Predicting NYC Property Prices

GOALS

Accurately predict price by training a model with a NYC property dataset

Build a series of models for practice and develop an understanding of which model works best and why

METHODOLOGY

Data Cleaning

- Removed irrelevant and redundant columns such as address, street name, long name, and formatted address
- Removed outliers based on box plots

Data Preparation

- Removed categorical variables and replaced with dummy variables
- Added neighborhoods based on zip codes using a separate dataset
- Created our baseline model which had an RMSE of 1,457,868.74 (average price for each instance)

Modeling

- **Scaled** and **divided** the data into testing and training sets
- Compared and tuned the RMSE accuracy of different models, including:

Linear Regression
Lasso and Ridge
kNN
Decision Tree (Regression and Classification)
Random Forest
Boosting
Bagging
Neural network

FEATURE ENGINEERING & EXPLORATORY ANALYSIS

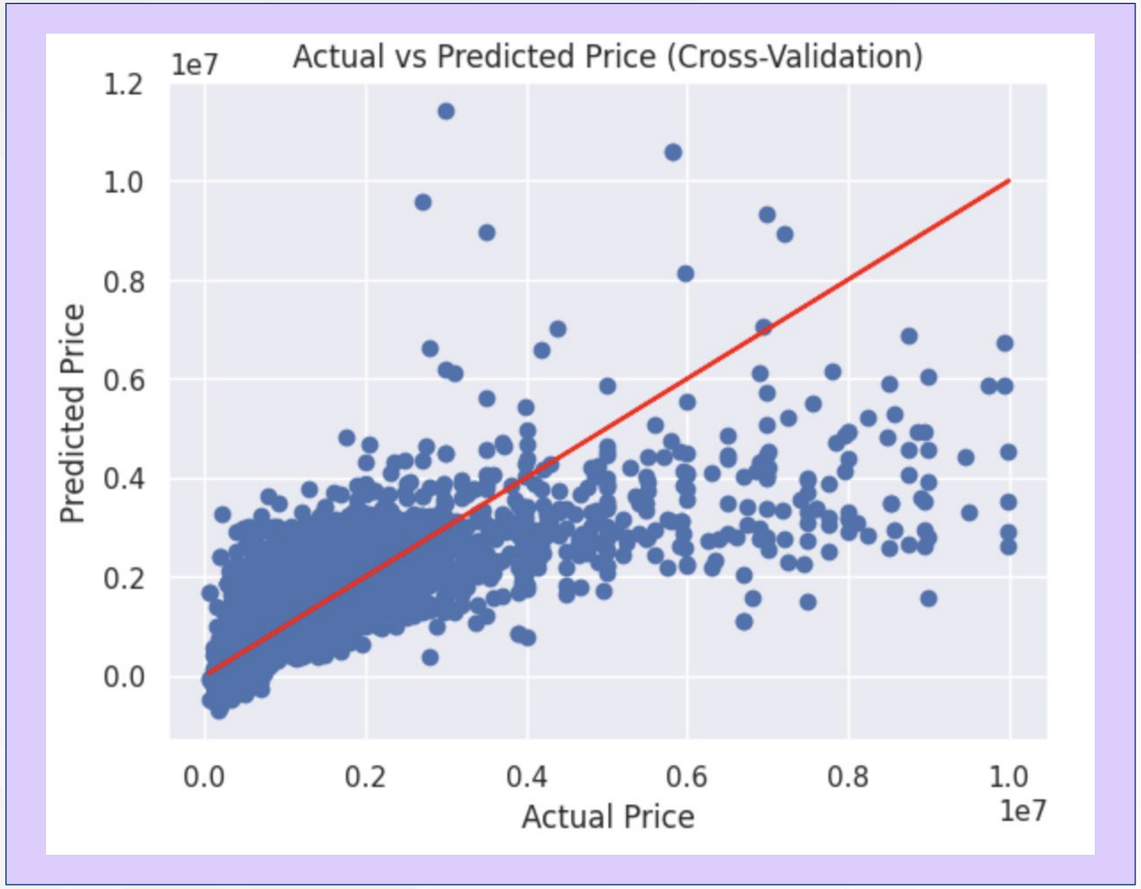
- LOG TRANSFORMATIONS
- COMBINED NYC NEIGHBORHOOD DATASET
- CREATED COLUMN: PRICE_PERCENTILE
- 16 PREDICTORS → 52 PREDICTORS

MOST $ NEIGHBORHOOD	AVG PRICE PER SQFT
Greenwich Village & Soho	$1,447.75
Lower Manhattan	$1,444.35
Chelsea & Clinton	$1,372.16
LEAST $ NEIGHBORHOOD	**AVG PRICE PER SQFT**
Hunts Point & Mott Haven	$305.33
High Bridge & Morrisania	$270.06
Bronx Park & Fordham	$222.62

NYC IS EXPENSIVE!!!

BASELINE RMSE: 1,457,868.74
(Naive model)

LINEAR REGRESSION AND RIDGE WITH CV



LINEAR REGRESSION WITH NESTED CV

Mean RMSE: 1,011,300

RIDGE

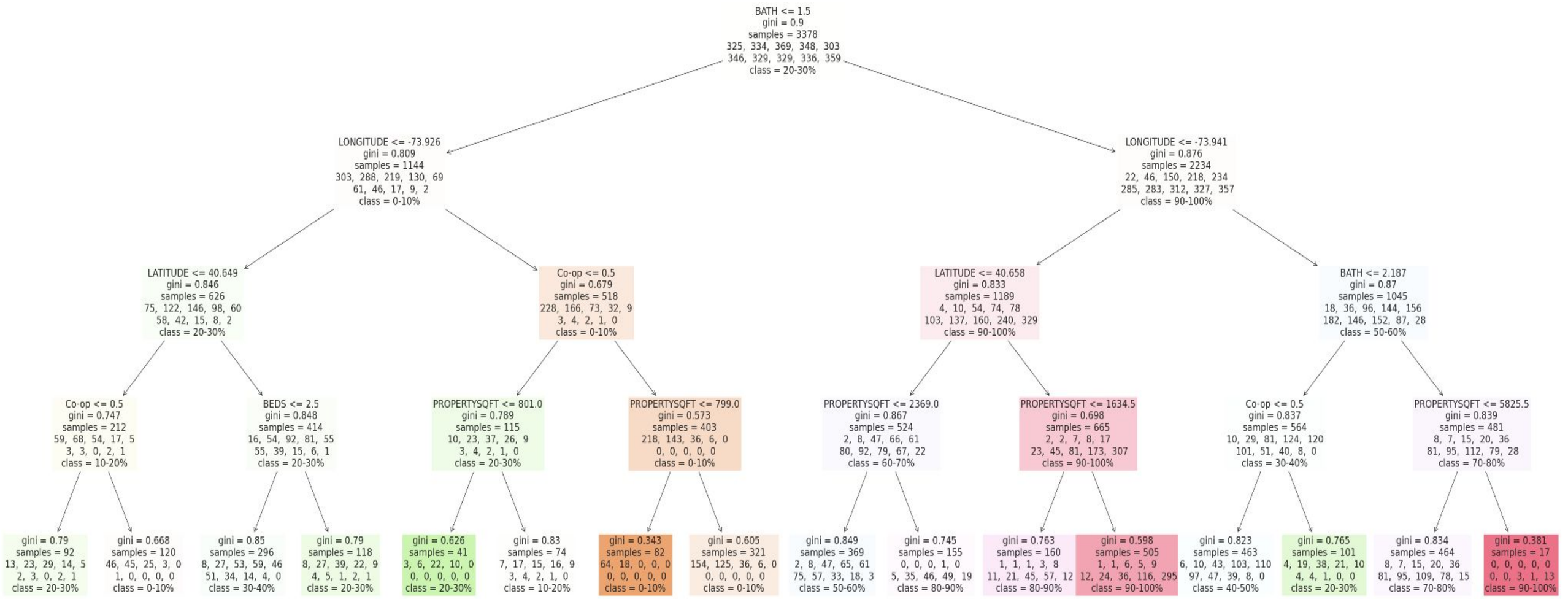
Train RMSE: 992,500

Test RMSE: 1,003,400

BASELINE RMSE: 1,457,868.74

DECISION TREE

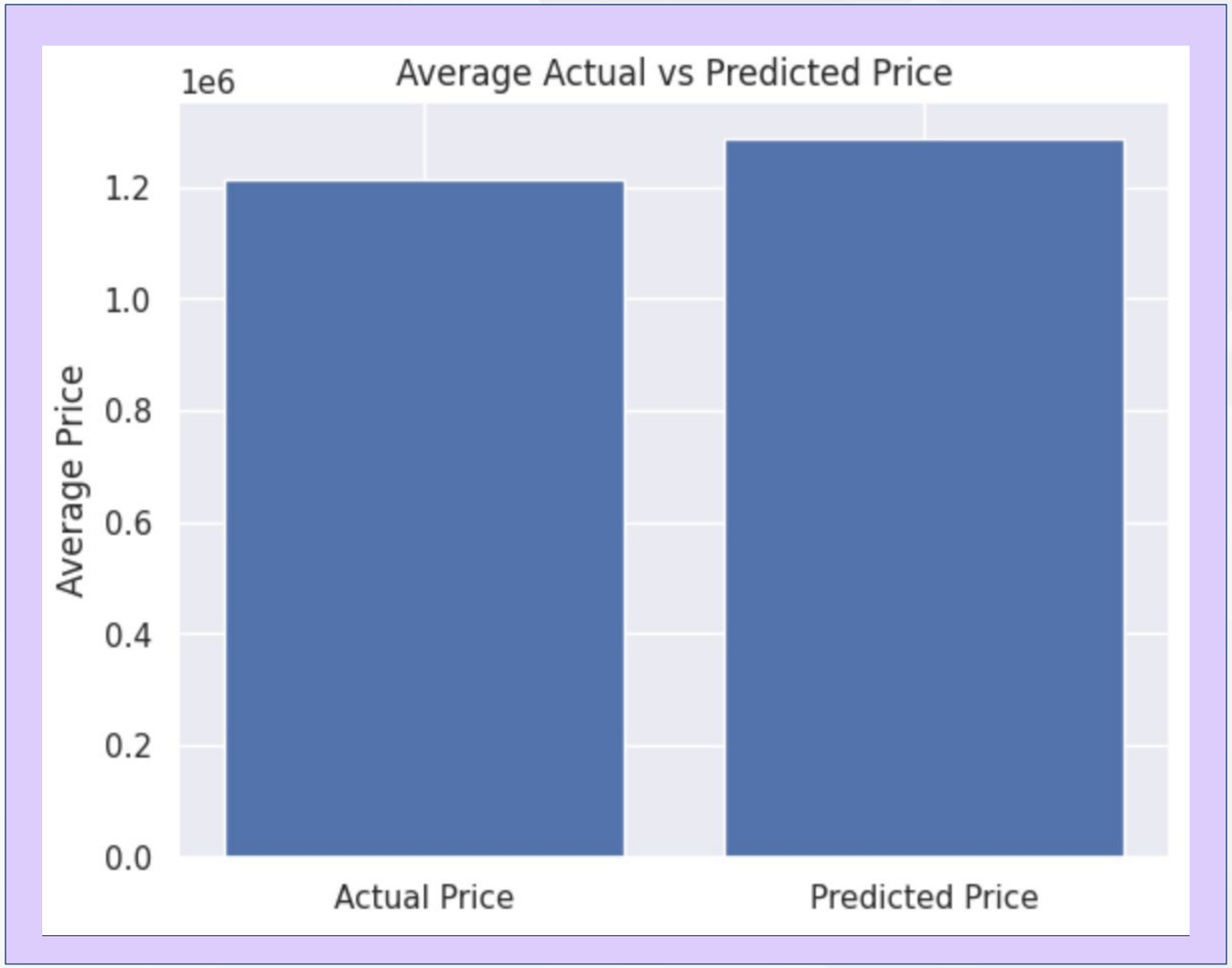
Classification Tree Training Accuracy: 35%
Classification Tree Testing Accuracy: 30%
70% of Baseline RMSE: 799,800



BASELINE RMSE: 1,457,868.74

BAGGING WITH NESTED CV

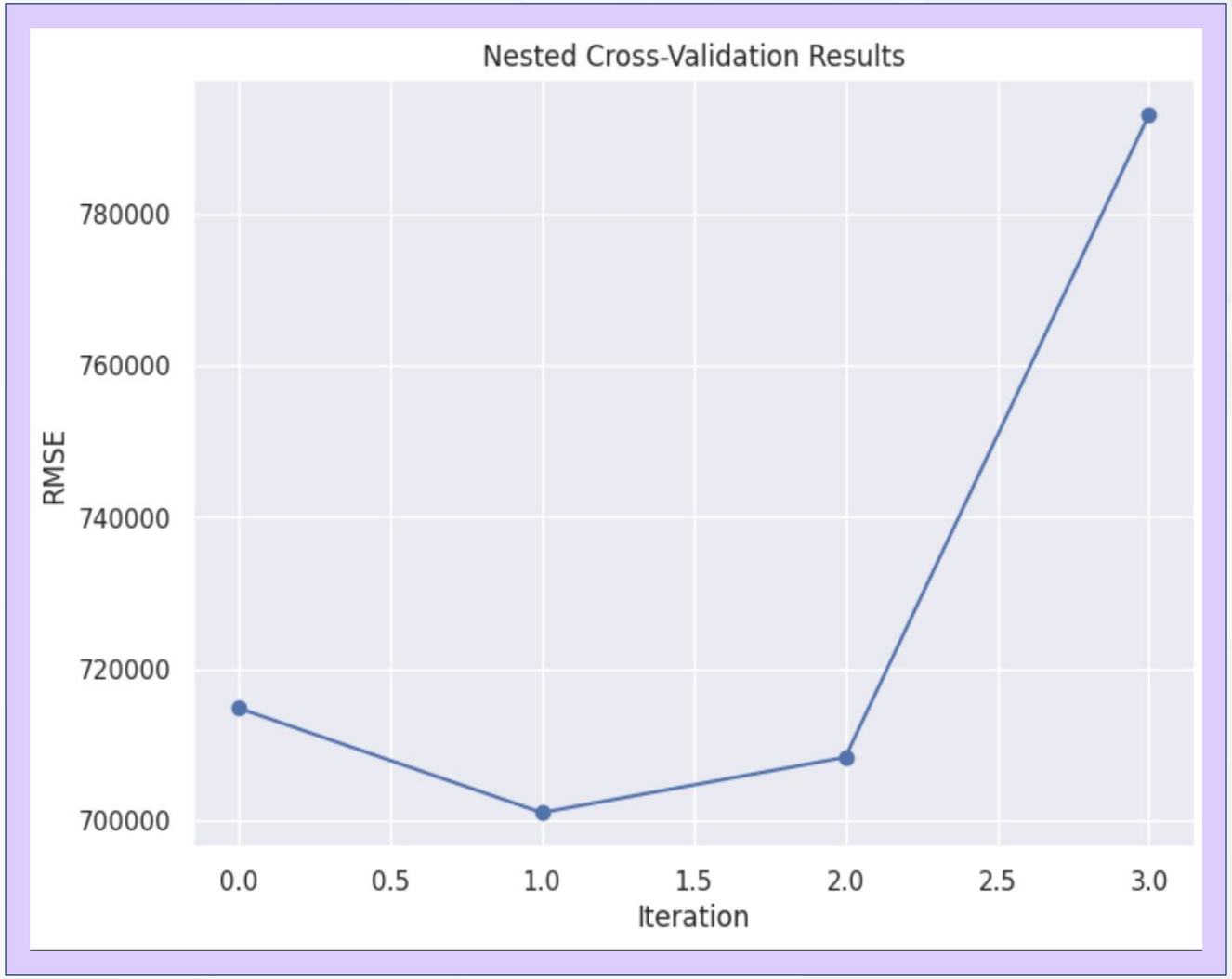
- Performed parameter tuning using grid search CV, nested CV, and regular CV
- Nested CV does the best out of these 3 parameter tuning techniques
- Nested CV RMSE: 717,600
- Slight improvement from decision tree model



BASELINE RMSE: 1,457,868.74

BOOSTING WITH NESTED CV

- Performed parameter tuning using grid search CV, nested CV, and regular CV
- Nested CV does the best out of these 3 parameter tuning techniques
- Nested CV RMSE: 729,300
- Substantial improvement: over 50% decrease from baseline model

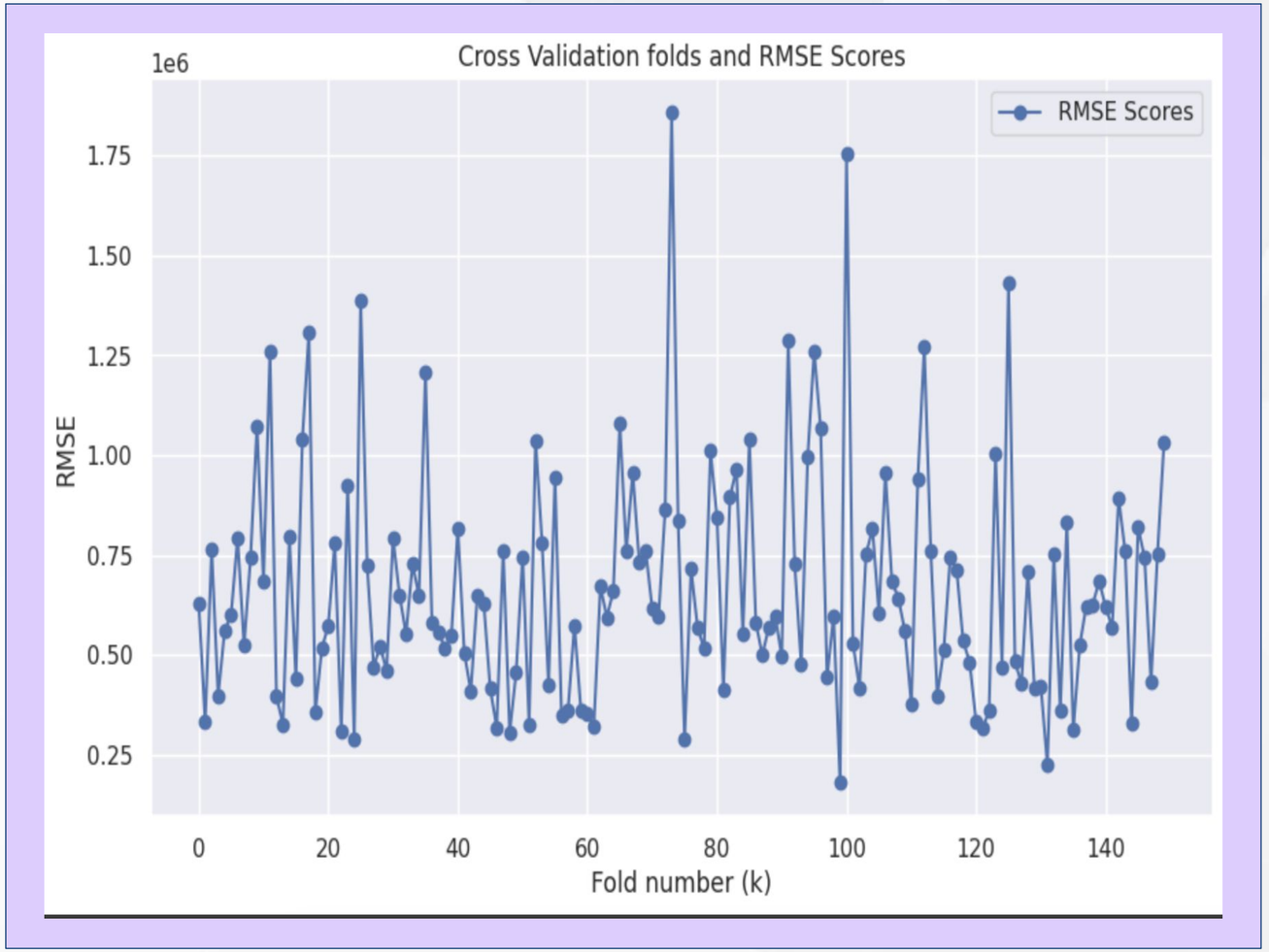


BASELINE RMSE: 1,457,868.74

OUR BEST MODEL: RANDOM FOREST WITH CV

- Tradeoff: Higher computational costs with parameter tuning of the CV parameter
  - CV = 150: 37 seconds, MSE: 600,000s
  - CV = 2,000: 7 minutes, MSE: 400,000s

- Why Random Forest is our best model
  - Lowest RMSE, simpler model, linear based
  - Binary variables from dummy coding
  - Application/Implication: Used a Google API to predict price based on a user's preference



BASELINE RMSE: 1,457,868.74

CONSOLIDATED VIEW OF MODELS (note these are rounded values)

Models	Training RMSE	Testing RMSE	Cross Validation
Ridge	992,500	1,003,359.69	1,003,400
Lasso	990,800	1,008,100	1,008,100
Decision Tree	790,600	799,800	863,300
Bagging	1,272,100	1,144,800	717,600
kNN	667,600	769,700	699,400
Boosting	1,120,800	1,013,800	729,300
Random Forest	**618,700**	**700,300**	**424,000**
Neural Network	992,800	992,900	

CONCLUSION

- Takeaways:
  - Simpler models generally led to better RMSE scores
    - ie. Regular CV vs. Nested or Grid search CV
  - Tuning parameters by different CV techniques are valuable and can drastically change the outcome
  - Precise and diligent data preprocessing greatly increased model's accuracy