Ames, Iowa

Capstone Project by Aakash Sharma

Introduction

- I am a Data Scientist and Data Engineer
- I am doing my masters degree in this field
- I work for IBM
- I enjoy talking about the new revolutionary algorithms and AI tools

About this Project

- I decided to explore and model a linear regression algorithm on the Ames, Iowa data set
- I've never fully explored the data set
- I perform multiple and different visual analytic methods, SQL statements and extensive statistical inference discussions on the data itself
- Extensively clean the data, performing preprocessing steps, and deciding the best methods for moving forward without losing key information
- Linear Regression Model using the Ridge Penalty

About the Data Set

- The Ames Housing Dataset is a well-known dataset in the field of machine learning and data analysis. It contains various features and attributes of residential homes in Ames, Iowa, USA. The dataset is often used for regression tasks, particularly for predicting housing prices.
- Number of Features: There are 81 different features or variables that describe various aspects of the residential properties.
- Target Variable: The target variable in the dataset is the "SalePrice," representing the sale price of the houses.
- Data Types: The features include both numerical and categorical variables, covering a wide range of aspects such as lot size, number of rooms, location, construction, and more.
- Data Collection occurred from Kaggle with further enrichment from Google Maps

Project Purpose

- Problem Statement: We are given a dataset that is relatively complete. The data set contains multiple errors that can corrupt the data, skew the graphs while making it difficult for us to find patterns or correlations to a variable we would like to predict. Our variable sales price can be used to sell someone on a dataset for considering changing or purchasing a house in a certain neighborhood.
- We want to clean the data & model it using linear, ridge & lasso regressions to predict the sales price of the house. We can pass in features of the data frame to help with predicting this or help correlate a relationship.

Executive Summary

EDA/Data Cleaning

EDA

- Look at distributions
- Look at correlations
- Look at relationships to target

Data Cleaning

- How to impute null value.
- How to handle outliers
- Want to combine any features
- **Want to have interaction terms**
- Want to manually drop collinear features

Model Prep

Train/Test Split

- Look at distributions

Scale the Model

- Standardize the data
- Useful for data which has negative values
- Arranges the data in a normal distribution

Instantiate the Model

Cross Validation

- On the 3 models to see which model is the best for our data
- Applied to more subsets.

The Model

Linear Regression

Lasso Regression - The Best

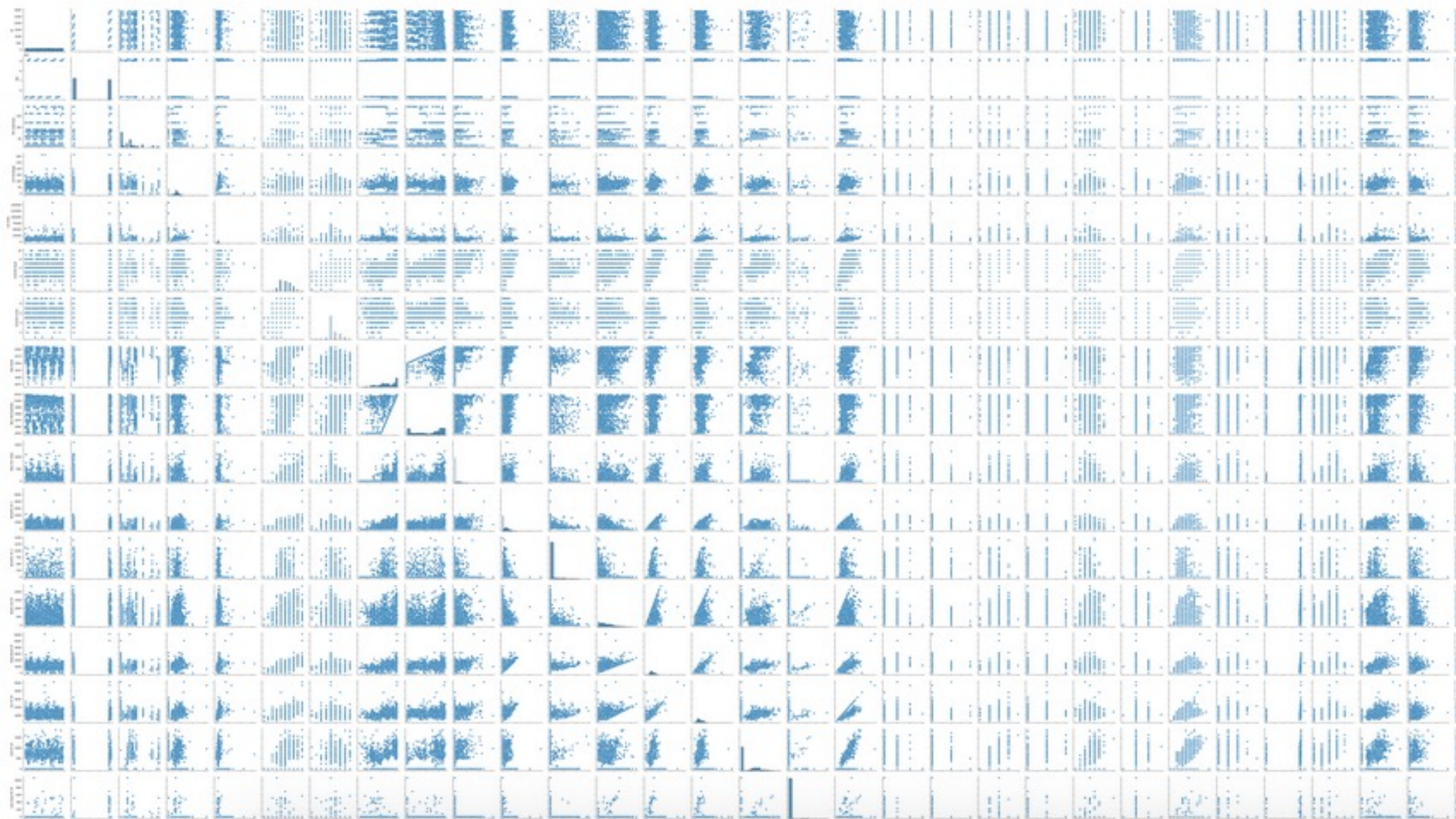
Ridge Regression - Runner Up

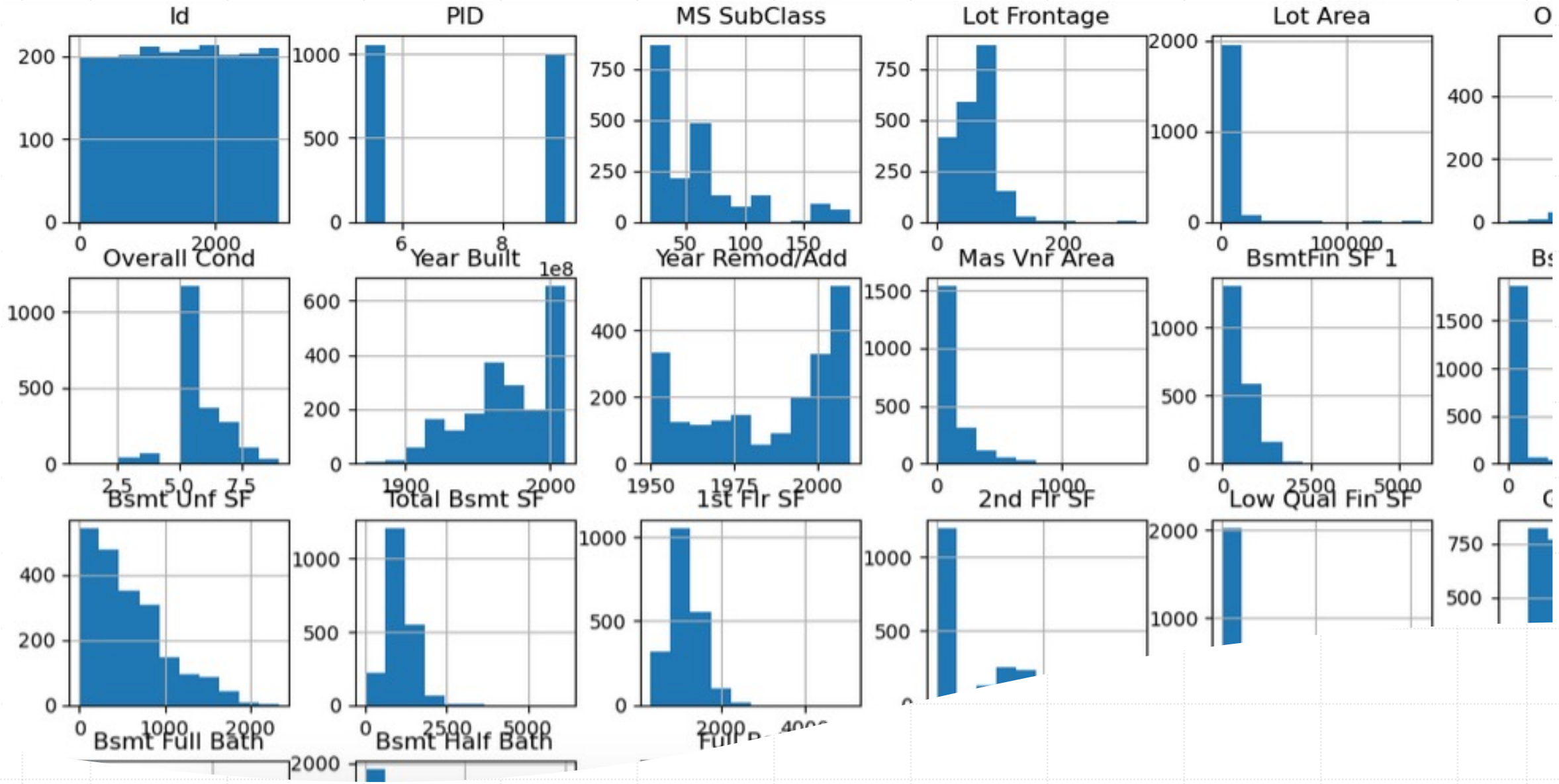
First Five Rows of the Training Data Set

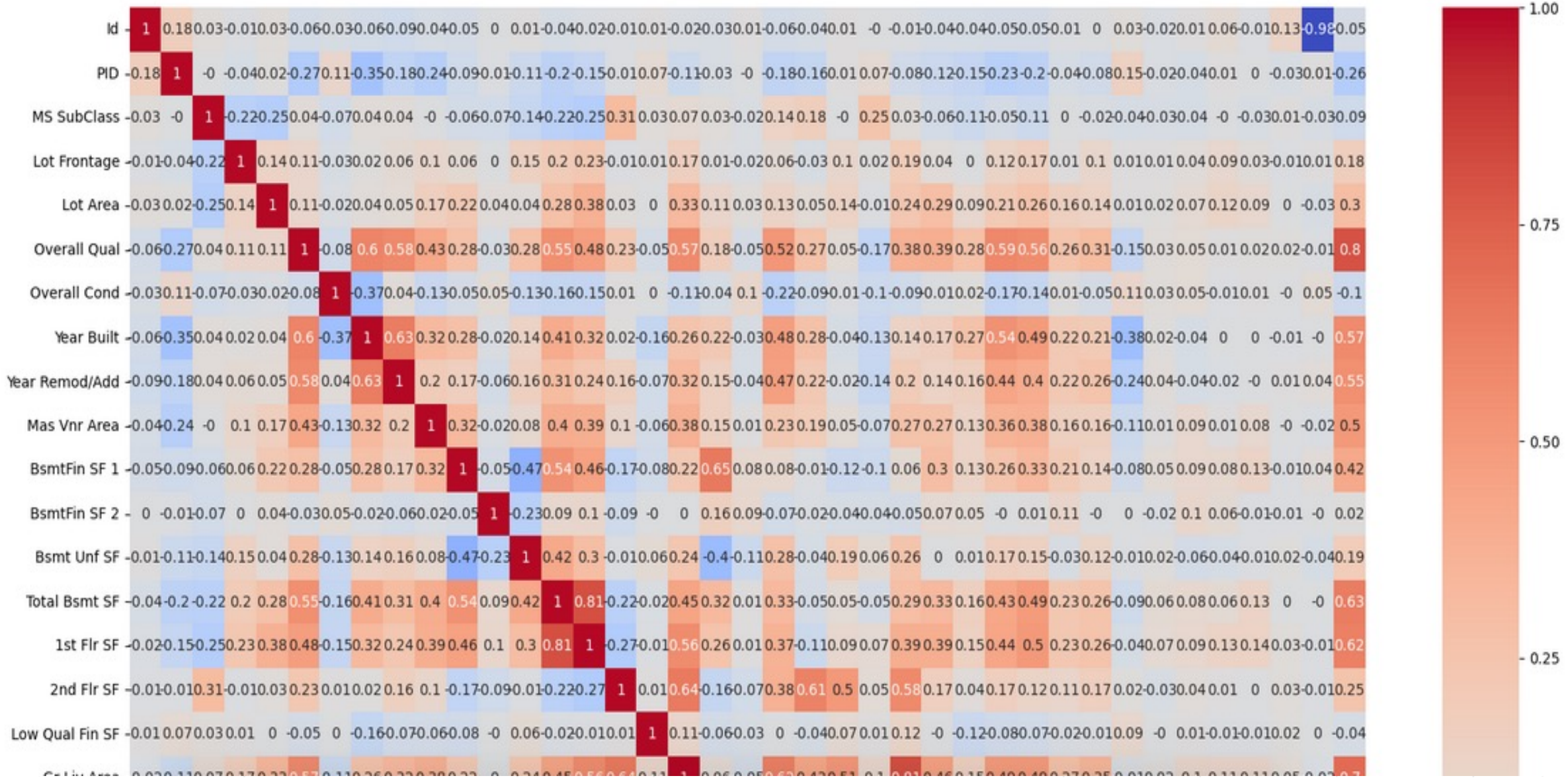
	Id	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Lot Shape	Land Contour	...	Screen Porch	Pool Area	Pool QC	Fence	Misc Feature	Misc Val	Mo Sold	Yr Sold	Sale Type	SalePrice
0	109	533352170	60	RL	NaN	13517	Pave	NaN	IR1	Lvl	...	0	0	NaN	NaN	NaN	0	3	2010	WD	130500
1	544	531379050	60	RL	43.0	11492	Pave	NaN	IR1	Lvl	...	0	0	NaN	NaN	NaN	0	4	2009	WD	220000
2	153	535304180	20	RL	68.0	7922	Pave	NaN	Reg	Lvl	...	0	0	NaN	NaN	NaN	0	1	2010	WD	109000
3	318	916386060	60	RL	73.0	9802	Pave	NaN	Reg	Lvl	...	0	0	NaN	NaN	NaN	0	4	2010	WD	174000
4	255	906425045	50	RL	82.0	14235	Pave	NaN	IR1	Lvl	...	0	0	NaN	NaN	NaN	0	3	2010	WD	138500

Initial Functions

- Wrote a function that uses the describe and info method(s) providing summary statistics of the numeric columns within the Data Frame, including count, mean, standard deviation, minimum, and maximum values, as well as quartile information.
- The .info() method offers a concise overview of the Data Frame, displaying information about the data types, non-null counts, and memory usage for each column, making it useful for quickly assessing the structure and completeness of the data.
- A pairplot visualizes pairwise relationships between variables, a histogram displays the distribution of a single variable, and a heatmap illustrates the correlation between variables in a matrix format.
  - Shown Respectively on the next 3 slides




Id
PID
MS SubClass
Lot Frontage
Lot Area
Overall Cond
Year Built
1e8
Year Remod/Add
Mas Vnr Area
BsmtFin SF 1
Bsmt Unf SF
Total Bsmt SF
1st Flr SF
2nd Flr SF
Low Qual Fin SF
Bsmt Full Bath
Bsmt Half Bath


Id
PID
MS SubClass
Lot Frontage
Lot Area
Overall Qual
Overall Cond
Year Built
Year Remod/Add
Mas Vnr Area
BsmtFin SF 1
BsmtFin SF 2
Bsmt Unf SF
Total Bsmt SF
1st Flr SF
2nd Flr SF
Low Qual Fin SF
1.00
0.75
0.50
0.25

SQL Analysis

- Created 4 different SQL Python Queries
- Selecting the first 5 rows
- Selecting all rows where SalePrice < 20,000 and Year Sold = 2010
- Aggregating data by grabbing all rows with Neighborhood, and the average SalePrice of all homes in each neighborhood then grouping by Neighborhood
- Calculating the minimum, maximum, average, and median SalePrice of homes – How expensive are the houses?
  - Cheapest house sold for $12,789
  - Most expensive for $611,657
  - Average sales price is $181,470
  - Median sales price is $162,500

SQL Queries

```python
# Example 2: Filtering rows with specific conditions
query = 'SELECT * FROM train_df WHERE SalePrice < 200000 AND "Yr Sold" = 2010'
result = pd.DataFrame(psql(query, locals()))
print("\nExample 2 - Rows with SalePrice < $200,000 and Yr Sold = 2010:")
result
```

```
Example 2 - Rows with SalePrice < $200,000 and Yr Sold = 2010:
```

	Id	PID	MS SubClass	MS Zoning	Lot Frontage	Lot Area	Street	Alley	Lot Shape	Land Contour	...	Screen Porch	Pool Area	Pool QC	Fence	Misc Feature	Misc Val	Mo Sold	Yr Sold	Sale Type	SalePrice
0	109	533352170	60	RL	0.0	13517	Pave	0	IR1	Lvl	...	0	0	0	0	0	0	3	2010	WD	130500
1	153	535304180	20	RL	68.0	7922	Pave	0	Reg	Lvl	...	0	0	0	0	0	0	1	2010	WD	109000
2	318	916386060	60	RL	73.0	9802	Pave	0	Reg	Lvl	...	0	0	0	0	0	0	4	2010	WD	174000
3	255	906425045	50	RL	82.0	14235	Pave	0	IR1	Lvl	...	0	0	0	0	0	0	3	2010	WD	138500
4	138	535126040	20	RL	137.0	16492	Pave	0	IR1	Lvl	...	0	0	0	0	0	0	6	2010	WD	190000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
162	306	911202100	50	C (all)	66.0	8712	Pave	Pave	Reg	HLS	...	0	0	0	0	0	0	1	2010	WD	102776
163	25	527402250	20	RL	0.0	12537	Pave	0	IR1	Lvl	...	0	0	0	0	0	0	4	2010	WD	149900
164	311	914475090	80	RL	74.0	9620	Pave	0	Reg	Lvl	...	0	0	0	GdPrv	Shed	80	5	2010	WD	190000
165	300	909455040	120	RM	35.0	3907	Pave	0	IR1	HLS	...	0	0	0	0	0	0	3	2010	WD	162500
166	10	527162130	60	RL	60.0	7500	Pave	0	Reg	Lvl	...	0	0	0	0	0	0	6	2010	WD	189000

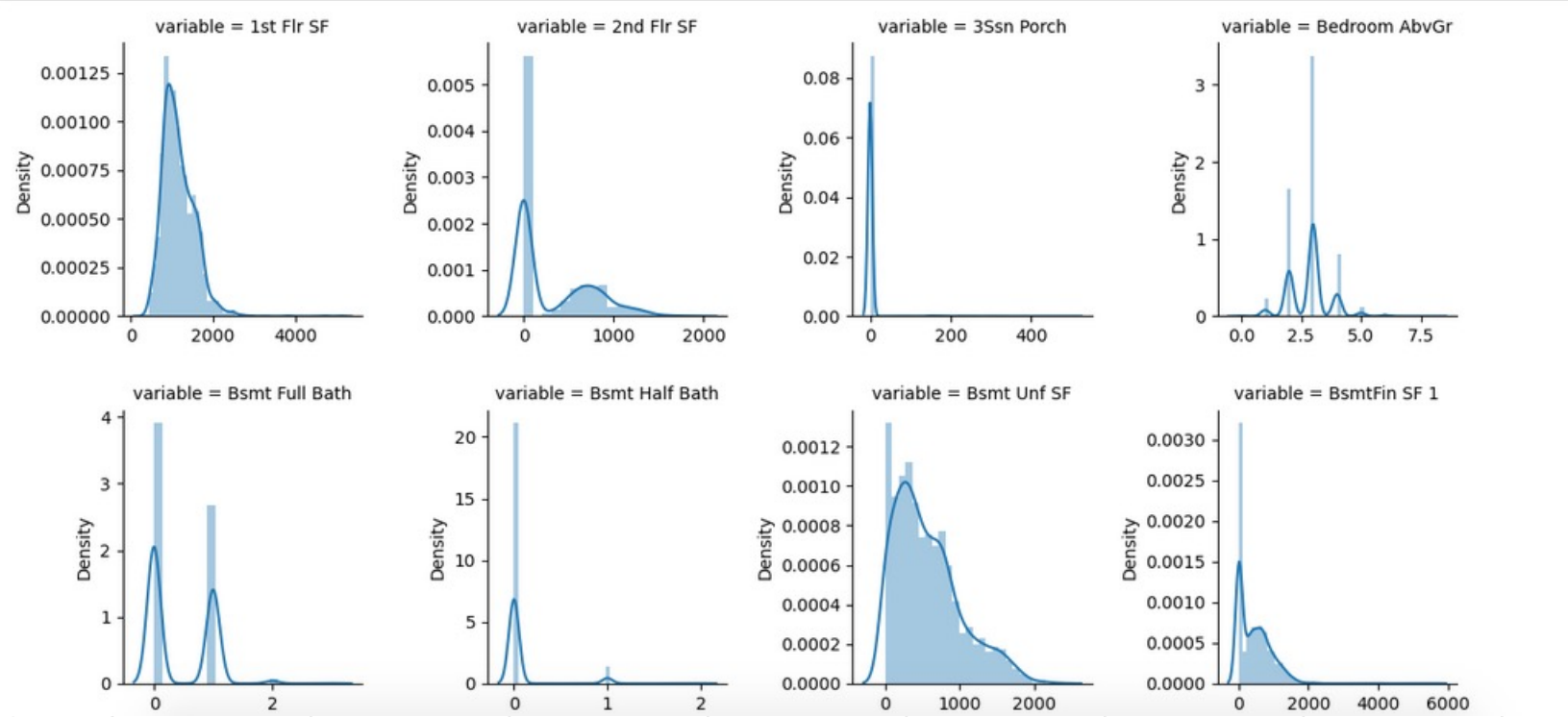
```python
# SQL query to calculate the minimum, maximum, average, and median SalePrice
sql_query = """
SELECT
    MIN(SalePrice) AS MinSalePrice,
    MAX(SalePrice) AS MaxSalePrice,
    AVG(SalePrice) AS AvgSalePrice,
    (SELECT SalePrice FROM train_df ORDER BY SalePrice LIMIT 1 OFFSET (SELECT COUNT(*) FROM train_df) / 2) AS MedianSalePrice
FROM
    train_df
"""

# Execute the SQL query using pandasql
result = pd.DataFrame(psql(sql_query, locals()))

# Display the result
print("How expensive are houses?")
print("The cheapest house sold for ${:,.0f} and the most expensive for ${:,.0f}".format(result['MinSalePrice'][0], result['Max
print("The average sales price is ${:,.0f}, while median is ${:,.0f}".format(result['AvgSalePrice'][0], result['MedianSalePric
```

```
How expensive are houses?
The cheapest house sold for $12,789 and the most expensive for $611,657
The average sales price is $181,470, while median is $162,500
```

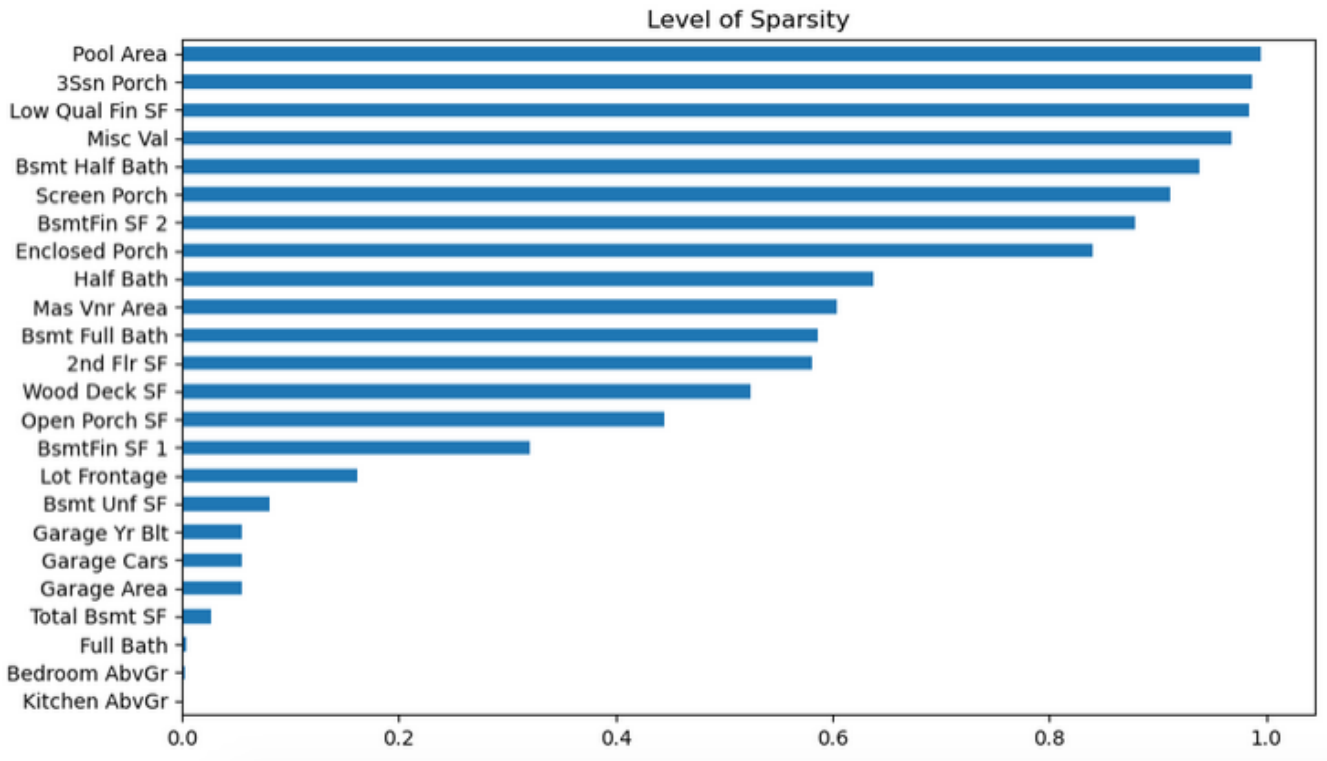
More Plots



Distribution Plot of different features on our SalesPrice variable

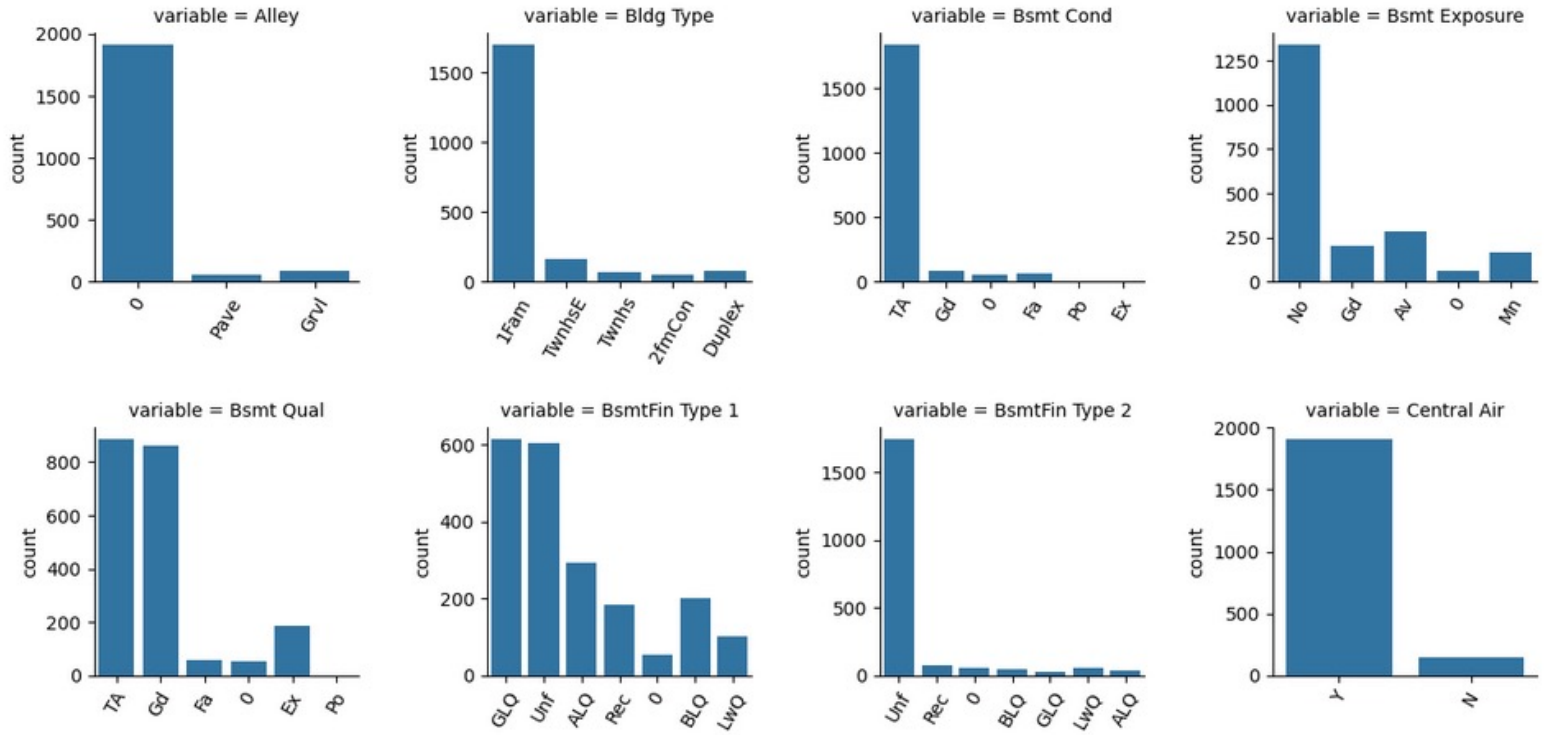
More Plots Cont.

- Looking at % of 0 values
- Horizontal Bar Graph representing the Sparsity within the correlated features



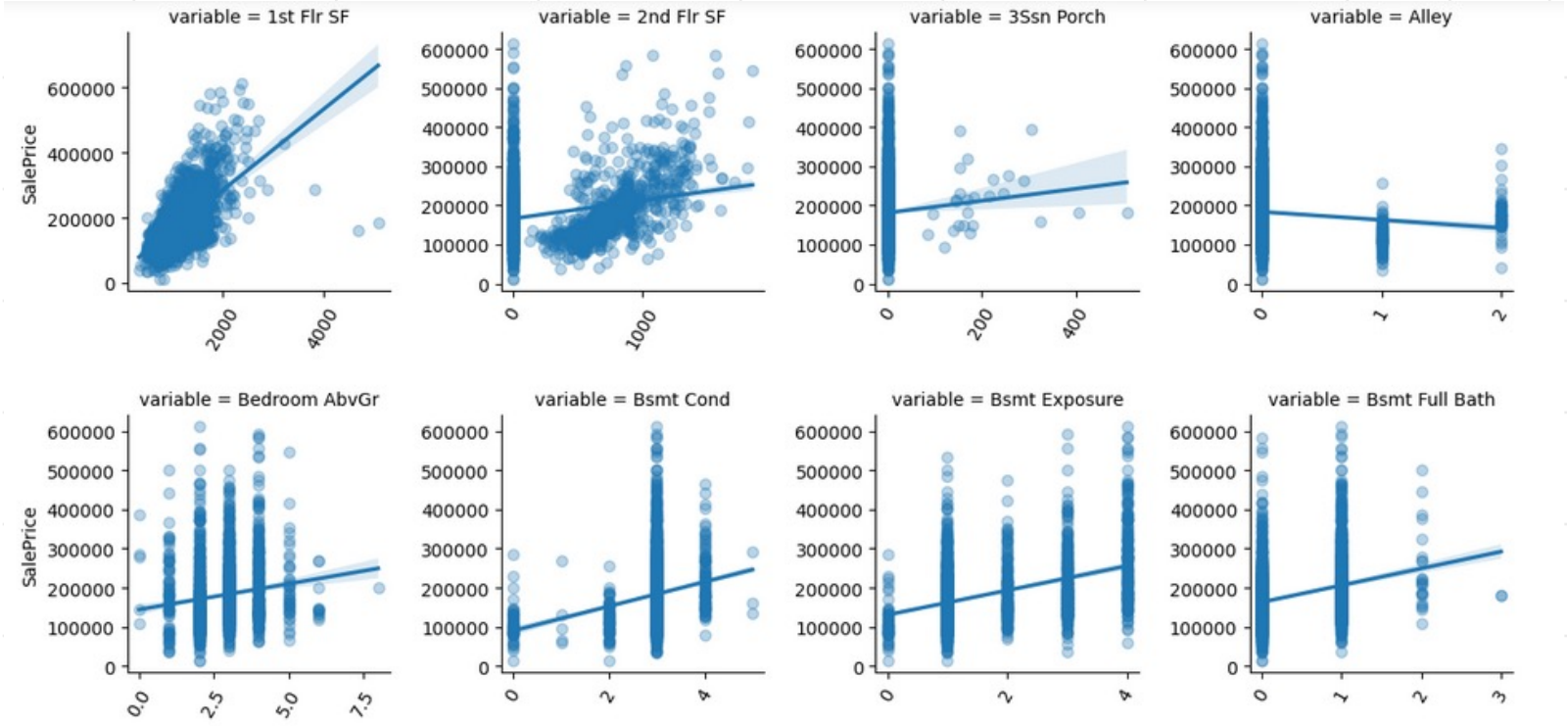
More Plots Cont.

- Distribution Plots for Categorical variables and their relationship to the SalePrice



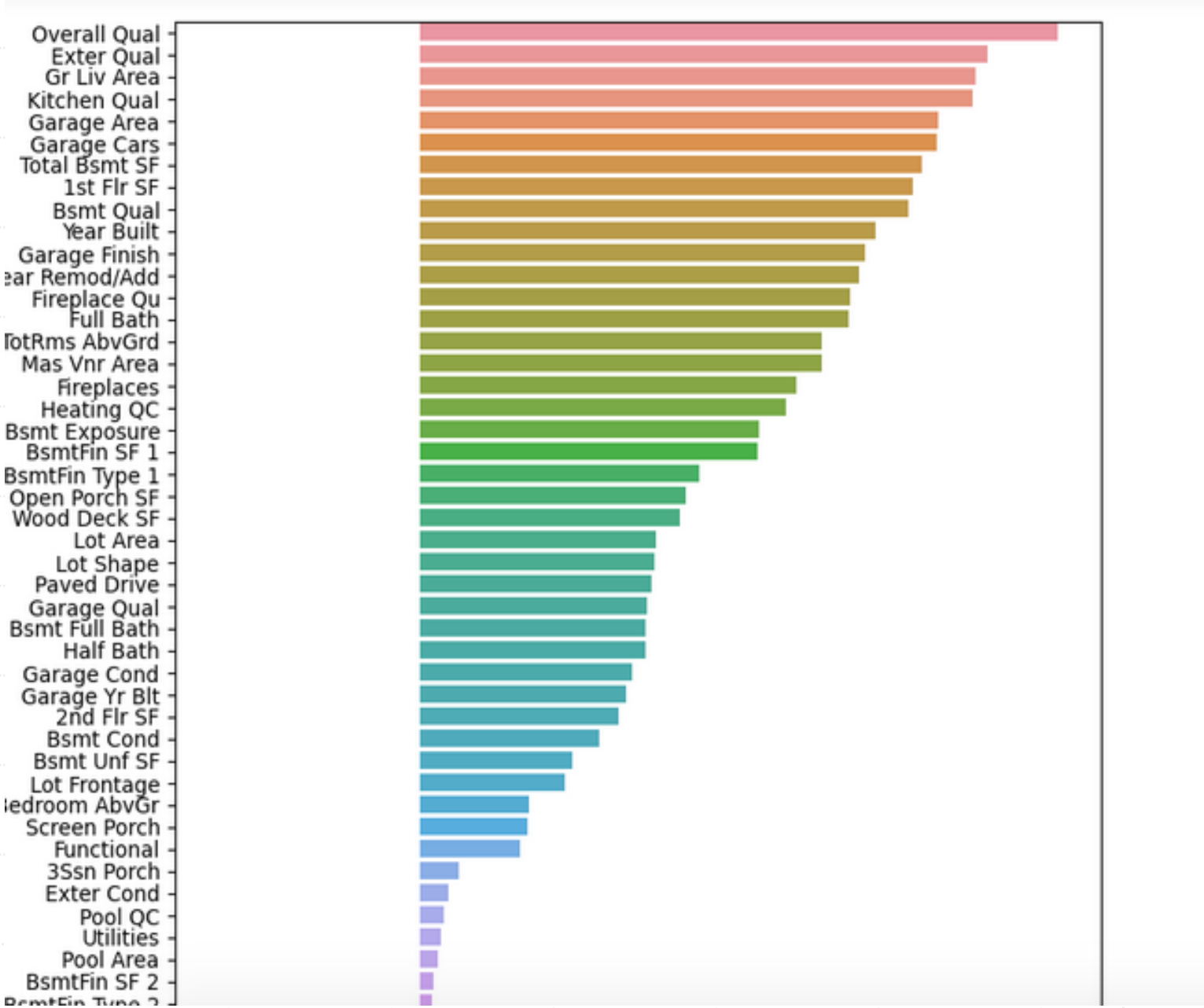
More Plots Cont.

- Bivariate Analysis
- Want to change from categorical to numerical because they contain ranked information – ordinal (e.g. quality ratings)



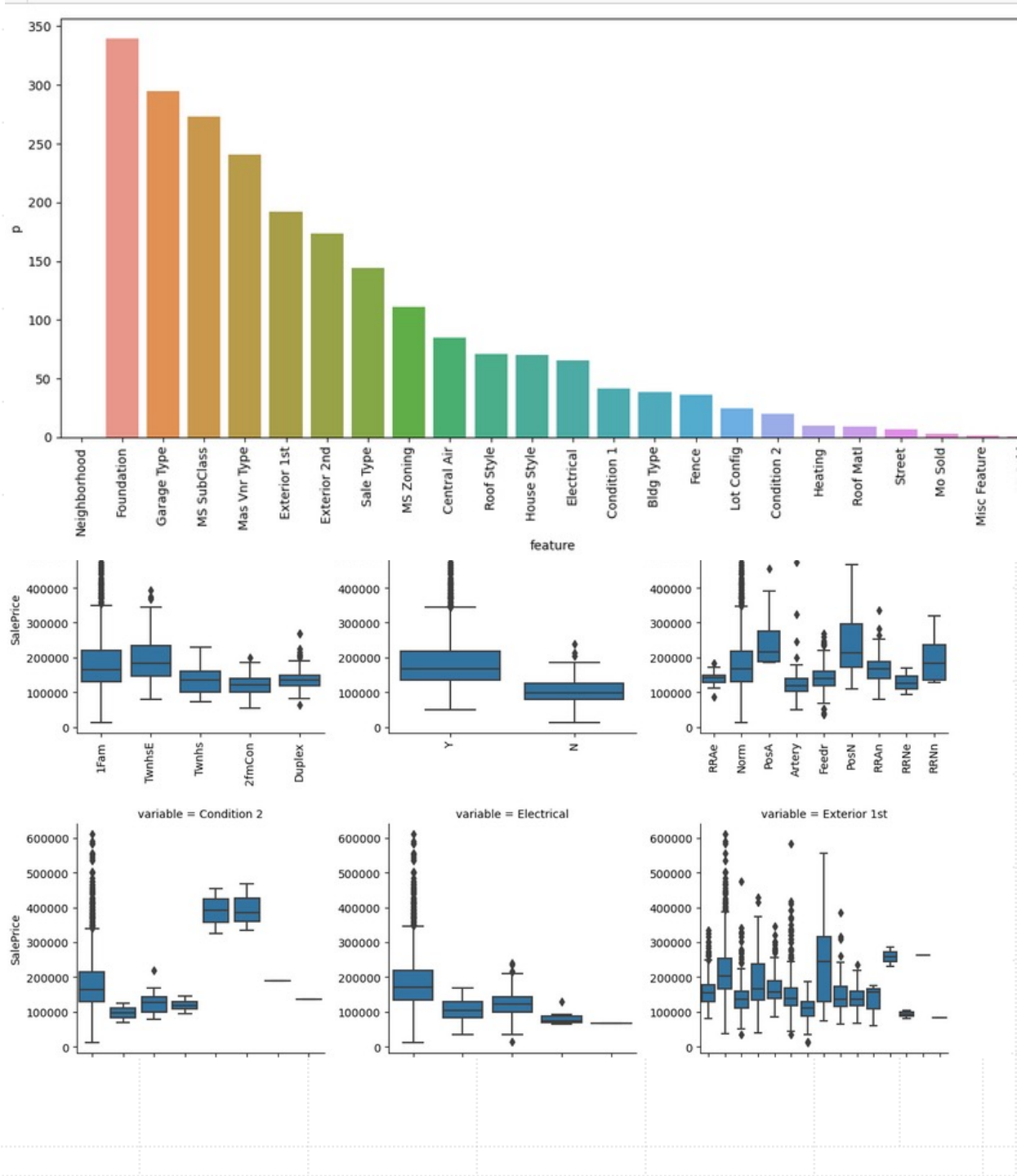
More Plots Cont.

- All the highly, positively correlated features on the SalePrice column
- Shown in descending order
- Least correlated, and negatively correlated are shown below
- Negatively correlated is perpendicular but not shown due to the amount of total features
- Correlated features are essential for predicting SalePrice



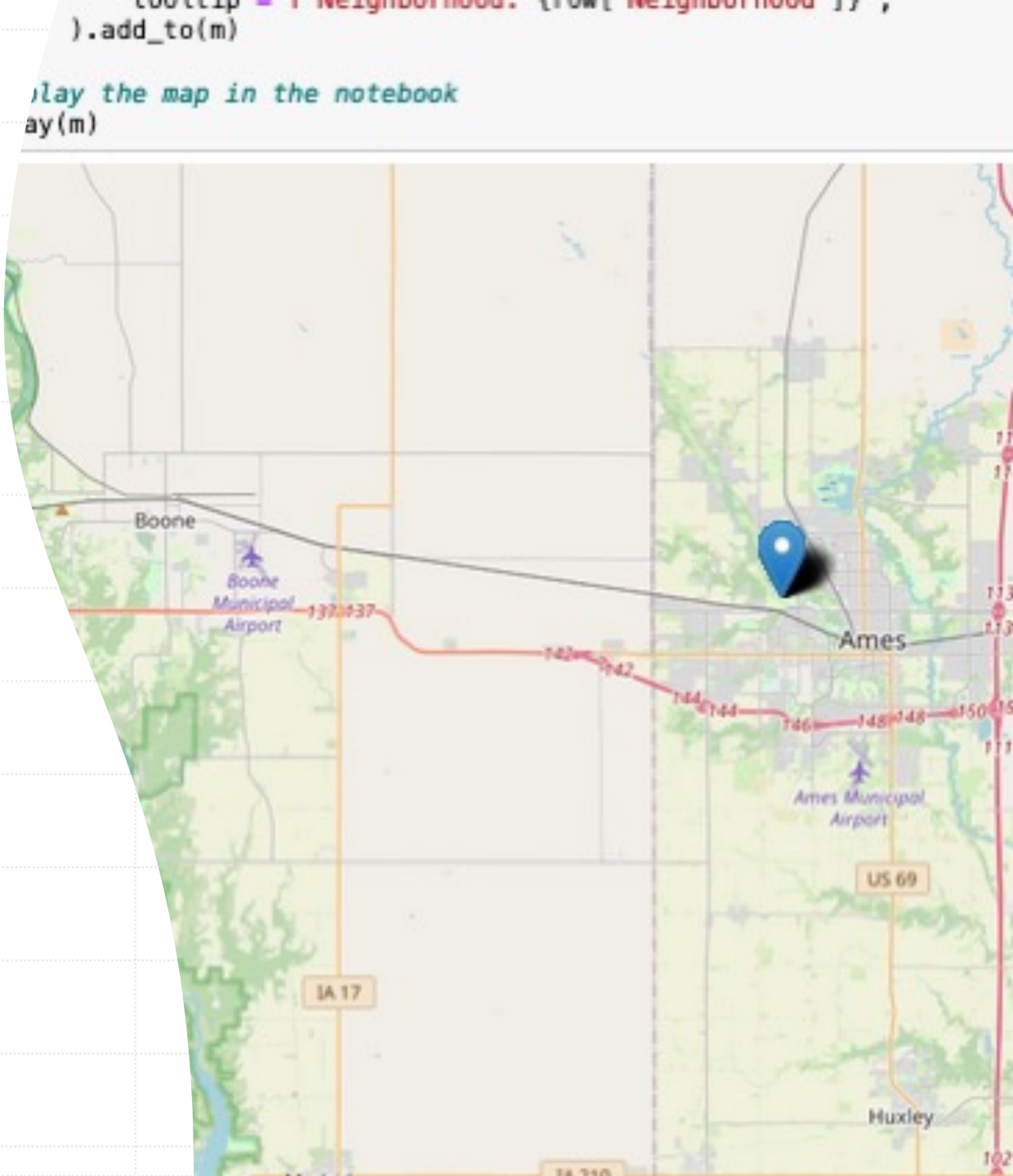
More Plots Cont.

- It looks like some features show significant variance in the mean of SalePrice between different groups, eg. Neighborhood, SaleType or MSSubClass.
- However, we'd like to have a better sense of which feature influences SalePrice more than others.
- ANOVA tests for each categorical feature againt SalePrice. This will give us both the F statistic and p-values for each feature. The higher the F statistic, the higher the p-value



Folium Interactive Map

- Calculated the Longitudes and Latitudes of the Neighbors in the data set
- Used Google Geo-encoding to find the names globally, clean our data set so the names match, and calculate the geocoordinates



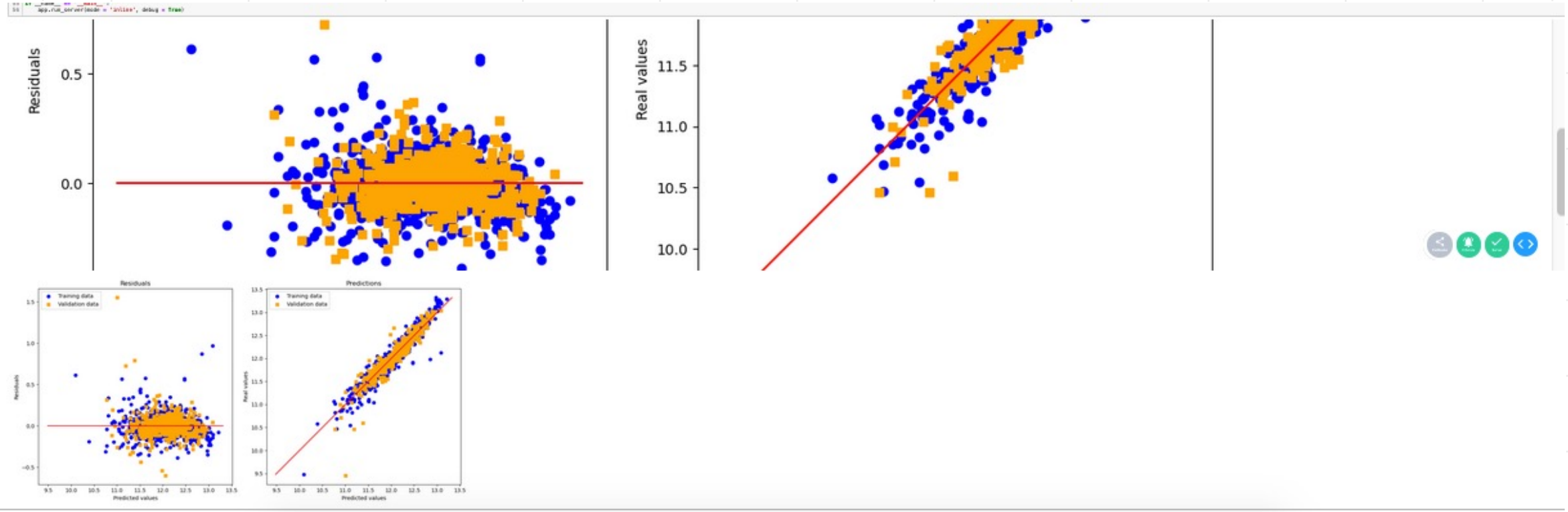
Data Wrangling and Data Modeling

- Applied Standardization (Standard Scaler) to transform features such that they have a mean of 0 and a standard deviation of 1
  - Used to normalize data, making it easier to compare and work with features that may have different scales or units, ensuring that no single feature dominates
- Error handling using the mean squared error to measure of the average squared difference between predicted values and actual values in a dataset
- Feature engineering, combining two or more features to derive further features
- Data Transformation using Logarithmic application
- One Hot Encoding – Binary Conversion
- Cross Validation
- Linear Regression with Ridge Penalty

Data Model Results with Ridge Penalty

```
Dummy Features: 16 | Train RSME: 0.111 | Test RSME: 0.075
Dummy Features: 23 | Train RSME: 0.111 | Test RSME: 0.077
Dummy Features: 25 | Train RSME: 0.109 | Test RSME: 0.094
Dummy Features: 30 | Train RSME: 0.109 | Test RSME: 0.094
Dummy Features: 58 | Train RSME: 0.118 | Test RSME: 0.099
Dummy Features: 67 | Train RSME: 0.118 | Test RSME: 0.101
Dummy Features: 75 | Train RSME: 0.119 | Test RSME: 0.108
Dummy Features: 80 | Train RSME: 0.119 | Test RSME: 0.113
Dummy Features: 88 | Train RSME: 0.120 | Test RSME: 0.096
Dummy Features: 94 | Train RSME: 0.118 | Test RSME: 0.096
Dummy Features: 100 | Train RSME: 0.115 | Test RSME: 0.097
Dummy Features: 115 | Train RSME: 0.114 | Test RSME: 0.126
Dummy Features: 130 | Train RSME: 0.137 | Test RSME: 0.130
Dummy Features: 135 | Train RSME: 0.138 | Test RSME: 0.135
Dummy Features: 141 | Train RSME: 0.139 | Test RSME: 0.136
Dummy Features: 146 | Train RSME: 0.143 | Test RSME: 0.161
Dummy Features: 148 | Train RSME: 0.158 | Test RSME: 0.168
Dummy Features: 153 | Train RSME: 0.157 | Test RSME: 0.136
Dummy Features: 160 | Train RSME: 0.256 | Test RSME: 2497910230.555
Dummy Features: 165 | Train RSME: 0.137 | Test RSME: 3401606773.304
Dummy Features: 171 | Train RSME: 0.157 | Test RSME: 170120851.825
Dummy Features: 183 | Train RSME: 0.155 | Test RSME: 3223220457.379
Dummy Features: 188 | Train RSME: 0.155 | Test RSME: 407669117.792
Dummy Features: 197 | Train RSME: 0.152 | Test RSME: 1093977.978
```

Dash Dashboard with Model Results



Recommendations

- Which features appear to add the most value to a home?
  - Square Feet, Overall Quality, Garage Area/Cars, Beds & Baths
- Which features hurt the value of a home the most?
  - Having a kitchen above the garage, the condition of the home
- What are things that homeowners could improve in their homes to increase the value?
  - Homeowners could renovate the home, make sure the overall condition is on the same level as the overall quality
- What neighborhoods seem like they might be a good investment?
  - Ames, Iowa was ranked the #1 choice based on median monthly rent of $785 according to “The Crazy Tourist.” The best neighborhoods would be those that fit in that area and according to the data, that would

Further Research and Insight

- Do you feel that this model will generalize to other cities?
- How could you revise your model to make it more universal OR what date would you need from another city to make a comparable model?
  - I would add other features that could correlate to an increase in a sales price of a house. Maybe even use some interaction-based columns to help see correlations between 2 or multiple columns.
  - The model would not generalize to other cities based on the sample of an entire population.
  - We would have to use every possible variation of independent variable which differs tremendously to measure every variation of dependent variables.

Conclusion

- Started off building an intuition about the data, dove into the details, made decisions about what to do with our features, and finally compared several different regression models. In the end we achieved the best results using the LASSO regression.
- Next steps would be to do additional feature engineering, further experiment with other models, and start combining them into ensembles to push the envelope.
- I chose to stick to transparent linear models where the goals of both prediction and statistical inference may be pursued together. This meant running ordinary least squares (OLS), Ridge and Lasso regression models. Please note that the analysis I undertook was on the full original Ames not the Kaggle dataset