Processo Seletivo - Cientista de Dados 2018

Candidato: Túlio Vieira de Souza

Introdução

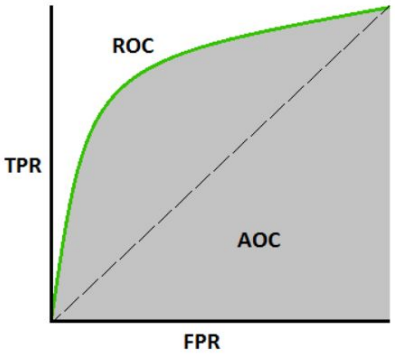
A conferência Knowledge Discovery and Data Mining, em 2009, ofertou uma competição onde foram disponibilizados dados da empresa de telecomunicações francesa Orange.

O objetivo da competição é criar um modelo para predizer o cancelamento da conta(churn), a tendência de usar novos produtos e serviços(appetency) e propensão a comprar novos produtos da companhia (upselling).

Para isto é fornecido um conjunto de dados referente a gestão de relacionamento dos clientes desta empresa, entretanto, por ser uma competição focada no aprendizado de máquinas, não temos condições entender a semântica das variáveis disponibilizadas, processo importantíssimo para formulação de hipóteses e construção do conhecimento.

A métrica utilizada para avaliação dos resultados escolhida é a AUC(Area Under Curve), que consiste na área formada abaixo da curva ROC (Receiver Operating Characteristics), perfeitamente adequados para se medir a performance de modelos em problemas de classificação.

Fonte: TowardsScience



A métrica AUC varia entre 0 e 1, sendo que quanto mais próximo de 1, melhor o modelo está classificando os dados, também é possível observar o desvio entre o que o modelo classifica verdadeiramente como positivo e o que ele classifica falsamente como positivo.

Banco de dados

O banco de dados consiste em 50.000 observações de 230 variáveis, sendo 192 numéricas e 38 categóricas.

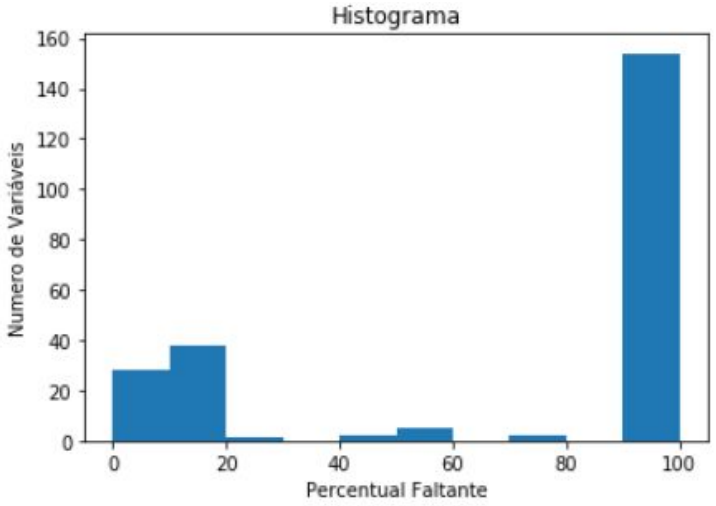
Um ponto de atenção foi a quantidade de casos faltantes no banco de dados, entre as 230 variáveis, somente 19 estavam com todos os dados.Para analisar mais a fundo, criamos

uma tabela contendo o percentual de itens faltantes de cada variável. Abaixo estatísticas descritivas sobre esta tabela:

	Percentual Missing
count	230.000000
mean	69.775235
std	41.549023
min	0.000000
25%	11.078000
50%	97.026000
75%	98.596000
max	100.000000

Em média, 69% das observações estão faltando banco de dados, já no segundo quartil observa-se variáveis com 97% de seus dados faltando.

Abaixo, plotamos o histograma relacionando o percentual faltante de acordo com o número de variáveis que se encontram nesta faixa.



Além da grande quantidade de variáveis com muitos dados faltantes, nota-se que em um agrupamento de variáveis falta apenas 20% ou menos de observações.

Foi observado que algumas variáveis categóricas contavam com mais de 5.000 diferentes tipos de categorias, o que pode ser ocasionado por diversos fatores.

Para pré processamento dos dados e criação de um primeiro modelo, foi realizada a seguinte modificação nos dados:

- Retirados as variáveis com mais de 20% dos dados faltando;

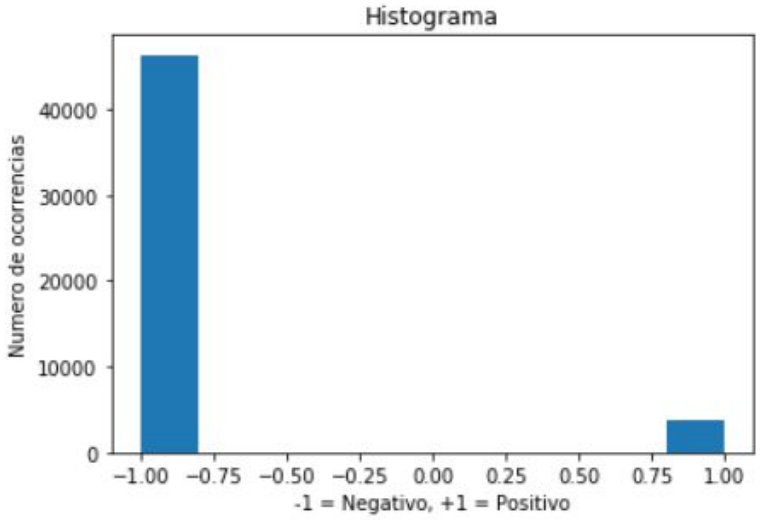
- Excluídos as variáveis categóricas;
- Transformado a variável do tipo Integer para Float;
- Preenchido as observações faltantes com as médias dos valores;

Com este fracionamento, pretendemos analisar primeiramente a parte numérica do problema, e com isso evoluir de um modelo mais simples para um mais complexo.

O banco de dados no qual iremos treinar o modelo tem 38 variáveis, todas elas do tipo float (números reais).

Churn

Ao analisar a label churn, o primeiro fato que percebemos foi o desbalanceamento dos dados, ilustrado através do histograma abaixo:



Quando o eixo X assume o valor de 1 estão representados os casos onde houve churn, -1 o oposto.

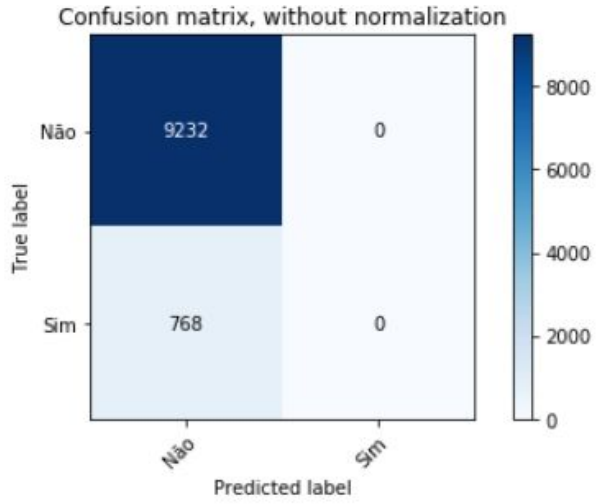
No primeiro momento foi treinado um modelo, sabendo do desbalanceamento dos dados, utilizando o framework tplot, que é construído sobre a biblioteca sklearn.

Dividimos os dados em treino, teste e validação, na seguinte proporção:

```
Treino: 36000
Teste: 10000
Validação: 4000
```

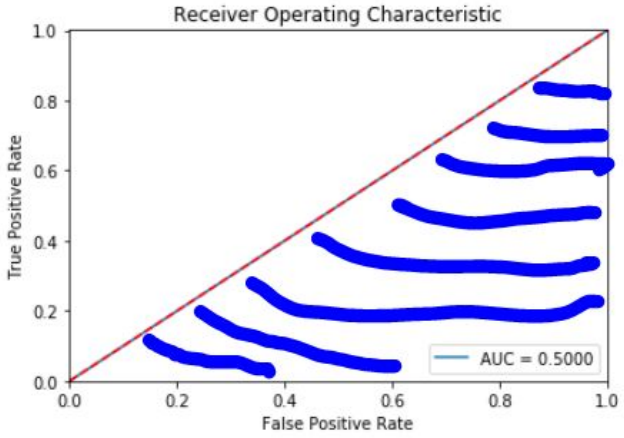
O algoritmo escolhido aplica aos dados diversas técnicas de machine learning e nos retorna a melhor configuração para um modelo inicial.

O resultado inicial foi o ExtraTreesClassifier, no entanto, com o modelo proposto treinado, observamos a matriz de confusão:



Nosso modelo fez a predição de todos resultados como se não houvesse churn, ou seja, nosso modelo ainda não é capaz de dizer se o cliente irá efetuar o cancelamento da conta.

Quando analisamos as métricas do modelo, enquanto o modelo está certo 92% das vezes (score) a área abaixo da curva AUC(hachurado em azul) é de 0.5000, assim observa-se que a métrica escolhida para analisar o problema é bastante pertinente.

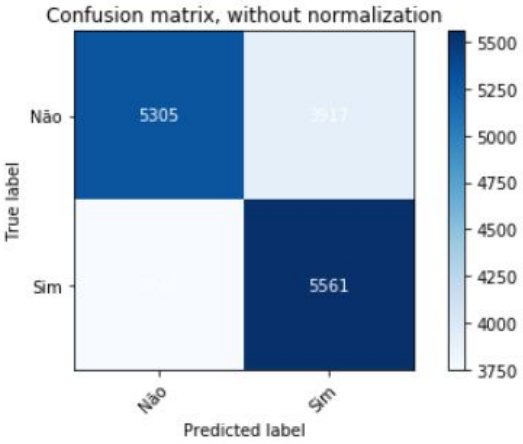


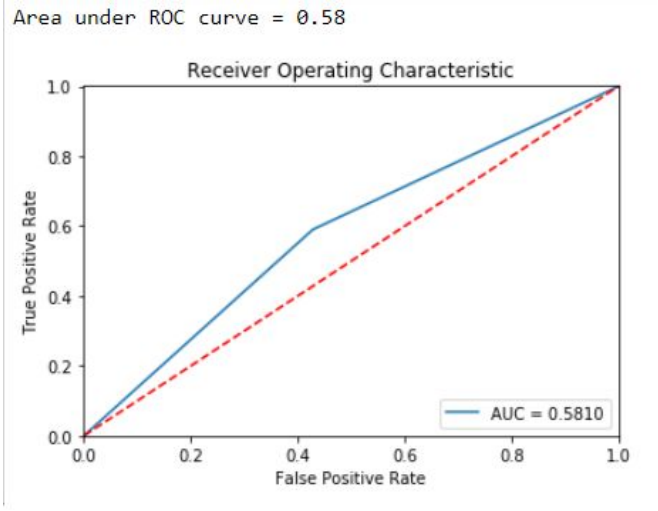
Utilizaremos de uma técnica conhecida como upsampling, que consiste em aumentar as observações da classe minoritária, replicando-os randomicamente a fim de fornecer mais exemplos sobre objeto a ser predito.

No primeiro momento replicamos os casos da classe minoritária a fim de igualar com a majoritária, a nova divisão em base de treino, teste e validação é:

```
Quantidade Observações:
Treino: 66711
Teste: 18532
Validação: 7413
```

Foi treinado um modelo utilizando as mesmas configurações do anterior atingindo os seguintes resultados:





	precision	recall	f1-score	support
-1	0.59	0.58	0.58	9222
1	0.59	0.60	0.59	9310
micro avg	0.59	0.59	0.59	18532
macro avg	0.59	0.59	0.59	18532
weighted avg	0.59	0.59	0.59	18532

Por mais que os resultados ainda estejam aquém do desejado, agora o modelo já é capaz de classificar se o cliente vai sair.

Appetency

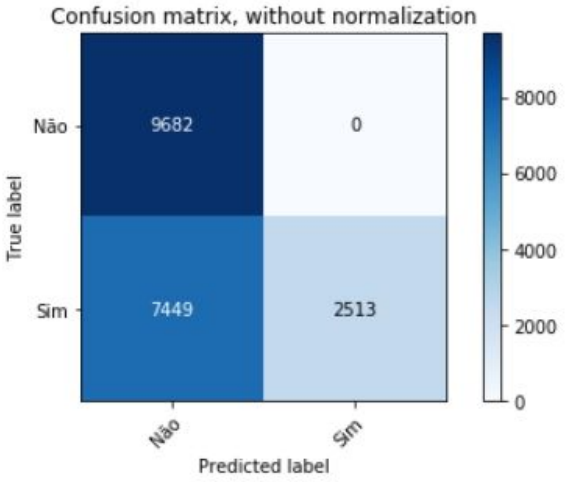
Uma vez que nós tratamos de três problemas de classificação que, dispensando o contexto, são da mesma natureza, o processo de construção do conhecimento será bastante parecido.

Com a restrição de tempo para realizar análises mais complexas, na predição da appetency e upsell, foram replicadas as técnicas descritas anteriormente, portanto serão reportados de forma mais sucinta.

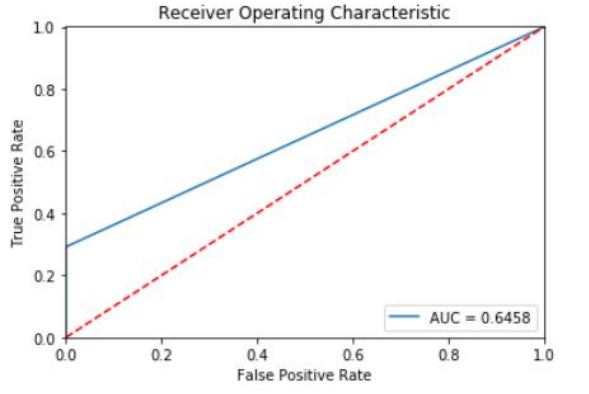
Após realizado o processo de upsampling e a divisão da base de treino, teste e validação para o problema foi a seguinte:

```
Quantidade Observações:
Treino: 70718
Teste: 19644
Validação: 7858
```

Desta vez, para predizer o appetency, será utilizado o RandomForestClassifier do sklearn, com os resultados:




```
Area under ROC curve = 0.65
```



	precision	recall	f1-score	support
-1	0.57	1.00	0.72	9682
1	1.00	0.25	0.40	9962
micro avg	0.62	0.62	0.62	19644
macro avg	0.78	0.63	0.56	19644
weighted avg	0.79	0.62	0.56	19644

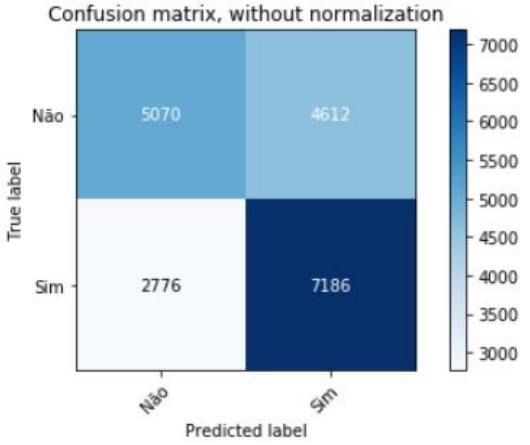
Upselling

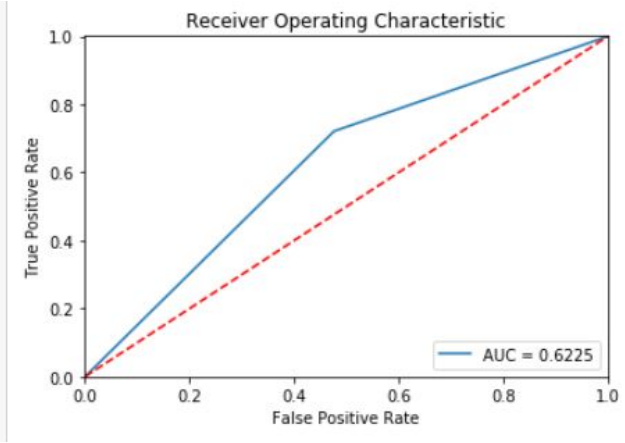
Para predizer o upselling, realizamos o processo de upsampling nos dados e utilizamos o algoritmo DecisionTreeClassifier do sklearn.

A divisão da base de treino, teste e validação é a seguinte:

```
Quantidade Observações:
Treino: 70718
Teste: 19644
Validação: 7858
```

Os resultados do modelo foram:





	precision	recall	f1-score	support
-1	0.65	0.52	0.58	9682
1	0.61	0.72	0.66	9962
micro avg	0.62	0.62	0.62	19644
macro avg	0.63	0.62	0.62	19644
weighted avg	0.63	0.62	0.62	19644

Estado da arte e conclusão

AUC Churn: 0.58
AUC Appetency: 0.65
AUC Upselling: 0.62

Estamos em um estágio inicial na construção do conhecimento através dos dados para o desafio apresentado.

Com os modelos treinados, conseguimos observar algumas características, entender quais variáveis estão contribuindo com a predição e o resultado apresentado pelos algoritmos utilizados.

Alguns tipos de modelos treinados apresentaram o problema de overfitting, configurado quando o modelo funciona muito bem para um propósito específico, mas não consegue generalizar para outras situações.

Para investigações futuras faz-se necessário:

- Limpar os dados contendo as variáveis categóricas e adicioná-los ao modelo;
- Entender se a superamostragem está causando overfitting;
- Testar diferentes frameworks;