Deciphering House Price Dynamics Using Predictive Modeling on Housing Data

Allen Zou, Ting-An Lu, Zachary Chao, and Casey Hild

May 12, 2024

Abstract

1 Introduction

The real estate market is a complex ecosystem influenced by various factors, ranging from property characteristics to broader economic trends. Understanding the dynamics of house prices is essential for stakeholders involved in buying, selling, or investing in real estate. This research seeks to unravel the intricacies of house price dynamics through predictive modeling, offering insights that can inform strategic decision-making in the housing market.

In this study, we aim to address two fundamental research questions: What factors influence house prices, and can we construct a predictive model to estimate house prices based on these factors? By delving into these questions, we seek to unravel the complex interplay of factors shaping house price dynamics and explore the feasibility of developing a robust predictive model.

Based on the dataset variables, we hypothesize that certain factors such as property type, location, lot size, overall quality, and condition will have a significant impact on house prices. Additionally, we anticipate that incorporating these factors into predictive models such as linear regression, decision trees, and support vector machines (SVMs) will enable accurate estimations of house prices. Finally, we hypothesize that linear regression, decision trees, and SVMs will effectively estimate house prices based on these factors. SVMs are expected to provide the most accurate predictions, while decision trees offer superior interpretability, albeit with a slight reduction in prediction accuracy. While we believe linear regression is viable for both roles, we also anticipate that it won't perform as effectively as the other two models in their respective hypothesized strengths.

2 Dataset

The dataset used in this study consists of two comma-separated values (CSV) files: train.csv and test.csv. The train.csv file comprises a tabular format with 1,460 rows and 81 columns, while the test.csv file contains 2,919 rows and 81 columns. Each row in both datasets represents a distinct property sale transaction, serving as a unique sample in the dataset.

The first column of each dataset is labeled "Id" and contains a unique identifier for each property sale. The target variable, "SalePrice," is included as the last column, representing the sale price of each property. The features of each property sale transaction are captured in the columns between the first and last columns of the dataset.

The dataset encompasses a wide array of features that describe various aspects of residential properties. These features include categorical variables such as MSSubClass, which identifies the type of dwelling involved in the sale, and MSZoning, which indicates the general zoning classification of the sale. Additionally, numerical variables such as LotFrontage (linear feet of street connected to property) and LotArea (lot size in square feet) provide quantitative measurements of property characteristics. Other essential features cover aspects such as property configuration (LotShape, LotConfig), neighborhood information (Neighborhood), and conditions related to the property (Condition1, Condition2). Variables like OverallQual and OverallCond rate the overall material/finish and condition of the house, respectively, while variables like YearBuilt and YearRemodAdd provide information about the original construction and remodel dates. These features, along with many others like Heating, KitchenQual, and GarageType, offer comprehensive insights into the properties included in the dataset.

2.1 Data Pre-processing

The dataset initially comprised separate files for training and testing, with an unusual split of 1461:1460. To rectify this, the data were combined and shuffled, followed by an 80:20 train-test split. Subsequently, due to the presence of numerous missing entries, preprocessing was necessary to ensure compatibility with scikit-learn, which does not support NaN values. We devised an approach to minimize data loss while addressing missing values. Columns with a substantial proportion of NaN values (approximately 10% or more) were initially pruned, removing entries such as LotFrontage, Alley, FireplaceQu, PoolQC, Fence, and MiscFeature. This step significantly reduced the number of NaN values while retaining most of the variables. Subsequently, the remaining rows containing NaN values were removed, resulting in a total of 1581 rows being eliminated. Despite this, the dataset retained 1070 rows, deemed sufficient for subsequent modeling tasks involving linear regression, decision trees, and support vector machines.

3 Methods

3.1 Model Selection

In our study, we adopted a deliberate approach to model selection, aiming to achieve multiple objectives with each chosen algorithm to answer our research questions. In short, we employed three distinct models to analyze housing price data and extract insights into the factors influencing property values. Our initial choice of linear regression was motivated by its simplicity and interpretability. Subsequently, we turned to decision trees as our second model choice. Decision trees also offer interpretability, aligning with our goal of identifying key variables driving house prices. We also recognized the importance of predictive accuracy in our analysis, so we included support vector machines (SVMs) as another model in our study.

3.2 Benefits and Limitations of Models

Before delving into the hyperparameter selection and training process, it's essential to understand the respective strengths and weaknesses of these models, as they influenced our decision to choose them. Let's explore the benefits and limitations of linear regression, decision trees, and Support Vector Machines (SVMs) for modeling tasks.

3.2.1 Linear Regression

Linear regression offers a few strengths that make it a popular choice for modeling tasks. One of its primary advantages is its simplicity, as it is straightforward to implement and interpret. This simplicity makes linear regression particularly suitable for initial exploratory analysis or as a baseline model for comparison. Additionally, linear regression models are computationally efficient, making them practical for quick model prototyping and experimentation. Another key strength is the interpretability of linear regression coefficients, which provide insights into the relationship between predictor variables and the target variable.

However, linear regression also has notable weaknesses. One limitation is its assumption of a linear relationship between predictors and the target variable, which may not adequately capture complex nonlinear patterns present in the data. Linear regression is also sensitive to outliers, as they can disproportionately influence the model's predictions. Furthermore, linear regression relies on several assumptions about the data, including linearity, homoscedasticity, and independence of errors, which may not always hold true in real-world datasets. Nevertheless, we still chose to include linear regression to serve as a useful baseline to compare with the other models.

3.2.2 Decision Trees

Decision trees offer unique strengths that distinguish them from linear regression models. One of their primary advantages is their ability to capture nonlinear relationships between predictors and the target variable, providing flexibility in modeling complex data. Decision trees also inherently rank predictors based on their importance in predicting the target variable, facilitating feature selection and providing insights into variable importance. Additionally, decision trees are highly interpretable, as they provide intuitive visualizations of decision paths, making it easy to understand the model's decision-making process.

Decision trees also have notable weaknesses. They are prone to overfitting, especially with deep trees that capture noise in the data rather than true patterns. Decision trees can also exhibit instability,

as small variations in the data can lead to significantly different decision trees, impacting model robustness. Furthermore, decision trees produce piecewise constant predictions, which may not capture gradual changes in the target variable across predictor space. However, we do have ways to address overfitting and instability, which we will discuss in later sections, particularly through grid search to find the best hyperparameters for this model.

3.2.3 Support Vector Machines

While decision trees are valuable for feature interpretation, we also recognized the importance of predictive accuracy in our analysis. Support Vector Machines (SVMs) offer versatility and robustness in modeling various types of data. One of their key strengths is their ability to model both linear and nonlinear relationships through different kernel functions, providing flexibility in capturing complex patterns in the data. SVMs aim to maximize the margin between classes, leading to robust generalization performance and resistance to overfitting, especially with large margins. Additionally, SVMs perform well in high-dimensional spaces, making them suitable for datasets with many predictors such as the one described in the **Dataset** section.

However, SVMs do have some limitations. They can be complex to tune, requiring careful selection of hyperparameters such as the choice of kernel function and regularization parameters. Training SVMs can be computationally intensive, particularly with large datasets or complex kernel functions, which may limit their scalability. Most imporatntly, SVMs often lack interpretability, particularly with nonlinear kernel functions, making it challenging to understand the model's decision rationale. Despite the lack of interpretability, we chose to include SVMs for their renowned robust predictive performance. Additionally, as described above, we utilize grid search to assist with parameter tuning.

It's also worth mentioning the use of SVMs in predicting house prices because house price is a continuous variable. Support Vector Machines (SVMs) for house price prediction utilize hyperplanes to create a decision boundary in the feature space. The hyperplane is positioned to minimize the error between predicted and actual prices in the training data. To handle continuous variables like house prices, SVM regression discretizes the price range into bins or intervals. This works on the housing dataset because virtually all the house prices are rounded to the nearest thousands. This process divides house prices into distinct segments, allowing the model to effectively predict which price category a house belongs to based on its features. Therefore, even though house prices are rounded to the nearest thousands, SVM regression adapts by discretizing the price range and focusing on minimizing prediction errors within each bin.

3.3 Hyperparameter Selection and Model Training Process

In this section, we delve into the process of hyperparameter selection and model training for each algorithm employed in our study. We discuss our approach to fine-tuning hyperparameters for linear regression, decision trees, and support vector machines (SVMs) and insights gained during the process.

3.3.1 Linear Regression

Starting with linear regression, a straightforward yet fundamental model in machine learning, it has no hyperparameters other than epochs. We initially set the epochs to 1000; however, upon observation, we found that the model's performance plateaued after just 10 epochs. To expedite the training process without compromising accuracy, we adjusted the epochs to this lower value.

Additionally, all models require categorical variables to be encoded in some manner. However, in the case of linear regression, one-hot encoding significantly impacts the model's performance. Specifically, while the predictions are accurate relative to the true prices, they are consistently lower by a factor of several hundred thousand. This discrepancy is detailed in later plots. To rectify this issue, we opted to remove categorical variables from the linear regression model exclusively.

3.3.2 Decision Trees

Moving on to decision trees, we adopted a more comprehensive approach to hyperparameter tuning. Leveraging grid search, along with one-hot encoding for categorical variables (required by sklearn), we explored various combinations within predefined ranges for parameters such as 0-20 for 'max_depth', 1-4 for

'min_samples_split', 1-4 for 'min_samples_leaf', various functions for 'max_features', and 0.0-0.5 for 'min_impurity_decrease'. These ranges were carefully selected based on empirical evidence from manual experimentation, where we observed optimal performance around specific values. However, no significant challenges were encountered during this tuning process, highlighting the robustness of decision trees.

3.3.3 Support Vector Machines

Hyperparameter tuning for the support vector machine (SVM) model introduced unique considerations, particularly in the context of kernel selection and parameter optimization. Leveraging grid search methodology and incorporating one-hot encoding for categorical variables (a prerequisite for SVM implementation in sklearn), we explored various parameter combinations. However, manual inspection of the tuning process revealed an unexpected trend. Despite rigorous experimentation with hyperparameters such as gamma and regularization parameter (`C`), marginal improvements in model performance were observed.

Remarkably, we observed that the choice of kernel type exerted the most significant influence on model performance. Notably, the linear kernel consistently produced satisfactory results, achieving a mean squared error (MSE) below 0.001. In stark contrast, employing alternative kernels resulted in a substantial escalation of MSE, often exceeding a million. This observation underscores the superior performance of the linear kernel compared to other kernel functions, highlighting its efficacy in the context of our dataset.

Furthermore, despite exploring additional hyperparameters specific to non-linear kernels (e.g., degree for polynomial kernel, coef0 for polynomial and sigmoid kernels), grid search consistently favored the linear kernel configuration for its superior predictive capability and generalization performance.

Lastly, our experimentation revealed that lowering the learning rate below 0.1 drastically increased the training time. Upon reaching approximately 0.03 learning rate, training progress stagnated. Interestingly, variations in the learning rate did not significantly affect model performance when combined with other hyperparameters. Consequently, we omitted the learning rate as a hyperparameter to streamline computational resources without compromising model efficacy.

3.4 Final Hyperparameters

After hyperparameter tuning using sklearn's GridSearchCV, we finalized the hyperparameters for the decision tree and SVM models in the tables shown below.

Hyperparameter	Value
max_depth	10
max_features	'auto'
min_impurity_decrease	0.4
min_samples_leaf	2
min_samples_split	2

Table 1: Optimal hyperparameters for the decision tree model.

Hyperparameter	Value
C	0.1
epsilon	0.1
gamma	'scale'
kernel	'linear'

Table 2: Optimal hyperparameters for the SVM model.

Since linear regression does not involve hyperparameters and its performance plateaued after 10 epochs, we set the number of epochs to 10 for efficiency.

4 Results

4.1 Performance Metrics

The performance of each model was evaluated using two common metrics: Mean Absolute Error (MAE) and Mean Squared Error (MSE). MAE measures the average magnitude of errors in predictions without considering their direction. Conversely, MSE calculates the average of the squares of the errors, giving higher weights to larger errors and penalizing outliers more heavily. It's also worth mentioning that, due to the issues with categorical variables outlined in earlier sections, the information displayed describes a linear regression model that was trained after pruning categorical variables.

Model	MSE
Linear Regression	864,593,724
Decision Tree	1,201,665,016
SVM	1,205,920,170

Table 3: Mean Squared Error (MSE) for each model

Model	MAE
Linear Regression	22,025
Decision Tree	22,568
SVM	23,690

Table 4: Mean Absolute Error (MAE) for each model

In this section, we compare the performance of the three selected models: linear regression, decision trees, and support vector machines (SVMs). Our goal is to assess the effectiveness of each model in predicting housing prices and to gain insights into their strengths and limitations.

Contrary to our hypothesis, linear regression demonstrates relatively better performance compared to the other models, with lower MAE and MSE values. Decision trees exhibit comparable performance to linear regression, albeit with slightly higher MAE and MSE values. Lastly, support vector machines (SVMs) demonstrate the worst predictive accuracy by a small margin among the three models, as indicated by their higher MAE and MSE values.

In terms of MAE, we can easily rank the performances from best to worst in the order of linear regression, decision tree, and SVM. However, if we use MSE as the evaluation metric, then the decision tree and SVM seem to perform equally while linear regression outperforms both. Nevertheless, it's still important to note the removal of categorical variables before training the linear regression model. It's possible that this was the reason the model performed better. We'll examine this in more detail in section 5.

4.1.1 Model Suitability for Hypothesis Testing

Among the models considered, linear regression emerges as a promising candidate for addressing the questions outlined in our proposal, with the decision tree not far behind. While the SVM performed relatively similarly to the other two models, the interpretability of SVMs remains limited. Therefore, for the interpretability aspect of our hypothesis, we rely mostly on the other two models. The subsequent analysis will primarily focus on the insights derived from linear regression, but we will still extract insights from the decision tree model.

5 Interpretation and Analysis

In this section, we delve into the interpretation of our models and the visualization of their results. Additionally, we present visualizations to illustrate the performance and predictive patterns of our models, providing a comprehensive understanding of their behavior and effectiveness in predicting house prices. We begin by exploring the comparison of predicted prices and true prices. Each plot showcases the predicted values in orange and the true prices in blue, with all test data points displayed to provide a comprehensive overview of the model's performance.

5.1 Comparison of Predictions and True Prices

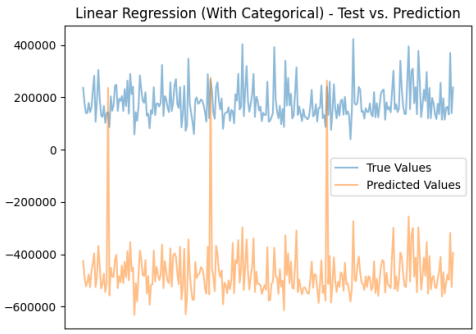

Figure 1: Linear Regression (With Categorical)

First, let's address the issue encountered when using categorical variables in linear regression. As depicted in Figure 1, the predicted prices from linear regression appear visually accurate relative to the true prices, but it's noteworthy that the predictions consistently fall short by a considerable margin, typically by several hundred thousand dollars. Despite attempts to scale the data and address the fixed values resulting from one-hot encoding, no significant improvement in model performance was observed. Consequently, we opted to exclude categorical variables from the linear regression model. For the rest of the analysis, we'll focus solely on

the linear regression model, excluding categorical variables.

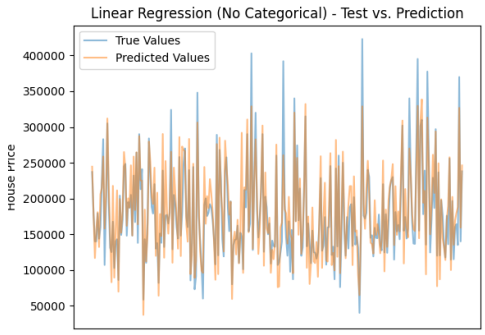

Figure 2: Linear Regression (No Categorical)

After removing categorical variables from the linear regression model, we observed a noticeable improvement in the alignment between predicted and true prices, as shown in Figure 2. The predictions now exhibit a closer match to the true prices. However, it's important to note that while the overall accuracy improved, there still exists some variance in the accuracy of predictions. Certain data points still demonstrate considerable deviations from the true prices, indicating that while categorical variable removal enhanced performance, further refinements may be necessary to achieve consistent accuracy across all instances.

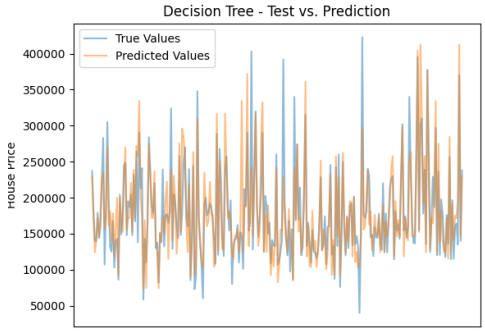

Figure 3: Decision Tree

Figure 3 illustrates the predictions generated by the decision tree model. The plots for the test and prediction values for both linear regression and decision tree appear almost identical. Therefore, to determine more objectively which model performs better overall, we must examine previous metrics like MSE or MAE, which quantify the error of the predictions. Once again, this contradicts our hypothesis, as we anticipated that decision trees and SVMs would both outperform linear regression in terms of prediction accuracy. However, as shown in Figure 3, there isn't any noticeable difference in the performance of these two models.

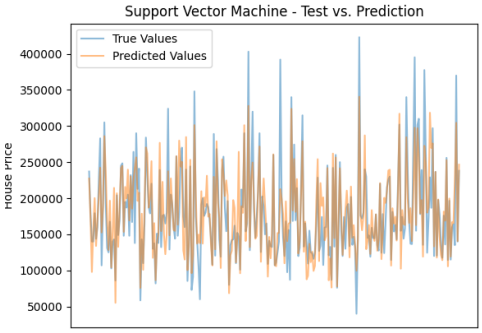

Figure 4: Support Vector Machine

In Figure 4, we visualize the predictions made by the Support Vector Machine (SVM) model. Similar to decision trees and linear regression, there isn't any noticeable difference in performance among these three models. Specifcally, the overall performance remains indistinguishable from the others based on the figures.

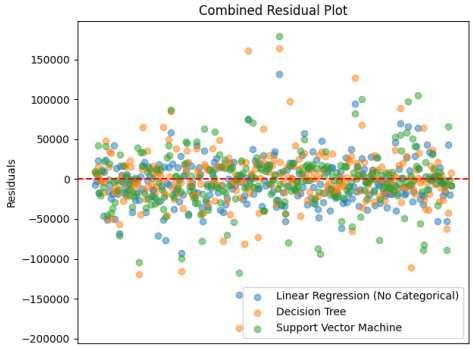

Figure 5: Combined Residual Plot

Additionally, we generated residual plots for each model to assess the distribution of errors. As depicted in Figure 5, the residual plots reveal consistent patterns across all models. The majority of residuals cluster around zero, indicating that the models generally predict house prices close to the true values. However, outliers are visible in the plots, corresponding to the outliers in the line plots, as we maintained the order of data points in the residual plot. Despite these outliers, the residual plots confirm the overall accuracy and effectiveness of the models in predicting house prices.

5.2 Model Interpretation and Insights

In this section, we examine the coefficients of the linear regression model and interpret the decision tree model's variables. However, due to the inherent complexity and lack of interpretability of SVMs, we opt not to interpret their variables.

5.2.1 Interpreting Linear Regression Coefficients

Presented in Table 5, the coefficients are sorted from greatest to least magnitude. Additionally, "SF" stands for "Square Feet."

The coefficients of certain variables in the linear regression model stand out due to their notably larger magnitudes compared to others. For instance, the variables from Ground Living Area to Basement Finished SF Type 2 exhibit coefficients of immense values, suggesting a substantial influence on predicted house prices. As soon as we move down to Overall Quality, the coefficient magnitude drops significantly.

At first, concerns arose regarding whether the inflated coefficients were driven by extremely low mean values of these variables. However, upon observing the mean values provided in Table 6, no apparent correlation emerged between the means and the coefficients. This observation indicates that the significance of these variables extends beyond mere scale, implying potentially strong predictive power. Before comparing these interpretations to our hypothesis, however, we should interpret the decision tree next.

5.2.2 Interpreting Decision Trees

While interpreting decision trees can provide valuable insights into the underlying patterns within the data, in our analysis, we chose not to visualize the decision tree. This decision was influenced by a technical constraint related to the handling of categorical variables within the scikit-learn library. As mentioned in earlier sections, scikit-learn's models do not handle categorical variables natively; they require encoding them in some manner.

Table 5: Coefficients of Linear Regression Model

Feature	Coefficient
Above Grade Living Area	-4.495357e+16
Second Floor Area	3.730668e+16
First Floor Area	3.325525e+16
Type 1 Basement Area	3.752504e+15
Unfinished Basement Area	3.572657e+15
Total Basement Area	-3.349934e+15
Low Quality Finished Area	3.348654e+15
Type 2 Basement Area	1.341976e+15
Overall Quality	2.489973e+04
Year Built	1.156974e+04
Garage Cars	1.055985e+04
Total Rooms Above Grade	9.187627e+03
Bedrooms Above Grade	-7.827398e+03
Dwelling Type Subclass	-7.063226e+03
Masonry Veneer Area	6.610016e+03
Overall Condition	6.165559e+03
Basement Full Bathrooms	5.199578e+03
Lot Area	4.255464e+03
Screen Porch	4.247150e+03
Kitchens Above Grade	-3.970248e+03
Wood Deck Area	3.839578e+03
Half Bathrooms	-3.337457e+03
Fireplaces	2.909189e+03
Remodel Date	2.812860e+03
Pool Area	-1.698960e+03
Basement Half Bathrooms	1.510361e+03
Year Garage Built	-1.368263e+03
Open Porch Area	-9.881667e+02
Month Sold	-9.205764e+02
Three Season Porch Area	7.801328e+02
Year Sold	-5.423254e+02
Miscellaneous Value	-3.486382e+02
Garage Area	2.505075e+02
Enclosed Porch	1.546821e+02
Full Bathrooms	-1.443987e+02

Table 6: Mean Values of Variables with High Coefficients

Variable	Mean
Ground Living Area	55.929907
Second Floor Area	10762.350467
First Floor Area	6.237383
Type 1 Basement Area	5.603738
Unfinished Basement Area	1973.131776
Total Basement Area	1985.992523
Low Quality Finished Area	109.553271
Type 2 Basement Area	473.315888

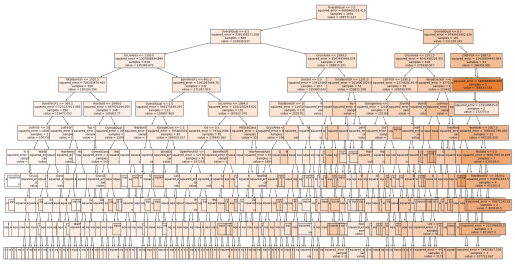

Figure 6: Visualization of Decision Tree

In our case, we utilized one-hot encoding to represent categorical variables. However, this approach results in the decision tree generating an extensive number of branches as shown in 6, rendering visualization challenging and interpretation cumbersome. Due to the sheer complexity and size of the resulting tree, attempting to visualize it would not provide clear or meaningful insights into the relationships between the predictor variables and the target variable. Therefore, instead of relying on visualizations, we examined the decision tree's feature importance to interpret it and extract actionable insights.

The feature importance table (Table 7) displays the top 20 variables with the highest importance as determined by the decision tree model. Importance values are expressed as percentages, representing each variable's contribution to the model's predictive accuracy. It's worth noting that entries such as 'Gd' and 'No' in the table represent variables created for one-hot encoding purposes and are challenging to interpret due to their separated nature. Given that their importance values are less than 0.5%, we chose to treat them as having relatively low significance in predicting house prices.

'Overall Quality' tops the list with an importance of 65.36%, indicating its significant influence on predicting house prices. Additionally, the importance values seem to drop sharply after 'Above Grade Living Area', which holds the second-highest importance at 11.45%. Other features include 'Second Floor Area', 'Basement Area', and 'First Floor Area', contributing 4.65%, 2.76%, and 2.62% respectively. These findings underscore the importance of property quality, size, and layout in determining market value. Additionally, attributes related to time, such as the year built and remodel date, also hold some importance.

Table 7: Feature Importance from Decision Tree

Feature	**Importance**
Overall Quality	65.36%
Above Grade Living Area	11.45%
Second Floor Area	4.65%
Basement Area	2.76%
First Floor Area	2.62%
Type 1 Basement Area	1.68%
Year Built	1.55%
Wood Deck Area	1.43%
Detached from Home	0.96%
Size of Garage	0.66%
Lot Size	0.60%
Remodel Date	0.59%
Unfinished Basement Area	0.50%
Gd	0.43%
Month Sold	0.37%
Three Season Porch Area in SF	0.37%
Vinyl Siding	0.31%
No	0.29%
Open Porch Area in SF	0.29%
Size of Garage in Car Capacity	0.23%

6 Discussion

In comparing the decision tree model with the linear regression model, it's evident that they prioritize different variables in predicting house prices. The decision tree model emphasizes 'Overall Quality' as the most influential factors, while the linear regression model places more importance on various area-related variables such as 'Above Grade Living Area', 'Second Floor Area', and 'Basement Area'. Interestingly, when examining the variables below the top seven for each model, it becomes apparent that the decision tree still assigns some importance to area-related variables, albeit to a lesser degree compared to 'Overall Quality'. This suggests that while decision trees prioritize quality, they still acknowledge the significance of area-related attributes, though to a lesser extent.

Our initial hypothesis posited that factors such as property type, location, lot size, overall quality, and condition would significantly impact house prices. However, the models' results indicate that the primary influencers are area-related variables for linear regression 'Overall Quality' being prominent in decision trees. This divergence from our hypothesis underscores the complexity of house price prediction and highlights the importance of

considering multiple factors beyond the ones initially hypothesized.

The predictive models, including linear regression, decision trees, and support vector machines (SVMs), demonstrate relatively accurate estimations of house prices, as evidenced by the MSE and MAE values. Despite variations in the importance of different variables, the models consistently provide predictions with deviations in the low 20,000's range, which are relatively small compared to the actual house prices in the data. However, the observed ranking of performance contradicts our initial hypothesis, with the literal interpretation of MSE and MAE values suggesting a different ranking. Specifically, while our hypothesis predicted SVMs to have the best performance followed by decision trees and then linear regression, the opposite appears true when interpreting the MSE and MAE values directly.

Additionally, it's important to note the limitations of the SVM model in terms of interpretability due to its reliance on hyperplanes. Furthermore, despite the low variance in MSE and MAE values across the models, the literal interpretation of these values relative to MSE and MAE values for linear regression and decision trees suggests that SVMs may not be the best-performing model in this context, contrary to our expectations.

7 Research Limitations and Future Directions

While our study provides valuable insights into predicting house prices using machine learning models, several limitations and avenues for future research should be considered.

Firstly, a significant limitation stems from the pruning of data due to missing entries. 1,851 rows were lost across the dataset. While this didn't noticeably impact the model's training and predictive capabilities, this is over half the total data. While addressing missing values, we experimented with imputation techniques, such as mean imputation, to handle the data loss. However, these approaches resulted in decreased model performance. Future research should explore advanced imputation methods, such as k-nearest neighbors or predictive modeling, to effectively mitigate data loss while maintaining model accuracy.

Secondly, it's important to recognize that Dean De Cock's dataset is specific to residential homes in Ames, Iowa. While the findings offer insights into housing market dynamics in this locality, generalizing the results to other geographical areas should be done cautiously. Additionally, the peculiarities of the train-test split in our dataset may warrant reshuffling and redistributing the data or cross-validation for more robust model evaluation.

Furthermore, the inability to analyze categorical variables due to compatibility issues with certain models, such as linear regression, or interpretability challenges with decision trees using one-hot encoding, presents a notable issue. Future research directions could involve exploring alternative encoding techniques or model architectures to incorporate categorical variables effectively and enhance model interpretability.

Moreover, while SVMs performed adequately, interpreting them can be challenging due to their complex decision boundaries, especially in high-dimensional feature spaces. Understanding the relationship between input features and model predictions in SVMs is not as straightforward as in simpler models like linear regression or decision trees. Therefore, work on techniques for interpreting SVMs and extracting meaningful insights from them could be a valuable direction for future research.

Lastly, given that only a subset of variables, such as specific areas within the house and overall quality, significantly influences the models, exploring dimensionality reduction techniques like Principal Component Analysis (PCA) could be beneficial. PCA can help identify latent variables and reduce the dimensionality of the feature space while preserving the most critical information, potentially improving model performance and interpretability.

Addressing these limitations and pursuing future research directions will contribute to advancing the accuracy, robustness, and interpretability of models for predicting house prices.

8 Conclusion

In this study, we explored the use of models—linear regression, decision trees, and support vector machines—to predict house prices based on various features. Our analysis revealed that while all models provided relatively accurate predictions, some prioritized different variables in determining house prices.

The linear regression model emphasized area-

related attributes, such as above-grade living area and basement area, while the decision tree model focused on overall quality as the most influential factor. Contrary to our initial hypothesis, the SVM model did not outperform linear regression and decision trees in terms of predictive accuracy. Additionally, the SVM's inherent complexity and lack of interpretability limits our ability to extract insights.

Despite the limitations of our study, including data pruning due to missing entries and compatibility issues with categorical variables, our findings offer valuable insights into the complexities of housing market dynamics and highlight the need for a more nuanced understanding of the factors influencing house prices.

Moving forward, future research should address these limitations and explore advanced techniques, such as imputation methods and dimensionality reduction techniques, to enhance model performance and interpretability. By doing so, we can further advance the accuracy and robustness of machine learning models for predicting house prices, contributing to more informed decision-making in the real estate industry.