Housing Affordability Data

Gautham Gowda

Contents

Motivation for the study

Housing dataset is rich with consumer information and housing information.

Contains many important features about housing costs, income and burden of homeowners.

The data is used to predict the housing affordability of consumers based on location, income, burden, home size etc.

Aim is to build a model to use minimum features while keeping the prediction accuracy at the highest level.

Clients/ Intended audience

The model can be used by individual home buyers wanting to know the house they can afford based on their income, costs, housing locations.

It can be used by the lenders to screen loan applicants as well.

The clientele could be banks, mortgage lending institutions, government agencies determining housing affordability such as census bureau.

Dataset – prudential life data

The dataset used for this analysis is from HADS database from the huduser.gov website and the data source is listed below:

https://www.huduser.gov/portal/datasets/hads/hads.html

The database contains many features such as income, burden, average housing cost, poverty income that are relevant for the analysis. The following table summarizes all the features available

Data cleaning and wrangling

Duplicate features removed

- Original dataset has 99 features with some formatted duplicate features
- 25 duplicate features removed with the following method:

```
df = df.drop(df.filter(regex='FMT').columns, axis=1)
```

Highly correlated independent variables are reduced

- Used correlation matrix to reduce the features

Data cleaning and wrangling

Outliers and Missing values removed with following snippet

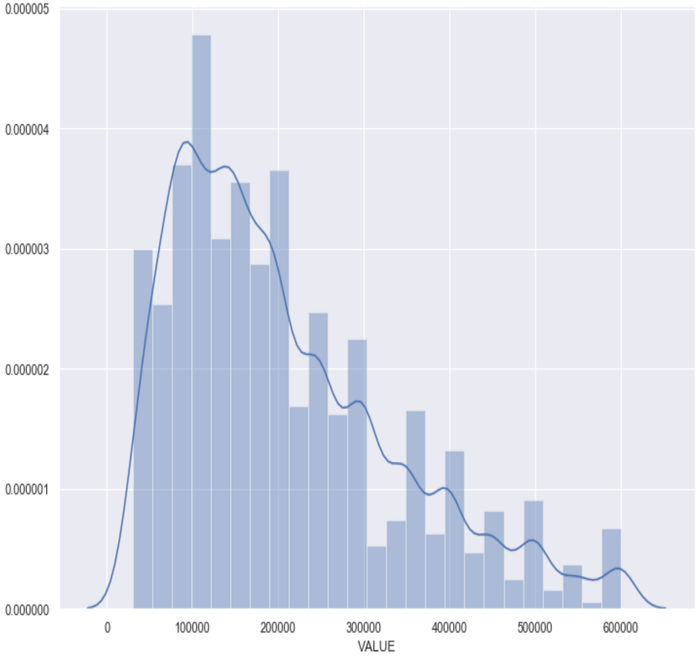
```
# Remove missing values and negative values for AGE and home
value
#Use fillna method for ZINC2
df.loc[df.VALUE < 25000 ] =np.nan
df.loc[df.VALUE > 600000 ] =np.nan
df.loc[df.AGE1 <5]=np.nan
df.loc[df.ZINC2 < 1000] =np.nan
df.ZINC2 = df.ZINC2.fillna(method='ffill').fillna(method ='bfill')

df=df[df['VALUE'].notnull()]
df=df['AGE1'].notnull()]
```

home values above 25K and below 600K are used for the analysis to remove outliers
Head of house Age < 5 are removed (Outliers have values -1, 0, and 4)
Household Income <1000 are also removed

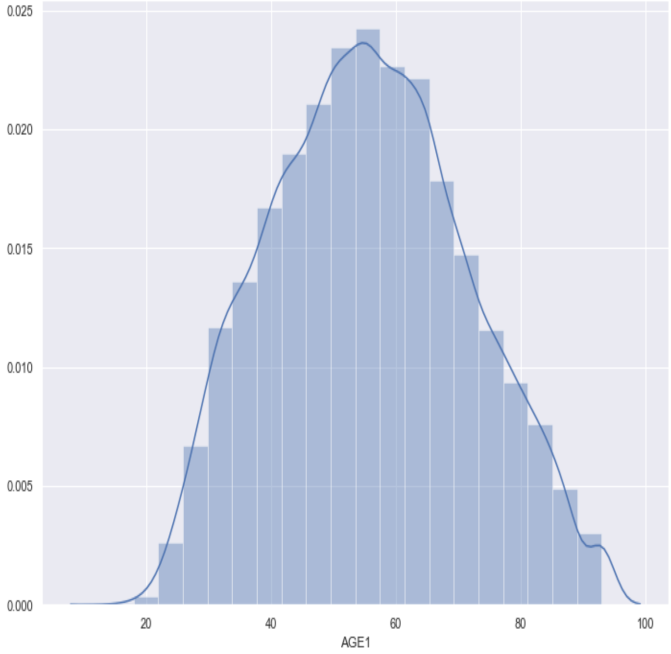
Data visualization- independent variables

Home value distribution



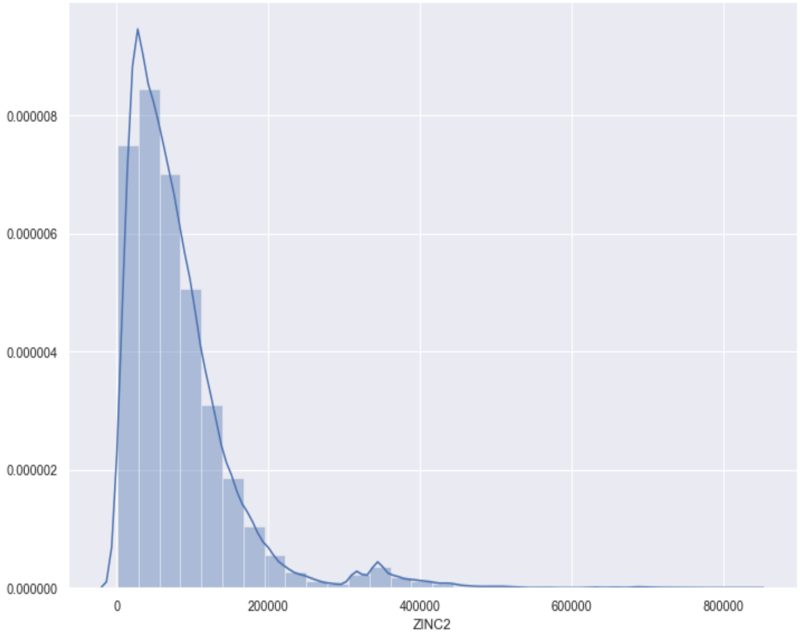
Summary Stats	
mean	206,751
std	131,450
25%	100,000
50%	180,000
75%	280,000

Age- head of household



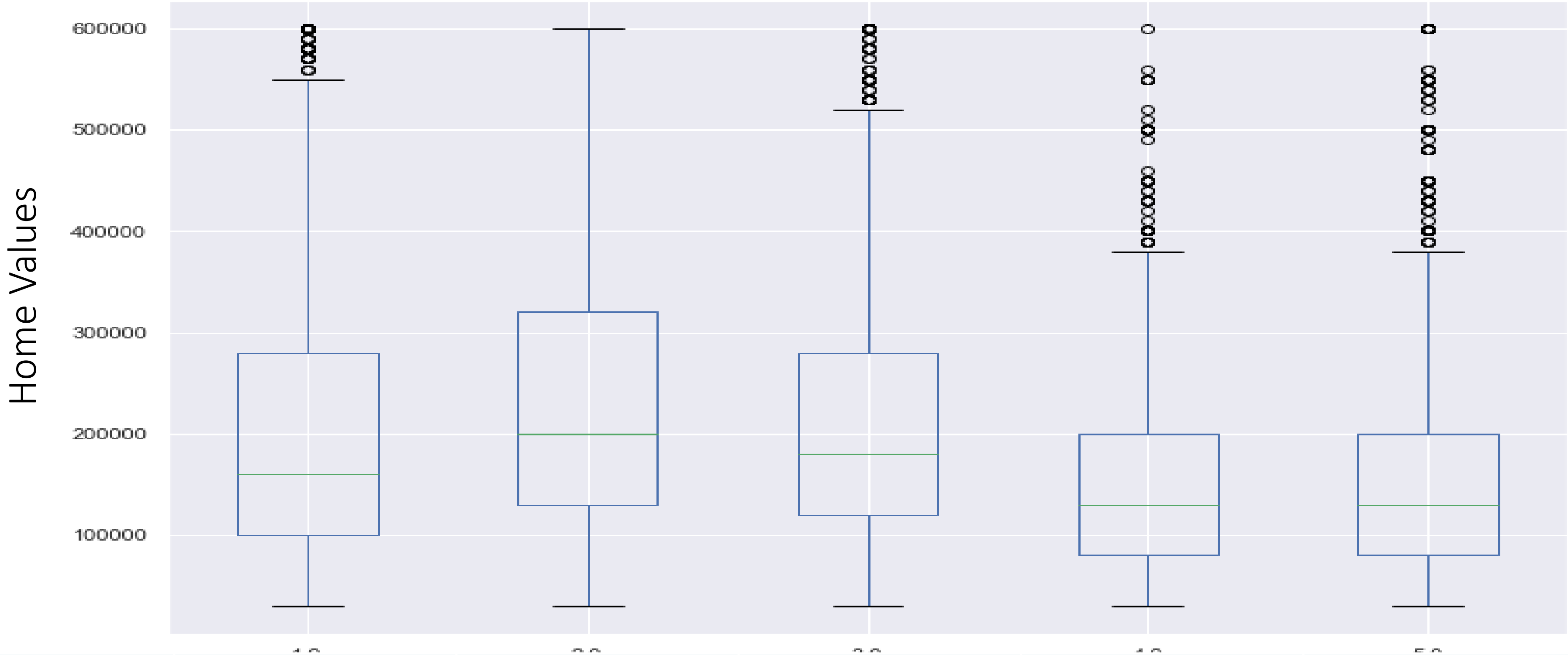
Summary Stats	
mean	56
std	16
25%	44
50%	55
75%	67

Income-house hold



Summary Stats	
mean	82,134
std	74,060
25%	33,650
50%	63,987
75%	104,987

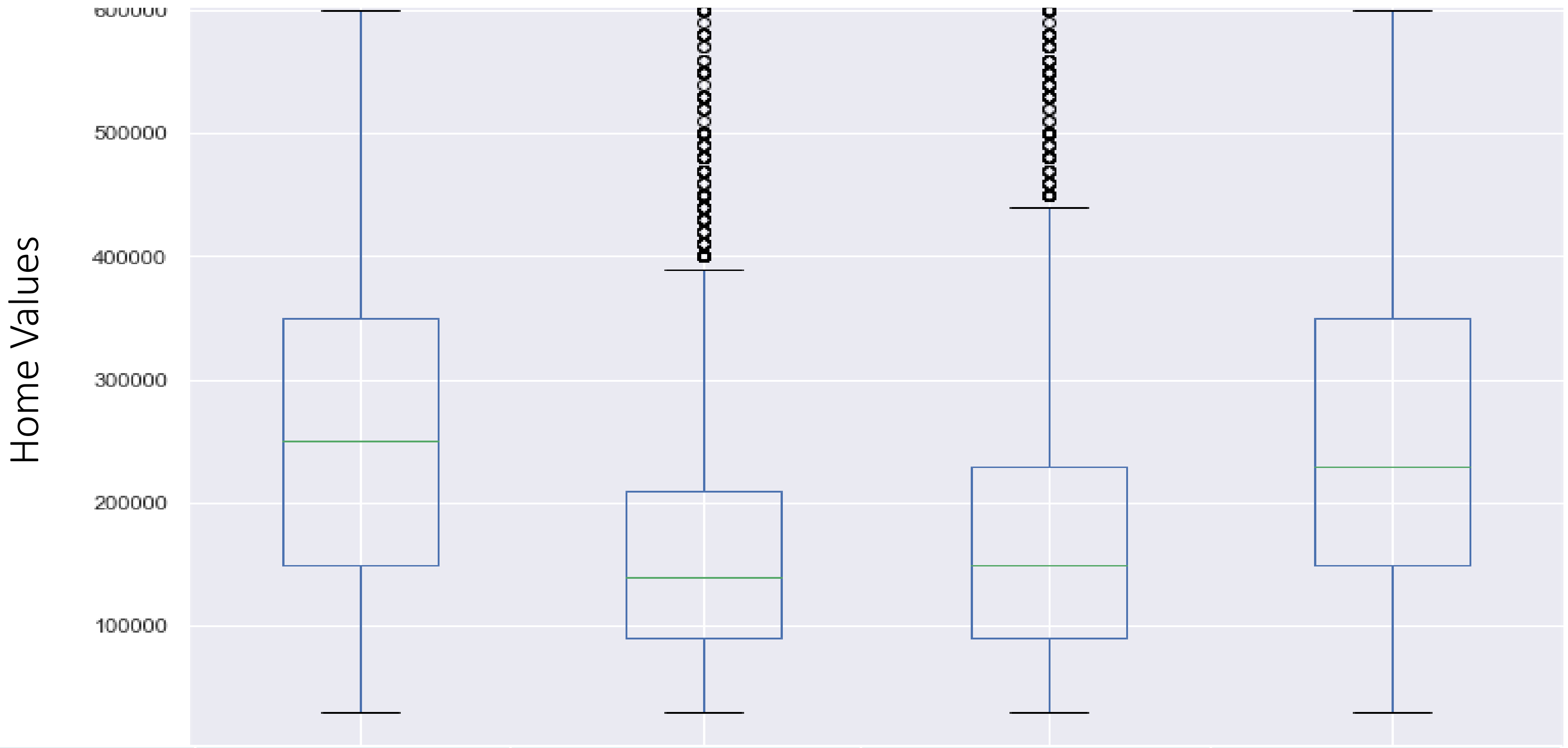
EDA – Home prices based on metro



	City Center-1	Suburb-2	Suburb-3	Suburb-4	Suburb-5
Mean	200049	232890	210006	150685	159886
STD	135984	136487	123674	96539	105864
25%	100000	130000	120000	80000	80000
50%	160000	200000	180000	130000	130000
75%	280000	320000	280000	200000	200000

Box plot of home values for each metro region (1= central city, 2-5 =suburban zones)

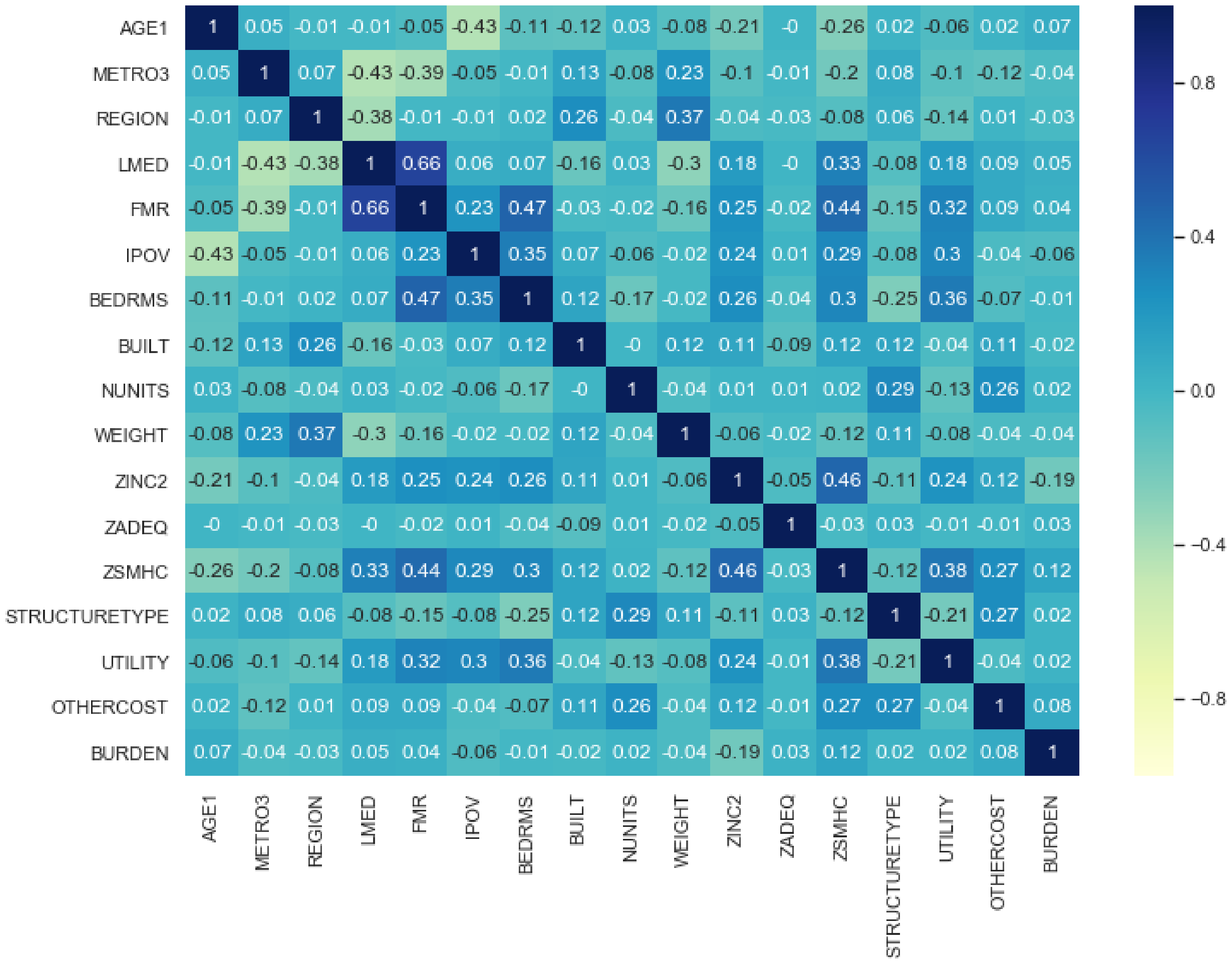
EDA – Home prices based on census zones



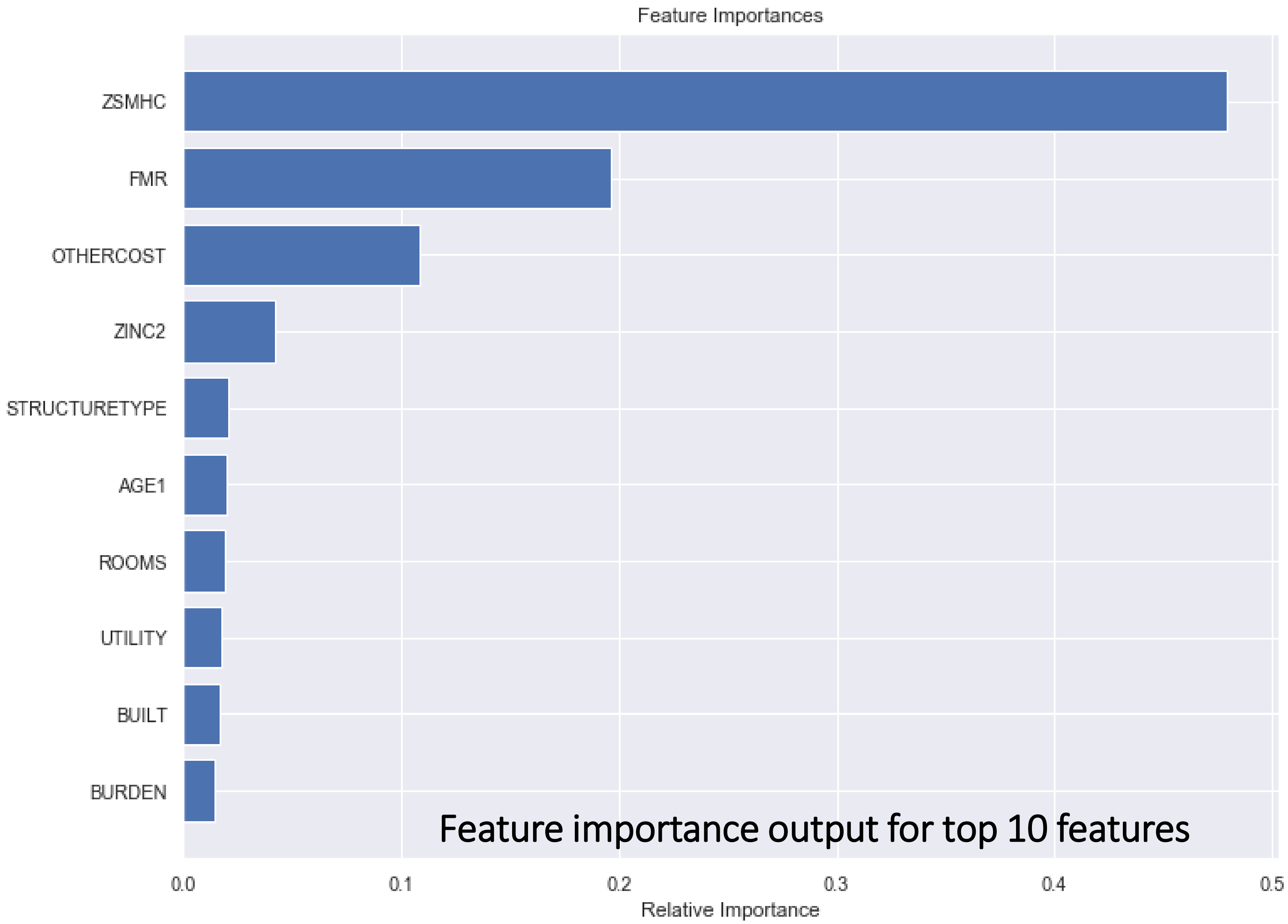
	East	Midwest	South	West
Mean	262,255	167,288	176,431	257,377
STD	139,491	107,529	115,719	143,399
25%	150,000	90,000	90,000	150,000
50%	250,000	140,000	150,000	230,000
75%	350,000	210,000	230,000	350,000

East and West US has higher median home prices

EDA – correlation matrix of independent variables



EDA – top 10 significant features



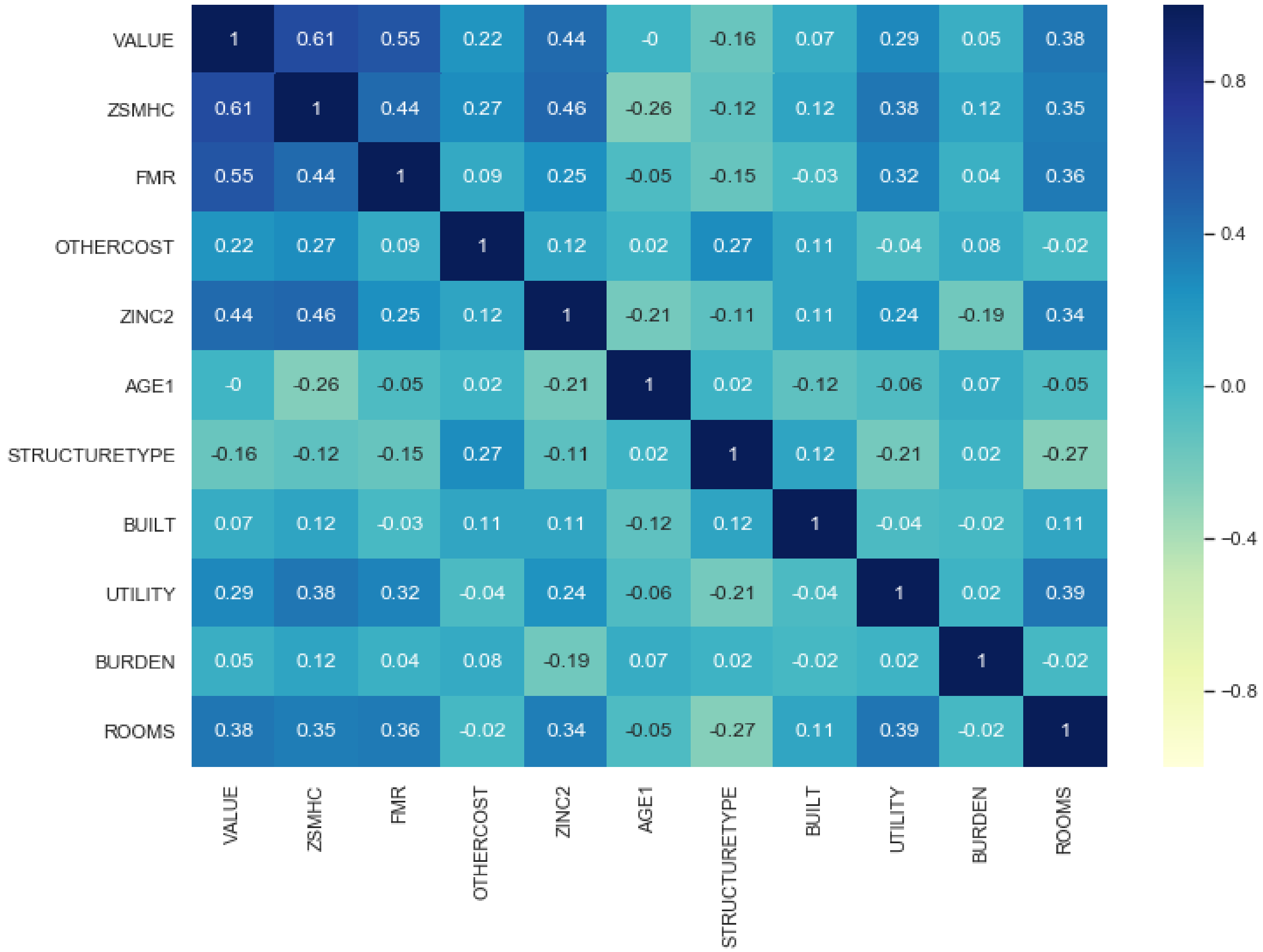
Feature importance output for top 10 features

Features considered for model

VALUE = Value of Unit (dependent feature)
ZSMHC = Monthly Housing Costs
FMR = Fair Market Rent
OTHERCOST = Insurance, HOA, land rent
ZINC2 = Household Income
STRUCTURE TYPE = Single Family/ Multi
AGE1 = Age of Head of Household
ROOMS = Total Rooms in the house
UTILITY = Monthly Utility Cost
BUILT = Year unit built
BURDEN = Housing Cost as fraction of Income
REGION = Census Region
METRO3 = City Center or Suburb
COSTMED = Housing cost at Median Interest

Final list after feature reduction from 100 to 14 features

EDA – correlation matrix of features with target variable



Age of the head of household is dropped from the model based on correlation matrix

Machine learning – models comparison

Three different regressor techniques are used

model	Training Accuracy	Test set accuracy	Delta RSME (Test-Train) *
Linear Regression (8 features)	0.47	0.45	2566
Linear Regression (4 features)	0.45	0.45	117
K-nearest neighbor (KNN)	0.54	0.36	16824
KNN hyper tuned with Random Search CV	0.50	0.38	11021
Random Forest	0.67	0.54	14809

* Higher Delta RSME and R2 indicate overfitting

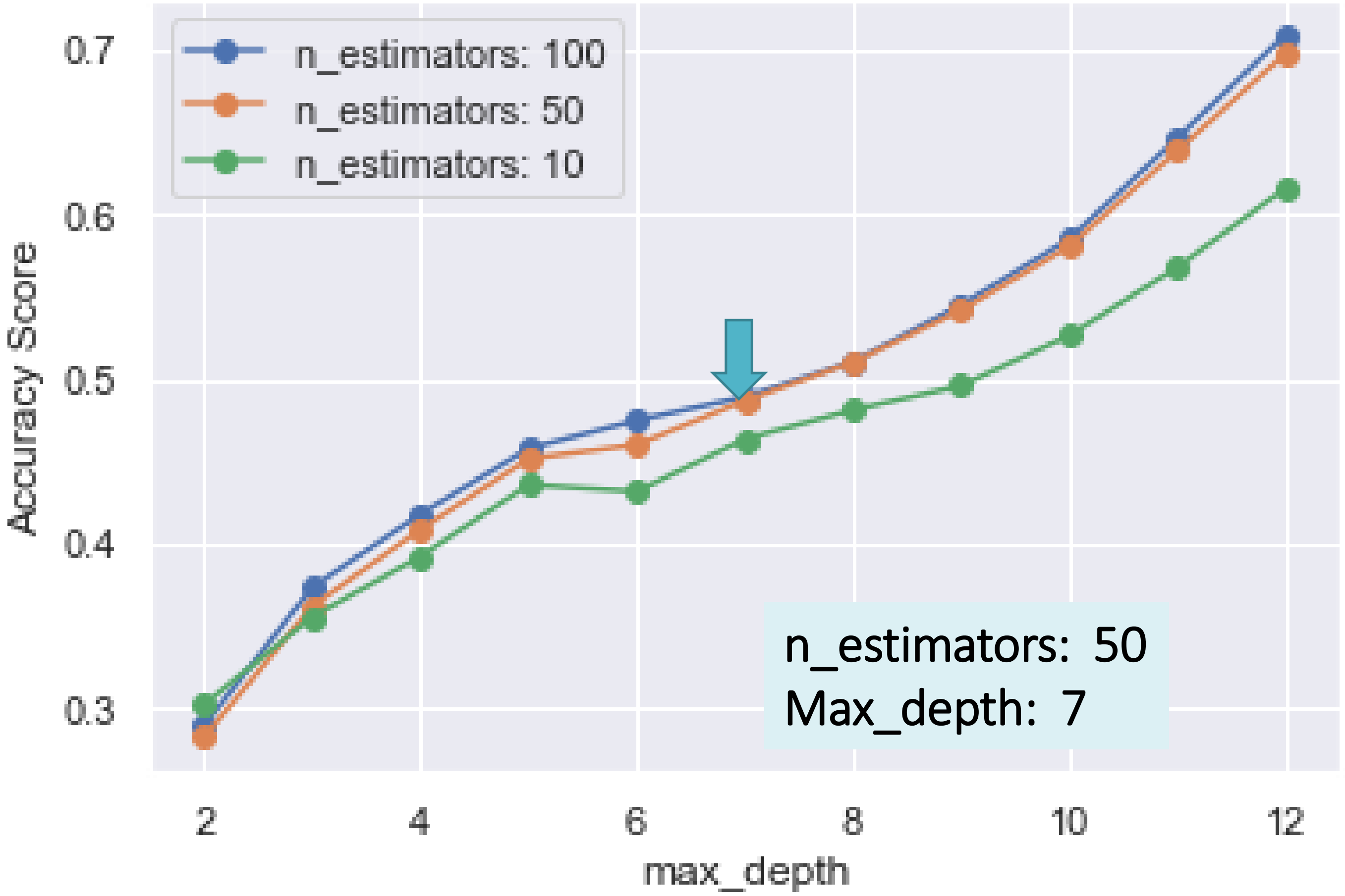
Linear Regression has the least overfitting between train and test data
Nearest neighbors (KNN) model has high overfitting
Random Forest model is selected based on higher R2 score

Random Forest -- Hyper parameter tuning

Randomized Search CV

Tuned Decision Tree
Parameters: max_depth: 9
max_features: 4
min_samples_leaf: 6
n_estimators: 100
Best score: 0.56

Grid Search CV



Randomized search CV is faster and also gives comparable results to Grid Search CV

Hyper parameter tuning–Random Forest

Cross- Validation of training set to minimize overfitting

Hyper parameter tuning using GridSearchCV

Hyper parameter tuning RandomizedSearchCV

model	Training Accuracy	Test set accuracy	Delta R2 (Test-Train)	Delta RSME (Test-Train)
Random Forest Default	0.67	0.54	0.13	14809
Random Forest Randomized CV	0.62	0.54	0.08	9103
Random Forest Grid Search CV	0.58	0.53	0.05	5954

Model overfitting is reduced with cross validation & parameter tuning

Grid Search CV has the best performance in terms of minimizing oversetting

Conclusion

The dataset has 100 features. Using dimension reduction, only 10 most important features are selected for decision making.

ML algorithms considered: Linear Regression, KNN, Random Forest

Based on the accuracy scores obtained, Random Forest model is chosen to train the data

Hyper parameter tuning is done to minimize overfitting using Randomized Search CV and Grid Search CV