نیما طاهری

1- Data cleaning

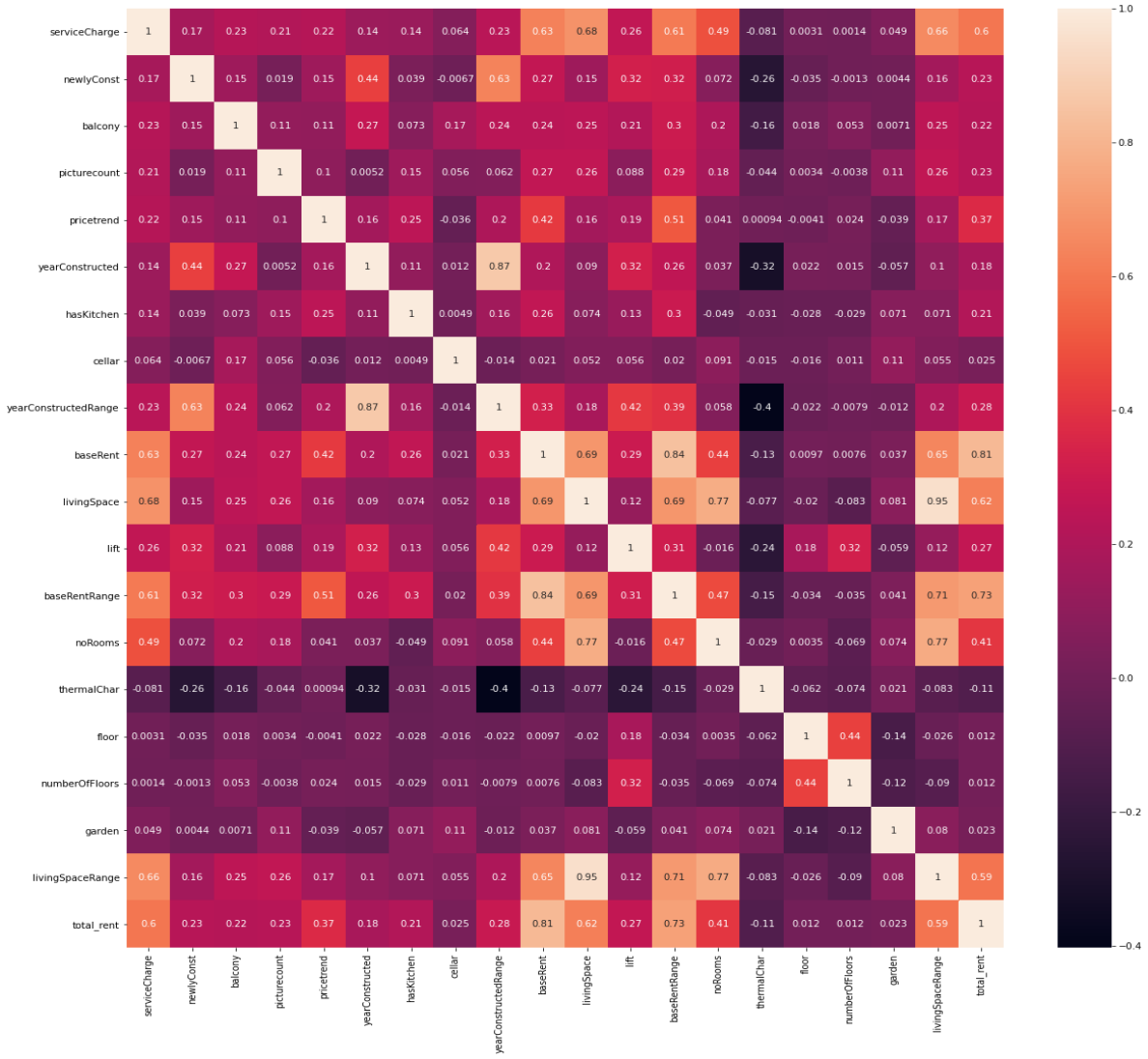
انجام پاکسازی داده ها: از (268850, 49) داده به (254677, 26) داده رسیدیم و همینطور که مشخص است 23 ستون اضافی حذف شده و حدود 14000 سطر هم در همان پاکسازی های اولیه حذف شده. مقادیری که وجود نداشتند با استفاده میانگین مقادیر دیگر پر کردیم و بعضی ستون های کتگوریکال هم که موارد زیادی داشتند گروهی از آنها ترکیب شدند و به اسم other شناخته میشوند.

2- Feature analyzing and data visualization

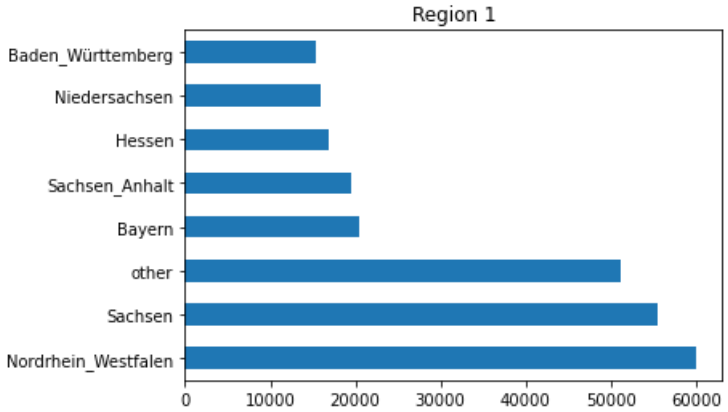
از تابع describe() مختص دیتافریم که در پکیج pandas قرار دارد برای برسی فیچر ها و اطلاعات آماری مربوط به آنها استفاده کردیم.

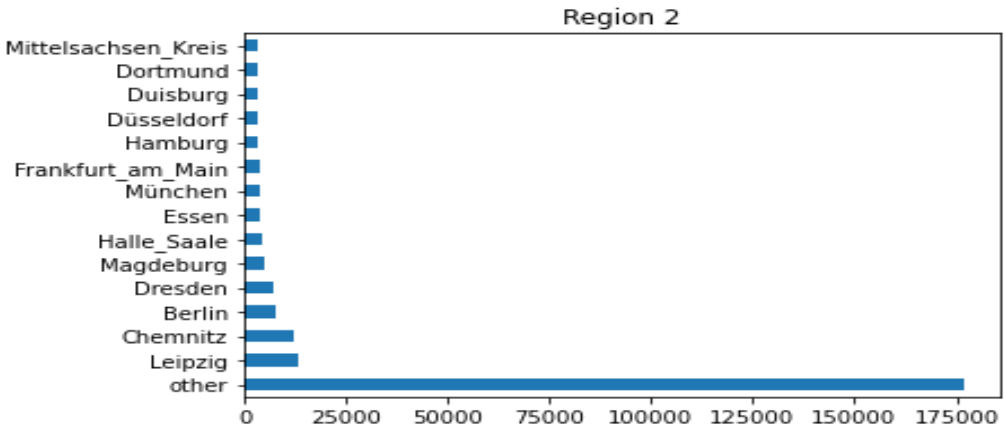
	serviceCharge	picturecount	pricetrend	totalRent	yearConstructed	yearConstructedRange	baseRent	livingSpace	baseRentRange	noRooms	thermalChar	floor	numberOfFloors	livingSpaceRange
count	254677	254677	254677	254677	254677	254677	254677	254677	254677	254677	254677	254677	254677	254677
mean	148.906198	9.391284	3.321719	814.69747	1967.901314	3.724828	633.902392	72.968869	3.715475	2.619179	112.564203	2.070356	3.50459	3.051096
std	81.108617	5.502439	1.853113	523.97402	33.682238	2.412433	486.086939	31.111296	2.187344	0.969525	35.914764	1.361657	1.37465	1.388935
min	0	0	-2.44	1	1842	1	0	1	1	1	0.1	-1	0	1
25%	96	6	1.98	491.72	1959	2	335	54	2	2	97.35	1	3	2
50%	139	9	3.37	730	1966.411388	3.715457	485	67	3	3	114.72574	2	3.574554	3
75%	183	12	4.51	902.26867	1992	5	784	86	5	3	116	3	4	4
max	1057.5	28	9.21	64651	2090	9	39200	649	9	10	258.5	11	19	7

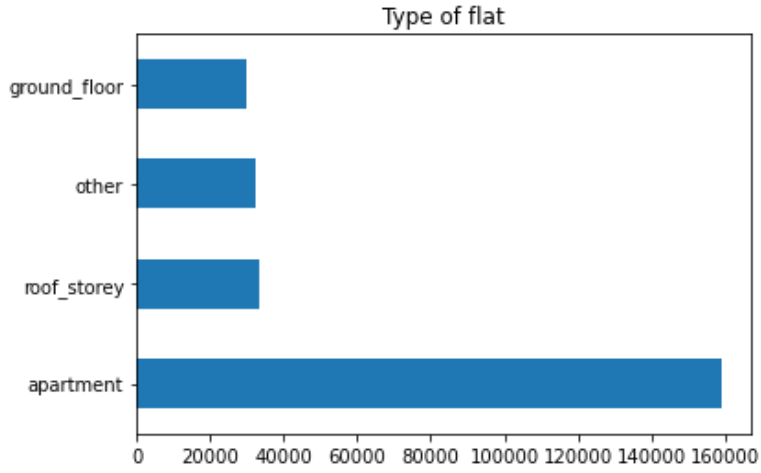
از heatmap برای نشان دادن ارتباط بین فیچر ها با هم و با قیمت اجاره استفاده کردیم.



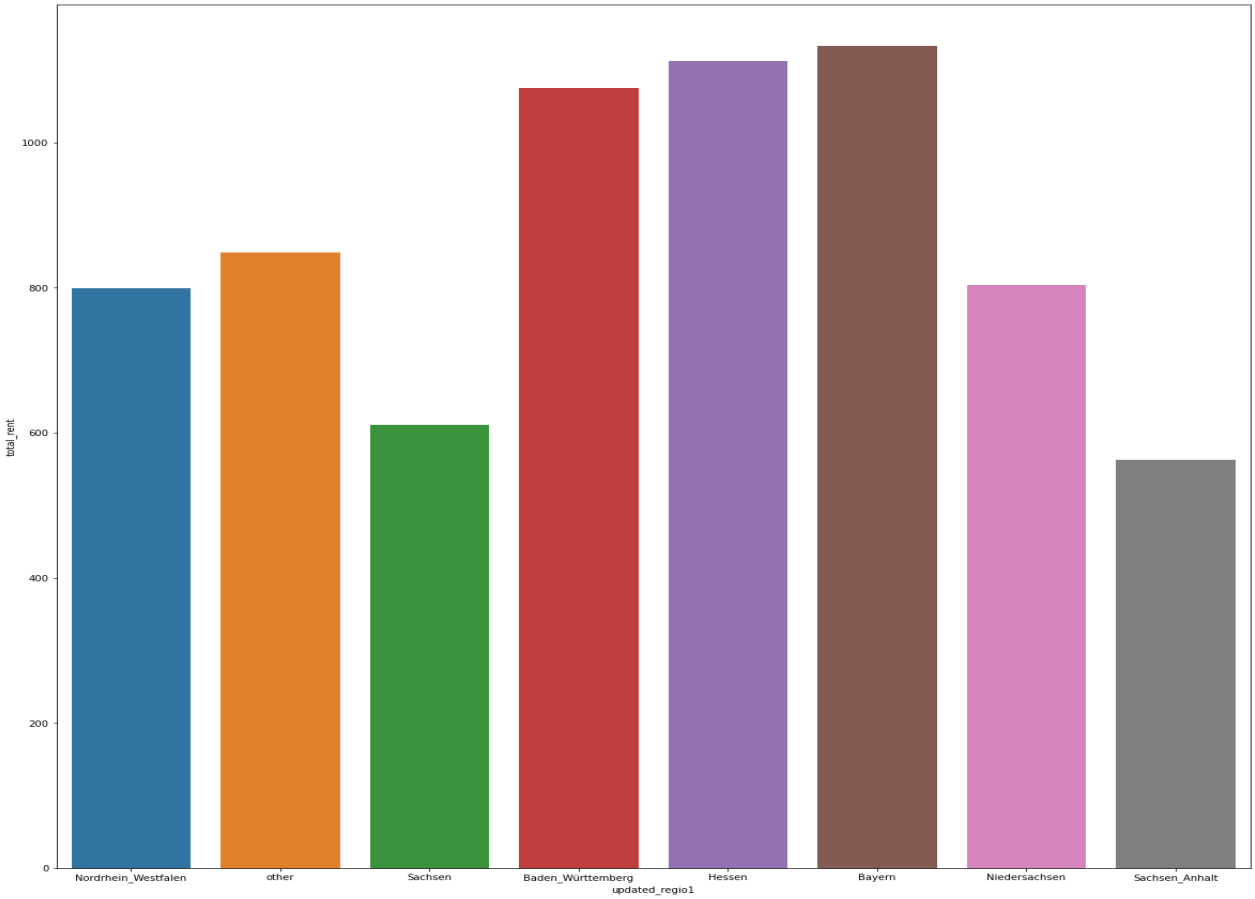
حال تعداد آگهی ها در مناطق مختلف و نوع خانه ها را بررسی میکنیم.

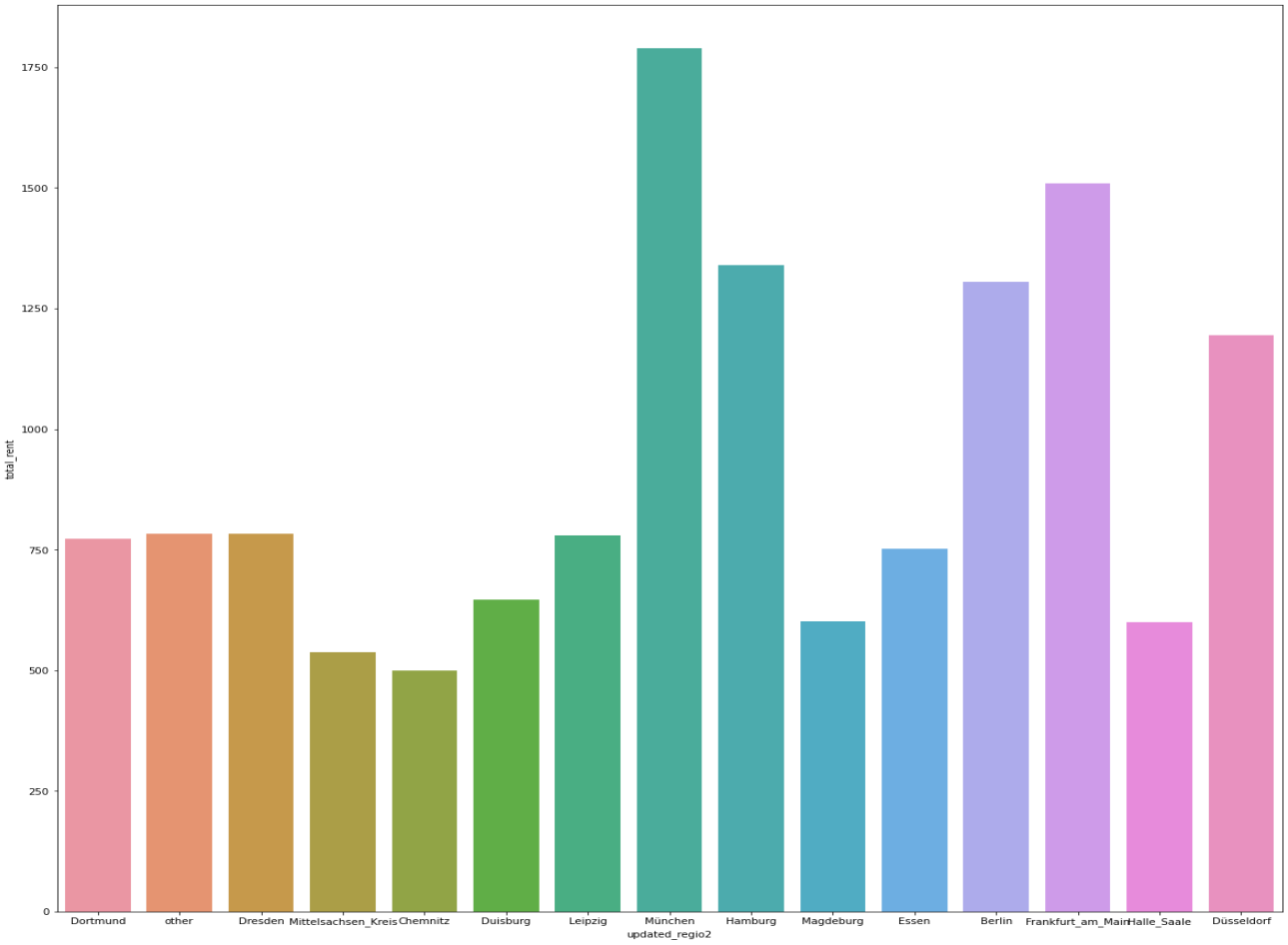






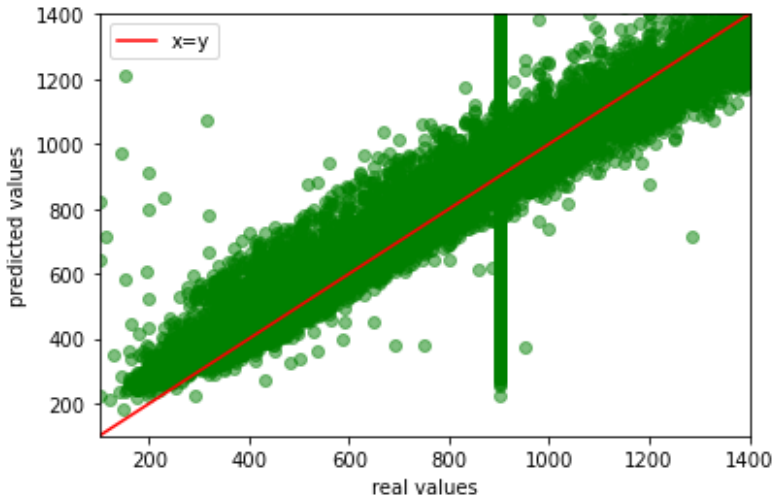
قیمت را بر اساس مناطق مختلفی که تبلیغات کردیم بررسی میکنیم.


1000
800
600
400
200
0
total_rent
Nordrhein_Westfalen
other
Sachsen
Baden_Württemberg
Hessen
Bayern
Niedersachsen
Sachsen_Anhalt
updated_regio1



Training -3

ستون های کتگوریکال را به حالت عددی در میاوریم با استفاده از get_dummies() که در پانداز موجود است. سپس داده ها را به نسبت 2 به 8 برای تست و ترین جدا میکنیم. حال از StandardScaler برای scale کردن داده ها به جز آنها که فقط 0 و 1 اند استفاده میکنیم. در آخر از رگرسیون خطی برای آموزش استفاده کرده و پس از دادن داده های تست به مدل و چک کردن mean_absoloute_error میبینیم که با توجه به مقدار واقعی و پیشبینی شده داده های تست مقدار خطا 112.79414820155971 است.



با توجه به شکل میبینیم که تقریبا پیشبینی به خوبی صورت گرفته.

Multiprocessing and dask -5-4

روی توابع هر سه حالت single process, multiprocess, dask تست میکنیم و متوجه میشویم dask سرعت را خیلی بالا میبرد و در این مثال حالت single process از multi process بهتر عمل میکند.

single processing : 0.06467008590698242

multi-processing : 0.23775863647460938

dask processing : 0.012067317962646484