Housing-Prediction

Naveen, Greg, Riza

Knime Project

01

Data Exploration

About the Data

Statistics

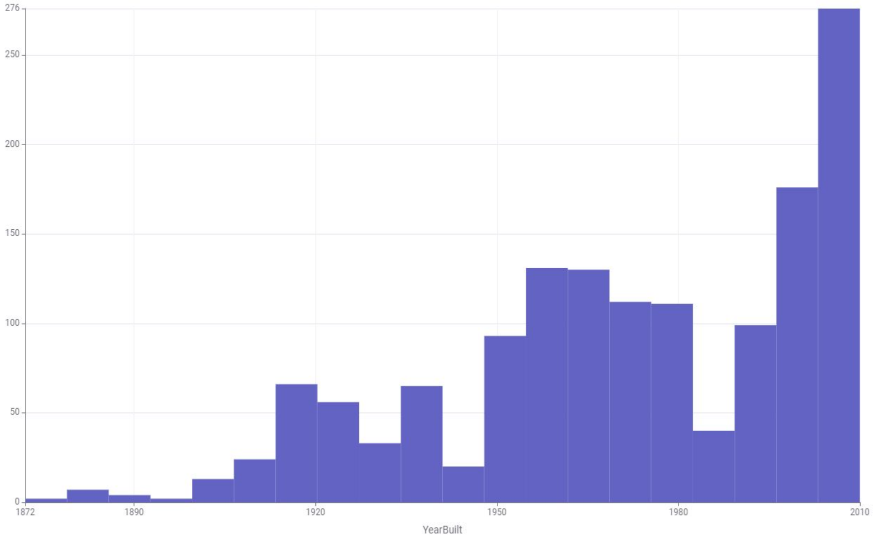
Rows: 81 | Columns: 14

Name	Type	# Missing values	# Unique values	Minimum	Maximum	25% Quantile	50% Quantile (Med...	75% Quantile	Mean	Mean Absolute De...	Standard Deviation	Sum	10 most common ...
Id	Number (integer)	0	1460	1	1,460	365.25	730.5	1,095.75	730.5	365	421.61	1,066,530	1 (1; 0.07%), 2 (1; 0.0
MSSubClass	Number (integer)	0	15	20	190	20	50	70	56.897	31.283	42.301	83,070	20 (536; 36.71%), 60
MSZoning	String	0	5	?	?	?	?	?	?	?	?	?	RL (1151; 78.84%), RI
LotFrontage	String	0	111	?	?	?	?	?	?	?	?	?	NA (259; 17.74%), 60

- A Combination of Strings & Numbers (The train and test have some different data types for some columns)
- Missing Values in many columns (Replaced by Median in Numbers & Mode for Strings)
- The Target Variable is Nominal, therefore this is a regression problem

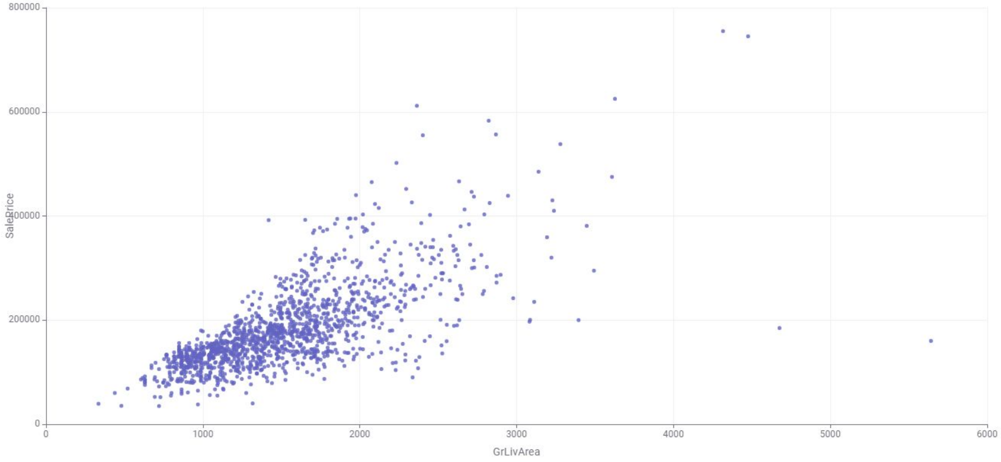
Data Visuals

- Most of the houses in the data set are built between 1960-2010s
- Which indicates the cost of the houses is on the higher end
- Potential Bias in the dataset since the houses are more modern



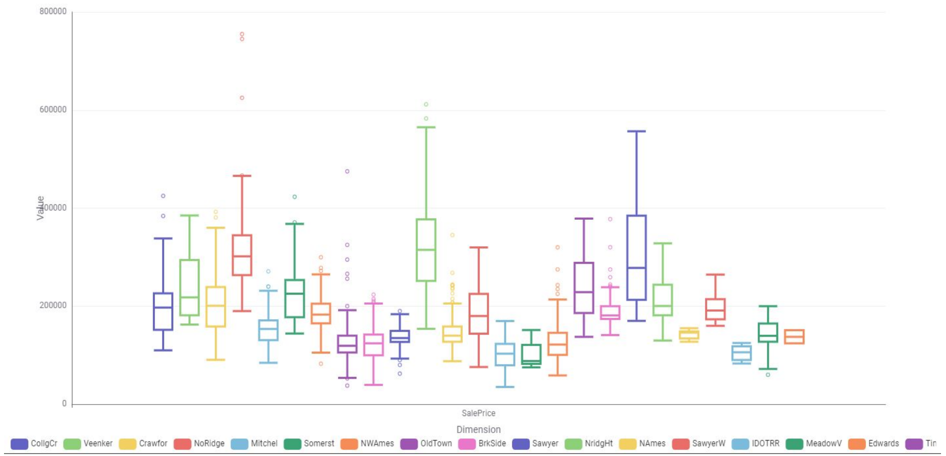
Data Visuals

- Higher Ground Living Area indicates a higher Price, but there is a lot of variability.
- Price Range is close around the 1000-2000 Area mark.
- Could be hard to predict price beyond 400



Data Visuals

- Box plot of Sale Price based on neighborhood
- For the most part there is an even distribution of the Sales Prices among the neighborhood
- Some Neighborhoods have less data compared to others which results in a very small box plot. (N Ames)

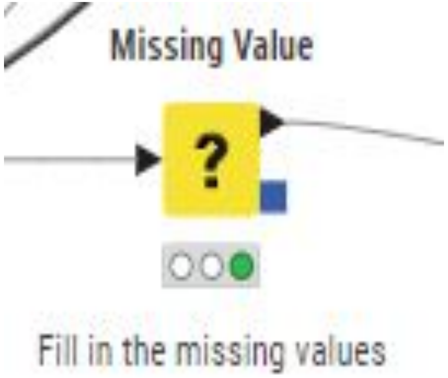


02

Data PreProcessing

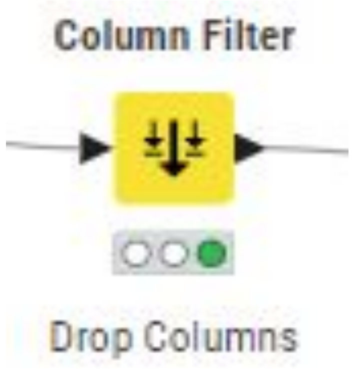
Data Cleaning

- Fill in Missing data with Median (Nominal columns)
- Fill in Missing data with Mode (String columns)



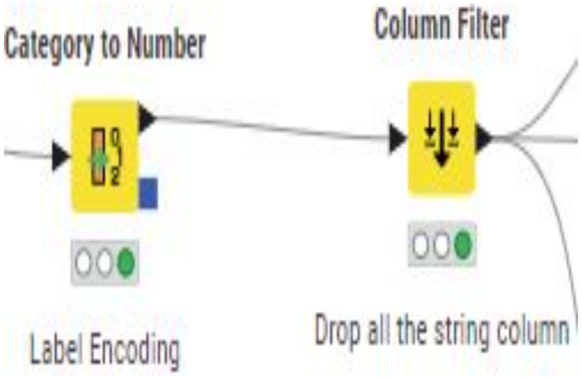
Data Cleaning

- Drop the Columns that had to many missing values: Alley, Misc Value, etc.



Data Cleaning

- Label Encoding the String Values
- Drop the String columns after the Encoding

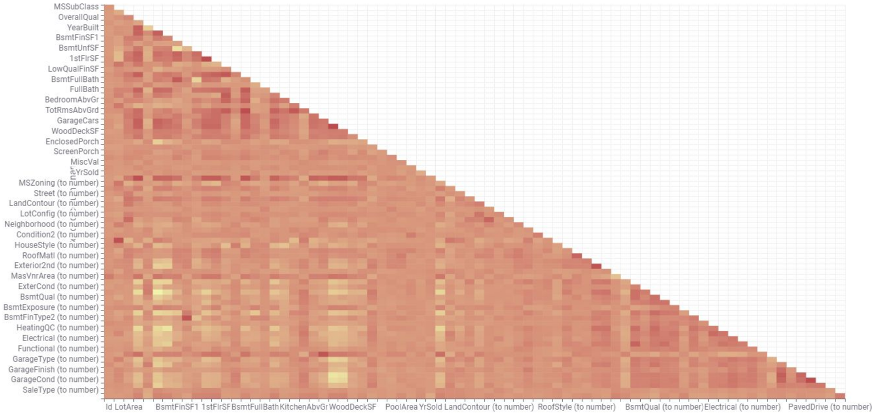


03

Feature Engineering

Correlation Map

- Created Correlation Map to pick features for feature engineering
- The highest Correlations for Sale Price are: Overall Qual (0.79), 1stFlrSF (0.6), and GrLiveArea (0.781)



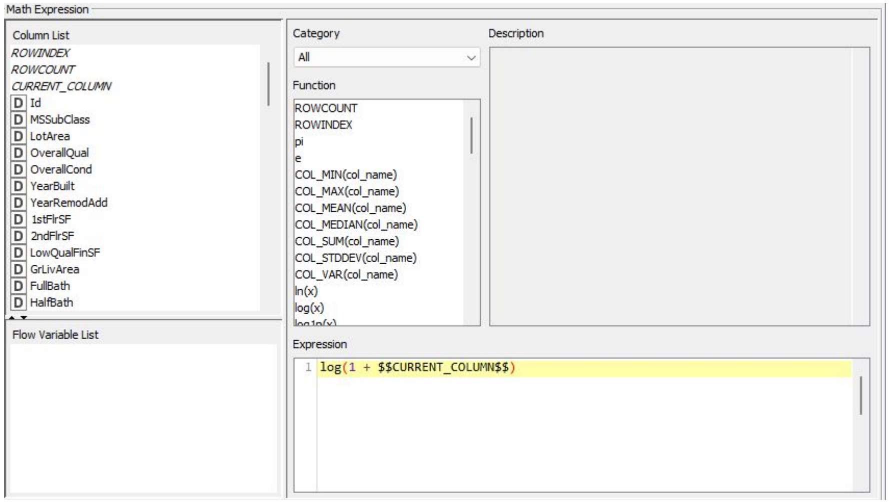
Add New Features

- Created Features based off highest correlations
- Created HouseAge, YearsSinceReModel, Total Bathrooms, and Liv area Ratio

```
$YrSold$ - $YearBuilt$

$YrSold$ - $YearRemodAdd$

$FullBath$ + (0.5 * $HalfBath$) + $BsmtFullBath$ + (0.5 * $BsmtHalfBath$)

$GrLivArea$ / $LotArea$
```

Log Transformation

- Log Transform the Skewed data for more even distribution
- Helps handle the outliers

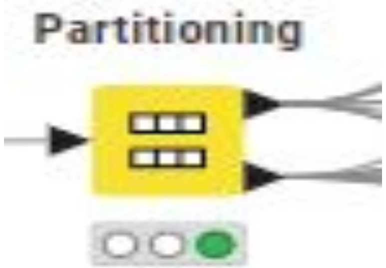


04

Machine Learning

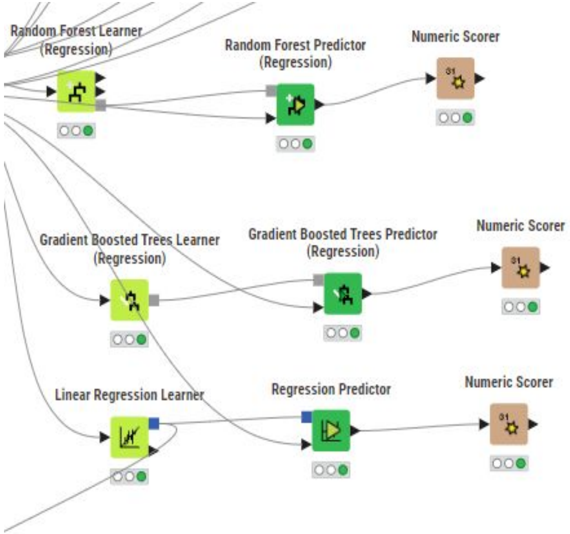
Partitioning

- Split the data of 80-20 percent
- 80 percent is training
- 20 percent is test



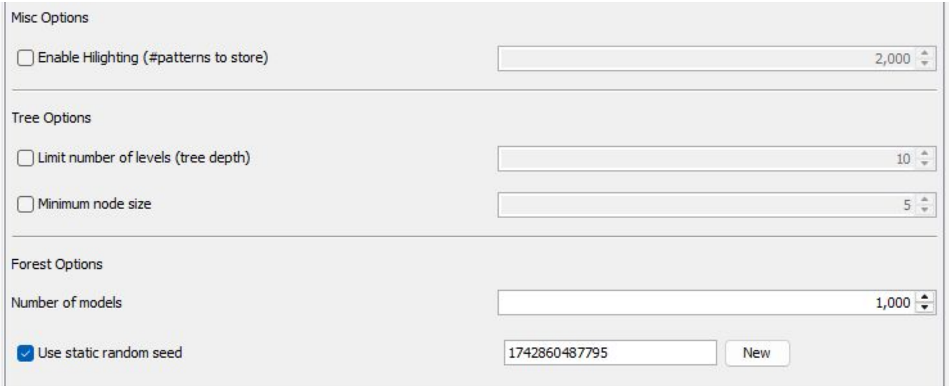
Training the Models

- Trained Three ML models: Random Forest, Gradient Boost, Linear Regression
- Used Numeric Scorer to evaluate the Models



Hyper Parameter Tuning

- Adjusted the Number of Models in Gradient and Random to 1,000
- Changed the Tree depths and checked to see increase of accuracy
- Higher Number of Models produced higher accuracy



Results (No Feature/Hyperparameter) vs With the Feature/Hyperparameter

Linear:

Model	Baseline (No Feature Engineering)	New Feature(s) Added	Features Created	Notes/Observations
Linear Regression	MAE: 20,980.46 RMSE: 31,419.19 R^2: 0.7895 Adjusted R^2: 0.7895 Mean Squared Error: 987,165,466.41 Mean Signed Difference: -1,396.58 MAPE: 0.1277	MAE: 0.0372 RMSE: 0.0597 R^2: 0.8790 Adjusted R^2: 0.8790 Mean Squared Error: 0.0036 Mean Signed Difference: 0.0028 MAPE: 0.0071	HouseAge, YearsSinceRemodel, TotalBathRooms, LivingAreaRatio, log transformations on skewed features	Linear Regression improved significantly after feature engineering but still lags behind Random Forest and Gradient Boosting in R^2 and RMSE.

Results (No Feature/Hyperparameter) vs With the Feature/Hyperparameter

Gradient:

Gradient Boosting	MAE: 15,828.04 RMSE: 23,056.82 R^2: 0.8866 Adjusted R^2: 0.8866 Mean Squared Error: 531,617,106.47 Mean Signed Difference: 1,914.32 MAPE: 0.0906	MAE: 0.0361 RMSE: 0.0540 R^2: 0.9011 Adjusted R^2: 0.9011 Mean Squared Error: 0.0029 Mean Signed Difference: 0.0009 MAPE: 0.0069	HouseAge, YearsSinceRemodel, TotalBathRooms, LivingAreaRatio, log transformations on skewed features	Gradient Boosting showed the best performance after feature engineering, with the highest R^2 and lowest RMSE among all models.

Results (No Feature/Hyperparameter) vs With the Feature/Hyperparameter

Random:

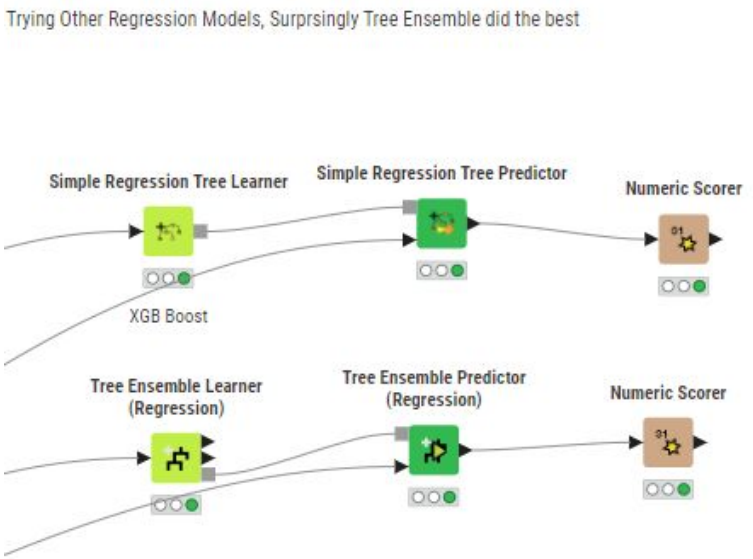
Model	Baseline (No Feature Engineering)	New Feature(s) Added	Features Created	Notes/Observations
Random Forest	MAE: 15,836.94 RMSE: 24,025.92 R^2: 0.8769 Adjusted R^2: 0.8769 Mean Squared Error: 577,244,885.66 Mean Signed Difference: 246.94 MAPE: 0.0929	MAE: 0.0393 RMSE: 0.0590 R^2: 0.8820 Adjusted R^2: 0.8820 Mean Squared Error: 0.0035 Mean Signed Difference: -0.0025 MAPE: 0.0075	HouseAge, YearsSinceRemodel, TotalBathRooms, LivingAreaRatio, log transformations on skewed features	Random Forest also improved significantly but performed slightly worse than Gradient Boosting in most metrics after feature engineering.

05

Conclusion & Bonus

Other ML Models

- Tried other ML models to cross check, and see if it performed better
- Simple Tree was the worst
- Tree ensemble was on par with Random Forest



Conclusion

- Used Linear Regression for final model since it had the best overall attributes out of the three models
- Kaggle submission got a score of 0.19