House Price Prediction using Lasso Regression

Ishan Bhargava (5)

Abhiram Raju (30)

Jeevan Ravi Kumar (34)

Abstract

Housing price prediction is important for a number of reasons. For one, it can assist in real estate investments, allowing individuals to make informed decisions about buying, selling, or holding properties. Additionally, accurate estimates of market value can help those looking to buy or sell a property to make more informed pricing and negotiating decisions. Beyond individual investments and purchases, housing prices can also provide valuable insights into the health of the economy, consumer spending patterns, and demographic trends. Accurate predictions can help governments and businesses make strategic decisions based on these trends. Furthermore, accurate housing price predictions can assist city planners and policymakers understand the housing market and make informed decisions about zoning, infrastructure, and other issues affecting urban development.

Introduction

Predicting housing prices is an important problem in the real estate industry, as it helps homeowners, buyers, and sellers make informed decisions based on the current market trends and the property's value. Accurate prediction of housing prices requires the analysis of various factors such as the location, size, condition, and features of the property, as well as the current market conditions and economic trends. Machine learning and statistical modeling techniques have

proven to be highly effective in predicting housing prices, by analyzing large and complex datasets and identifying the key factors that influence property value.

Accurate prediction of housing prices can provide numerous benefits, such as enabling homeowners to make informed decisions regarding their property, helping buyers and sellers to negotiate a fair price, and facilitating the planning and development of real estate projects.

Python packages used:

* Pandas
* Numpy
* Math
* Matplotlib
* Seaborn
* Scipy

LASSO Regression

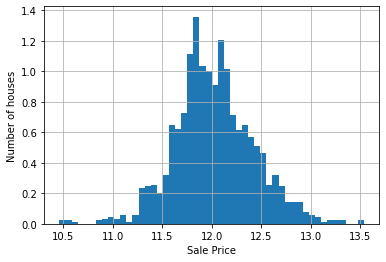
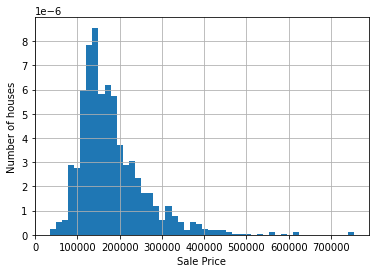
Lasso regression is a type of linear regression that uses regularization to prevent overfitting and improve the accuracy of the model. It is particularly useful when working with datasets that have many features or variables, as it can help identify which features are most important for predicting the outcome of interest. Lasso regression achieves this by adding a penalty term to the ordinary least squares regression equation, which shrinks the coefficient estimates towards zero. This penalty term is based on the absolute value of the coefficients, and as a result, it has the effect of "lassoing" or shrinking the less important coefficients to zero. This can help to simplify the model and improve its interpretability, while still maintaining good predictive accuracy. Lasso regression is widely used in machine learning and data science, particularly when working with high-dimensional datasets or when feature selection is an important consideration.

Data Analysis

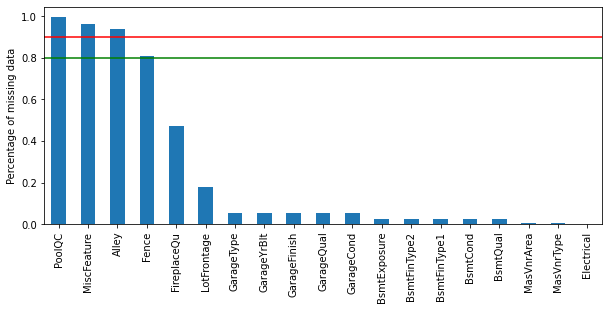
When conducting data analysis, several factors must be taken into consideration in order to ensure accurate results. One such factor is the analysis of the target variable, which is the variable that we are trying to predict or explain. Additionally, it is important to consider the types of variables in the dataset, including both categorical and numerical variables. When analyzing numerical variables, it is important to consider whether they are discrete or continuous, as well as their distributions and any necessary transformations. Categorical variables, on the other hand, require consideration of cardinality, rare labels, and special mappings. It is also important to examine the dataset for any missing data, as this can significantly impact the analysis. By considering these factors and utilizing additional resources as needed, data analysts can ensure that their analyses are both accurate and effective.

Logarithmic transformation

Before After



Missing Data distribution across features

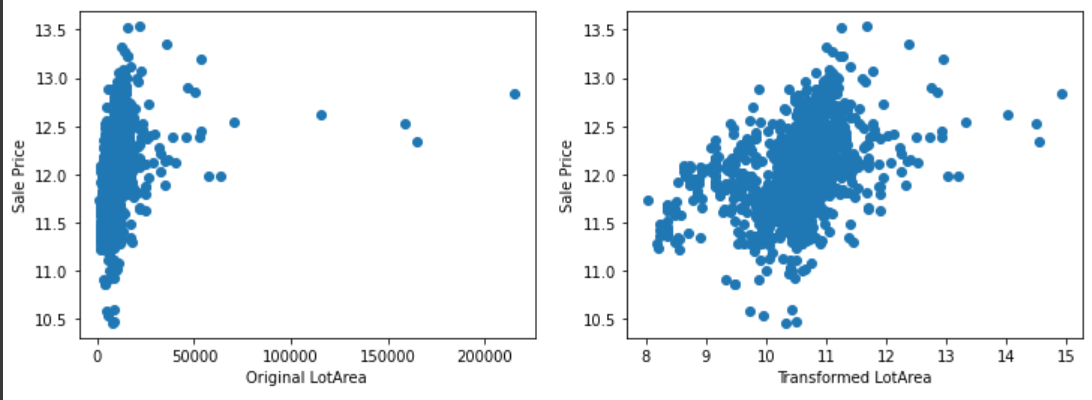


Feature Engineering

Feature engineering is the process of selecting and transforming raw data features into meaningful and informative input features for machine learning algorithms. In other words, it involves selecting relevant features, extracting new features, and transforming them into a format that can be easily consumed by a machine learning algorithm. Feature engineering is an important aspect of machine learning because it can have a significant impact on the accuracy and efficiency of a model. The goal is to create a set of features that captures the underlying patterns and relationships in the data, and helps the machine learning algorithm to make better predictions. Feature engineering can involve a range of techniques, including data cleaning, feature selection, feature scaling, dimensionality reduction, and feature construction. It is a crucial step in the machine learning pipeline and requires domain knowledge, creativity, and analytical skills to perform effectively.

Yeo-Johnson Transformation

The Yeo-Johnson transformation is a method for transforming data that can be useful in certain situations, particularly when working with data that is not normally distributed. It is similar to the Box-Cox transformation, but can handle data that includes negative values. The Yeo-Johnson transformation works by applying a power transformation to the data, with a parameter lambda that is chosen to optimize the normality of the transformed data. Unlike the Box-Cox transformation, the Yeo-Johnson transformation can handle both positive and negative data values, and is therefore more versatile. Additionally, the Yeo-Johnson transformation can be particularly useful when working with skewed or heavy-tailed data, as it can help to normalize the data and make it more suitable for analysis. Overall, the Yeo-Johnson transformation is a valuable tool for data analysts and researchers who need to work with non-normal data, and can help to improve the accuracy and validity of their analyses.



Developing the pipeline

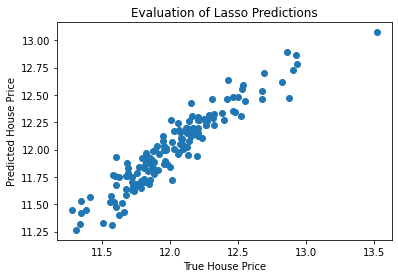
This is the end to end pipeline. The pipeline covers everything from data analysis, data engineering to creating the Lasso Regression Model.

Conclusion

Based on the metrics, the Lasso regression model appears to have performed fairly well. The train and test mean squared errors (MSE) are 781396630 and 1060769014 respectively, which suggests that the model has some level of predictive accuracy. Additionally, the train and test root mean squared errors (RMSE) are 27953 and 32569 respectively, which indicates that the model's predictions are, on average, within a reasonable range of the actual values. Finally, the train and test R-squared values are 0.8748530315439078 and 0.8456415571208442 respectively, which suggests that the model explains a significant portion of the variance in the target variable.

Below is the predicted price against the real sale price of houses, which is relatively linear.



References

* [What is a Yeo-Johnson Power Transformation? | Ready Signal](https://readysignal.com/what-is-a-yeo-johnson-power-transformation/#:~:text=A%20Yeo-Johnson%20Power%20Transformation%20works%20similarly%20to%20the,the%20ability%20to%20transform%20data%20with%20negative%20numbers.)
* [Lasso Regression Explained, Step by Step (machinelearningcompass.com)](https://machinelearningcompass.com/machine_learning_models/lasso_regression/)
* [Complete Guide to Feature Engineering: Zero to Hero (analyticsvidhya.com)](https://www.analyticsvidhya.com/blog/2021/09/complete-guide-to-feature-engineering-zero-to-hero/#:~:text=Feature%20engineering%20fulfils%20mainly%20two%20goals%3A%201%20It,improving%20the%20performance%20of%20machine%20learning%20models%20magically.)
* [House Prices - Advanced Regression Techniques | Kaggle](https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques)
* [Regularization Part 2: Lasso (L1) Regression - YouTube](https://www.youtube.com/watch?v=NGf0voTMlcs)