Relatório do desafio de ciência de dados

Pergunta [1]:
Pontos levantados a partir da análise exploratória dos dados (EDA):

- Das 16 *features*, as que apresentaram mais elementos faltando foram "ultima_review", "host_name" e "nome". Uma hipótese que foi assumida foi que essas características não exercem grande impacto no preço e, portanto, as colunas que as representam foram removidas.
- Através da análise do *heatmap*, verificamos que as oito *features* numéricas, no geral, causam pouco impacto no preço. As três principais são "disponibilidade_365" (0,08), "calculado_host_listings_count" (0,06) e "minimo_noites" (0,04);
- Analisando o gráfico de densidade do preço verificamos que os preços, no geral, estão mais concentrados na faixa de 0 a 1000 dólares.
- Analisando o *scatterplot* que mostra a relação entre "disponibilidade_365", "price" e o "room_type" verificamos que o preço varia mais e é mais alto quando o "room_type" selecionado é "Entire home/apt". Além disso, percebemos que a relação entre preço e disponibilidade não é linear e, por último, que a quantidade de quartos alugados do tipo "Entire home/apt" é maior que o tipo de quarto privado, independente do número de dias nos quais o anúncio está disponível.
- Por último, observamos que os preços mais caros são dos aptos alugados em Manhattan e os mais baratos são os aptos alugados em Bronx.

Pergunta [2]:

a. Caso uma pessoa quisesse alugar um apartamento, o local que seria indicado dependeria do quanto a pessoa está disposta a pagar. Com base nisso, o local mais adequado seria escolhido. No entanto, do ponto de vista de retorno financeiro para a plataforma, apartamentos/casas inteiros localizados em Manhattan seriam indicados.
b. Com base no *heatmap*, a disponibilidade ao longo do ano e o número mínimo de noites interferem pouco no preço, com uma correlação entre as variáveis e o preço de cerca de 0,08 e 0,04, respectivamente.

Pergunta [3]:

Antes de iniciar a codificação, as etapas para a análise e resolução do problema definidas foram:

1. Decidir quais as *features* que serão utilizadas, já que sabemos que a variável alvo é o preço.
2. Após a primeira seleção de *features*, verificar se as colunas de dados estão completas, caso contrário, é necessário fazer um pré-processamento dos dados.
3. Para refinar a decisão das *features*, plotar gráficos em pares como bairro vs preço, tipo de quarto vs preço, etc, para verificar quais as *features* que mais impactam no preço dos aluguéis.
4. Após a seleção final das *features*, dividir os dados em subconjuntos para treinamento e teste.
5. Construir um modelo para treinar e testar os dados.
6. Selecionar uma ou mais métricas adequadas e calculá-las para avaliar o aprendizado do modelo no treinamento.
7. Caso o modelo tenha tido um desempenho considerado bom, testá-lo com exemplos novos e gerar uma ou mais métricas de desempenho.

8. Caso contrário, ajustar os hiperparâmetros ou realizar outros tipos de alterações para melhorar o desempenho do modelo.
9. Testar o modelo novamente após o ajuste de parâmetros e outras alterações.

Após a codificação, ajustes precisaram ser realizados, por conta das seguintes questões:

- Na primeira tentativa, a seleção das *features* para desenvolver o modelo foi feita com base em gráficos que mostravam a relação de algumas delas com o preço e, além disso, outras *features* foram desconsideradas por assumir que elas tinham pouco impacto no preço. Então, nessa tentativa, dois subconjuntos de *features* foram testados:
    1. "disponibilidade_365, ""calculado_host_listings_count" "minimo_noites", "bairro_group", "bairro" e "room_type";
    2. "bairro_group", "bairro" e "room_type"
- Testando o modelo de *Linear Regression* com o segundo conjunto, o desempenho foi baixo, na faixa de aproximadamente 0.085, usando a métrica R2 (coeficiente de determinação). Com o primeiro conjunto, o desempenho foi similar.
- Por conta disso, foi necessário encontrar uma forma melhor de selecionar as características mais relevantes para a construção do modelo e, além disso, aplicar normalização, remoção de outliers e fazer o ajuste de hiperparâmetros

Ao todo, 13 variáveis foram selecionadas: " room_type_Private room", "longitude", "latitude", "id", "bairro_group_Manhattan", "host_id", "disponibilidade_365", "room_type_Shared room", "minimo_noites", "numero_de_reviews", "calculado_host_listings_count", "bairro_Williamsburg", "bairro_Midtown".

As variáveis " room_type_Private room", "bairro_group_Manhattan", "room_type_Shared room", "bairro_Williamsburg" e "bairro_Midtown" foram geradas após aplicar a técnica *one-hot encoding* utilizada para converter variáveis categóricas em valores numéricos.

A melhor forma encontrada para selecionar as *features* mais relevantes foi utilizando o método de aprendizado em conjunto ExtraTreesRegressor para medir o grau de importância das *features*. Como resultado, um gráfico de barras foi gerado e as 13 *features* com maior grau de importância foram selecionadas.

Além disso, *outliers* foram calculados e removidos usando o *Interquartile Range* (IQR), que mede estatisticamente a dispersão dos dados contidos no *dataset*. Por último, antes de partir para a construção do modelo, os dados das *features* "id", "host_id", "minimo_noites", "numero_de_reviews", "calculado_host_listings_count", "disponibilidade_365", "latitude", "longitude" foram normalizados convertendo os valores para um intervalo entre 0 e 1.

Todas essas alterações foram feitas com o objetivo de melhorar o desempenho do modelo.

Como o problema que estamos lidando é um problema de regressão, pois os valores que a variável alvo preço pode assumir são contínuos, os algoritmos utilizados foram: *Linear Regressor*, *Random Forest Regressor*, *Support Vector Machine* (SVM) e *CatBoost Classifier*.

O algoritmo que obteve o melhor desempenho foi o *CatBoost Classifier*, com R2 *score* de 0,5893 antes do ajuste de hiperparâmetros utilizando a técnica de otimização *GridSearch*, e 0,5907 após o *GridSearch*. O *CatBoost* utiliza a técnica de boosting juntamente com árvores de decisão e funciona bem com variáveis categóricas.

A métrica escolhida foi o R2 *score* ou coeficiente de determinação que mede o quão bem o modelo se ajusta aos dados. O R2 *score* varia entre 0 e 1 e, quanto mais próximo de 1, melhor é o desempenho do modelo.

Pergunta [4]:

A sugestão de preço utilizando o modelo construído foi de 157,67 dólares. O preço alvo é 225 dólares, logo, ocorreu uma diferença de preço de 67,33 dólares. Como o R2 *score* do modelo no conjunto de teste foi de 0,5907, essa diferença era esperada.