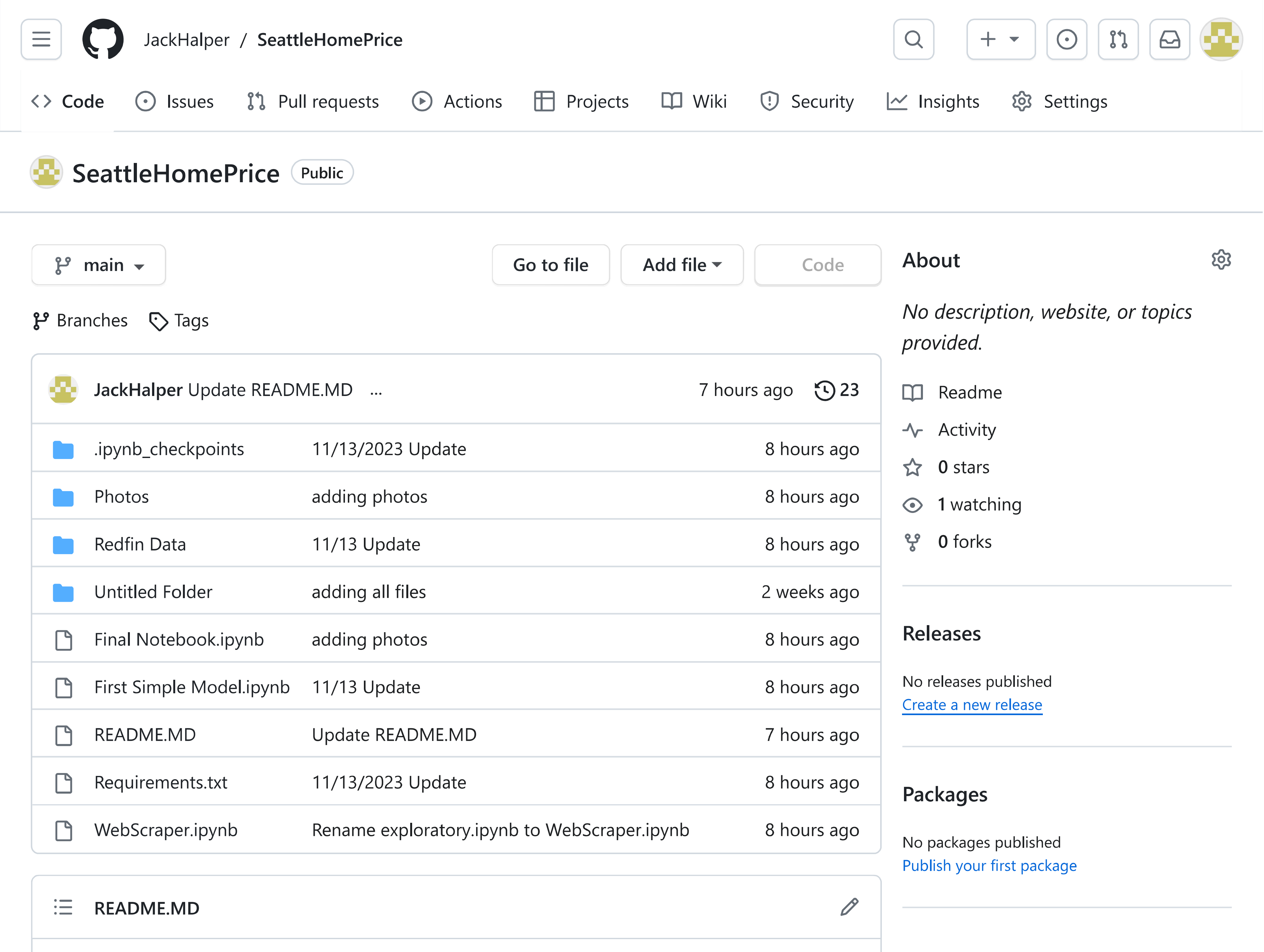


SeattleHomePrice Public

JackHalper Update README.MD … 7 hours ago 23

.ipynb_checkpoints	11/13/2023 Update	8 hours ago
Photos	adding photos	8 hours ago
Redfin Data	11/13 Update	8 hours ago
Untitled Folder	adding all files	2 weeks ago
Final Notebook.ipynb	adding photos	8 hours ago
First Simple Model.ipynb	11/13 Update	8 hours ago
README.MD	Update README.MD	7 hours ago
Requirements.txt	11/13/2023 Update	8 hours ago
WebScraper.ipynb	Rename exploratory.ipynb to WebScraper.ipynb	8 hours ago

README.MD

About

No description, website, or topics provided.

- Readme
- Activity
- **0** stars
- **1** watching
- **0** forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Redfin Real Estate Price Regressor for The City of Seattle

The Objective

The aims of this project are two-fold; the first objective is to formulate as accurate of a housing price model as is feasible for the City of Seattle. The second is to leverage model-interpretability to deduce the features that predict housing values in the city market. Additionally, a secondary goal of this project is to utilize realtor descriptions and natural language processing to augment the performance of the housing model.

The Data

Languages

● **Jupyter Notebook** 100.0%

The Data was manually collected on November 31st, 2023 from the Redfin Website. The redfin website provides an option to manually download CSV files with basic information about the property sale. Data were collected from all zip codes encompassing or partially encompassing the City of Seattle. Properties, not in the City of Seattle, were filtered out later in the analysis. Property information included in the download include: Beds, Baths, Price, Sale Date, Lot Size, Square Footage, Submarket Location, Longitude, Latitude, Sale Type, Year Built, Zipcode and More. The data cover the entirety of the city of Seattle and encompass all homesales registered on the website from the 12 months preceding the collection date. The homesales data is comprised of a variety of product types including; Single Family Detached Homes, Townhomes, Condos/Co-Ops, Multi-Family Properties, Vacant Land, Mobile Homes, Etc. For the purposes of this model and analysis, however, only SFD, TH's and Condos have been included due to other property types' relative infrequency in the Seattle City market. The data directly sourced from Redfin, however, lacked realtor descriptions. Therefore, the data collection process for this analysis included a custom webscraping script to pull descriptions from each properties unique redfin page. While the majority of homes sold in the City are listed on Redfin and include descriptions, some are not, and Redfin sources these description-less sales through MLS (Multiple Listing System). Properties that lacked descriptions were excluded for the purposes of the analysis and model. The data encompasses 6,308 Sales over the past 12 months in the City of Seattle.

Initial Findings

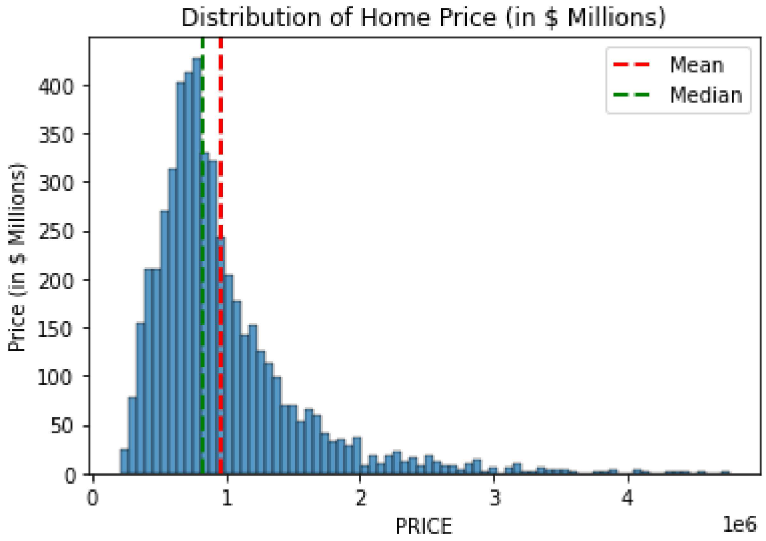
Disclaimer: the following section includes data only on properties qualifying for analysis and excludes some property types and descriptionless listings

######Price:

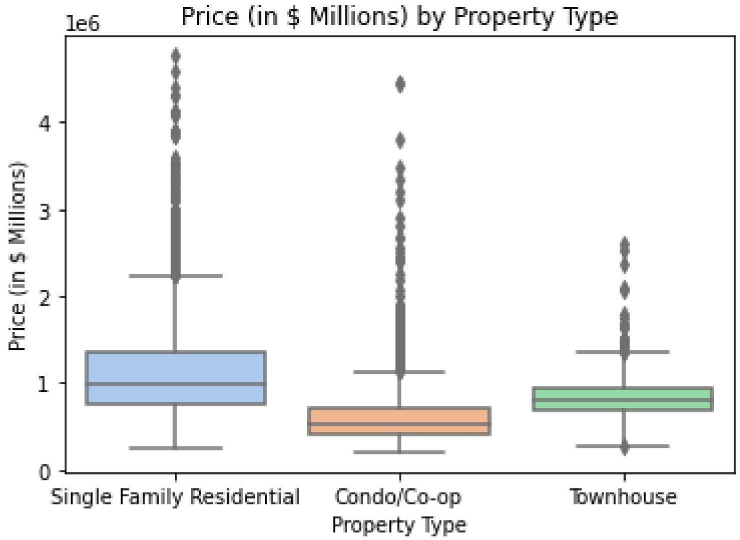
- Prices in the Dataset range from $202,500 - $11,620,000.
- The Mean Home Price in the Set is $958K

- The Median Home Price in the Set is $820K
- The Dataset is skewed to the right

Price Histogram:



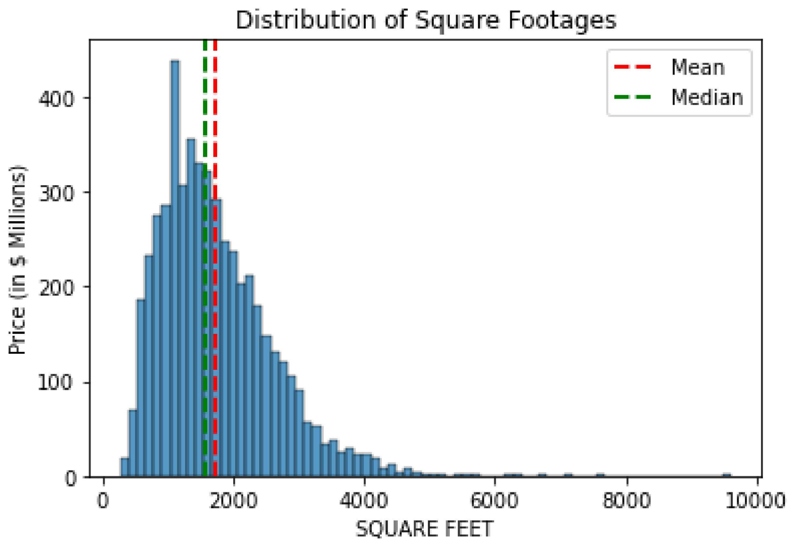
Price Boxplot Distribution by Property Type:



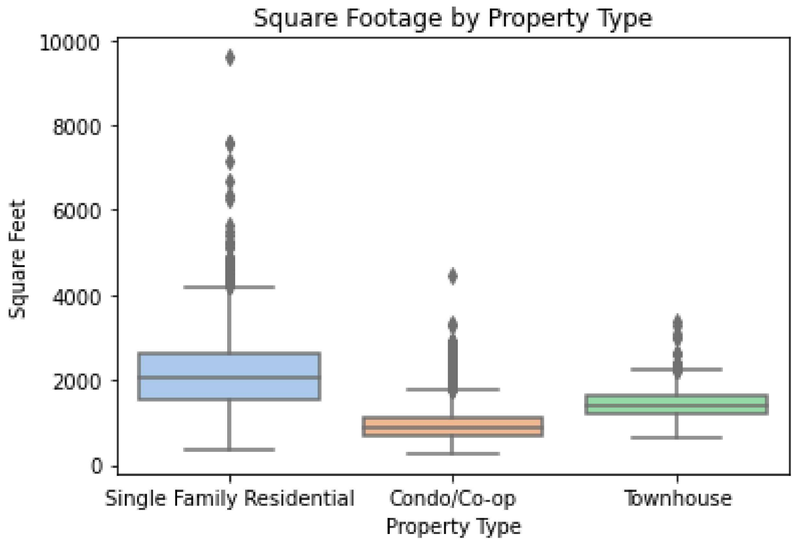
######Square Footage:

- Square Footages in the Dataset range from 268SF - 11,573 SF
- The Mean Home Square Footage in the Set is 1,729 SF
- The Median Home Square Footage in the Set is 1,568 SF
- The Dataset is skewed to the right

Square Footage Histogram:



Square Footage Boxplot Distribution by Property Type:



The Approach

The approach taken, in this project, is basically, to combine traditional housing features and NLP Bag-of-Words into ensemble regressors to acheive the most accurate result possible. The traditional numeric and categorical housing features were fed into a pipeline that either applied a Standard Scaler or One-Hot Encoding to prepare them for the regression models. The Description's were preprocessed, lemmatized and count vectorized into a simple bag of words array and fed into the model.

The Metric

Mean Absolute Error & Mean Absolute Error Percentage

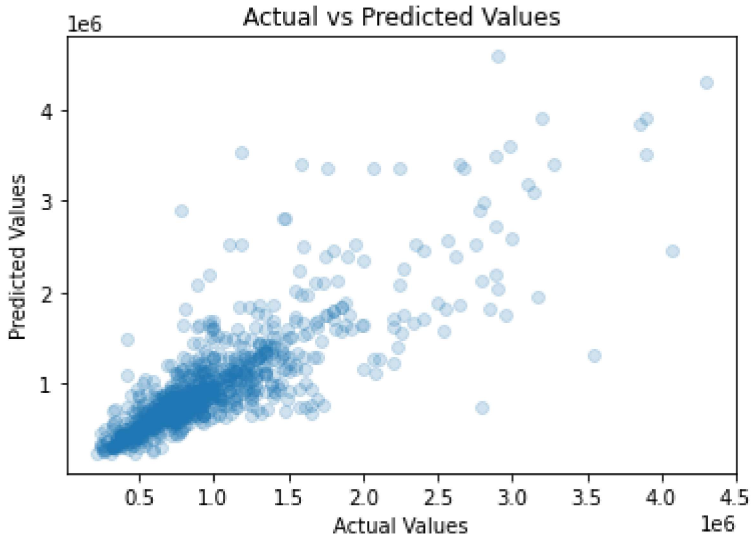
- Why???

The Models

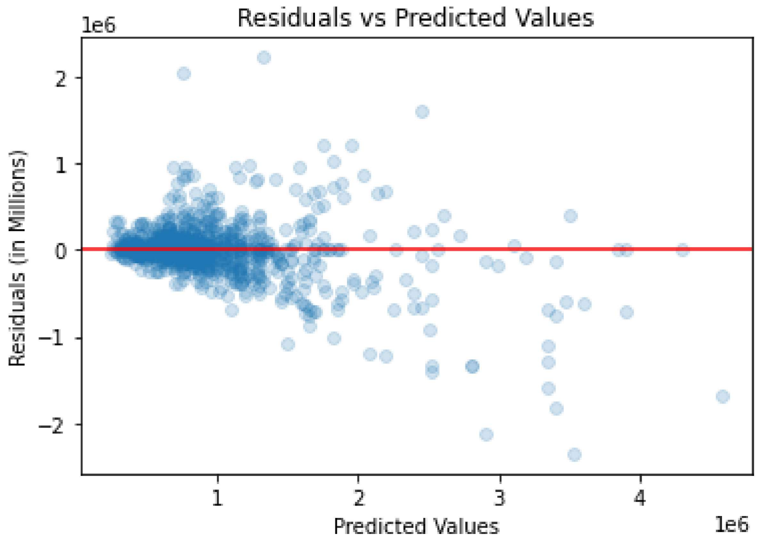
Model #1: Decision Tree (No Hyperparameter Tuning)

- The 1st model was a simple decision tree with no hyperparameter tuning just to provide a baseline model on which to evaluate future models. This simple model acheived an R2 of approximately .65. The Mean Absolute Error acheived was approximately 183k$. The Mean Absolute Percentage Error is approximately 18.5%.

Decision Tree Regressor Predicted vs. Actual Values:



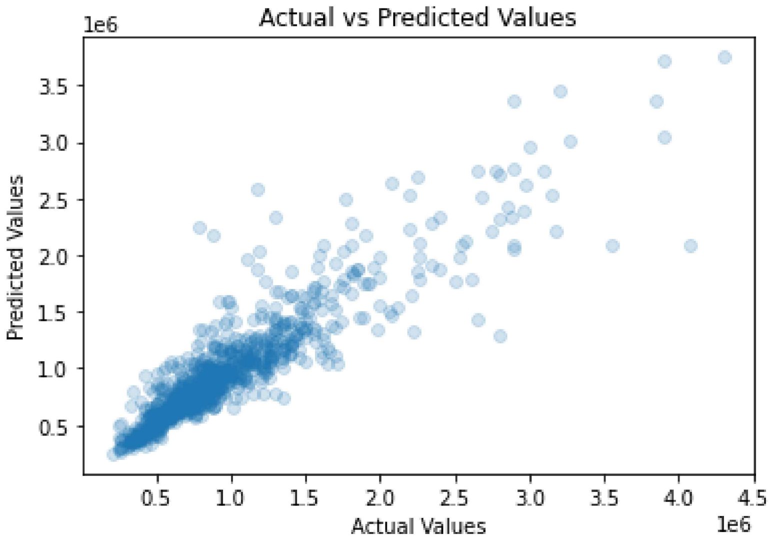
Decision Tree Regressor Residuals vs. Actual Values:



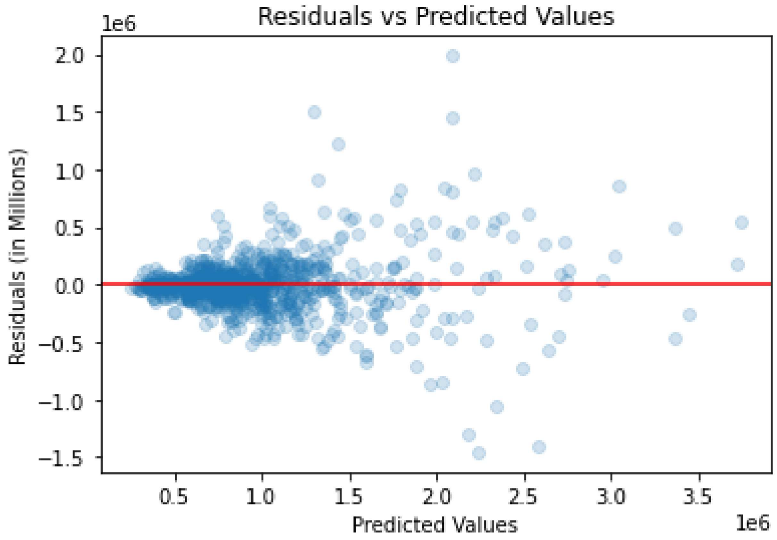
Model #2: Random Forest (Hyperparameters Tuned)

- The 2nd model was produced through an exhaustive cross-validation gridsearch. The Hyperparameters tuned include: Number of Estimators, Max Tree Depth, Minimum Sample Split, Minimum Samples Per Leaf. The best model from the CV search utilized the following final hyperparameters: No Max Depth, Minimum Sample Split of 2, Minimum Sample Per Leaf of 1, and 500 total trees in the ensemble model. This ensemble model acheived an R2 of approximately .82. The Mean Absolute Error acheived was approximately 137k$. The Mean Absolute Percentage Error is approximately 14.35%.

Random Forest Regressor Predicted vs. Actual Values:



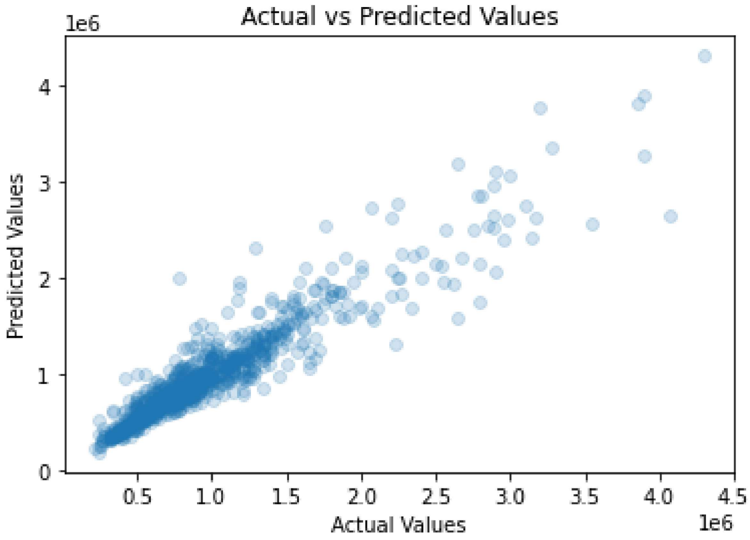
Random Forest Regressor Residuals vs. Actual Values:



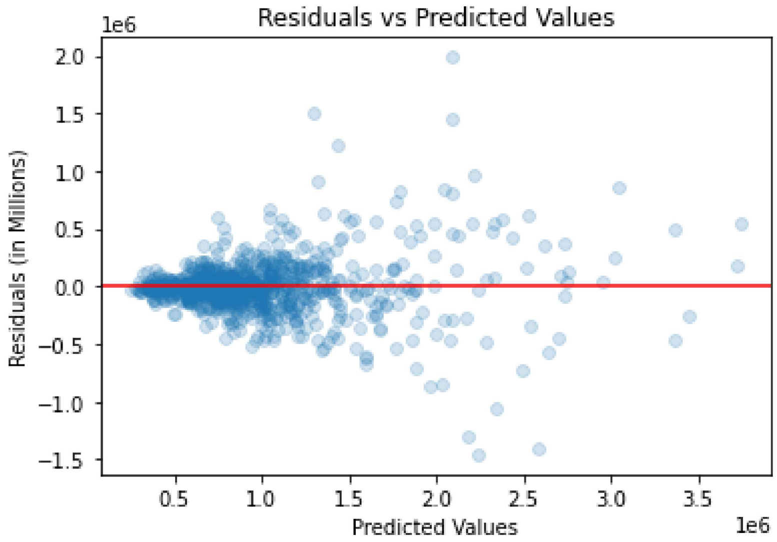
Model #3: XGBoost Regressor (Hyperparameters Tuned) (BEST MODEL)

- The 3rd model was produced through an exhaustive cross-validation gridsearch. The Hyperparameters tuned include: Sample of Features per Tree, Learning Rate, Max Depth, Minimum Child Weight, Number of Estimators, Proportion of Samples per estimator. The best model from the CV search utilized the following final hyperparameters: .8 Proportion of Samples of Features Per Tree, Max Depth of 10, Minimum Child Weight of 4, and 500 total estimators in the ensemble model, .6 Proportion of samples of Features Per Tree. This ensemble model acheived an R2 of approximately .88. The Mean Absolute Error acheived was approximately 111k$. The Mean Absolute Percentage Error for this model is approximately 11.64%.

XBBoost Regressor Predicted vs. Actual Values:



XBBoost Regressor Residuals vs. Actual Values:


Residuals vs Predicted Values
1e6
2.0
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
Residuals (in Millions)
0.5
1.0
1.5
2.0
2.5
3.0
3.5
Predicted Values
1e6