به نام خدا

گزارش تمرین سری اول – قسمت اول درس داده کاوی

نام و نام خانوادگی: پویا شاعری

شماره دانشجویی: 400422105

لینک Colab:

https://colab.research.google.com/drive/1t1T1LpOPuxfCUYjpWKpYwUMCgcxiBTY0?usp=sharing

چکیده

در این گزارش قصد داریم مرحله به مرحله کارهای انجام شده را توضیح دهیم و تحلیل کنیم. گرچه در داخل نوت‌بوک، به صورت تکست بالای هر سلول توضیح داده شده است ولی ضروری است توضیحات و تحلیل های لازم و مقایسه ها را در اینجا داشته باشیم.

مقدمه

در ابتدا با توجه به آموزش داده شده در صورت تمرین، دیتاست، توسط سلول هایی، از سایت کگل مستقیم خوانده و در df_original ذخیره شد. آن را نمایش دادم و نگاهی اجمالی به داده کرده و از پایین سمت چپ دیتاست، shape آن را نگاه کردم. در سلول بعد از روی آن کپی گرفتم و از آنجا به بعد روی کپی به نام df کار کردم، این کار هم با copy(). و هم با assign کردن امکان‌پذیر است (تا یک بار). این کار به این دلیل است که اگر جایی اشتباه شد و df از دست رفت، از سلول کپی گرفتن دوباره کار را از سر بگیرم و به بالای بالا مراجعه نکنم.

بخش اول: پیش‌پردازش داده ها

در سلول بعدی بااستفاده از تابع ()info. روی df، اطلاعات کلی از اسم ستون ها و تعداد غیر null ها و Dtype داده ها (در هر ستون) و وضعیت حافظه را مشاهده کردم.

هندل کردن میسینگ‌ها: در سلول بعدی با استفاده از ()df.isna().sum تعداد null های هر ستون‌را نگاه کردم و در سلول بعدی ستون هایی که بیش از 50٪ آن ها missing بودند، حذف کردم که در اینجا چنین ستونی موجود نبود. در این قسمت هدف این است که تا جایی که امکان دارد با استدلال منطقی به ستون ها دست ببریم و اطلاعات را اصلاح کنیم. مثلا با دقت به دیتاست پی بردم که آن رکوردهایی که تعداد review آنها صفر بود ، تعداد کامنت ها در ماه و تاریخ آخرین کامنت آنها null بود، پس اگر روی ستون تعداد کامنت ها در ماه و تاریخ آخرین کامنت با میانگین گرفتن، میسینگ ها را هندل کنیم، برای تعداد کامنت دریافتی 0، بطور میانگین یک مقدار بزرگتر از صفر به عنوان تعداد کامنت ها در ماه که ستون عددی است دریافت می‌کردیم و این منطقی نیست. پس گفتم جاهایی که تعداد کامنت دریافتی برابر صفر بود، تعداد کامنت ها در ماهش رو صفر بذار. این کار را می توانستم با گروپ بای هم انجام دهم یعنی فقط گروه صفر را نگه دارم یا با اپلای کردن یک تابع ولی با for زدم. ستون تاریخ آخرین کامنت DateTime می تواند باشد ولی آبجکت در نظر گرفته شده. آن را تبدیل به DateTime کردم. سپس ستون های id و name و host_name را دراپ کردم چون نیازی نداشتم و با ستون host_id ، به مشخصه آگهی گذار دسترسی دارم و نال های عددی را با میانگین ستون آنها جایگزین کردم. سپس با info گرفتن می‌بینیم که با انجام تبدیل DateTime و کار های گفته شده حافظه کاهش یافته است و در این قسمت میبینیم که داده های موجود دیگر میسینگ ندارند بجز در last_review که در مقادیری که کامنت آن ها 0 بوده خب طبیعتا نباید تاریخ آخرین کامنت هم داشته باشند، میتوان این ها را با مد گرفتن هندل کرد، ولی منطقا اشتباه است! پس NaT یعنی null داده های last_review را خود به عنوان یک نوع در نظر گرفتم و به آن دست نزدم چرا که قبل از مدل زدن چند ستون مثل طول و عرض جغرافیایی و همچنین این ستون تاریخ قرار است دراپ شوند.

در ادامه مقادیر یونیک را گرفتیم و تعدادشون رو محاسبه کردیم (nunique) و تحلیل کردیم که ستون neighbourhood که آدرس است، 221 مقدار یونیک دارد و این خوب نیست، چراکه ستون neighbourhood_group با 5 مقدار unique به صورت محله به محله کلی تر همین کار را می کند و تبدیل کردن آن به ستون های عددی با وان‌هات به تعداد 5 (یا شاید 4 ستون اگر همه 0 را مثلا یک گروه دیگر در نظر بگیریم) به صرفه تر از 221 است. پس این ستون دراپ شد. و در ادامه یک بررسی آماری روی دیگر ستون های categorical انجام شد (value های آن ها).

```
for cols in df.columns:
    if df[cols].dtype == 'object' or df[cols].dtype == 'bool':
        print('------------------------------')
        print('cols : {} ,\n{}'.format(cols,df[cols].value_counts()))
```

```
------------------------------
cols : neighbourhood_group ,
Manhattan        21661
Brooklyn         20104
Queens            5666
Bronx             1091
Staten Island      373
Name: neighbourhood_group, dtype: int64
------------------------------
cols : room_type ,
Entire home/apt    25409
Private room       22326
Shared room         1160
Name: room_type, dtype: int64
```

در ادامه سه سلول به صورت کامنت شده داریم. کاری که انجام داده بودم این بود که مانند آموزش شما مثلا خواستم تا به جای 5 دسته، 3 دسته در ستون neighbourhood_group داشته باشم چون سه نوع آخر این ستون (یعنی Queens و Bronx و Staten Island) اگر درصد value nunique میگرفتیم در مقایسه با دو دسته Manhattan و Brooklyn بسیار کمتر بودند و در این سلول ها سه نوع آخر این ستون را به عنوان other در این ستون ذخیره کردم. ولی طی صحبتی که با شما داشتم و به این دلیل که در ادامه و تحلیل های آینده تسک 2 نیاز به آن ها دارم، آن را نگه داشتم. اما کد این عملیات را به صورت کامنت برای بازبینی شما داخل نوتبوک نگه داشتم.

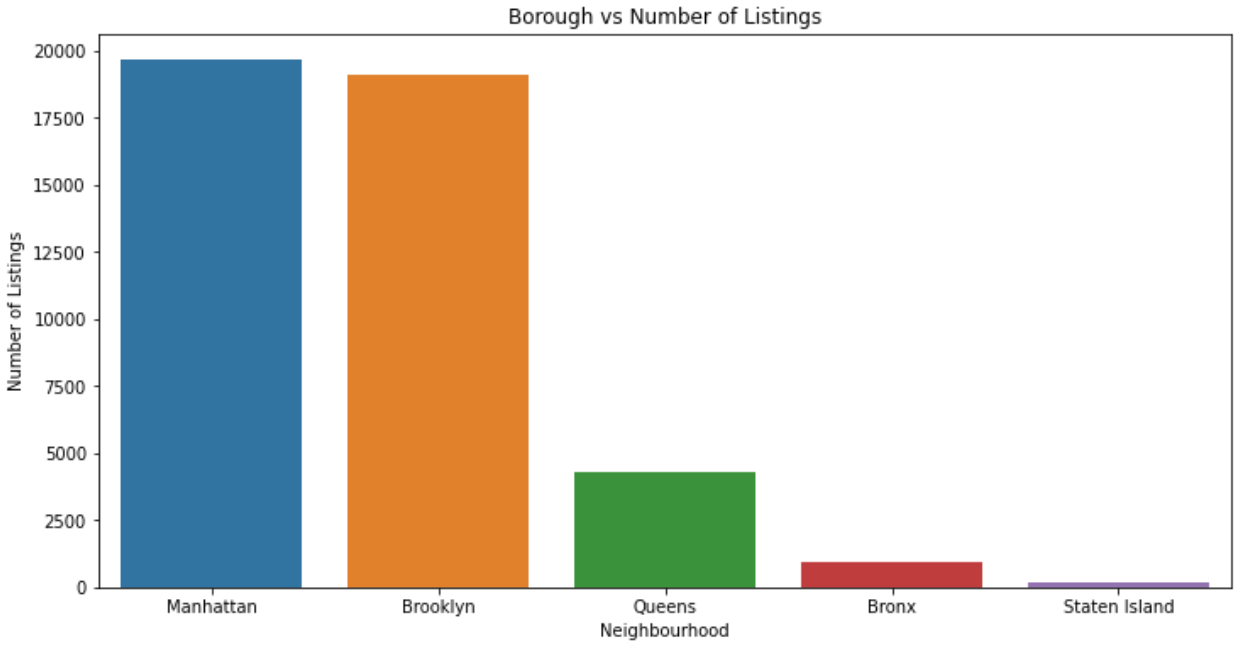
حال می‌خواهیم دو ستون categorical (['neighbourhood_group', 'room_type']) را numeric کنیم. این کار را با دستور get_dummies و سپس concat انجام دادیم.

سپس با روش z-score با 3 برابر std داده های پرت رو حذف کردیم.

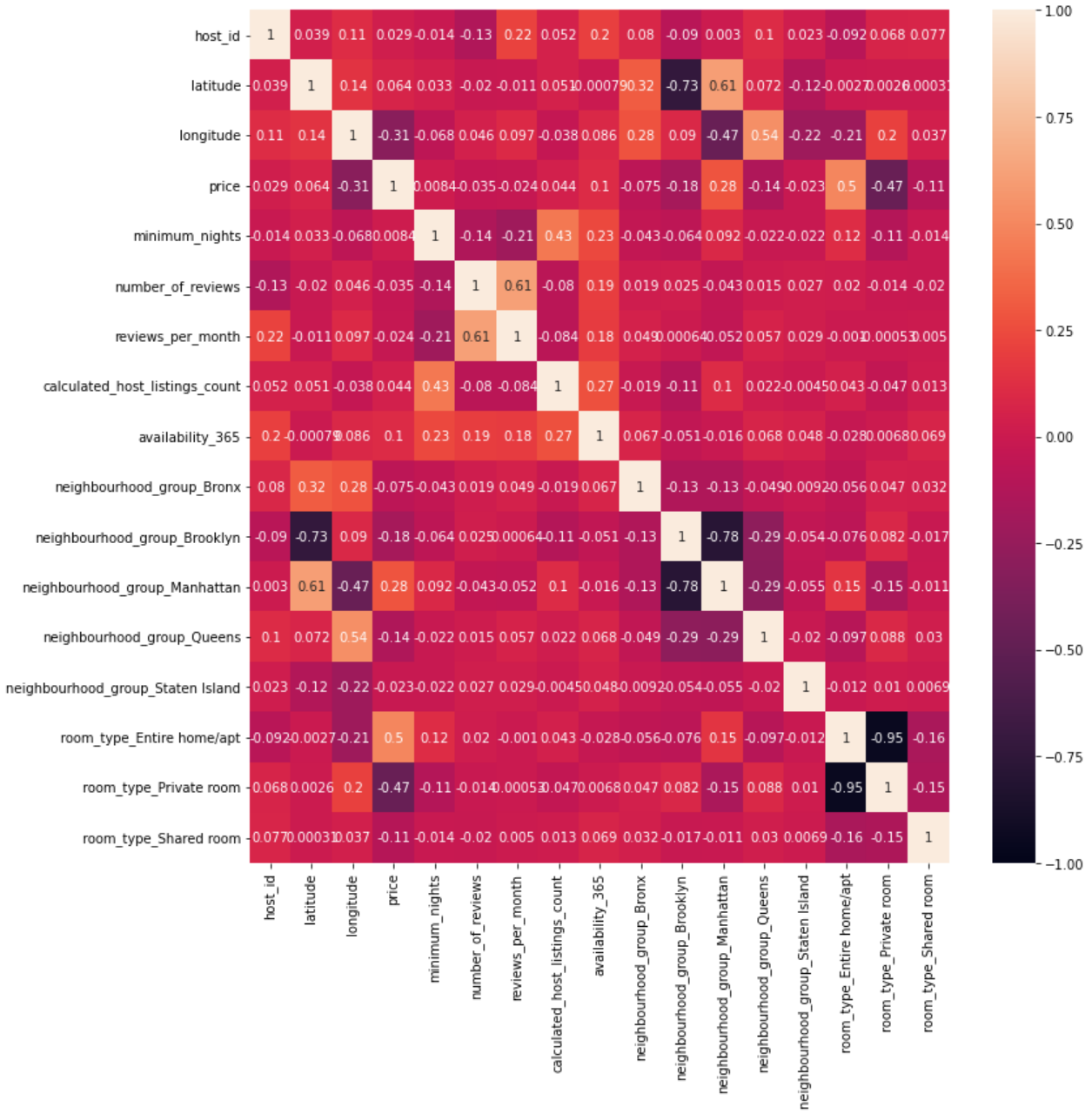
بخش دوم: تحلیل آماری داده ها و مصورسازی

با دستور describe() کل ستون های عددی بررسی شد : مینیمم ماکزیمم میانگین و ... . اگر ستون های categorical داشتیم که داریم باید حواسمان باشد که روی این ستون ها مجزا describe بزنیم که مد و فراوانی و ... را ببینیم ولی ستون های categorical باقی مانده را در قبل بررسی کردیم.

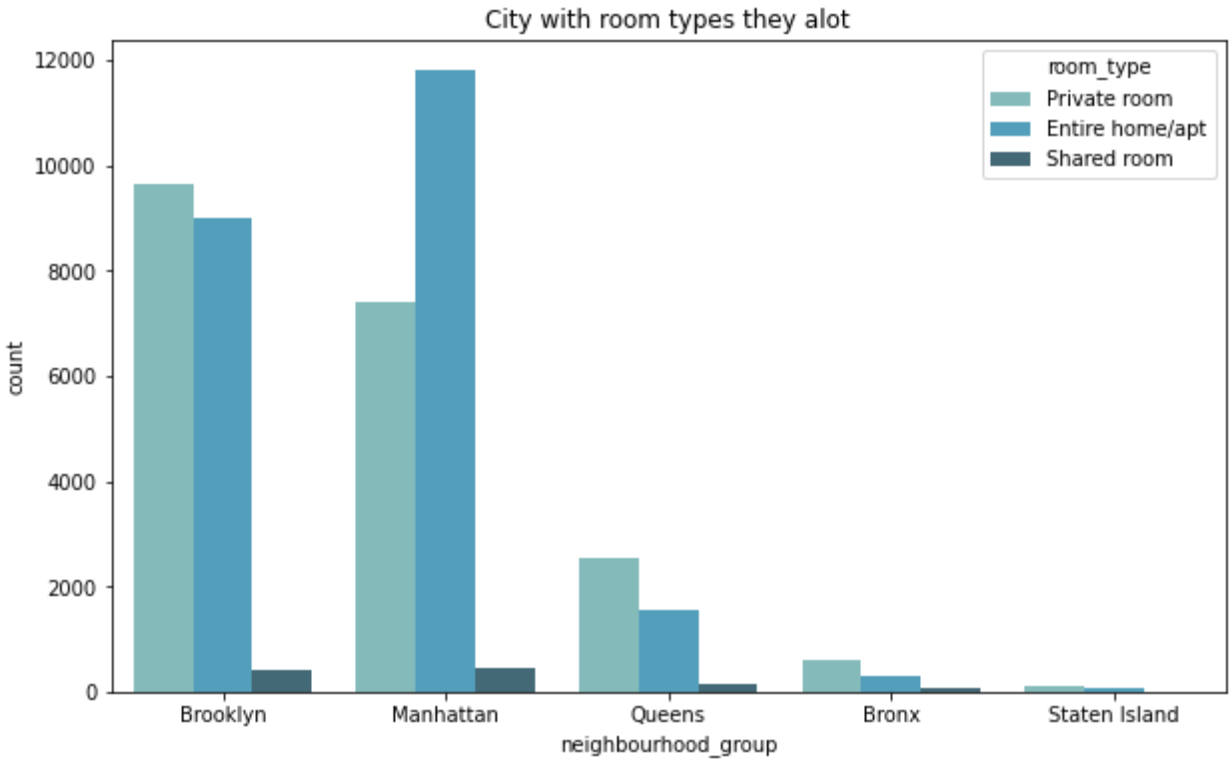
تعداد آگهی ها در مناطق جغرافیایی بررسی کردیم و نمودار میله ای محله برحسب تعداد آگهی رسم شده است.



ماتریس کوریلیشن بین فیچر ها رسم شد.



و نمودار دیگر که رسم شده، نمودار میله ای محله بر حسب نوع اتاق.



بخش سوم

محاسبه اینکه چند آگهی برای یک نفر است: این کار با value_counts() گرفتن روی ستون host_id انجام شد. از آنجا که host_id ها همگی unique بودند این مقدار قابل اعتماد است.

```
137358866    103
16098958      96
12243051      96
61391963      91
22541573      87
             ...
545273         1
7609268        1
30789837       1
25233652       1
68119814       1
Name: host_id, Length: 34837, dtype: int64
```

بخش چهارم

اگر تعداد کامنت های برای یک آگهی را بتوان شاخصی از تعداد مشتریان در نظر گرفت مطلوب است یافتن صاحبان آگهی که بیشترین مشتری را دارا می باشند و بررسی علت های آن. برای اینکار با استفاده از متد sort_values و قراردان by آن برابر number_of_reviews و ترتیب نزولی، دیتافریمی بدست آوردیم که براساس تعداد کامنت ها به صورت نزولی مرتب شده اند. حالا host_id این دیتا فریم را نمایش می‌دهیم.

	host_id	number_of_reviews
152	142684	155
4280	2074433	155
7578	22020703	155
21292	2788934	155
18062	85375269	155
...	...	...
27546	4696622	0
27549	157825168	0
27551	102685703	0
27555	2179407	0
48894	68119814	0

44123 rows × 2 columns

بخش پنجم: آزمون های فرض

در این بخش آزمون های فرض مطرح شده تقریبا مشابه آموزش شما در لینک کلب عمل کرده ام و مفروضاتی را در نظر گرفتم و با روش های مختلف این فرض ها را آزموده ام. ولی در آزمون اول یک سری کار ها کردم مثلا موضوع مورد بحث میانگین کرایه خانه بود و من در سلول های مختلف یکبار با پارامتر آلفا بازی کردم و یکبار با استفاده از میانگین واقعی خود آزمون فرض را آزمایش

کردم. همچنین دیگر تفاوت کاری من با شما، این بود که آزمون فرض 5م را از سایت پایتون چیتکد که در ویدیو فرمودید برای بررسی distribution از الگوریتم کروسکال موجود در آن سایت استفاده کردم. به طور کلی برای توضیح این بخش: ما یه فرض در نظر میگیریم سپس با آزمون فرض، با یه تقریبی یا ردش میکنیم یا می پذیریمش! بالای هر سل مثل شما نوشتم که رابطه چی رو با چی در نظر گرفتم یا آزمون فرضم چی هست ...

بخش ششم: مدلسازی

در این بخش می‌خواهیم با مدلسازی، ستون price رو پیشبینی کنیم. در اینجا ابتدا چند ستون که به کار مدلسازی نمی آیند، حذف کردیم. سپس با نرمال کردن اسکیل خود لایبراری و اسپلیت تست و ترین داده ها، رگرسین خطی را روی آن ها اعمال کردیم و خطا را بدست آوردیم.

در ادامه خودم نیز به پیاده سازی GD مانند شما پرداخته ام و خطا ها را پرینت کردم و برای آن نمودار خطا نیز رسم شده است.