Architectural Decisions Document

Submission by Soham Ghosh

Data Sources and Use Cases
The Ames Housing dataset was compiled by Dean De Cock for use in data science education. This dataset proves that much more influences price negotiations than the number of bedrooms, height of the basement ceiling, the proximity to an east-west railroad or a white-picket fence. With 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, it is possible to try to predict the final price of each home using machine learning.

Extract Transform Load (ETL)
The dataset can be downloaded from Kaggle (link) in CSV format.

Data Cleaning
I checked for missing values in the data in train and test sets created. I decided to get rid of features that have more than half of missing information or do not correlate to “SalePrice”.

I treated the NA values in “MSZoning”, “MasVnrType”, “Electrical”, “KitchenQual”, “SaleType” columns by filling with most popular values.

I replaced the NA values in “Alley” column with “NOACCESS” (I assumed NA means “No Access” here).

I replaced the NA values in “FireplaceQu” column with “NoFP” (I assumed NA means “No Fireplace” here).

I replaced the NA values in “GarageType”, “GarageFinish”, “GarageQual” columns with “NoGRG” (I assumed NA means “No Garage” here). I replaced the NA values in “GarageCars” with 0 (I assumed NA means 0 cars here).

I treated the NA values in “BsmtQual”, “BsmtCond”, “BsmtExposure”, “BsmtFinType1”, “BsmtFinType2” columns with “NoBSMT” (I assumed NA means “No Basement” here).

I turned the MSSubClass, KitchenAbvGr, year sold, month sold columns to categorial types.

The “SalesPrice” data is skewed right. I log-transformed it to make it closer to the normal distribution.

Feature Creation
I added total square footage feature (“TotalSF”).

I have also standardized all numeric features and converted categorical features into dummy variables.

Model Definition
My analysis revealed that Gradient Boosting and Elastic Net (using Standardized Features) show best results.

Model Training
I had used a lot of features and have many outliers. So, I used max_features='sqrt' to reduce overfitting of my model. I also used loss='huber' because it more tolerant to outliers. All other hyper-parameters were chosen using GridSearchCV. I used ElasticNetCV estimator to choose best alpha and l1_ratio for the Elastic Net model. The final ensemble model is an average of Gradient Boosting and Elastic Net predictions. But before that I retrained the models on all train data.

Model Evaluation
I used average R2 score and standard deviation of 5-fold cross-validation to evaluate models.

Model Deployment
Model has been deployed on IBM Cloud.