Project 2 – House Price Prediction

Milestone 3 : White Paper

****

College of Science and Technology, Bellevue University

DSC680-T301 Applied Data Science (2245-1)

Sashidhar Bezawada

April 26 , 2024

Topic : House Price Prediction

- **Introduction**

  In recent years, the real estate industry has experienced rapid growth, making it an interesting area for research and experimentation. One common task in this field is forecasting house prices, which can be challenging due to the complex and dynamic nature of the market. In this experiment, we will perform EDA and build models, to analyze data and make predictions about house prices. By applying this technique, we aim to gain insights into the factors that affect house prices and improve our ability to forecast them accurately.

- **Business Problem**

  Develop a machine learning model to address the below:

  - Predict the sales price for each house.
  - Evaluate Root-Mean-Squared-Error (RMSE) between the logarithm of the predicted value and the logarithm of the observed sales price

- **Datasets**

  I am using the dataset, which is from Kaggle. The Ames Housing dataset was compiled by Dean De Cock for use in data science education. It's an incredible alternative for data scientists looking for a modernized and expanded version of the often cited Boston Housing dataset.

  https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data

File descriptions:

train.csv - the training dataset having 1460 records and 81 columns.

data_description.txt - full description of each column, originally prepared by Dean De Cock but lightly edited to match the column names used.

Here's a brief version of what you'll find in the data description file:

- SalePrice - the property's sale price in dollars. This is the target variable that you're trying to predict.
- MSSubClass: The building class
- MSZoning: The general zoning classification
- LotFrontage: Linear feet of street connected to property

- LotArea: Lot size in square feet
- Street: Type of road access
- Alley: Type of alley access
- LotShape: General shape of property
- LandContour: Flatness of the property
- Utilities: Type of utilities available
- LotConfig: Lot configuration
- LandSlope: Slope of property
- Neighborhood: Physical locations within Ames city limits
- Condition1: Proximity to main road or railroad
- Condition2: Proximity to main road or railroad (if a second is present)
- BldgType: Type of dwelling
- HouseStyle: Style of dwelling
- OverallQual: Overall material and finish quality
- OverallCond: Overall condition rating
- YearBuilt: Original construction date
- YearRemodAdd: Remodel date
- RoofStyle: Type of roof
- RoofMatl: Roof material
- Exterior1st: Exterior covering on house
- Exterior2nd: Exterior covering on house (if more than one material)
- MasVnrType: Masonry veneer type
- MasVnrArea: Masonry veneer area in square feet
- ExterQual: Exterior material quality
- ExterCond: Present condition of the material on the exterior
- Foundation: Type of foundation
- BsmtQual: Height of the basement
- BsmtCond: General condition of the basement
- BsmtExposure: Walkout or garden level basement walls
- BsmtFinType1: Quality of basement finished area
- BsmtFinSF1: Type 1 finished square feet
- BsmtFinType2: Quality of second finished area (if present)
- BsmtFinSF2: Type 2 finished square feet
- BsmtUnfSF: Unfinished square feet of basement area
- TotalBsmtSF: Total square feet of basement area
- Heating: Type of heating
- HeatingQC: Heating quality and condition
- CentralAir: Central air conditioning
- Electrical: Electrical system
- 1stFlrSF: First Floor square feet
- 2ndFlrSF: Second floor square feet

- LowQualFinSF: Low quality finished square feet (all floors)
- GrLivArea: Above grade (ground) living area square feet
- BsmtFullBath: Basement full bathrooms
- BsmtHalfBath: Basement half bathrooms
- FullBath: Full bathrooms above grade
- HalfBath: Half baths above grade
- Bedroom: Number of bedrooms above basement level
- Kitchen: Number of kitchens
- KitchenQual: Kitchen quality
- TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)
- Functional: Home functionality rating
- Fireplaces: Number of fireplaces
- FireplaceQu: Fireplace quality
- GarageType: Garage location
- GarageYrBlt: Year garage was built.
- GarageFinish: Interior finish of the garage
- GarageCars: Size of garage in car capacity
- GarageArea: Size of garage in square feet
- GarageQual: Garage quality
- GarageCond: Garage condition
- PavedDrive: Paved driveway
- WoodDeckSF: Wood deck area in square feet
- OpenPorchSF: Open porch area in square feet
- EnclosedPorch: Enclosed porch area in square feet
- 3SsnPorch: Three season porch area in square feet
- ScreenPorch: Screen porch area in square feet
- PoolArea: Pool area in square feet
- PoolQC: Pool quality
- Fence: Fence quality
- MiscFeature: Miscellaneous feature not covered in other categories.
- MiscVal: $Value of miscellaneous feature
- MoSold: Month Sold
- YrSold: Year Sold
- SaleType: Type of sale
- SaleCondition: Condition of sale

- **Exploratory Data Analysis**

The dataset doesn't have any duplicates or missing values. The category attributes were also converted to numerical types using one-hot encoding. Most of the columns seem to be poorly correlated with one another. Generally, when making a predictive model, it would be preferable to train a model with features that are not too correlated with one another so that we do not need to deal with redundant features. All the variables were explored for numeric and non-numeric separately. Nearly 20 Columns were dropped as they were not adding value or poor correlation. For eight Variables, Labelling of Categories in Numerical Features were performed.

After the dataset was cleansed and transformed, different explorations were performed to gain a better understanding of the data especially in relation to the target variable ('SalePrice'). In assessing the 'SalePrice' variable's class distribution, it became apparent that the classes were highly imbalanced in their representation of the home sales , as visualized in **Appendix**.

Besides investigating how to predict the Saleprice of the homes, I also reviewed the factors which were important in the home sale price.
The Saleprice figures are skewed towards left. So, I have applied the log transformation to obtain a centralized data. Based on the bar plots, properties in some of the neighborhoods are high priced. Overall Condition & Overall Quality of the Homes also contributed in the Higher Sale Price of the homes. Based on the Joint plot and the scatterplot, there is a positive relationship between the columns 'SalePrice' and 'GrLivArea' because if the values of one variable have increased, so does the other. And the other joint plot shows the negative relationship between the columns 'SalePrice' and 'PropAge' , increase in Property Age shows a decreasing saleprice trend i.e newer the property, high is the value.

Correlation plots are instrumental in house price prediction machine learning. They help identify important features by revealing their strength of association with house prices. By prioritizing highly correlated features, correlation plots streamline feature selection and enhance model interpretability. Using this I could understand how much a feature affects the final price of the house. However a major problem with this data set is the fact that both variables are in the dataset which could lead to multicollinearity. (as visualized in **Appendix**.)

Scatter plots are essential visual tools in the realm of house price prediction. They helped me understand how different features relate to house prices by visualizing the relationships between them. Scatter plots helped me detect any patterns, identify outliers, assess correlation, explore feature engineering possibilities, and evaluate model performance. This can be seen in the next phase of this assignment as I used the scatter plots to find outliers and hopefully it resulted in a more accurate representation of the housing market. (as visualized in **Appendix**.)

For model selection, 4 models which are appropriate for a classification model were used to evaluate performance accuracy. The models evaluated are listed below :

- **LinearRegression** - Linear regression seeks to find the best-fitting straight line that describes the relationship between the input variables and the output variable. This line is determined by minimizing the difference between the predicted values from the model and the actual values observed in the data.However this model assumes that the model is a linear relationship, however in most real-life situations it is hard to draw a accurate conclusion soley on the use of a linear relationship between two variables

- **Random Forest** - The non-linear model I chose is the Random Forest, this is because random forest is a machine learning algorithm that builds multiple decision trees during training, each trained on a random subset of the data and using a random subset of features at each split. It combines the predictions of these trees (either by voting for classification or averaging for regression) to produce a final prediction. Random Forest is known for its high accuracy, resistance to overfitting, and ability to handle noisy data effectively.

- **GradientBoost Classifier** - Gradient boosting works by building simpler (weak) prediction models sequentially where each model tries to predict the error left over by the previous model.

- **CatBoostRegressor** - CatBoost is a powerful and efficient machine-learning library for gradient boosting on decision trees. It is particularly well-suited for regression tasks, where the goal is to predict a continuous variable.

Results interpretation :

Accuracy Score : Examining the updated accuracy score reveals distinct performance characteristics among the algorithms. GradientBoosting Regressor stands out with an impressive accuracy score of 0.8554, signifying its exceptional ability to explain 85.54% of the variance in the target variable. Maintaining a robust position, CatBoost Regressor yields an accuracy score of 0.8411, showcasing reliable predictive capabilities with an explanation of approximately 84.11% of the variance. Random Forest, though slightly behind with an accuracy score of 0.8237, still provides a strong explanation of around 82.37% of the variance. Linear regression has the lowest of all with an accuracy score of 0.8106. These results underscore the nuanced strengths of each algorithm, with GradientBoosting demonstrating superior accuracy.

RMSE : GradientBoosting outperforms all 4 in terms of RMSE, achieving the lowest score of 28911.1100. This suggests that GradientBoosting provides more accurate predictions.

So, the best result is obtained by **Gradient Boosted Classifier**.

- **Ethical Considerations**

  Legal compliance: This dataset is a fictional dataset from Kaggle and doesn’t have any legal issue.

  The housing market in itself is an incredibly unpredictable entity. The supply and the demand for houses is slowly starting to decrease, which is putting home buyers in a tricky spot for finding affordable houses. Therefore, with the excitement around buying a home needs to be ensured that the lifestyle of Americans is taken into account from an ethical standpoint. Hopefully, the information is not utilized to scam home buyers or home sellers, but it is a possibility that needs to be envisioned.

- **Assumption**

  This lack of representation was a challenge and limitation in being able to reap value from the predictive modeling portion of this project.
  However, this dataset was limited in scope in that it contained actual data. I made the assumption that this would be enough information to create reasoning on the Home Sale Price, but it would have been beneficial to find other datasets that focused more on other livability features.

- **Challenges/Issues**

  The predictive modeling portion of this project posed a great challenge, in terms of yielding even relatively acceptable accuracies and RMSE. The target variable had 81 classes, and once the dataset was split into training and test sets, there was not enough representation per class to give the models proper training.

- **Conclusion**

  With the use of data analytics, many individuals can stay informed on the factors which play into market hotness such as location, population and time as explored above. By bringing everything under one sphere of investigation, it is easier to consolidate patterns, trends and identify the factors that can give best value for their homes. The decision of buying a home is one that should not be made lightly, and one should consider all helpful factors, to make an informed decision to buy the home at an appropriate price , with all possible living situations for their needs and wants.

- **Future Uses**

  This information can be used as reference for understanding the Sale Price of the Homes.

- **Recommendations**

  In the current model, feature reduction is not explored to it's full extent. And that can be considered to make this model better.

- **Implementation Plan**

  As we already discussed in the methodology section about some of the implementation details. So, the language used in this project is Python programming. We're running python code in anaconda navigator's Jupyter notebook. Jupyter notebook is much faster than Python IDE tools likePyCharm or Visual studio for implementing ML algorithms. The advantage of Jupyter notebook is that while writing code, it's really helpful for Data visualization and plotting some graphs like histogram

- **References**

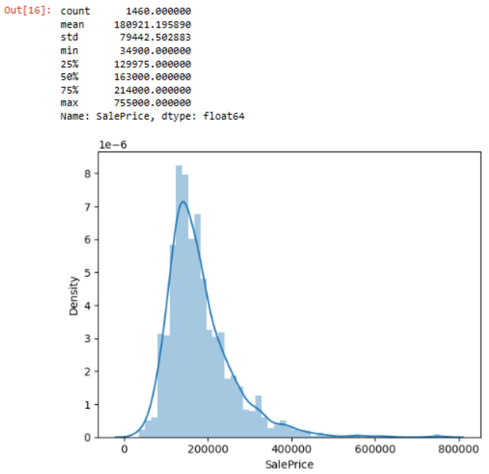
Dataset : https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data

1. "How Much Is My House Worth?" - By Brai Odion-Esene, Rachel Witkowski
https://www.forbes.com/advisor/mortgages/how-much-is-my-house-worth/

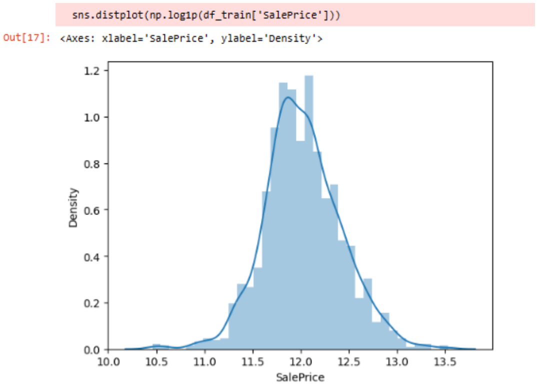
2. "How Much Is My Home Worth?" By AJA MCCLANAHAN,
https://www.investopedia.com/how-much-is-my-home-worth-5213913

3. "U.S. Housing Analysis: Factors Explaining Variations in Prices Independent Econometric Project" Written by Jianbin Chen,
https://www.researchgate.net/publication/340418499_US_Housing_Analysis_Factors_Explaining_Variations_in_Prices_Independent_Econometric_Project

- ## Appendix A - Target Class Distribution



```
Out[16]: count      1460.000000
         mean     180921.195890
         std       79442.502883
         min       34900.000000
         25%      129975.000000
         50%      163000.000000
         75%      214000.000000
         max      755000.000000
         Name: SalePrice, dtype: float64
```

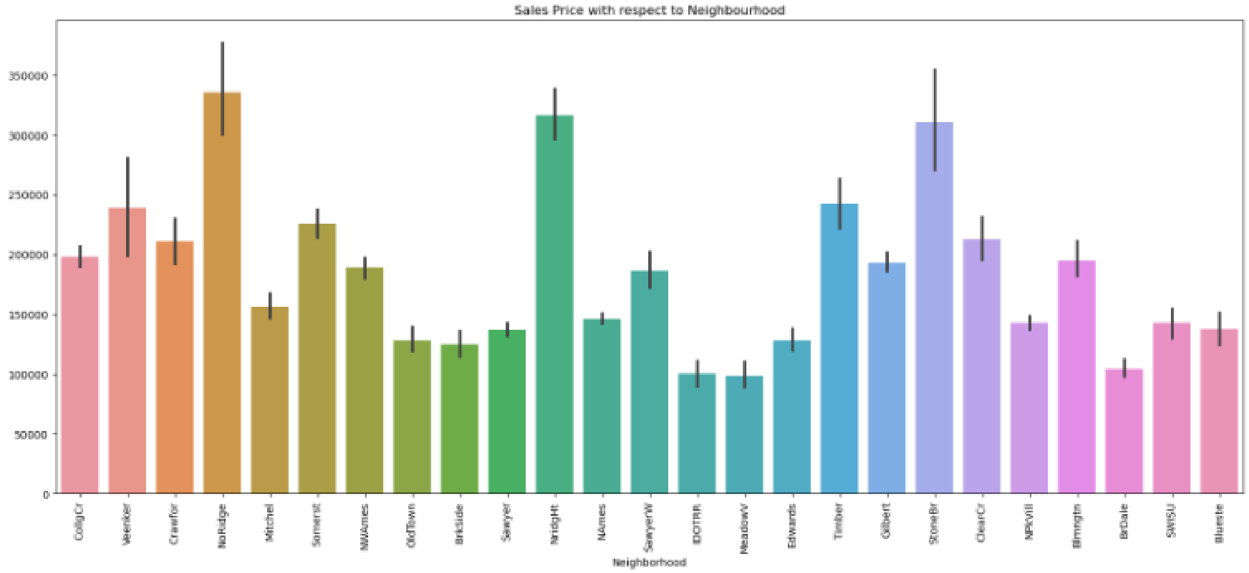


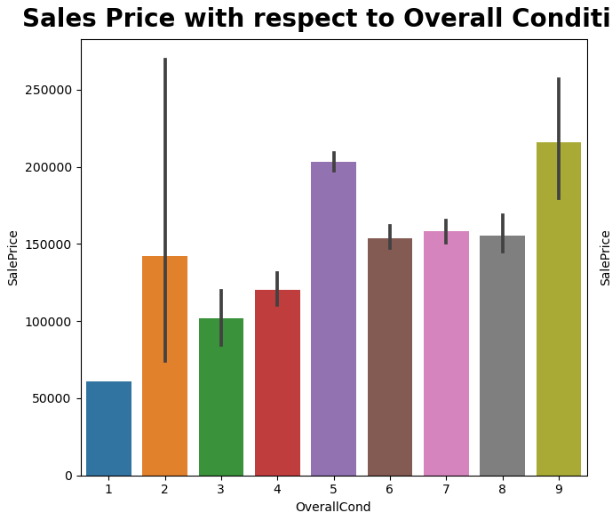
```
sns.distplot(np.log1p(df_train['SalePrice']))
```

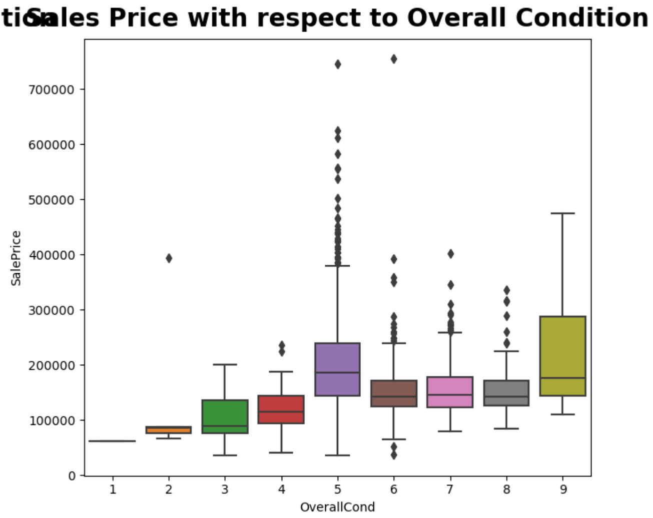
```
Out[17]: <Axes: xlabel='SalePrice', ylabel='Density'>
```

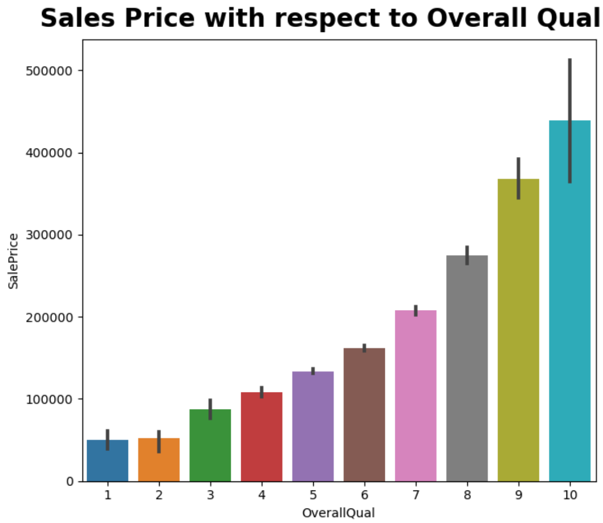
- ## Appendix B - Feature Importance Visualizations

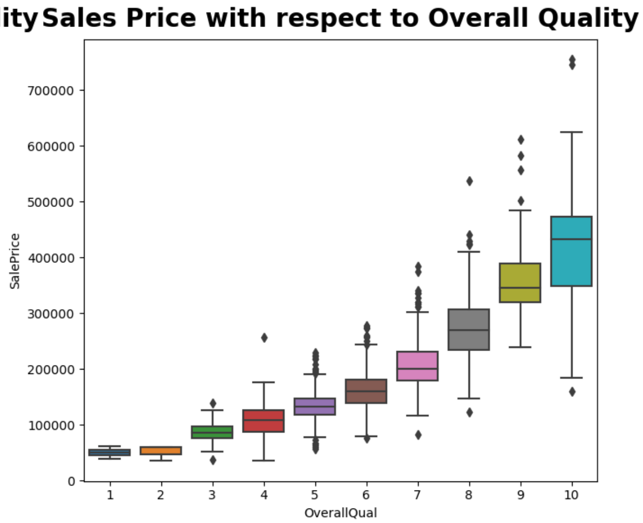
Visualization to show 'Sale Price' with respect to 'Neighborhood'



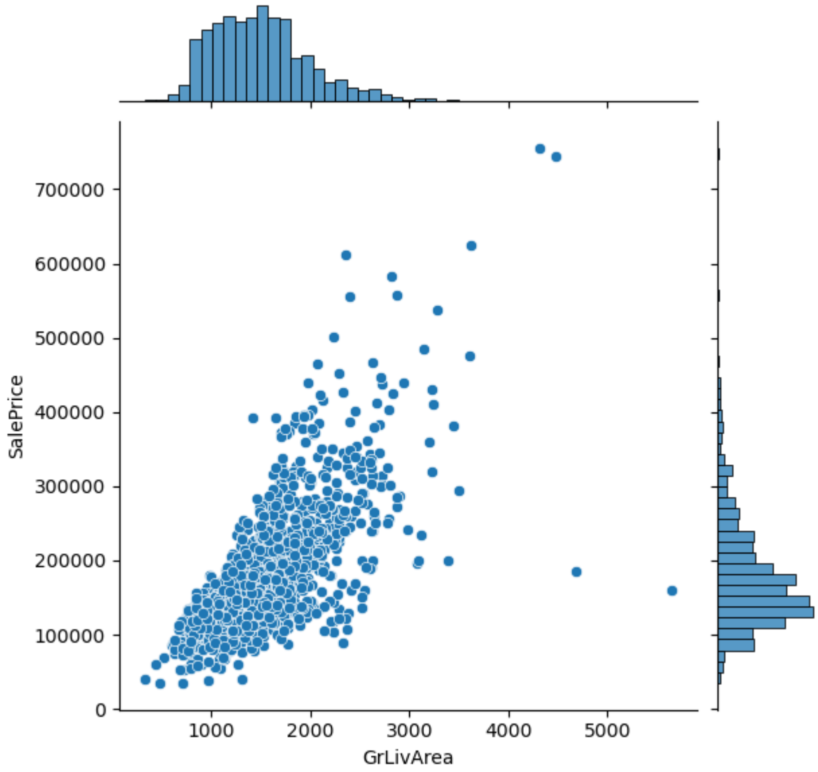

Sales Price with respect to Overall Condition
SalePrice
0
50000
100000
150000
200000
250000
1
2
3
4
5
6
7
8
9
OverallCond


Sales Price with respect to Overall Condition
SalePrice
0
100000
200000
300000
400000
500000
600000
700000
1
2
3
4
5
6
7
8
9
OverallCond

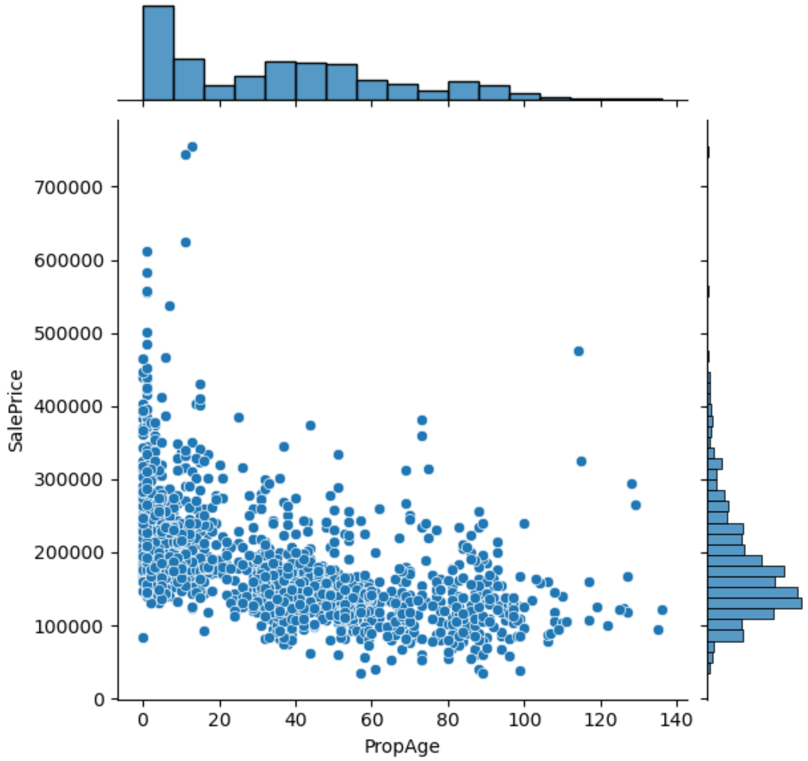

Sales Price with respect to Overall Quality
SalePrice
0
100000
200000
300000
400000
500000
1
2
3
4
5
6
7
8
9
10
OverallQual


Sales Price with respect to Overall Quality
SalePrice
0
100000
200000
300000
400000
500000
600000
700000
1
2
3
4
5
6
7
8
9
10
OverallQual

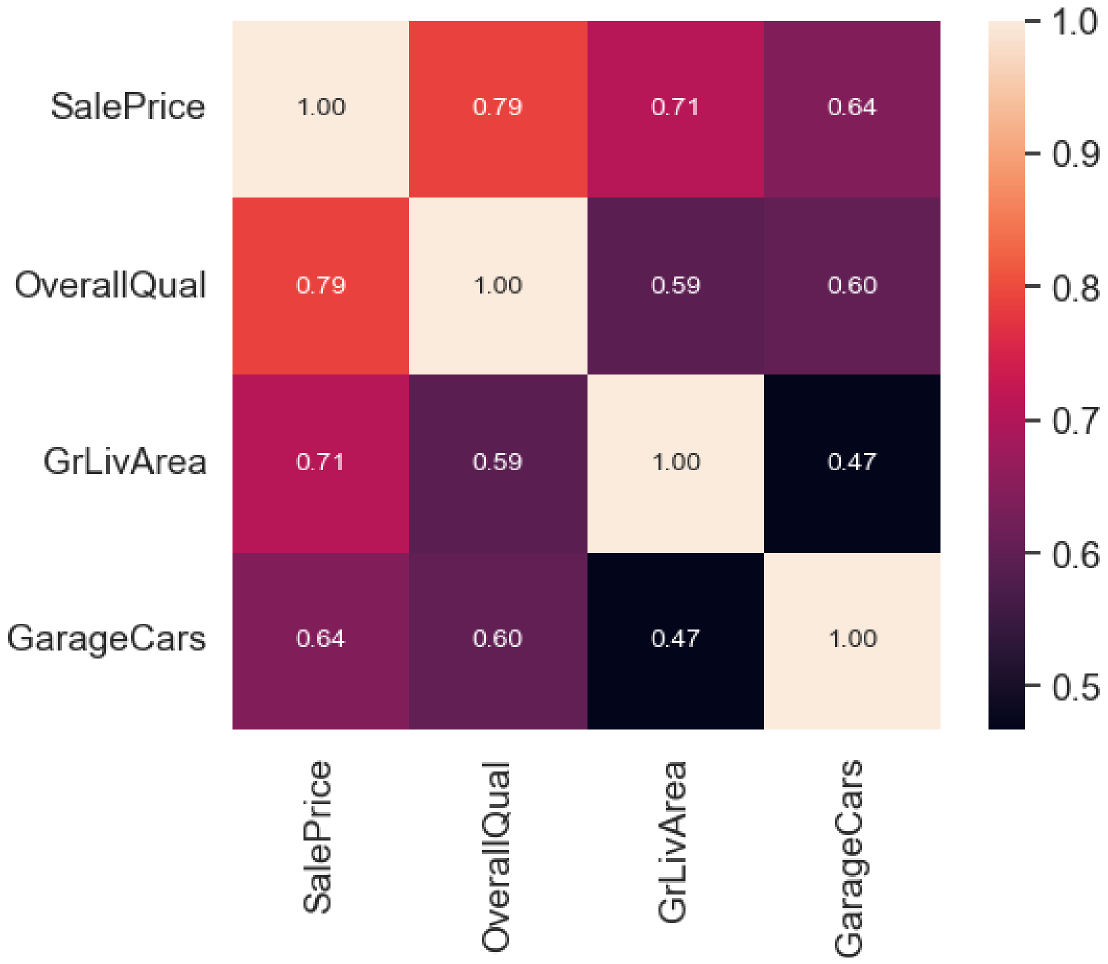
GrLivArea vs SalePrice



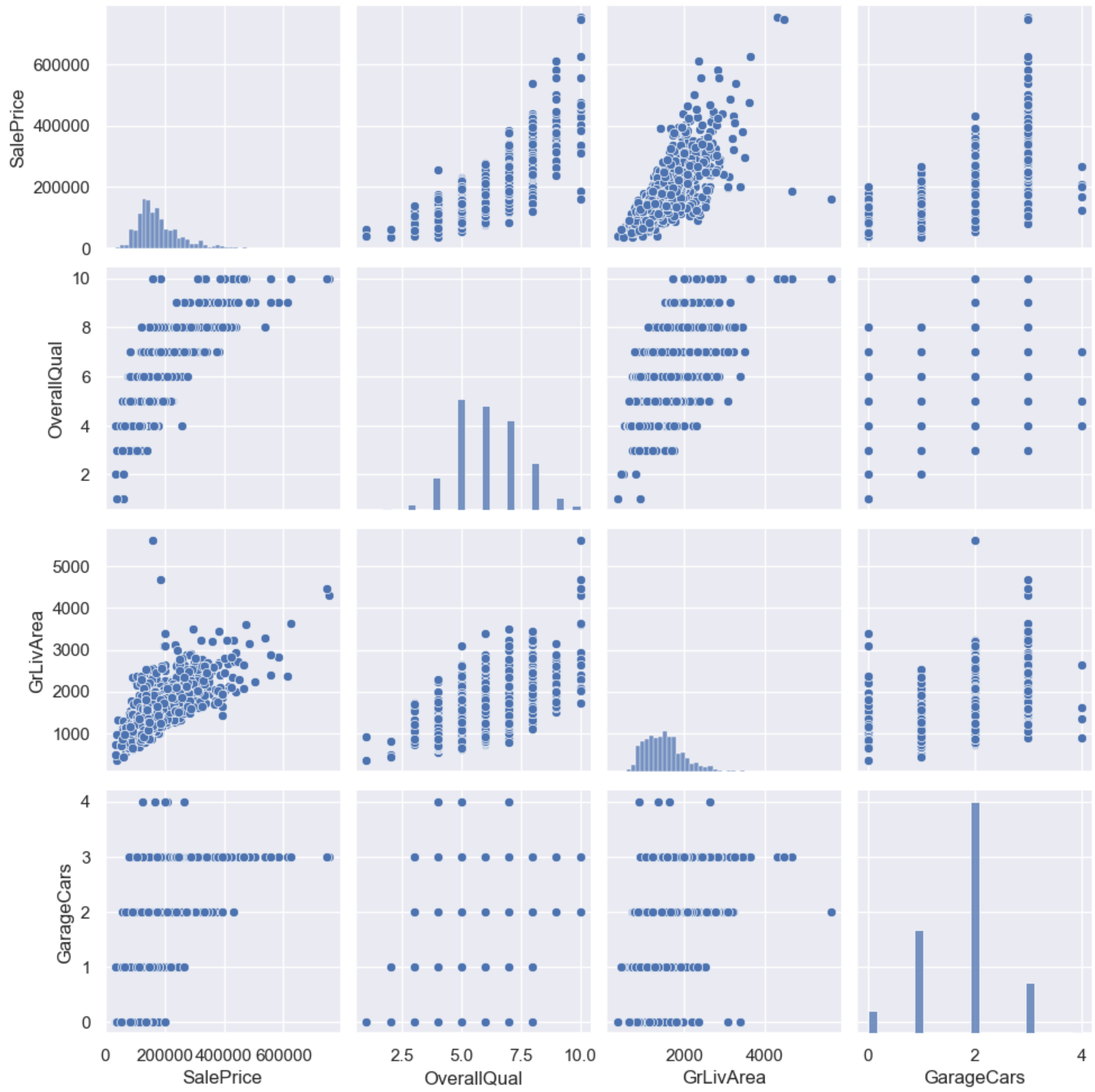
PropAge vs SalePrice

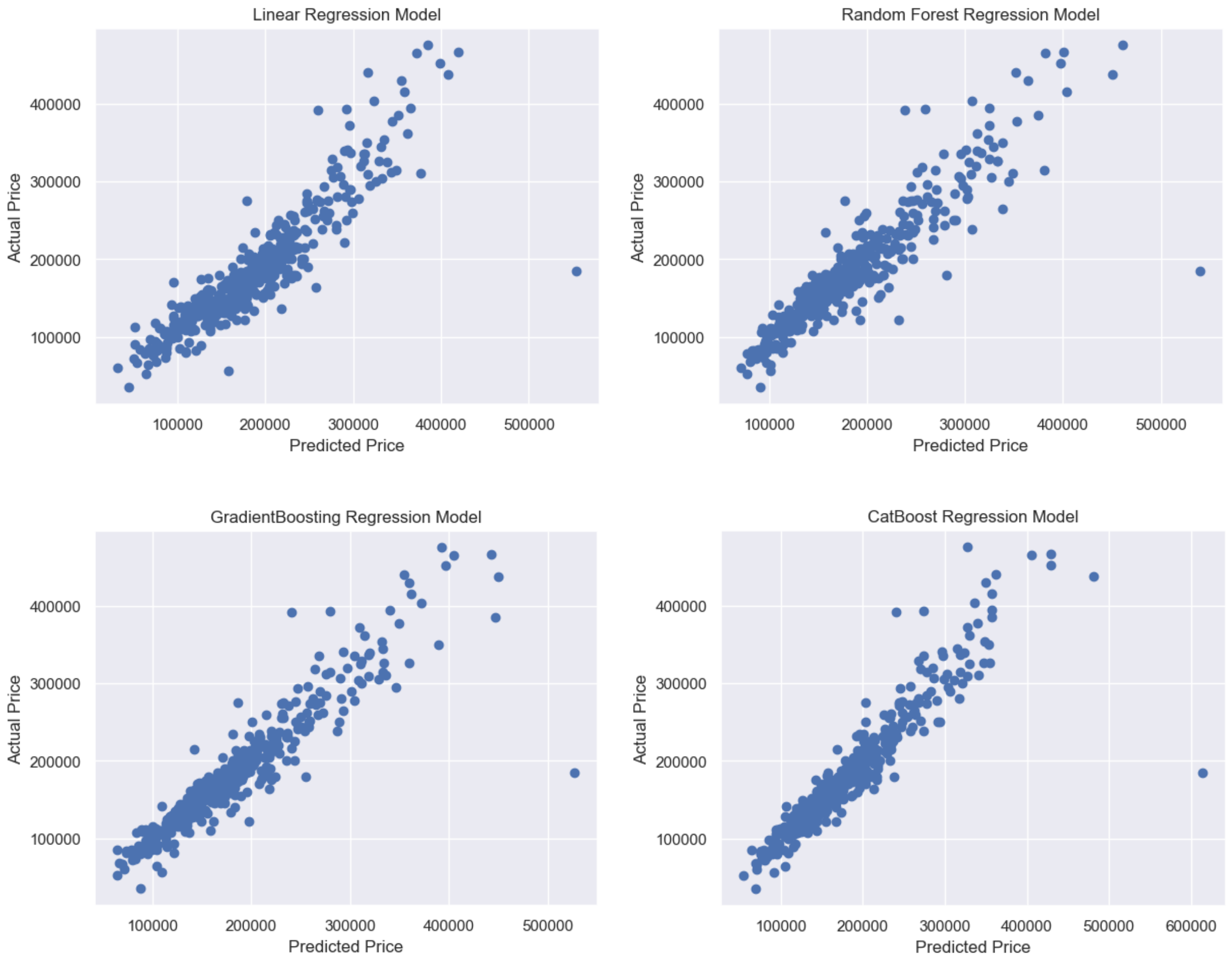


Correlation Matrix

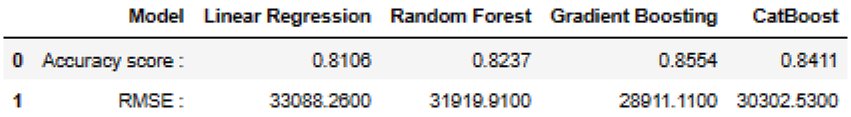


	SalePrice	OverallQual	GrLivArea	GarageCars
SalePrice	1.00	0.79	0.71	0.64
OverallQual	0.79	1.00	0.59	0.60
GrLivArea	0.71	0.59	1.00	0.47
GarageCars	0.64	0.60	0.47	1.00


SalePrice
600000
400000
200000
0
OverallQual
10
8
6
4
2
GrLivArea
5000
4000
3000
2000
1000
GarageCars
4
3
2
1
0
0
200000
400000
600000
SalePrice
2.5
5.0
7.5
10.0
OverallQual
2000
4000
GrLivArea
0
2
4
GarageCars



- ## Appendix C - Model Evaluation Results



	Model	Linear Regression	Random Forest	Gradient Boosting	CatBoost
0	Accuracy score :	0.8106	0.8237	0.8554	0.8411
1	RMSE :	33088.2600	31919.9100	28911.1100	30302.5300