**Introduction**

This report conducts an in-depth analysis of the Boston housing market, leveraging a blend of machine learning models to predict housing price trends.

1. **Data Description and Exploration**

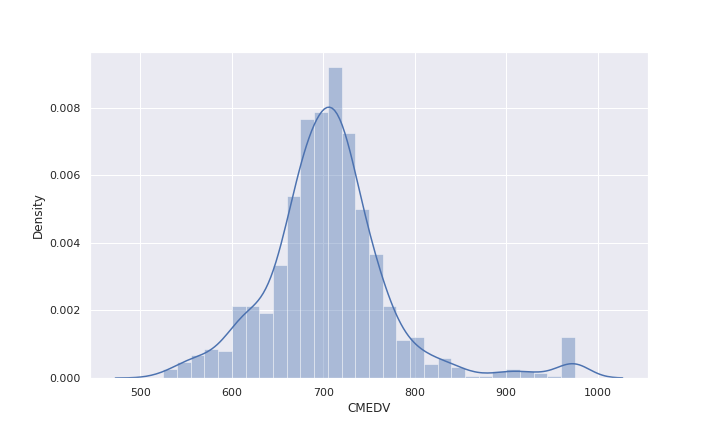
The dataset comprises 14 explanatory variables (features) and one dependent variable (CMEDV), making up a total of 1,000 records. Upon initial exploration, CMEDV displays a somewhat normal distribution with a minor right skew (Figure:1a). Importantly, a smaller proportion of houses (30.2%) have a value greater than $725,000, potentially leading to imbalanced class issues (Figure:1b).

Figure a: Histogram of CMEDV (Median Value of Owner-Occupied Housing)

A picture containing line, rectangle, screenshot, text

Description automatically generated

Figure 1b: Distribution of CMEDV Prices Classified as Low (left) and High (right) using a Threshold of $725,000

Several variables, such as ZN and INDUS, display irregular distributions and may require preprocessing. ZN, for instance, shows several outliers, with a maximum value of 100 sharply contrasting a mean value of 11 and a 75th percentile value of 12.5. INDUS is significantly right-skewed, suggesting a need for transformation to enhance model performance (Figure:3).

Investigating correlations, NOX, RM, and CRIM exhibit strong associations with CMEDV, whereas AGE, DIS, and CINEMA show little correlation. Interestingly, NOX, RM, and CRIM have high intercorrelations, suggesting potential multicollinearity [1], which could affect model interpretability and stability (Figure:2&3).

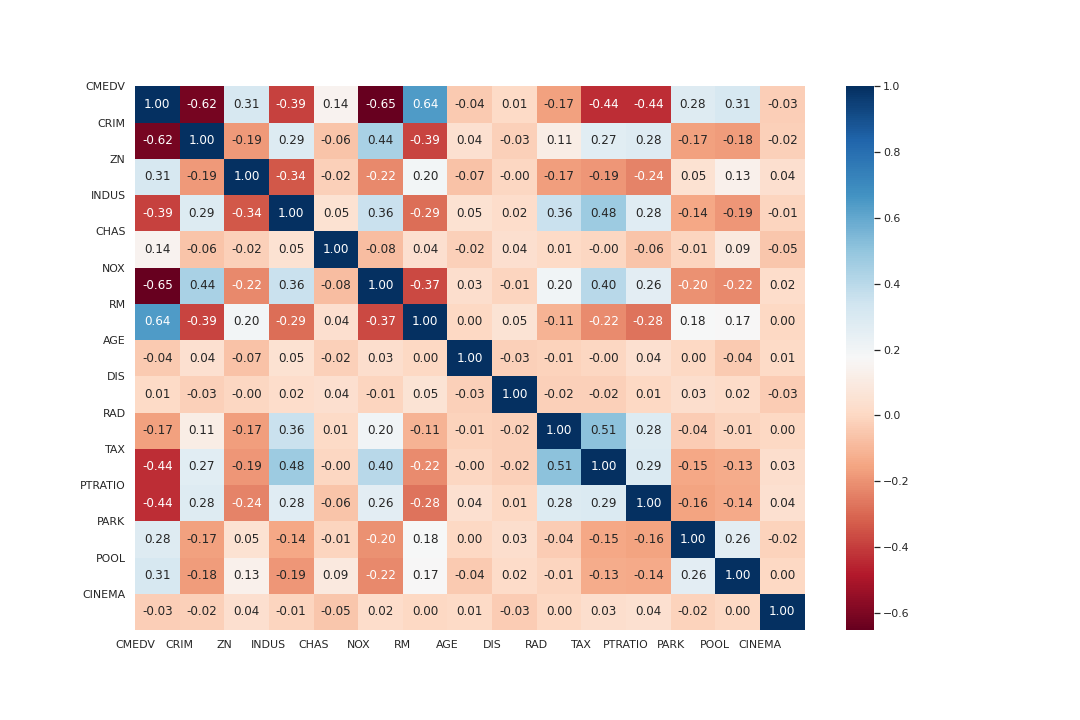


Figure : Correlation Heatmap of All Variables

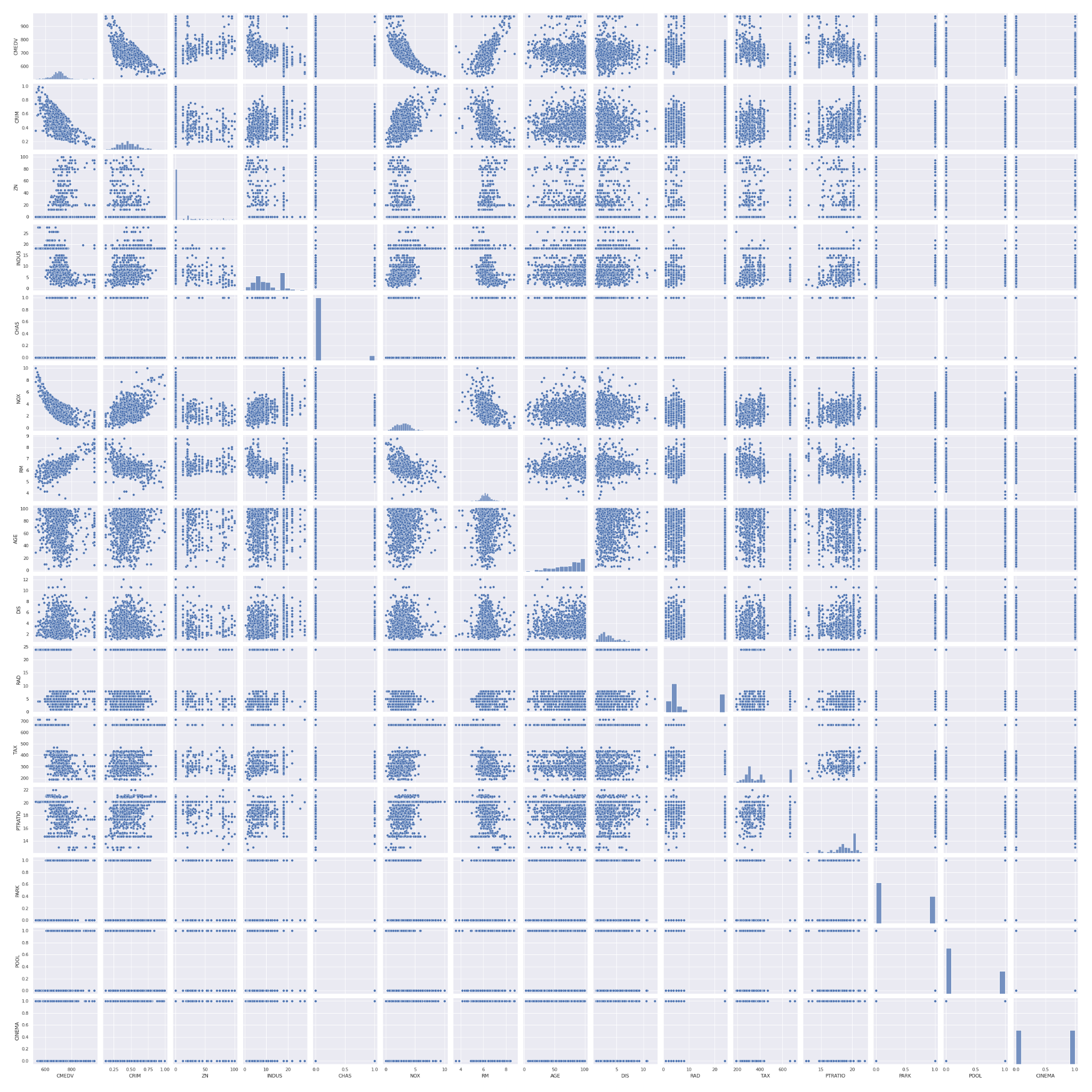


Figure :Scatter Plot (Pair Plot) Showing the Relationship Between Variables and Distribution of Each Variable on the Diagonal

These initial findings emphasise the importance of data pre-processing, addressing skewness, outliers, and potential multicollinearity for reliable and interpretable modelling.

1. **Linear Regression Model with All Features**

A linear regression model was trained on 80% of the dataset and tested on the remaining 20%. We normalised the dependent variable (CMEDV) and standardised the independent features. The model performance was assessed via R-squared, adjusted R-squared, MSE, and MAE.

On the training dataset, the model explained approximately 75.5% of the variance, with MSE and MAE being 0.236 and 0.343, respectively. However, the model exhibited a slight dip in performance when tested, with an R-squared value of 0.715, and a higher MSE (0.318) and MAE (0.371). A visible discrepancy was seen in the model's residuals (Figure 4) which indicated substantial prediction errors for higher CMEDV values, suggesting the model's limitation in accounting for more complex, possibly non-linear, relationships.

A picture containing text, plot, line, screenshot

Description automatically generated

Figure :Actual vs Predicted Values (Test Set) with Error Bars, showcasing model accuracy and prediction errors.

Furthermore, the model's coefficient analysis (Table:1) revealed RM, NOX, and CRIM as significant predictors (P-value <0.05) contributing to CMEDV prediction, while AGE, DIS, and CINEMA (P-value >0.05) showed a lack of significance.

Despite the model's reasonable predictive capability, its limitations, particularly when forecasting higher CMEDV values, must be acknowledged.

Table 1: Table for our linear model:

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Coefficient | T-statistic | P-value |
| CRIM | -1.4299 | -2.9127 | 0.0000 |
| ZN | 0.3631 | 0.7397 | 0.0000 |
| INDUS | 0.0620 | 0.1264 | 0.0000 |
| CHAS | 0.2778 | 0.5658 | 0.0000 |
| NOX | -2.0339 | -4.1430 | 0.0000 |
| RM | 3.1443 | 6.4049 | 0.0000 |
| AGE | 0.0137 | 0.0279 | 0.3000 |
| DIS | -0.0930 | -0.1895 | 0.7099 |
| RAD | 0.2334 | 0.4754 | 0.0000 |
| TAX | -0.5691 | -1.1593 | 0.0000 |
| PTRATIO | -0.7350 | -1.4972 | 0.0000 |
| PARK | 0.1386 | 0.2823 | 0.0000 |
| POOL | 0.1425 | 0.2902 | 0.0000 |
| CINEMA | -0.0461 | -0.0940 | 0.4605 |

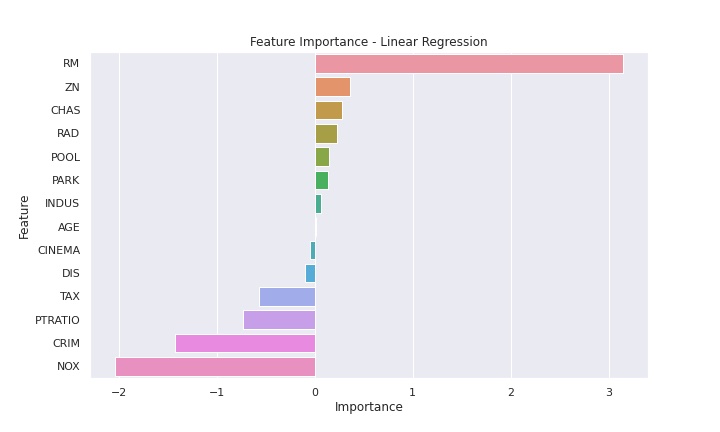


Figure : Coefficients bar-chart visualisation for Linear Regression model

1. **Classification Using Linear Regression**

Utilising a repurposed linear regression model for house price classification, we achieved 82.5% accuracy, as illustrated in the confusion matrix (Figure 6). To ensure a balanced representation of both 'high' and 'low' priced houses in the training and testing sets, we employed a stratified split strategy during model training. This was vital due to the skewed nature of the dataset, with fewer high-priced houses.

Despite these measures, the model’s precision varied significantly; it was impressive for low-priced houses (90%) but dropped to 71% for high-priced ones, suggesting difficulties in accurately predicting the latter. Yet, the model demonstrated commendable recall rates, correctly identifying 83% of low-priced houses and 82% of high-priced houses. This suggests a minor tendency for the model to misclassify 'low' as 'high', particularly because of the data's skewness towards lower prices. This underlines the complexity of modelling outliers and the requirement for models capable of capturing complex, non-linear relationships in the data.

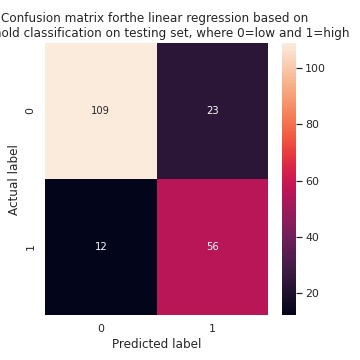


Figure : Confusion Matrix for Linear Regression-based Classification on Testing Set, where 0 represents low median house prices and 1 represents high median house prices.

1. **Alternative Models for Forecasting and Classification**

To address the dual challenges of (a) forecasting median house prices and (b) classifying them as 'high' or 'low', we propose adopting the eXtreme Gradient Boosting (XGBoost) model [2]. This algorithm falls under the Gradient Boosting framework [1] and provides us with the flexibility to handle intricate nonlinear patterns and interactions between variables – aspects identified as critical in our dataset.

To tackle the task of price forecasting, we utilized an XGBoost regression model. Given the intricacy of the model and the limited data points at our disposal, we took a meticulous approach to hyperparameter tuning. We employed a grid search technique coupled with 5-fold cross-validation, focusing on a combination of model complexity control parameters and regularization features such as 'n\_estimators', 'learning\_rate', 'max\_depth', 'min\_child\_weight', 'subsample', 'colsample\_bytree', 'gamma', 'random\_state', and 'reg\_alpha' for L1 regularization. This approach led us to optimal hyperparameters as follows: {'colsample\_bytree':0.6, 'gamma':0.1, 'learning\_rate':0.01, 'max\_depth':5, 'min\_child\_weight':1, 'n\_estimators':1000, 'random\_state':42, 'reg\_alpha':1}. These settings allowed us to build a model that manages the balance between learning complexity and overfitting, which is critical given our data constraints.

On the training set, the XGBoost regression model achieved an MSE of 0.0338 and an R2 score of 0.965, indicating superior generalisation compared to the linear regression model. When evaluated on the test set, the XGBoost regressor further demonstrated its robustness with an MSE of 0.1193, an MAE of 0.28, and an R2 score of 0.89. These metrics are substantial improvements over the linear model, confirming our decision to adopt a more complex, nonlinear modelling approach.

For the classification task, we employed an Gradient Boosting classification model that comes with Sklearn. It was trained and tuned in a similar manner, and the optimal hyperparameters are: {'learning\_rate': 0.05, 'max\_depth': 5, 'max\_features': 'sqrt', 'min\_samples\_leaf': 4, 'min\_samples\_split': 2, 'n\_estimators': 300, 'random\_state': 42}

This classification model achieved an accuracy of 86.5%, an F1 score of 0.8029, and a ROC AUC score of 0.9281 on the test set, showcasing its superior capability in classifying median house prices.

Moreover, for high-priced houses, which is the minority class in our dataset, the XGBoost classifier showed notable improvements (Figure:7) . The precision for high-priced houses improved, which means that when the model predicted a house to be high-priced, it was correct more often. At the same time, the recall rate also increased, implying that the model was able to identify a larger proportion of high-priced houses correctly. This balanced performance is particularly valuable considering the imbalanced nature of the classes in the dataset.

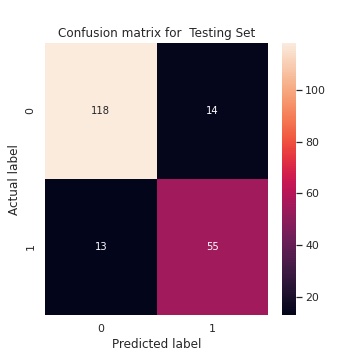


Figure : Confusion Matrix for Gradient Boosting-based Classification on Testing Set, where 0 represents low median house prices and 1 represents high median house prices

Comparing the performance metrics of the XGBoost models to the initial linear regression model (Table:2&3), we can observe significant improvements in both regression and classification tasks. Furthermore, XGBoost offers interpretability advantages over more complex models like deep neural networks. It provides transparent feature importance metrics [1] (Figure:8&9), which are pivotal in our scenario for understanding which features drive housing price predictions and thus facilitating informed decision-making.

Gradient Boosting models align well with our problem space, offering not only improved predictive performance but also valuable insights into the factors influencing house prices, thereby serving as a superior alternative to our initial linear regression model.

Table 2: Regression Model Comparison

|  |  |  |
| --- | --- | --- |
| Metric | Linear Regression Model | XGBoost Regressor |
| MSE | 0.318 | 0.1193 |
| MAE | 0.371 | 0.28 |
| R-squared | 0.715 | 0.89 |

Table 3: Classification Model Comparison

|  |  |  |
| --- | --- | --- |
| Metric | Baseline Linear Regression Classifier | XGBoost Classifier |
| Accuracy | 0.825 | 0.865 |
| Precision | 0.709 | 0.797 |
| Recall | 0.824 | 0.809 |
| F1 Score | 0.762 | 0.803 |
| ROC AUC Score | 0.825 | 0.928 |
| Matthews Correlation Coefficient | 0.629 | 0.700 |

1. **Feature Importance and Selection**

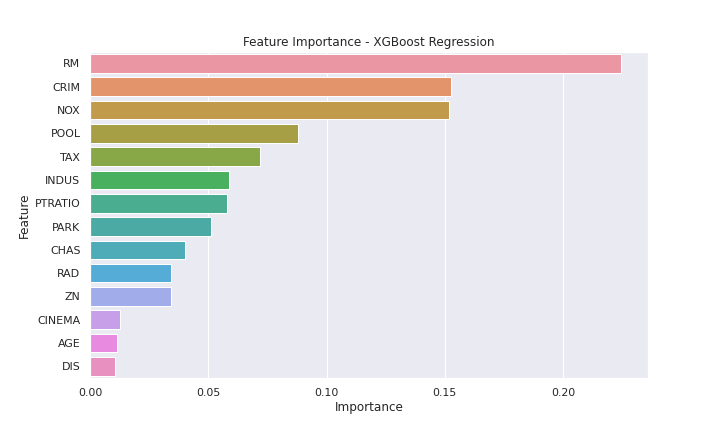


Figure : Confusion Matrix for Linear Regression-based Classification on Testing Set, where 0 represents low median house prices and 1 represents high median house prices

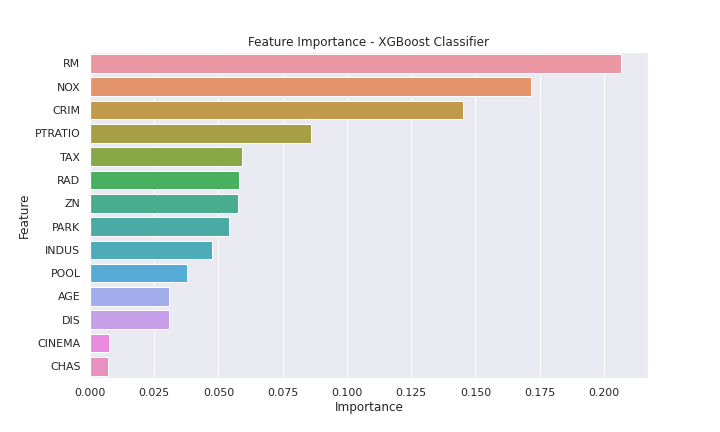


Figure : Confusion Matrix for Linear Regression-based Classification on Testing Set, where 0 represents low median house prices and 1 represents high median house prices

Our Gradient Boosting models for both regression and classification provided feature importance scores indicating the relevance of each feature in predicting housing prices (see Figures 6 and 7). For both tasks, 'RM' (average number of rooms), 'NOX' (nitric oxides concentration), and 'CRIM' (crime rate) consistently emerged as highly influential. These findings underscore the significant impact of room count, environmental quality, and neighbourhood safety on housing prices.

Our variable selection strategy employed XGBoost's built-in feature selection. We started with all available features and let the algorithm rank them based on their importance. This approach ensures we do not overlook any potential influential variables, while the model brings forward those with the highest predictive power.

Given these insights, we recommend Nomad Housing Consultants to prioritise properties with larger room counts, lower nitric oxide concentrations, and lower crime rates when estimating high property values. For BAM Ltd., focusing on developing properties that meet these criteria – more spacious layouts, eco-friendly practices for lower NOX levels, and safer neighbourhoods – could enhance their property value and appeal to potential buyers.

By concentrating on these key features, both companies can sharpen their strategies and decisions, enhancing their competitiveness in the property market.

1. **Prediction and Classification for a Specific Property**

Using the XGBoost regression model, the median house price for the property Saria and Nanuel are considering is predicted to be around $756,866. This price is based on the input features, which include crime rates, zoning restrictions, proximity to the Charles River, nitric oxide concentrations, room count, and accessibility to employment centers, among other factors.

However, when employing the Gradient Boosting classifier model to categorise the house price as 'high' or 'low', it predicted the house price to be 'low'. While this may appear contradictory, we should note that the classifier's predicted probability for this instance is 0.5278, suggesting that the model is relatively unsure about its prediction.

The discrepancy between the regression and classification models can be attributed to the difference in their objectives. The regression model aims to predict a specific price, while the classifier aims to categorise it into 'high' or 'low' price categories based on a predefined threshold. This underlines the fact that modelling decisions and interpretations should be context-dependent.

For Saria and Nanuel, the regression model's prediction is more relevant because they are interested in a specific price estimate. For BAM Ltd., while the classification might be insightful, it's worth noting the classifier's uncertainty in this prediction. Given the proximity to the decision boundary (as indicated by the classifier's probability), BAM might want to treat this area with caution. Depending on their risk tolerance and investment strategy, they could opt to set a different price threshold that better aligns with their business goals.

Overall, it's vital to consider not just the model predictions but also the model's confidence in those predictions when making decisions.

**References**

[1] James, G., Witten, D., Hastie, T. and Tibshirani, R., 2013. An introduction to statistical learning (Vol. 112, p. 18). New York: springer.

[2] XGBoost documentation (no date) XGBoost Documentation - xgboost 1.7.6 documentation. Available at: https://xgboost.readthedocs.io/en/stable/ (Accessed: 29 June 2023).