Housing Prices

Eyes to Analyze
(Derek H., Clemance K., Felipe L., Aderonke A.)

Introduction

- The purpose of our presentation was to predict housing pricing for a small town - Ames, Iowa.
- Our team joined a Kaggle competition which gave us the dataset.
- We chose this competition due to our shared interests in finances and housing market.

Questions our Project Answered

1. What is the relationship between overall quality over sales price?
2. Do homes remodeled after 2005 have better conditions than those remodeled before 2005?

The Dataset

- The datasets we used were multiple large datasets consisting of approximately 80 columns per dataset.
- The tools we used to analyze the dataset was:
    - Jupyter Notebook
    - Python

Data Steps

- First, we imported the datasets.
- To clean the data we removed the null values and transformed the categorical features to numerical form so our model will run.
- Next, we created a new data set by listing the variables.
- We shifted through the variables to find the variables with poor correlation to the Sales Price.
- From there, we started the machine learning process.

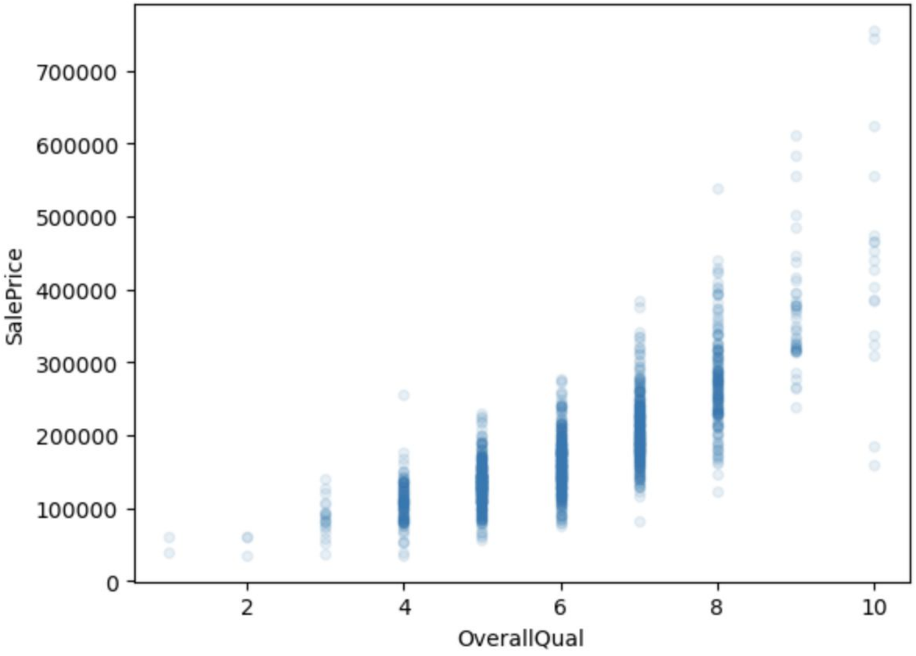
Exploring our Dataset

- We evaluated how much each of the data feature correlated with the Sales Price.
- The correlation results range from -1 to 1.
  - When the coefficient is closer to 1, there is a positive correlation.
  - When it is closer to 0, there is no correlation.
  - When it is closer to -1, there is a strong negative correlation

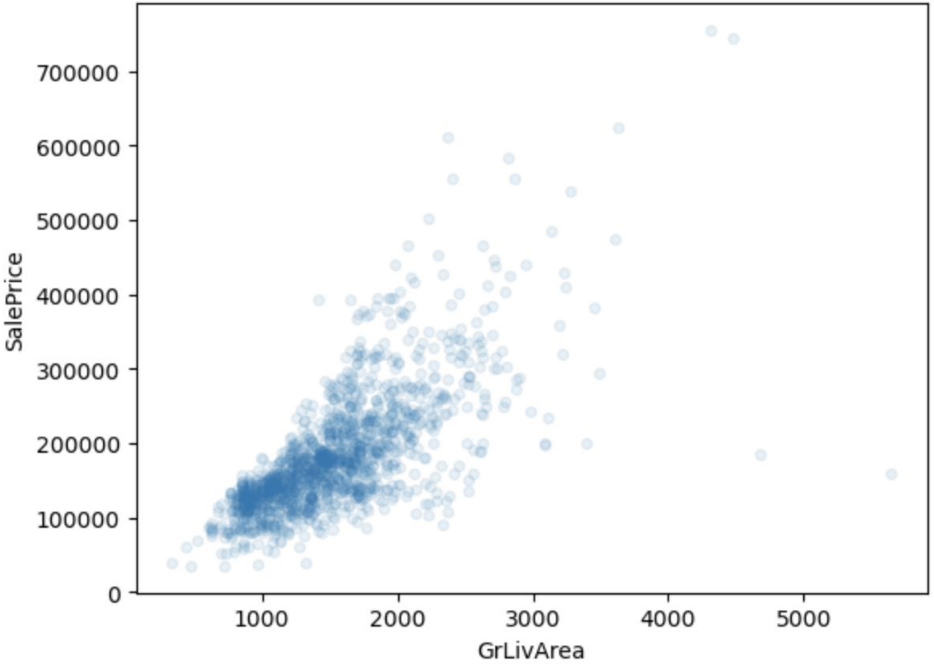
In [14]:
```
corr_matrix["SalePrice"].sort_values(ascending=False)
```

Out[14]:
```
SalePrice        1.000000
OverallQual      0.790982
GrLivArea        0.708624
GarageCars       0.640409
GarageArea       0.623431
TotalBsmtSF      0.613581
1stFlrSF         0.605852
FullBath         0.560664
TotRmsAbvGrd     0.533723
YearBuilt        0.522897
YearRemodAdd     0.507101
GarageYrBlt      0.486362
MasVnrArea       0.477493
Fireplaces       0.466929
BsmtFinSF1       0.386420
LotFrontage      0.351799
WoodDeckSF       0.324413
2ndFlrSF         0.319334
OpenPorchSF      0.315856
HalfBath         0.284108
LotArea          0.263843
BsmtFullBath     0.227122
BsmtUnfSF        0.214479
BedroomAbvGr     0.168213
ScreenPorch      0.111447
PoolArea         0.092404
MoSold           0.046432
3SsnPorch        0.044584
BsmtFinSF2      -0.011378
BsmtHalfBath    -0.016844
MiscVal         -0.021190
Id              -0.021917
LowQualFinSF    -0.025606
YrSold          -0.028923
OverallCond     -0.077856
MSSubClass      -0.084284
EnclosedPorch   -0.128578
KitchenAbvGr    -0.135907
Name: SalePrice, dtype: float64
```

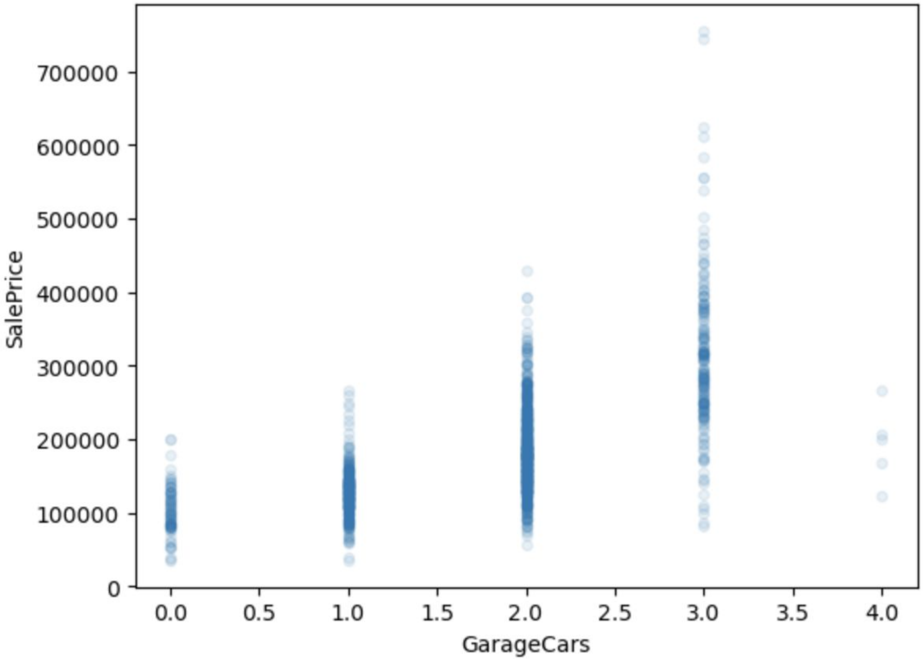
Sales Price vs Overall Quality



SalesPrice/Great Living Area



Sales Price/Garage



Data Cleaning

This is a vital aspect of our project as this determined the success of our model. The process consisted of:

1. Substituting our null values with the median value of the numerical data using the imputer function.
2. Transforming our categorical data to numerical data using the Hot Encoder Function.

Selecting and Training a Model

We used supervised machine learning regression models to analyze the continuous variable in the dataset which was the Sales Price.

We started out with three options:

1. Decision Tree Regressor
2. Random Forest Regressor
3. Gradient Boosting Regressor

Scikit Learn's Cross Validation

To make our decision, we evaluated our dataset using Scikit Learn's Cross Validation Feature. This feature split our dataset in 10 subsets, trains, and evaluated each data subset 10 times. The advantages include:

A. Giving us an estimate of our model's performance
B. Calculated the Standard derivation which lets us know how precise the estimate is.

Fine Tuning the Model

- To fine-tune our model we needed to decide on the best hyperparameter to use.

- To help with this, we utilized Scikit Learn’s GridSearch function.
    - Using GridSearch, we were not only able to determine the best hyperparameter but we were also able to determine the best estimator to pair the hyperparameter with.

- With our model decided, we are successfully able to make predictions.

Results

```
house_price_predictions = pd.DataFrame({"id":X_test.Id, "SalePrice":housing_predictions})

house_price_predictions.to_csv('price_predictions.csv', index=False)
```

```
predictions = pd.read_csv('price_predictions.csv')
```

```
y_test.to_csv('given_price.csv' , index=False)
```

```
given_price = pd.read_csv('given_price.csv')
```

```
predictions["given_price"] = given_price
```

```
predictions.head()
```

	id	SalePrice	given_price
0	893.0	140871.108818	154500
1	1106.0	325577.314975	325000
2	414.0	110867.154656	115000
3	523.0	155851.268078	159000
4	1037.0	326279.438576	315500

Conclusion/Summary

Many features have an impact on the pricing of the home such as:

- Built in garage
- Greater living area included
- Pool
- Alley Feature

Next Steps

Our next steps would be to answer some other pressing questions.

Such as,

- What are the top 5 most expensive neighborhoods?
- What is the average selling price of different building types?