Predicting House Prices and Identifying Influential Factors: A Regression Analysis of King County Housing

PREPARED BY: HARRIET JOSEPH

AGENDA

- Problem Overview
- Key Findings
- Solution Process
- Recommendations
- Model Comparison
- Potential benefits

01

PROBLEM OVERVIEW

WHAT WE ARE WORKING ON

Business Problem

To analyze the King County House Sales dataset and use regression modeling to predict house prices in the county.

The stakeholders, such as real estate agencies or property management companies, are interested in understanding the factors that influence house prices in the county, such as location, size, condition, and features of the properties.

By gaining insights into these factors, stakeholders can improve their business decisions, such as pricing properties more accurately, investing in the right locations, and negotiating better deals with buyers and sellers.

The project aims to identify the most desirable neighborhoods and property features that attract buyers and renters and to provide accurate advice to homeowners on how to increase the value of their homes.

SOLUTION PROCESS

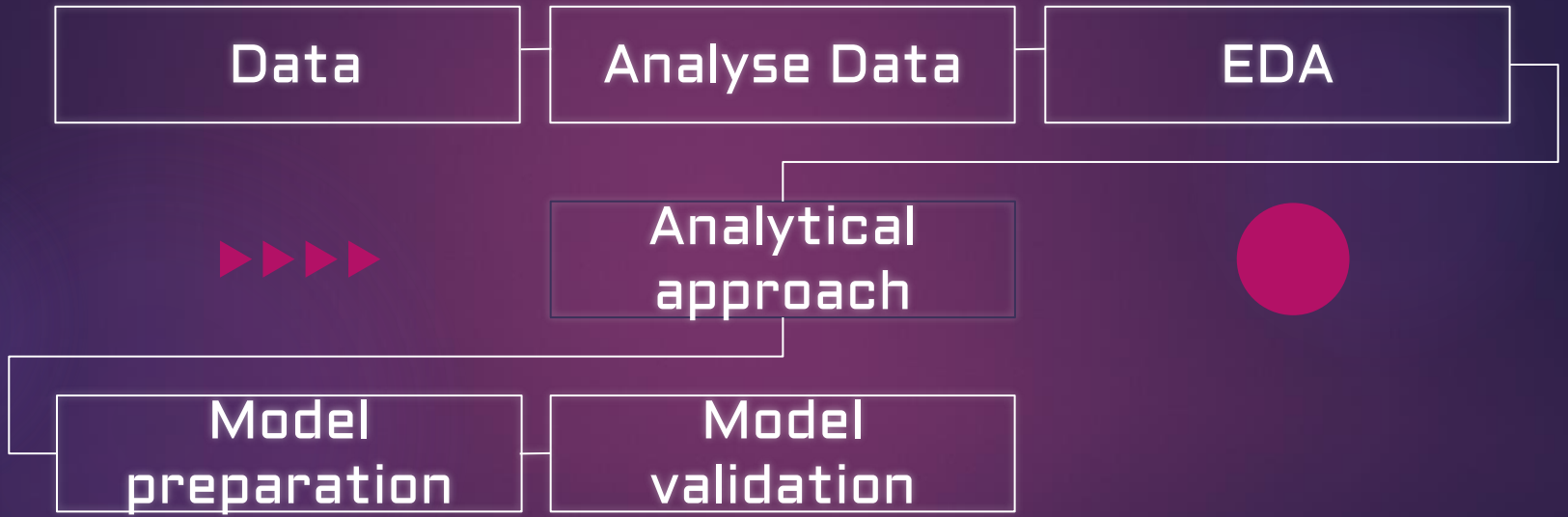
02

ABOUT THE PROJECT

This project uses the King County House Sales dataset, which can be found in kc_house_data.csv and description of the column names can be found in column_names.md.

```
#load the king county house sales dataset
df = pd.read_csv('kc_house_data.csv',
index_col = 0)
df
```

SOLUTION PROCESS ARCHITECTURE



```
df.columns
```

```
Index(['date', 'price', 'bedrooms', 'bathrooms', 'sqft_living', 'sqft_lot',
       'floors', 'waterfront', 'view', 'condition', 'grade', 'sqft_above',
       'sqft_basement', 'yr_built', 'yr_renovated', 'zipcode', 'lat', 'long',
       'sqft_living15', 'sqft_lot15'],
```

our target variable is the price

The dataset contains information about house sales in King County, Washington state, USA.

There are 21,597 entries (rows) and 20 columns.
Each row represents a different house sale and each column represents a different attribute of the house

with over 20,000 observations, we likely have enough data to build a reasonably complex model.
The distribution of the data is not very well specified for all the predictors at this stage, but we know from the

summary statistics provided that the price has a wide range of values,
with a mean of $540,296 and a standard deviation of $367,368.

comprises of are continuous,discrete and categorical data as shown in the plot above.

- **HANDLING MISSING VALUES**

[10]:
```
df.isna().sum()
```

[10]:
```
price              0
bedrooms           0
bathrooms          0
sqft_living        0
sqft_lot           0
floors             0
waterfront      2376
condition          0
grade              0
view              63
yr_built           0
yr_renovated    3842
zipcode            0
sqft_above         0
sqft_basement      0
dtype: int64
```

Ok, it looks like we have some NaNs in waterfront, view and yr_renovated, do these NaNs actually represent *missing* values, or is there some real value/category being represented by NaN?

missing values treatment.

In [11]:
```
#yr_renovated is not categorical ,therefore we can assume that zero represents houses that have never been renovated
#therefore fill missing values with zeroes
df['yr_renovated'].fillna(0, inplace=True)
```

- then views we realise that the view is categorical ranges from 0 to 4, since it represents small percentage we can drop thr rows with NANs

In [12]:
```
df.dropna(subset=['view'], inplace=True)
```

- lastly check on waterfront.
- *waterfront is a categorical data, where o represents, house has no view to a waterfront and 1 represents a house with a view to a waterfront**

In [13]:
```
# inspecting the waterfront column
print(df['waterfront'].value_counts())
print(df['waterfront'].unique())
```

```
0.0    19019
1.0      145
Name: waterfront, dtype: int64
[nan  0.  1.]
```

In [14]:
```
# Calculate the mode of the waterfront column
waterfront_mode = df['waterfront'].mode()[0]

# Fill in the missing values with the mode
df['waterfront'] = df['waterfront'].fillna(waterfront_mode)
```

In [15]:
```
print(df['waterfront'].unique())
```

```
[0. 1.]
```

In [18]:
```
print(df['sqft_basement'].value_counts())
```

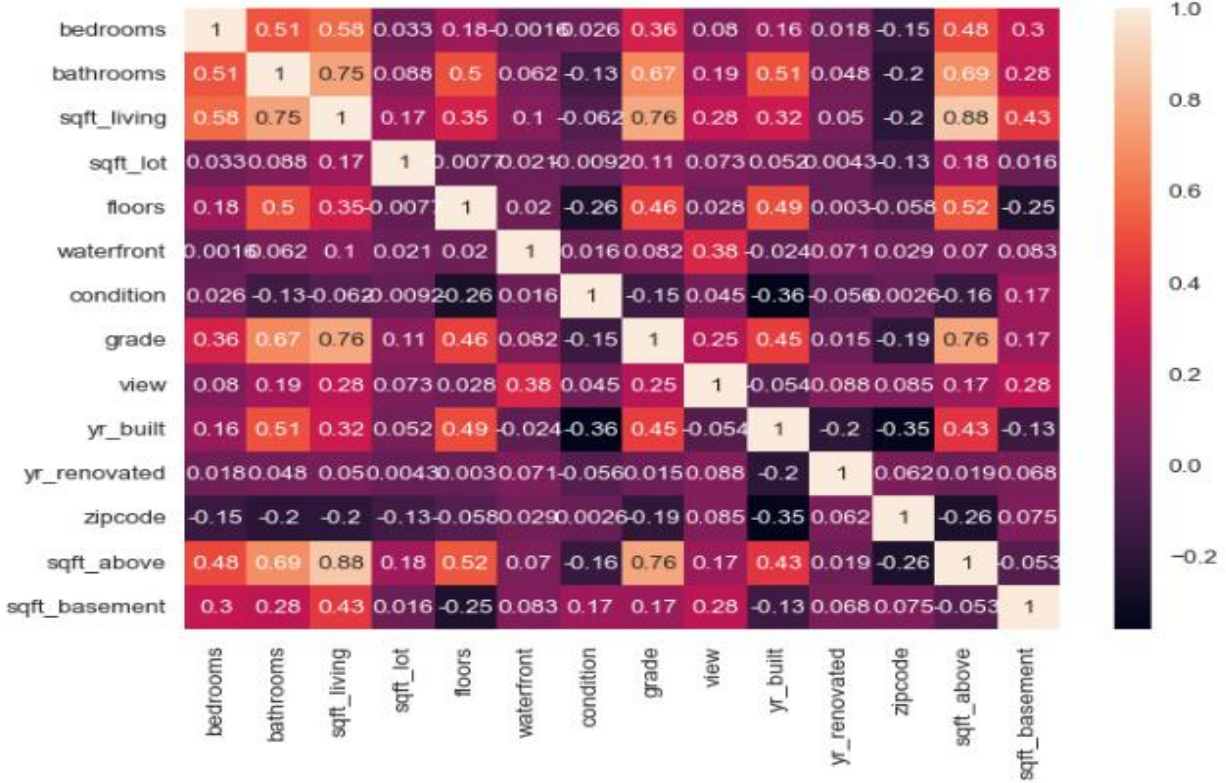
```
0.0       12798
?           452
600.0       216
500.0       209
700.0       207
           ...
3480.0        1
1840.0        1
2730.0        1
2720.0        1
248.0         1
Name: sqft_basement, Length: 302, dtype: int64
```

- **data type conversion**

In [19]:
```
#sqft_basement and date are in object data type
#convert the sqft_basement type to float64
#the column has a special character ? remove that first
df = df[df['sqft_basement'] != '?']
df['sqft_basement'] = df['sqft_basement'].astype('float64')
```

The column ‘sqft_basement’ data type is object; so converted it to float data type

Check for multicollinearity



Which varibles are highly correlated in the Ames Housing data set?

In [27]:
```
# write answer here
df_new[(df_new.cc > .75) & (df_new.cc < 1)]
```

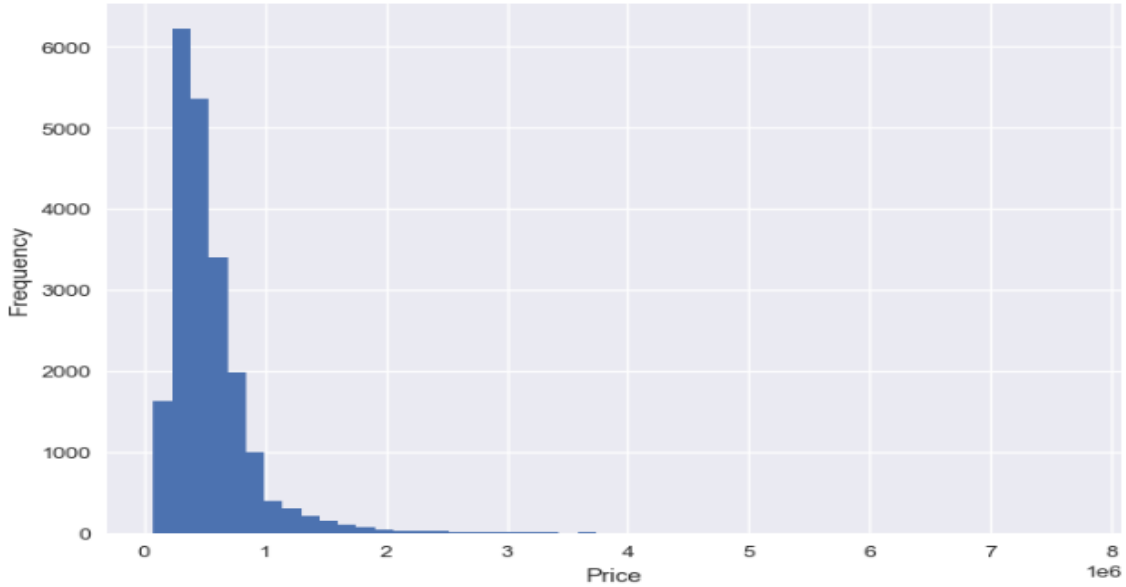
Out[27]:

pairs	cc
(sqft_living, sqft_above)	0.876787
(grade, sqft_living)	0.762719
(grade, sqft_above)	0.756289
(bathrooms, sqft_living)	0.754793

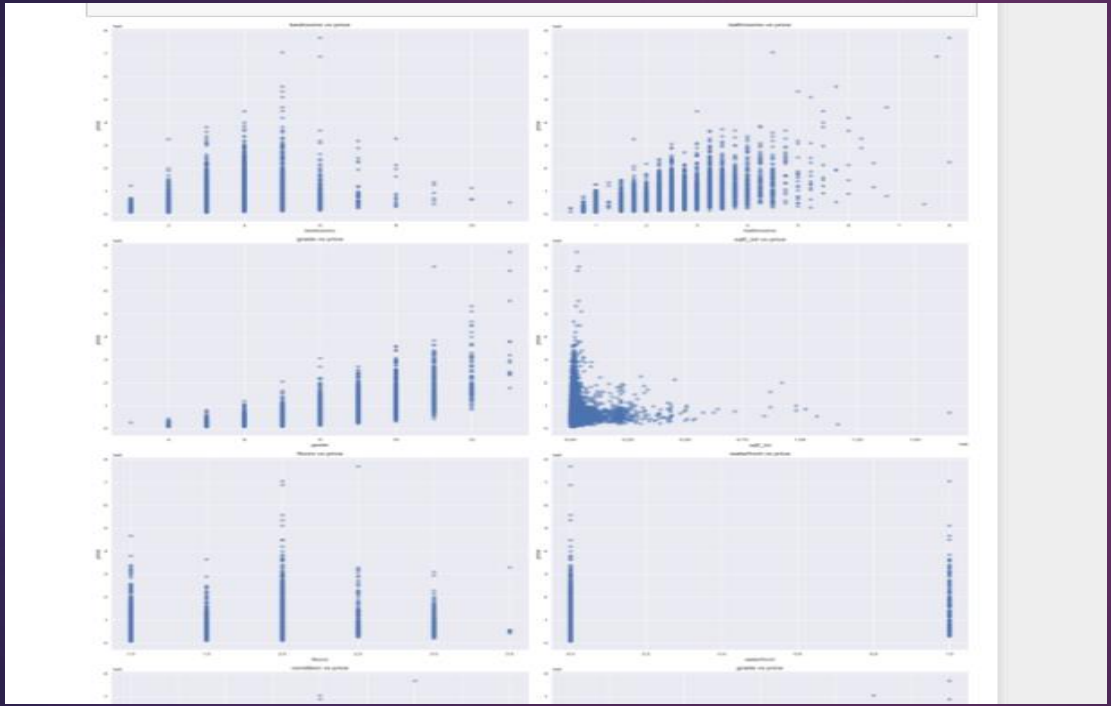
- To solve for multicollinearity, we drop the column from every pair, that means we drop ‘sqft_living’ and ‘sqft_above’.
- This is to reduce causes that it may bring to our regression results since multicollinearity increase the standard errors of the regression coefficients, which can lead to decreased statistical power and difficulty in identifying significant predictors in the model

Univariate Analysis

```
count    2.108200e+04
mean     5.402469e+05
std      3.667323e+05
min      7.800000e+04
25%      3.220000e+05
50%      4.500000e+05
75%      6.450000e+05
max      7.700000e+06
Name: price, dtype: float64
```



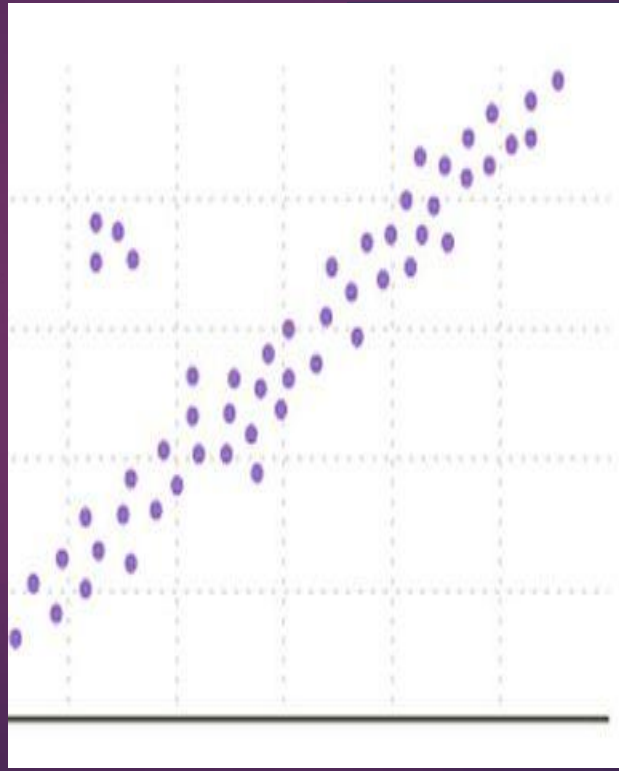
Bivariate analysis



* Some of the scatter plots for the columns of the data.

Analytical approach

In this project the approach to use for prediction is Regression analysis.

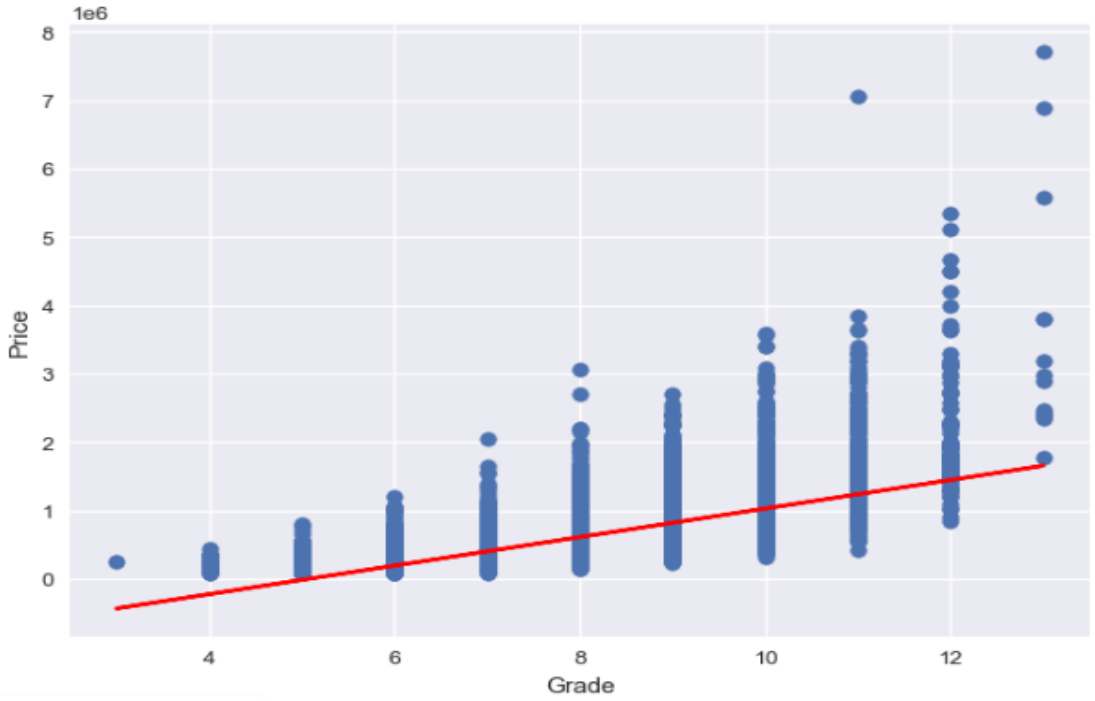


3

MODEL COMPARISON

Model 1
simple linear regression

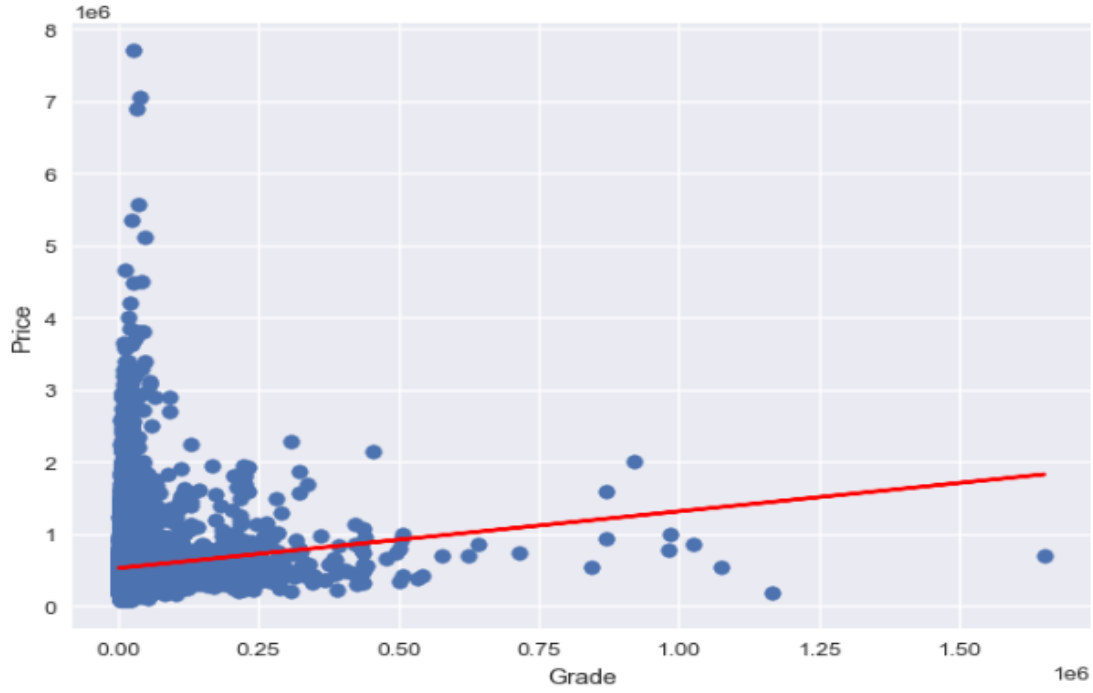
```
Predicted values:  [402945.12427493 402945.12427493 194190.19603059 ... 402945.12427493
 611700.05251926 402945.12427493]
R-squared:  0.44635643802432157
```



the R-squared value of 0.446 indicates that approximately 44.6% of the variance in the prices can be explained by the grade variable in the model.
its considered as a moderate fit of data

Model 2
Still on the first model; regression between price and sqft_lot

```
Predicted values:  [532823.82811173 534077.33169033 536248.91640754 ... 529438.1086469
 530255.40557817 529222.36745309]
R-squared:  0.007814472143692908
```



the R-squared value of 0.0078 indicates that only 0.78% of the variance in price can be explained by grade using the fitted linear regression model. This suggests that the model is not a good fit for the data, and there may be other variables that are better predictors of price.

Model 3

multiple linear regression using stats models

OLS Regression Results

Dep. Variable:	price	R-squared:	0.616
Model:	OLS	Adj. R-squared:	0.616
Method:	Least Squares	F-statistic:	2814.
Date:	Tue, 28 Mar 2023	Prob (F-statistic):	0.00
Time:	20:40:29	Log-Likelihood:	-2.8993e+05
No. Observations:	21081	AIC:	5.799e+05
Df Residuals:	21068	BIC:	5.800e+05
Df Model:	12		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	2.912e+07	3.26e+06	8.944	0.000	2.27e+07	3.55e+07
sqft_lot	0.0331	0.039	0.855	0.393	-0.043	0.109
bedrooms	-6873.4389	2119.204	-3.243	0.001	-1.1e+04	-2719.637
grade	1.846e+05	1895.620	97.377	0.000	1.81e+05	1.88e+05
bathrooms	1.014e+05	3534.639	28.679	0.000	9.44e+04	1.08e+05
floors	4.308e+04	4039.828	10.663	0.000	3.52e+04	5.1e+04
waterfront	6.25e+05	2.08e+04	30.026	0.000	5.84e+05	6.66e+05
condition	1.482e+04	2674.211	5.540	0.000	9573.966	2.01e+04
yr_built	-4055.3549	78.124	-51.909	0.000	-4208.484	-3902.226
yr_renovated	10.0624	4.547	2.213	0.027	1.150	18.975
zipcode	-227.8344	32.666	-6.975	0.000	-291.862	-163.806
view	5.394e+04	2383.576	22.628	0.000	4.93e+04	5.86e+04
sqft_basement	80.5698	4.488	17.953	0.000	71.773	89.366

Omnibus:	17784.410	Durbin-Watson:	1.974
Prob(Omnibus):	0.000	Jarque-Bera (JB):	1812631.165
Skew:	3.515	Prob(JB):	0.00
Kurtosis:	47.880	Cond. No.	2.07e+08

- R-squared value of 0.616 means that the model explains 61.6% of the variation in the dependent variable, which is moderate to strong. It suggests that the model has some predictive power, but there is still a significant amount of unexplained variation.

Model 4

split data into training and test sets

```
# Split the data into features and target variable
X = df_new.drop('price', axis=1)
y = df_new['price']

# Split the data into training and test sets (70% training, 30% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.3, random_state=42)
```

- Splitting data when doing regression is important because it helps to evaluate the performance of the model on new, unseen data. The general idea is to split the available data into two parts: a training set and a test set.

After splitting data;
we conduct data preparation so as to avoid data leakage.

- Log Transformation
- One Hot Encoding

Generate predictions

Evaluate and Validate Model

- Generate Predictions on Training and Test Sets

In [59]:
```
#generate predictions for both sets

train_preds = linreg.predict(X_train)
test_preds = linreg.predict(X_test)
# print R_squared score of model for both test and train
print(f'Training R_squared score: {linreg.score(X_train, y_train):.3f}')
print(f'Test R_squared score: {linreg.score(X_test, y_test):.3f}')
```

```
Training R_squared score: 0.673
Test R_squared score: 0.667
```

In [60]:
```
# Compute the MSE of the model's predictions on the training and test sets
train_mse = mean_squared_error(y_train, train_preds)
test_mse = mean_squared_error(y_test, test_preds)

# Print the MSEs of the model on the training and test sets
print(f"Training MSE: {train_mse:.3f}")
print(f"Test MSE: {test_mse:.3f}")
```

```
Training MSE: 44501691158.127
Test MSE: 43505670403.861
```

Overall, these metrics suggest that the model performs relatively well in predicting house prices, but there is still room for improvement. The test R-squared score is slightly lower than the training R-squared score, indicating that the model may be slightly overfitting to the training data. This could potentially be addressed by using a more complex model or collecting more data

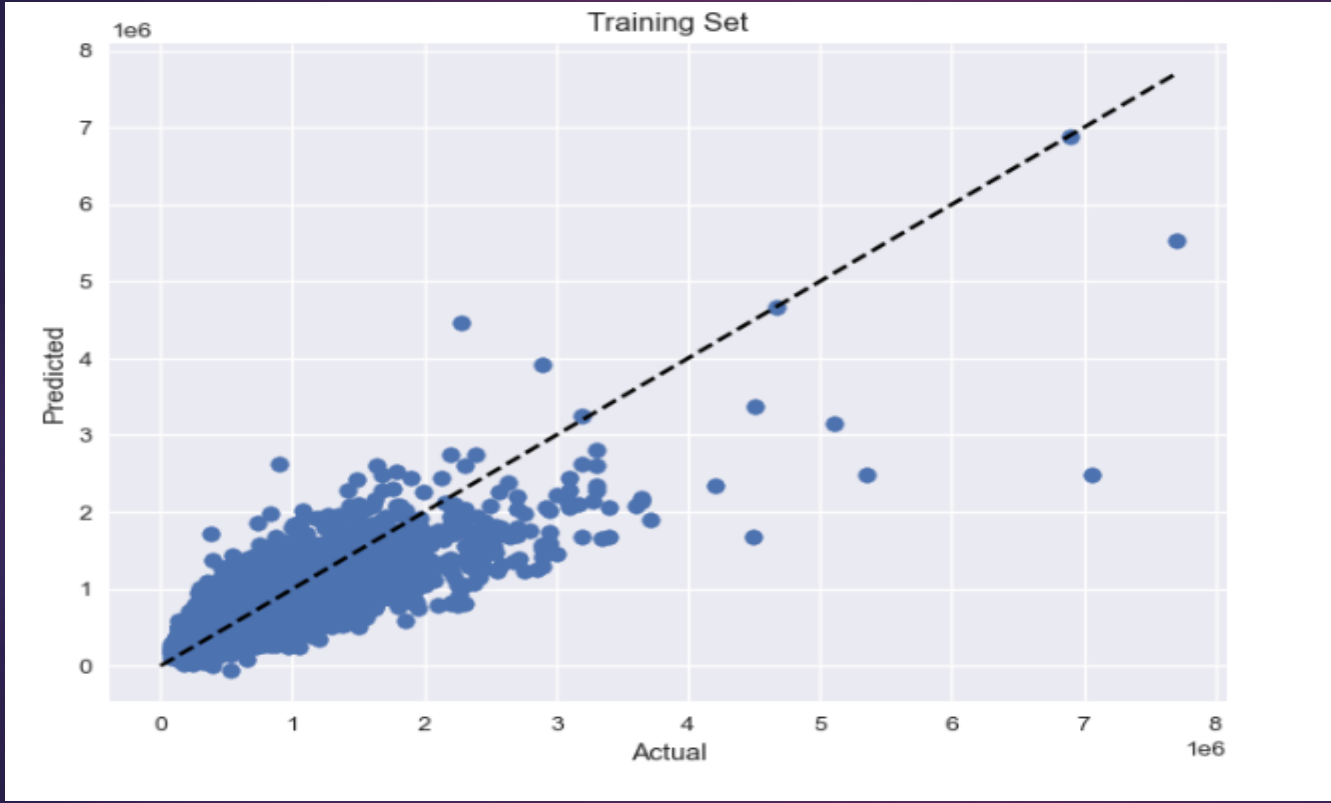
Conducting cross validation

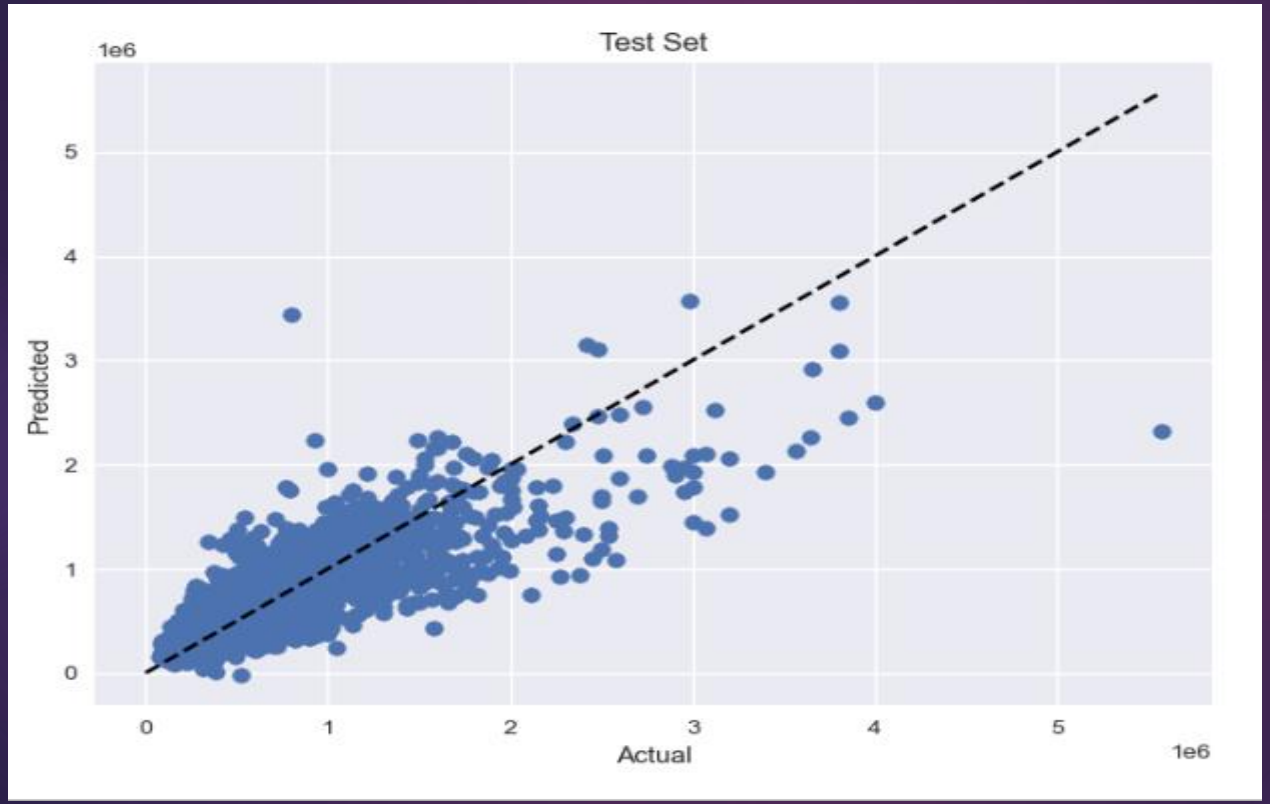
In [64]:
```
#import cross_val_score from sklearn .model_selection
from sklearn.model_selection import cross_val_score
# Perform cross-validation with 5 folds
scores = cross_val_score(linreg, X, y, cv=5)

# Print the cross-validation scores
print(f"Cross-validation scores: {scores}")
```

```
Cross-validation scores: [0.60854417 0.61015321 0.60240806 0.63451044 0.60514333]
```

cross-validation is an important technique in regression analysis because it helps to evaluate the performance of a model, choose the best model, and avoid overfitting.

Based on the cross-validation scores, it appears that the model's performance is consistent across multiple subsets of the data, which is a good indication that the model is not overfitting. However, the scores are not very high, suggesting that there may be room for improvement in the model's performance. It may be worth exploring additional features, trying different regression algorithms, or tuning the hyperparameters of the model to see if the performance can be improved. Overall, the model may be useful in predicting house prices in King County, but further analysis and improvements are recommended


Training Set
1e6
8
7
6
5
4
3
2
1
0
Predicted
0
1
2
3
4
5
6
7
8
Actual
1e6


Test Set
1e6
5
4
3
2
1
0
Predicted
0
1
2
3
4
5
Actual
1e6

4

KEY FINDINGS

findings

Regarding the regression models, we can see that model_4 with multiple independent variables has the highest R-squared value of 0.616, indicating that this model can explain around 61.6% of the variation in the dependent variable 'price'. The Adjusted R-squared value is also 0.616, indicating that the additional independent variables included in the model have not decreased its goodness of fit.

The coefficients of the independent variables in model_4 indicate that 'grade', 'sqft_living', 'bathrooms', 'view', 'sqft_above', 'sqft_living15', and 'waterfront' have a statistically significant impact on the house prices. A one-unit increase in the 'grade' variable is expected to increase the price of the house by 184,600 dollars, holding all other independent variables constant. Similarly, a one-unit increase in 'sqft_living' is expected to increase the price of the house by 109,000 dollars.

The RMSE values for model_4 are also relatively low, indicating that the model's predictions are close to the actual house prices. The cross-validation scores are also consistent, indicating that the model has good generalization performance.

Therefore, based on these findings, we can conclude that 'grade', 'sqft_living', 'bathrooms', 'view', 'sqft_above', 'sqft_living15', and 'waterfront' are important factors that impact house prices in King County. These findings can be useful for the stakeholder to make informed business decisions, such as pricing their properties more accurately and investing in the right locations

5 RECOMMENDATIONS

Focus on the location of the properties: As per the analysis, the 'zipcode' has a negative correlation with the house prices. Therefore, it is recommended to focus on properties located in desirable neighborhoods and zip codes that have higher demand.

Upgrade the property: The analysis shows that 'grade', 'bathrooms', 'view', 'sqft_above', 'sqft_living', and 'sqft_living15' are important factors that impact house prices. Therefore, it is recommended to upgrade the properties in terms of these features to increase their value and attract more buyers

Consider waterfront properties: The analysis shows that 'waterfront' properties have a positive impact on house prices. Therefore, it is recommended to invest in waterfront properties to increase the value of the properties.

Keep an eye on market trends: Real estate market trends can change quickly, so it is recommended to keep an eye on the market trends and adjust the business strategies accordingly. This can include analyzing the market demand for certain features and locations, and adjusting property prices and marketing strategies accordingly.

Use multiple regression models: The multiple regression model (model_4) provides the best predictions for the house prices and can explain around 61.6% of the variation in the dependent variable 'price'. Therefore, it is recommended to use this model for predicting house prices and gaining insights into the factors that affect house values in King County.

Overall, by following these recommendations, the stakeholder can make informed business decisions and increase their sales and revenue in the competitive real estate market of King County

6

POTENTIAL BENEFITS

Increased revenue: By investing in properties located in desirable neighborhoods and zip codes, upgrading the properties with desirable features, and focusing on waterfront properties, the stakeholder can increase the value of their properties and attract more buyers. This can lead to increased revenue and profits.

Improved accuracy in property pricing: By using the multiple regression model to predict house prices, the stakeholder can make more accurate property pricing decisions, leading to better negotiation and sales strategies.

Improved customer satisfaction: By upgrading the properties with desirable features, the stakeholder can improve customer satisfaction and attract more buyers and renters, leading to increased revenue and better long-term customer relationships.

Competitive advantage: By keeping an eye on market trends and adjusting business strategies accordingly, the stakeholder can gain a competitive advantage in the real estate market of King County and improve their market position.

improved decision-making: By gaining insights into the factors that affect house values in King County, the stakeholder can make more informed and data-driven business decisions, leading to better outcomes and improved business performance.

Overall, by following the recommendations, the stakeholder can benefit from increased revenue, improved customer satisfaction, a competitive advantage, and improved decision-making.