به نام خداوند بخشنده و مهربان

موضوع : Newyork city airBNB

نام اساتید : جناب آقای دکتر فراهانی / جناب آقای دکتر خرد پیشه

شکیلا جابری
۹۹۴۲۲۰۴۶

گزارش کار تمرین ۱

با توجه به **Data** هایی که داده شده چند سؤال مطرح است .

سؤال اول در خصوص تحلیل داده ها و آشنایی با کلیت برنامه نویسی پایتون در صورت تمرین ها سؤال شده است .

سؤالی که مطرح است ؛ آیا قیمت اجاره خانه به منطقه آن بستگی دارد یا خیر؟

برای پاسخ به این سؤال از تست **Anova** استفاده میکنیم .

در خصوص سؤال اول که در مورد آشنایی اولیه با آموزش پایتون است :

پایتون یک زبان مفسری است یعنی در هر خط کدی که میزنیم باید **run** بگیریم و **error** ها را برطرف کنیم ، چون پایتون مانند **C++** و **Java** کامپایلری نیست.

نکته : دیگر اینکه تمام امکاناتی که زبان های دیگر در اختیار کاربران قرار میدهند ، پایتون هم قرار میدهد (بسیار راحت تر و سریع تر)

یاد گرفتیم که در پایتون یکسری توابع پایه داریم مثل **nampay** و **pandas** که در داده کاوی مورد نیاز است .

در خصوص ماهیت **data** میتوان گفت که هر کدام از نمودار ها برای یک دستور خاص استفاده میشود و برای یکسری از کار ها امکان دارد چندین نمودار داشته باشیم .

برای رسم نمودار ها نیاز به استخراج اطلاعات از **data set** داریم .

گفتیم که به ۲ پکیج نیاز داریم :

Nampay : برای کار با **data** نیاز میباشد .

Pandas : برای کار های **processing** احتیاج است .

در قدم اول باید با یک فایل csv ، data را بخوانیم ، با این کار یک دیدی نسبت به data پیدا میکنیم ← خط دوم

توضیح خط دوم : یکی از دستوراتی که در python میتواند به ما کمک کند دستور info است که مشخص میکند هر کدام از ficher ها به چه صورت هستند (یا categorical اند و یا Bolion و یا ...) و این که چه تعداد از data های ما غیر null هستند .

این دید خیلی خوبی به ما میدهد ، null بودن نوعی اطلاعات است ، نکته ای که وجود دارد این است که مرسوم است که ficher هایی که ۵۰٪ اطلاعات آن ها null است را حذف میکنیم ، اما باید تست شود چون ممکن است null بودن data هم برای ما با ارزش باشد .

برای مدیریت data های null چند راه حل وجود دارد :

یکی اینکه با مقدار میانگین ، مقادیر data های null را پر میکنیم . با دستور df.mean به سادگی میتوانیم ficher ها را دریافت کنیم اما مشکلی که بوجود می آید این است که اگر data set ما بزرگ باشد وقتی دستور df.mean را میزنیم چون یکی یکی چک میکنیم که مثلا کدام یک از data های ما object هستند ، روند اجرا را کند میکند برای همین با دستور df.get_numeric_data فقط data های numeric را فیلتر میکنیم و بعد میانگین میگیریم که این کار باعث افزایش سرعت کار میشود.

با دستور df.isna میتوانیم بفهمیم که کدام یک از ستون های ما بولین به ما برمیگرداند به این معنیست که کدام یک ficher ها پر و کدام خالی هستند و وقتی sum را میزنیم True و False را تبدیل به صفر و یک کرده و بعد جمع میکند ، اگر حاصل جمع برابر با یک شود data را null معرفی میکند .

پس با **sum** متوجه میشویم که چه تعداد از **data** ها **null** هستند ← توضیح خط پنجم

با دستور **df.price** هم قیمت اجاره خانه را مشخص میکنیم ← خط هشتم

که این قیمت باید بزرگتر از صفر قرار بگیرد چون امکان ندارد که قیمت اجاره خانه صفر باشد و خانه را رایگان اجاره دهند ، اگر بزرگتر از صفر نبود یعنی ایرادی وجود دارد و باید اصلاح شود یا با میانگین قیمت پر شود و یا با استفاده از مدل پیاده سازی کنیم.

پس اگر **mean price** ما صفر باشد یعنی **data** های ما مشکل دارند ، یا همین **standard deviation** میتواند به ما دید بدهد که **data** های ما به چه صورت پراکنده شده اند ← خط نهم

توضیح خط دهم ← با دستور **len (data)** میتوانیم بفهمیم چه تعداد نمونه وجود دارد . میتوانیم از دستور **shape** هم استفاده کنیم که تعداد ستون ها را هم مشخص میکند و به ما اطلاع میدهد که باید یکسری کار آماری انجام دهیم / که الان ۴۸۸۸۴ **data** داریم.

در این قسمت **id** را باید از سیستم حذف کنیم چون اطلاعات ارزشمندی به ما نمیدهند .

خط یازدهم ← که در این جا ارزش های استاندارد سازی **data** استفاده کرده ایم .

ممکن است **data** ها را **normalite** کنیم ، ممکن است **data** ها **range** تغییرات مختلفی داشته باشند و باعث خراب شدن کار ما شوند ، برای حل این مشکل ،

data ها را از میانگین کم میکنیم و بر standard deriation تقسیم میکنیم که این کار را برای data های numeric انجام میدهیم .

اما نکته ی مهم دیگر out layer ها هستند که داده های پرت هستند و میتوانند در روند کار ما اختلال ایجاد کنند یا حتی آزمون های فرض ما را دچار مشکل کنند و پیچیدگی زیادی به مدل ما اضافه میکندد ، پس لازم داریم که این داده ها را از سیستم حذف کنیم ، یکی از راه های حذف داده های پرت این است که data هایی که سه برابر standard ، سه برابر بیشتر یا سه برابر کمتر هستند را از سیستم حذف کنیم و ی اینکه از روش IQ R این کار را انجام دهیم که با استفاده از چارک های اول و سوم داده های ما انجام میگیرد (ما از روش اول استفاده کرده ایم)

Visualitation ویژگی مثبتی دارد که باعث میشود با یک نگاه اجمالی اطلاعات زیادی کسب منیم .

توضیح خط چهاردهم ← neighbourhood-group : مناطق پنجگانه new-york را به ما نشان میدهد

- ✓ Bronx
- ✓ Brooklyn
- ✓ Manhattan
- ✓ Queens
- ✓ Staten Island

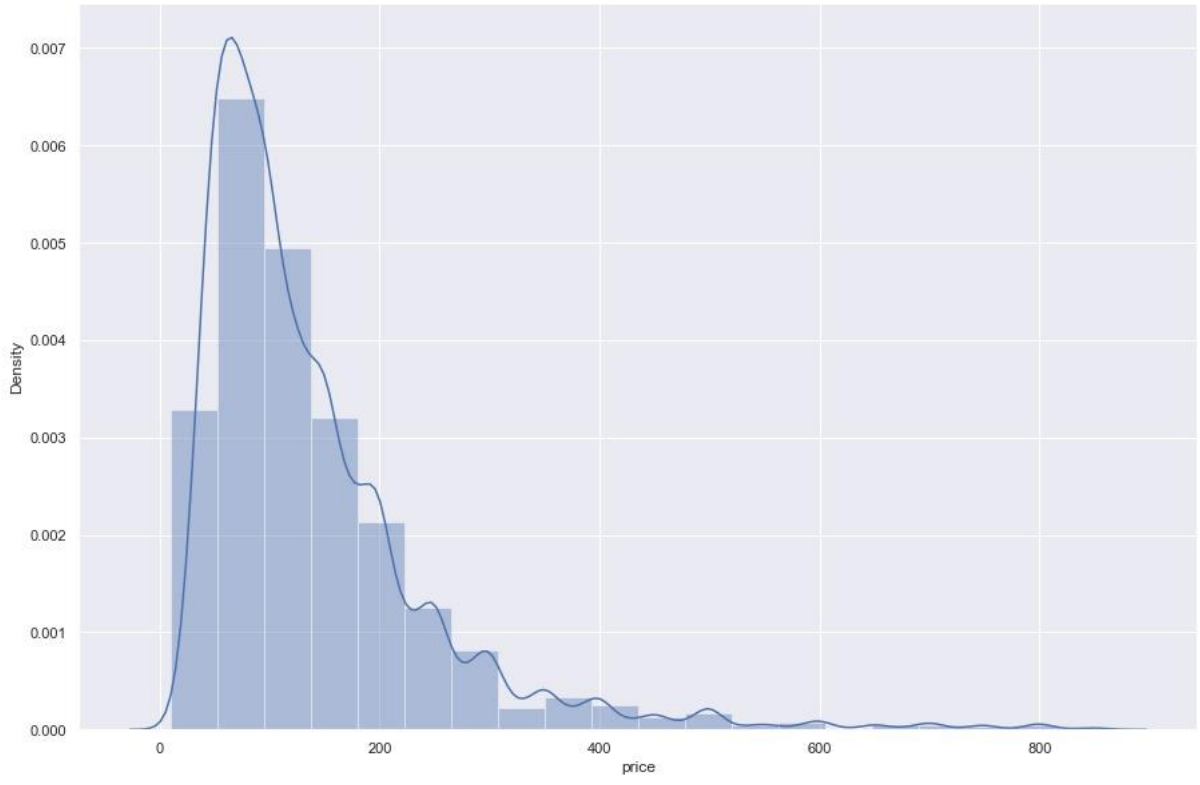
این دستور به ما کمک میکند داده ها را گروه بندی کنیم و بر اساس id جدا کردیم و گفتیم از این داده ها count بگیر (یعنی متوجه شویم در این مناطق پنجگانه چه تعداد request یا آگهی داشته ایم) چون id یک چیز uniqe میباشد.

در توضیح خط بیستم میرسیم به بحث **distribution** : در تمام کار های آماری که انجام میدهیم فرض ما بر این است که **data set** نرمال است ، اما باید بررسی شود ، که در این نمودار که با استفاده از **seaborn** کشیده شده این کار را انجام داده ایم که یکی از پکیج های مصور سازی **data** است.

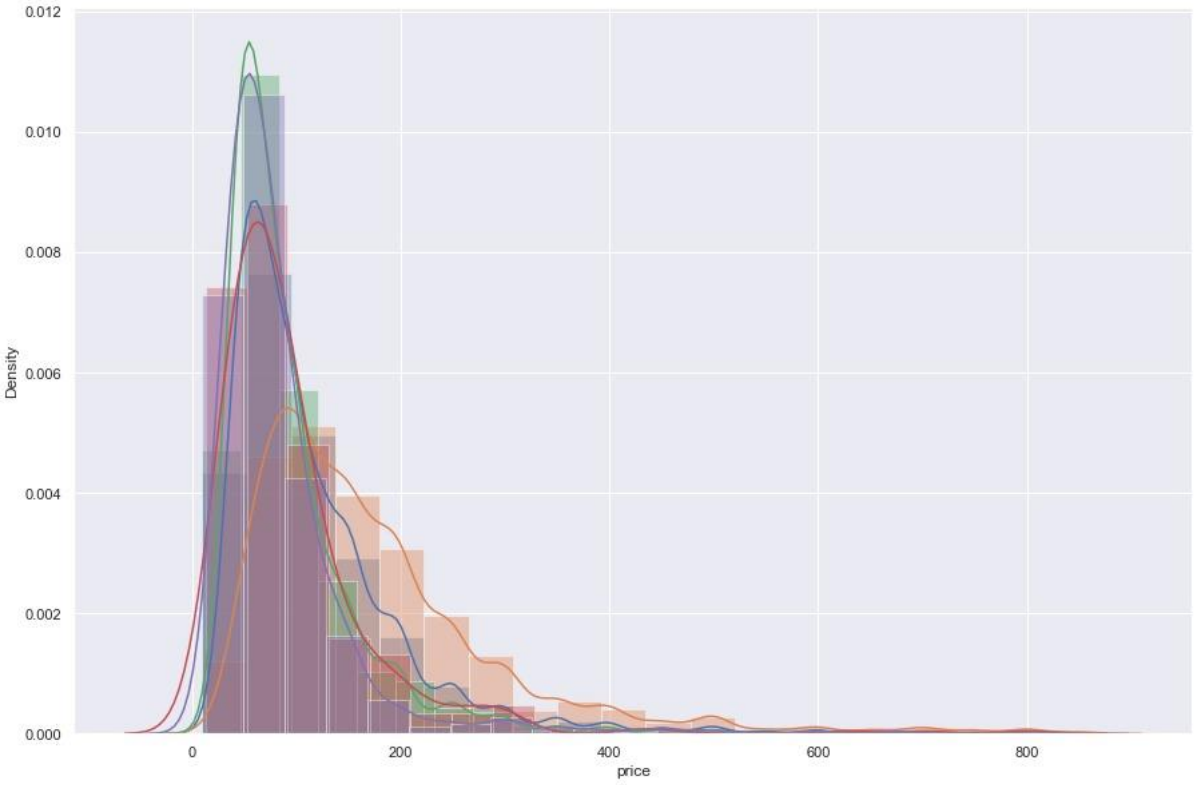
تمام هرف گذاری ما این است که روی **price** کار کنیم

یک هیستو گرام رسم کرده ایم ، که این هیستو گرام توضیح **data** های ما را رسم کرده است که توزیع نرمالی نیست و از یک طرف کشیدگی بیشتری دارد پس باید یکسری تغییرات اعمال کنیم تا داده های ما داده های نرمال تری شوند اما باز مشاهده میکنیم که نرمال نیست چون قرینگی نداریم و چولگی داریم .

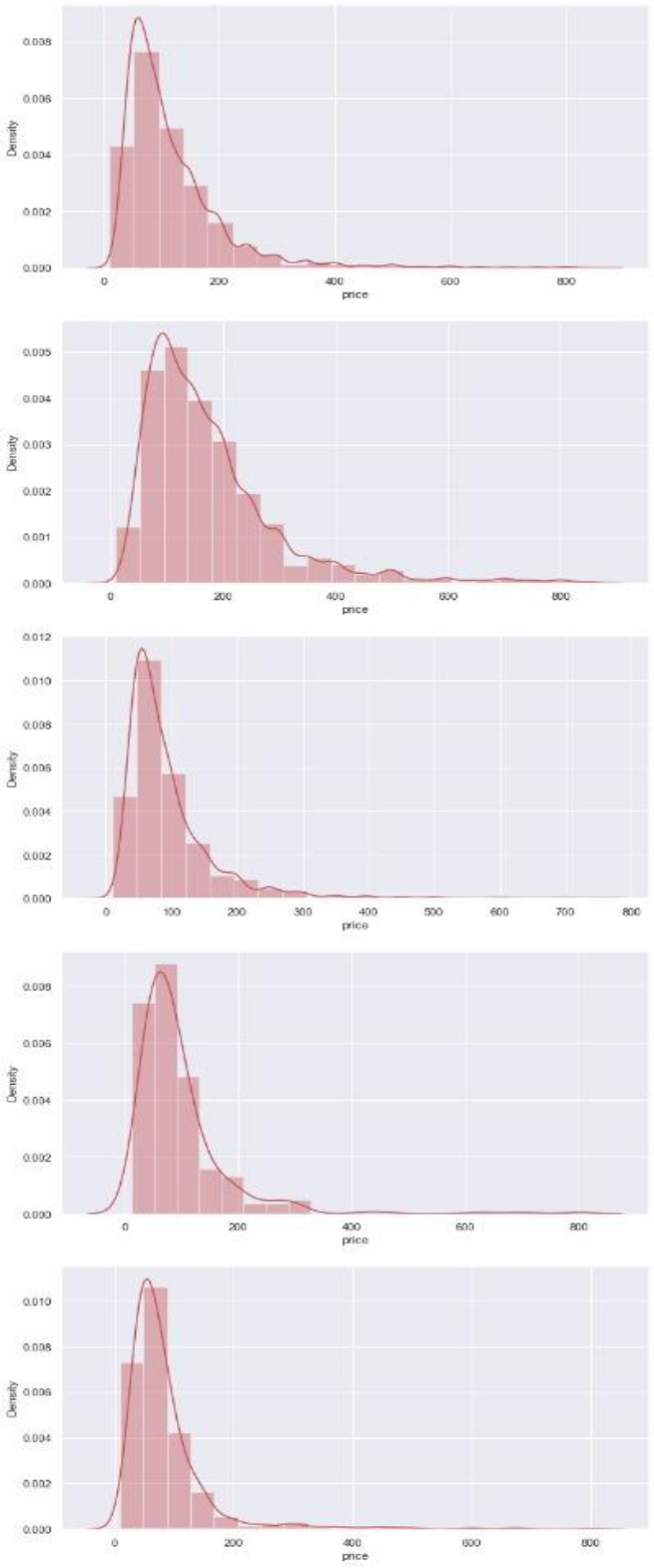
حال برای اینکه **data set** را به حالت نرمال نزدیک کنیم باید یکسری تغییرات اعمال کنیم مثلا لگاریتم بر پایه **2** ، رادیکال و یا توان **2** بگیریم که باید تست کنیم و ببینیم کدام یک کارآمد است.



نمودار بعدی هم همین توزیع ها را نشان میدهد اما اما ۵ تا توزیع را همزمان روی هم رسم کرده ایم که وضعیت نرمال بودن را بسنجیم که باز هم میبینیم که نرمال نیست اما برای سهولت در کار های آماری **data** را نرمال در نظر میگیریم.

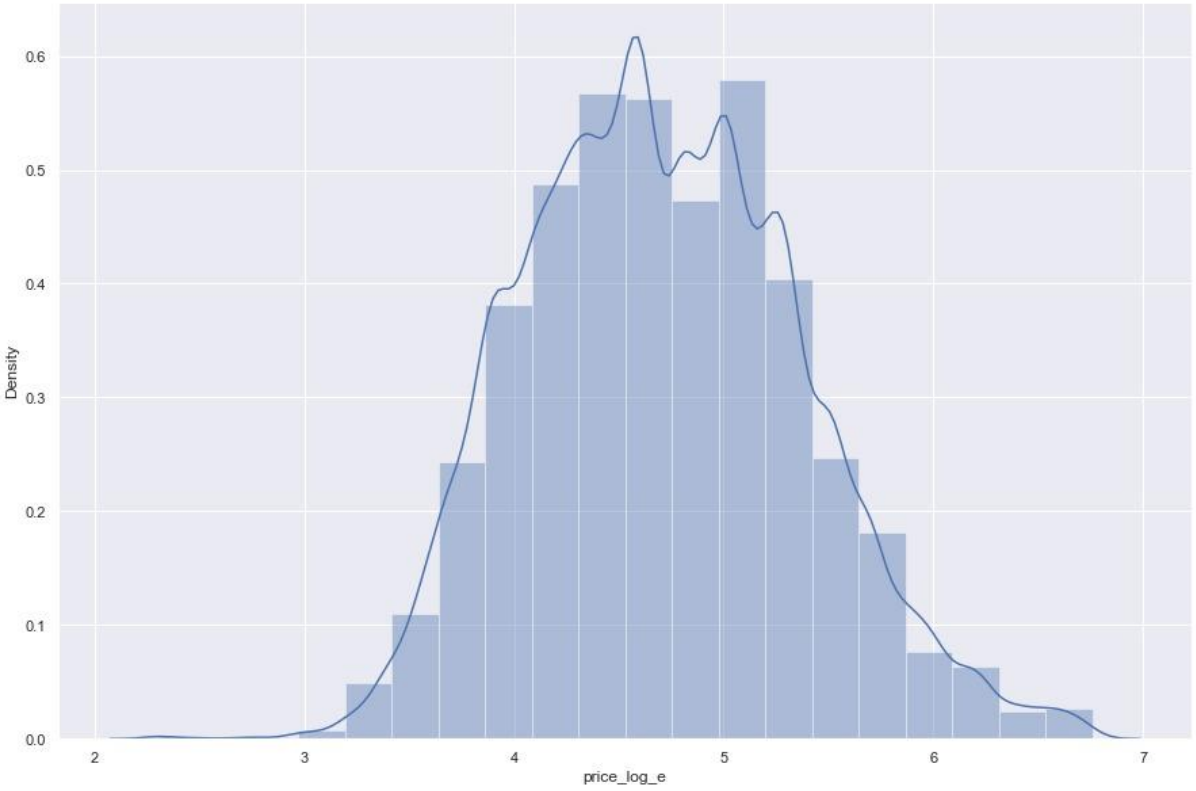


که ۵ نمودار صورتی بعد هم تک تک همین نمودار هاست که بصورت جداگانه ترسیم شده اند .



توضیح خط بیست و سوم ← این خط توضیح price-log میباشد با توجه به توضیحات قبل جهت نرمال سازی اینجا از log e استفاده کرده ایم ، پس ficher جدیدی ایجاد میکنیم بنام price-log-e که باز هم شبیه نرمال نیست اما بهتر از قبل است

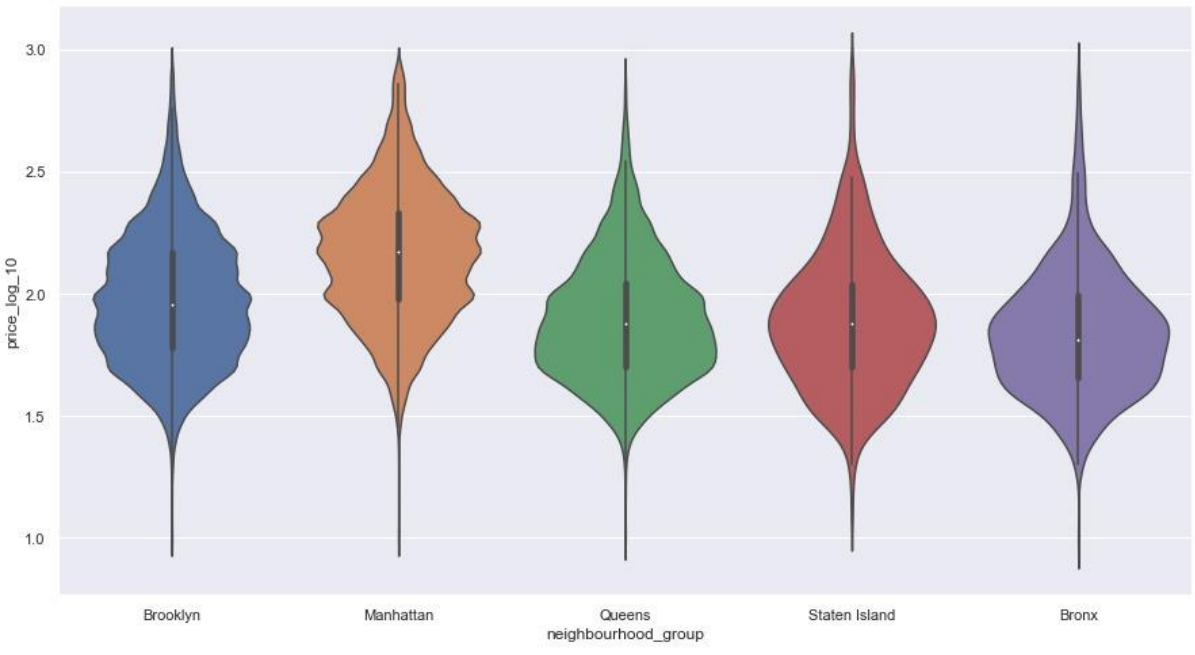
ممکن است نمودار ها بسیار به هم شبیه باشند برای تشخیص نرمال بودن توزیع در کتابخانه stats یک آزمون داریم بنام نرمال تست ؛ که یک آزمون فرض است ، چک میکنیم که آیا این data set توزیع نرمال دارد یا خیر ؟ که به ما یک p-value برمیگرداند که با استفاده از آن متوجه میشویم که توزیع نرمال میباشد یا خیر که با توجه به عدد بدست آمده توزیع ما نرمال نبوده اما برای راحتی کار نرمال در نظر میگیریم .



توضیح خط سی و سوم ← با استفاده از کد **plt.figure** نمودار ویالن را ترسیم میکنیم . یکسری آزمون فرض مطرح کردیم و به آنها پاسخ دادیم . سؤال اول این است که آیا واقعا **relation** بین مناطق پنجگانه نیویورک ، **price-log-10** وجود دارد یا خیر؟ بعبارت دیگر یعنی واقعا میشود گفت قیمت در مناطق پنجگانه نیویورک متفاوت هست و یا خیر؟

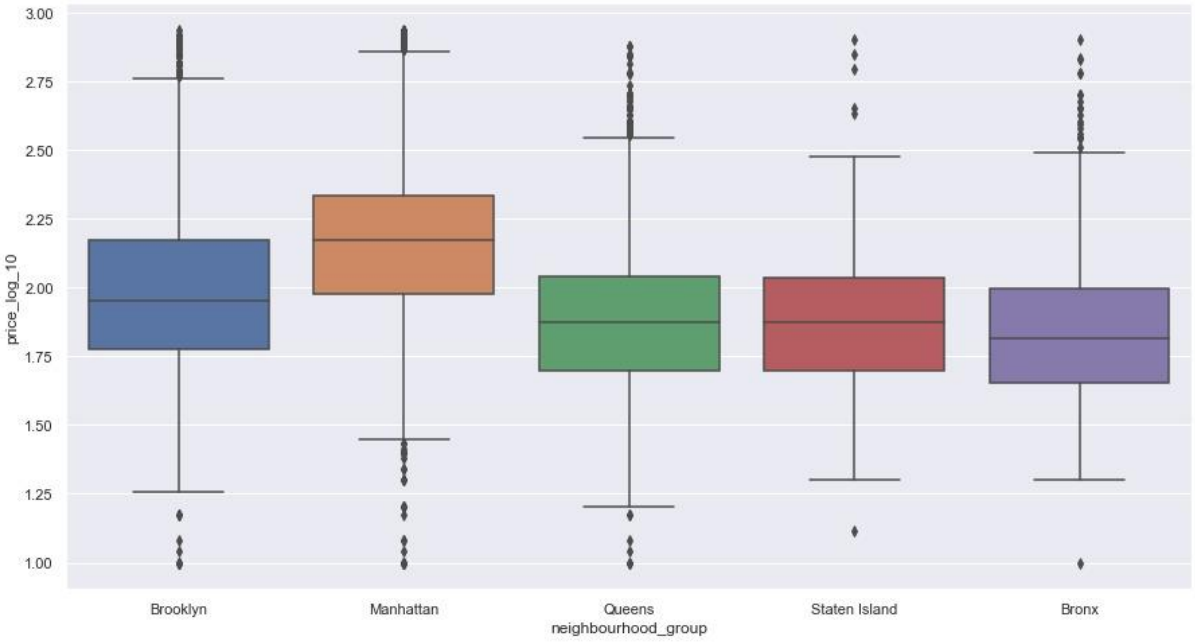
برای بررسی این سؤال نمودار ویالن را رسم میکنیم که این نمودار شبیه به نمودار **box-plot** است که به ما جایگاه **min** و **max** را نشان میدهد و فراوانی هر یک از مقادیر را به ما اعلام میکند . که نشان میدهد در مرکز شکل فراوانی بیشتری داریم .

این نمودار را با استفاده از **seaborn** رسم کرده ایم . طبق این شکل میانگین ها متفاوت است و باید از آزمون فرض استفاده کنیم.



توضیح خط سی و چهارم ← نمودار box-plot هم شبیه به نمودار ویالن است و اطلاعات ارزشمندی به ما میدهد .

با فرض نرمال بودن داده ها میخواهیم بررسی کنیم که این مناطق پنجگانه تاثیری در قیمت اجاره خانه دارد یا خیر؟ برای این کار از تست Anova استفاده میکنیم و از کتابخانه ی stats استفاده کرده ایم که به ما f و p-value را برمیگرداند که p-value صفر به ما برمیگرداند ، در نتیجه Hصفر ما نقض میشود یعنی Hصفر ما reject میشود :



همانطور که در جلسات کلاس آموختیم ، طبق مبحث

testing pairwise association

تا حالا یکسری سؤالات در مورد داده هایمان داشتیم و آزمون های فرض هم داشتیم حالا میتوانیم سؤالات جدیدی درمورد داده ها داشته باشیم و سعی کنیم برای آنها آزمون های فرض داشته باشیم که بتوانیم به صورت آماری و مبنی بر داده ها به سؤالات پاسخ دهیم

pairwise association یعنی آیا دو متغیر به یکدیگر ارتباط دارند یا خیر؟

که بیشتر برای متغیر های **categorical** بکار میرود یعنی متغیر هایی که چند حالت خاص داشتند مثل جنسیت ، مدرک تحصیلی

Ordinal variable هم یک متغیر **categorical** است که در آن ترتیب وجود دارد ، میتوانیم از تست هایی مثل **t-test** استفاده کنیم اگر **One sample** باشیم مثلا قد دانشجویان شهید بهشتی .

t-test فرض نرمال بودن داده ها را میخواهد اگر **one sample** بودیم اما فرض نرمال بودن را نداشتیم میتوانیم از **willcoxon test** استفاده کنیم

حال اگر **2** تا **sample** داشتیم مثلا قد آقایان و خانم ها ، میتوانیم از **Two sample test** با فرض نرمال بودن جامعه استفاده کنیم و اگر فرض نرمال بودن را نداشتیم از **monn whitney** استفاده کنیم

زمانی که بیشتر از دو تا **sample** داشتیم میتوانیم از

One way Anova استفاده کنیم (Anovaفرض نرمال بودن دارد ،
friedmon فرض نرمال بودن ندارد)

Analysis of Variance (Anova)

زمانی از Anova استفاده میکنیم که تعداد نمونه هایمان از 2 تا بیشتر باشد مثلا دانشگاه شریف ، دانشگاه شهید بهشتی ، دانشگاه تهران

سؤال : آیا میانگین قد 3 جامعه یکسان است یا خیر؟ فرض Hصفر فقط اپاسخ را میگوید اما نمیگوید که کدام بیشتر و کدام کمتر است

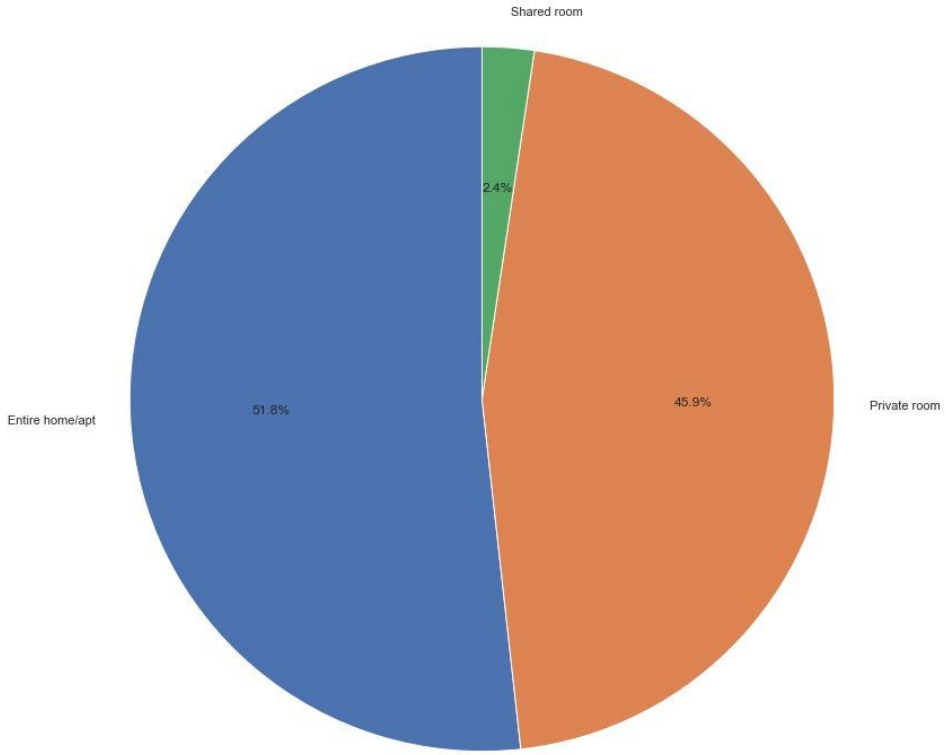
توضیح خط چهل ← در توضیح این خط از Pay chart استفاده کرده ایم که میرسیم به نمودار دایره ای که به ما این اطلاعات را میدهد :

51/8٪ از آگهی هایی که دریافت کرده ایم کل خانه را اجاره کرده اند

2/4٪ share room بوده اند

46 ٪ private room بوده اند

و همچنین میتوتن نتیجه گرفت که تراکم خانه در manhattan بیشتر بوده و در staten land تراکم private room بیشتر بوده که میتوانیم از bar chartهم استفاده کنیم



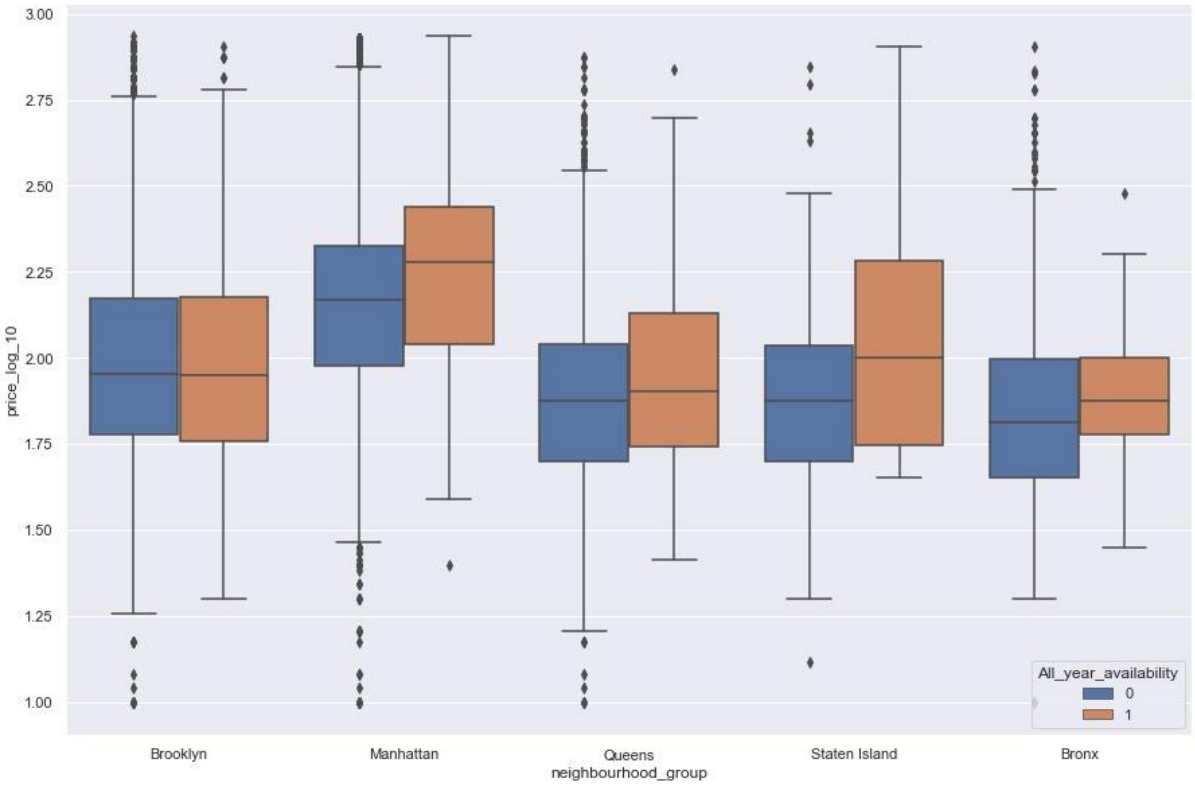
توضیح خط چهل و چهار ← در این خط خواستیم neighbor hood group را که ficher آن manhattan بوده مورد بررسی قرار دهیم یا مثلا با کد

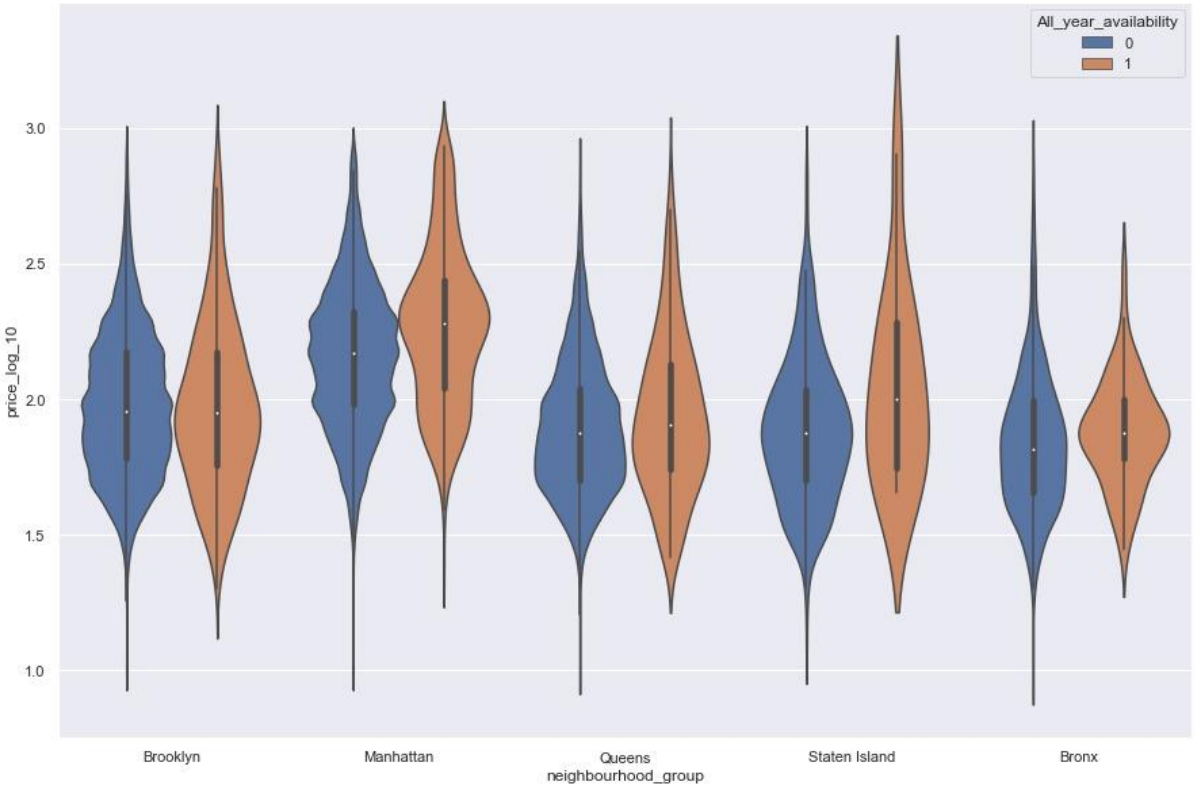
Neighbourhood = manhattan .room –type.value count

میتوانیم بفهمیم که در محله manhattan در نیویورک چه تعداد از اتاق ها هرکدام به چه صورت هستند .

توضیح کد 48 ← در این کد میخواهیم بررسی کنیم چه تعداد از آگهی ها در تمام 365 روز سال قابل اجاره بوده اند کد یک اختصاص گرفته اند و چه تعداد از آنها کد صفر اختصاص گرفته اند (قابل اجاره نبوده اند) و آنها را مصور سازی کرده ایم

و نمودار های ویالن پلات و باکی پلات در شماره کد های 50 و 51 میزان در دسترس بودن خانه در تمام ایام سال را نشان میدهد




All_year_availability
0
1
price_log_10
3.0
2.5
2.0
1.5
1.0
Brooklyn
Manhattan
Queens
Staten Island
Bronx
neighbourhood_group