HOUSING PRICES PREDICTION

Introduction

The goal of this project is to build a model to predict house prices. The dataset contains information about houses sold between 2006 and 2010 in Ames, Iowa, USA.

This study included the following steps:

- Data cleaning
- Performing Exploratory Data Analysis
- Building a Multiple Linear Regression model
- Using K-Nearest Neighbors for prediction and feature engineering

About the Data

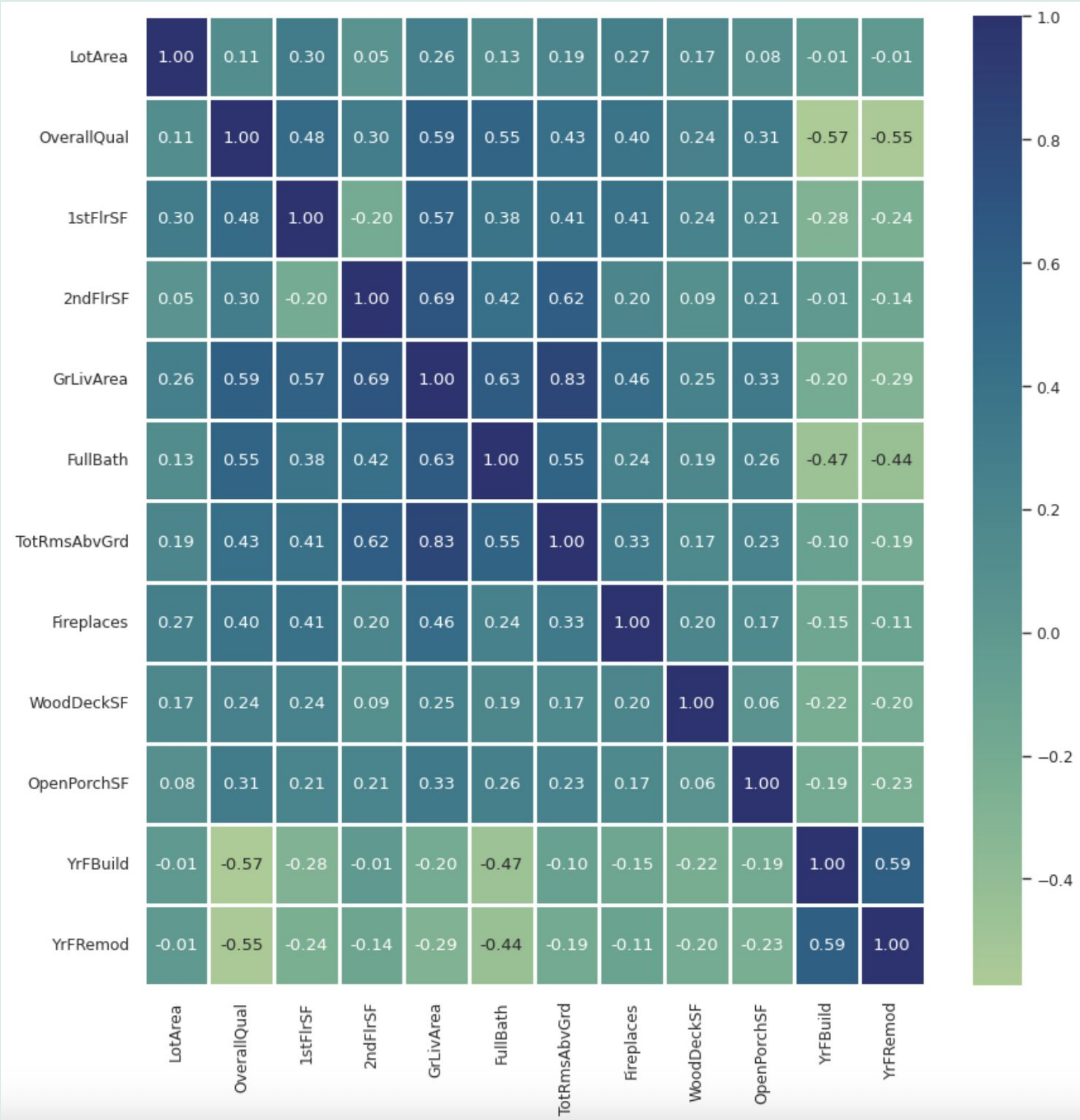
Initially, the dimensions of the training set were (1460, 81), and the dimensions of the test set were (1459, 80). After dropping columns and rows with missing values, the dimensions of both datasets became(1459, 47).

Of the 47 features, 26 are int64 and 21 are object type.

The target variable 'SalePrice' is int64.

I encourage you to familiarize yourself with the features using the data story in Tableau Public. This data story explores features related to size of property, location, overall condition and quality, among others, and their relationship to housing prices. Check out the link in post.

The following correlation matrix represents a subset of features that are highly or moderately correlated with SalePrice.



Multiple Linear Regression

Feature Engineering for Linear Regression

To use the categorical variables for prediction, I used target encoding. Target encoding involves replacing a category with the mean or median of the target variable for that category. The median was chosen as it is more robust to outliers. I applied this technique to the following variables: 'Neighborhood', 'BldgType', 'HouseStyle', 'SaleCondition'.

Model Summary

OLS Regression Results

Dep. Variable:	SalePrice	**R-squared:**	0.796
Model:	OLS	**Adj. R-squared:**	0.794
Method:	Least Squares	**F-statistic:**	421.6
Date:	Wed, 24 Sep 2025	**Prob (F-statistic):**	0.00
Time:	10:29:49	**Log-Likelihood:**	-13021.
No. Observations:	1094	**AIC:**	2.606e+04
Df Residuals:	1083	**BIC:**	2.612e+04
Df Model:	10		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1.812e+05	2.08e+04	-8.726	0.000	-2.22e+05	-1.4e+05
LotArea	0.4376	0.112	3.922	0.000	0.219	0.656
OverallQual	1.518e+04	1349.876	11.243	0.000	1.25e+04	1.78e+04
FirstFlrSF	64.1917	3.930	16.333	0.000	56.480	71.903
SecondFlrSF	40.7902	3.032	13.451	0.000	34.840	46.740
Fireplaces	6669.3847	2011.863	3.315	0.001	2721.794	1.06e+04
WoodDeckSF	38.9913	9.176	4.249	0.000	20.986	56.996
YrFRemod	-197.6277	66.561	-2.969	0.003	-328.231	-67.024
MedianPriceNeighborhood	0.4042	0.029	14.059	0.000	0.348	0.461
MedianPriceBldgType	0.3432	0.118	2.910	0.004	0.112	0.575
MedianPriceSaleCondition	0.2786	0.045	6.182	0.000	0.190	0.367

Omnibus:	309.161	**Durbin-Watson:**	1.938
Prob(Omnibus):	0.000	**Jarque-Bera (JB):**	26335.398
Skew:	0.150	**Prob(JB):**	0.00
Kurtosis:	27.034	**Cond. No.**	5.62e+06

Different sets of features were used to build the model. This summary represents the linear regression results with the final selected features.

The adjusted R-squared of the model is 0.794, indicating that the independent variables explain 79.4% of the variability in SalePrice.

The p-value for all coefficients is less than 0.05, meaning all coefficients are statistically significant at the p=0.05 level.

Variance Inflation Factor as an Indicator of Multicollinearity

Variance Inflation Factor (VIF) quantifies how much each variable's variance is "inflated" by correlations with other variables.

The smallest value a VIF can take is 1, which indicates no correlation between the variable in question and the other predictor variables in the model. A high VIF (5 or higher), according to the *statsmodels* documentation, can indicate the presence of multicollinearity.

	VIF
LotArea	2.474236
OverallQual	58.444088
FirstFlrSF	19.198118
SecondFlrSF	2.315138
Fireplaces	2.632816
WoodDeckSF	1.774867
YrFRemod	3.263983
MedianPriceNeighborhood	23.462997
MedianPriceBldgType	66.677079
MedianPriceSaleCondition	43.439326

Unfortunately, the VIF is significantly higher than 5 for several variables, meaning the coefficients cannot be used to understand the individual impact of each predictor due to multicollinearity.

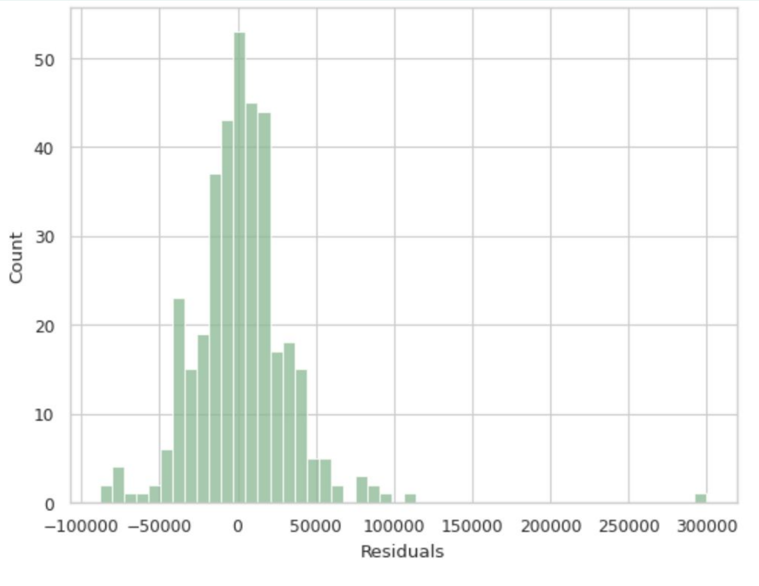
However, multicollinearity does not affect the model's overall predictive power or the accuracy of the predictions. The issue is with attributing that predictive power to a specific variable.

Model Evaluation on Validation Data

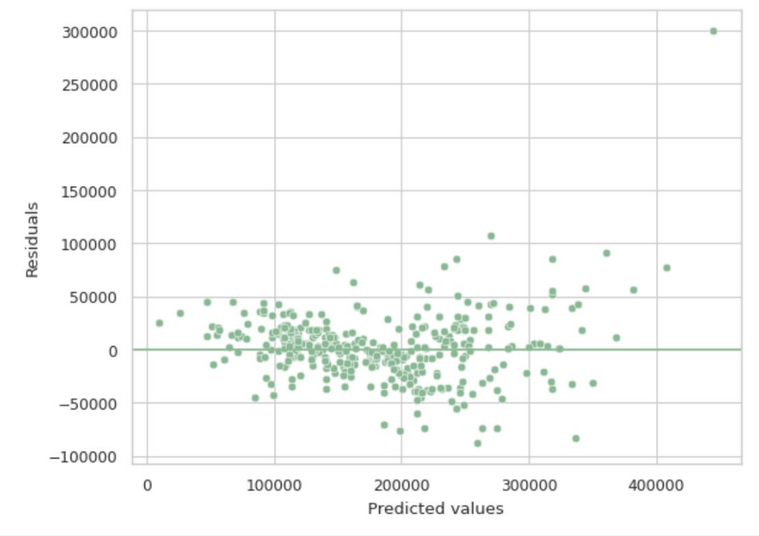
The explained variance score on the validation data is 0.8438, which means that 84.38% of the variance in SalePrice is explained by the model.

Mean Absolute Error (MAE) is 21605.07.

Root Mean Squared Error (RMSE) is 32011.56.



As shown in the histogram, the residuals are nearly normally distributed.



Residuals appear to be randomly scattered around zero without any systematic pattern, so the assumption of homoscedasticity is also met.

Conclusion on Multiple Linear Regression

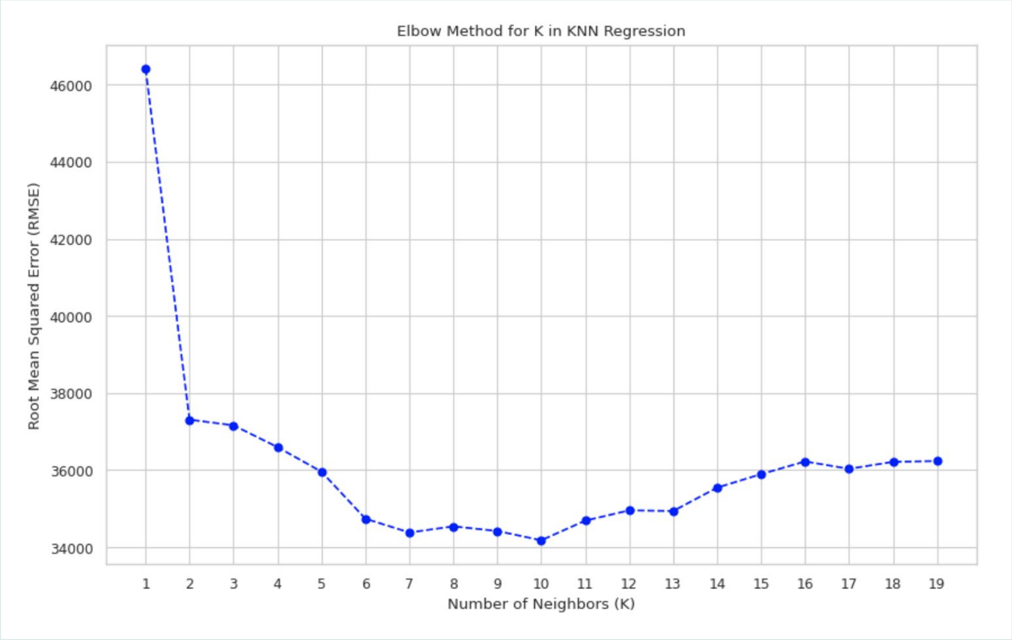
The built model has a good fit, as indicated by an explained variance score of 0.8438 on the validation data. The model meets assumptions of normality and homoscedasticity, and the observations are independent. However, there is multicollinearity between the predictive variables, which makes this model unsuitable for explaining the individual contribution of each variable.

K-Nearest Neighbors (KNN)

For Prediction

KNN is relatively simple prediction technique: for prediction, it takes the average of K records with similar predictor values. Similarity is determined using a distance metric; therefore, prediction results depend highly on how the features are scaled.

The Elbow method was used for choosing the number of neighbors.



On validation data, KNN as a prediction technique showed the following results:
Root Mean Squared Error (RMSE): 34180.18
R-squared: 0.82

K-Nearest Neighbors for Feature Engineering

The average of K-Nearest Neighbors can be used as a predictor variable for second-stage (non-KNN) modeling. In terms of multicollinearity, there can be concern about using some predictors twice. This is not an issue, since the information incorporated in KNN predictions is highly local, derived only from a few nearby records.

OLS Regression Results

Dep. Variable:	SalePrice	R-squared:	0.843
Model:	OLS	Adj. R-squared:	0.842
Method:	Least Squares	F-statistic:	832.0
Date:	Mon, 20 Oct 2025	Prob (F-statistic):	0.00
Time:	11:58:14	Log-Likelihood:	-12877.
No. Observations:	1094	AIC:	2.577e+04
Df Residuals:	1086	BIC:	2.581e+04
Df Model:	7		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
Intercept	-1.149e+05	1.76e+04	-6.530	0.000	-1.49e+05	-8.04e+04
OverallQual	3763.9554	1275.662	2.951	0.003	1260.914	6266.997
FirstFlrSF	21.7640	4.079	5.335	0.000	13.760	29.768
SecondFlrSF	17.7980	2.914	6.108	0.000	12.080	23.516
KNNpred	0.8000	0.040	20.196	0.000	0.722	0.878
MedianPriceNeighborhood	0.1909	0.027	6.967	0.000	0.137	0.245
MedianPriceBldgType	0.2240	0.102	2.187	0.029	0.023	0.425
MedianPriceSaleCondition	0.1757	0.039	4.500	0.000	0.099	0.252

Omnibus:	379.799	Durbin-Watson:	1.936
Prob(Omnibus):	0.000	Jarque-Bera (JB):	13269.972
Skew:	0.917	Prob(JB):	0.00
Kurtosis:	19.963	Cond. No.	6.41e+06

The Final Model Summary

On validation data, this model showed the following results:
Root Mean Squared Error (RMSE): 30551.03
R-squared: 0.86

On testing data, Root Mean Squared Error is 21303.97 (Kaggle submission).

Conclusion

Overall, using KNN predictions as a feature slightly improved results of multiple linear regression model while using fewer original predictor variables.
Since there are outliers in the data, I suggest also trying models such as Decision Tree and Random Forest, which are more robust to outliers.