Sumário

1 Introdução

2 Introduction

O advento da era da informação revolucionou a indústria hoteleira à medida que os clientes da hotelaria são contemplados com um leque abrangente de escolhas de reversas de quartos de hotel. Contudo, esse aumento de escolhas para os clientes, implica no aumento de cancelamento de reservas de hotel, já que, por exemplo, um cliente, por receber uma oferta mais atrativa de um concorrente, pode cancelar a sua reserva. Diante disso, o presente trabalho propõe métodos de aprendizado supervisionado para prever se um cliente irá cancelar a reserva de um hotel. Em especial, isso é feito supondo os dois cenários, sendo o primeiro:é importante explicar as predições do modelo e o segundo: a predição é mais importante em relação à interpretação dos resultados.

2.1 Aprendizado supervisionado aplicada ao contexto de prever o cancelamento do hotel

O problema de prever o cancelamento de um agendamento de hotel pode ser visto da seguinte forma: dada uma base de dados, com exemplos rotulados como agendamento cancelado ou agendamento não cancelados, divida a base em um conjunto de treinamento e de teste, utilize a base de treinamento para treinar um modelo que ira fornecer a probabilidade estimada de cancelamento $\hat{p} = f(X_{train})$ e então, na base de teste, aplique a $\hat{p}$ aos seus atributos e verifique a desempenho comparando com os rótulos conhecidos.

Com objetivo de avaliar o desempenho dos modelos convém definir a matriz de confusão conforme:

VERDADE/PREDITO	Cancelado	Não cancelado
Cancelado	VP	FP
Não Cancelado	FN	VN

E partindo dessa matriz é possível definir métricas de desempenho. Em especial, nesse trabalho serão utilizadas as seguintes métricas: acurácia balanceada, recallrecall,precision e área sobre a curva precision recall. Note que essas métricas fogem da tradicional acurácia/área curva ROC, isso deve-se ao fato de que a base de dados a ser analisade tem desbalanceamento nos rótulos e ,então, é preciso utilizar métricas adequadas. Ademais, é fato que todo modelo tem um viés, por isso, na fase de avaliação das predições, é preciso adotar estratégicas que reduzem o viés da avaliação, nesse sentido será empregado a técnica de validação cruzada k-fold. Nesse ínterim, o ténica de k-fold crossvalidation divide a base dados em k conjuntos de treino/teste e para cada um dessas divisões um modelo é treinado e seu desempenho é avaliado no conjunto de teste, sendo que o resultado é um vetor de "k-métricas", para que possamos interpretar o resultado do k-fold foi proposto resumuir esse vetor reportando sua variância e sua média.

2.2 A troca entre precision-recall

Precision é calculado da seguinte forma $\frac{VP}{TP+FP}$ e recal da seguinte forma: $\frac{VP}{TP+FN}$. Maximizar precision envolve também minimizar a taxa de Falsos positivos, isso é reservas que foram canceladas, mas o modelo previu como não cancelada e maximizar recall envolve minimizar o falso negativo, isso é reservas que não foram canceladas, mas o modelo previu como cancelada. A escolha da métrica para maximizar depende do contexto de negócio. Como será visto posteriormente, os clientes que fazem

reserva online tendem a ter maior probabilidade de cancelar, nesse sentido, uma proposta de negócio é utilizar um modelo de predição para prever a probabilidade de cancelamento da reserva e,então,para aqueles com alta probabilidade., enivar e-mails , propagandas direcionadas, novas propostas, etc. Nesse contexto é melhor maximizar a precisão, já que o objetivo é "mandar muitos e-mails para quem vai cancelar" , isso é : desejam-se poucos falsos positivos , mas também pode ser irritante se um cliente que não cancelar receber muitos e-mails, então busca-se maximizar a precisão, mas também ter um recall decscente.

Uma boa métrica é a área sobre a curva precison-recall que é implementada pela função avarage-precision, visto em **??** essa é uma boa métrica para maximizar a precision sem deixar de lado o recall.

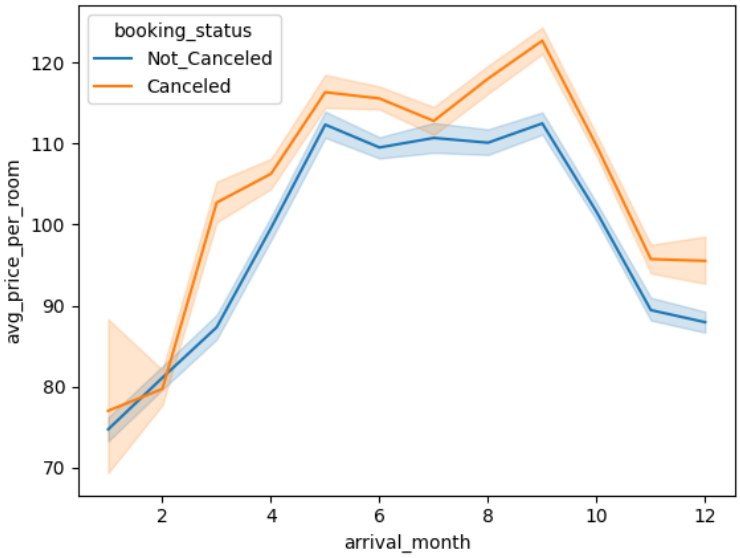
3 Análise exploratória

3.1 Considerações sobre a base de dados

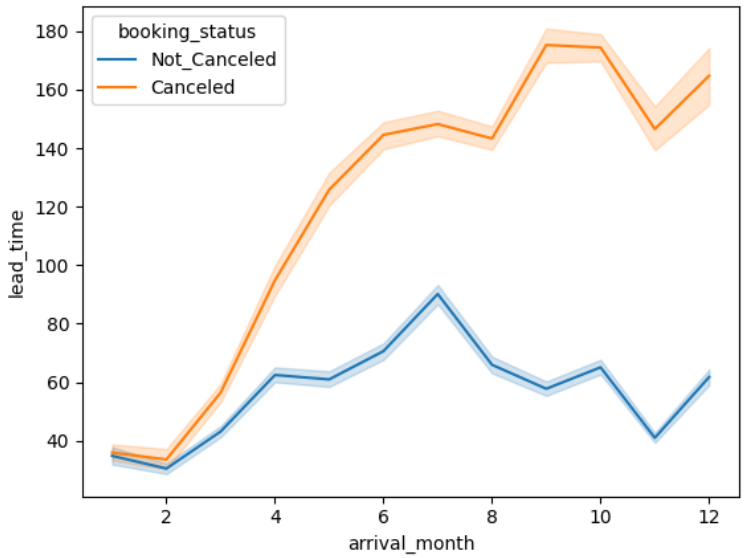
A base de dados a ser analisa possui um desbalanceamento entre as classes, pois há menos exemplos de cancelamento. Além disso, há atributos numéricos e qualitativos. Os atributos numéricos não estão correlacionados, sendo que a maior correlação de pearson encontrada foi entre preço médio da reserva e número de crianças. Além disso, não há valores nulos.

3.2 A correlação temporal

O gráfico a seguir mostra que o preço das reservas é maior entre os meses 4 e 10e há mais cancelamentos nesses meses;

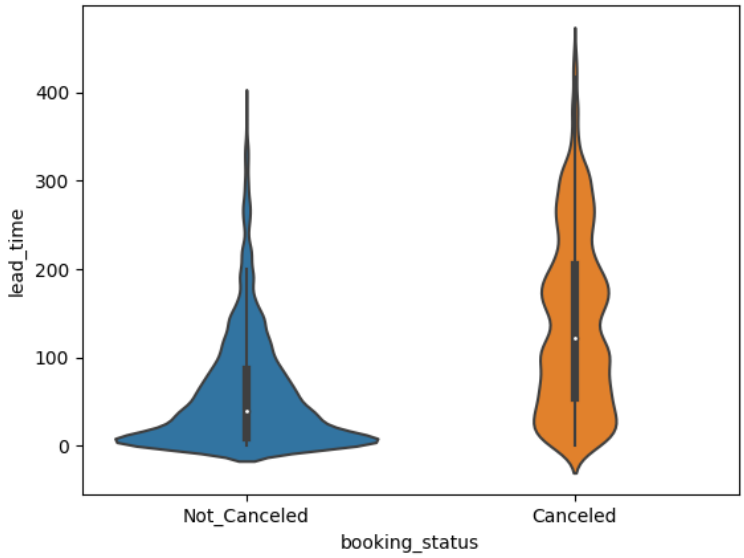


O gráfico a seguir mostra que o tempo de espera é maior no final do ano e os cancelamentos aumentam nesse período.



3.3 variáveis qualitativas

Para investigar a relação de tempo de espera e o cancelamento foi feito um gráfico de violino e pode-se observar que a distribuição do tempo de espera dado o cancelamento da reserva é diferente do tempo de espera dado o não cancelamento



Já a distribuição do preço médio tem calda mais pesada para os pedidos não cancelados, possui estatística de máximo maior e quantis à direita maior, isso indica que há mais pedidos cancelados quando o preço é muito alto.

No que diz respeito ao canal de marketing em que foi feito a reserva, há uma tendência maior de cancelamento nas reservas feita online e uma tendência de não cancelamento nas reservas feita por empresas

Quanto ao tipo de plano de refeição escolhido, notou-se que o tipo 3 é muito pouco escolhido, e o tipo 2 é mais escolhido entre as reservas canceladas.

Não há exemplos de canal de marketing do tipo Complementary para os cancelados, há poucos exemplos de tipo de refeição 3 , os tipos de quartos 7,5,3 possuem poucos exemplos dentre as observações da

base de dados

4 Baseline

Arvores de decisão, k vizinhos mais próximos, regressão logística, floresta aleatórias: esses são alguns dos algoritmos de aprendizado supervisionado mais comuns, mas qual deles devo aplicar ? Para responder essa pergunta é útil, desenvolver uma "baseline" de modelos, essa "baseline" consiste em verificar o desempenho de vários modelos diferentes. O desenvolvimento da "baseline" é extremante útil, pois serve de guia para futuros modelos e pode ser comparada com futuros modelos mais complexos.

Para desenvolver a "baseline" foi adotado os seguintes passos :

1. Transformam-se atributos qualitativos em numéricos usando ONE HOT ENCONDING
2. Normalizam-se os atributos para terem média 0 e variância 1
3. Selecionamos os modelos de aprendizado mais comuns
4. Para cada modelo realizamos k-fold Cross-Validation
5. Armazenamos a média e o desvio padrão obtidos no k-fold Cross-Validation

O resultado é o exibido a seguir:

index	balaneced_acc	recall	precision	average_precision
KNeighborsClassifier	0.8156 +/- 0.0109	0.7244 +/- 0.0223	0.791 +/- 0.0077	0.804 +/- 0.0141
GaussianNB	0.5447 +/- 0.007	0.9707 +/- 0.0106	0.3493 +/- 0.0038	0.6045 +/- 0.0093
LogisticRegression	0.7513 +/- 0.0179	0.6091 +/- 0.0361	0.7355 +/- 0.012	0.7546 +/- 0.0276
SVC	0.7668 +/- 0.0159	0.6042 +/- 0.034	0.8064 +/- 0.011	0.8047 +/- 0.0158
DecisionTreeClassifier	0.8369 +/- 0.0121	0.7788 +/- 0.0209	0.7832 +/- 0.0105	0.6917 +/- 0.0169
RandomForestClassifier	0.8587 +/- 0.0048	0.7819 +/- 0.0081	0.8552 +/- 0.0056	0.9094 +/- 0.006
BaggingClassifier	0.8498 +/- 0.0062	0.7683 +/- 0.0124	0.8451 +/- 0.004	0.8753 +/- 0.0086
AdaBoostClassifier	0.7734 +/- 0.0119	0.6569 +/- 0.0249	0.744 +/- 0.009	0.7891 +/- 0.022

Note que os modelos baseados em árvore obtiveram melhor desempenho, em especial a random florest foi o modelo que mais perfomou em todas as métricas,com exceção do recall. O naive-bayes teve esse comportamento "bizarro" de ter muito recall devido a classes poucos frequentes que foram relatadas na seção de análise exploatória 3. Essa baseline indica que é muito conveniente explorar ainda mais os modelos boseados em árvores.

Um pequeno adendo, o problema da separação total ou quase-total é conhecido nos modelos de regressão logística, embora o algoritmo proposto pelo pacote sklearn convirja, os coeficientes da regressão logística terão erro-padrão muito alto. Note que nem todos pacotes de aprendizado de máquina emitem algum aviso para esse problema e,ao menos a primeira vista, olhando a vaseline, parece que o modelo de regressão logística performou razoavelmente bem.

5 Árvore de decisão

Conforme visto em 4 o modelo de árvore de decisão não obteve desempenho muito alto, isso deve-se ao fato de que esse modelo é sensível à escolha de hyper-paâmetros, que são aqueles parâmetros

definidos pelo usuário, como,por exemplo, a profundidade da árvore. Sendo assim, com fito de aprimorar a capacidade desse modelo será utilizado a otimização bayesiana para selecionar os hyper-parâmetros, que é implelementado em python por meio da biblioteca hyperopt [Bergstra et al., 2013] . Além disso, já que a base de dados está desbalanceada, foi proposta a aplicação do SMOTE que é um algoritmo de balanceamento artificial do conjunto de treinamento.

Antes de analisamos os resultados ressaltamos alguns pontos

1. A seleção de hyper-parâmetros é realizada na base de treinamento.

2. O balanceamento das classes deve ser aplicado somente na base de treinamento,

3. como o SMOTE utiliza o kNN em seu algoritmo é, conveniente normalizar a base de dados

Por fim, eis o resultado:

index	balaneced_acc	recall	precision	average_precision
DecisionTreeClassifier	0.8369 +/- 0.0121	0.7788 +/- 0.0209	0.7832 +/- 0.0105	0.6917 +/- 0.0169
DCT + Hyperopt	0.8045 +/- 0.0095	0.6971 +/- 0.0283	0.7949 +/- 0.0128	0.8345 +/- 0.0073
DCT + SMOTE + Hyperopt	0.8264 +/- 0.0057	0.7738 +/- 0.0063	0.7571 +/- 0.0111	0.8321 +/- 0.0103

Note que, em relação aos hyper-parâmetros padrões, observa-se um grande aumento em todas as métricas de desempenho. Além disso, a combinação de SMOTE + otimização de hyper-parâmetros obteve recall com média maior e desvio padrão menor e acurácia balanceada maior, isso mostra como o balanceamento artificiais dos dados impacta na relação precisão-recall.

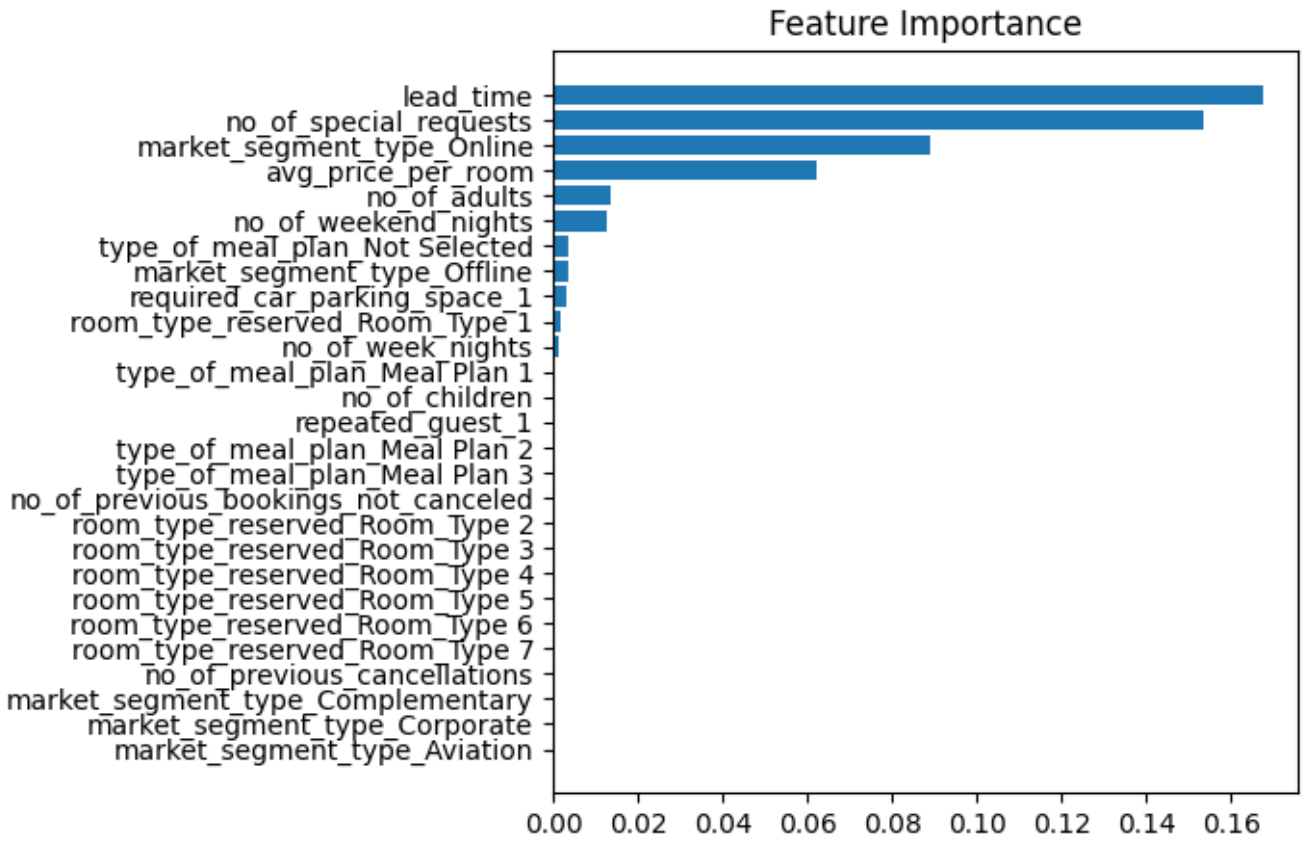
Uma vantagem das árvores de decisão é que elas são interpretáveis, pois para explicar a predição de um exemplo basta achar o seu caminho até o nó folha. Contudo, devido à profundidade da árvore e prezando pela legibilidade do artigo não iremos exibir a imagem da árvore completa, embora no código disponível em anexo há uma implementação que mostra a árvore até um determinado nível de profundidade.

Além disso, pode-se aproveitar da estrutura das de decisão para das árvores um gráfico de importância de atributos exbido a seguir :

Nesse gráfico nota-se que a variável lead time (tempo de espera) é importante par predição do cancelamento da reserva, ou seja, se o time de negócios do hotel está interessado em diminuir os cancelamentos de reserva é interessante que sejam implementados métodos para diminuir esse tempo de espera. Outrossim, a variável avg price per room (preço médio por quarto) e a variável market segment type online (reservas feitas pela internet) , também são importantes para predição. Note que o primeiro corrobora os resultados da análise exploratória e o segundo corrobora as ponderação feita na introdução acerca do impacto da internet no setor de hoteleiro.

6 Random Florest (Floresta aleátoria)

Já foi visto que é possível obter bom desempenho utilizando as árvores de decisões, mas e se se juntássemos o resultado de várias árvores de decisões em um modelo? LEO BREIMAN estudou esse problema e,no artigo [Breiman, 2001], desenvolveu o modelo de florestas de aleatórias. Conforme visto na baseline 4, o modelo de floresta aleatória proposto por BREIMAN já apresentou ótimo desempenho, contudo, como a base de dados é desbalanceada, foram testadas algumas alternativas para tentar melhorar o desempenho da random florest, sendo ela : aplicar o SMOTE ([Chawla et al., 2002]) e uma adaptação



da floresta aleatória para base de dados desbalanceadas a imbalanced random florest proposta no trabalho [Chen et al., 2004]

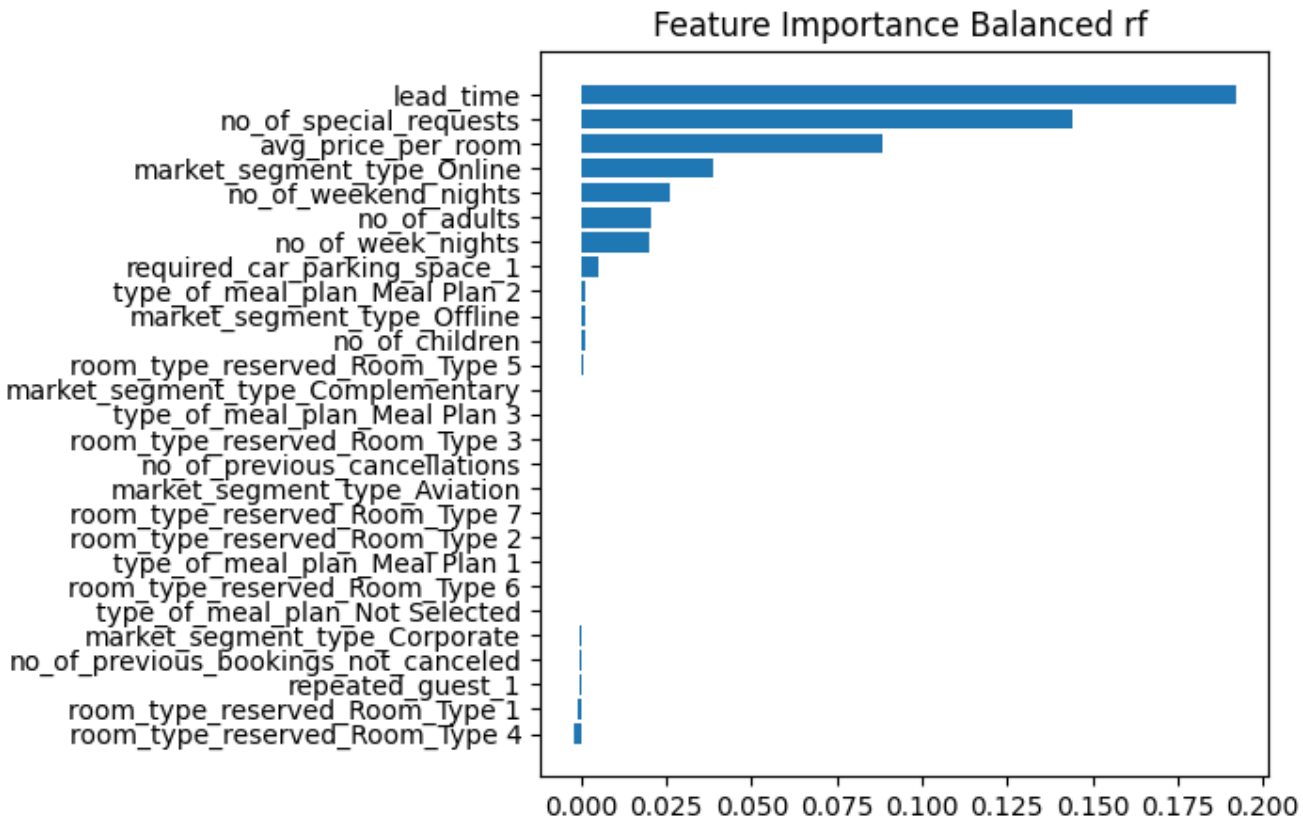
No que diz respeito ao pré-processamento de dados, foi realizado one-hot-encoding em ambas estratégias e na estrategia de SMOTE + Randon Florest foi realizado a normalização dos dados (Lembrar que smote usa o knn em seu algoritmo) . Finalmente, uma vez pré-processados, pode-se aplicar os modelos aos dados, então, eis o resultados:

index	balaneced_acc	recall	precision	average_precision
RandomForestClassifier	0.8587 +/- 0.0048	0.7819 +/- 0.0081	0.8552 +/- 0.0056	0.9094 +/- 0.006
RF + SMOTE	0.8639 +/- 0.0025	0.8184 +/- 0.0044	0.8149 +/- 0.0065	0.9035 +/- 0.007
balanced_rf	0.8687 +/- 0.0022	0.857 +/- 0.0044	0.7773 +/- 0.0069	0.9105 +/- 0.0056

Note que ambas as estratégias tiveram melhor desempenho na métrica de acurácia balanceada e recall. Note também, que, entre as duas estratégias propostas a floresta aleatória balanceada teve melhor desempenho em todas as métricas, com exceção da precisão .

Os modelos de floresta aleatórias não são interpretáveis,como a árvore de decisão.Todavia, é possível obter insights sobre suas predições utilizando os gráficos de importância de atributos. A seguir apresentamos o gráfico de importância de atributo do imabalanced randon florest (No códogo em anexo há os dois gráficos , mas não foram encontradas diferença entre eles)

Além disso, pode-se aproveitar da estrutura das de decisão para das árvores um gráfico de importância de atributos exbido a seguir :



7 Extreme Gradient Boosting (XGBoost)

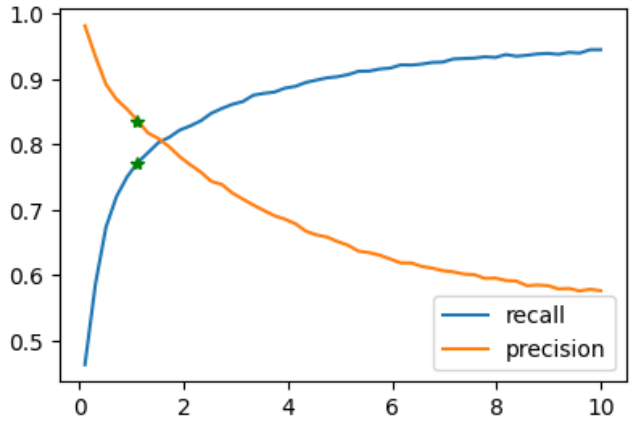
O extreme gradient Extreme Gradient Boosting (XGBoost) é um algoritmo de aprendizado de máquina que vem ganhando relevância devido ao seu desempenho em competições de ciência de dados. O seu funcionamento é baseado na minimização iterativa de uma função de perda, a principal diferença entre o XGboost é que o random florest é um algoritmo de bagging, enquanto o XGBoost é um algoritmo baseado em gradient boosting, uma ótima explicação do XGBoost pode ser encontrada Playlist do youtube e no artigo [Chen and Guestrin, 2016].

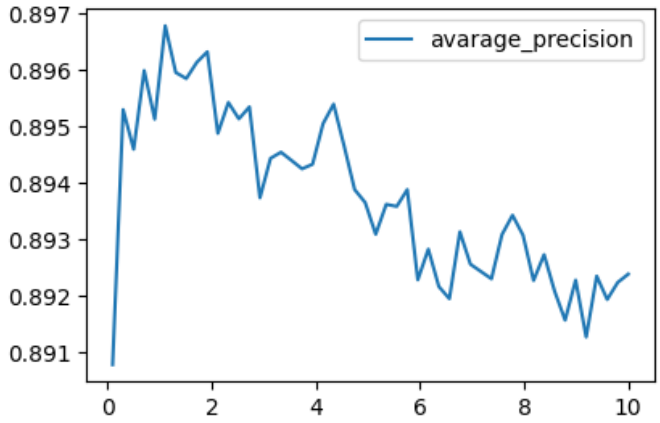
Nesse trabalho foram propostas três abordagens: o XGBoost com hyper-parâmetros padrões e o SMOTE + XGBoost. O esquema de pré-processamento é o mesmo utilizado nas florestas aleatórias. Além disso, o XGBoost tem o hyper-parâmetro 'scale-pos-weight' que ajusta o peso dado as observações da classe positiva, como a poucos exemplos da classe positiva (poucos cancelamentos) foi também explora o impactado do scale-pos-weight na relação precisão-recall.

Para estudar o impacto foi realizado o seguinte esquema :

1. Geramos valores 50 entre 0.1 e 10
2. Para valor realizamos 5-fold cross validation com o scale pos weight configurado para esse valor.
3. armazenamos em um vetor as médias das métricas de interesse
4. plotamos dois gráficos, um com a avarage precision(a área estimada sobre curva precision-recall) e outro com duas curvas, sendo uma do recall e outra do precision.

O gráfico é exibido a seguir:





Esse gráfico mostra claramente como a escolha do scale pos weight impacta nas métricas de precison/recall . O ponto verde no gráfico indica o valor máximo da avarage precision , note que isso indica que de fato a métrica avarage precision é uma boa métrica para aqueles que estão interessados na precisão, mas se preocupam também com o recall.

A seguir é exibido o resultado obtido por : XGBOOST padrão , XGBOOST com scale pos weight configurado e XGBoost + smote.

index	balaneced_acc	recall	precision	average_precision
xgboost	0.8503 +/- 0.0055	0.7649 +/- 0.0112	0.8529 +/- 0.0039	0.9026 +/- 0.0036
xgboost + scale_pos_weight	0.8534 +/- 0.0046	0.7756 +/- 0.0104	0.8461 +/- 0.0029	0.9031 +/- 0.0028
XGB + SMOTE	0.8561 +/- 0.0025	0.811 +/- 0.0065	0.7999 +/- 0.0037	0.8987 +/- 0.0028

Em ambos os modelos as métricas são muito semelhantes, com excessão do SMOTE + XGBOOST que tem maior recall e menor precisão . Nesse caso, o scale pos weight obtido pela simulação é muito próximo ao padrão do xgboost por isso não houve muita diferença entre eles.

a seguir exibimos um gráfico de importância de atributos para o XGBoost

Outr proposta foi fazer uma combinação de

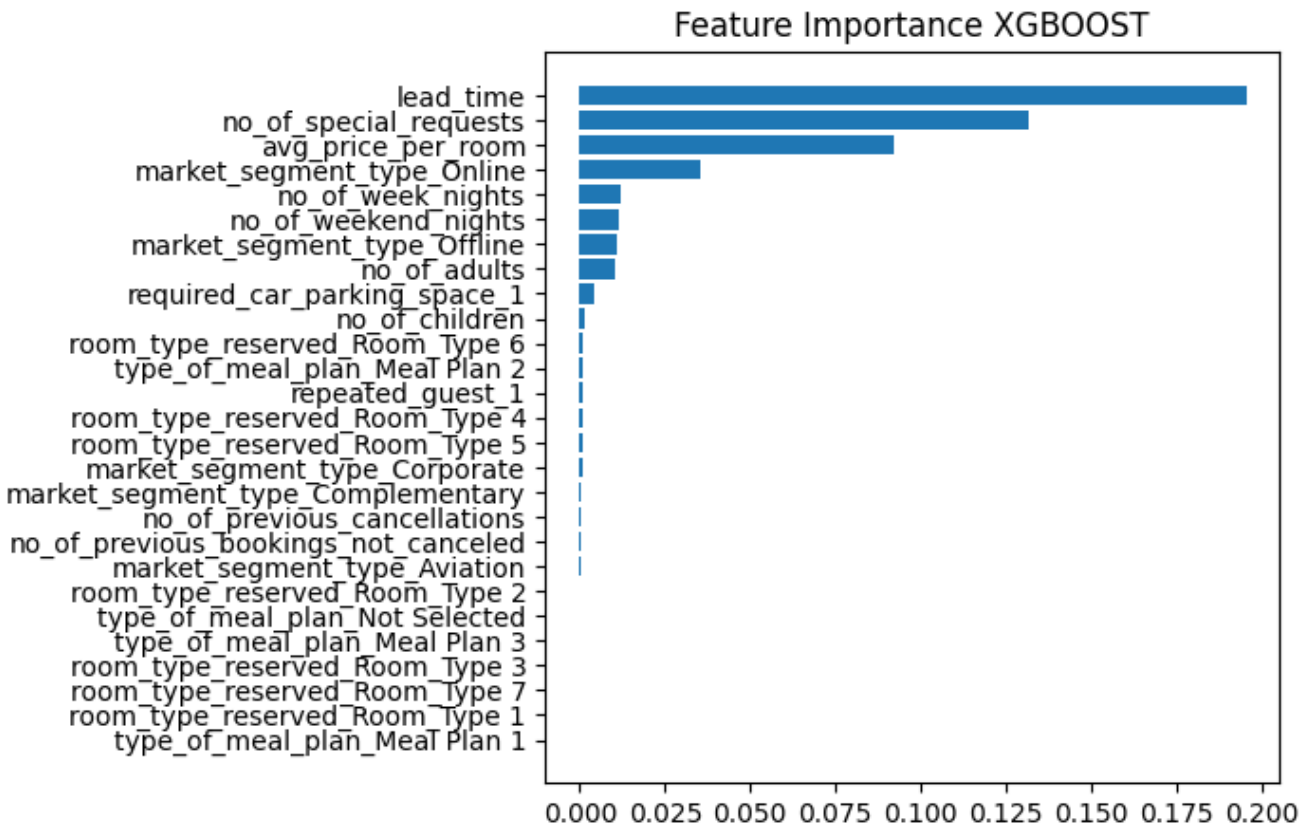
8 Proposta de modelos com engenharia de atributos)

Como visto na parte de análise de exploratória, as variáveis que envolvem tempo podem ser interessantes para a modelagem do problema. Há algumas abordagens para esse problema, sendo uma adaptar os modelos apresentados para incorporar uma estrutura temporal, contudo isso tornaria os modelos muito complexos e seria necessário propor novas métricas de aviação, por isso essa abordagem não foi explorada no nosso trabalho. Outra abordagem é criar um novo atributo categórico com base nos atributos de tempo , no nosso caso, ele foi categorizado em trimestre.

Nessa etapa aplicamos XGBoost e Balanced RandomForest em dois cenários : considerando o atributo trimestre e considerando o atributo tremeste mais uma combinação de atributos por meio de polinômio de grau 2 (por exemplo $y = x + y$ torna-se $y = x^2 + y_2 + xy + x + y$ Os resultados são exibidos a seguir:

e o gráfico de importância de atributos :

Note que o resultado é muito semelhante aos anteriores, mas o tempo aparece como a quinta variável mais importante.



index	balaneced_acc	recall	precision	average_precision
Balanced Rf + fe	0.8745 +/- 0.0016	0.8617 +/- 0.004	0.7885 +/- 0.0072	0.9149 +/- 0.0054
balanced_rf + fe + poly	0.8767 +/- 0.0035	0.864 +/- 0.0041	0.7922 +/- 0.01	0.9185 +/- 0.0038
XGboost + fe	0.8579 +/- 0.005	0.7791 +/- 0.0109	0.8572 +/- 0.0068	0.9106 +/- 0.0026
XGboost + fe + poly	0.8613 +/- 0.0034	0.7841 +/- 0.0079	0.8615 +/- 0.0076	0.9146 +/- 0.0028

8.1 Combinação de classificadores

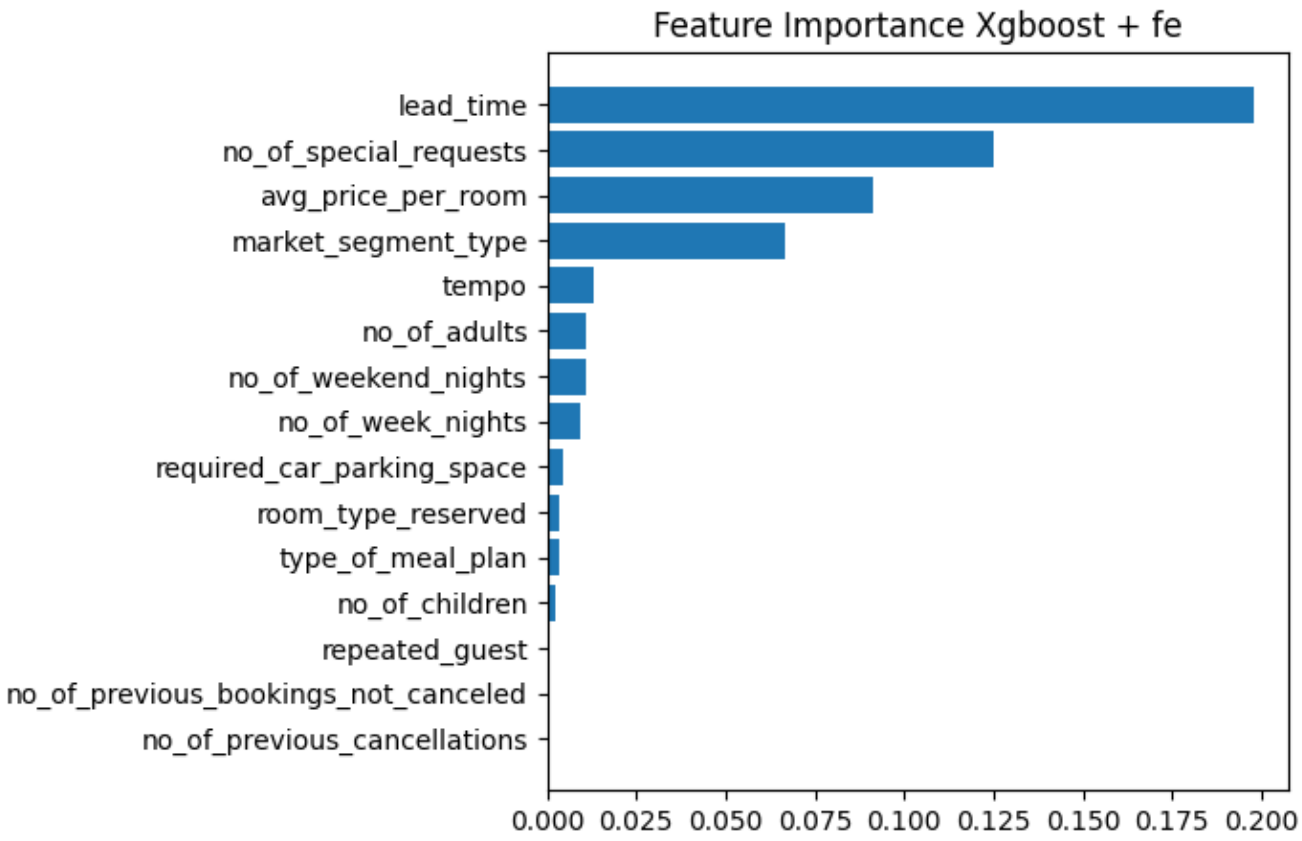
Já que ambos o XGBoost e o random florest performam bem, foi proposto um ensamble de quatro classificadores : XGBoost padrão , XGBoost com scale pos weight otimizado, random florest padrão e imbalanced random florest . O esquema de engenheria de variável foi categorização do tempo em trimestre + transformação de polinômios. E o resultado é o seguir:

index	balaneced_acc	recall	precision	average_precision
pipe_voting	0.8754 +/- 0.0034	0.8175 +/- 0.0072	0.8568 +/- 0.0086	0.9242 +/- 0.0022

Esse é o nosso melhor classicador em termo de acurácia balancead e avarage precision,

9 Conclusão

Nesse trabalho foram comprados várias abordagens para o problema de predizer o cancelamento de uma revesa de hotel. A respeito dos modelos interpretáveis notou-se que o naive bayes e a regressão logística sofrem com o problema da separação total que está presente nessa base de dados , já a árvore



decisão aprestou excelente desempenho e não sofre desse problema. Isso posto, como modelo interpretável foi escolhido a árvore e decisão e, a fim de aumentar seu poder preditivo, foi realizada uma otimização de hyper-parâmetros.Ainda, em anexo, há um código que fornece uma representação gráfica dessa árvore de decisão. Se a interpretabilidade do modelo é importante para o negócio, recomenda-se utilizar a estratégia de árvore de decisão com otimização de hyper-parâmetros.

Além disso , para os modelos caixa preta ,foram testadas estratégias de bagging (random florest) e boosting (XGBoost) e suas adaptações para o desbalanceamento dos dados. Nesse sentido, ambas estratégias forneceram bons resultados e o xgboost ,por meio de um hyper-parâmetro, foi capaz de fornecer uma maneira de "controlar" a relação de recall-precision. Outrossim, testamos categorizar um atributo relacionado ao tempo e obtivemos melhores resultados, embora seja necessário ponderar essa categorização. Um modelo mais complexo foi proposto, incluindo a categorização de uma variável temporal para trimestre e aplicando uma transformação polinomial de grau 2, essa estratégia forneceu maior acurácia balanceada e maior Average Precision dentre os modelos testados.

O algoritmo de balanceamento artificial SMOTE proporcionou um melhor "balanço" na relação precisão-acurácia e ,em geral, melhorou a acurácia balanceada, em especial a sua utilização com a árvore de decisão + otimização de hiper-parâmetros produziu ótimos resultados .

Os gráficos de importância de atributos mostram que as variáveis que mais influenciam no cancelamento são : tempo de espera, número de pedidos especiais (quarto em andares mais altos, vista bonita etc), reserva feita por canal online e preço médio por reserva. Essa informação é muito importante para o contexto de negócio, pois,por meio delas. é possível propor mudanças no plano de negócio para evitaro cancelamento. Por fim, ao setor hoteleiro, recomenda-se propor alternativas para diminuir o tempo de espera até a reserva e , se possível, dar mais atenção para os clientes que fazem reserva online.

10 Anexo dos codigos

O código pode ser encontrado no google colaboratory por meio do link : https://colab.research.google.com/drive/1MZv2cfWvBRAdn-iI76ZekUkt_CmCNgEB?usp=sharing e no repositório do github: https://github.com/Gabrieldpll/Hotel-Reservations/.

Posteriormente serão postadas novas versões e pretende-se publicar os resultados no Kaggle.

11 Resumo siminários

11.1 Trabalho Catboost

De fato o problema da representação dos atributos é presente nos problemas de aprendizado de máquina, nesse trabalho foi visto como o naive bayes foi afetado pelos atributos categóricos com categorias pouco frequentes, outro exemplo de dificuldade em lidar com atributos categóricos são os algoritmos baseado em distância, note que não há nos pacotes mais comuns uma distância que leve em consideração a natureza dos atributos, por exemplo,a distância de Gower não é implementada nas bibliotecas mais comuns de python. Sendo assim, é extremamente vantajoso ter um algoritmo que seja capaz de lidar com o "problema das variáveis qualitativas".

No artigo foi comparado o desemepenho do catboost com o xgboost em base de dados reais e em muito deles o catboost obteve desempenho melhor , o que indica que é interessante utilizar esse modelo. Além disso, o catboost também é rápido. A desvangem desse modelo é semelhante aos algoritmos de boosting sendo elas : sensibilidade aos hyoer-parâmetros e Interpretabilidade

11.2 Trabalho predição de câncer

Esse trabalho é muito interessante, pois ele é um exemplo de como o aprendizado pode ser utilizado para contribuir com a sociedade. Um fato interessante é que foram utilizados dados banco de dados da Fundação Oncocentro do Estado de São Paulo (FOSP) , que é um banco de dados aberto. O foco do artigo é na predição, embora na área de saúde seja bem comum aplicar análise de sobrevivência, regressão logísticas ou outros modelos interpretáveis. Uma característica interessante das base de dados de saúde é o desbalanceamento dos dados, geralmente há poucos exemplos de paciências doentes, foram reportadas métricas adequadas para o desbalanceamento (f1 e precision) e, além disso além foi reportada a matriiz de confusão (Isso basta para que outros pesquisadores calculem outras métricas).

Para tratar o desbalanceamento dos dados o autor utilizou de balanceamento artificial dos dados por meio do SMOTE , ou seja, ele gerou dados artificiais da classe minoritária até que as classes ficassem desbalanceadas e obteve uma melhoria no desempenho. Convém ressaltar que o SMOTE não gera observações duplicadas (diferente de amostragem aleatória com repetição) e que o SMOTE utiliza k-nn em sua implementação. Atualmente, há várias variantes do SMOTE e também há algoritmos que removem observações da classe majoritária , sendo que uma alternativa comum é combinar SMOTE com undersampling. Outra alternativa é, em vez de aplicar métodos de balanceamento, adaptar os algoritmos de aprendizado de máquina para o desbalanceamento de dados, que é conhecido como "cost sensitive".

11.3 Trabalho predição de detecção de água envenenada

Primeiramente, há um valor social desse trabalho, já que a água é um recurso essencial para vida e , principalmente em países periféricos, a água nem sempre é potável e desenvolver ferramentas capazes

de testar a qualidade da água pode ser útil nesses cenários. Outro ponto interessante é que os pesquisadores coletaram os dados manualmente, geralmente os artigos de machine learing usam dados simulados,confidicenciais ou abertos e, além disso o método coleta é um tanto quanto inusitado, não é todo pesquisador que pensaria em utilizar o sinal de wifi para medir qualidade da água (um dos pesquisadores é do departamento de química,talvez seja ele quem teve a ideia).

Foram feitas 5470 observações e as classes são balanceadas e testados quatro modelos : SVM,KNN,LSTM e Ensembles , foi realizada um k-fold crossvalidation. Os modelos foram testados em três cenários contendo diferentes quantidade de água limpa e contaminada. A conclusão foi que os métodos de classificação foram eficientes com acurácia minimiza no pior dos cenários de obteves-se 0.89 de acurácia com o algortimo LSTM e no melhor dos cenários 0.92 usando esambles.

Referências

[Bergstra et al., 2013] Bergstra, J., Yamins, D., Cox, D. D., et al. (2013). Hyperopt: A python library for optimizing the hyperparameters of machine learning algorithms. In *Proceedings of the 12th Python in science conference*, volume 13, page 20. Citeseer.

[Breiman, 2001] Breiman, L. (2001). Random forests. *Machine learning*, 45:5–32.

[Chawla et al., 2002] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357.

[Chen et al., 2004] Chen, C., Liaw, A., Breiman, L., et al. (2004). Using random forest to learn imbalanced data. *University of California, Berkeley*, 110(1-12):24.

[Chen and Guestrin, 2016] Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.