Apartment Rental Prediction System

Dr. Ivan S. Zapreev

2020-01-06

Introduction

Dataset overview

As stated on the webpage of the 'Apartment rental offers in Germany' dataset, it contains `198,379` rental offers scraped from the Germany's biggest real estate online platform ß ImmobilienScout24.

The data set consists of a single CSV file: *immo_data.csv* which only contains offers for rental properties. The data features important rental property attributes, such as the living area size, the rent (both base rent as well as total rent), the location, type of energy, and etc. The `date` column present in the data set defines the time of scraping, which was done on three distinct dates: *2018-09-22*, *2019-05-10* and *2019-10-08*.

The complete list of data set columns is extensive[1] and thus in this study we will use the following subset:

```
##  [1] "hasKitchen"           "heatingType"           "balcony"
##  [4] "lift"                 "garden"                "cellar"
##  [7] "noParkSpaces"         "livingSpace"           "typeOfFlat"
## [10] "noRooms"              "floor"                 "numberOfFloors"
## [13] "condition"            "newlyConst"            "interiorQual"
## [16] "yearConstructed"      "energyEfficiencyClass" "regio1"
## [19] "regio2"               "regio3"                "baseRent"
## [22] "electricityBasePrice" "heatingCosts"          "serviceCharge"
## [25] "totalRent"            "date"
```

This sub-selection reduces the number of considered data set columns[2] from 48 to 26 and is motivated by the personal preferences of the report's author and has no scientifically proven motivation. On the contrary, this column selection shall be seen as a part of problem statement. In other words, the task is to build an accurate[3] rental price prediction model based on the predictors from this set of columns.

The additional data preparation steps will be described in the *"Data wrangling"* section of this document.

Project goal

Execution plan

Data wrangling

In this section we present cleaning, restructuring and enriching the raw data taken from the 'Apartment rental offers in Germany' dataset.

[1] Please consider reading *"Appendix A"* for the complete list of the data set columns.
[2] Please consider reading *"Appendix B"* for the column descriptions.
[3] Please consider reading the *"Project goal"* section for an exact goal formulation.

First, let us note that the number of data entries in the original data set is equal to 198332. This data is however not ready to be worked with as it contains multiple `N/A` values and other inconsistencies. For example, the next table summarizes the number of `N/A` values per column:

```
## # A tibble: 26 x 3
##    `Column name`          `N/A count` `N/A percent`
##    <chr>                        <int>         <dbl>
##  1 electricityBasePrice        151158         76.2
##  2 energyEfficiencyClass       143315         72.3
##  3 heatingCosts                135154         68.2
##  4 noParkSpaces                130405         65.8
##  5 interiorQual                 83001         41.8
##  6 numberOfFloors               71792         36.2
##  7 condition                    50317         25.4
##  8 yearConstructed              42293         21.3
##  9 floor                        37612         19.0
## 10 heatingType                  32605         16.4
## 11 totalRent                    29762         15.0
## 12 typeOfFlat                   27571         13.9
## 13 serviceCharge                 5110          2.58
## 14 hasKitchen                       1          0
## 15 lift                             1          0
## 16 garden                           1          0
## 17 cellar                           1          0
## 18 livingSpace                      1          0
## 19 noRooms                          1          0
## 20 baseRent                         1          0
## 21 balcony                          0          0
## 22 newlyConst                       0          0
## 23 regio1                           0          0
## 24 regio2                           0          0
## 25 regio3                           0          0
## 26 date                             0          0
```

As one can see, about $\frac{1}{2}$ of the columns has 10–80% `N/A`^{s}, whereas the other half has (almost) no `N/A`^{s}.

The remainder of the section will be organized as follows. First we explain how we cleaned the data and solved some of its inconsistencies. Then we provide a summary of the cleaned data set. In the end we explain how we split the entire data set into the `validation` and `modeling` sub-sets[4].

Data cleaning

The data cleaning will be explained in the next steps:

1. We begin with the `totalRent` column as this is the value that we want to predict;
2. We proceed with the columns with the marginal ($< 1\%$) of `N/A` values;
3. We cover the remaining columns in the descending order of the number of `N/A` values.
4. We consider and sole some other data inconsistencies.

The `totalRent` column contains data that we want to predict. Therefore, the rows with `totalRent` == `N/A` are useless to us and shall be removed. Unfortunately, this will reduce the data set by 15.01%. There are also 13 columns with a marginal (0 to 1) number of `N/A` values. The latter can be seamlessly removed as even if all of these `N/A`^{s} appear in different rows, we will remove at most 13 entries which is 0.0066% of data.

[4]The latter will also be split into the `training` and `testing` set for the sake of model cross-validation.

The remaining columns

Additional issues

Clean data summary

Let us now summarize the resulting clean data:

```
## # A tibble: 24 x 3
##    `Column name`          `N/A count` `N/A percent`
##    <chr>                        <int>         <dbl>
##  1 hasKitchen                       0             0
##  2 heatingType                      0             0
##  3 balcony                          0             0
##  4 lift                             0             0
##  5 garden                           0             0
##  6 cellar                           0             0
##  7 noParkSpaces                     0             0
##  8 livingSpace                      0             0
##  9 typeOfFlat                       0             0
## 10 noRooms                          0             0
## 11 floor                            0             0
## 12 numberOfFloors                   0             0
## 13 condition                        0             0
## 14 newlyConst                       0             0
## 15 interiorQual                     0             0
## 16 energyEfficiencyClass            0             0
## 17 regio1                           0             0
## 18 regio2                           0             0
## 19 regio3                           0             0
## 20 baseRent                         0             0
## 21 heatingCosts                     0             0
## 22 serviceCharge                    0             0
## 23 totalRent                        0             0
## 24 date                             0             0
```

As one can notice, the dat set size has been reduced from 168543 to 198332. The major reason for that is excluding the rows with the `N/A` values of the `totalRent` column. Let us recall that the number of such raws was 15.01% of the data set, e.g. 29770 rows. It now remains to notice that $198332 - 29770 = 168562 \approx 168543$. The remaining delta is explained by cleaning the `floor`/`typeOfFlat` column inconsistencies.

Splitting data

Data analysis

Modeling approach

Results

Conclusions

Appendix A: The complete list of data set columns

Hereby we present the list of columns from the original data set:

```
##  [1] "regio1"                    "serviceCharge"
##  [3] "heatingType"               "telekomTvOffer"
##  [5] "telekomHybridUploadSpeed" "newlyConst"
##  [7] "balcony"                   "electricityBasePrice"
##  [9] "picturecount"              "pricetrend"
## [11] "telekomUploadSpeed"        "totalRent"
## [13] "yearConstructed"           "electricityKwhPrice"
## [15] "scoutId"                   "noParkSpaces"
## [17] "firingTypes"               "hasKitchen"
## [19] "geo_bln"                   "cellar"
## [21] "yearConstructedRange"      "baseRent"
## [23] "houseNumber"               "livingSpace"
## [25] "geo_krs"                   "condition"
## [27] "interiorQual"              "petsAllowed"
## [29] "streetPlain"               "lift"
## [31] "baseRentRange"             "typeOfFlat"
## [33] "geo_plz"                   "noRooms"
## [35] "thermalChar"               "floor"
## [37] "numberOfFloors"            "noRoomsRange"
## [39] "garden"                    "livingSpaceRange"
## [41] "regio2"                    "regio3"
## [43] "description"               "facilities"
## [45] "heatingCosts"              "energyEfficiencyClass"
## [47] "lastRefurbish"             "date"
```

Appendix B: Data set column descriptions

Here is the list of the initially considered data set columns with the descriptions thereof:

1. `hasKitchen` – has a kitchen
2. `balcony` – does the object have a balcony
3. `cellar` – has a cellar
4. `lift` – is elevator available

5. `floor` – which floor is the flat on
6. `garden` – has a garden
7. `noParkSpaces` – number of parking spaces
8. `livingSpace` – living space in sqm
9. `condition` – condition of the flat
10. `interiorQual` – interior quality
11. `regio1` – Bundesland
12. `regio2` - District or Kreis, same as geo krs
13. `regio3` – City/town
14. `noRooms` – number of rooms
15. `numberOfFloors` – number of floors in the building
16. `typeOfFlat` – type of flat
17. `yearConstructed` – construction year
18. `newlyConst` – is the building newly constructed
19. `heatingType` – Type of heating
20. `energyEfficiencyClass` – energy efficiency class
21. `heatingCosts` – monthly heating costs in €
22. `serviceCharge` – auxiliary costs such as electricity or Internet in €
23. `electricityBasePrice` – monthly base price for electricity in €
24. `baseRent` – base rent without electricity and heating
25. `totalRent` – total rent (usually a sum of base rent, service charge and heating cost)
26. `date` – time of scraping