Danish Omar Mohammed DOM200002
Hamzah Weldingwala MHW200002

Predicting House Price Categories in King County, Seattle

---

1. Introduction

The goal of this project is to develop a machine learning model that classifies houses in King County, Seattle into three price categories: Low, Medium, or High, which could possibly be converted into a tool to help sellers and realtors price properties. Instead of predicting the exact sale price, we simplified the problem by placing the price values into categories. This allowed us to better evaluate model performance using classification metrics such as accuracy scores, confusion matrices, and ROC-AUC.

We used the publicly available King County House Sales dataset from Kaggle, which includes 20,000+ data points and 21 features such as the number of bedrooms, bathrooms, square footage, year built, location coordinates, and more.

---

2. Data Preprocessing

2.1 Data Cleaning

As our first step, we cleaned the data set by removing the ***id*** and ***date*** features, as these don't play a part in determining a property's price and would just be unnecessary data. We also checked the dataset for any values that were missing any features. Since we had over 20k data points, we automated this task rather than doing it manually.

2.2 Feature Engineering

We replaced the original price column, which had continuous data with separate price points for each property, with a new column called ***price_category,*** which had three price categories: low, medium, and high. Low consisted of properties priced at or lower than $400k, medium was properties greater than $400k and less than or equal to $800k, while high consisted of any properties greater than $800K. This change made it easier to classify data rather than using the original price system.

Danish Omar Mohammed DOM200002
Hamzah Weldingwala MHW200002

2.3 Features Used by the Model

The model used the following features:

- bedrooms: Number of bedrooms
- bathrooms: Number of bathrooms
- sqft_living: Square footage of the interior living space
- sqft_lot: Square footage of the lot
- floors: Number of floors
- waterfront: Whether the house has a waterfront view
- condition: Overall condition of the house
- grade: Construction and design grade
- sqft_above: Square footage above ground level
- sqft_basement: Square footage of the basement
- yr_built: Year the house was built
- yr_renovated: Year the house was renovated (0 if never)
- zipcode: Zip code of the property
- lat: Latitude
- long: Longitude
- sqft_living15: Living area of 15 nearest neighbors
- sqft_lot15: Lot area of 15 nearest neighbors

---

3. Model Selection and Rationale

We selected these five classifiers:

1. **Decision Tree Classifier**
2. **Naive Bayes**
3. **k-Nearest Neighbors (k-NN)**

Danish Omar Mohammed DOM200002
Hamzah Weldingwala MHW200002

4. **Logistic Regression**
5. **Support Vector Machine (SVM)**

---

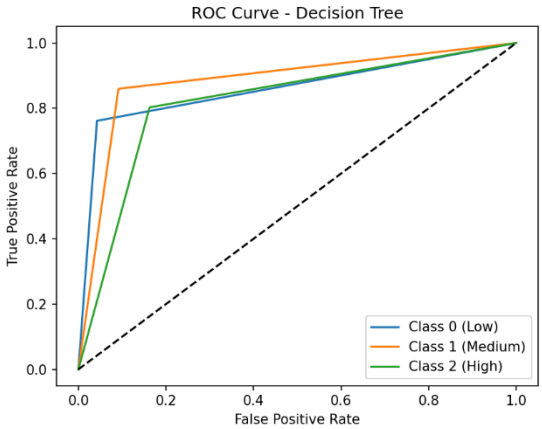
4. Experimental Setup

We decided to do an 80/20 train-test split, but in retrospect, we think it might've been a better idea to also include a validation set to prevent overfitting. For evaluation, we decided to base it on multiple metrics: accuracy rates to determine the overall winner, confusion matrices to identify distribution of correct and incorrect classifications, ROC-AUC, and TPR/FPRs.

---

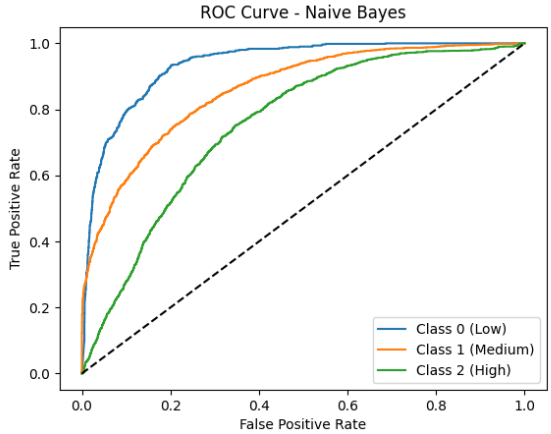
5. Results and Visualizations

Decision Tree Classifier




```
Decision Tree
Accuracy: 0.8191071015498497
Confusion Matrix:
 [[ 460    3  141]
 [   4 1523  244]
 [ 155  235 1558]]
AUC Score: 0.8545484818653207
```

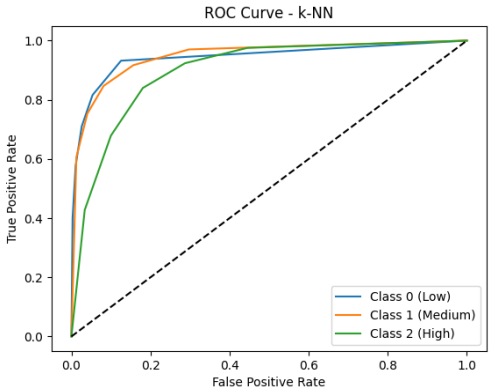
Naive Bayes




```
Naive Bayes
Accuracy: 0.6585704371963914
Confusion Matrix:
 [[ 412   17  175]
 [   9 1537  225]
 [ 177  873  898]]
AUC Score: 0.8501974069898469
```

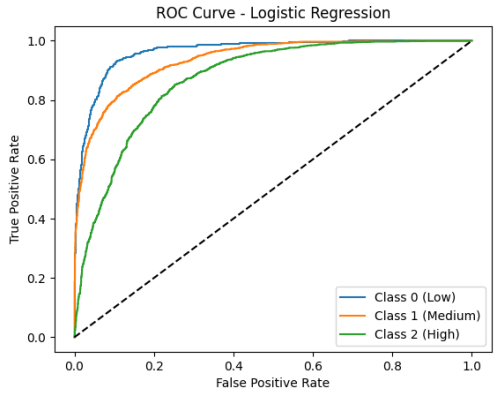
Danish Omar Mohammed DOM200002
Hamzah Weldingwala MHW200002

k-Nearest Neighbors



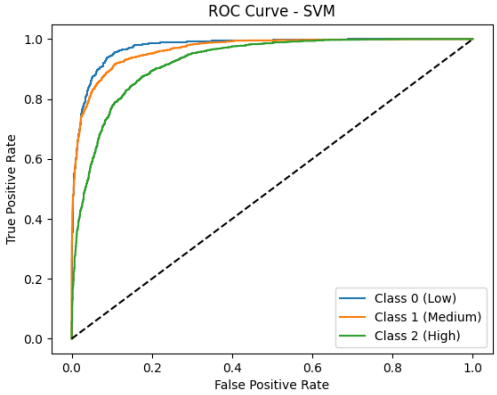

```
k-NN
Accuracy: 0.8272033310201249
Confusion Matrix:
 [[ 433    4  167]
 [   2 1507  262]
 [ 103  209 1636]]
AUC Score: 0.9289487984964072
```

Logistic Regression




```
Logistic Regression
Accuracy: 0.7922738838769373
Confusion Matrix:
 [[ 400    5  199]
 [   1 1437  333]
 [  91  269 1588]]
AUC Score: 0.9233827803728308
```

Support Vector Machine (takes a bit longer in the code to generate charts)




```
SVM
Accuracy: 0.8452463566967384
Confusion Matrix:
 [[ 441    2  161]
 [   1 1523  247]
 [  84  174 1690]]
AUC Score: 0.955260849212252
```

Danish Omar Mohammed DOM200002
Hamzah Weldingwala MHW200002

---

6. Discussion and Analysis

Linear Regression performed well in terms of AUC, but its accuracy rate was the second lowest; however, the AUC rate implies that the price categories are linearly separable, which validates the splitting of data into three categories. Decision Trees had a better accuracy rate, but its AUC was lower; it might've captured more of the complex feature interaction but could've also overfitted on the data. Naïve Bayes performed the worst, with the lowest accuracy and AUC rates, making it the least optimal classifier. K-NN performed well, with the 2nd highest AUC and accuracy rates, but it fell short to SVM, which had the highest AUC and accuracy rates, making it the best classifier for this dataset. It might've performed even better if tested across more dimensions.

---

7. Conclusion

Its success could be attributed to SVMs ability to model complex patterns and identify relations between non-linear data, like the relationship between the ***sqft_living***, ***grade***, and ***zipcode*** features. Also, SVM works by maximizing the margin between classes, such as Low, Medium, High, whose boundary the SVM can find accurately. Lastly, the high number of features in this dataset made it so that a classifier like SVM, which handles high-dimensional feature spaces more efficiently, performs better.

If given 6 more months to work on the project, we'd improve it by adding more classifiers to the project, such as random forest, to ensure that we've found the classifier that's the most accurate. We'd also build on the SVM by adding more kernels to train across higher dimensions. After making the model as accurate as we can, we'd find a business use case for it, by creating a dashboard for users to input features and getting an appropriate price for that property, and build on the dataset with newer data to make sure the model accounts for appreciation in King County, Seattle.