New York City Airbnb Open Data

گردآوری گزارش توسط: فاطمه نظرمحمدی

مقدمه

دیتاستی که در این پروژه مورد بررسی قرار گرفته است، مربوط به اطلاعات املاک اجاره‌ای در شهر نیویورک ایالات متحده و شامل 48895 رکورد است و ویژگی‌های id، name، host_id، host_name، neighbourhood_group، neighbourhood، latitude، longitude، room_type، Price، minimum_nights، number_of_reviews، last_review، reviews_per_month، calculated_host_listings_count و availability_365 را در بر می‌گیرد.

در گام اول به مدیریت داده‌های null می‌پردازیم؛ این کار را در آغاز با بررسی و حذف columهایی که بیش از 20 درصد آن‌ها null هستند انجام می‌دهیم و سپس به حذف columهایی می‌پردازیم که ارزش افزوده‌ای برای مدل‌سازی مد نظر در این پروژه(از نگاه تحلیل‌گر) ندارند.

کار پاکسازی داده‌ها را با پرکردن داده‌های null به کمک میانگین ادامه می‌دهیم و در ادامه نیز به سراغ حذف داده‌های outlier که باعث پیچیدگی مدل مد نظر می‌شوند، می‌رویم.

پس از این‌که کار پاکسازی داده‌ها با بررسی داده‌های کتگوریکال به سرانجام رسید، به مصورسازی داده‌ها برای به دست آوردن نتایج مورد نظر می‌پردازیم؛ در واقع می‌خواهیم به روابط بین فیچرها پی ببریم؛ بنابراین تعداد آگهی و پارامترهای مختلف آگهی از جمله: تعداد آگهی‌ها در مناطق مختلف جغرافیایی، اطلاعات در خصوص تعداد انواع خانه‌ها ، قیمت و... را بررسی می‌کنیم.

روش‌ها

رسیدگی به دیتاهای null

در نخستین گام از کار به سراغ پاکسازی داده‌ها و در ابتدای این بخش به سراغ داده‌های null می‌رویم. با دستورات 7، 8 و 9 پس از بررسی داده‌های null، به روش آزمون و خطا به این می‌رسیم که فیچرهایی را که بیش از 20 درصد آن‌ها null هستند با دستور drop حذف کنیم. (بررسی‌ها نشان‌دهنده آن هستند که فیچری در این دیتاست وجود دارد که بیش از مقدار 20 درصد null داشته باشد)

بدین ترتیب، در این مرحله last_review و review_per_month از جمع فیچرها حذف می‌شوند.

حذف columهایی که مفید نیستند

با یادآوری این‌که با توجه به نگاه تحلیل‌گر داده و همچنین مدل مد نظر، فیچرهایی که می‌توان به آن‌ها مفید نبودن را نسبت داد متفاوت هستند، به سراغ فیچرهایی می‌رویم که از دید ارائه‌دهنده این پروژه چندان مفید به نظر نمی‌رسند؛ از این رو، در این مرحله نیز id، host_id و number_of reviews را حذف می‌کنیم. (دستورات 10 و 11)

پر کردن داده‌ها

پرکردن دیتاهای null باقی‌مانده مرحله بعدی کار است. در این بخش (دستورات 12 و 13) دیتاهای numeric را با مقدار میانگین پر می‌کنیم.

در ادامه دیتاهای numeric را با مقدار (airbnb[cols] - airbnb[cols].mean())/(airbnb[cols].std()) نورمالایز می‌کنیم. (دستور 15)

در بخش دیگری از پاکسازی داده‌ها، لازم است تا دیتاهای outlier را که باعث پیچیده شدن مدل می‌شوند، حذف کنیم. این کار را با دو مقدار upper_range و lower_range انجام می‌دهیم. (دستور 17)

پس از این عمل، تعداد رکوردها و فیچرهای دیتاست ما به (46189, 11) درخواهد آمد.

در ادامه داده‌های کتگوریکال پر می‌شوند. برای این کار در واقع ابتدا مقدار value_counts برای هر یک از فیچرهای کتگوریکال محاسبه شدند. و شرط دستور20 و در ادامه 21 برای آن‌هایی که بیشترین فرکانس را دارند اجرا می‌شود.

اگر پس از این مرحله، جمع باینری مقدار null هر یک از فیچرها را بررسی کنیم، همگی صفر خواهد شد.

```
[22]: #checking
      airbnb.isna().sum()
```

```
[22]: name                              0
      host_name                         0
      neighbourhood_group               0
      neighbourhood                     0
      latitude                          0
      longitude                         0
      room_type                         0
      price                             0
      minimum_nights                    0
      calculated_host_listings_count    0
      availability_365                  0
      dtype: int64
```

پس از این طی دستورات 23 تا 33 درخصوص هر یکی از فیچرها ابتدا بررسی می‌کنیم که چه میزان پراکندگی داده دارند و سپس در صورتی که تجمیع این پراکندگی‌ها در یک مقدار ثانویه other به مدل کمک کند، داده‌هایی از هر فیچر که در اقلیت قرار دارند را تجمیع می‌کنیم. این کار برای هر یک از فیچرها با یک تابع انجام می‌شود که مقدار other را برای مقادیر تعیین‌شده برمی‌گرداند.

ساخت ماتریس و مصورسازی داده

پس از پاکسازی داده در ابتدا یک correlation Matrix را با دستور 34 و 35 می‌سازیم.

همچنین با شرط دستور 36 تا 38 داده‌های convert categorical را به dummies variables تبدیل می‌کنیم.

با دستور 40 نیز فیچرها را از تارگت جدا می‌کنیم که در اینجا availability_365 به عنوان تارگت قرار گرفته است.

نتایج

در این پروژه وضعیت فیچرهای دیتاست در ابتدا به شکل زیر است:

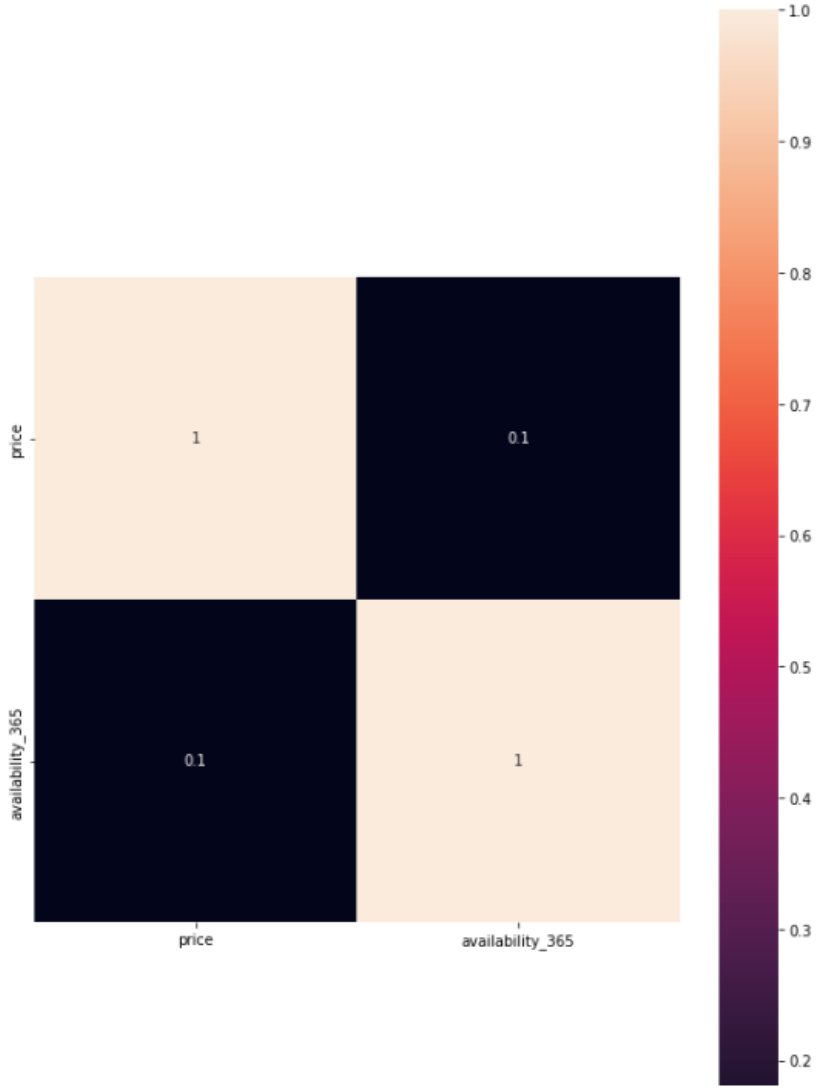
```
RangeIndex: 48895 entries, 0 to 48894
Data columns (total 16 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   id                              48895 non-null  int64
 1   name                            48879 non-null  object
 2   host_id                         48895 non-null  int64
 3   host_name                       48874 non-null  object
 4   neighbourhood_group             48895 non-null  object
 5   neighbourhood                   48895 non-null  object
 6   latitude                        48895 non-null  float64
 7   longitude                       48895 non-null  float64
 8   room_type                       48895 non-null  object
 9   price                           48895 non-null  int64
 10  minimum_nights                  48895 non-null  int64
 11  number_of_reviews               48895 non-null  int64
 12  last_review                     38843 non-null  object
 13  reviews_per_month               38843 non-null  float64
 14  calculated_host_listings_count  48895 non-null  int64
 15  availability_365                48895 non-null  int64
dtypes: float64(3), int64(7), object(6)
memory usage: 6.0+ MB
```

همچنین وضعیت داده‌های null دیتاست بررسی‌شده به شکل زیر است:

```
id                                0.000000
name                              0.000327
host_id                           0.000000
host_name                         0.000429
neighbourhood_group               0.000000
neighbourhood                     0.000000
latitude                          0.000000
longitude                         0.000000
room_type                         0.000000
price                             0.000000
minimum_nights                    0.000000
number_of_reviews                 0.000000
last_review                       0.205583
reviews_per_month                 0.205583
calculated_host_listings_count    0.000000
availability_365                  0.000000
dtype: float64
```

همچنین در ادامه شمایی از AxesSubplot به‌دست‌آمده، قرار داده شده است:

[34]: <AxesSubplot:>



که از این طریق به ارتباط خطی میان دو فیچر availability_365 و price پی می‌بریم.