Nikita Traynin, nxt170002, 2021413175

9/19/2021

Linear Regression on the Russian Housing Market

**Introduction**

Our goal is to predict a property's sale price based on a collection of other known variables related to that property. We take a dataset from Kaggle and perform exploratory data analysis, followed by model application (linear regression in this case), and finally evaluating the model and re-tuning it.

We are using the Sberbank Russian Housing Market dataset. It is pre-split into two sets for us - train and test. These use an approximately 80/20 train/test split pattern. In total, there are 38,133 observations. The train observations are before 7/1/2015, and the test observations are after. The train/test split is not random, since it is based on timestamp, so I will avoid re-mixing the data and performing cross-validation.

There are 292 columns in this data. One is a superfluous id variable, and another is our target variable "price\_doc". So, we have a large 290 predictors. For this reason, we will implement a gradient descent, iterative algorithm for our linear regression instead of the closed-form solution.

**EDA**

There are 157 integer, 119 float, and 16 categorical variables. There are too many to fully describe here, though some examples are: "full\_sq" (integer – the total square footage),"green\_zone\_part" (float – the proportion of greenery in the nearby area), and "hospital\_beds\_raion" (float – the number of hospital beds in the district). We now analyze the normality of the numeric variables. Since there are so many, I will only show the first 10.

Chart, bar chart

Description automatically generated

Graphical user interface, chart

Description automatically generated with medium confidence

As we can see, the histograms are very skewed. With these variables, they are all skewed right. Furthermore, extreme outliers are very common, as most of the bins are not big enough to be visible in the plots. As such, we must standardize and normalize all the variables. Although this won't remove the asymmetry, this is as far as we should go as empirically it is good enough.

We have to deal with the categorical and Boolean variables before applying linear regression. In this dataset, we will convert booleans into 0s and 1s and leave them untouched. The categoricals will be one-hot-encoded, with one column dropped afterwards to prevent multicollinearity. However, there is one categorical that has 146 categories. This is such a huge amount that we will drop this variable for performance reasons, but try to put it back later.