Data Wrangling

Overview

This section describes the various data cleaning and data wrangling methods applied to the “house-prices-advanced-regression-techniques” data.

Summary Files

This dataset is split into two separate files:train and test sets.
I took a train.csv file since I have a SalePrice target variable in there to investigate dependencies and this file is basically used to train a ML model.

Original version of the dataset
https://www.kaggle.com/c/house-prices-advanced-regression-techniques.

Duplicates and Missing Values

For the analysis it was necessary to check for duplicates.
The dataset also had a lot of missing values due to the not being available in the Database, excluding categorical values.The loss is about 27%, this is a significant loss of data. The difficulty of handling missing data in my dataset is that I do not have the same datatype for all the data.
There are 6 ways to handle missing values in a dataset:

- 1- Do Nothing:However, other algorithms will panic and throw an error complaining about the missing values (ie. Scikit learn — LinearRegression). In that case, you will need to handle the missing data and clean it before feeding it to the algorithm.
- 2- Imputation Using (Mean/Median) Values:

Pros:

- Easy and fast.
- Works well with small numerical datasets.

Cons:

- Doesn’t factor the correlations between features. It only works on the column level.
- Will give poor results on encoded categorical features (do NOT use it on categorical features).
- Not very accurate.
- Doesn’t account for the uncertainty in the imputations.

- ## 3- Imputation Using (Most Frequent) or (Zero/Constant) Values:

Pros:

- Works well with categorical features.

Cons:

- It also doesn’t factor the correlations between features.
- It can introduce bias in the data.

- ## 4- Imputation Using k-NN:

Pros:

- Can be much more accurate than the mean, median or most frequent imputation methods (It depends on the dataset).

Cons:

- Computationally expensive. KNN works by storing the whole training dataset in memory.
- K-NN is quite sensitive to outliers in the data (**unlike SVM**)

- ## 5- Imputation Using Multivariate Imputation by Chained Equation (MICE)
- ## 6- Imputation Using Deep Learning (Datawig):

I applied only pandas methods in python . I do not touch scikit learn yet, this more advanced, but probably can do it.
I replaced missing data with the most common values for object data type values, but columns that have 90% missing I dropped since It can introduce bias in the data. For int or float data type values I replaced missing values with the median since the median will be the same for any distribution, still not very accurate for training a prediction model.
Also, tested melting, pivoting, unstacking, and more techniques to sharpen my skills and keeping the original.
For easy navigation through my dataset I performed sorting of column names in an alphabetical order.

Outliers and handling them

After making a plot of SalePrices it is easy to see outliers.(unusual pattern of data that is really different or far from normal) But are those really outliers?
To decide outliers or not, I performed an IQR test for Sale Prices. So this test gives me an outcome of where my outliers are and knowing that I can remove this data so that would not create a problem for training a prediction model. This is my approach at this point.