Apartment rental offers in Germany

گردآوری گزارش توسط: فاطمه نظرمحمدی

مقدمه

دیتاستی که در این پروژه مورد بررسی قرار گرفته است، مربوط به اطلاعات املاک اجاره‌ای در آلمان و شامل 268850 رکورد است و ویژگی‌های regio1، serviceCharge، heatingType، telekomTvOffer، telekomHybridUploadSpeed، newlyConst، balcony، picturecount، pricetrend، telekomUploadSpeed، totalRent، yearConstructed، scoutId، noParkSpaces، firingTypes، hasKitchen، geo_bln، cellar، yearConstructedRange، baseRent، houseNumber، livingSpace، geo_krs، condition، interiorQual، petsAllowed، street، streetPlain، lift، baseRentRange و... را در بر می‌گیرد.

در گام اول به مدیریت داده‌های null می‌پردازیم؛ این کار را در آغاز با بررسی و حذفcolum هایی که بیش از 60 درصد آن‌ها null هستند انجام می‌دهیم و سپس به حذفcolum هایی 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 دیتاهایnull

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 داده‌هایnull ، به روش آزمون و خطا به این می‌رسیم که فیچرهایی را که بیش از 20 درصد آن‌ها null هستند با دستور drop حذف کنید. (بررسی‌ها نشان‌دهنده آن هستند که فیچری در این دیتاست وجود دارد که بیش از مقدار 20 درصد null داشته باشد) بدین ترتیب، فیچرهای باقی‌مانده در این مرحله چنین است:

```
[8]: Index(['regio1', 'serviceCharge', 'heatingType', 'telekomTvOffer',
       'newlyConst', 'balcony', 'picturecount', 'pricetrend',
       'telekomUploadSpeed', 'totalRent', 'yearConstructed', 'scoutId',
       'firingTypes', 'hasKitchen', 'geo_bln', 'cellar',
       'yearConstructedRange', 'baseRent', 'houseNumber', 'livingSpace',
       'geo_krs', 'condition', 'interiorQual', 'petsAllowed', 'street',
       'streetPlain', 'lift', 'baseRentRange', 'typeOfFlat', 'geo_plz',
       'noRooms', 'thermalChar', 'floor', 'numberOfFloors', 'noRoomsRange',
       'garden', 'livingSpaceRange', 'regio2', 'regio3', 'description',
       'facilities', 'date'],
      dtype='object')
```

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 picturecount، description، geo_plz، interiorQual، newlyConst، houseNumber، scoutId، geo_bln، firingTypes، geo_krs، condition، thermalChar، streetPlain، garden، balcony و typeOfFlat را حذف می‌کنیم. (دستورات 11 و 12)

پر کردن داده‌ها

پرکردن دیتاهای null باقی‌مانده مرحله بعدی کار است. در این بخش (دستورات 14 و 15) دیتاهای numeric را با مقدار میانگین پر می‌کنیم.

در ادامه دیتاهای numeric را با مقدار (p2[cols] - p2[cols].mean())/(p2[cols].std()) نورمالایز می‌کنیم. (دستور 16)

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 18)

پس از این عمل، تعداد رکوردها و فیچرهای دیتاست ما به (264272, 26) 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 دستور22 و در ادامه 23 برای آن‌هایی که بیشترین فرکانس را دارند اجرا می‌شود.

پس از این، ضمن حذف دوباره برخی دیگر از فیچرها برای راحتی ادامه کار، طی دستورات 27 تا 41 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 42 و 43 می‌سازیم.

همچنین با شرط دستور 44 تا 46 داده‌های convert categorical را به dummies variables تبدیل می‌کنیم .

با دستور 48 نیز فیچرها را از تارگت جدا می‌کنیم که در اینجا date به عنوان تارگت قرار گرفته است .

نتایج

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

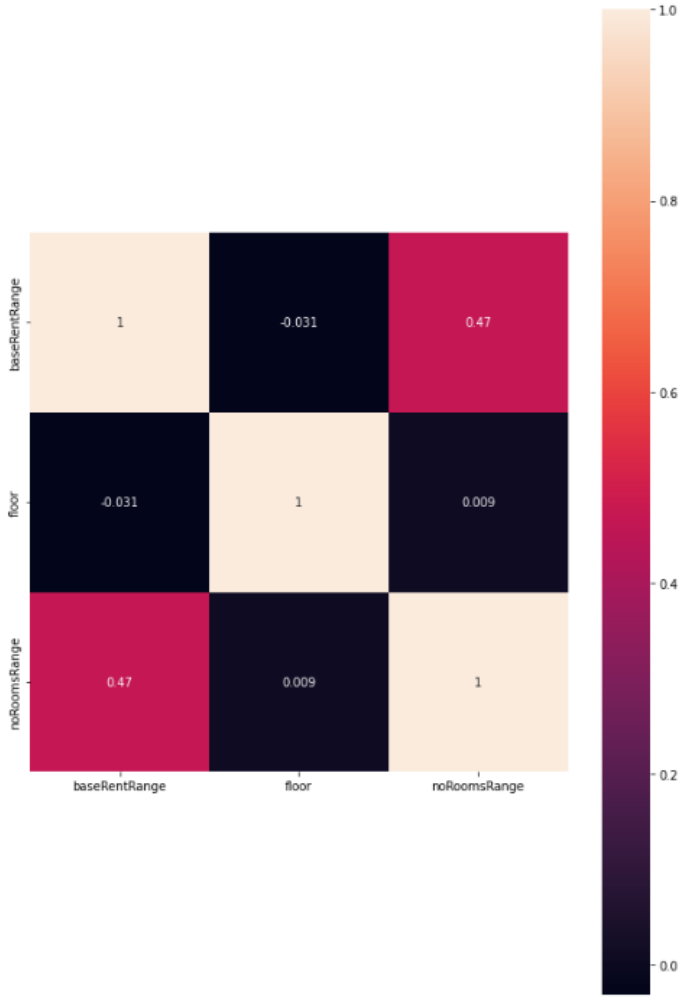
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 268850 entries, 0 to 268849
Data columns (total 49 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   regio1                    268850 non-null  object
 1   serviceCharge             261941 non-null  float64
 2   heatingType               223994 non-null  object
 3   telekomTvOffer            236231 non-null  object
 4   telekomHybridUploadSpeed  45020 non-null   float64
 5   newlyConst                268850 non-null  bool
 6   balcony                   268850 non-null  bool
 7   picturecount              268850 non-null  int64
 8   pricetrend                267018 non-null  float64
 9   telekomUploadSpeed        235492 non-null  float64
 10  totalRent                 228333 non-null  float64
 11  yearConstructed           211805 non-null  float64
 12  scoutId                   268850 non-null  int64
 13  noParkSpaces              93052 non-null   float64
 14  firingTypes               211886 non-null  object
 15  hasKitchen                268850 non-null  bool
 16  geo_bln                   268850 non-null  object
 17  cellar                    268850 non-null  bool
 18  yearConstructedRange      211805 non-null  float64
 19  baseRent                  268850 non-null  float64
 20  houseNumber               197832 non-null  object
 21  livingSpace               268850 non-null  float64
 22  geo_krs                   268850 non-null  object
 23  condition                 200361 non-null  object
 24  interiorQual              156185 non-null  object
 25  petsAllowed               154277 non-null  object
 26  street                    268850 non-null  object
 27  streetPlain               197837 non-null  object
 28  lift                      268850 non-null  bool
 29  baseRentRange             268850 non-null  int64
 30  typeOfFlat                232236 non-null  object
 31  geo_plz                   268850 non-null  int64
 32  noRooms                   268850 non-null  float64
 33  thermalChar               162344 non-null  float64
 34  floor                     217541 non-null  float64
 35  numberOfFloors            171118 non-null  float64
 36  noRoomsRange              268850 non-null  int64
 37  garden                    268850 non-null  bool
 38  livingSpaceRange          268850 non-null  int64
 39  regio2                    268850 non-null  object
 40  regio3                    268850 non-null  object
 41  description               249103 non-null  object
 42  facilities                215926 non-null  object
 43  heatingCosts              85518 non-null   float64
 44  energyEfficiencyClass     77787 non-null   object
 45  lastRefurbish             80711 non-null   float64
 46  electricityBasePrice      46846 non-null   float64
 47  electricityKwhPrice       46846 non-null   float64
 48  date                      268850 non-null  object
dtypes: bool(6), float64(18), int64(6), object(19)
```

همچنین وضعیت داده‌های null دیتاست بررسی‌شده به شکل زیر است:

```
[6]: regio1                      0.000000
     serviceCharge               0.025698
     heatingType                 0.166844
     telekomTvOffer              0.121328
     telekomHybridUploadSpeed    0.832546
     newlyConst                  0.000000
     balcony                     0.000000
     picturecount                0.000000
     pricetrend                  0.006814
     telekomUploadSpeed          0.124077
     totalRent                   0.150705
     yearConstructed             0.212182
     scoutId                     0.000000
     noParkSpaces                0.653889
     firingTypes                 0.211880
     hasKitchen                  0.000000
     geo_bln                     0.000000
     cellar                      0.000000
     yearConstructedRange        0.212182
     baseRent                    0.000000
     houseNumber                 0.264155
     livingSpace                 0.000000
     geo_krs                     0.000000
     condition                   0.254748
     interiorQual                0.419063
     petsAllowed                 0.426160
     street                      0.000000
     streetPlain                 0.264136
     lift                        0.000000
     baseRentRange               0.000000
     typeOfFlat                  0.136187
     geo_plz                     0.000000
     noRooms                     0.000000
     thermalChar                 0.396154
     floor                       0.190846
     numberOfFloors              0.363519
     noRoomsRange                0.000000
     garden                      0.000000
     livingSpaceRange            0.000000
     regio2                      0.000000
     regio3                      0.000000
     description                 0.073450
     facilities                  0.196853
     heatingCosts                0.681912
     energyEfficiencyClass       0.710668
     lastRefurbish               0.699792
     electricityBasePrice        0.825754
     electricityKwhPrice         0.825754
     date                        0.000000
     dtype: float64
```

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

[43… <AxesSubplot:>



که ارتباط میان فیچر های baseRentRange، floor، noRoomsRange را نشان می‌دهد.