New Model Output Results:

--- Test Set ---

Mean Absolute Error: ... 6.078747834396562

Mean Squared Error:..... 67.18

RMSE: .................. 8.196381335337657

Coeff of det (R^2):..... **0.809** (1.4 % better)

--- Val Set ---

Mean Absolute Error: ... 4.498476271962494

Mean Squared Error:..... 38.02

RMSE: .................. 6.166026942280251

Coeff of det (R^2):..... **0.904** (0.7 % better)

--- Train Set ---

Mean Absolute Error: ... 2.339212417385334

Mean Squared Error:..... 10.24

RMSE: .................. 3.200258043072929

Coeff of det (R^2):..... **0.965**

Current Production Model R^2:

Test: 0.795

Validation: 0.897

Train: 0.962

New Model Hyperparameters (XGBoost-based)

max\_depth=7

min\_child\_weight=6

gamma = 10

subsample=0.75

colsample\_bytree = 0.5

reg\_alpha = 100

reg\_lambda = 1

n\_estimators=750 *(can add more if desired)*

learning\_rate=0.16

seed=42

tree\_method='hist

Features

*+ storey\_range\_avg*

*(derived from storey\_range\_max and*

*storey\_range\_min)*

Keep Historic

This seemed to be pretty good…

--- Test Set ---

Mean Absolute Error: ... 6.078747834396562

Mean Squared Error:..... 67.18

RMSE: .................. 8.196381335337657

Coeff of det (R^2):..... 0.809

--- Val Set ---

Mean Absolute Error: ... 4.498476271962494

Mean Squared Error:..... 38.02

RMSE: .................. 6.166026942280251

Coeff of det (R^2):..... 0.904

--- Train Set ---

Mean Absolute Error: ... 2.339212417385334

Mean Squared Error:..... 10.24

RMSE: .................. 3.200258043072929

Coeff of det (R^2):..... 0.965

xg = xgb.XGBRegressor(

objective='reg:squarederror',

max\_depth=7,

min\_child\_weight=6,

gamma = 10,

subsample=0.75,

colsample\_bytree = 0.5,

reg\_alpha = 100,

reg\_lambda = 1,

n\_estimators=1000,

learning\_rate=0.16,

nthread=4,

seed=42,

tree\_method='hist')