Nanodegree Engenheiro de Machine Learning

Rebeca Andrade Baldomir

Junho 2018

1 Definição

1.1 Visão geral do projeto

Esse projeto irá resolver o problema de reserva de novos usuários do Airbnb (Airbnb New User Bookings), procurando prever em qual país esses novos usuários irão procurar reservas e, assim, o Airbnb pode proporcionar um conteúdo mais personalizado pro seus clientes e prever melhor a demanda. Os dados são divididos em 5 arquivos:

- train_users.csv: conjunto de treino dos usuários,
- test_users.csv: conjunto de teste dos usuários,
- sessions.csv: dados da sessões web dos usuários,
- countries.csv: estatística dos países de destino,
- age_gender_bkts.csv: estatística dos usuários em questão de idade, gênero e país de destino.

1.2 Descrição do problema

O objetivo desse notebook é prever qual o país de destino de um usuário da plataforma Airbnb, baseado em seus dados demográficos, registros de sessão da Web e algumas estatísticas de resumo. Todos os usuários desse conjunto de dados são dos EUA e podem ter 12 possíveis destinos: 'EUA', 'FR', 'CA', 'GB', 'ES', 'IT', 'PT', 'NL', 'DE', 'AU', 'NDF' e 'outro', sendo 'NDF' nenhum destino encontrado.

1.3 Métricas

A métrica de avaliação desses modelos que serão propostos é NCDG, normalized discounted cumulative gain, pois é como o kaggle classifica as submissões feitas. NCDG é calculado fazendo:

$$DCG_{\mathrm{k}} = \sum_{n=1}^{k} \frac{2^{\mathrm{rel_i}} - 1}{log_2(i+1)} \tag{1}$$

$$nDCG_{\mathrm{k}} = \frac{DCG_{\mathrm{k}}}{IDCG_{\mathrm{k}}} \tag{2}$$

Onde $\mathrm{rel_i}$ é a relevância do resultado na posição i e $\mathrm{IDCG_k}$ é o valor máximo possível de $\mathrm{DCG_k}$ para um determinado conjunto de consultas. Todos os cálculos de NDCG são valores relativos no intervalo de 0,0 a 1,0.

2 Análise

2.1 Exploração dos dados

As Figuras 1 e 2 são amostras da train_users.csv que foram utilizadas para treinar os modelos.

	id	date_account_created	timestamp_first_active	date_first_booking	gender	age	signup_method	signup_flow	language	affiliate_channel
0	gxn3p5htnn	2010-06-28	20090319043255	NaN	-unknown-	NaN	facebook	0	en	direct
1	820tgsjxq7	2011-05-25	20090523174809	NaN	MALE	38.0	facebook	0	en	seo
2	4ft3gnwmtx	2010-09-28	20090609231247	2010-08-02	FEMALE	56.0	basic	3	en	direct
3	bjjt8pjhuk	2011-12-05	20091031060129	2012-09-08	FEMALE	42.0	facebook	0	en	direct
4	87mebub9p4	2010-09-14	20091208061105	2010-02-18	-unknown-	41.0	basic	0	en	direct

Figure 1: Amostra do arquivo de treino dos algoritmos

affiliate_provider	first_affiliate_tracked	signup_app	first_device_type	first_browser	country_destination
direct	untracked	Web	Mac Desktop	Chrome	NDF
google	untracked	Web	Mac Desktop	Chrome	NDF
direct	untracked	Web	Windows Desktop	IE	US
direct	untracked	Web	Mac Desktop	Firefox	other
direct	untracked	Web	Mac Desktop	Chrome	US

Figure 2: Amostra do arquivo de treino dos algoritmos

A Figura 3 é uma amostra de algumas estatísticas do arquivo train_users.csv.

A Figura 4 é uma amostra do arquivo sessions,csv que contém estatísticas sobre os usuários.

Alguns dados nesse conjunto são desconhecidos -unknown- e outros são inconsistentes, como, por exemplo, valores de idade muito alto ou muito baixo. No caso dos valores desconhecidos, houve substituição por 0 e os valores inconsistentes foram removidos.

	timestamp_first_active	age	signup_flow
count	2.134510e+05	125461.000000	213451.000000
mean	2.013085e+13	49.668335	3.267387
std	9.253717e+09	155.666612	7.637707
min	2.009032e+13	1.000000	0.000000
25%	2.012123e+13	28.000000	0.000000
50%	2.013091e+13	34.000000	0.000000
75%	2.014031e+13	43.000000	0.000000
max	2.014063e+13	2014.000000	25.000000

Figure 3: Amostra das estatísticas do arquivo de treino dos algoritmos

	user_id	action	action_type	action_detail	device_type	secs_elapsed
0	d1mm9tcy42	lookup	NaN	NaN	Windows Desktop	319.0
1	d1mm9tcy42	search_results	click	view_search_results	Windows Desktop	67753.0
2	d1mm9tcy42	lookup	NaN	NaN	Windows Desktop	301.0
3	d1mm9tcy42	search_results	click	view_search_results	Windows Desktop	22141.0
4	d1mm9tcy42	lookup	NaN	NaN	Windows Desktop	435.0

Figure 4: Amostra do arquivo estatística dos usuários

2.2 Visualização exploratória

Nas visualizações abaixo é possível observar a quantidade de usuários por gênero e também a quantidade de vezes que cada país foi escolhido como destino pelos os usuários.

Em ambas as visualizações é possível verificar que a maior parte dos dados é de informações desconhecidas ou inconsistentes, NDF e NaN, respectivamente. Esse fator pode dificultar um pouco o nosso algoritmo a alcançar bons resultados.

2.3 Algoritmos e técnicas

Esse é um problema multi-classe e será resolvido utilizando algoritmos de classificação, como árvores de decisão, naive bayes para modelos de Bernoulli, gaussian naive bayes, k-nearest neighbors e logistic regression.

Para o algoritmo k-nearest neighbors foi utilizado 3 vizinho como parâmetro, pois foi o que alcançou melhor pontuação, os demais seguiram com a configuração default.

Vamos falar mais porque foi utilizado cada um desses algoritmos, os seus

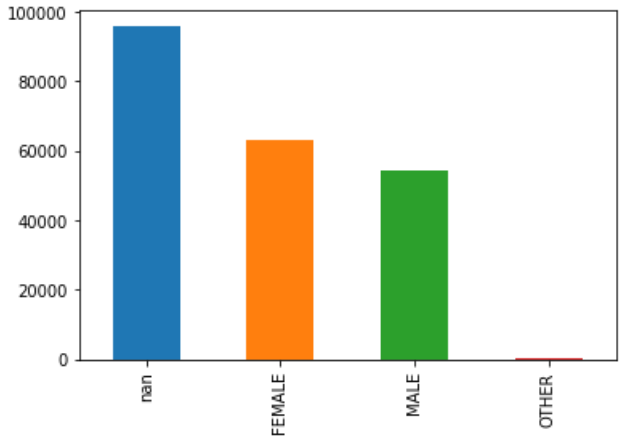

Figure 5: Quantidade de usuários por gênero.

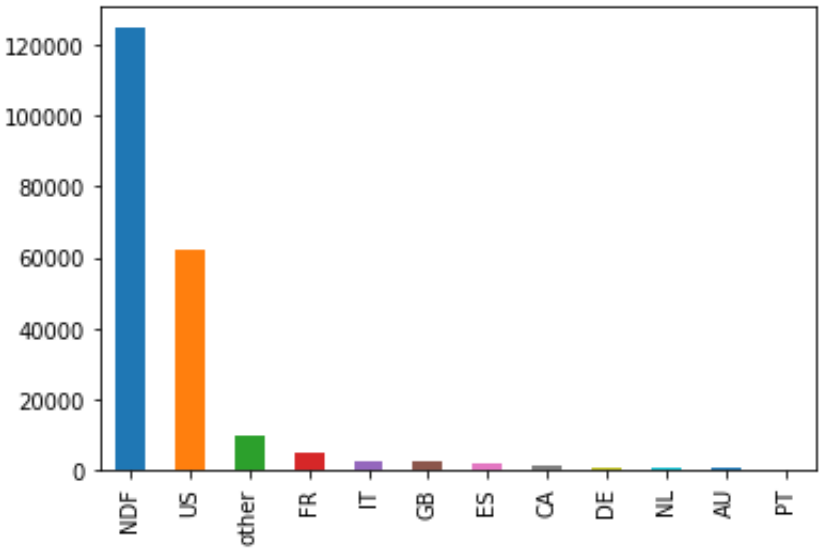

Figure 6: Quantidade de vezes que cada país foi escolhido como destino pelos os usuários.

prós e contras e o porquê dele ser ideal para o problema.

1. Árvores de Decisão

   No caso do nosso problema, cada país seria uma classe, então os algorit-

mos utilizados devem conseguir classificar os dados em mais de uma classe, ou seja, deve ser multiclasse. Um dos algoritmos que usamos foi o DecisionTreeClassifier que é um calssificador capaz de executar classificação de várias classes em um conjunto de dados. Esse algoritmo requer pouca preparação de dados, tem um custo logarítmico no número de pontos de dados usados para treinar a árvore e funciona bem, mesmo que suas suposições sejam de algum modo violadas pelo verdadeiro modelo do qual os dados foram gerados. No entanto, pode criar árvores super complexas que não generalizam bem os dados e as árvores de decisão podem ficar instáveis porque pequenas variações nos dados podem resultar na geração de uma árvore completamente diferente. Fonte: Sklearn - Decision Trees

2. Gaussian Naive Bayes e Naive Bayes para Modelos de Bernoulli

   Os classificadores Naive Bayes estão trabalhando com base no teorema de Bayes, que descreve a probabilidade de um evento, com base no conhecimento prévio das condições, estar relacionado às condições do evento. É um classificador muito simples e rápido e às vezes funciona muito bem, e mesmo sem muito esforço você consegue uma precisão perfeita. Fonte: Medium - Gaussian Naive Bayes

3. K-Nearest Neighbors

   O princípio por trás dos métodos de vizinho mais próximo é encontrar um número pré-definido de amostras de treinamento mais próximas da distância do novo ponto e prever o rótulo a partir delas. No nosso caso, o número de amostras foi 3. A distância pode, em geral, ser qualquer medida métrica: a distância euclidiana padrão é a escolha mais comum. Esse algoritmo é flexível para opções de recursos / distância, lida naturalmente com casos de várias classes e, geralmente, se sai bem com dados suficientemente representativos. No entanto, tem um custo computacional alto para encontrar vizinhos mais próximos e deve saber que temos uma função de distância significativa que vai trazer bons resultados. Fonte: Nearest Neighbours: Pros and Cons, Sklearn - Nearest Neighbors Classification

4. Logistic Regression

   A regressão logística é um modelo linear de classificação onde as probabilidades que descrevem os resultados possíveis de um único teste são modeladas usando uma função logística. A regressão linear é um método extremamente simples. É muito fácil e intuitivo de usar e entender, além disso, funciona na maioria dos casos. No entanto, a regressão linear é muito sensível às anomalias nos dados e assume que há uma relação linear entre eles, o que é incorreto às vezes. Fonte: Sklearn - Linear Regression, What are the advantages and disadvantages of linear regression?

2.4 Benchmark

O benchmark a ser utilizado como referência está no leaderboard do kaggle que está entre 0.881 e 0.882 entre os 10 primeiros participantes.

3 Metodologia

3.1 Pré-processamento de dados

O pré-processamento dos dados consistiu na extração das labels do conjunto de treino e na transformação dos dados em dados numéricos utilizando o LabelEncoder para poder utilizar os algoritmos escolhidos.

3.2 Refinamento

Após a aplicação de todos os algoritmos selecionados, foi verificado o score de cada algoritmo aplicado ao conjunto de dados (Tabela 1) e o algoritmo que apresentou a melhor score foi o DecisionTreeClassifier, então ele foi utilizado para fazer as predições com o conjunto de teste.

Table 1: Pontuação dos Algoritmos

Algoritmo	**Score**
GaussianNB	0.8731
DecisionTreeClassifier	1.0
BernoulliNB	0.8757
KNeighborsClassifier	0.8840
LogisticRegression	0.8756

4 Resultados

4.1 Modelo de avaliação e validação

É possível visualizar a quantidade de vezes que cada país foi escolhido como destino dos usuários na Figura 7 de acordo com o resulto da aplicação do algoritmo DecisionTreeClassifier. Essa predição alcançou uma pontuação de ndcg_score de 81%.

Podemos visualizar também qual seria o melhor estimador utizando o GridSearchCV que seria:

```
DecisionTreeRegressor(criterion='mse', max_depth=1, max_features=None,
           max_leaf_nodes=None, min_impurity_decrease=0.0,
           min_impurity_split=None, min_samples_leaf=1,
           min_samples_split=2, min_weight_fraction_leaf=0.0,
           presort=False, random_state=None, splitter='best')
```

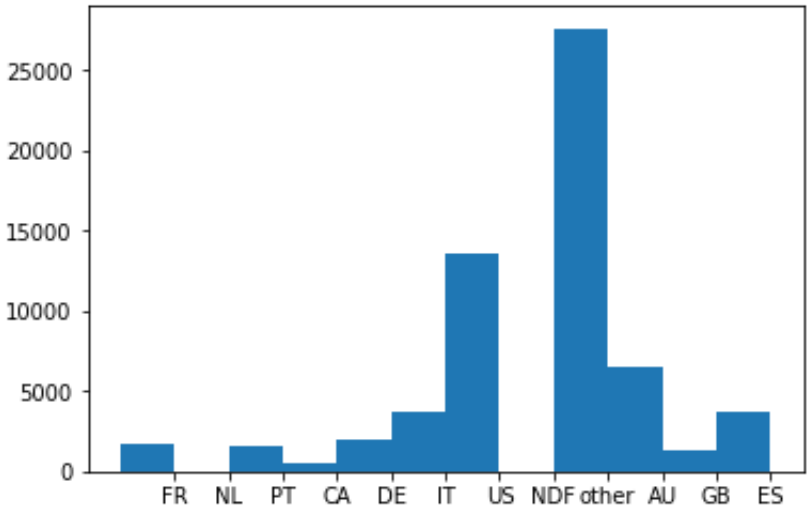

Figure 7: Quantidade de vezes que cada país foi escolhido

4.2 Justificativa

O resultado encontrado possui uma pontuação inferior ao modelo de referência que é de 0.886, mas ainda assim é uma pontuação satisfatória para resolução do problema.

5 Conclusão

5.1 Reflexão

Para realizar esse projeto, começamos entendendo os dados para dar início a limpeza de dados nulos e inconsistentes. A partir daí foi feita uma visualização para entender as características desses dados. Com os dados entendidos, é a hora da escolha dos algoritmos e, assim, foi observada o desempenho de cada um dos algoritmos pré-selecionados nesse conjunto de dados. Finalizada essa etapa, foram feitas as predições e comparação com o benchmark.

Um aspecto muito difícil, não só nesse projeto, é o entendimento e a limpeza dos dados. A escolha e a decisão de quais dados podem ser inconsistentes não é trivial, pois depende muito do domínio de aplicação do problema. Apesar das dificuldades, o modelo e resultado final ficaram alinhados com a minha expectativa para o problema.

5.2 Melhorias

A maioria dos algoritmos aplicados usaram a configuração default dos algoritmos, então várias melhorias poderiam ser feitas no modelo desenvolvidos, mas uma melhoria importante pode ser alteração dos parâmetros dos algoritmos aplicados para treinar os modelos para buscar um melhor desempenho no resultado da predição dos dados.