NC STATE UNIVERSITY

Housing Price Prediction

Decision Tree | Random Forest | XGBoost | ANN

Kaijun Zhang | Wesley Meredith | Yiran Zhu

Problem Description

Objectives:

1. Determine the most effective machine learning technique for predicting home values in the Ames, Iowa Housing Dataset.
2. Understand the factors contributing to performance differences among Decision Tree, Random Forest, XGBoost, and Artificial Neural Network (ANN).

Dataset:

Ames, Iowa Housing Dataset[1]

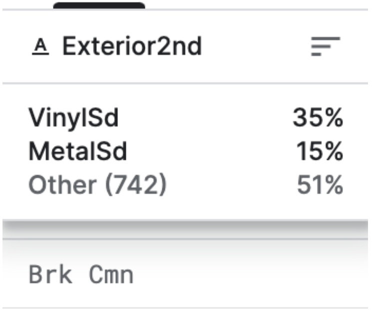
- 79 features, 1460 rows of training data
- Includes diverse property attributes like living area, lot size, and number of bedrooms.
- A mix of both numerical and categorical data.

Outline

1. Data Preprocessing
   – Feature Selection, PCA, Missing values, Encoding, Typos and Corrupt Data…
2. Feature Engineering
3. Model training
4. Model comparison
5. Conclusion

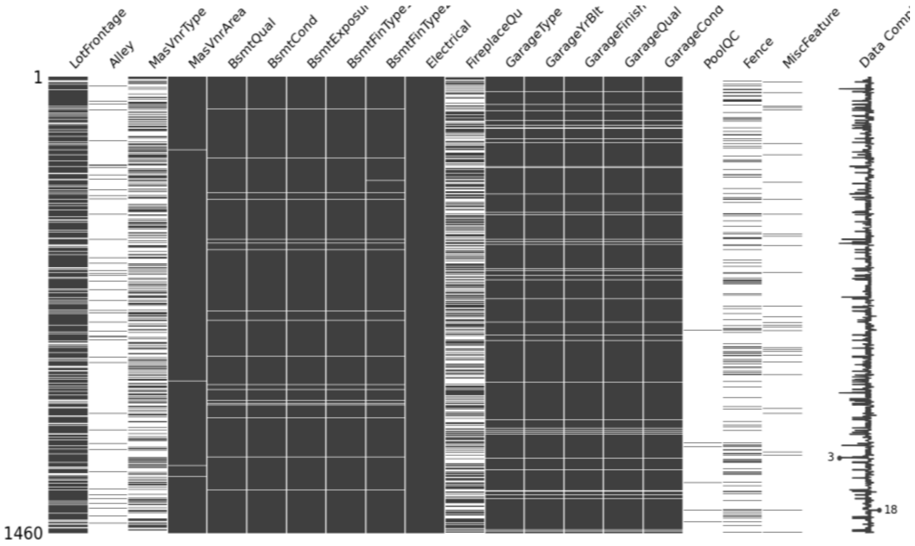
Data Preprocessing

- Typos and Corrupt Data



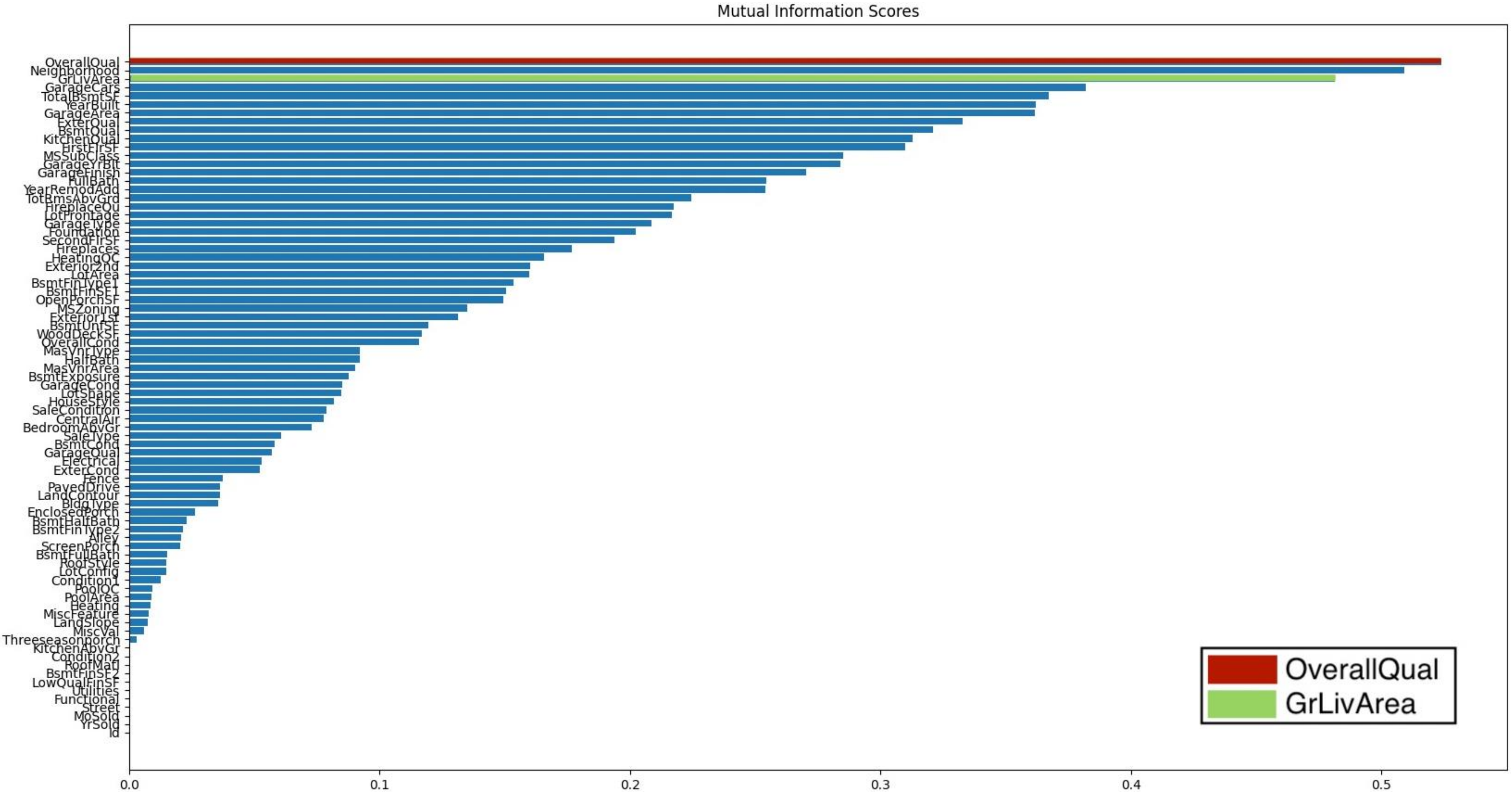

```
Exterior2nd: Exterior covering on house (if more than one material)

       AsbShng   Asbestos Shingles
       AsphShn   Asphalt Shingles
       BrkComm   Brick Common
```

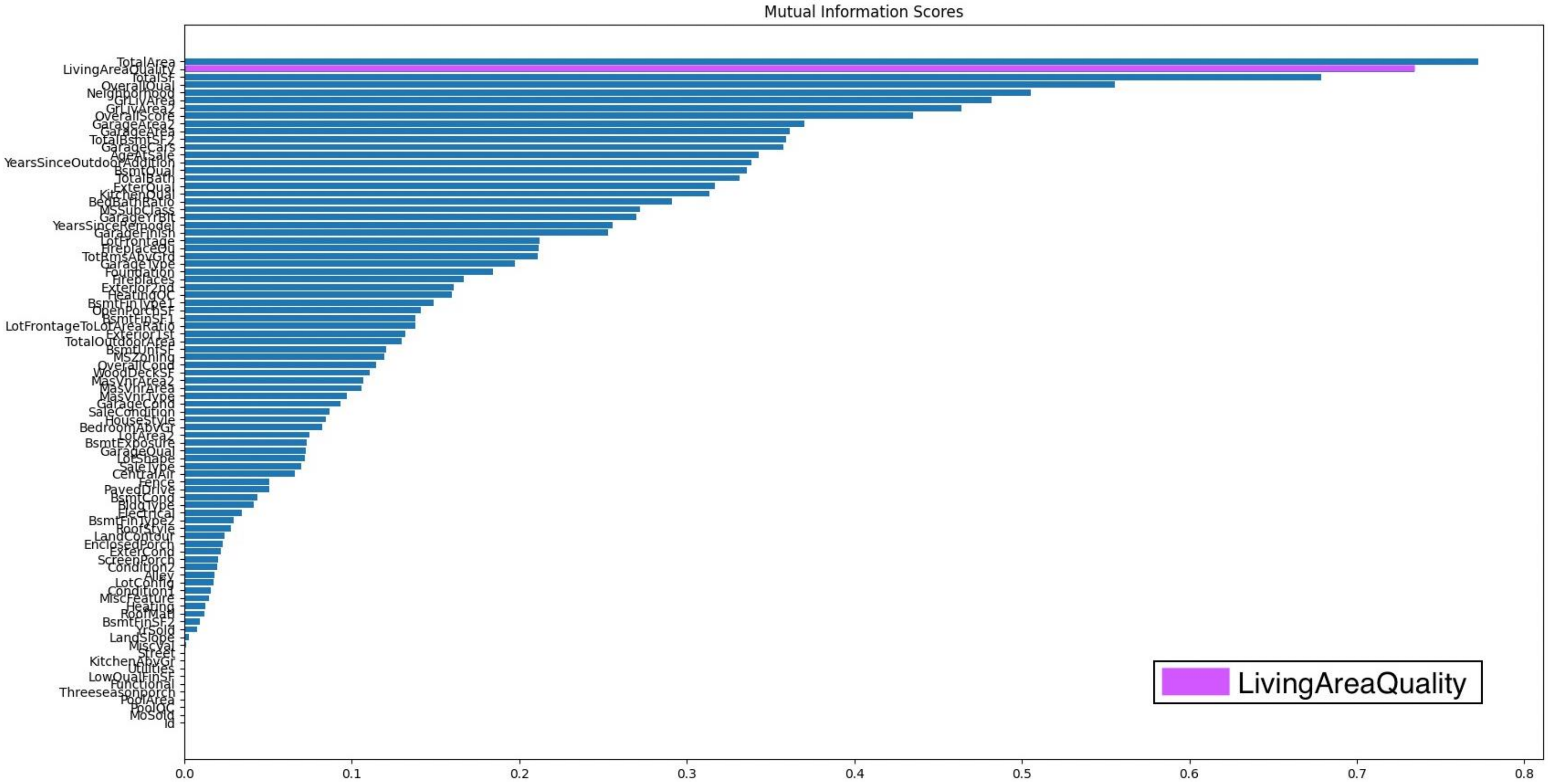
- Missing Values
  - 0 for numerical values
  - 'None' for categorical values



Feature Engineering

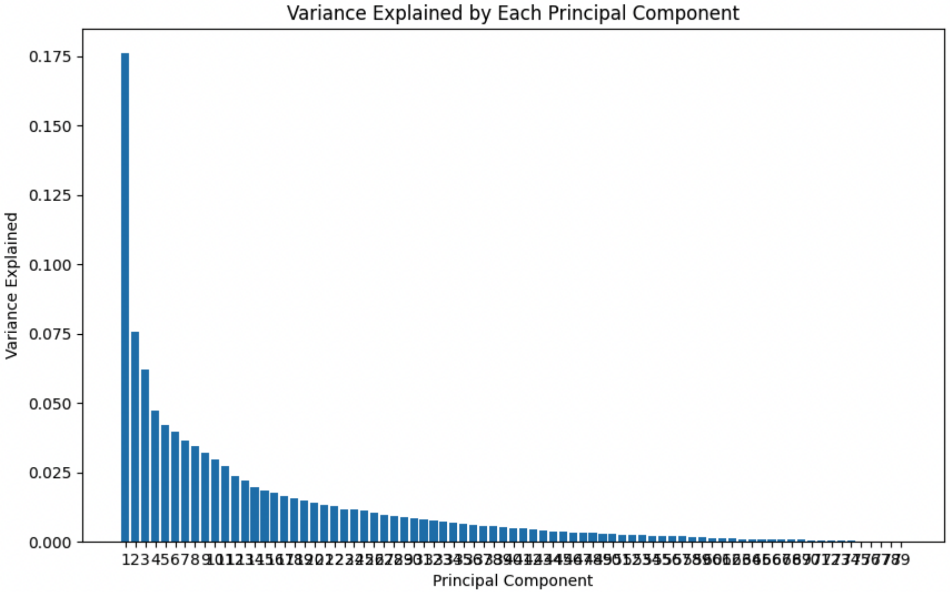
- Interaction Features
  - LivingAreaQuality (GeneralLivingArea * OverallQuality)
- Ratios
  - BedBathRatio (Number of bed / Number of Bath + epsilon)
- Polynomial Features [2]
  - $GeneralLivingArea^2$ …
- Age Related
  - AgeAtSale (YrSold – YrBuilt)


Mutual Information Scores
OverallQual
Neighborhood
GrLivArea
GarageCars
TotalBsmtSF
YearBuilt
GarageArea
ExterQual
BsmtQual
KitchenQual
FirstFlrSF
MSSubClass
GarageYrBlt
GarageFinish
FullBath
YearRemodAdd
TotRmsAbvGrd
FireplaceQu
LotFrontage
GarageType
Foundation
SecondFlrSF
Fireplaces
HeatingQC
Exterior2nd
LotArea
BsmtFinType1
BsmtFinSF1
OpenPorchSF
MSZoning
Exterior1st
BsmtUnfSF
WoodDeckSF
OverallCond
MasVnrType
HalfBath
MasVnrArea
BsmtExposure
GarageCond
LotShape
HouseStyle
SaleCondition
CentralAir
BedroomAbvGr
SaleType
BsmtCond
GarageQual
Electrical
ExterCond
Fence
PavedDrive
LandContour
BldgType
EnclosedPorch
BsmtHalfBath
BsmtFinType2
Alley
ScreenPorch
BsmtFullBath
RoofStyle
LotConfig
Condition1
PoolQC
PoolArea
Heating
MiscFeature
LandSlope
MiscVal
Threeseasonporch
KitchenAbvGr
Condition2
RoofMatl
BsmtFinSF2
LowQualFinSF
Utilities
Functional
Street
MoSold
YrSold
Id
0.0
0.1
0.2
0.3
0.4
0.5
OverallQual
GrLivArea

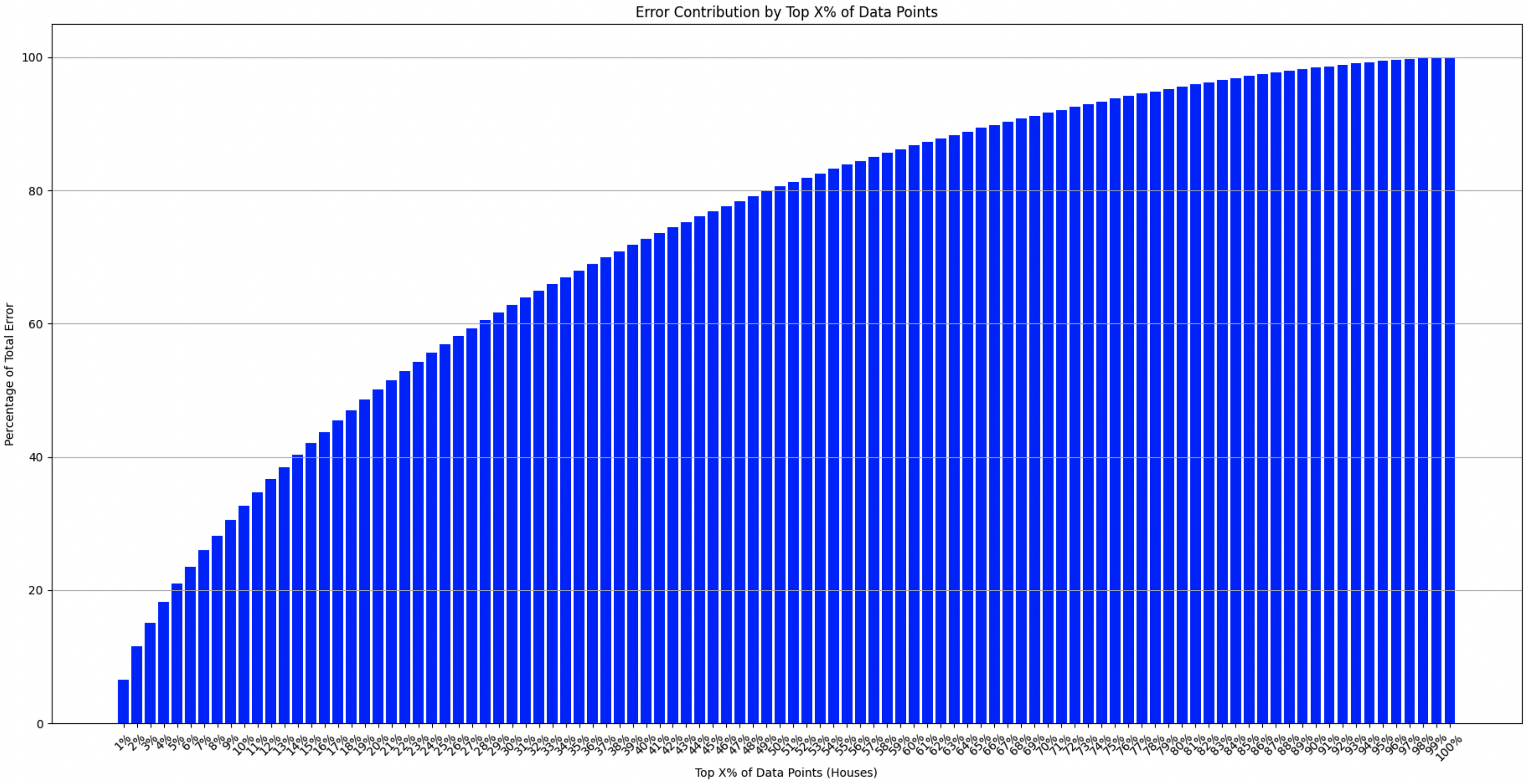

Mutual Information Scores
TotalArea
LivingAreaQuality
TotalSF
OverallQual
Neighborhood
GrLivArea
GrLivArea2
OverallScore
GarageArea2
GarageArea
TotalBsmtSF2
GarageCars
AgeAtSale
YearsSinceOutdoorAddition
BsmtQual
TotalBath
ExterQual
KitchenQual
BedBathRatio
MSSubClass
GarageYrBlt
YearsSinceRemodel
GarageFinish
LotFrontage
FireplaceQu
TotRmsAbvGrd
GarageType
Foundation
Fireplaces
Exterior2nd
HeatingQC
BsmtFinType1
OpenPorchSF
BsmtFinSF1
LotFrontageToLotAreaRatio
Exterior1st
TotalOutdoorArea
BsmtUnfSF
MSZoning
OverallCond
WoodDeckSF
MasVnrArea2
MasVnrArea
MasVnrType
GarageCond
SaleCondition
HouseStyle
BedroomAbvGr
LotArea2
BsmtExposure
GarageQual
LotShape
SaleType
CentralAir
Fence
PavedDrive
BsmtCond
BldgType
Electrical
BsmtFinType2
RoofStyle
LandContour
EnclosedPorch
ExterCond
ScreenPorch
Condition2
Alley
LotConfig
Condition1
MiscFeature
Heating
RoofMatl
BsmtFinSF2
YrSold
LandSlope
MiscVal
Street
KitchenAbvGr
Utilities
LowQualFinSF
Functional
Threeseasonporch
PoolArea
PoolQC
MoSold
Id
LivingAreaQuality
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8

Decision Tree

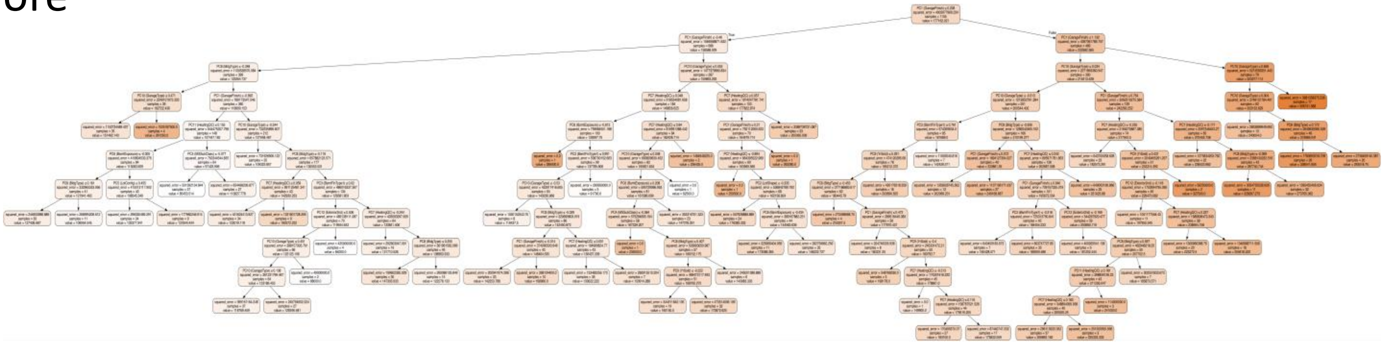
- 5-fold CV
- PCA
  - Before
    - RMSE: 36,500
    - Kaggle*: 0.238
  - After
    - RMSE: 29,000
    - Kaggle*: 0.246
  - Does not capture non-linear relationships
  - Loses crucial information
- Hyperparameters Search
  - Max_depth, min_samples_split…
  - Did not change results
- Outliers in dataset



Kaggle score* : measures the avg error percentage on unseen test data, lower better


Error Contribution by Top X% of Data Points
Percentage of Total Error
Top X% of Data Points (Houses)
0
20
40
60
80
100
1%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%

Before



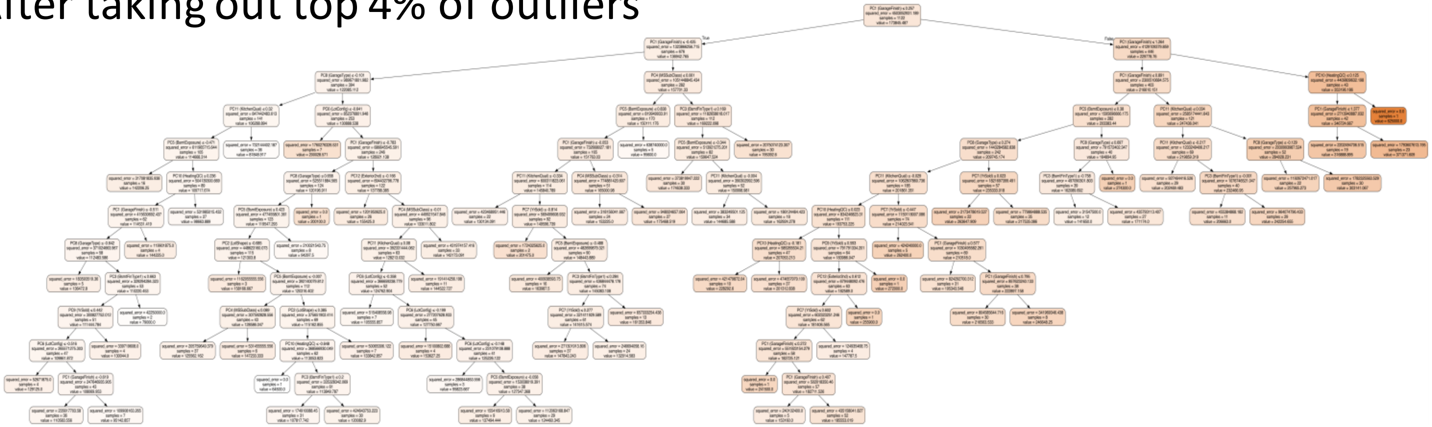
Score

RMSE: 36,500

Kaggle: 0.238

No PCA

After taking out top 4% of outliers



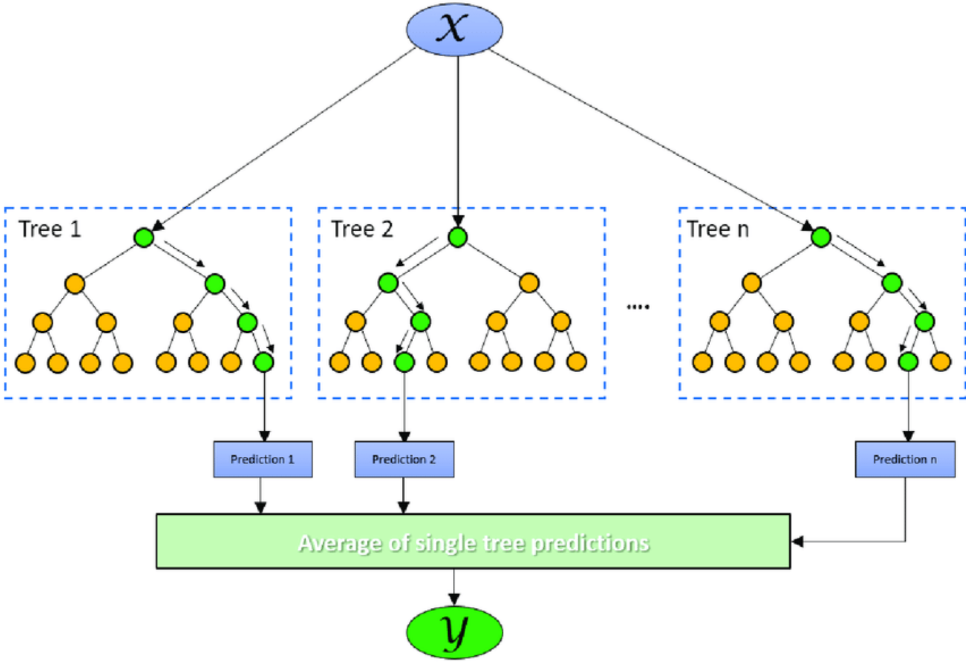
RMSE: 29,000

Kaggle: 0.246

No PCA

Random Forest

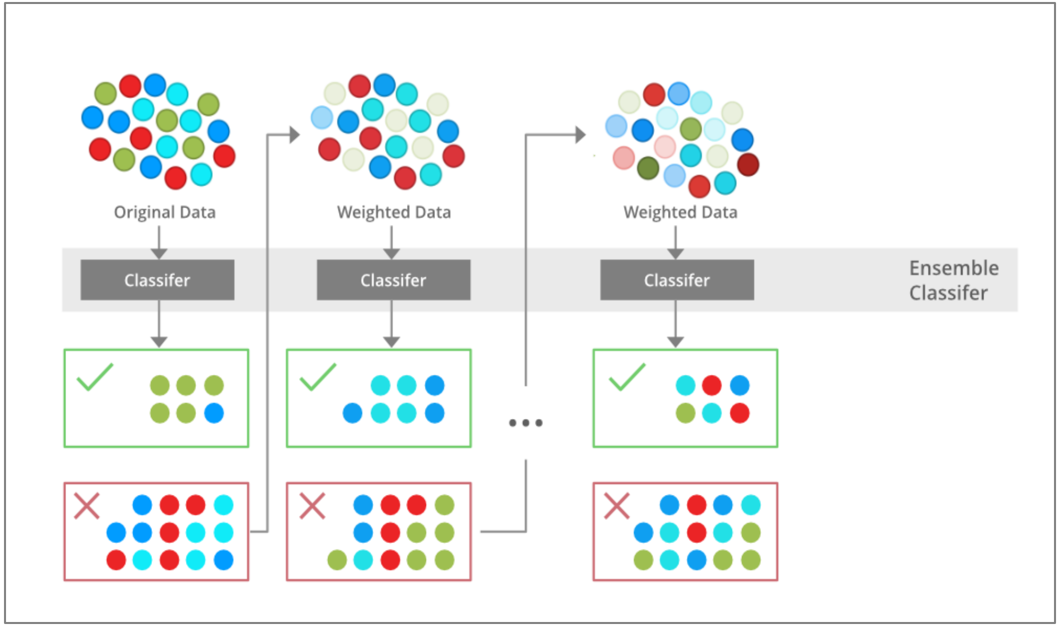
- Characteristics
  - Ensemble of 100 decision trees
  - Square Root of features per tree
  - Bagging
- PCA
  - Before
    - RMSE: 28284
    - Kaggle: 0.142
  - After
    - RMSE: 39841
    - Kaggle: 0.197
- No decision correction/boosting, perhaps too simple for our task



XGBoost

- Differs slightly from Random Forests due to ***Gradient Boosting***.

- ***Extreme Gradient Boosting***[3]
  - Ensemble learning method
  - Combines predictions of multiple weak learners to create a strong learner (*boosting*) via a gradient descent algorithm (*gradient*)
  - Handles missing values by default
  - XGBoost works sequentially, where each new tree learner aims to correct mistakes from those that came before it.



XGBoost

Method:

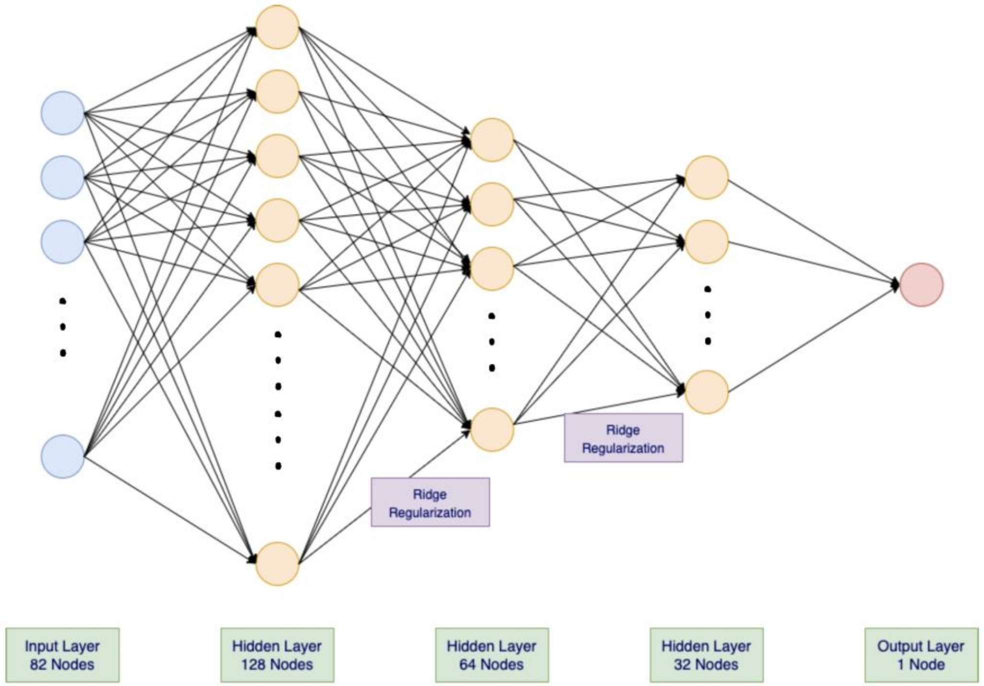
- No PCA
- 5-fold CV
- Hyperparameter optimization
  - Before:
    - RMSE: 29,041
  - After:
    - RMSE: 27,314

```
# Define the hyperparameter grid to search
param_grid2 = {
    'learning_rate': [0.01, 0.1, 0.2],
    'n_estimators': [100, 200, 300],
    'max_depth': [3, 4, 5],
    'min_child_weight': [1, 2, 3],
    'subsample': [0.8, 0.9, 1.0],
    'colsample_bytree': [0.8, 0.9, 1.0],
}
```

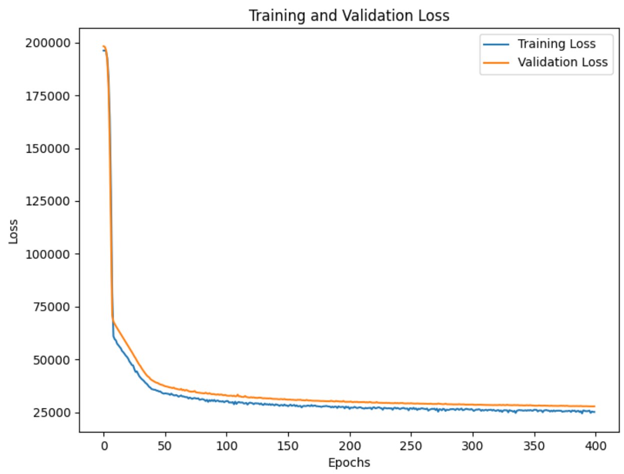
- Hyperparameter grid search did not yield significant improvements to the RMSE.

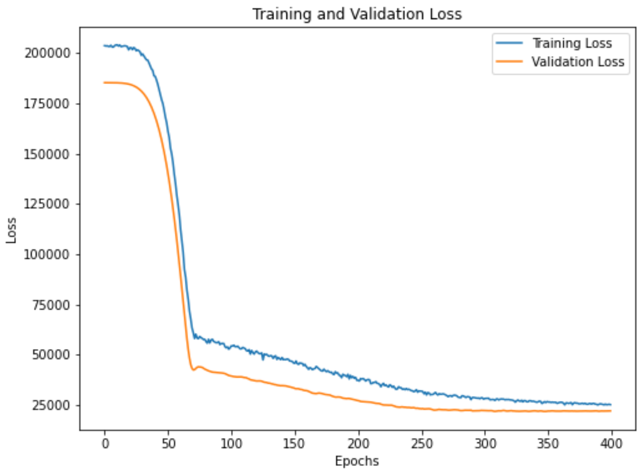
- Therefore, pre-processing and hyper-parameter optimization did not significantly affect performance on this specific dataset for XGBoost.

Artificial Neural Networks (ANN)



- **Data Split:**
  - 80% training, 20% validation from 'training.csv'
- **Neural Network:**
  - Architecture: 3 hidden layers (128, 64, 32 neurons)
  - Activation Function: ReLU
- **Model Evaluation:**
  - Loss Function: RMSE
- **Overfitting Prevention:**
  - Used Ridge Regularization (L2) in hidden layers





Before Regularization:

- Validation loss exceeded the training loss.
- Model convergence occurred at approximately 27,000.
- Kaggle score: 0.1872

After Regularization:

- In this phase, validation loss appears smaller than the training loss due to exclusion of regularization during its calculation.
- Model convergence improved to 22,000.
- Kaggle score: 0.13651

Comparison and Conclusion

Table 1: Comparison of Models Before and After Feature Engineering

Feature Engineering	**Decision Tree**	**Random Forest**	**XGBoost**	**ANN**
Before	32,873 | 0.251	36,924 | 0.198	29,041 | 0.167	32,483 | 0.166
After	26,542 | 0.198	33,314 | 0.142	27,314 | 0.139	21,776 | 0.137

Note: The values in the table represent RMSE | Kaggle scores.

- Feature Engineering generally improves model performance across all models.
- XGBoost and ANN perform better on the housing price prediction task.

Takeaway

- PCA does not perform well on datasets with non-linear relationships.
- Hyper parameter tuning has minimal impact on the accuracy of the model.
- Model selection and feature engineering ultimately determines the ceiling of our performance.

Sources

- [1] Data: Kaggle. (2017). House Prices Competition Data. Retrieved from https://www.kaggle.com/competitions/house-prices-advanced-regression-techniques/data"
- [2] Fan C, Cui Z, Zhong X. House Prices Prediction with Machine Learning Algorithms. Proceedings of the 2018 10th International Conference on Machine Learning and Computing - ICMLC 2018. doi:10.1145/3195106.3195133.
- [3] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. https://doi.org/10.1145/2939672.2939785