do próprio contribuinte (via processo administrativo). “N” são os débitos que não podem ser cobrados, seja porque foram para o parcelamento, dívida ativa ou pagos.

Código	Quantidade de registros
I	1.266.321
A	12.773
N	2.059.901

DESC_TIP_CTBU_ORIG: descreve a natureza do contribuinte, podendo este ser um imóvel ou uma pessoa, física ou jurídica. Existem também os classificados como contribuintes não tributários, esses débitos não estão sob a competência da Coordenação de Cobrança e por isso serão deletados do conjunto analisado.

Tipo de contribuinte	Quantidade de registros
Inscrição Imobiliária	2.794.292
Inscrição Mobiliária (autônomos)	156.784
Inscrição Mobiliária (estabelecimentos)	387.603
Contribuinte não tributários	316

DESC_TIP_DEB: é o tipo de débito do contribuinte, podendo ser impostos, contribuições ou dívidas de outra natureza. Também indica a localização do tributo, como em “Dívida Ativa de ISS”. Esse parâmetro será utilizado, também, para identificar os débitos em dívida ativa. Estes débitos estão na competência da Procuradoria do Município e, portanto, estão fora do escopo desse trabalho. Abaixo segue a lista dos 5 com maiores quantidades.

Tipo de débito	Quantidade de registros
IPTU	1.398.376
TRSD	1.395.655
TFF	286.388
ISS	171.324
TFF Dívida ativa	49.334

DESC_TIPO_LANC: descrição de como surgiu o débito. Segue os 5 maiores:

Tipo de débito	Quantidade de registros
Direto	2.536.063
Inscrição	267.079
Estabelecimento	235.379
Autônomo	150.462
RDT próprio	46.532

DT_ATLZ_TAB, DT_EXCL_DEB, DT_INCL_DEB: são os únicos campos de data do conjunto, e serão utilizados para encontrar a variável alvo. Possui as seguintes regras de negócio:

1. Quando o débito é excluído DT_ATLZ_TAB = DT_EXCL_DEB
2. Quando o débito ainda está aberto DT_INCL_DEB > DT_EXCL_DEB

MÊS: o mês que nasceu o crédito que originou a dívida

VAL_JURO_MRA, VAL_MLTA_MRA, VAL_MLTA_PUNI: são valores acrescidos ao principal do débito, porém só são calculados quando há alguma atualização do débito. Quando a dívida nasce já com esses valores eles são automaticamente adicionados ao principal, zerados e adicionados como dívidas.

VAL_PRIN_DEB: é o valor do principal da dívida.

debitos_contribuinte: é a quantidade total de débitos (dos anos 2016 e 2017) que aquele contribuinte possui. Todos os débitos desse contribuinte terão o mesmo valor nesse campo.

Limpeza e Engenharia de Características

Para utilizar o conjunto de dados foram executadas as seguintes etapas:

Remoção de Registros

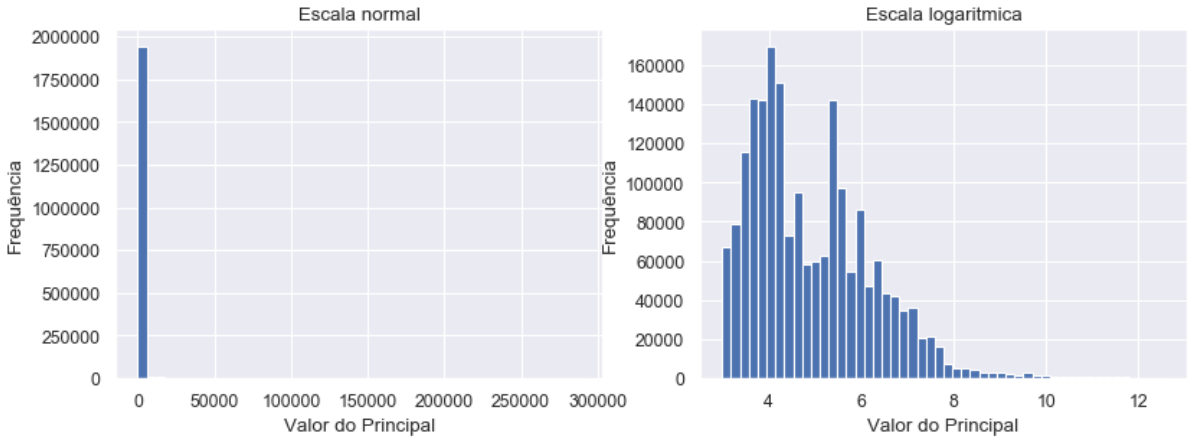
- Removidos os débitos que não possuíam contribuintes associados no cadastro imobiliário. Esses débitos existem, e são oriundos de inconsistências dos sistemas legados. Dessa forma só foram usados os débitos que possuía o atributo debitos_contribuinte. Pois para encontrar este valor foi necessário associar os débitos a um contribuinte antes.
- Remoção de débitos de contribuintes não tributários, inscritos em dívida ativa e oriundos de Parcelamentos. Não são do escopo deste projeto.
- Total de débitos removidos: 738.988 (22.13% do total)

Engenharia de Características

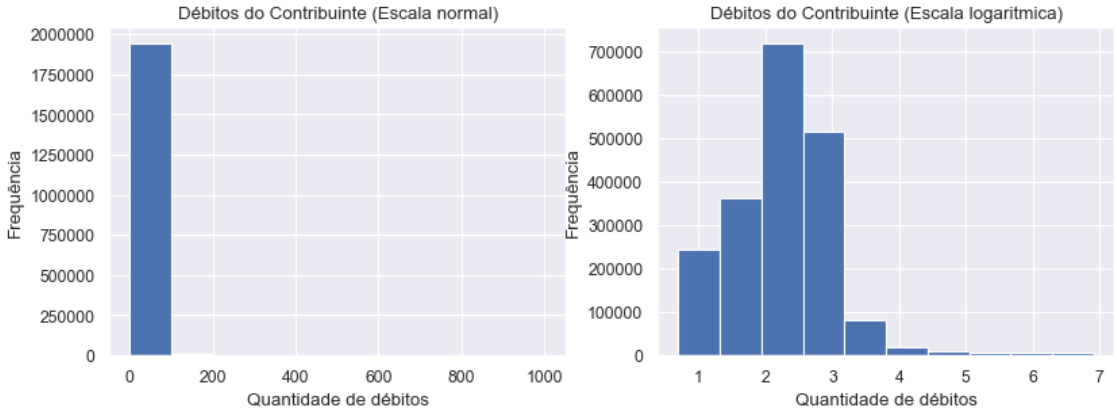
- Campo **dias_ate_pagar**: para os débitos em aberto (Data_Exclusão < Data_inclusão) a data de exclusão foi setada para a data_atual (18 de dezembro de 2018). Após isso, dias_ate_pagar = Data Exclusão – Data Inclusão;
    - Não foi encontrada utilidade para a Data_atualização e por isso ela foi excluída da análise.
- Target (variável alvo): Cobrável são aqueles débitos considerados “saudáveis”, ou seja, pagos entre o intervalo de 30 a 360 dias. Assim:
    - Foram eliminados da análise os débitos pagos em menos de 30 dias. Eles são enquadrados na categoria de Self Cure e não demandariam uma ação de cobrança.
    - Todos os débitos com COD_TIP_ATLZ_DEB == “N” e dias_ate_pagar <= 360 foram setados para “Cobrável”. Os demais para “Não cobrável”.
- Log_principal:
    - O valor do principal é altamente *right skewd*, por isso foi preferido utilizar uma transformação logarítmica para destacar melhor a distribuição, a transformação foi feita apenas após a análise dos outliers:

Outliers:

- Principal:
    - Foram removidos os valores muito baixos (menor do que 20 reais). Esses valores não podem ser alvo de uma ação de cobrança. Total de valores removidos: 109.089
    - Para encontrar os demais outliers foi encontrado um threshold = média + 2 * desvio padrão.
    - Esses outliers foram removidos. Eles representavam uma percentagem de 0.055% do conjunto de dados.
    - Após a remoção dos outliers, o principal foi transformado para a escala logaritmica:



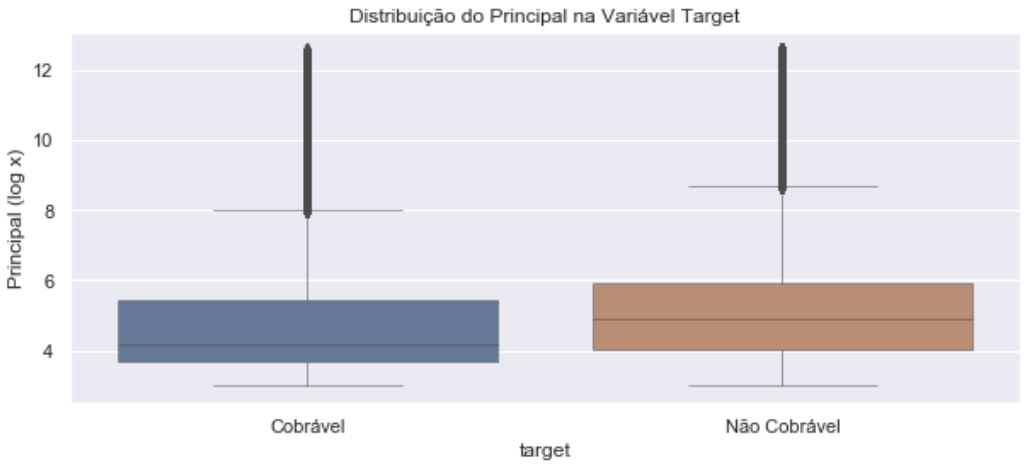
- Debitos_contribuinte: não faz sentido remover esses outliers. Pois é importante capturar o comportamento desses grandes empreendimentos que possuem muito débitos. Para melhorar a visualização e também analisar melhor a distribuição eles também foram transformados utilizando a função logarítmica:



Exploração Visual

Variáveis Numéricas

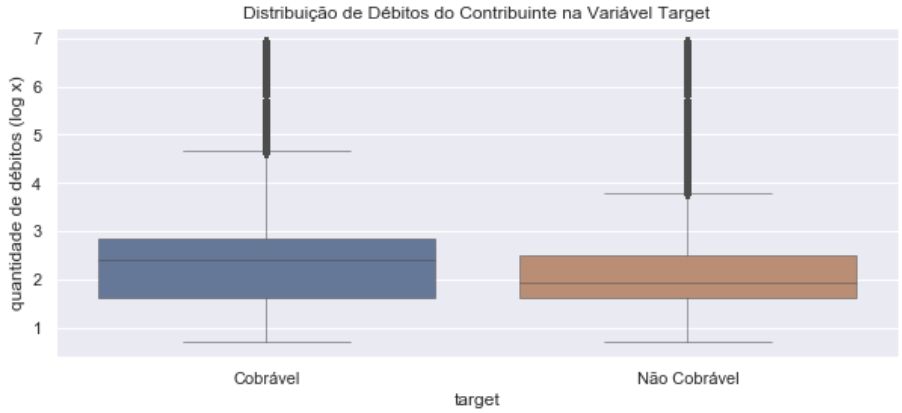
- Principal



  - O valor do principal possui influência na variável alvo. Quanto maior o valor do principal, maior é a probabilidade de o contribuinte não pagar. Porém a

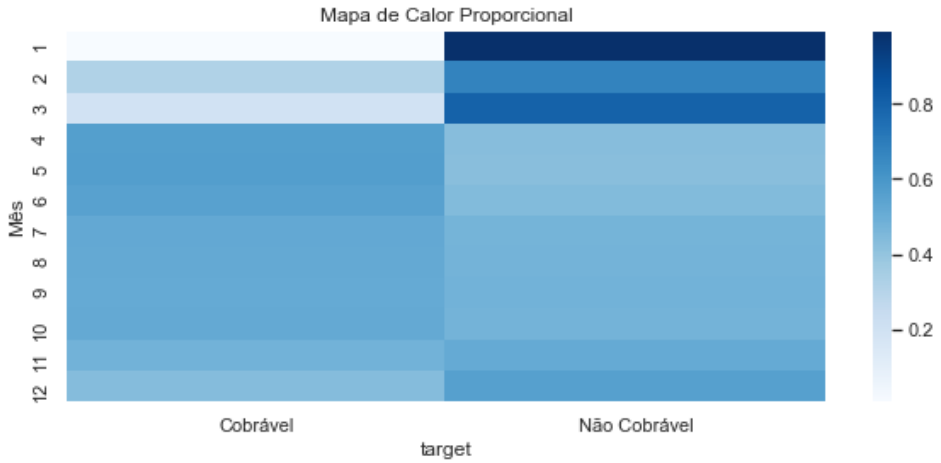
proximidade das caixas indica que não se trata de uma variável tão determinante.

- Débitos Contribuinte
    - Outra Variável não tão determinante para definir a variável alvo. Observa-se apenas uma tendência que indica que quanto mais débitos têm o contribuinte maior a chance de ele pagar o débito.

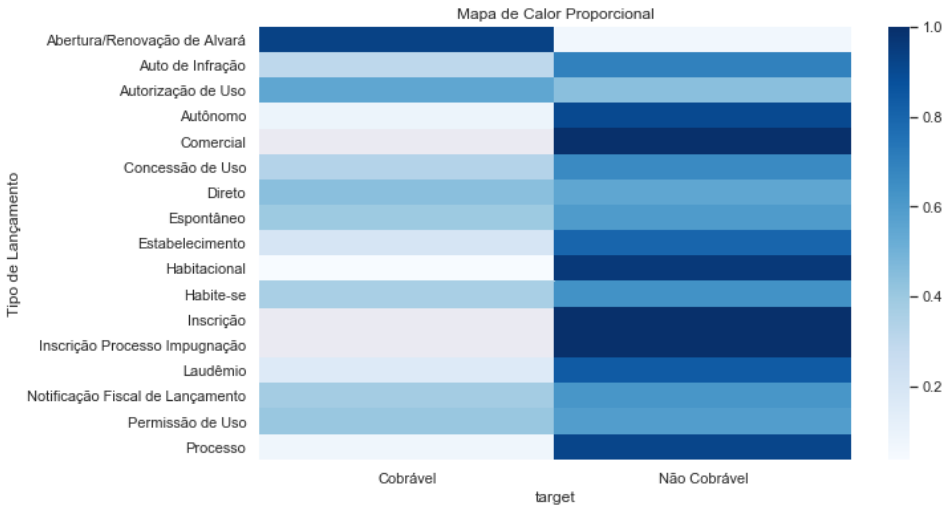


Variáveis Categóricas

- Estratégia de visualização: foram utilizados mapas de calor para visualizar a importância das variáveis categóricas como determinantes da variável alvo. Porém como os valores estão bastante desbalanceados (+90% de inscrições imobiliárias e etc) os mapas de calor foram alterados para facilitar a visualização.
    - Foi utilizado uma proporção para cada valor de variável. Por exemplo: para cada valor único de DESC_TIP_LANC foi determinado um valor de forma que a soma de todas as linhas deste valor some 1.
    - Assim é possível visualizar a contribuição de cada valor individual para se determinar a variável alvo.
- **Mês**

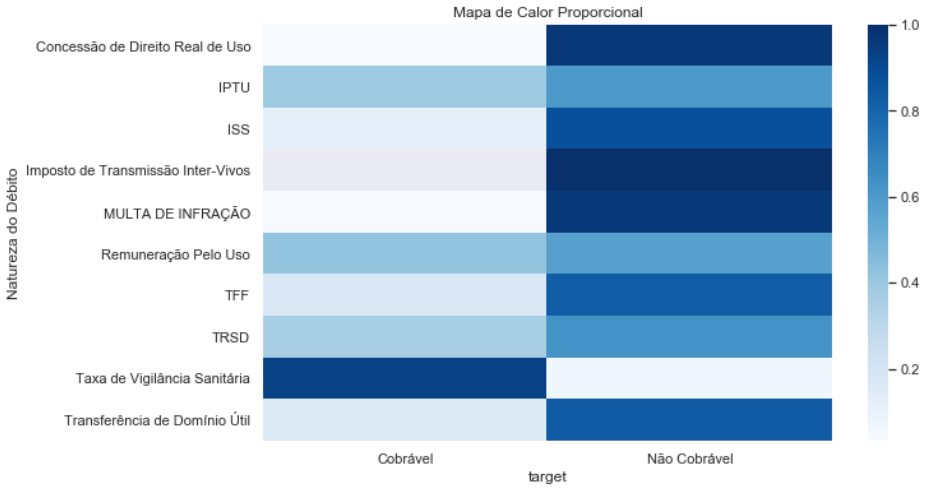


    - O início do ano é marcado por débitos com pouca saúde. É nessa época que é lançado os débitos de IPTU, TFF e TRSD. É nessa época também que as famílias possuem mais gastos e assumem mais dívidas. Talvez isso explique essa defasagem.
- **Tipo de Lançamento**



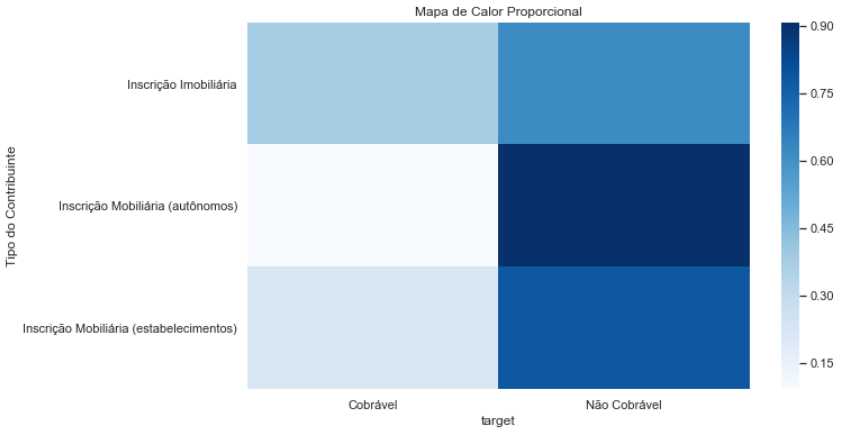
- Observa-se que os débitos que possuem melhor índice de retorno são aqueles cuja a punição pela inadimplência é maior.

- **Natureza do débito**



- O que mais se destaca aqui é a ineficiência da SEFAZ de cobrar ISS, o principal tributo da prefeitura. Tanto o ISS como a multa de infração (geralmente advinda da fiscalização de ISS) possuem péssimas taxas de retorno. Outro tributo atrelado ao comércio é a TFF que também possui uma péssima taxa de retorno

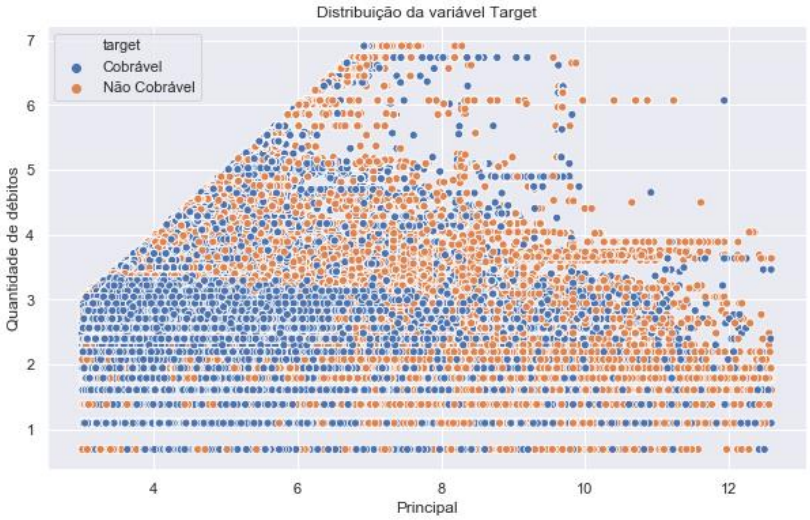
- **Tipo de Contribuinte:**



- Vê-se mais claramente o cenário esboçado no gráfico anterior. Débitos advindo tributos por atividade comercial (inscrição mobiliária) possui péssima saúde e devem ser tratados com mais atenção.

Target

- A proporção final de débitos cobráveis ficou em 34.11%. Abaixo segue um scatterplot entre a quantidade de débitos e o principal. Aparentemente usando essas duas características já é possível visualizar uma separação entre as classes da variável target. Ambos os eixos estão na escala logarítmica.



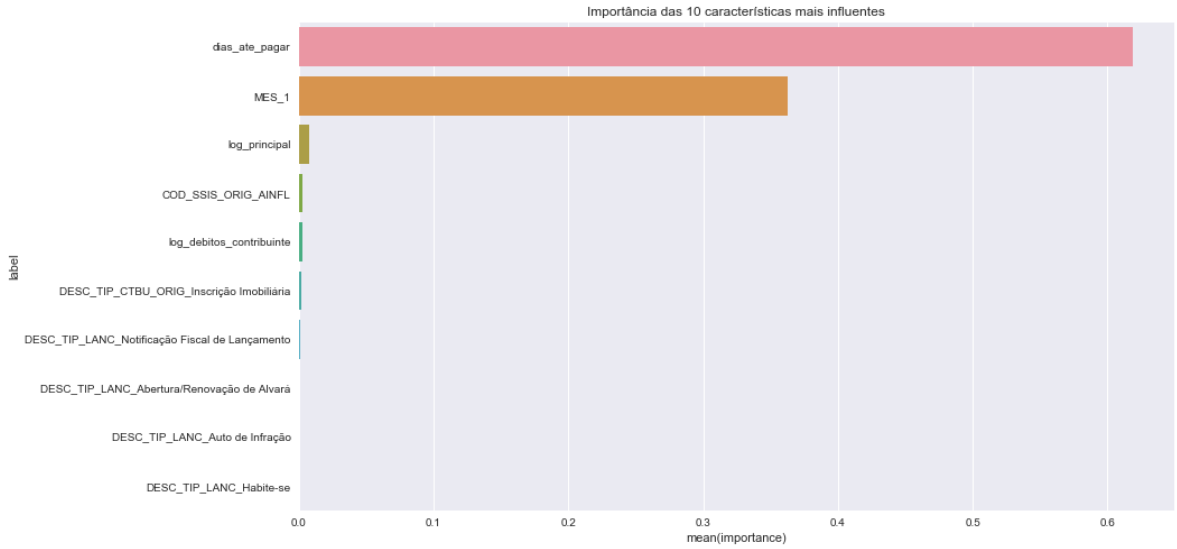
Algoritmos e Técnicas

Avaliação de Características

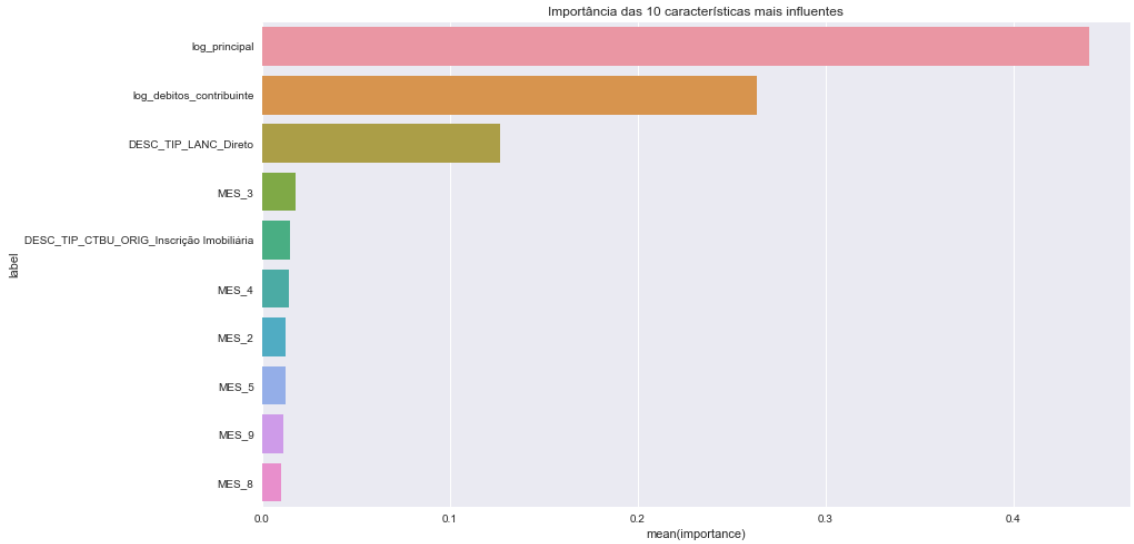
Para avaliar as variáveis independentes foi utilizado uma árvore de decisão. Porém antes de encaixar o modelo, todas as variáveis categóricas foram transformadas em *dummie variables* e as variáveis numéricas foram escaladas utilizando o StandardScaler do sklearn. Apesar de árvores de decisão não precisarem de variáveis escaladas, o momento foi aproveitado para preparar o conjunto de dados para os modelos que serão usados para a previsão das porcentagens.

Antes do encaixe da árvore o conjunto foi separado em teste e treino na proporção 30% e 70% respectivamente. Essa separação foi feita para que o desempenho da árvore fosse testado no conjunto de teste. Caso as métricas de acurácia e fbeta fossem superiores aos 90%, seria prova de que uma das variáveis estaria dominando a análise.

O primeiro encaixe resultou em uma acurácia de 99.7% e fbeta de 99.5%:



A variável dias_ate_pagar está dominando a análise. Isso acontece porque ela foi utilizada para se definir a variável alvo, logo a correlação entre elas é muito grande. Para diminuir esta correlação e tornar o algoritmo mais robusto a variável foi removida. Um segundo encaixe gerou acurácia 77.3% de e fbeta de 64%. Esses valores no conjunto de teste demonstram que a correlação entre as variáveis dependentes e independentes diminuiu e, portanto, o conjunto de variáveis é um bom candidato para se construir um modelo robusto. Segue as importâncias das características:



Como mostrado no scatter plot anterior, as variáveis principal e quantidade de débitos foram as mais importantes para a determinação da fronteira de classificação.

Benchmark

A acurácia do modelo aleatório é de 34%. Para traçar uma baseline para avaliar os diversos modelos a serem testados, foi utilizado o modelo padrão GaussianNB(Naive Bayes) para encontrar valores de fbeta e acurácia. Para o modelo básico:

- Acurácia: 0.52;
- Fbeta (beta=2): 0.77

III. Metodologia

Pré-processamento de dados

Remoção de variáveis
Foi removida a variável dias_ate_pagar por dominar muito a análise, conforme descrito no capítulo anterior.

Transformação logarítmica
As variáveis débitos_contribuintes e PRINCIPAL foram transformadas para a função logarítmica. Suas funções de densidade ficam mais visíveis e melhor definidas quando transformadas.

Padronização
Foi utilizado o StandardScaler() do sklearn para padronizar todas as variáveis numéricas.

Divisão em Teste e Treino
O conjunto foi dividido em teste e treino na proporção 20% e 80% respectivamente. Para tanto foi utilizado o *train_test_split* do sklearn.

Seleção de variáveis
Foram testados os métodos de PCA e SelectkBest para identificar a melhor estratégia de seleção de variáveis. Os resultados foram comparados com o método básico (descrito em Benchmark).

Para a seleção do número de componentes para o PCA foi utilizado um scree plot, e depois testado valores próximos ao “cotovelo” do gráfico. Foram utilizados os mesmos valores para o SelecKBest.

Após encontrados os resultados, eles foram organizados na seguinte tabela:

Método	N_components	fbeta	Accuracy
Nenhum	0	0.768	0.52
SelectkBest	8	0.767	0.515
SelectkBest	7	0.767	0.515
SelectkBest	6	0.756	0.453
PCA	8	0.756	0.591
PCA	7	0.748	0.586
PCA	6	0.744	0.589

Os valores de fbeta encontrados variam muito pouco. Menos de 2 pontos percentuais, uma diferença que provavelmente seria superada com o uso de tunning. Porém a variação encontrada na acurácia é bastante significativa. Dessa forma o método de feature selection escolhido foi o PCA(n_components=8).

Implementação

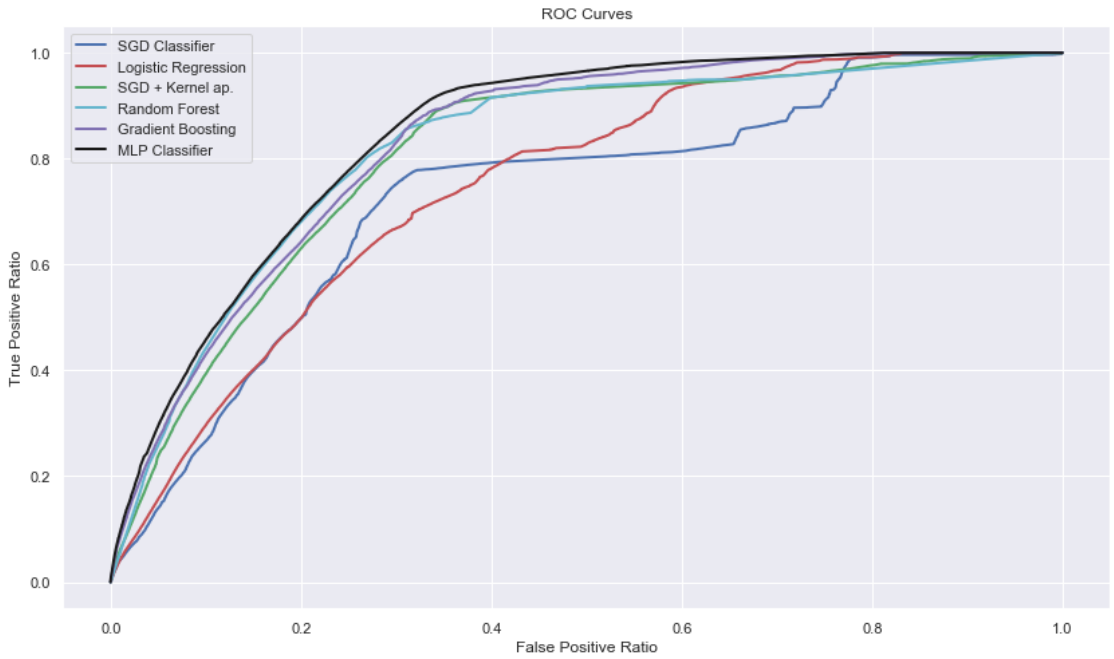
Seleção do Modelo

Foram testados 6 modelos de machine learning. Foram usados os parâmetros padrões para todos os modelos testados. Somente o random_state foi setado para 42 em cada um dos modelos. O encaixe foi feito no conjunto de treino e as métricas foram levantadas no conjunto de teste. Para a comparação dos modelos foi considerado o tempo de treino, o fbeta e a acurácia. Abaixo segue a tabela com as referidas métricas. As linhas estão ordenadas em ordem descrescente do Fbeta.

Modelo	Tempo de Treino (seg.)	Fbeta	Acurácia
MLPClassifier	216.011	0.7	0.76
SGD + Kernel	6.915	0.665	0.741
Random Forest	69.824	0.638	0.759
Gradient Boosting	298.874	0.633	0.747
SGD Classifier	1.306	0.571	0.701
Logistic Regression	5.743	0.493	0.697

Além do levantamento desta tabela, também foi criado um gráfico ROC-AUC para avaliar o desempenho geral de cada algoritmo.

É importante observar que quase todos os modelos apresentaram uma perda em fbeta em relação ao modelo de benchmark, porém o ganho em acurácia foi bastante significativo, implicando que o modelo de benchmark tenha provavelmente muitos falsos positivos.



O modelo com melhor desempenho foi a rede neural (MLP Classifier). Porém seu tempo de treino foi de aproximadamente dois minutos, o que para a fase de tunning com GridSearchCV e Cross Validation fica inviável. Outro obstáculo para a escolha de um modelo com alto tempo de treino é a validação do resultado frente a área de negócio. Quando o modelo for apresentado, há chances de mudanças serem propostas. Um modelo rápido dará respostas mais rápidas à área de negócio.

Um modelo muito mais rápido e com métricas de acurácia e fbeta próximo à rede neural é o modelo Kernel Approximation + SGD Classifier. Além de apresentar bons resultados e um tempo de treino bom, esse modelo não tem tanta variância como os modelos de ensemble e a rede neural. Para conjuntos muito grandes o risco de overfitting de modelos com mais variância é maior.

Modelo escolhido: Kernel Approximation + SGD classifier.

Refinamento

Para o processo de refinamento foi utilizado o Pipeline + GridsearchCV do sklearn. A necessidade do uso do Pipeline se deu para a calibragem dos parâmetros do RBFSampler (kernel approximation).

Para o método de avaliação foi utilizado um StratifiedKFold, com n_splits=5, tendo como referência a métrica fbeta com beta = 2. A necessidade do StratifiedKFold é devido ao desbalanceamento da variável dependente (apenas 34% é uma resposta positiva).

Foram calibrados os seguintes parâmetros:

1. RBF Sampler: Gamma e N_components;
2. SGDClassifier: Loss, Penalty, Alpha e tol

Resultando em 2160 fits e 427 minutos de treino.

Este refinamento levou o fbeta a uma pontuação de 0.774, superior ao melhor modelo padrão encontrado na fase de seleção de modelos.

Após essa etapa foi feito um segundo refinamento testando parâmetros próximos aos encontrados. O único parâmetro alterado foi gamma=3 do RBFSampler. Resultando em uma melhora do fbeta para 0.783.

IV. Resultados

Avaliação do Modelo e Validação.

O modelo final foi avaliado com as seguintes estratégias:

1. K-fold Cross Validation (K=3) no conjunto de treino;
2. Encaixe no conjunto de treino e validação no conjunto de teste;
3. K-fold Cross Validation (K=3) no conjunto teste;

Antes de aplicar as estratégias, os conjuntos de treino e teste foram redefinidos com o random_state=0 (antes era 42) e com uma proporção de 0.7 e 0.3 respectivamente. Abaixo segue a tabela comparando os valores:

Estratégia	Fbeta	Accuracy
1	0.784	0.736
2	0.783	0.736
3	0.784	0.737

A variação nas estratégias permite que diversos conjuntos de input sejam testados. Dessa forma é possível avaliar a robustez do algoritmo. Como a tabela demonstra, há pouca variação em ambas as métricas em todos os cenários. A pouca variação nas métricas revela robustez do algoritmo.

Justificativa

O modelo de benchmark adotado foi o uso do GaussianNB padrão do sklearn. Nos mesmos cenários descritos acima este modelo obteve as seguintes pontuações:

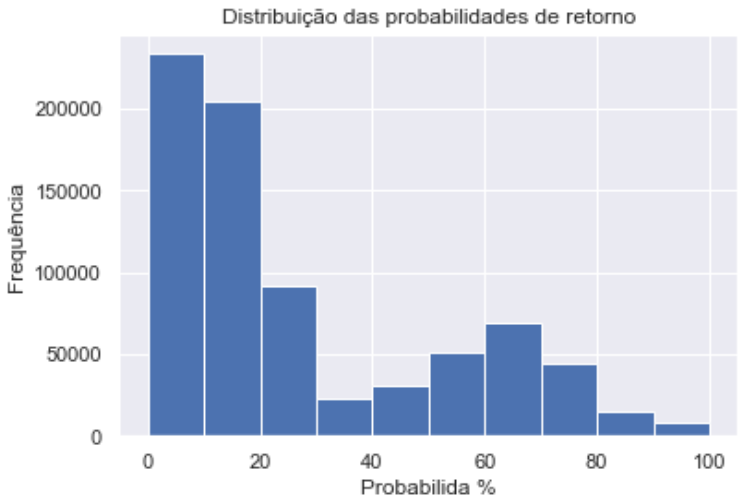
Estratégia	Fbeta	Accuracy
1	0.767	0.521
2	0.767	0.52
3	0.767	0.52

O modelo de benchmark se mostrou também robusto, porém o custo em acurácia é muito grande. Um alto fbeta (beta=2) e uma baixa acurácia implica que a SEFAZ iria desperdiçar muitos recursos cobrando de pessoas que teriam poucas chances de pagar o débito (muitos falsos positivos).

Desta forma, o modelo proposto traz um enorme ganho em relação ao modelo de benchmark. Pois além de ter uma média de fbeta um pouco superior, o ganho em acurácia foi bastante significativo.

Aplicação do Modelo

O interesse do trabalho é avaliar os débitos em aberto. Dessa forma, do conjunto original de débitos foi extraído todos aqueles cujo COD_TIP_ATLZ_DEB é “I” ou “A”. As probabilidades estão distribuídas da seguinte forma:



Como esperado a maior parte dos débitos está concentrada em porcentagens abaixo de 40. Porém, muitos débitos possuem boa recuperabilidade, 137.353 débitos acima de 60 %, somando um total de R$ 76.678.006,65, uma quantidade bastante relevante para o munícipio.

O principal aspecto e resultado do modelo é dar um conjunto de porcentagens confiáveis de forma que seja possível ranquear os débitos dos contribuintes e a partir deles tomar ações de cobranças. O atual modelo representa um ganho bastante significativo em relação ao modelo atual (praticamente aleatório).

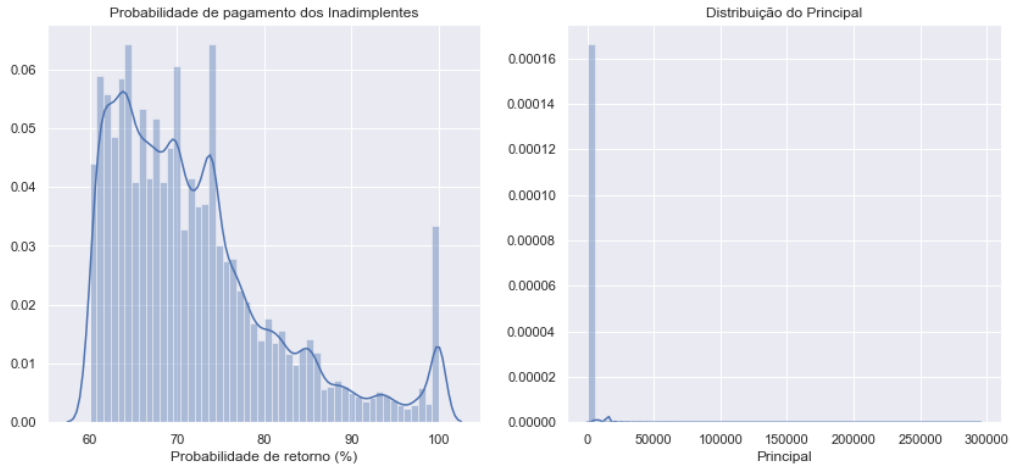
V. Conclusão

Analisando os resultados

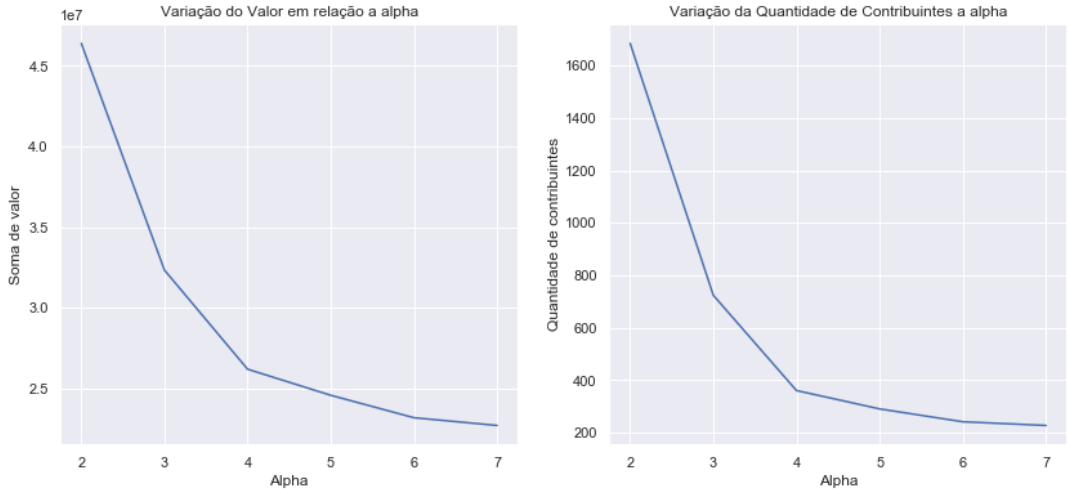
Contribuintes com alta recuperabilidade e altos valores do Principal

Esta análise diz respeito somente aos débitos em aberto. Foram classificados como de boa recuperabilidade os débitos com probabilidades (ou percentagens) maior que 60%. Dentre estes os outliers (em referência ao Principal) são aqueles poucos contribuintes que representam a maior parte do débito.

A ideia é que uma ação focada nesses poucos contribuintes tenha um retorno mais rápido à prefeitura. Para apoiar essa análise foram plotados dois gráficos, O gráfico da esquerda mostra a distribuição das probabilidades dos débitos com mais de 60% de retorno enquanto o gráfico da direita mostra a distribuição do principal do débito. Isso evidencia que existe grande quantidade de outliers.

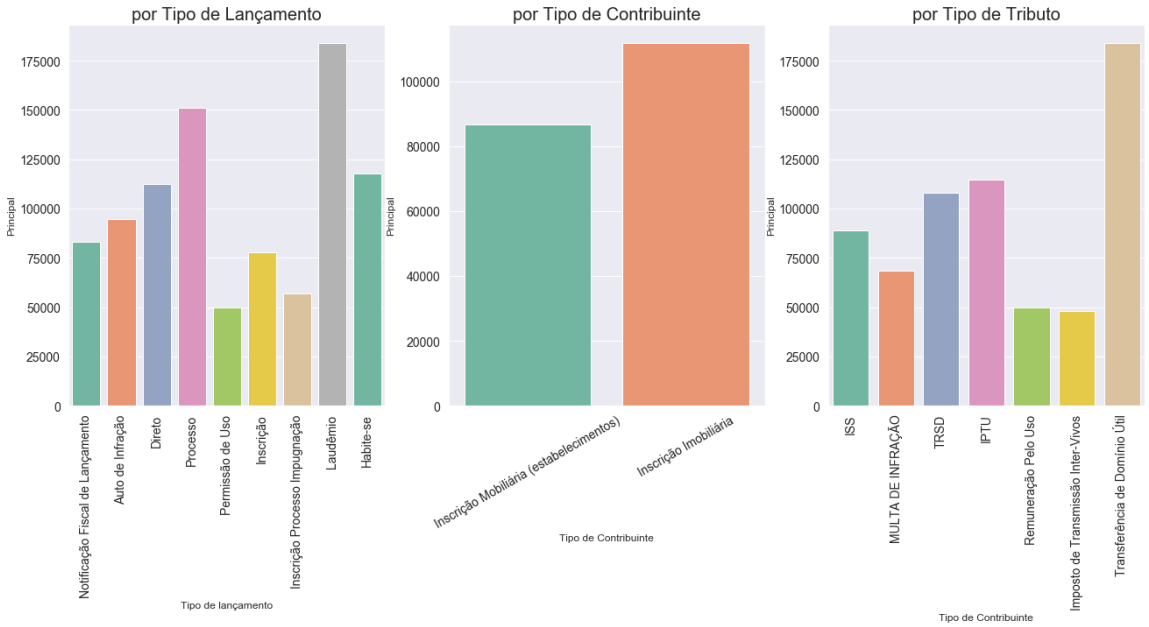


Para encontrar estes outliers foi usado um limite superior de média + alpha * desvio padrão. A ideia é que aumentando o alpha, o limite também aumenta, assim é possível plotar um gráfico para tentar encontrar um alpha ótimo para a equipe de cobrança:



Observa-se por exemplo que ao selecionar alpha=4 tem-se menos do que 400 contribuintes, representando uma quantia de 25 milhões de reais. O “cotovelo” no gráfico significa que não existe muita diferença em escolher um alpha maior que o “cotovelo”. A intenção dos gráficos é encontrar um ponto no qual a Fazenda possa despender o menor esforço para cobrar a maior quantia em dinheiro. A melhor estratégia ficará a cargo da área de negócio decidir.

Para ainda entender melhor o conjunto destes outliers pode ser plotado gráficos de barra para se decidir qual a melhor opção de cobrança. Esses gráficos irão ajudar a entender a relação de cada tipo de débito e contribuinte em relação ao principal. Para alpha=4:

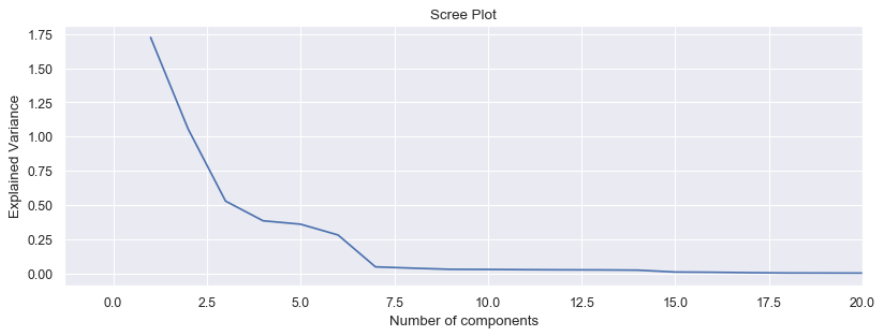


Similaridade entre aqueles que não pagam seus débitos

O objetivo é tentar entender os débitos com o menor índice de recuperabilidade e, talvez, a partir daí traçar uma estratégia para reverter a situação.

Para tanto foi utilizado um cluster para enquadrar os débitos em grupos de acordo com a sua similaridade. Só foi utilizado o conjunto dos débitos em aberto. Para diminuir a correlação existente com os percentuais encontrados, removi a variável “PRINCIPAL”, por se tratar da variável que mais contribuiu para encontrar as porcentagens.

Para encaixar o modelo de cluster as variáveis numéricas foram escaladas e foi utilizado o PCA para a seleção de características. Para encontrar o número de componentes foi utilizado um scree plot.



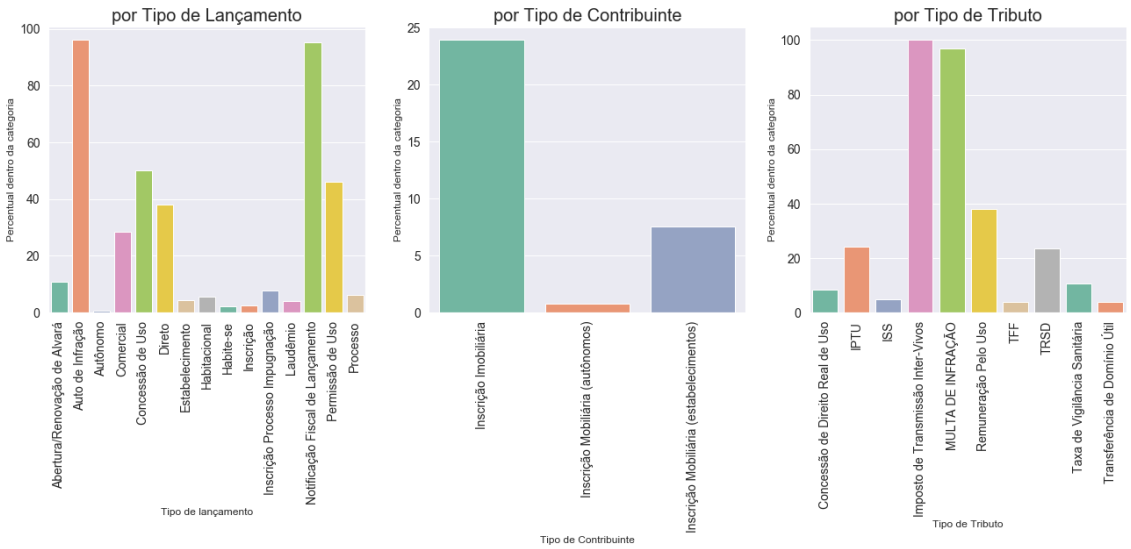
Assim, foi encontrado um valor de 7 para o número de componentes. O algoritmo escolhido foi o K-Means do sklearn, para encontrar o valor de K foi utilizado um gráfico que relaciona cada K com uma pontuação de silhueta:

Vê-se que a partir de 8 clusters o ganho de pontuação não se altera muito.

Após a aplicação do algoritmo de cluster, o conjunto de débitos foi agrupado conforme seu cluster. Abaixo segue uma tabela relacionando o cluster com a média da probabilidade de recuperação e a quantidade de contribuintes:

Cluster	Média da Probabilidade (%)	Quantidade de Contribuintes
3	4.82	104635
5	6.25	102136
4	15.5	100014
6	16.51	83726
0	22.4	86988
1	24.62	105686
2	66.19	142311
7	67.84	46460

Para analisar melhor o que acontece dentro de cada grupo foi utilizado gráficos de barras, porém o eixo y representando a porcentagem daquele valor na categoria. Por exemplo, se o valor de “Imposto de Transmissão Inter-Vivos“ está 100%, significa que dentre os débitos em aberto, 100 % dos registros deste imposto estão no cluster analisado. Abaixo segue o gráfico para o cluster 3:



A análise do gráfico mostra, por exemplo, que débitos advindos da ação dos auditores fiscais (geralmente resultam em Autos de Infração e Notificações fiscais de lançamento)

Estão com um índice de recuperabilidade baixíssimo. Talvez uma forma de resolver esse problema seja aumentando a qualidade do trabalho desses auditores.

Outra análise significativa é que 100 % dos débitos advindos de Imposto de Transmissão Inter-Vivos estão nesse cluster. Esse número indica que existe um problema no processo de cobrança desse imposto, indicando que talvez seja necessária uma reavaliação da legislação que regula este tributo.

Reflexão

Esse projeto atende a um anseio antigo da área de cobranças da SEFAZ, e trata-se de um projeto complexo. Uma das maiores dificuldades é que por se tratar de dados corporativos, a integração entre diversas pessoas e áreas é necessária, por exemplo, área de BI, área de negócio de cobrança, de arrecadação, de fiscalização e etc. Com certeza essa análise irá passar por mais mudanças até ser apresentada ao gabinete da SEFAZ. Fazer a gestão dessas integrações foi um desafio inédito em minha carreira de cientista de dados.

Outro desafio foi a quantidade de dados. Conciliar os recursos computacionais que estavam disponíveis para mim com a performance dos diversos algoritmos em um grande conjunto de dados foi uma das minhas prioridades no presente trabalho e mostrou-se como grande fonte de aprendizado.

Apesar dos desafios, os resultados alcançados no final foram bastante gratificantes. O trabalho mostrou uma real possibilidade da Fazenda recuperar mais rápido os seus débitos bem como revelou problemas em algumas formas de cobrança.

Melhorias

Na escolha de parâmetros

Certamente fazendo mais reuniões e entendo o problema mais pronfundamente, detalhes e variáveis podem surgir e facilitar ou melhorar a análise. Este é um trabalho que irá evoluir ao longo do tempo, observando os resultados das cobranças.

Incorporar esses resultados, bem como a forma como a cobrança foi feita (AR, telefone, email, protesto e etc) com certeza ajudará a alcançar resultados mais precisos.

Outros aspectos que podem ajudas são informações específicas do contribuinte como endereço, zona fiscal, patrimônio e etc. Essas informações tiveram que ser tiradas do trabalho por motivos de sigilo fiscal.

Outro aspecto seria incorporar anos anteriores e aumentar ainda mais o conjunto de dados.

Na construção do modelo

Testar mais algoritmos e também melhorar os recursos computacionais utilizados para realizar mais testes e validações.

A construção de modelos diferentes para cada tipo de contribuinte poderia trazer um ganho nas métricas de avaliação. Para isso também seria necessário trazer características mais específicas de cada tipo de contribuinte. Seria necessário entender

melhor o problema para escolher qual especificidade atacar primeiro (tipo de contribuinte, tipo de débito e etc).

Na análise dos resultados
Envolver mais áreas de negócio e pessoas para cada um trazer perguntas que possam ser respondidas com os dados.

VI. Referências bibliográficas

[1]. Forti, Melissa. **Técnicas de Machine Learning aplicadas na recuperação de crédito do mercado brasileiro**. 2018. Link: http://bibliotecadigital.fgv.br/dspace/bitstream/handle/10438/24653/Melissa_Forti_dissertacao.pdf?sequence=1&isAllowed=y

[2]. Abe, Naoki; Thomas, Vince; Kowalczyk et al. **Optimizing Debt Collections Using Constrained Reinforcement Learning.** 2010. http://www.cs.wayne.edu/~reddy/Papers/KDD10.pdf

[3]. Addo, Peter; Guegan, Dominique; Hassani, Bertrand. **Credit Risk Analysis Using Machine and Deep Learning Models.** https://www.mdpi.com/22279091/6/2/38/pdf

[4]. Galindo, Jorge & Tamayo, Pablo. (2000). **Credit Risk Assessment Using Statistical and Machine Learning: Basic Methodology and Risk Modeling Applications. Computational Economics.** 15. 107-43. 10.1023/A:1008699112516. https://www.researchgate.net/publication/5144412_Credit_Risk_Assessment_Using_Statistical_and_Machine_Learning_Basic_Methodology_and_Risk_Modeling_Applications

[5]**. Dumitrescu, Elena et al. Machine Learning for Credit Scoring: Improving Logistic Regression with Non Linear Decision Tree Effects**. 2017. https://editorialexpress.com/cgibin/conference/download.cgi?db_name=IAAE2018&paper_id=185