Shahid Beheshti University

Faculty of Mathematical Science

Department of Computer Science

Fundamentals of Machine Learning– Spring 2022 – Assignment 1 – Part 2

Apartment Rental Offers Germany

By:

Arman Davoodi

Abstract

This report gives an overview on a dataset consisting of information about some apartments and their rental offers in Germany. It also describes the processes performed on the dataset before training a Random Forest regressor on it and the results given by this model. The goal is to predict the total rent for any given apartment.

Dataset

The dataset referred in this report is obtained from the Kaggle and it has 48 columns of different types including some data about the date of each offer, construction year of the accommodations, their facilities, locations, base rents, service charges and total rents. Although this dataset has 268850 records, there are some missing information for different features. More details about each feature and the dataset can be found at the link where the dataset is obtained from.

Preprocesses

Since this dataset has missing data and too many features, especially categorical features with too many values, some preprocess had to be done on it to make it ready to analyze and use it for training.

Feature Selection

To make the dataset easier to work with columns with more than 50% missing data were removed. In addition some other columns that were not useful in our analysis or were repeated such as regio2 were also dropped leaving only 25 features to work with.

Handling Rare Values and Categories

From the remaining features, regio1, heatingType, firing Types, geo_krs, and condition were categorical features with too many unique values. For each of these features a threshold was chosen which all categories with the number of records less than the threshold were combined into a new category named other.

Moreover, since the features floor and numberOfFloors have discrete nature in addition to the fact that some of the values in these features are repeated a lot while some others can be considered outliers, these features were binned in a non-uniform manner.

These operations were executed on the dataset three times with different methods and the runtime of each method is computed to see which method is faster. The results are demonstrated in *table 1*. Also, these operations were executed on a system with Windows 10 Pro and Intel(R) Core(TM) i7-8550U.

Table 1

Method	Time(seconds)
Pandas – Single thread	1.73
Pandas – 8 threads	1.78
Dask – 8 threads	1.65

Since the dataset was not large enough, the simple multiprocessing methods overhead was more than the speedup it gave therefor it took 50 milliseconds longer to execute compared to

the single thread method. However Dask library is implemented better so its runtime is less than the other two methods.

Missing Values and Total Rent

All missing data for categorical, Boolean and discrete numerical features were replaced by the most repeated category. The numerical featurs were yearConstructionRange, floor and numberOfFloors.

Also, in order to handle missing data for continuous numerical features, they were replaced by the columns mean.

This approach is taken because the number of missing records were too much and dropping them would mean losing lots of the information.

However, since the total rent is the target of our regression, predicting the values of missing data may cause the model's performance to drop; so, all records with missing or zero total rent were dropped from the dataset.

Removing Outliers

By looking at *figure 1*, we can see that the distribution of our data is totally skewed. Due to this fact we use IQR approach for removing the outliers from these features.

To further demonstrate the effect of IQR, boxplot of our features before and after applying the outlier removal algorithm are shown in *figure 2* and *figure 3* respectively.

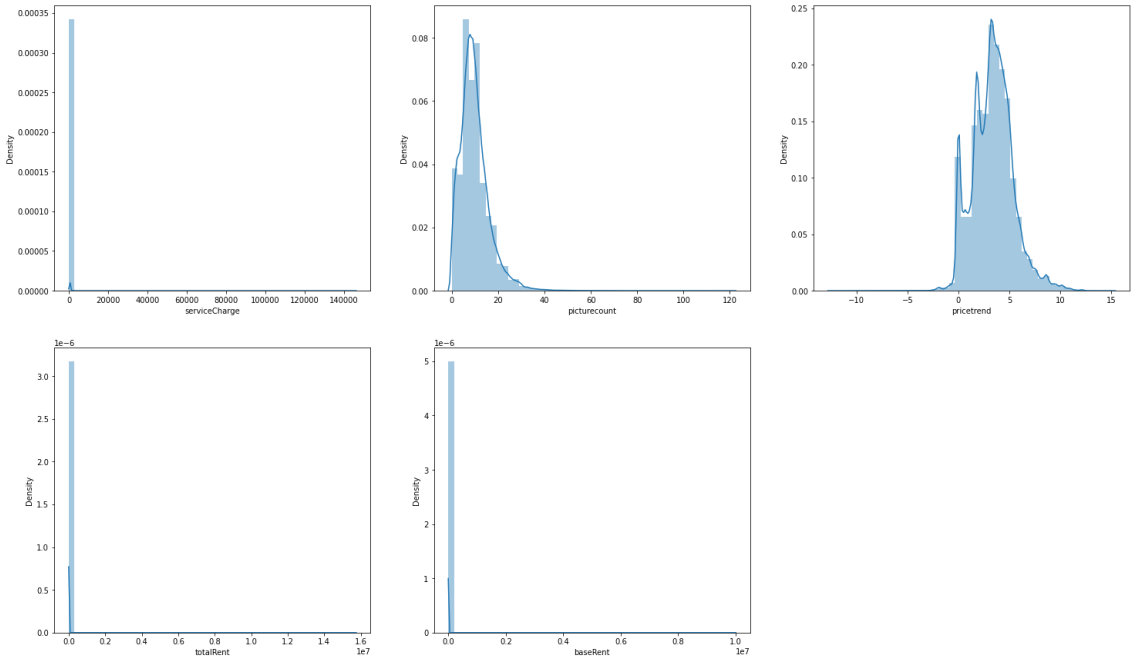

Figure 1

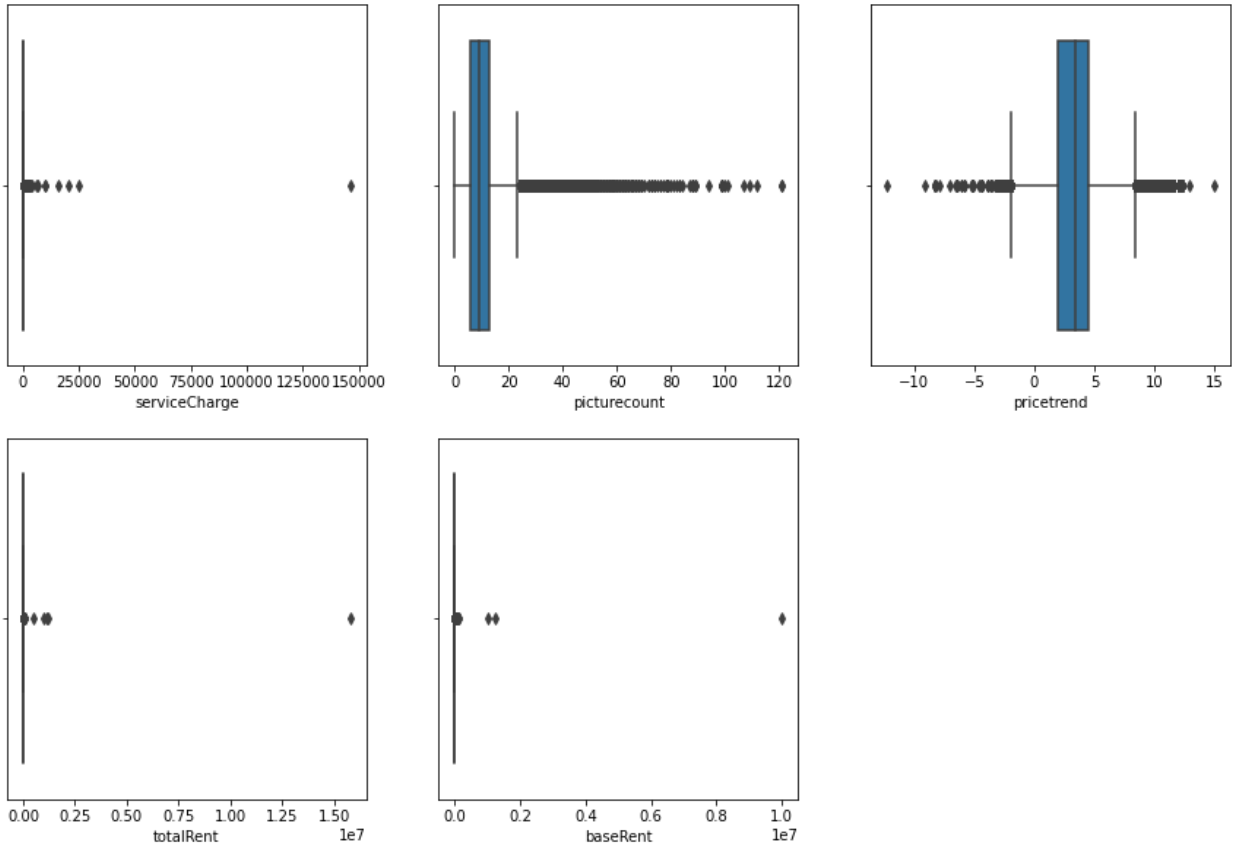

Figure 2

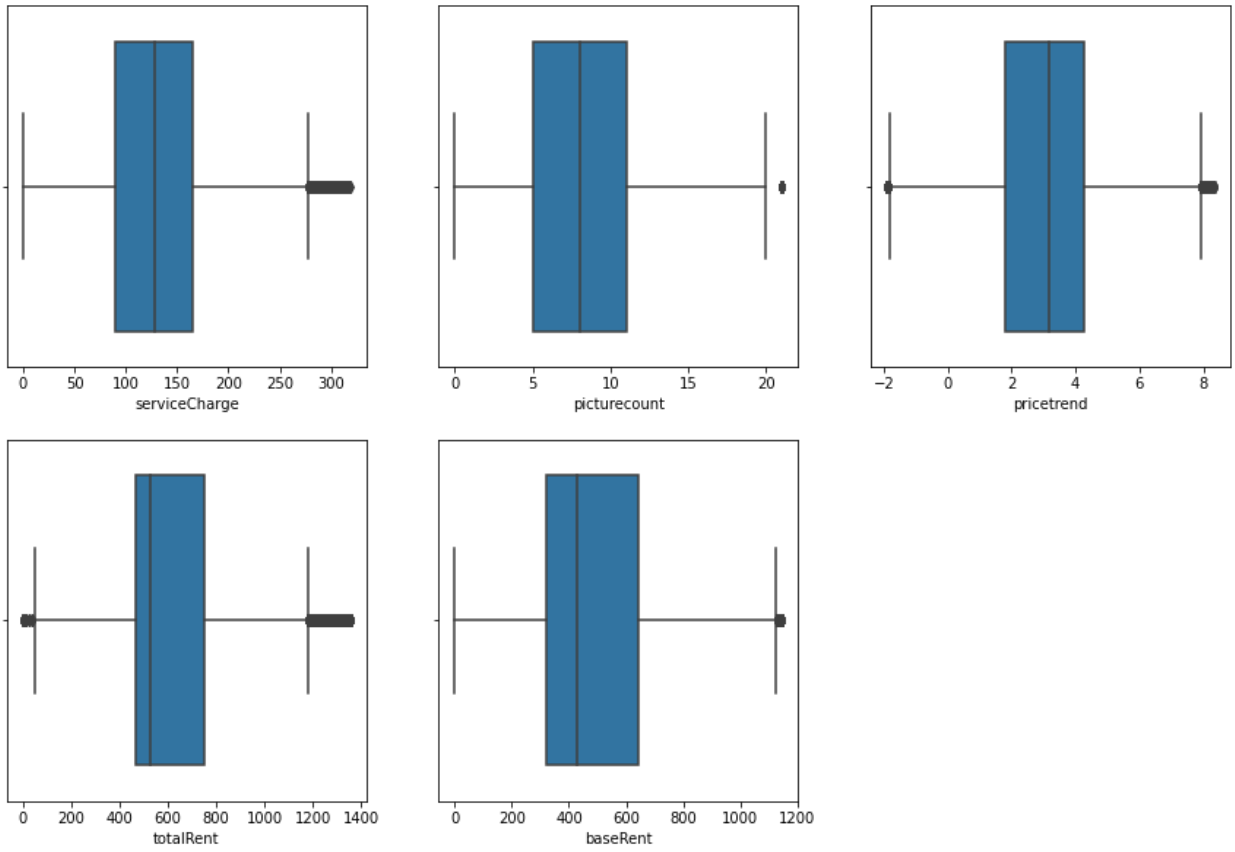

Figure 3

Data Analysis

Having done preprocessing steps, the dataset now has 222497 records and 25 parameters. 6 of these parameters are of type Boolean, 9 are of type category, 4 are of type floating-point, 1 is of type integer and 6 are of type ordinal categorical. The mean, standard deviation, min, max, and quartiles of numerical parameters are illustrated in *table 2*.

By looking at *figure 4*, we can see that after removing the outliers, the distribution of numerical features have become somewhat near normal.

In addition, histogram of all non-numeric parameters are shown in *figure 5* (this figure is also saved as ARO_hist in the dedicated git repository).

Table 2

	serviceCharge	picturecount	pricetrend	baseRent	totalRent
Mean	131.78	8.54	3.06	495.31	621.92
std	55.90	4.65	1.74	227.43	243.11
Min	0	0	-1.92	0	1
25%	90	5	1.82	320	469
50%	128	8	3.17	430	527
75%	165	11	4.26	642.5	753.98
Max	318.5	21	8.32	1145	1355

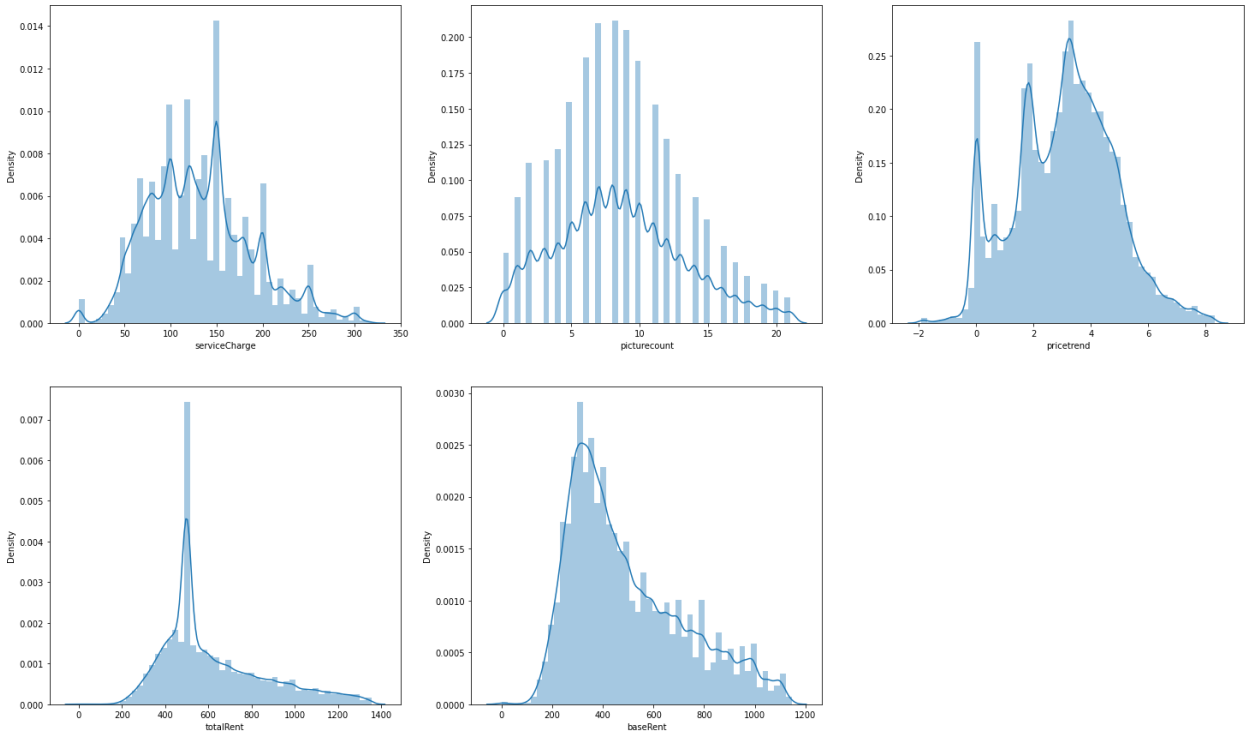

Figure 4

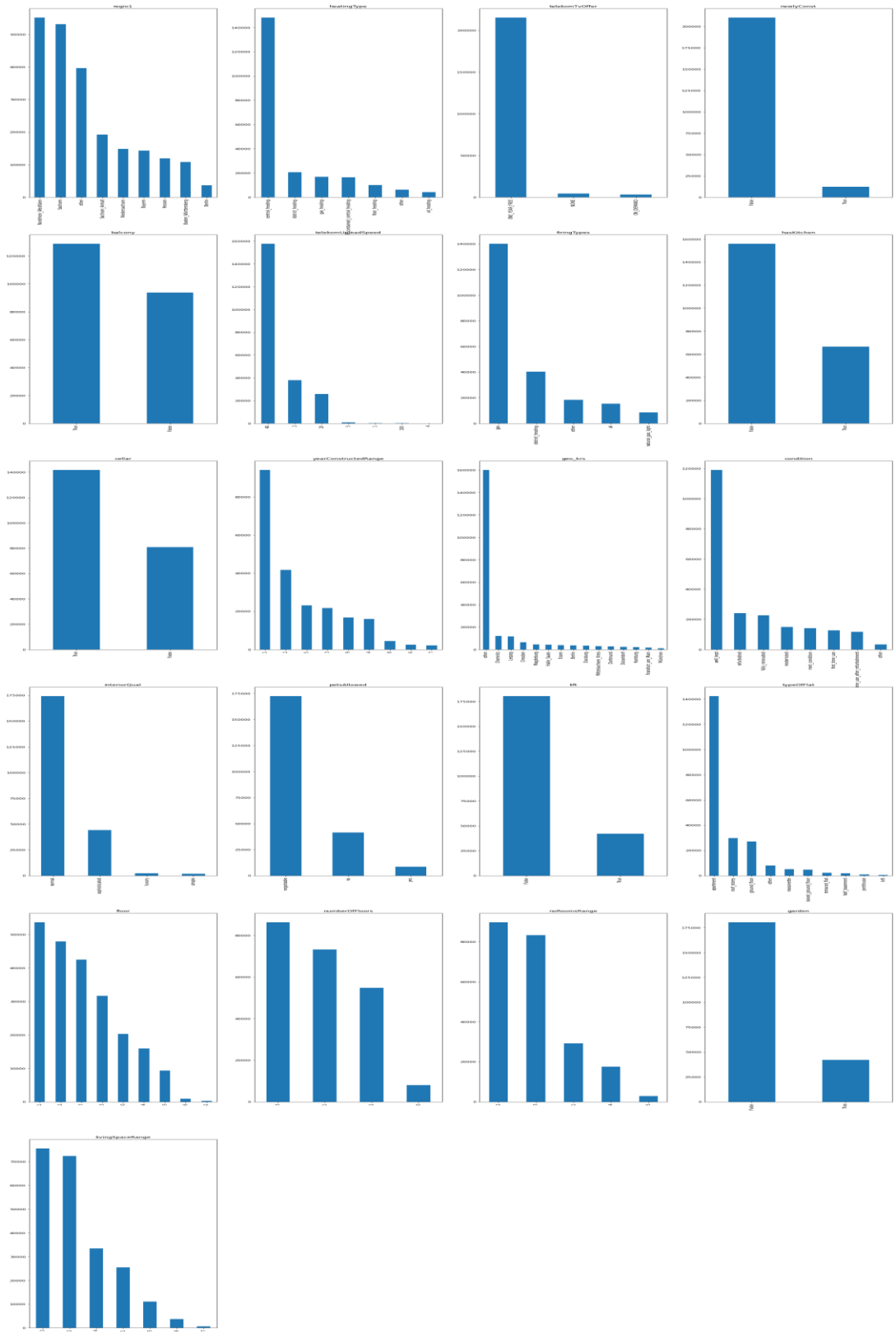
regio1
heatingType
telekomTvOffer
newlyConst
balcony
telekomUploadSpeed
firingTypes
hasKitchen
cellar
yearConstructedRange
geo_krs
condition
interiorQual
petsAllowed
lift
typeOfFlat
floor
numberOfFloors
noRoomsRange
garden
livingSpaceRange

Figure 5

To have an overview of the relationships between each pair of non-categorical parameters, the correlation matrix of this dataset is illustrated in *figure 6*.

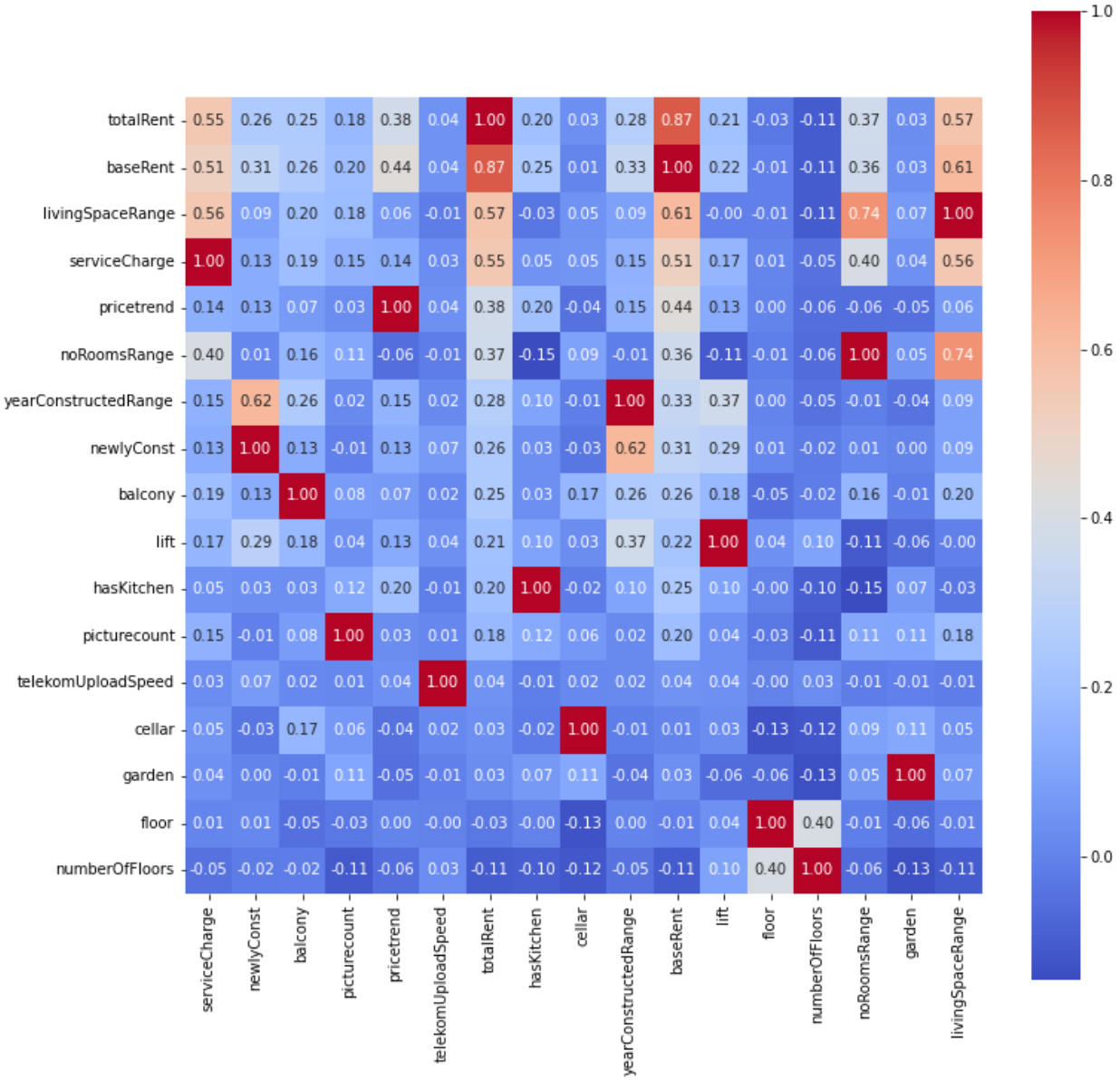

Figure 6

By looking at the correlation matrix, we can see that baseRent has a correlation score of 0.81 with totalRent. This can also be seen by looking at the *figure 7* which is the scatter plot of these to parameters.

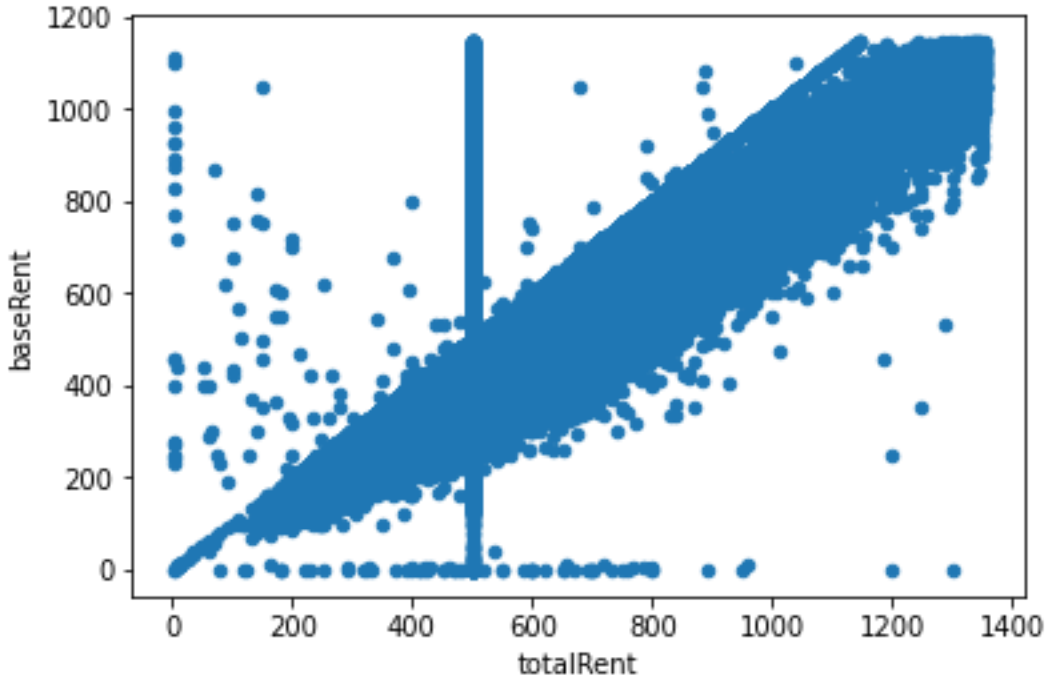

Figure 7

After that we can see that livingSpaceRange and serviceCharge also have a high correlation with totalRent. And after that we can see that there are also two weaker relationships between pricetrend and totalRent, and noRoomsRange and totalRent.

Figure 8 also demonstrates that the average totalRent in Munchen, Frankfurt_am_Main andHamburg is higher than other regions.

We can also conclude from *figure 9* that the average totalRent of accommodations with conditions first_time_use, mint_condition and first_time_use_after_refurbishment is higher than other accommodations.

Figure 10 also demonstrates that apartments which have a lift are typically more expensive.

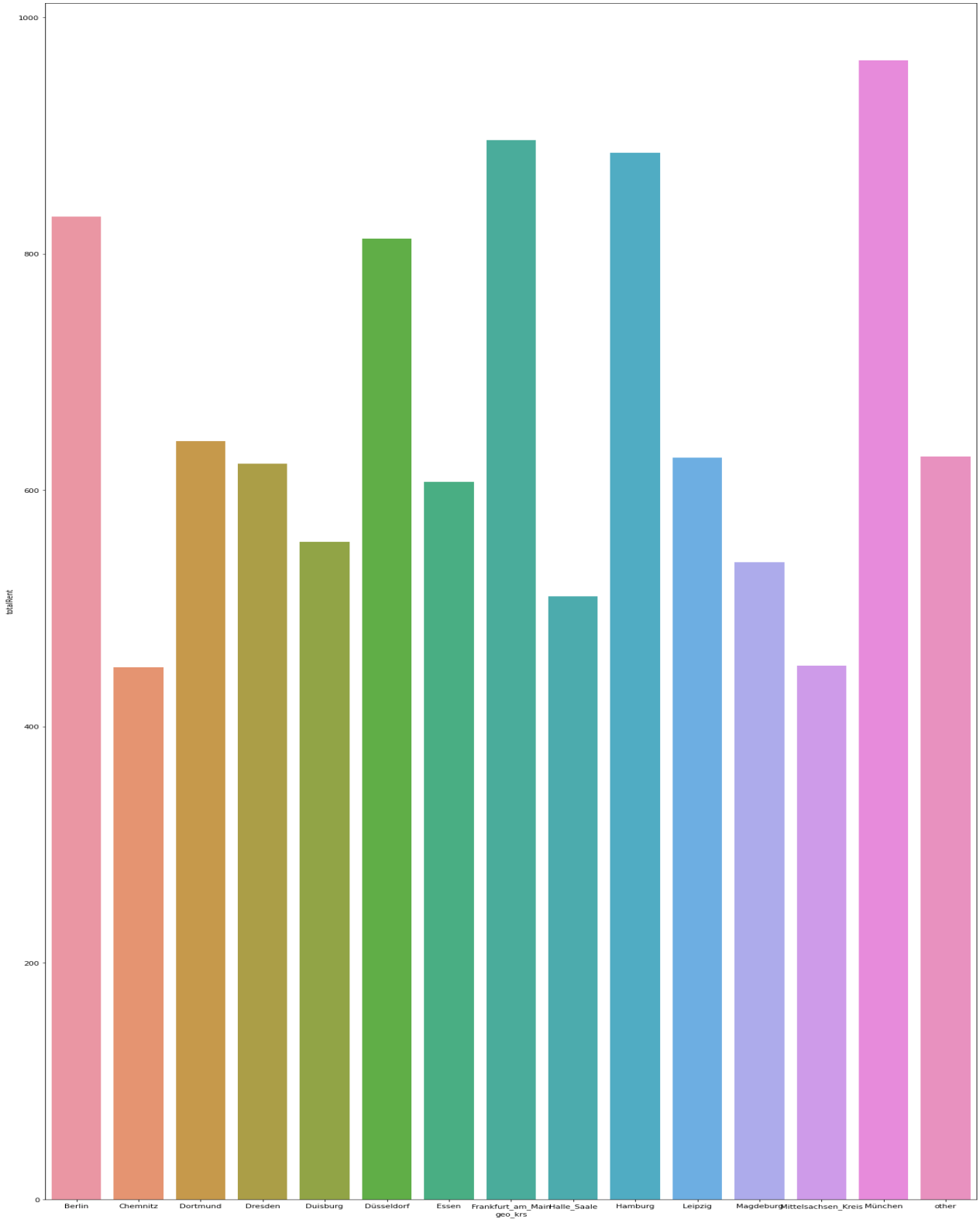
1000
800
600
400
200
0
totalRent
Berlin
Chemnitz
Dortmund
Dresden
Duisburg
Düsseldorf
Essen
Frankfurt_am_Main
geo_krs
Halle_Saale
Hamburg
Leipzig
Magdeburg
Mittelsachsen_Kreis
München
other

Figure 8

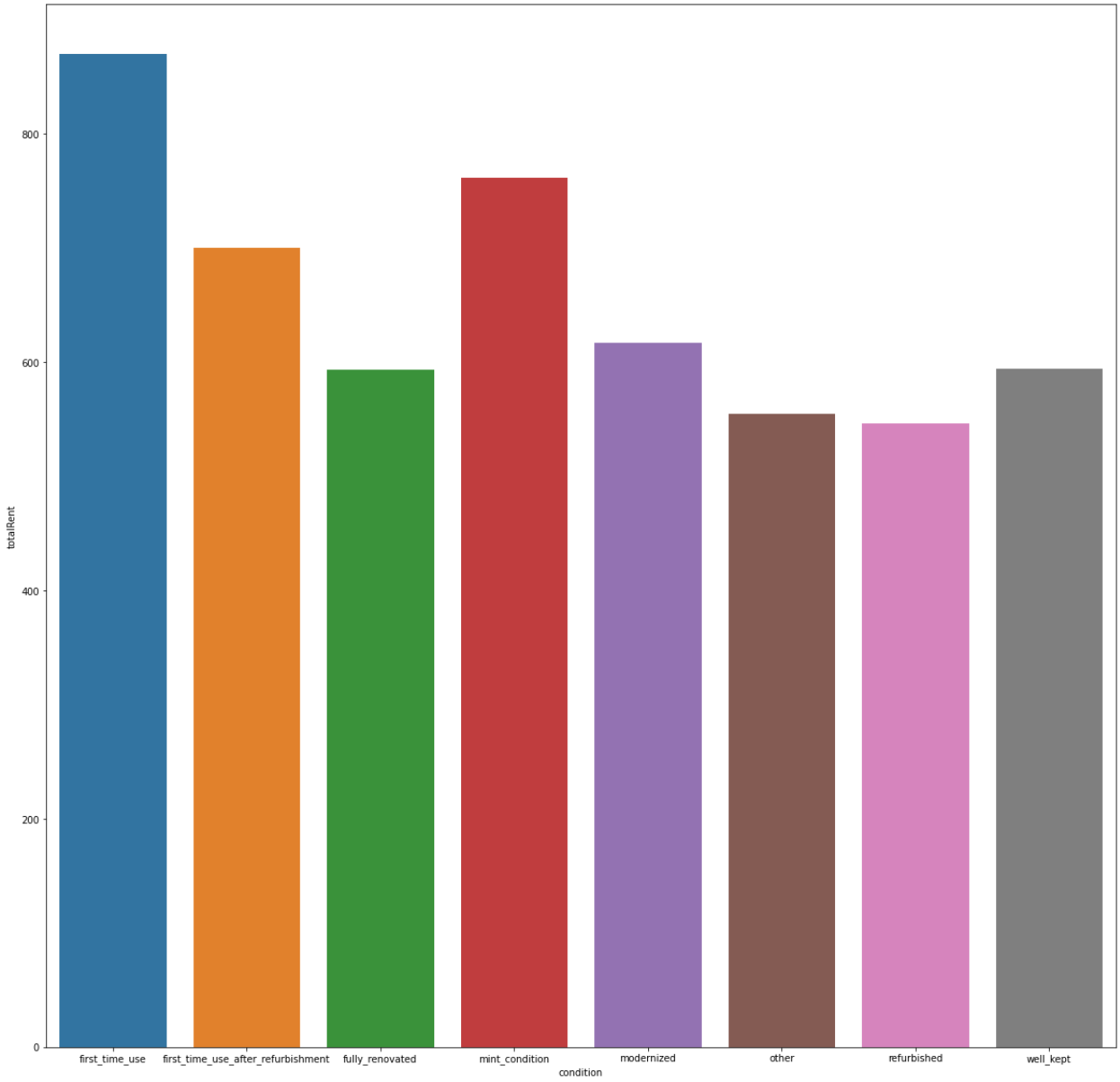
800
600
400
200
0
totalRent
first_time_use
first_time_use_after_refurbishment
fully_renovated
mint_condition
modernized
other
refurbished
well_kept
condition

Figure 9

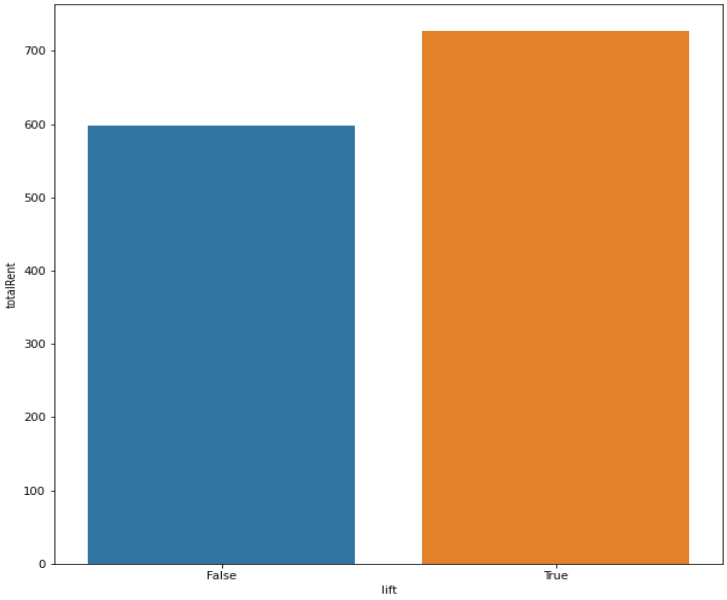

Figure 10

We can also understand from *figure 11* that 53.49% of the apartments that were in the dataset were in well-kept condition and near 20% of them were either refurbished or it was the first time that they were being used.

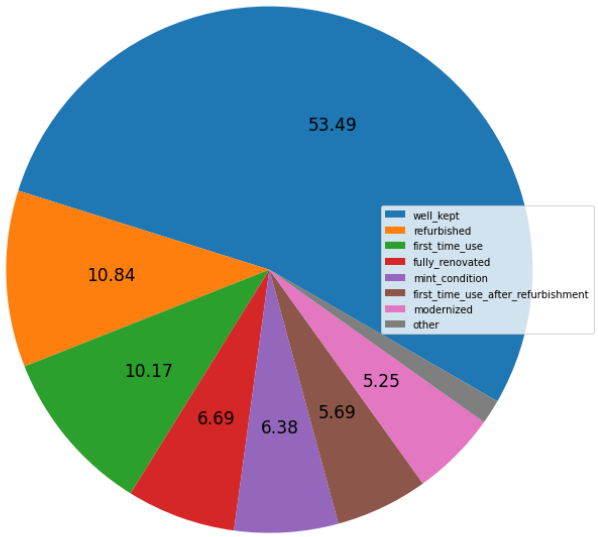

Figure 11

Model

Due to the fact that most of the datasets parameters are of categorical type, it was decided to use a Random Forest Regressor to predict the totalRent of apartments. In addition, since baseRent and serviceCharges are expenses and can make the model dependent, even though we are not always guaranteed to have them, these parameters were dropped in this step leaving only 21 predictors and 1 target. Also to demonstrate our models performance 20% of our data was used as test and only the other 80% were used to train the model.

To handle categorical data three methods were used. The first method was to use one-hot encoding for all non-numeric features. The second method was to only use one-hot encoding on non-binary cardinal categorical features and the third method was to use label encoding for all categorical features.

Also, some of the model's hyper parameters were tuned.

In addition, to speed up these tests, all of the CPU cores were used when training the model.

In the end, label encoding combined with maximum tree depth of 7, min_sample_split of 2, max_featurs equal to all features and ccp_alph of 0 gave the best result.

The rmse of our model was approximately 159.35 which is good considering the range of total rents given in *table 2*.

To further illustrate the performance of the model scatter and hexagonal plot of predicted values and true values are shown in *figure 12* and *figure 13* respectively and as it can be seen the data points are more aggregated near the line x = y. if a data point is on this line it means that the predicted value was the same as the true value.

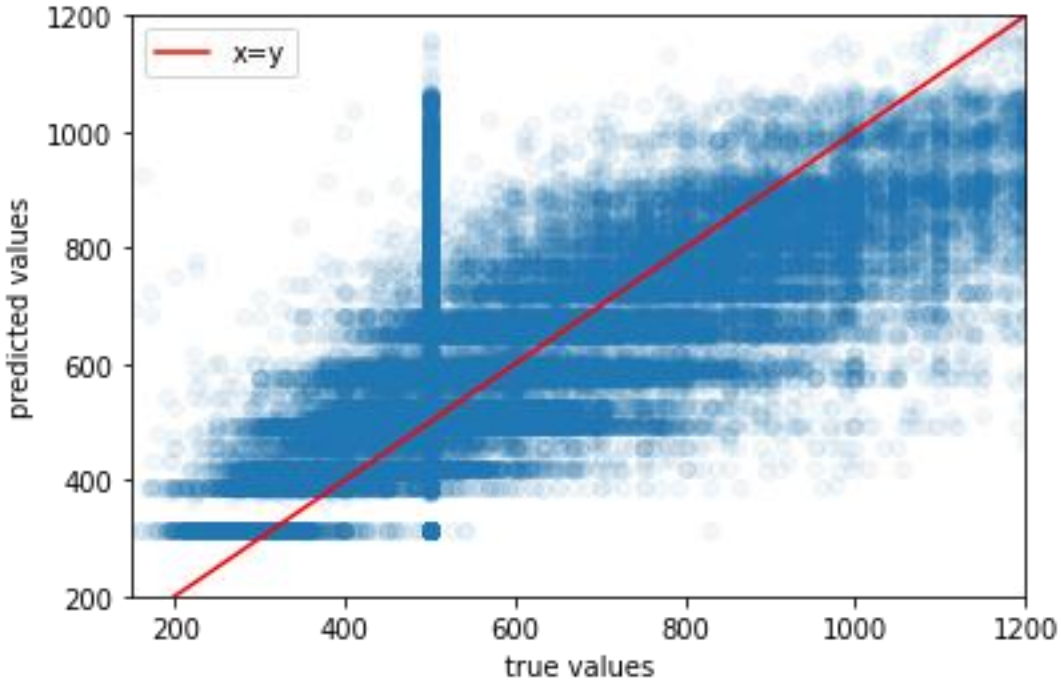

Figure 12

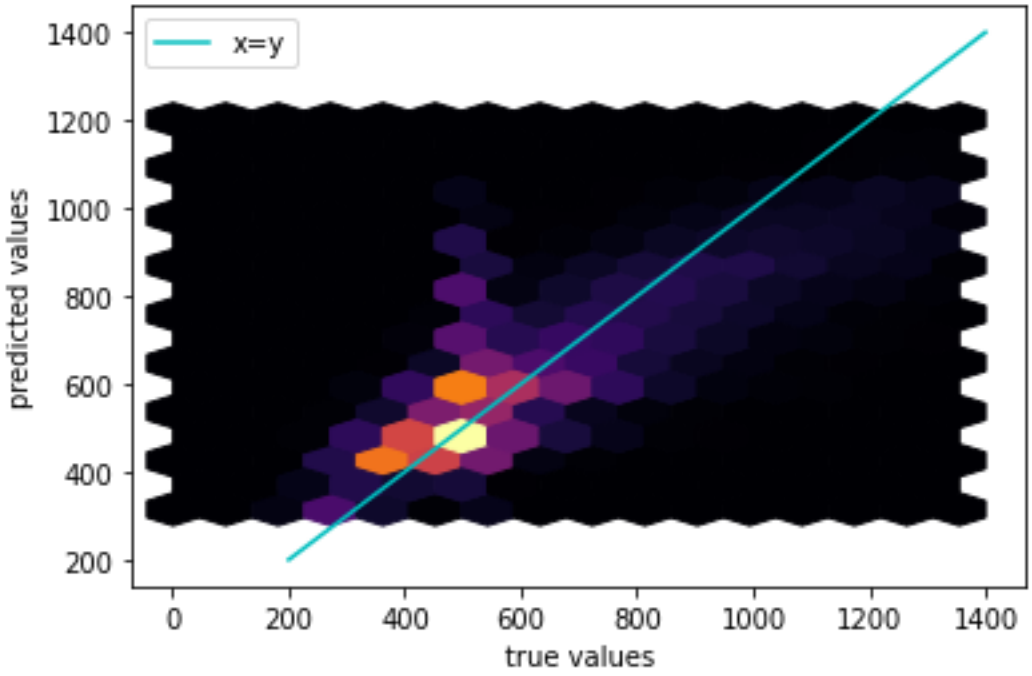

Figure 13