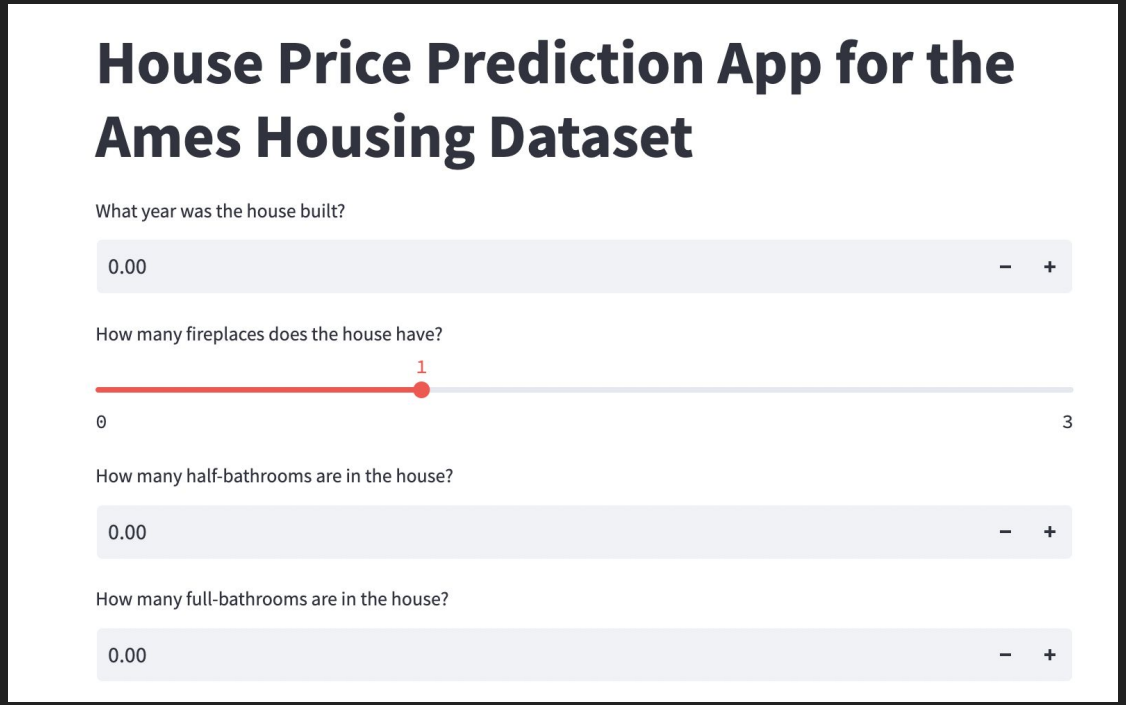
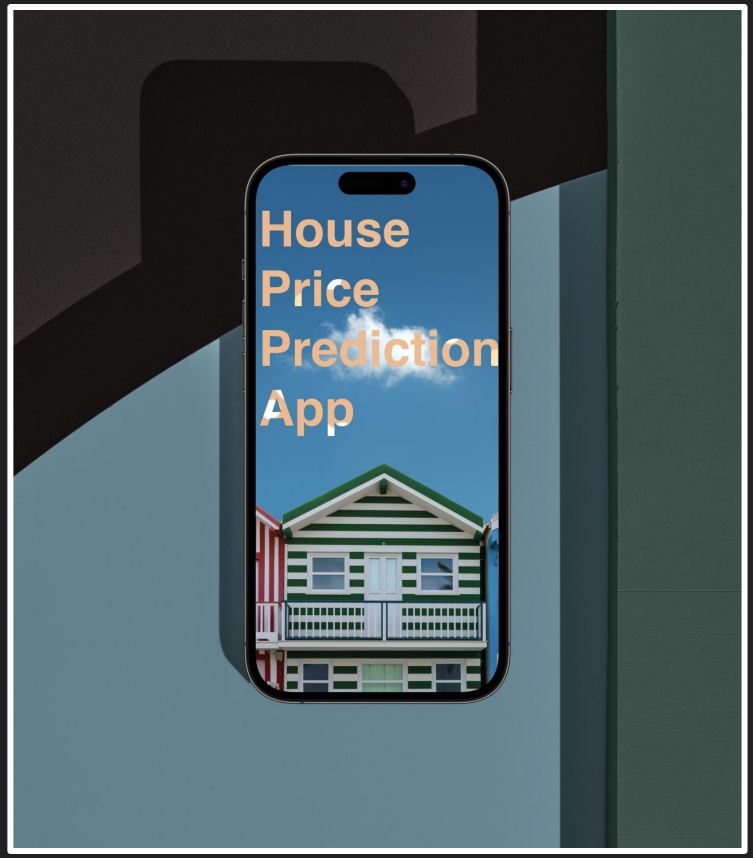



Student: Hamza Al Bustanji, Mentor: Dipanjan Sarkar
Springboard Capstone Project
August 2022 Cohort

# What are we trying to solve?

Given the Ames' Housing dataset, can we develop an accessible app the can predict house prices?



# Who might benefit from our app?

- The general public
- Real-estate agents
- Home sellers and buyers

# What is our data source?

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.

The dataset is from 2016.

The dataset is provided through a Kaggle competition.

The use of this dataset is for illustration purposes only.

The dataset contains 80 features and 1460 rows.

# What is our approach?

We need to reduce the number of features to no more than 20.

We will do this by picking the most important features in terms of predicting the house price.

What determines the importance of a feature?
The model will assign importance to a feature once it’s trained on the data.

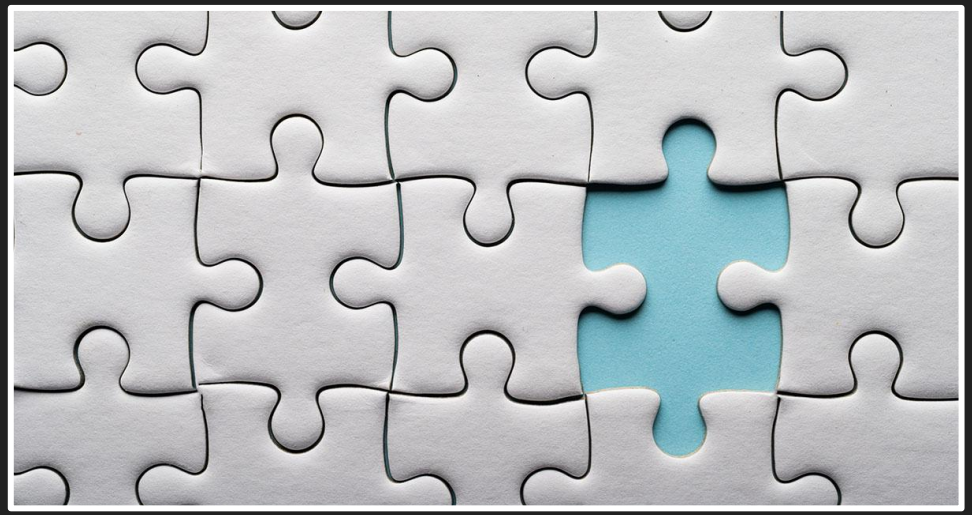
How will we pick our features?
We will look at positive indicators such as correlation with the sale price, and we can use a trial and error process given that we don’t expose our models to the testing data.

# Data Wrangling

We handled missing data in this step. How?

We referenced the data description file and it turned out that a lot of the missing data were just features that aren't included in the houses.
We replaced them with 'none'.

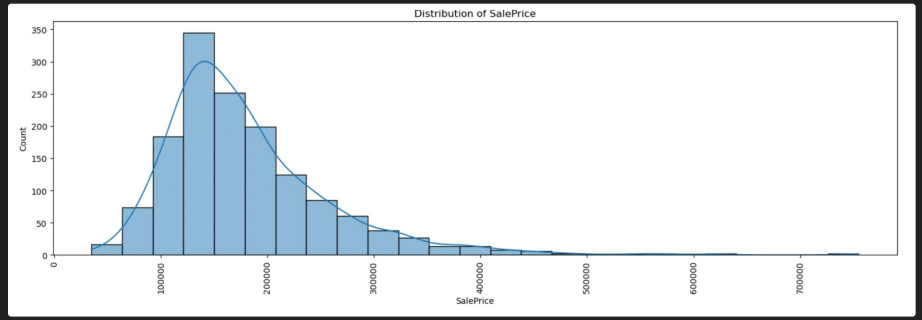
We also inspected our data types, and displayed a brief description of each feature.

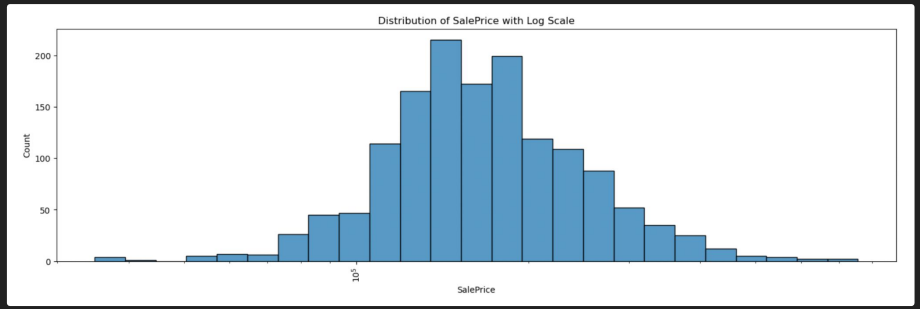


# Exploratory Data Analysis

Our target feature, the sale price:

We inspected the distribution of the Sale price. On a regular scale it seems that the distribution is right skewed, but on logarithmic scale the sale price seems to be normally distributed.



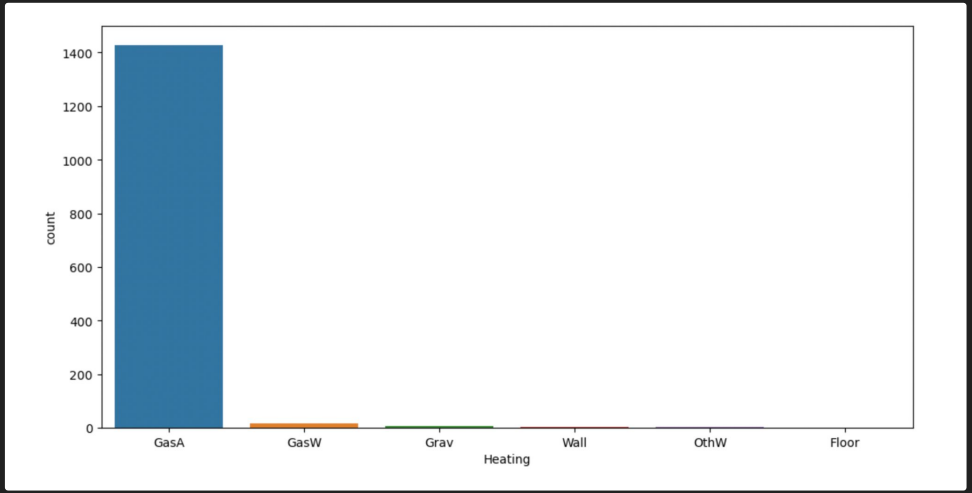



# Exploratory Data Analysis

Categorical features:

For our categorical features, we produced bar plots to inspect them further.

We found some features that were almost completely uniform.
We decided to disregard them from our models, since they provide little information.

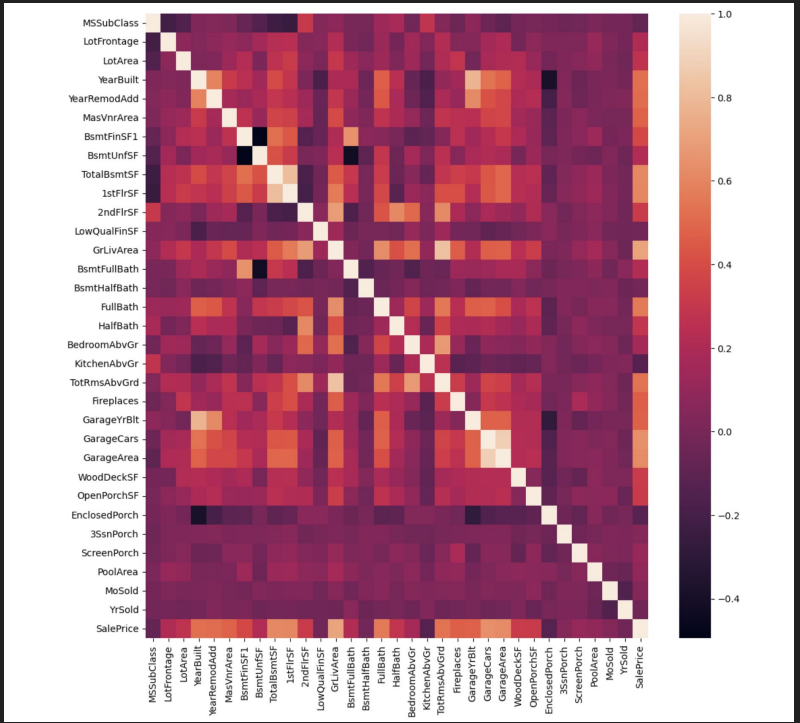


# Exploratory Data Analysis

Numeric Features: Correlation:

For our numeric features we produced summary statistics, and a heat-map that displayed the correlation between the numeric features.

We can see that ‘GrLivArea’,‘GarageCars’, and ‘GarageArea’ correlate strongly with ‘SalePrice’. These are all features that relate to the area or size of the house.



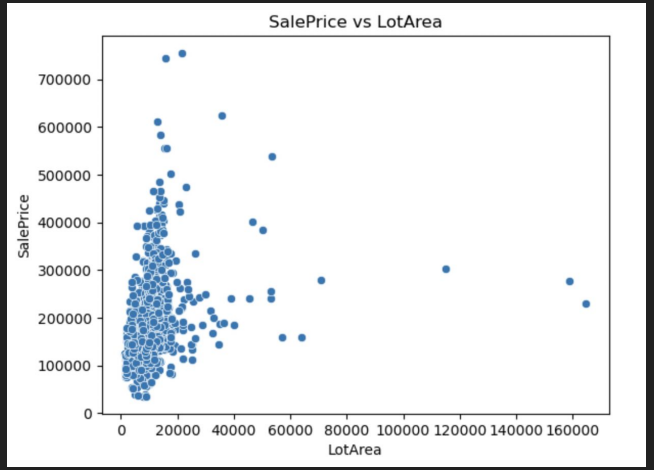
# Exploratory Data Analysis

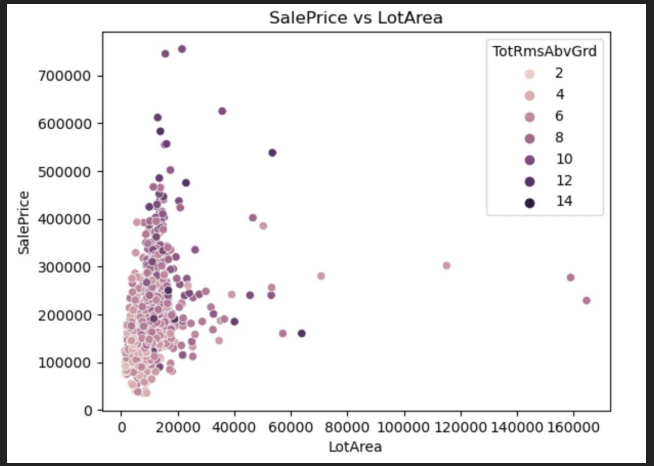
Numeric Features: Lot Area:

The lot area wasn’t highly correlated with the price.

We investigated this and theorized that the reason was that the lot area doesn’t convey information about the actual house.

We tested this by displaying the number of rooms in a house as the color dimension in our plot, and this seemed to support our theory.



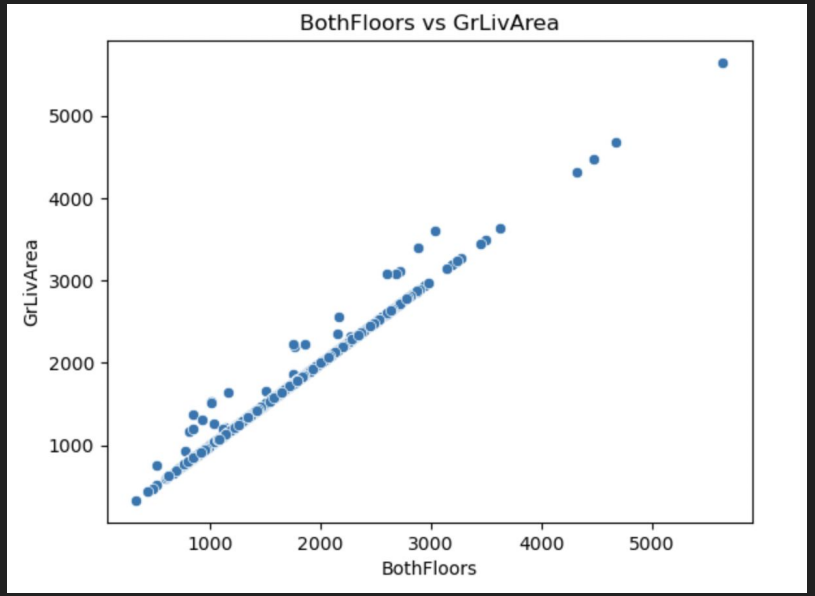
# Exploratory Data Analysis

Numeric Features: House Area:

There are a few area features available.

We examined how they relate to each other. ‘BothFloors’ and ‘GrLivArea’ are almost the same, except, we assumed where the house contains a basement.

We feature engineered a single feature that is the sum of all three. Which we’ll use to avoid redundancy.



# Feature Selection

The features that we selected were:

1. The neighborhood
2. The year the house was built
3. The number of fireplaces
4. The overall condition of the house
5. The house style
6. The building type
7. The number of bathrooms
8. The number of half bathrooms
9. The number of half bathrooms
10. The number of bedrooms
11. The house area
12. The lot area
13. The capacity of the garage in number of cars
14. The area of the masonry work
15. The exterior covering on house
16. The year the house was remodeled

# Preprocessing

The steps we took in preprocessing our data in order are:

- We created dummy variables for our categorical variables
- We created a train/test split, with the test set being 20% of the data
- We scaled our predictive features with a standard scaler

# Modeling

Baseline mode:

We created a baseline model that uses the mean of the house prices to make predictions.

This model will serve as a baseline model for our actual models.

We will see that all of our considered models outperformed this baseline model. Which means they were worthwhile.

# Modeling

Model Comparison:

We trained three types of models, and the results were as shown in the picture.

The best performing model was the Random Forest model. Which we chose to use in our app.

| Model | $R^2$ | MAE | RMSE |
|---|---|---|---|
| Elastic Net | 0.699 | 20971 | 42735 |
| Support Vector Regression | 0.686 | 20621 | 43630 |
| Random Forrest | 0.808 | 20757 | 34105 |

# App

We built the app using the Streamlit framework for python.

Streamlit makes the process of creating an interactive web application straightforward for data scientists.

The code and related files for the app are available in the ‘app’ folder.

The app was deployed and is available for use:

https://hamzabustanji-house-price-prediction-api-appapp-b89nuy.streamlit.app/