Housing in Ames, Iowa

Analyzing housing data to make your home ownership journey easier

DSI-SG-26: Project 2 (Lim Yangxiang)

The problem

Problem statement

Build a regression model to predict housing sale prices in Ames, Iowa. This model should support prospective homeowners assess list price reasonableness within Ames, Iowa, and inform their purchase decision.

As a project pre-requisite, this requires creating and iteratively refining a regression model to address the Ames, Iowa data set from Kaggle. Kaggle submissions determine outcome quality based on root mean squared error (rsme).

Methodology

Explore / Clean	Select / Iterate	Model / Assess
Exploratory visualizations	**Select and evaluate features iteratively**	Train and test models
Typecasting (e.g. as ‘category’ dtype)	Add / engineer features	Select production model
Data cleaning and other pre-processing steps		Assess and interpret

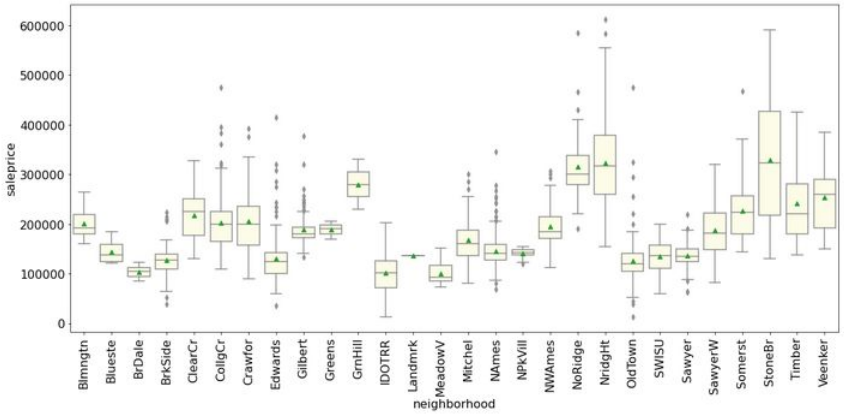
Explore

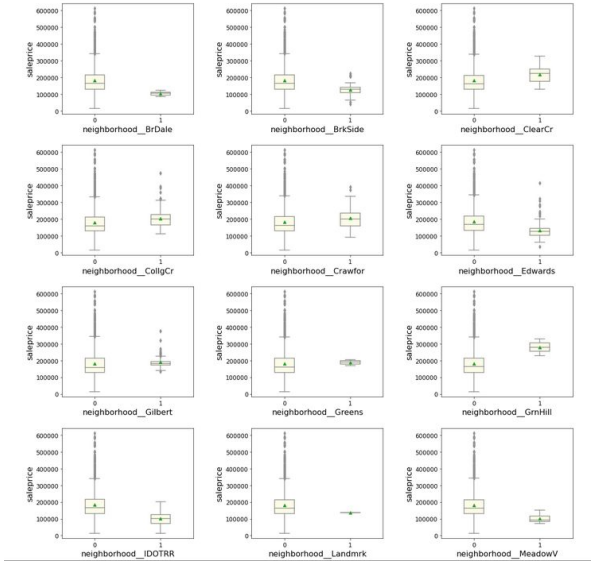
“Plain vanilla” displays

```
----- neighborhood -----
NAmes       310
CollgCr     180
OldTown     163
Edwards     143
Somerst     130
NridgHt     122
Gilbert     116
Sawyer      111
NWAmes       87
SawyerW      87
Mitchel      82
BrkSide      76
Crawfor      71
IDOTRR       69
NoRidge      48
Timber       48
StoneBr      38
SWISU        32
ClearCr      27
MeadowV      24
Blmngtn      22
BrDale       19
NPkVill      17
Veenker      17
Blueste       6
Greens        3
GrnHill       2
Landmrk       1
Name: neighborhood, dtype: int64

'Percentage of null_counts: 0.0%'

'Percentage of zero_counts: 0.0%'
```

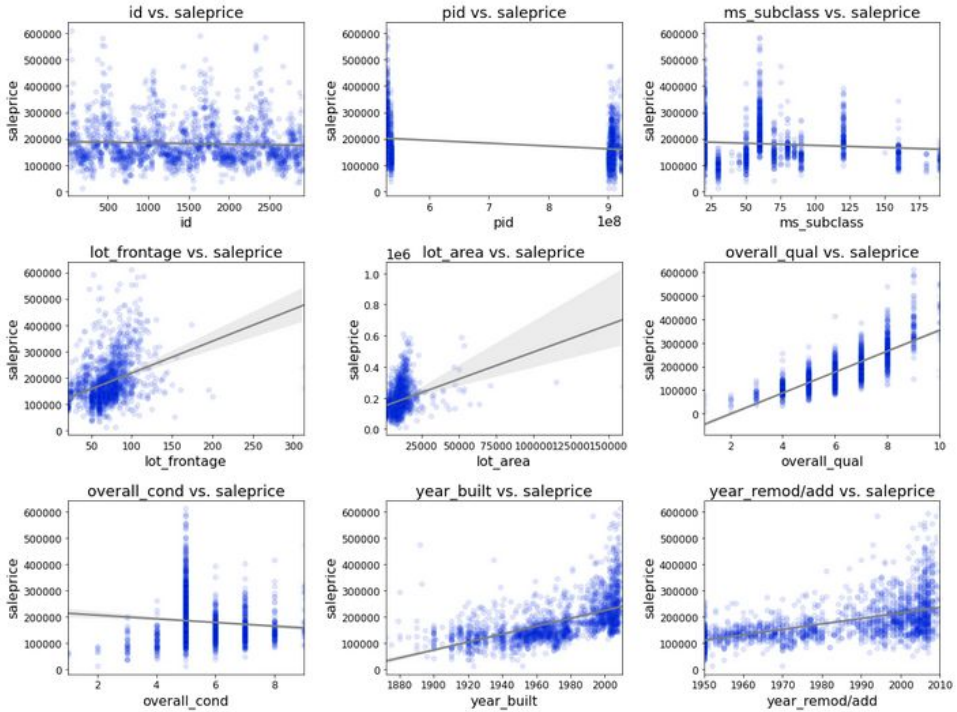




Multi-level boxplots (including get_dummies boxplot analysis)

Explore

Scatter-plot visualizations



Transformations

Managing skewness and heteroscedasticity with log(x+1) transformation

Standardization for Lasso model

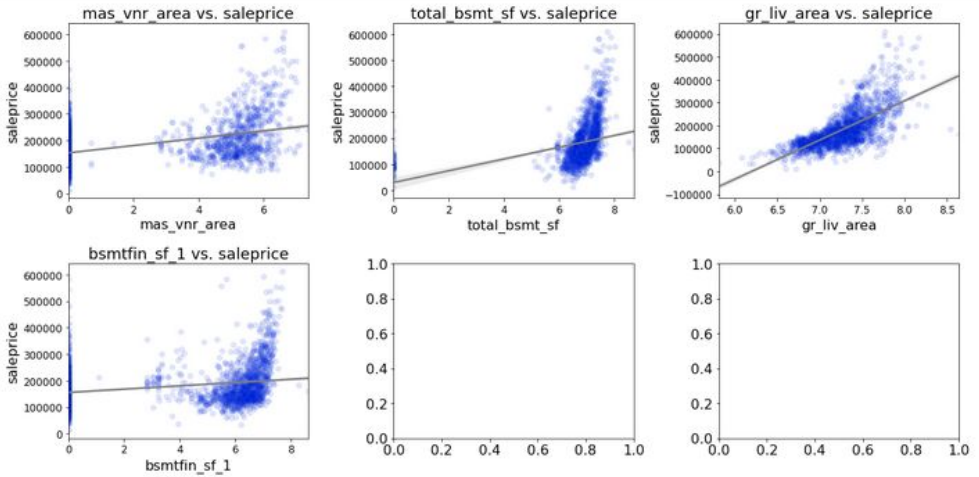
Using log(x+1) transformation to address heteroscedasticity & high zero-counts

In [55]:
```
# log(x+1) transformation for identified variables

def logx1_transform(df, col_to_transform):
    for col in col_to_transform:
        df[col] = np.log1p(df[col])

logx1_transform(df_train_clean, list_het_to_clean)
```

In [56]:
```
# visualize features after log(x+1) transformation
draw_subplot_scatter(df_train_clean, list_het_to_clean)
```



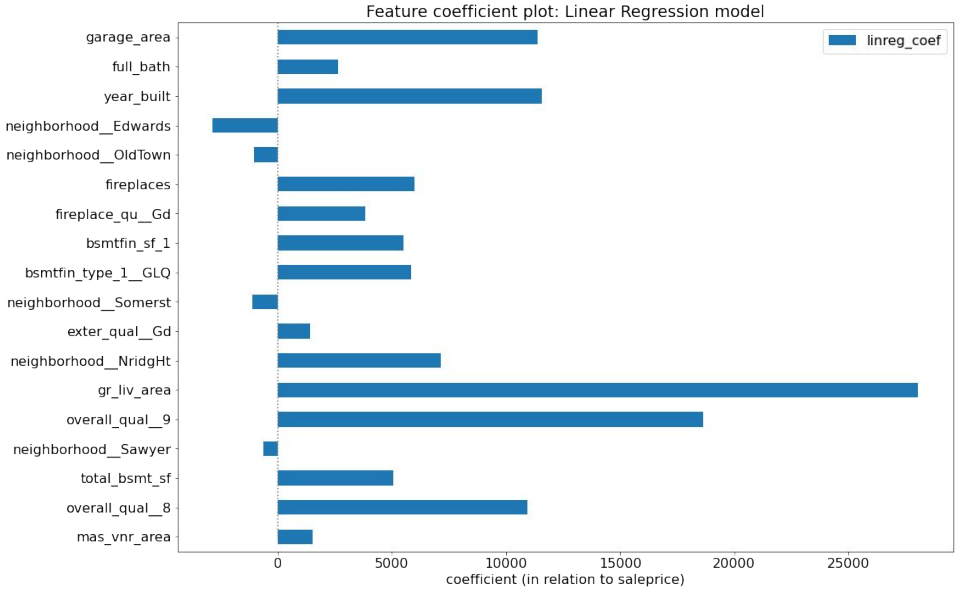
Production Model

Linear Regression with StandardScaler

Metrics

R2 score (CV) = 0.795

Kaggle RSME = 40,789



Production Model

Choose Linear Regression with StandardScaler

By Kaggle scores, our linear regression model seems to perform best.

- **Linear Regression with StandardScaler: 40,789.49472**
- Lasso with StandardScaler score: 40,811.07217
- RidgeCV with StandardScaler score: 40,838.36034
- ElasticNet with StandardScaler score: 43,960.71021

A linear regression model also has more interpretability due to the directness of how it works.

Takeaways

House **size**, house **quality**, house **features**, and housing **location** do have a demonstrable impact on how much a house can go for on the market.

Limitations: Market forces and agent (buyer / seller) psychology impact interpretation, location scope only within Ames, Iowa

Time-series information like past sales prices are usually very relevant in real life

Something to explore: price per square foot on net living area