Linear Regression Model for House Price Prediction

```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
import matplotlib.pyplot as plt
import seaborn as sns

# Load the dataset
train_data = pd.read_csv('/mnt/data/train.csv')

# Select relevant features
features = ['GrLivArea', 'BedroomAbvGr', 'FullBath']
X = train_data[features]
y = train_data['SalePrice']

# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create and train the model
model = LinearRegression()
model.fit(X_train, y_train)

# Make predictions
```

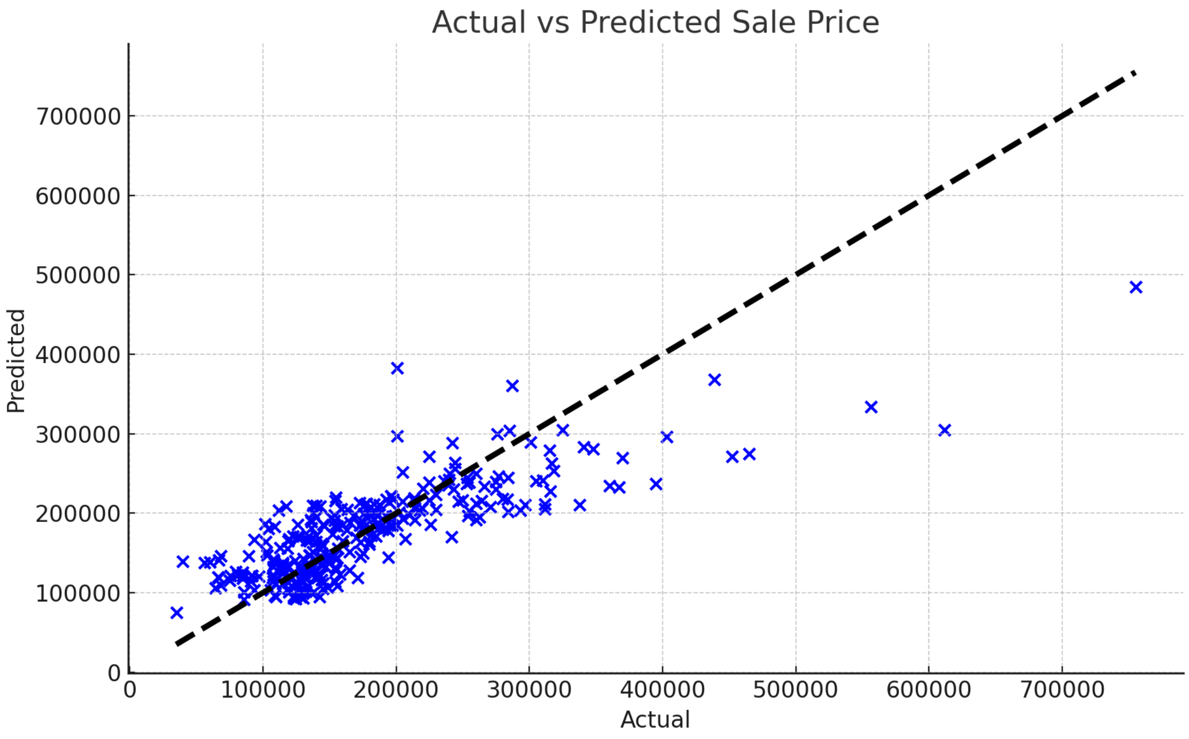
```
y_pred = model.predict(X_test)



# Evaluate the model

mse = mean_squared_error(y_test, y_pred)

r2 = r2_score(y_test, y_pred)



# Visualize the results

plt.figure(figsize=(10, 6))

plt.scatter(y_test, y_pred, color='blue')

plt.plot([y.min(), y.max()], [y.min(), y.max()], 'k--', lw=3)

plt.xlabel('Actual')

plt.ylabel('Predicted')

plt.title('Actual vs Predicted Sale Price')

plt.show()
```

Model Interpretation

Linear Regression Model Evaluation:

- Mean Squared Error (MSE): 2806426667.247853
- R-squared (R^2) Score: 0.6341189942328371

The R^2 score indicates that 63.41% of the variance in the sale price can be explained by the model using the given features. The scatter plot visualizes the relationship between the actual and predicted sale prices, showing how well the model's predictions align with the actual values.


Actual vs Predicted Sale Price
Predicted
700000
600000
500000
400000
300000
200000
100000
0
0
100000
200000
300000
400000
500000
600000
700000
Actual