Pennsylvania State University

Predicting House Prices in Philadelphia
Project1: Individual Project - Report

Xingyu Jiang
CMPSC 445 Machine Learning Data Science
Ph.D. Janghoon Yang
6 March 2024

Introduction

During the course of this project a model to predict the selling price of houses in Philadelphia was developed. There are several stages involved during the model development of data collection, preprocessing, feature engineering, model development, and evaluation. The main goal of this project is to ensure the accuracy of the model while enriching myself with the knowledge on how to handle issues and mistakes that can happen during the development process.

Data Collection

The dataset utilized in this project was sourced from the official website of the City of Philadelphia (www.phila.gov). Which is an extremely reputable and trustworthy entity on the subject matter. In this dataset they provided to the public, there is extensive details and high volume of data available, making perfect fit for the objectives of the project.

Scope of Data:

The dataset includes a diverse range of characteristics for each property, including but not limited to:

- Location Information:
  - Latitude and Longitude
- Size and Dimension:
  - Total Area
  - Total Livable Area

    - Number of Bedrooms
    - Number of Bathrooms
- Condition and Quality:
    - Exterior Condition
    - Interior Condition
    - Quality Grade
- Age and Year Built:
    - Year Built
- Amenities and Features:
    - Central Air
    - Fireplaces
    - Garage Spaces
- Market Value and Sale Information:
    - Market Value
    - Sale Price
    - Sale Date

Data Volume:

The dataset obtained from www.phila.gov comprises a substantial volume of data, exceeding 500,000 samples.

Data Preprocessing

An extensive preprocessing of the data was conducted to ensure it is ready to be analyzed. There were a considerable amount of categories of data that were deemed unnecessary and should be removed from the data. This includes those that contain too many missing values, those that don't have a clear meaning to, some personal information of the property owner, and some that were making it hard to process the data.

- Handling of missing values
    - For categories that are measured in dimension and can be filled in with median values. It will be performed.
    - For classification categories, it is very hard to find a value to fill those places. To improve the accuracy and make it easier, these rows were removed.
- Converting categorical variables
    - For the conversion of the categorical variables, LabelEncoder provided by sklearn were used.

Feature Engineering

To enhance the predictive capabilities of the model. I tried to utilize a pipeline consisting of polynomial feature expansion, standard scaling, and principal component analysis (PCA) for dimensionality reduction.

1. **Polynomial Feature Expansion**: Polynomial features were generated from the original dataset. This technique creates interactions between features, capturing nonlinear relationships that may exist within the data. I chose a degree of 2 for polynomial expansion, allowing for quadratic relationships between variables.
2. **Standard Scaling:** Standard scaling was applied to ensure that all features have a mean of 0 and a standard deviation of 1. This process prevents features with larger scales from dominating the model training process.
3. **Principal Component Analysis (PCA):** PCA was employed to reduce the dimensionality of the feature space while retaining 95% of the variance. By projecting the data onto a lower-dimensional subspace, I was expecting PCA to help mitigate issues related to multicollinearity and overfitting, thus improving model generalization.

Over all these three steps were applied with the goal of extracting relevant information from the raw data and transforming it into a more suitable format for predictive modeling.

Model Development

In this phase, I tried a few algorithms; including linear regression, random forests, and gradient boosting. Some of the key steps I used in the model development are as follow:

- **Data preparation:** Since the size of the data is quite large and would take a long time to process. A simple program was implemented to randomly select preprocessed data from the csv file. So it will be quicker for the algorithm to develop a model.
- **Hyperparameter Tuning:** Except for linear regression, this is performed on both random forests, and gradient boosting. Grid search with cross-validation was employed to find the optimal hyperparameters for the selected algorithms. By systematically searching through a predefined hyperparameter grid, I tried to maximize model performance while avoiding overfitting.
- **Model Evaluation:** The performance of each model was evaluated using common regression metrics such as mean absolute error (MAE), mean squared error (MSE), and R-squared (R^2) score. These metrics provide insights into the predictive accuracy and generalization capabilities of the models.

By iteratively refining the feature engineering process and experimenting with different algorithms, I tried to decrease the mean absolute error and mean square error, while keeping the R^2 above 80%.

Construction of Test Dataset and Model Evaluation

To evaluate the performance of the trained model, I split the data into two parts: one for training and one for testing. As the name suggested the training dataset was used to train the model, while the testing dataset was reserved for evaluating its predictive accuracy.

Discussion

Result

The Results I got from all three different machine learning algorithms I used, the results were quite similar. The mean absolute error, mean square error and the value for R squared were always quite high.

For a sample size of 5,000, the average R square value was 91%, which as expected were directly proportional to the sample size. As it increases R square will increase as well, vice versa for decreasing sample size.

The mean absolute error and mean square error were always extremely high, no matter what I chose to do. Mean absolute error always ranges from 21,000 to 16,000. Mean squared error ranges in the billions range. Which I think is acceptable since it is regarding the prices of the houses.

Challenges encountered

Most of the issues I encountered during the project were related to the data collection and data handling.

In the beginning, I wasn't expecting to take around four days just finding a proper dataset for this project. I thought there would be a ton of data that is open to the public and I can just choose one of them. However most of them weren't exactly the types I wanted to use. Majority of them were just on the fluctuation in the house prices.

Another issue that gave me a ton of issues was trying to figure out what is the value that is best fit for the missing values. A ton of times they can't be filled with median or mean values since they are categorical. I end up having to drop all of them to not give too much bias to the data.

Strengths and limitations

I believe the main strengths of my model were the size of the data. It can cover many cars and predict a value that is quite trustworthy. However there still a ton of work can be done, to reduce the mean absolute error and mean square error. The current version is not accurate enough to perfectly predict the market value as I envisioned at the beginning.