Home Prices Prediction

Introduction to Machine Learning

Dr. Michel Riveill & Dr. Diane LINGRAND

Hyelim LEE

January 5, 2023

Table of Content

1 Project Description

Ask a buyer to describe their dream home and they probably won't start with basement ceiling height or proximity to an east-west rail line. But the data set proves that price negotiations are influenced by much more than the number of bedrooms or a white picket fence. The goal of this project is to predict the sales price for each house using everything we have learned so far. And Our models are evaluated on the Root-Mean-Squared-Error (RMSE).

2 Overview

2.1 Overview about the data set

The dataset used for this project consists of two types: train.csv and test.csv. The training set has shape (1000, 80) and the test set has shape (460, 79). The column-specific data types for that dataset can be summarized as follows:

Numerical (37)	Categorical (43)
'MSSubClass', 'LotFrontage', 'LotArea', 'OverallQual', 'OverallCond', 'YearBuilt', 'YearRemodAdd', 'MasVnrArea', 'BsmtFinSF1', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF', '1stFlrSF', '2ndFlrSF', 'LowQualFinSF', 'GrLivArea', 'BsmtFullBath', 'BsmtHalfBath', 'FullBath', 'HalfBath', 'BedroomAbvGr', 'KitchenAbvGr', 'TotRmsAbvGrd', 'Fireplaces', 'GarageYrBlt', 'GarageCars', 'GarageArea', 'WoodDeckSF', 'OpenPorchSF', 'EnclosedPorch', '3SsnPorch', 'ScreenPorch', 'PoolArea', 'MiscVal', 'MoSold', 'YrSold', 'SalePrice'	'MSZoning', 'Street', 'Alley', 'LotShape', 'LandContour', 'Utilities', 'LotConfig', 'LandSlope', 'Neighborhood', 'Condition1', 'Condition2', 'BldgType', 'HouseStyle', 'RoofStyle', 'RoofMatl', 'Exterior1st', 'Exterior2nd', 'MasVnrType', 'ExterQual', 'ExterCond', 'Foundation', 'BsmtQual', 'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinType2', 'Heating', 'HeatingQC', 'CentralAir', 'Electrical', 'KitchenQual', 'Functional', 'FireplaceQu', 'GarageType', 'GarageFinish', 'GarageQual', 'GarageCond', 'PavedDrive', 'PoolQC', 'Fence', 'MiscFeature', 'SaleType', 'SaleCondition'

Table 1: Data types per column

Our target value, SalePrice is not normally distributed, it is positively or right skewed.

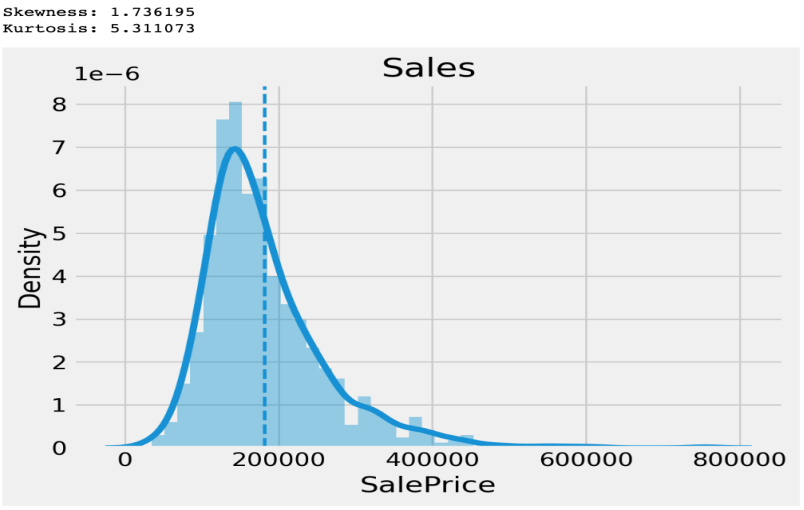

Figure 1: Distribution of Target Variable

So, Predictions are evaluated on Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price.

2.2 Work Flow

The problem can be solved by using a pipeline, which is represented in Figure 2. The pipeline consists of five steps:

- EDA
- Data Cleaning
- Features Selection
- Feature Engineering
- Modeling

We will detail these five steps in the following subsections.

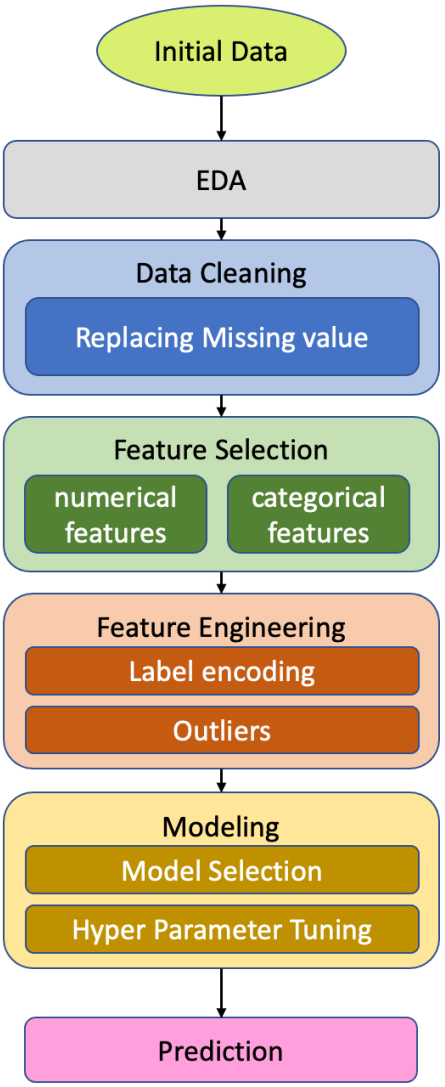

Figure 2: Pipeline of the project

3 EDA

Through Exploratory Data Analysis, first, I figured out the distribution of the target value, types by variable (chapter2 Overview). After that, the distribution of each variable and the correlation between the target value were identified by visualizing the distribution with the target value by dividing the numerical features and categorical features.

As a result, the features that have a strong correlation with the target value(SalePrice) were identified as follows.

	SalePrice
SalePrice	1.000000
OverallQual	0.797666
GrLivArea	0.734997
GarageCars	0.658204
GarageArea	0.647953
TotalBsmtSF	0.642127
1stFlrSF	0.627749
TotRmsAbvGrd	0.570375
FullBath	0.559980
YearBuilt	0.527193
YearRemodAdd	0.526195

Figure 3: Correlations with Target value

4 Data Cleaning

First of all, since original dataset were generally clean, only the Missing Value was processed at this stage. In order to preprocess the same to the train data and the test data, pre-processing was performed by concatting the two datasets at this stage.

4.1 Missing Value

In the initial dataset, a total of 19 columns contained missing values as follows.(Figure 4)
Among them, the PoolQC, MiscFeature, Alley, Fence, and FireplaceQu columns with a missing value ratio of 40% or more are excluded from the analysis. In addition, missing values were replaced with a median value for numerical features and 'None' for categorical features.

```
Selected dataframe has 79 columns and 1460 Rows.
There are 19 columns that have missing values.
```

	Zero Values	Missing Values	% of Missing Values	Data Type
PoolQC	0	1453	99.5	object
MiscFeature	0	1406	96.3	object
Alley	0	1369	93.8	object
Fence	0	1179	80.8	object
FireplaceQu	0	690	47.3	object
LotFrontage	0	259	17.7	float64
GarageType	0	81	5.5	object
GarageYrBlt	0	81	5.5	float64
GarageFinish	0	81	5.5	object
GarageQual	0	81	5.5	object
GarageCond	0	81	5.5	object
BsmtExposure	0	38	2.6	object
BsmtFinType2	0	38	2.6	object
BsmtFinType1	0	37	2.5	object
BsmtCond	0	37	2.5	object
BsmtQual	0	37	2.5	object
MasVnrArea	861	8	0.5	float64
MasVnrType	0	8	0.5	object
Electrical	0	1	0.1	object

Figure 4: Missing values of initial dataset

5 Feature Selection

Since the initial dataset has 74 columns since Data Cleaning, it is necessary to select appropriate Features to be used for modeling. We can reduce the training time and eliminate variable redundancy through Feature selection.

In a typical ML pipeline, we perform feature selection after completing feature engineering. However, at this stage, feature selection will be conducted separately between numerical features and categorical features, and since random forest learning is conducted in the process, to facilitate data processing, features for modeling are extracted using only from the train dataset before the feature engineering step, which includes processes such as one-hot encoding to expand the number of columns.

5.1 Removing correlated features

First of all, columns with a correlation of 0.8 or more between variables were removed due to the risk of overfitting. In this process, three columns were excluded: '1stFlrSF', 'GarageArea', and 'TotRmsAbvGrd'.

5.2 Features Selection of numerical data

In order to select features of the numerical data type, 10 features were extracted using the Sequential Forward Selection method. Sequential Forward Selection basically starts with a null set of features and then looks for a feature that minimizes the cost function. The following variables were extracted through this method.

- MSSubClass
- 'OverallQual
- YearBuilt
- YearRemodAdd
- BsmtFinSF1
- BsmtFinSF2
- BsmtUnfSF
- GrLivArea
- BedroomAbvGr
- GarageCars

5.3 Features Selection of categorical data

In order to select features of the categorical data type, 10 features were extracted using the Mutual information method. Mutual information (MI) of two random variables is a measure of the mutual dependence between the two variables. And It runs only for categorical features. The following variables were extracted through this method.

- Neighborhood
- ExterQual
- KitchenQual
- BsmtQual
- GarageFinish
- GarageType
- Foundation
- BsmtFinType1
- HeatingQC
- Exterior1st

As the results, 20 features for modeling were selected through the Feature selection step.

6 Feature Engineering

6.1 label encoding and factorizing variables

For the entire dataset, the columns related to quality use the following abbreviations : Ex - Excellent, Gd - Good, TA - Average/Typical, Fa - Fair, Po - Poor.
Since this can be replaced by a linear numerical variable, it was encoded with a numerical variable. The features replaced to numeric are 'ExterQual', 'ExterCond', 'BsmtQual', 'BsmtCond', 'HeatingQC', 'KitchenQual', 'GarageQual', 'GarageCond'

And some numeric variables was changed into factors(object). Typically, the data type of the column for month (‘MoSold’) is numeric. However, since it does not have continuous properties, it must be changed to a categorical variable.

6.2 Converting Categorical features

The categorical features were changed to dummy variables for use in regression modeling.

6.3 Outlier

As a result of searching for rows satisfying the following conditions to remove outliers, it was determined that only 2 outliers were extracted, and thus, outliers were not removed because it was not significant for modeling.

- OverallQual < 5 and SalePrice > 200000
- GrLivArea > 4000 and SalePrice > 500000

As the results, 78 features for modeling were prepared through the feature engineering step.

7 Modeling

7.1 Split and Scaling data

First of all, the train dataset was randomly split for learning. As a scaler for modeling, MinMaxScaler was used because the MinMaxScaler had the best performance after comparing the results scores of StandardScaler, MinMaxScaler, and RobustScaler.

7.2 Baseline Modeling

In order to identify the optimal algorithm, several regression models were learned without parameters and the result values were compared. The algorithm used and the RMSE values are as follows.

- Support Vector Regressor : 0.09628
- Gradient Boosting Regressor : 0.06188
- Decision Tree Regressor : 0.08098
- Random Forest Regressor : 0.062
- XGBoost Regressor : 0.06443
- LightGBM Regressor : 0.06598

7.3 Hyper Parameter tuning

Grid Search was conducted to find the optimal parameters for the Gradient Boosting Regressor algorithm with the best performance in the baseline modeling above.

As the result, this parameter was selected.

- {'n_estimators': 1000, 'max_depth': 5, 'min_samples_leaf':5, 'min_samples_split':5, 'learning_rate': 0.01, 'loss': 'huber', 'max_features':'sqrt'}
- RMSE : 0.05981

7.4 Ensemble

To improve the performance of the model, an ensemble model combining several models has been attempted. I stacked up all the models above, optimized using xgboost.

The algorithm used and the RMSE values are as follows.

- Stack up all the models : 0.06624

But, the resulting RMSE value is worse than before ensembling. Therefore, the gradient boosting regressor using the hyper parameter is maintained.

8 Conclusion

The best model derived through the above modeling is the stacking CV regressor combining the above algorithm and the result is summarized as follows.

Algorithm	Hyperparameter	RMSE
StackingCVRegressor (regressors=(XGB, LightBGM, SVR, GBR, RF, DT), meta_regressor=XGB)	{'n_estimators': 1000, 'max_depth':5, 'min_samples_leaf':5, 'min_samples_split':5, 'learning_rate': 0.01, 'loss': 'huber', 'max_features':'sqrt'}	0.05981

Table 2: Best model and hyperparameters