Group 10 CAP 4770 Final Project Report

Project Title

Home Value Prediction

Team Members

Joshua Bowman
Aidan Crawford
Connor Curcio
Matthew Nobleza

Introduction / Executive Summary

Problem Statement Synopsis

How can we accurately estimate the values of homes using basic, readily available information about them? Moreover, how could we explore the data to be able to better understand the data provided, and remove any unnecessary noise from our estimation techniques?

We created three models that estimate home values on the market using the Zillow property and transaction datasets. Specifically, we will use a linear regression model as our baseline, a random tree model, a ridge regressor model, and a MLP regressor (neural network) model. We will then compare the performance of each of these models. Along the way, the exploration of the property features and transaction datasets will be able to further assess our strategy and learn more about the data in hopes to fine tune the models. Upon completion, our project will allow home buyers and sellers to accurately predict home prices based on house attributes.

Data and Benchmark Description

The Zillow datasets include property features and transaction data from three California counties: Los Angeles, Orange, and Ventura. These datasets span specific periods, with training and testing data from 2016 and 2017.

Algorithms

In addition to the baseline model, we developed and used prediction models to solve our problem statement and estimate housing prices.

We utilized the following models:

- Linear Regression
- Random Forest
- Ridge Regression
- Neural Network

Improvements After Last Presentation

After receiving feedback, we improved our approach to handling missing data, explored more sophisticated imputation methods, and enhanced our result visualizations to better explain key concepts like log error.

Problem Statement Details, Data and Benchmark Description

Problem Statement

How can we accurately estimate the values of homes using basic, readily available information about them? Moreover, how could we explore the data to be able to better understand the data provided, and remove any unnecessary noise from our estimation techniques?

Dataset

The Zillow property datasets are used for training and testing purposes. The data is split into transactions and property features files, covering properties in Los Angeles, Orange, and Ventura counties for 2016 and 2017. The datasets can be found here: https://www.kaggle.com/competitions/zillow-prize-1/data.

Approaches and Algorithms

Algorithms

We developed and used the following prediction models to estimate housing prices:

- Linear Regression (baseline)
- Decision Tree
- Random Forest
- Ridge

Data Preparation

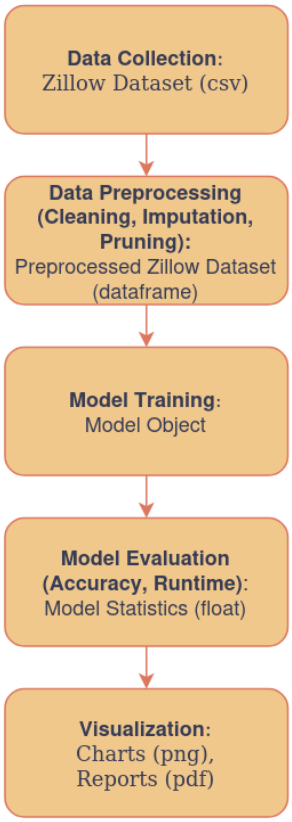
- Mean Imputation: Initially used to handle missing data by replacing missing values with the mean of each feature.
- Pruning: Dropping features with less than 80% data coverage to improve data integrity.

Programming Tools

- Python Libraries: Pandas, Matplotlib, Seaborn, Sklearn, PyTorch.
- Version Control: GitHub for source code management.
- Development Environment: Jupyter Notebook for code implementation and testing.

Source Code Structure

Our source code is primarily compartmentalized into model folders. The preprocessor and cleaner python libraries exist in the root source directory, while model library files and visualization Jupyter notebooks exist in each model directory.



Metrics and Evaluation Setup

The evaluation metric will be logerror using the following:

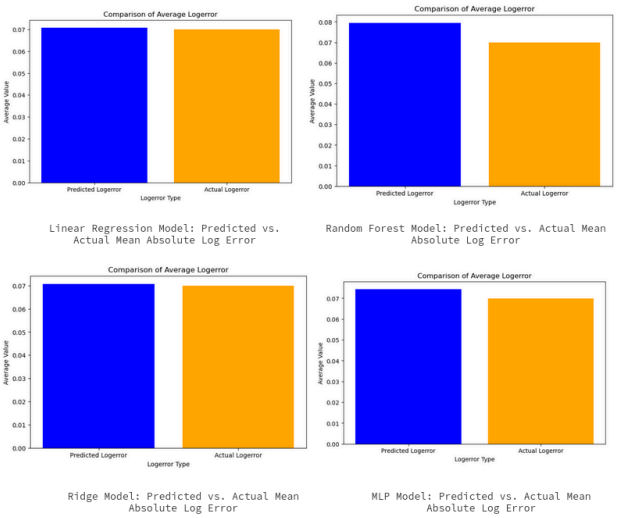
$$Mean\ Absolute\ logError\ =\ average(|estimatedLogError\ -\ actualLogError|)$$

A lower mean absolute LogError would be better. The test data performance estimates that will be compared are the transactions at a certain period not included in the training data.
For our baseline model, we created a basic linear regression model to show how effective this simple solution would be when estimating the log error. The model takes in attributes about houses present in the dataset and outputs an estimated LogError. Other models we developed are compared to this baseline model.

Results and Comparison of Approaches

Accuracy

- Linear Regression: The baseline predicted a log error with 1.38% difference from the actual log error.
- Random Forest Regression: Had a 12.71% difference in log error.
- Ridge: 1.92% difference
- MLP Regressor: 7.65% difference



Linear Regression Model: Predicted vs. Actual Mean Absolute Log Error

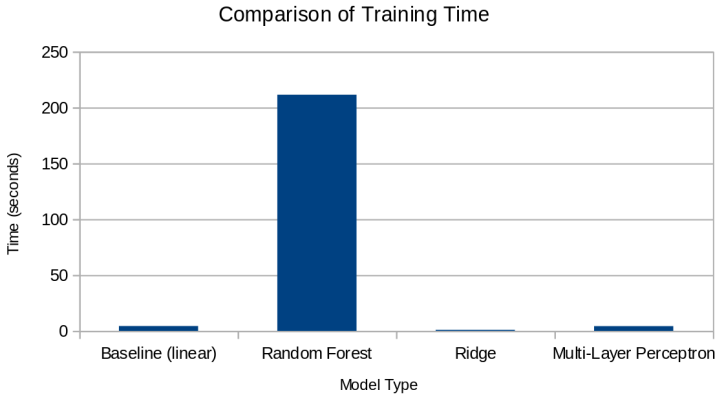
Random Forest Model: Predicted vs. Actual Mean Absolute Log Error

Ridge Model: Predicted vs. Actual Mean Absolute Log Error

MLP Model: Predicted vs. Actual Mean Absolute Log Error

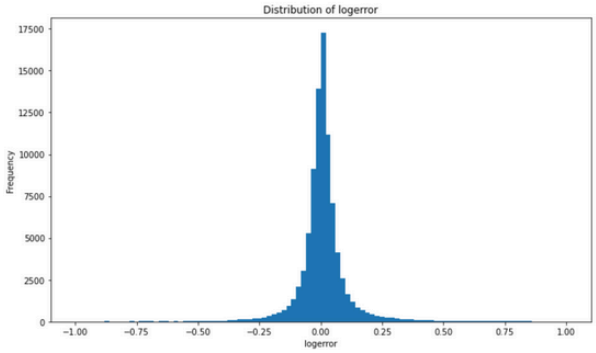
Scalability and Runtime

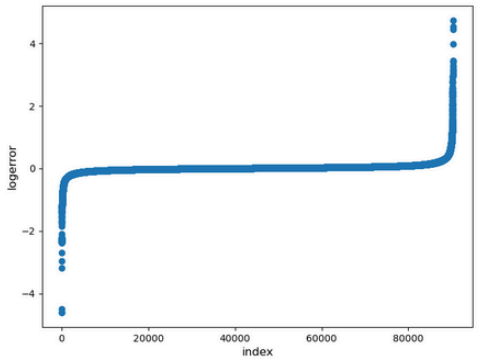
- The models all have similar runtimes for training except for one.
- Random Forest is an outlier with the greatest runtime.



Log Error Distribution

Visualizations of log error across different features highlighted the importance of understanding data transformations and their impact on model performance.





Conclusion

Challenges

- Handling Missing Data: Initial mean imputation introduced artifacts. Exploring more sophisticated methods helped improve data integrity.
- Model Interpretability: Balancing accuracy with interpretability, especially for complex models like Random Forest.
- Data Sparsity: Ensuring robust predictions despite missing values and sparse data coverage.

Lessons Learned

- Data Quality: High-quality data is crucial for reliable predictions and dropping features with less than 80% coverage improved the model performance.
- Metrics Explanation: Clearly explaining log error and its implications helped in better understanding and interpreting the results.
- Continuous Improvement: Iterative improvements based on feedback led to more robust and accurate models.