PROPERTY PRICING IN AMES, IOWA

HOUSING PRICE PREDICTION MODEL

DATA SCIENCE TRAINEE

CREATED BY:

ALESSANDRO GATTI

Problem Statement (Hypothesis Formation)

How can I develop a machine learning model to predict house sale prices with an accuracy that improves RMSE by at least 5% compared to baseline models, while minimizing overfitting, using the Ames Housing dataset?

H

1 Context

The housing market is highly competitive and dynamic, and accurately predicting the sale price of a property is critical for real estate businesses. This project focuses on creating a robust predictive model for house sale prices using the Ames dataset. Through different machine learning techniques, the model aims to minimize prediction error while avoiding overfitting, which has been a challenge due to high R-squared values, especially when combining models.

2 Criteria for success

- **Success in this project is defined as achieving the lowest RMSE on the test dataset while maintaining generalizability to unseen data. Specifically, this will be done by balancing predictive performance and reducing overfitting risks using cross-validation and residual analysis.**

3 Scope of solution space

- **This project will focus on supervised learning regression models, leveraging feature selection and dimensionality reduction techniques like PCA to enhance the performance of predictive models.**
- **Various tuning methods, including grid search and cross-validation, will be used to optimize model parameters.**

4 Constraints within solution space

- **The solution will be limited by the available data from the Ames Housing dataset, particularly in terms of missing data, outliers, and the inherent complexity of predicting real estate prices.**
- **Constraints include computing resources, the limitations of the machine learning models used, and potential overfitting with complex models like Random Forest.**

5 Stakeholders to provide key insight

- **Data Scientist (myself)**
- **Real Estate Analysts (could be useful for specific insights)**

6 Key data sources

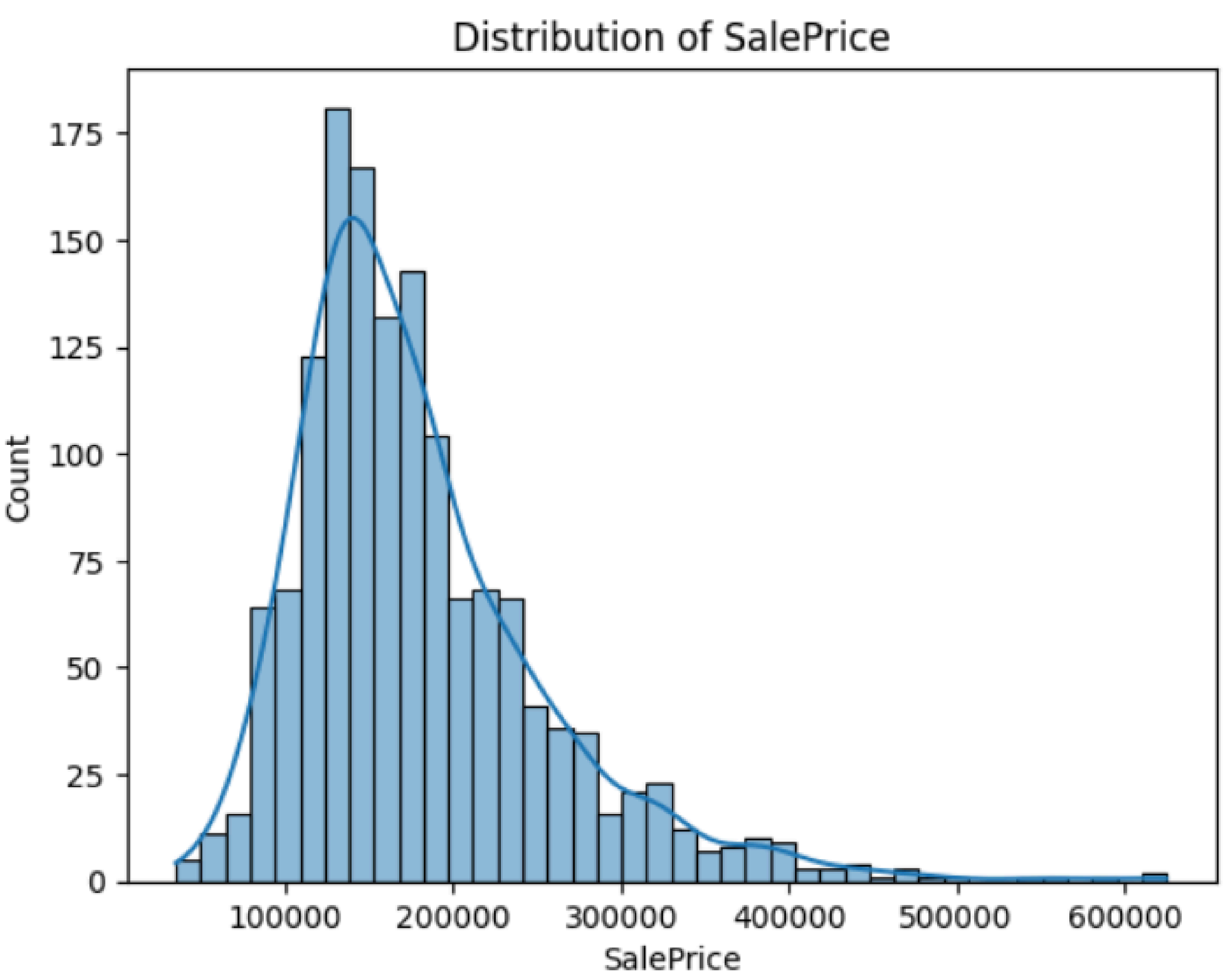
- **CSV file –** Ames Housing Dataset from Kaggle, contains information on various property features such as square footage, neighborhood quality, and house type. It will be the primary source of data for this analysis.

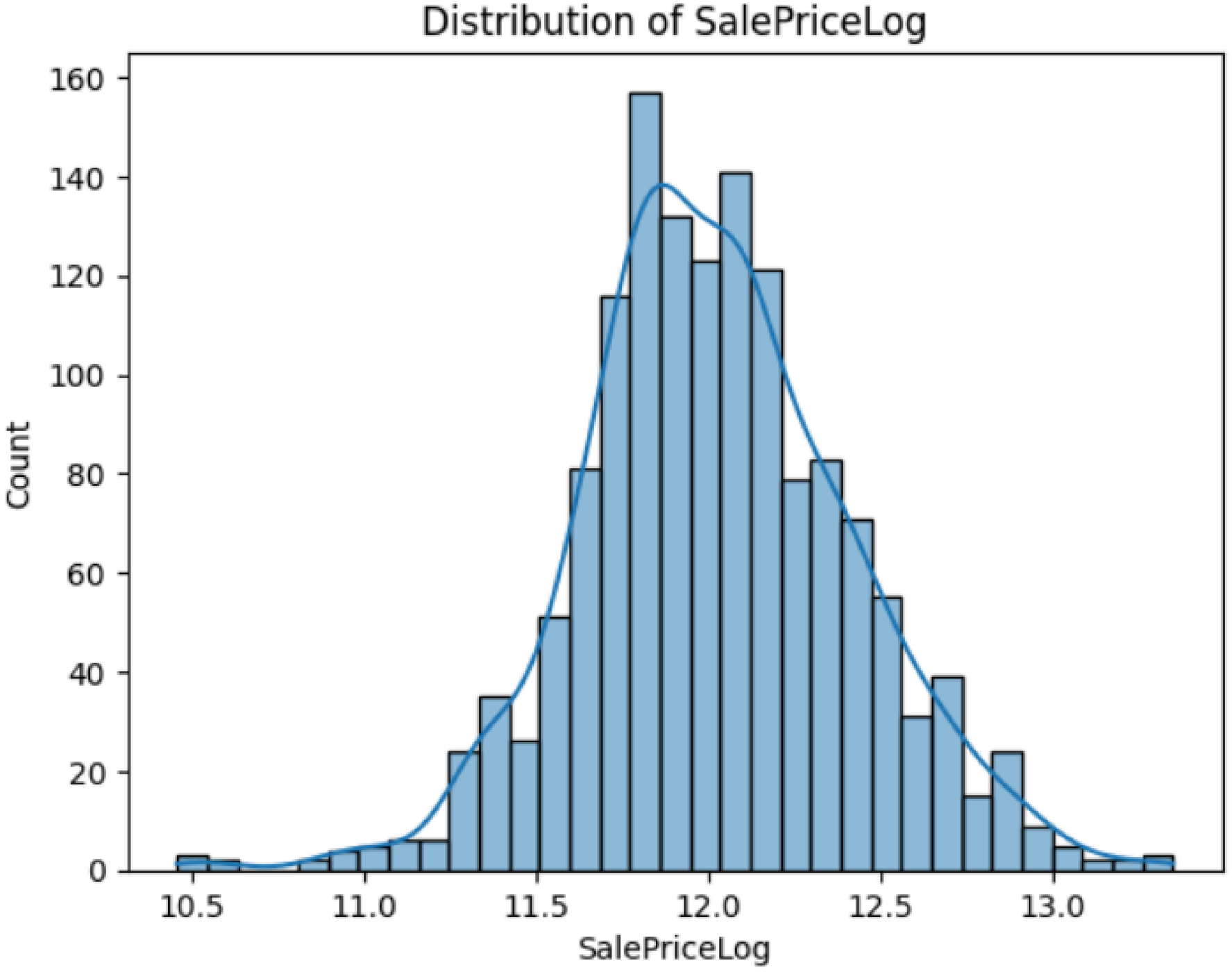
Data Overview - train.csv

	Id	MSSubClass	MSZoning	LotFrontage	LotArea	Street	Alley	LotShape	LandContour	Utilities	...	PoolArea	PoolQC	Fence
0	1	60	RL	65.0	8450	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN
1	2	20	RL	80.0	9600	Pave	NaN	Reg	Lvl	AllPub	...	0	NaN	NaN
2	3	60	RL	68.0	11250	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN
3	4	70	RL	60.0	9550	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN
4	5	60	RL	84.0	14260	Pave	NaN	IR1	Lvl	AllPub	...	0	NaN	NaN

5 rows × 81 columns

MiscFeature	MiscVal	MoSold	YrSold	SaleType	SaleCondition	SalePrice
NaN	0	2	2008	WD	Normal	208500
NaN	0	5	2007	WD	Normal	181500
NaN	0	9	2008	WD	Normal	223500
NaN	0	2	2006	WD	Abnorml	140000
NaN	0	12	2008	WD	Normal	250000


Distribution of SalePrice
Count
SalePrice
0
25
50
75
100
125
150
175
100000
200000
300000
400000
500000
600000


Distribution of SalePriceLog
Count
SalePriceLog
0
20
40
60
80
100
120
140
160
10.5
11.0
11.5
12.0
12.5
13.0

Feature Engineering

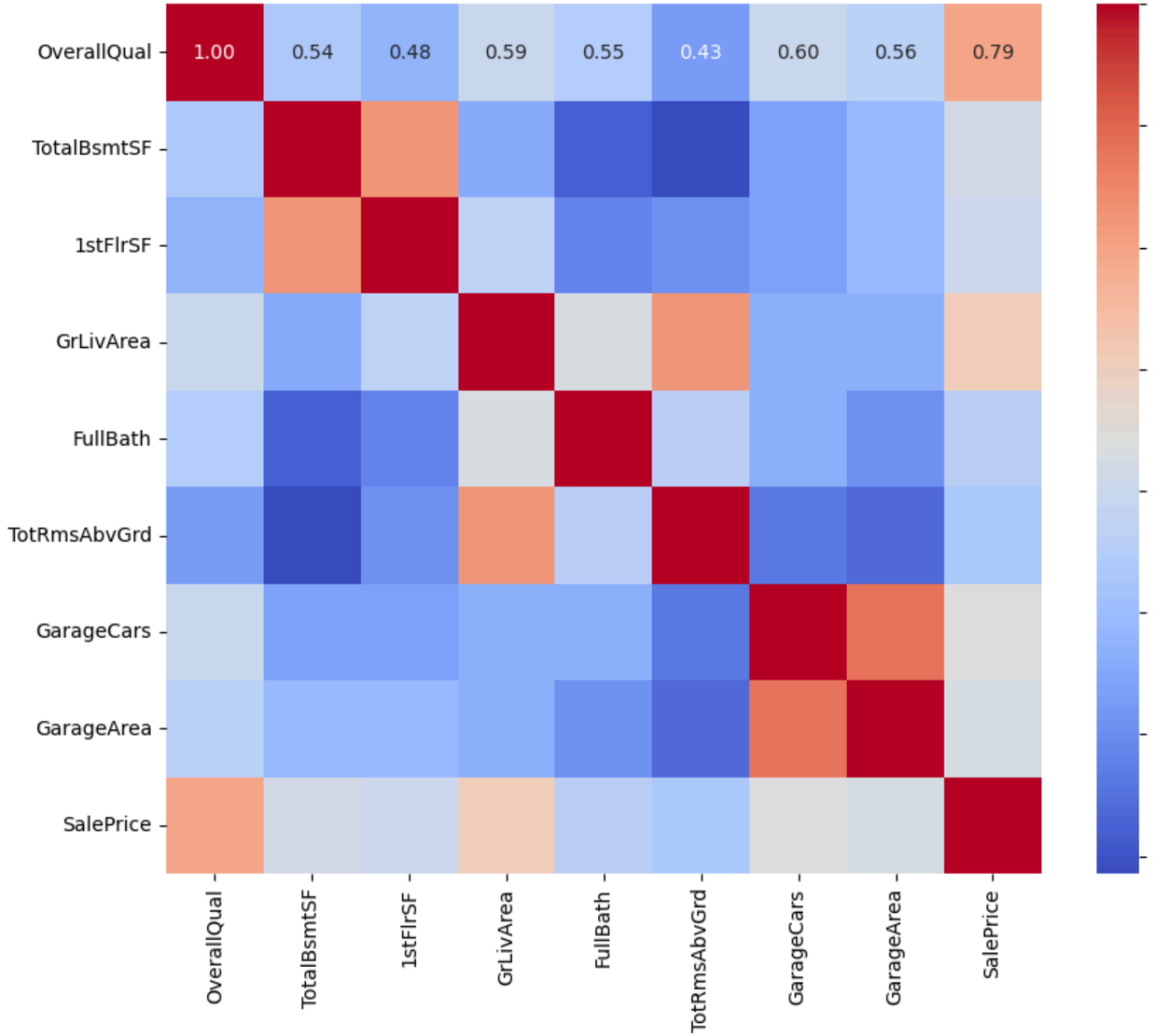
#creating new features that represents existing data
train_data['TotalSF'] = train_data['TotalBsmtSF'] + train_data['1stFlrSF'] + train_data['2ndFlrSF']
train_data['TotalFinishedSF'] = train_data['BsmtFinSF1'] + train_data['BsmtFinSF2'] + train_data['1stFlrSF'] + train_data['2ndFlrSF']
train_data['HouseAge'] = train_data['YrSold'] - train_data['YearBuilt']train_data['RemodelAge'] = train_data['YrSold'] - train_data['YearRemodAdd']
train_data['OverallQualCond'] = train_data['OverallQual'] * train_data['OverallCond']
train_data['TotalPorchSF'] = train_data['OpenPorchSF'] + train_data['EnclosedPorch'] + train_data['3SsnPorch'] + train_data['ScreenPorch']
train_data['TotalGarageSize'] = train_data['GarageArea'] + train_data['GarageCars']
train_data['TotalBsmtSF'] = train_data['BsmtFinSF1'] + train_data['BsmtFinSF2'] + train_data['BsmtUnfSF']
train_data['OverallGrade'] = train_data['OverallQual'] * train_data['YearBuilt']
train_data['BathsPerBed'] = (train_data['FullBath'] + 0.5 * train_data['HalfBath']) / train_data['BedroomAbvGr']
train_data['LotFrontageToLotArea'] = train_data['LotFrontage'] / train_data['LotArea']

#neighborhood pricing feature
neighborhood_median_price = train_data.groupby('Neighborhood')['SalePrice'].median()
train_data['NeighborhoodMedianPrice'] = train_data['Neighborhood'].map(neighborhood_median_price)

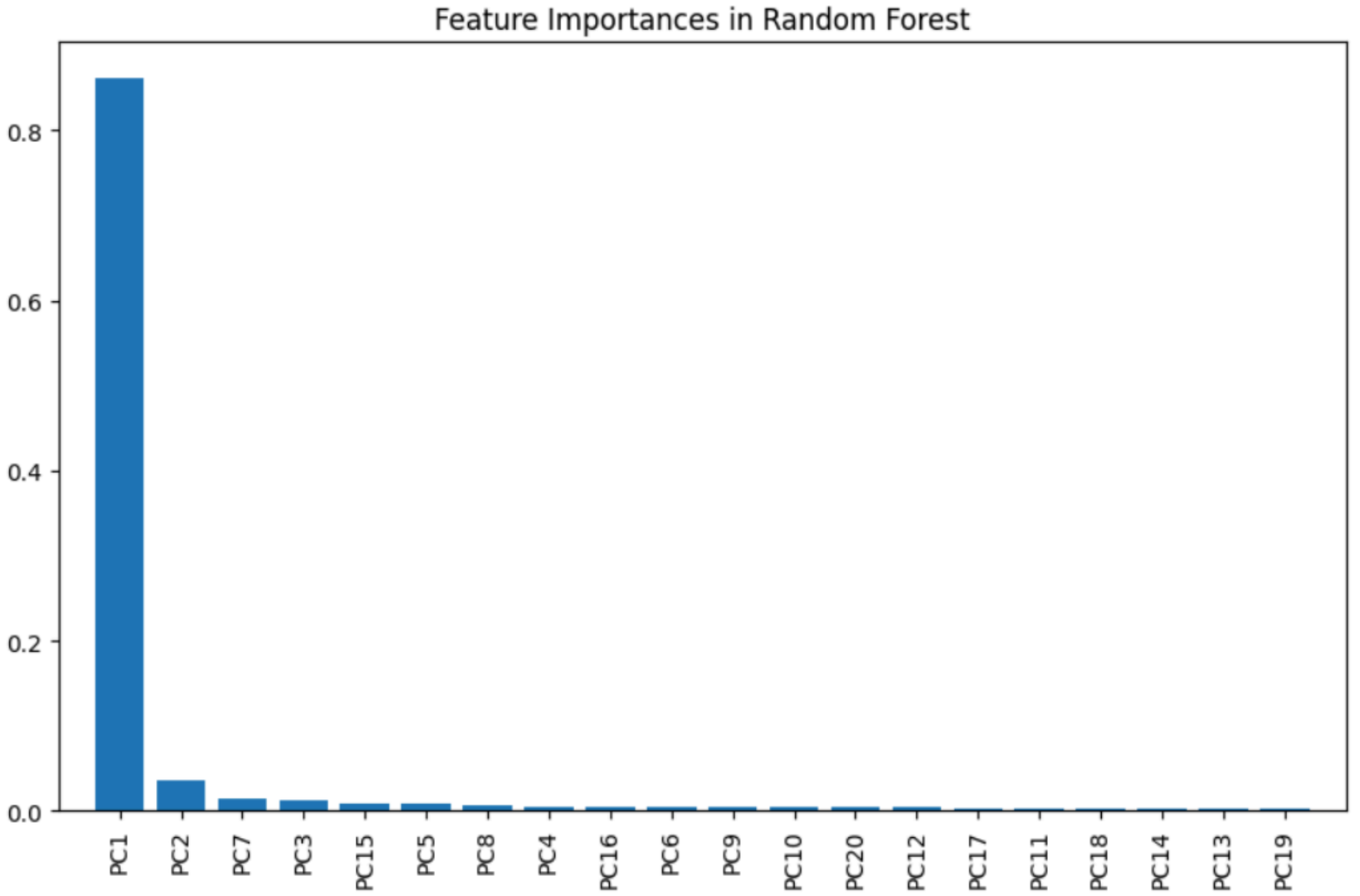
PREPROCESSING

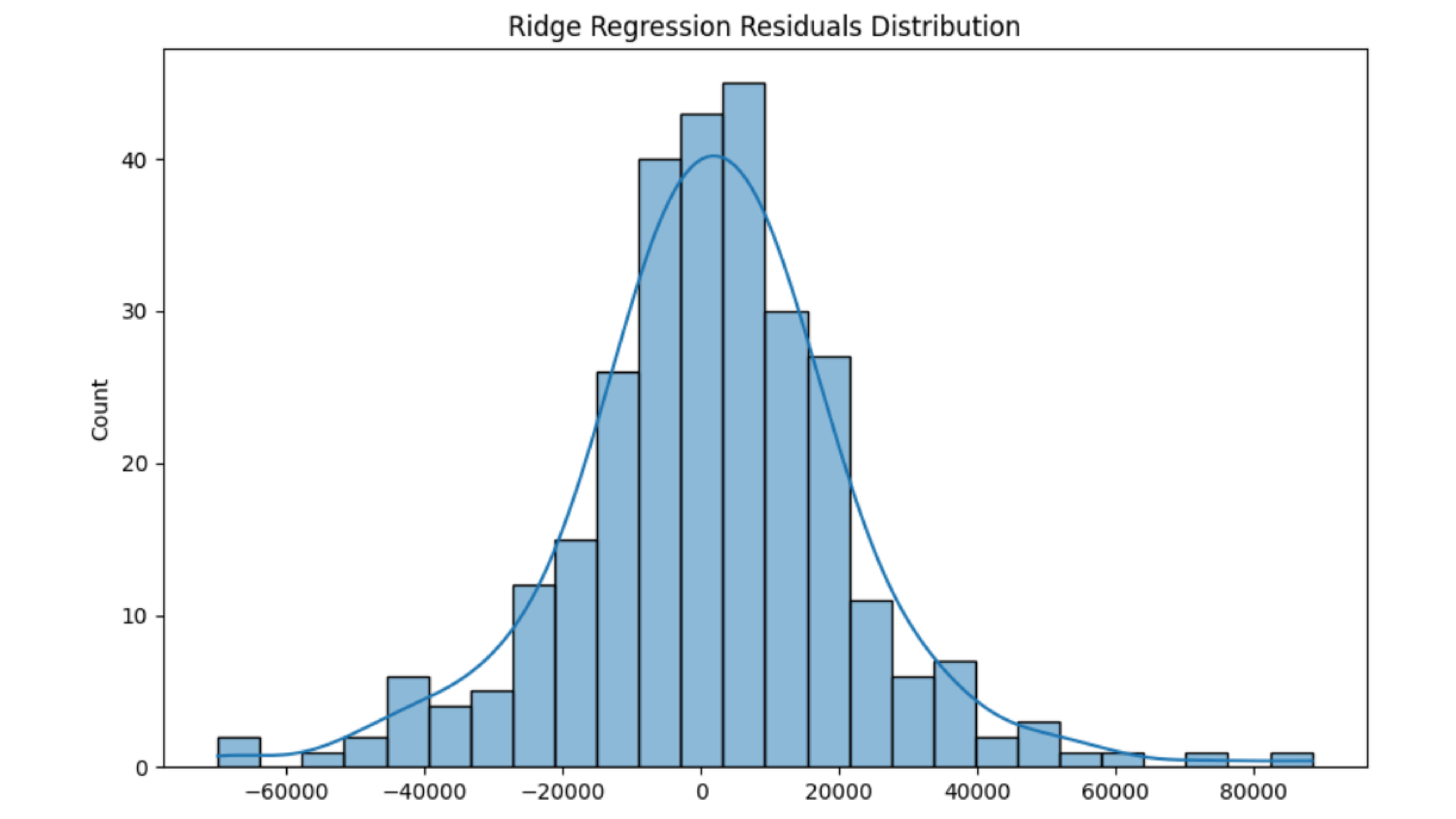
Scaler:	StandardScaler()
Correlation Threshold:	0.20
Variance Threshold:	0.01
Blended Model Weights:	Ridge:57%, RF:43%

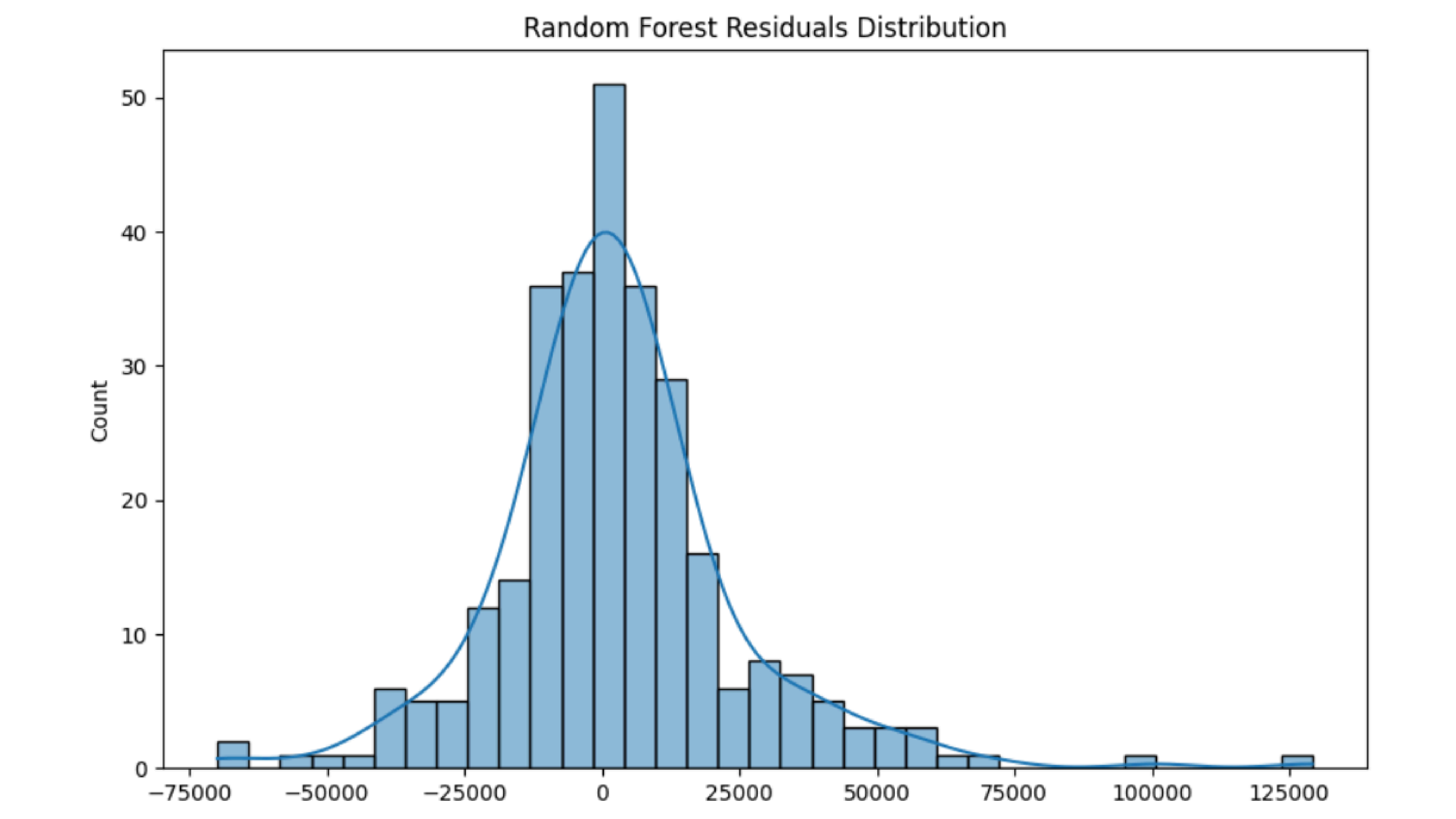
KEY FEATURES



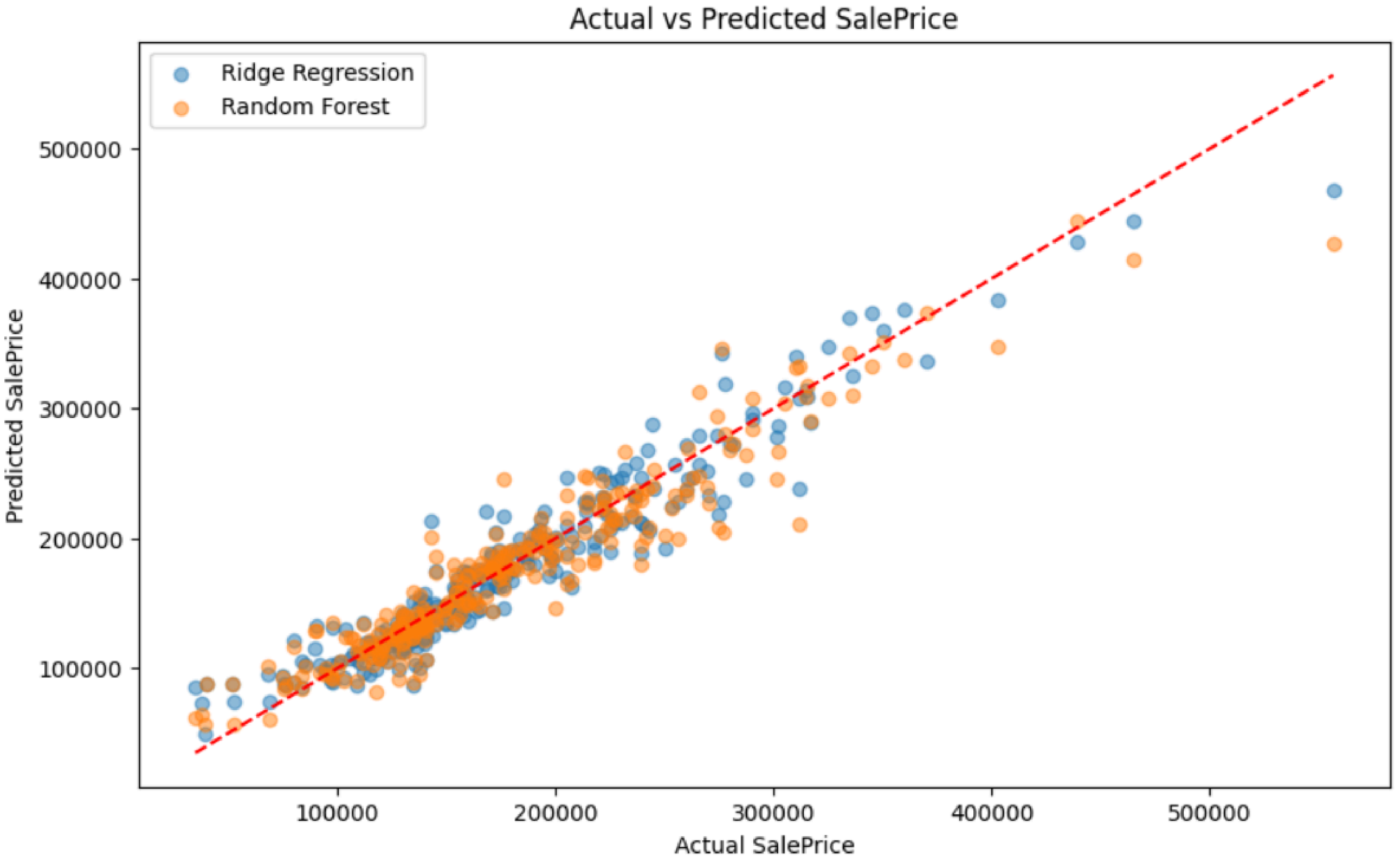
MODEL EVALUATION & DIM. REDUCTION

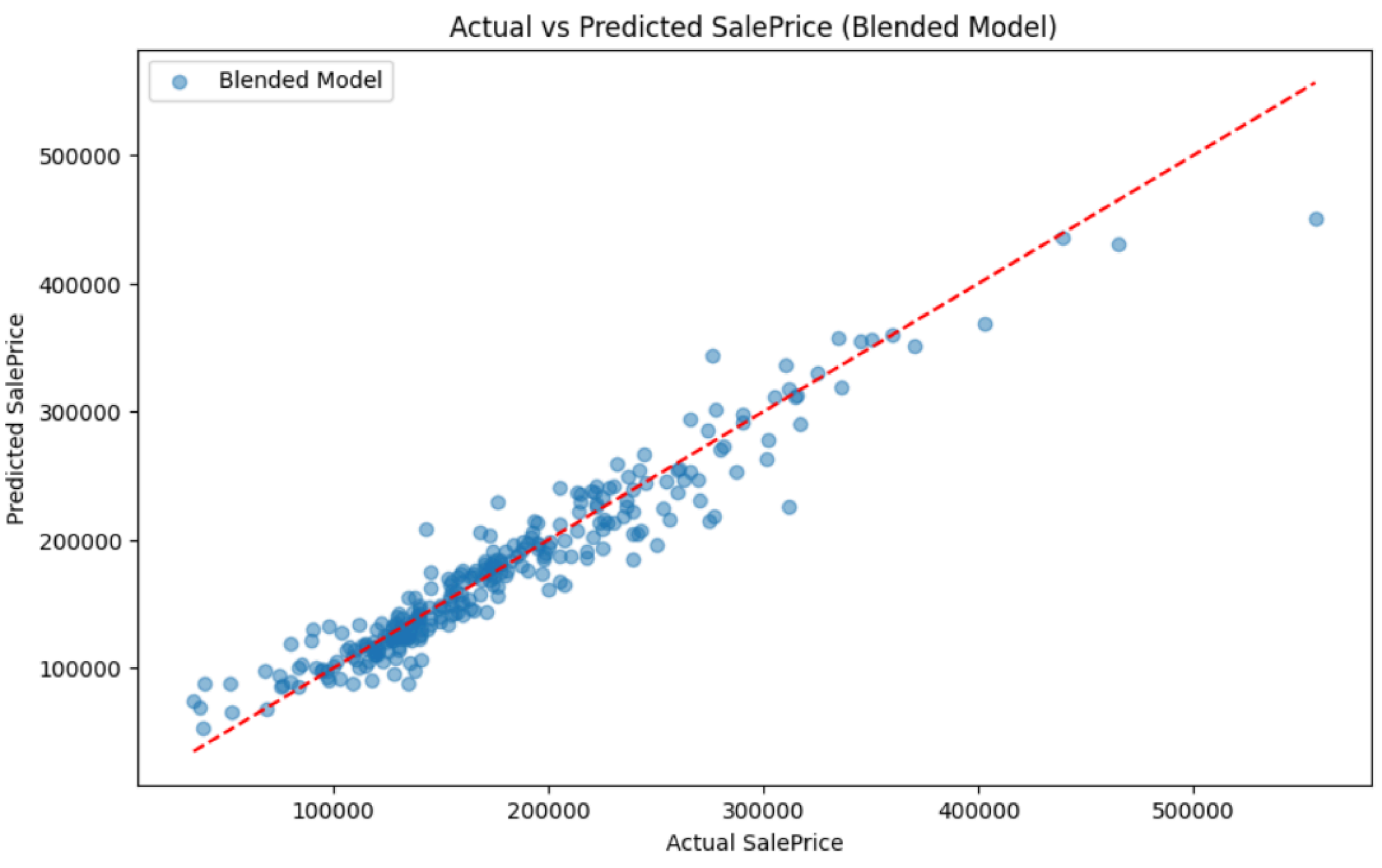


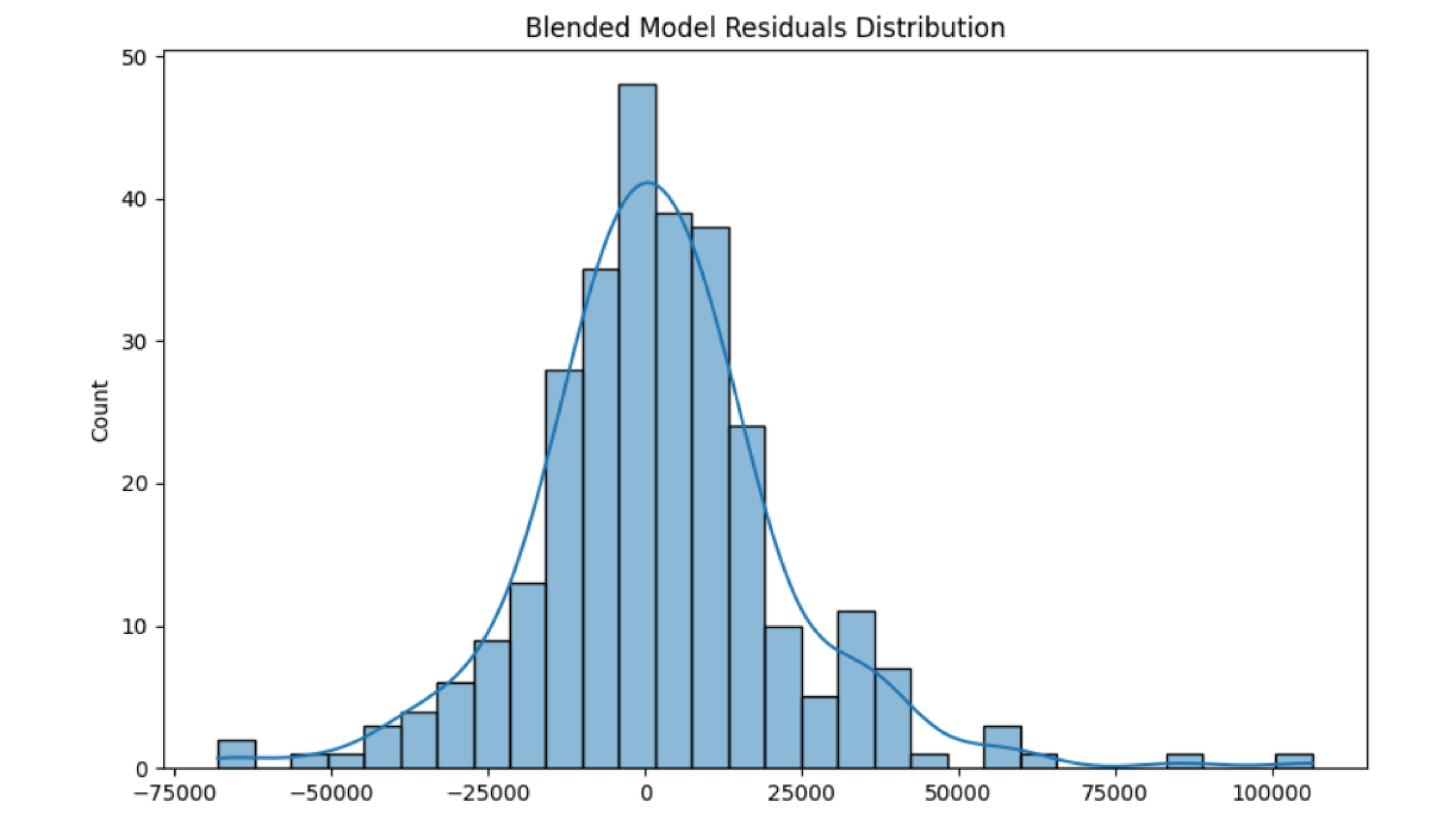

Ridge Regression Residuals Distribution
Count
0
10
20
30
40
−60000
−40000
−20000
0
20000
40000
60000
80000


Random Forest Residuals Distribution
Count
0
10
20
30
40
50
−75000
−50000
−25000
0
25000
50000
75000
100000
125000

Model R-Squared and RMSE Values		
Model	RMSE	R-Squared
Ridge	20452.6866	0.9210
Random Forest	22600.4205	0.9035
Blended	20446.5387	0.9210


Actual vs Predicted SalePrice
Ridge Regression
Random Forest
Predicted SalePrice
Actual SalePrice
100000
200000
300000
400000
500000


Actual vs Predicted SalePrice (Blended Model)
Blended Model
Predicted SalePrice
Actual SalePrice
100000
200000
300000
400000
500000


Blended Model Residuals Distribution
Count
50
40
30
20
10
0
−75000
−50000
−25000
0
25000
50000
75000
100000

CONCLUSIONS & KEY TAKEAWAYS

The blended model combining Ridge Regression and Random Forest achieved an improved predictive accuracy compared to individual models, with an RMSE of approximately 20,446.5. While this DID NOT meet the full 5% reduction target, it demonstrated the effectiveness of blending models to leverage complementary strengths, particularly balancing bias and variance.

Transforming the SalePrice variable and selecting features based on correlation and variance were instrumental in enhancing model performance. The log transformation normalized skewness, improving predictive accuracy, while feature importance analysis highlighted key contributors to sale price prediction, such as overall quality and living area.

Overfitting remains a consideration, especially in the Random Forest component of the blended model, as indicated by residual distributions and R-squared values. Future iterations could focus on tuning model parameters further or exploring additional regularization techniques to refine generalization and potentially meet the 5% RMSE improvement target.