Questões sobre o modelo de precificação

As análises a seguir foram feitas com base no arquivo '*indicium.ipynb*', disponível no repositório do github https://github.com/millennium164/indicium_desafio, e foram também comentadas no vídeo disponível em pasta compartilhada do google drive https://drive.google.com/drive/folders/1mAvmy01KzaDkOgEaanAm9KqF8C8A2sqz?usp=sharing.

1. Supondo que uma pessoa esteja pensando em investir em um apartamento para alugar na plataforma, onde seria mais indicada a compra?

Considerando que um investimento é feito pensando em retorno financeiro, um investimento em um apartamento para alugar deve levar em conta o preço de compra e o valor do aluguel do imóvel. Avaliando o contexto de Nova York, como se pode ver na seção 'preço por grupo de bairro', os aluguéis mais caros se concentram em Manhattan e Brooklyn. Assim, pensando no lucro apenas em termos de preço de aluguel, seriam boas opções.

No entanto, para considerar um bom investimento, é necessário também avaliar o retorno sobre capital investido, que depende do preço de compra. O preço de compra de um imóvel, assim como seu aluguel, depende de vários fatores, os quais não obstante certamente se entrelaçam. O diferencial, portanto, acredito que esteja na relação de oferta e demanda. Ou seja, mesmo que um imóvel esteja em uma ótima localização, tenha quartos espaçosos e tenha sido recém-reformado, se a oferta for muito alta, o preço tende a abaixar.

A relação de quantidade de imóveis por grupo de bairro pode ser vista na seção 'anúncios por grupo de bairro'. A maioria esmagadora é de Manhattan e Brooklyn, em termos absolutos. No entanto, existem mais anúncios mas a área é proporcionalmente maior, isso não faz com que os recursos sejam 'escassos' (alta demanda ou baixa oferta). Portanto, para uma análise mais aprofundada, poderíamos comparar a relação entre oferta e área do grupo de bairro. Quanto menor essa razão, mais escasso é o produto e, então, mais caro. De forma equivalente, quanto maior a razão entre demanda e área, mais escasso também. De forma visual:

$$\uparrow escassez = \frac{\downarrow oferta}{\uparrow área}$$

$$\uparrow escassez = \frac{\uparrow demanda}{\downarrow área}$$

A partir dessa rápida análise, percebe-se que um investimento rentável em imóvel para alugar é uma questão complexa que depende de vários fatores, mas cuja equação de todas as variáveis deve buscar minimizar o preço de compra e maximizar o valor do aluguel, para então maximizar o lucro.

2. O número mínimo de noites e a disponibilidade ao longo do ano interferem no preço?

Como é possível observar nas matrizes de correlação, tanto linear quanto não-linear, os atributos 'minimo de noites' e 'disponibilidade 365' não parecem estar muito relacionados ao target attribute, o preço do aluguel. Não obstante, quando retiramos essas features do treino dos modelos preditivos, perdemos um pouco de performance. Isso provavelmente ocorre porque o número de atributos (colunas) do nosso dataset não é muito grande. Então há um incremento de performance com a consideração desses atributos, mas são ganhos marginais.

Além disso, essas duas features apresentam um range de valores muito alto, o que fornece uma distribuição esparsa, com pontos de concentração e assimetria. Olhando pela representação visual na análise preço x minimo noites, não conseguimos extrair uma relação distinta entre as variáveis. Já os valores de disponibilidade, apesar de também apresentarem um range grande, estão mais uniformemente distribuídos, embora haja um pico próximo ao 0.

3. Existe algum padrão no texto do nome do local para lugares de mais alto valor?

Para fazer uma análise de texto, provavelmente seria necessário o uso das famosas LLM's (large language model) ou algum outro modelo que trabalhe com caracteres do alfabeto. No arquivo indicium.ipynb é feita uma investigação mais visual e subjetiva, tomando os anúncios em ordem crescente e em ordem decrescente de valor de aluguel. Com base nessa análise superficial, eu notei que os anúncios dos aluguéis mais caros

costumam ter termos menos comuns (mais ‘exóticos) na frase; seja a localização/ região /nome específico que pessoas fora de NY provavelmente não conhecem (Flatbush, TriBeCa), seja uma referência não relacionada à localização ou ao imóvel, porém chamativa (Super Bowl); e etc.

Já nos anúncios de aluguéis mais baratos, um padrão que pra mim foi saliente é que costumam ser descrições mais objetivas dos imóveis (cozy room, spacious, modern apartment, etc). Então ao invés de termos e conceitos mais subjetivos e não relacionados ao imóvel físico, como nos anúncios mais caros, os anúncios de aluguel menor parecem ser, geralmente, descrições simples, concretas e objetivas do imóvel físico.

4. Explique como você faria a previsão do **preço** a partir dos dados. Quais variáveis e/ou suas transformações você utilizou e por quê? Qual tipo de problema estamos resolvendo (regressão, classificação)? Qual modelo melhor se aproxima dos dados e quais seus prós e contras? Qual medida de performance do modelo foi escolhida e por quê?

Como explicado melhor no vídeo, dada a assimetria das distribuições de grande parte dos atributos, a capacidade de generalização dos modelos e, portanto, seu poder preditivo, é comprometida. Para contornar isso, utilizam-se transformações, como as Power transforms, para normalizar/tornar mais simétrica, uma distribuição de valores de uma variável. Essas transformações citadas envolvem potências e logaritmos, pois estes são mais sensíveis a valores infinitesimais ou tendendo a tal. Por exemplo, no caso do nosso dataset, temos assimetrias positivas, ou seja, dados mais dispersos em valores maiores (menos concentrados). Então, para ‘trazê-los’ mais para a esquerda, para ‘comprimi-los’, podemos usar logaritmos.

Assim, testei diferentes métodos para lidar com a não-uniformidade, range e assimetria dos dados. Para o target attribute, o preço, é imprescindível aplicar uma transformação dessas. Já quanto aos predictive attributes, decerto a normalização de suas distribuições melhora a performance do modelo, ainda que muitas vezes de forma bem menos expressiva e alarmante do que em relação ao target attribute.

No caso de precificação de aluguéis, estamos tratando de um problema de regressão. Um dado pode ser caracterizado por diferentes aspectos, os quais podem mensurá-lo quantitativamente (numéricos) ou descrevê-los a partir de categorias (categóricos). Quando queremos criar modelos para prever um atributo numérico, estamos

diante de uma tarefa de regressão; quando é uma característica qualitativa, um problema de classificação. No contexto de regressão,, o primeiro modelo que vem em mente é o Linear Regression. No entanto, outros modelos não-lineares, bem como originalmente destinados à tarefa de classificação porém transpostos para regressão, também são opções. Existe a regressão logística, decision trees for regression, suport vector machines por regression, e muitos outros.

Pela simplicidade, e até porque ainda não estou tão familiarizada com modelos além destes, eu optei por testar Linear Regression, Decision Tree e Random Forest. Cada um possui vantagens e desvantagens, como podemos observar nos comparativos a seguir, retirados do livro A General Introduction to Data Analytics [1].

Table 8.2 Advantages and disadvantages of linear regression.

Advantages	Disadvantages
• Strong mathematical foundation • Easily interpretable • Hyper-parameter free	• Poor fit if relationship between predictive attributes and target is non-linear • The number of instances must be larger than the number of attributes • Sensitive to correlated predictive attributes • Sensitive to outliers

Table 10.2 Advantages and disadvantages of decision trees.

Advantages	Disadvantages
• Interpretable both as a graph and as a set of rules • Pre-processing free since robust to outliers, missing data, correlated and irrelevant attributes, and do not need previous normalization	• The definition of a rule to split a node is evaluated locally without enough information to know whether the rule guarantees the global optimum, i.e. the minimum number of objects misclassified after the tree is completed. This is due to the greedy search and can result in a sub-optimal solution. • Since the rules are of the type $x < a$, where x is a predictive attribute and a is a value, a decision tree splits the bi-dimensional space with horizontal and vertical lines, as shown in Figure 10.2, which makes some problems hard to deal with.

Table 11.2 Advantages and disadvantages of random forests.

Advantages	Disadvantages
• Very good predictive performance in many problems • Easy to define/tune hyper-parameters	• Computationally expensive since the number of recommended trees is large, but, like bagging, can be parallelized • Randomization, but this can be minimized using the recommended number of trees

Para medir a performance dos modelos utilizados, eu utilizei como indicador quantitativo o R^2, que mede variância entre valores preditos e reais; como análise qualitativa, eu plotei gráficos com os preditos e reais. Além disso, como parâmetro de 'consolação', costuma-se indicar o desempenho de um modelo dummy, que seria um modelo bem simples. Para regressão, eu utilizei como dummy regressor um modelo que chuta todos os valores como sendo a média, e obtém um desempenho péssimo, já que temos um range de preços muito grande e uma distribuição bastante assimétrica.

5. Supondo um apartamento com as seguintes características:

```
{'id': 2595,
 'nome': 'Skylit Midtown Castle',
 'host_id': 2845,
 'host_name': 'Jennifer',
 'bairro_group': 'Manhattan',
 'bairro': 'Midtown',
 'latitude': 40.75362,
 'longitude': -73.98377,
 'room_type': 'Entire home/apt',
 'minimo_noites': 1,
 'numero_de_reviews': 45,
 'ultima_review': '2019-05-21',
 'reviews_por_mes': 0.38,
 'calculado_host_listings_count': 2,
 'disponibilidade_365': 355}
```

Qual seria a sua sugestão de preço?

Fornecendo diretamente o valor predito pelo modelo random forest, disponibilizado em arquivo .pkl, no repositório do github, que eu optei para ser o oficial, estimou-se um aluguel de $336.76:

```
pred_dado = model_pkl.predict(x)
dd = pd.DataFrame({'predicted': list(pred_dado)})
dd = predicts_test(dd, pred_dado)
dd
```

	predicted	predicted_reverted
0	3.717714	336.758543

[1] https://fliphtml5.com/pnbjj/slil/A_General_Introduction_to_Data_Analytics_%28Jo%C3%A3o_Moreira%2C_Andre_Carvalho%2C_Tom%C3%A1s_Horvath%29_%28z-lib.org%29/#google_vignette