ILLINOIS TECH

College of Computing

Group - 19

# Predictive Analysis and Clustering of Housing Data


**Nallamaddi Nachiketh - A20549679**

Nnallamadi@hawk.iit.edu

**Karukonda Anmol Rao - A20554502**

akarukonda@hawk.iit.edu

**Somu Medaka - A20548401**

smedaka@hawk.iit.edu


# Dataset Overview

```
housing_data.head()
```

| | longitude | latitude | housing_median_age | total_rooms | total_bedrooms | population | households | median_income | median_house_value | ocean_proximity |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | -122.23 | 37.88 | 41.0 | 880.0 | 129.0 | 322.0 | 126.0 | 8.3252 | 452600.0 | NEAR BAY |
| 1 | -122.22 | 37.86 | 21.0 | 7099.0 | 1106.0 | 2401.0 | 1138.0 | 8.3014 | 358500.0 | NEAR BAY |
| 2 | -122.24 | 37.85 | 52.0 | 1467.0 | 190.0 | 496.0 | 177.0 | 7.2574 | 352100.0 | NEAR BAY |
| 3 | -122.25 | 37.85 | 52.0 | 1274.0 | 235.0 | 558.0 | 219.0 | 5.6431 | 341300.0 | NEAR BAY |
| 4 | -122.25 | 37.85 | 52.0 | 1627.0 | 280.0 | 565.0 | 259.0 | 3.8462 | 342200.0 | NEAR BAY |

- The dataset we will use is the "California Housing Prices" , which is based on data from the 1990 census. This dataset offers great opportunities for learning. The prediction task for this dataset will be to predict housing prices based on several features.
- The Number of rows in dataset is 20640 and number of columns is 10

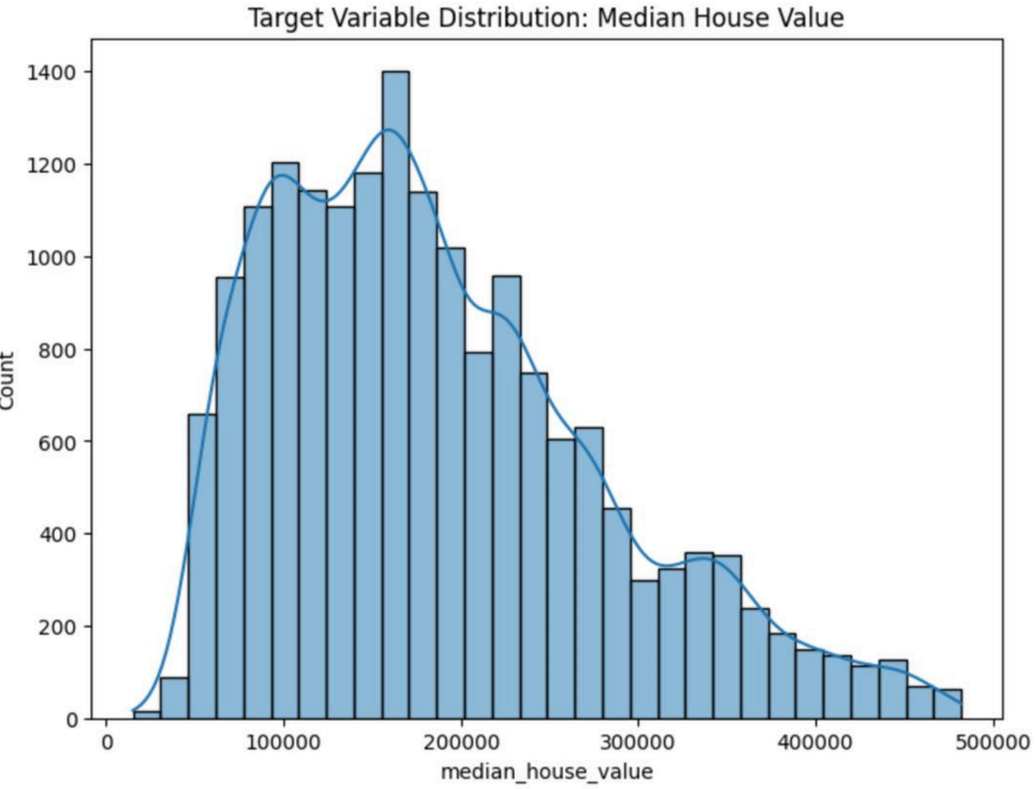
```
Data columns (total 10 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   longitude           20640 non-null  float64
 1   latitude            20640 non-null  float64
 2   housing_median_age  20640 non-null  float64
 3   total_rooms         20640 non-null  float64
 4   total_bedrooms      20433 non-null  float64
 5   population          20640 non-null  float64
 6   households          20640 non-null  float64
 7   median_income       20640 non-null  float64
 8   median_house_value  20640 non-null  float64
 9   ocean_proximity     20640 non-null  object 
dtypes: float64(9), object(1)
memory usage: 1.6+ MB
```

# Data Cleaning

- Handling Missing Values
- Removing Duplicates
- Handling Outliers
- Encoding Categorical Variables

# Target Variable Analysis

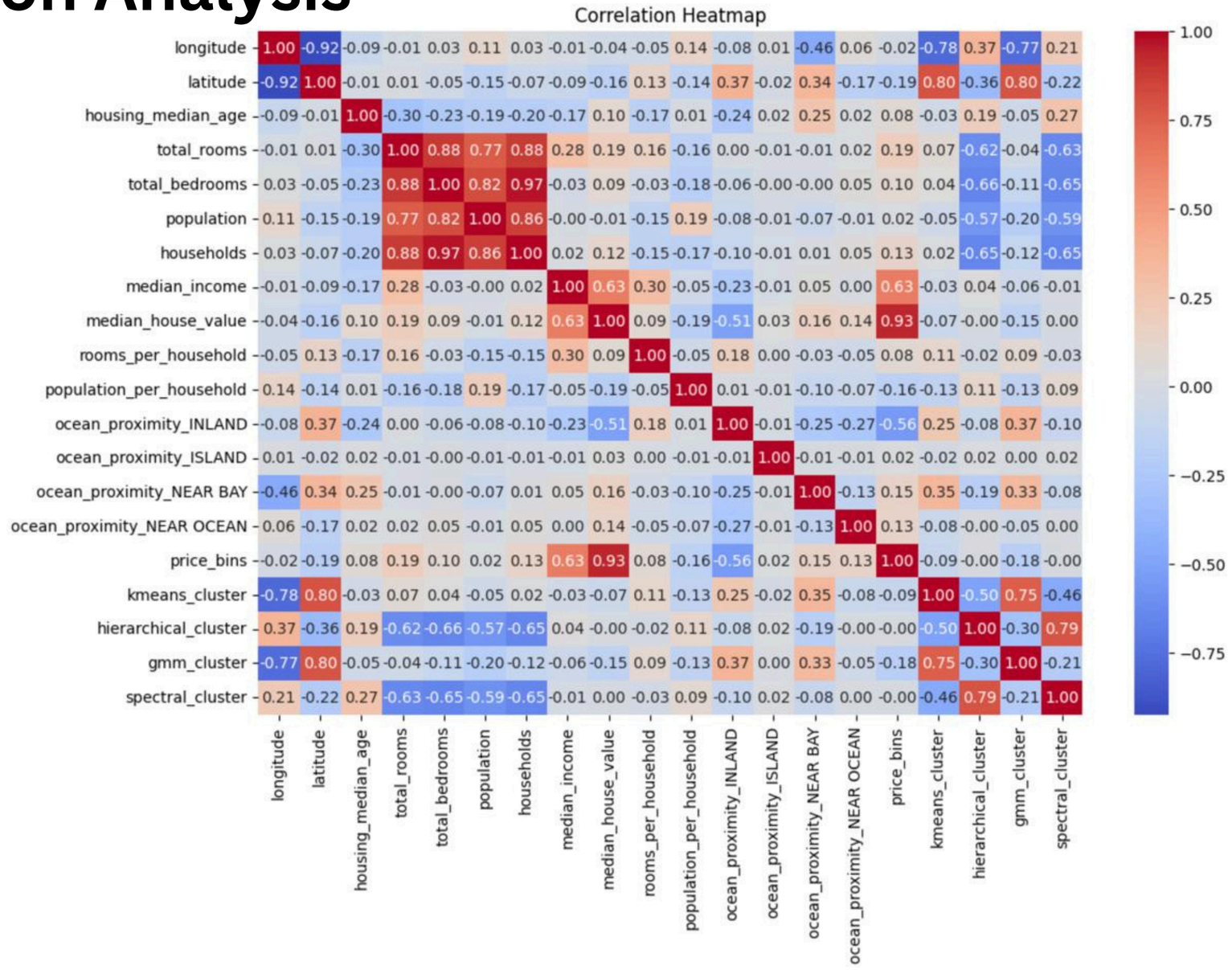
- The target variable in our dataset is median_house_value, which represents the median housing prices in different districts.




- The histogram visualizes the distribution of housing prices.The data shows a right-skewed distribution, indicating that most houses are priced below $200,000, with fewer houses in the higher price ranges.

- The peak of the distribution is around $150,000 to $200,000, suggesting that this is the most common price range for median house values.

# Correlation Analysis



- median_income (0.63): Shows the strongest positive correlation with median_house_value. This indicates that higher income levels are strongly associated with higher house prices.

- rooms_per_household (0.16): Weak positive correlation suggests that homes with more rooms per household slightly influence house value.

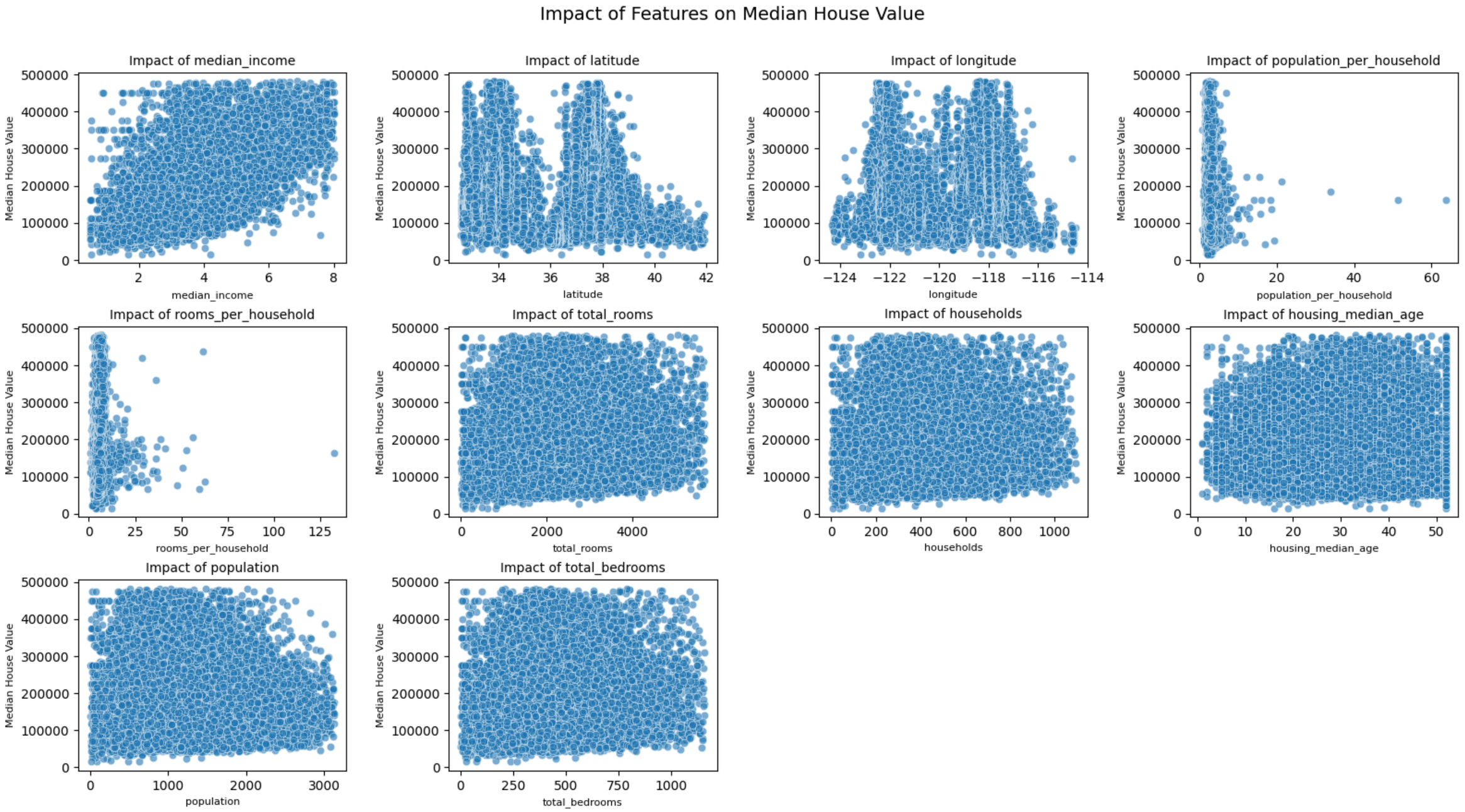
- ocean_proximity_NEAR BAY (-0.46): A negative correlation indicates houses near bays are priced lower.

# Mutual Information

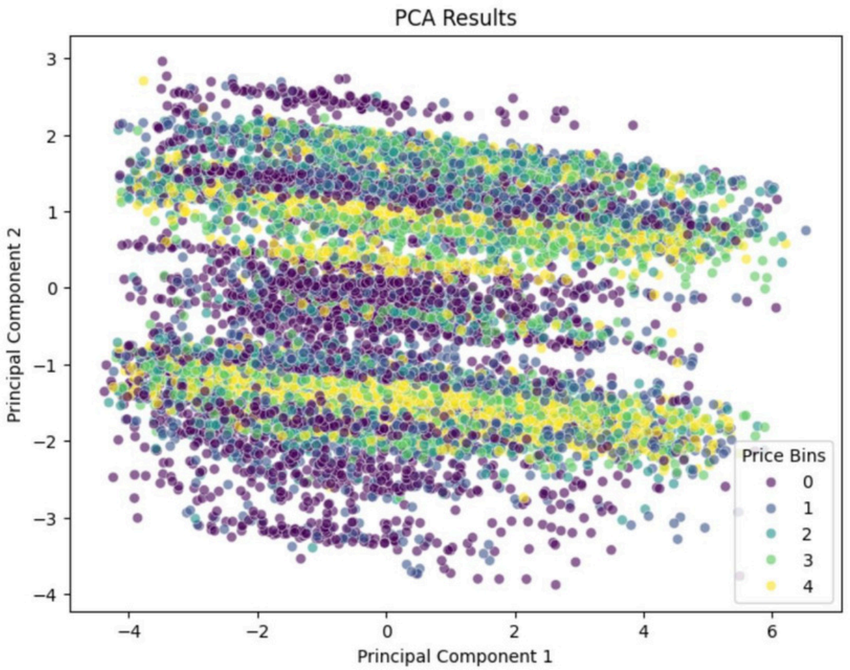
- median_income has the highest Mutual Information score (0.236), making it the strongest predictor of housing prices
- Features like latitude (0.069) and longitude (0.059) also significantly influence housing prices, capturing location-specific trends.

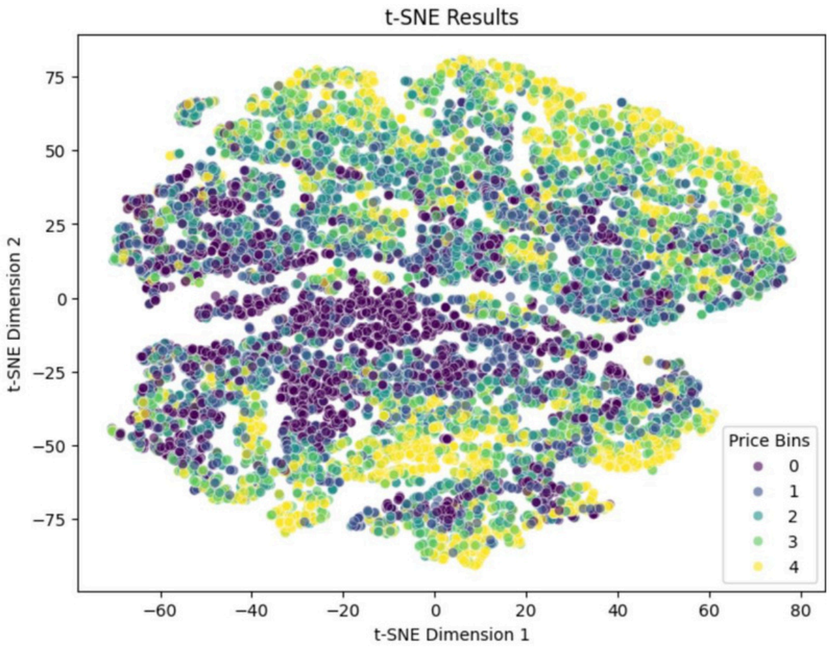
Mutual Information Scores with Binned Features and Target:

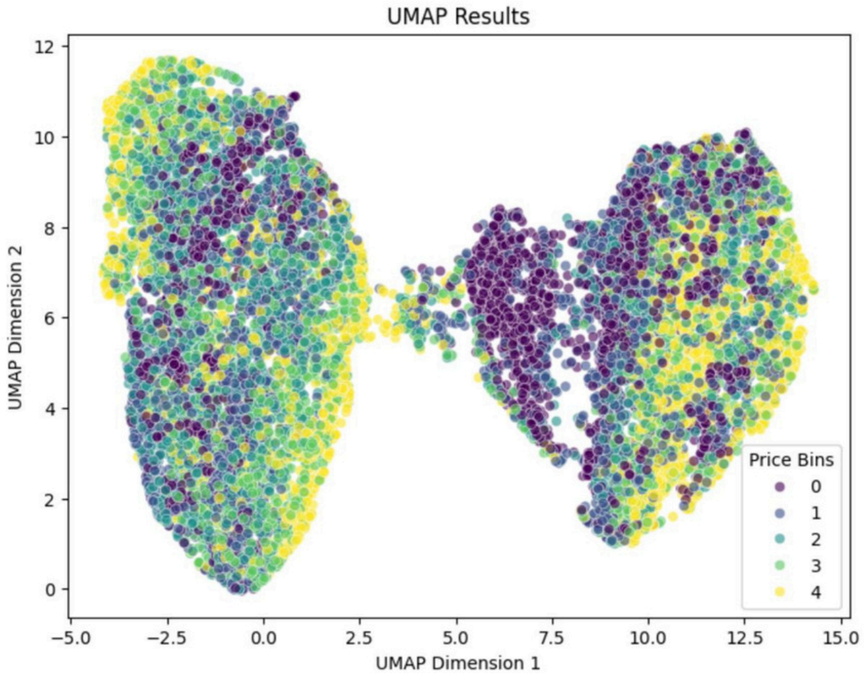
| | Feature | Mutual Information |
|---|---|---|
| 7 | median_income | 0.236216 |
| 1 | latitude | 0.068879 |
| 0 | longitude | 0.059122 |
| 9 | population_per_household | 0.044898 |
| 8 | rooms_per_household | 0.034240 |
| 3 | total_rooms | 0.021343 |
| 6 | households | 0.009282 |
| 2 | housing_median_age | 0.008517 |
| 5 | population | 0.006397 |
| 4 | total_bedrooms | 0.005943 |


Impact of Features on Median House Value
Impact of median_income
Median House Value
median_income
Impact of latitude
latitude
Impact of longitude
longitude
Impact of population_per_household
population_per_household
Impact of rooms_per_household
rooms_per_household
Impact of total_rooms
total_rooms
Impact of households
households
Impact of housing_median_age
housing_median_age
Impact of population
population
Impact of total_bedrooms
total_bedrooms

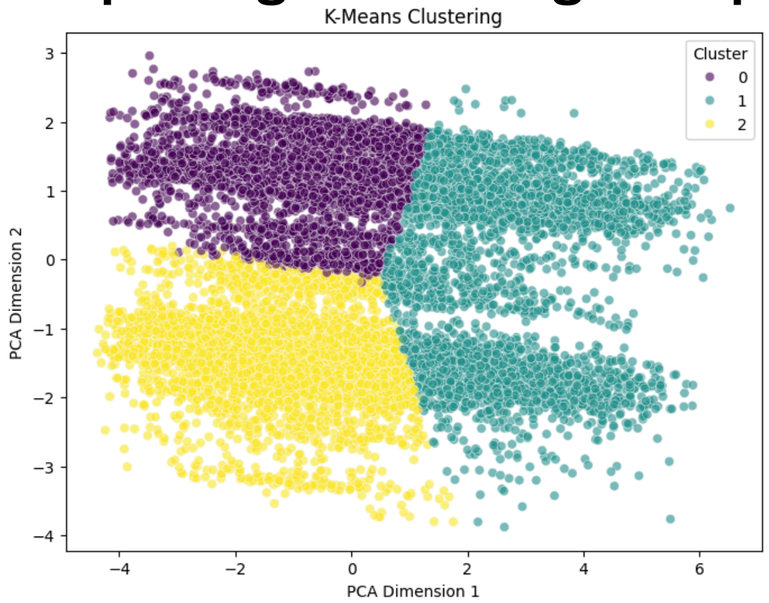
# Exploring Data through Visualization Techniques

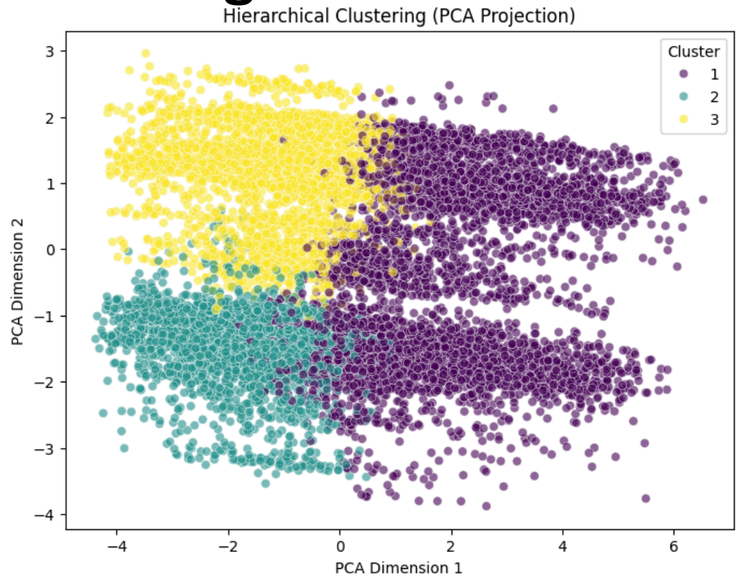


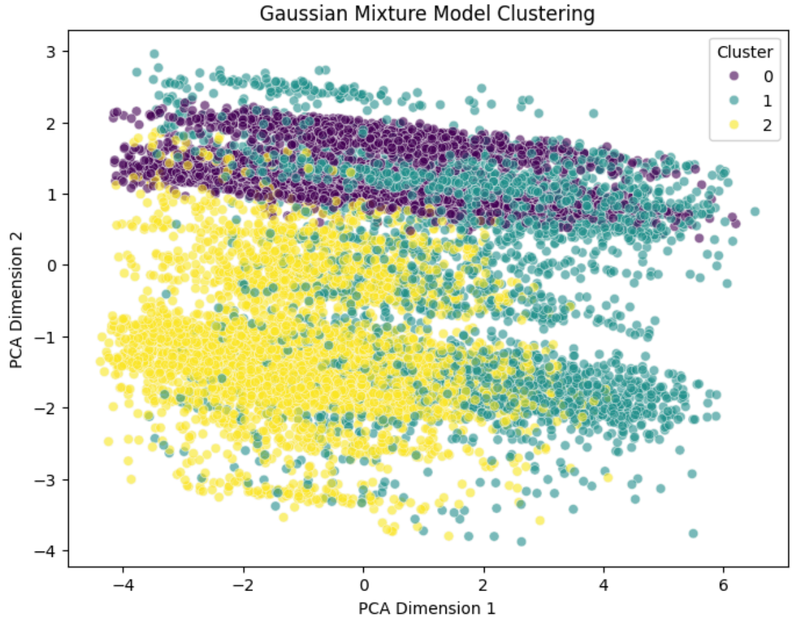



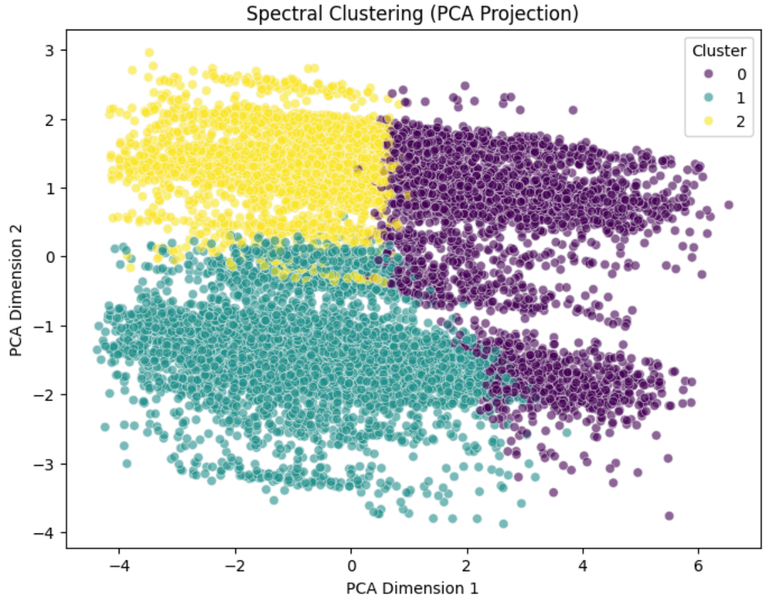



# Exploring Data Using Unsupervised Learning

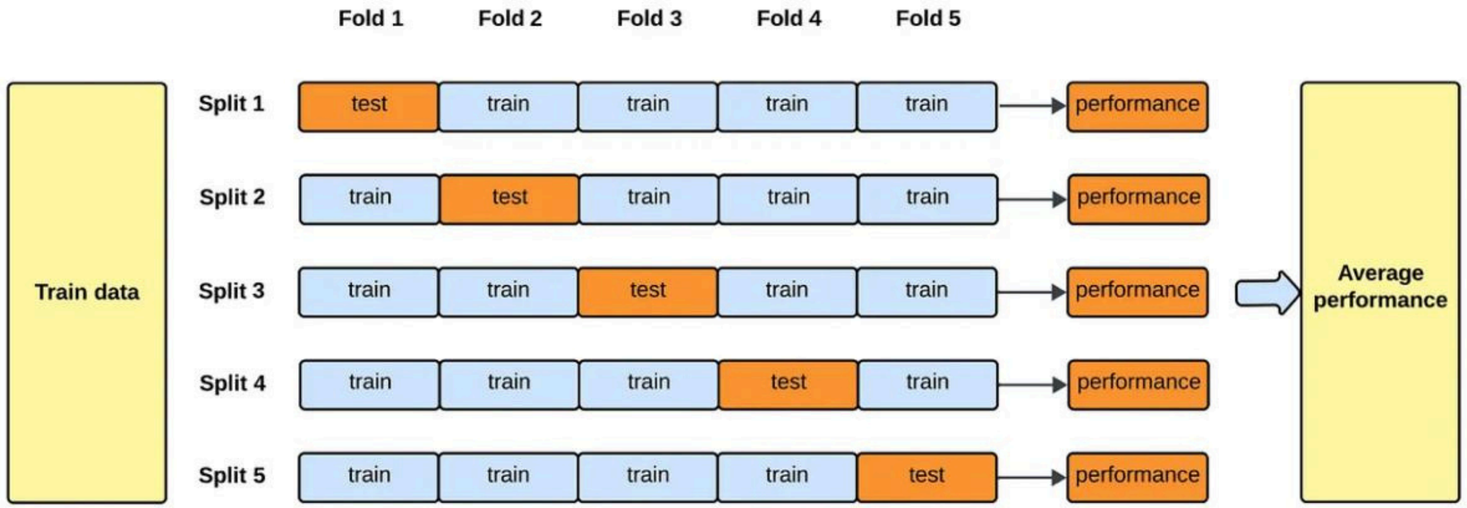


# Cross-Validation Strategy

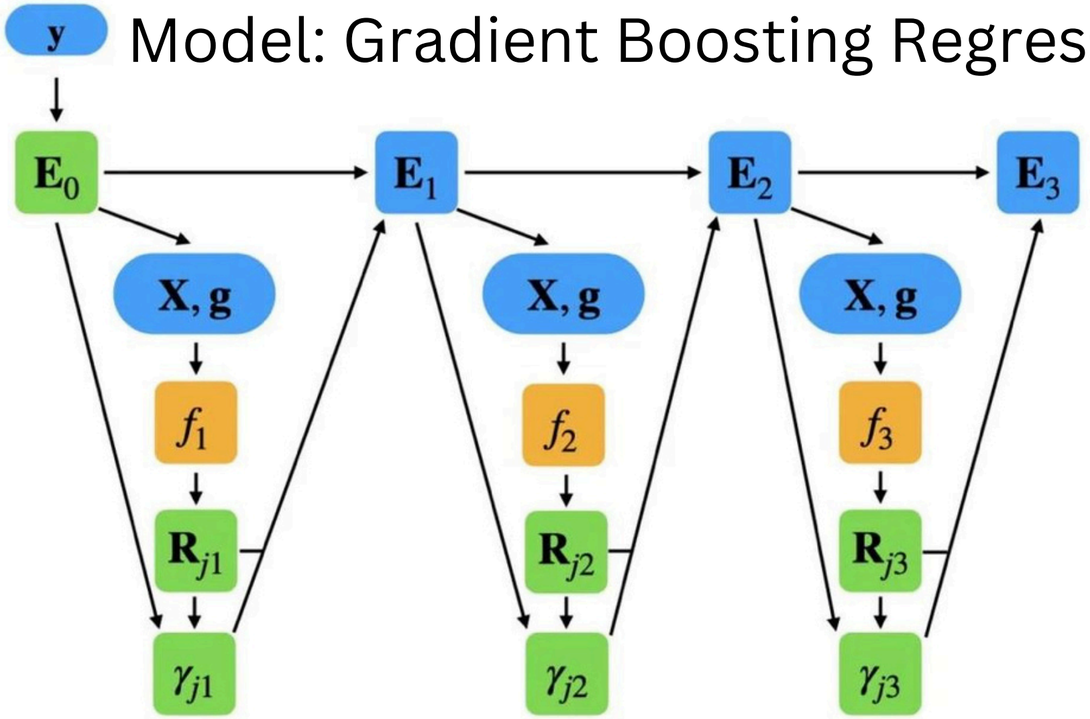



- K-Fold ensures that every data point is used for both training and validation, offering a robust and fair evaluation of the model's performance across the dataset.

- With a large dataset, K-Fold provides sufficient data for both training and validation in each fold, minimizing variance and ensuring reliable performance estimates.

# Model Training

Model: Gradient Boosting Regressor



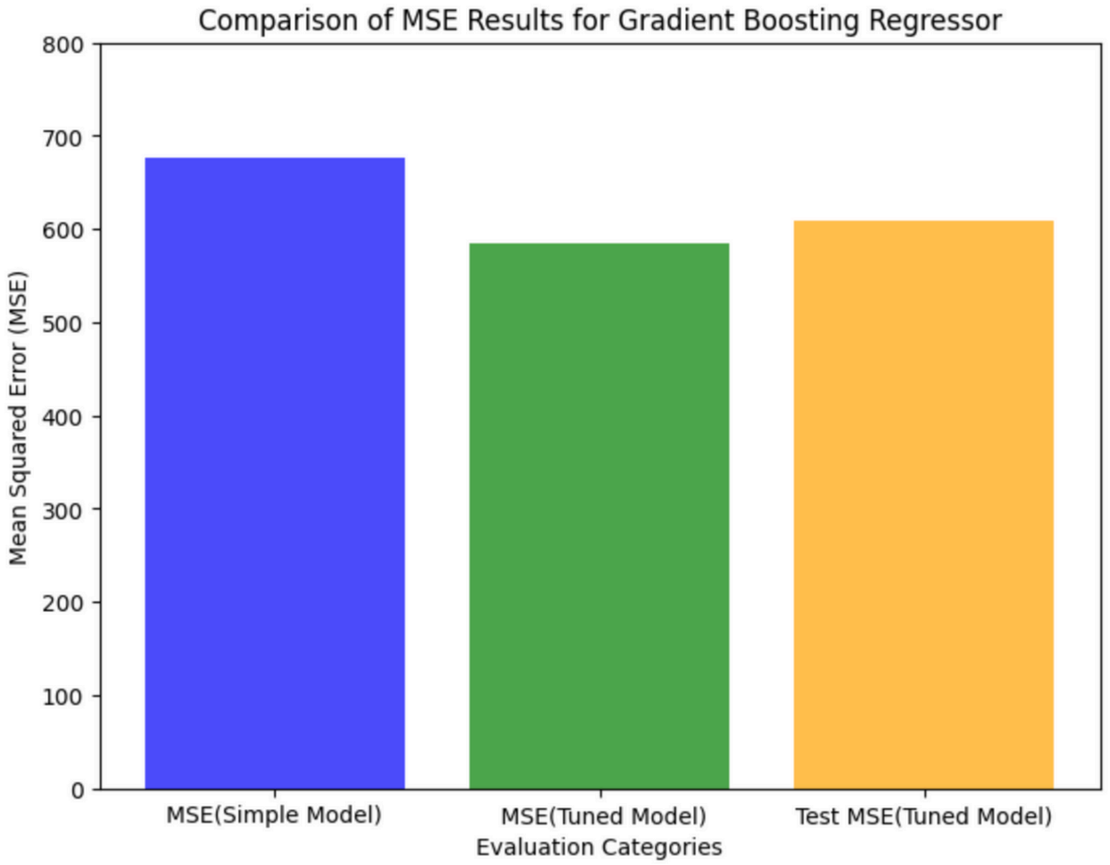

Initial Model Performance

```
Validation MSE (Simple Model): 677.06
Validation R^2 (Simple Model): 0.92
```

# Hyperparameter tuning using validation set

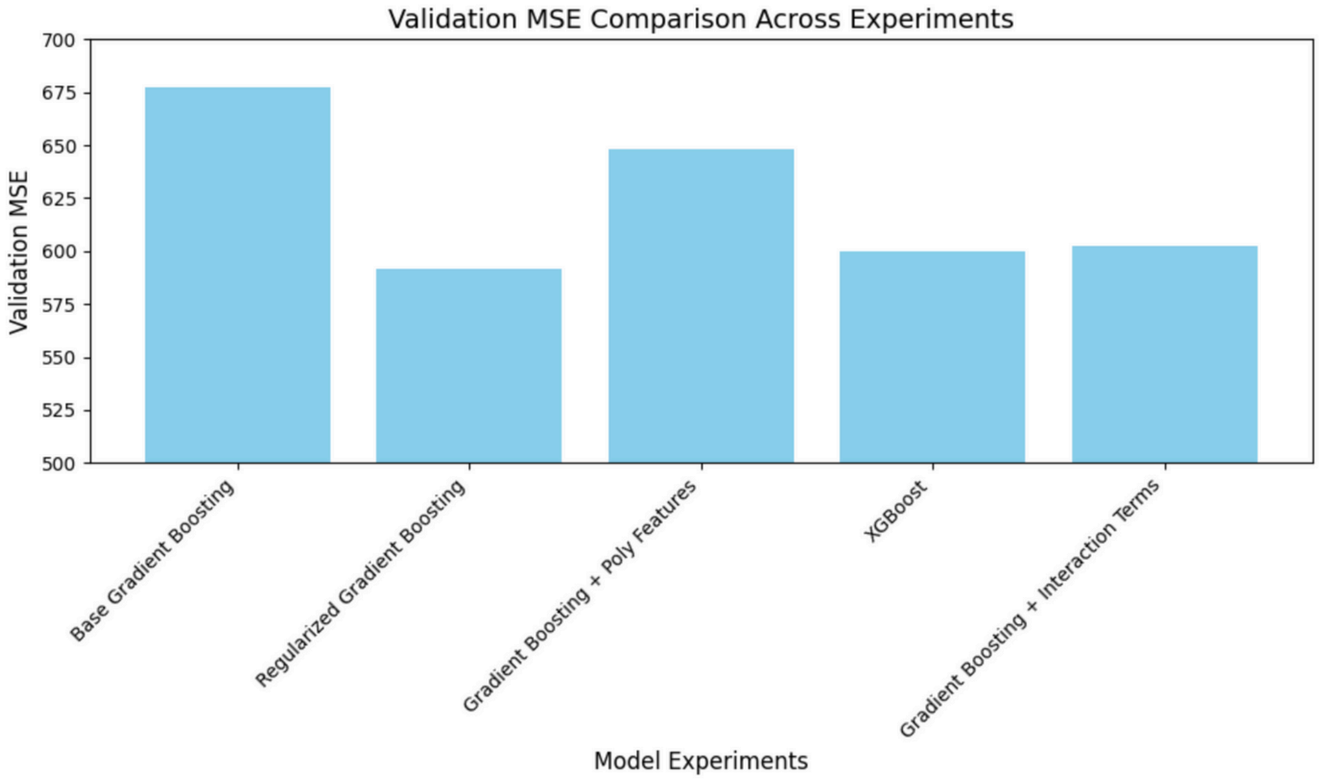
```
best_val_mse = np.inf
best_model = None

for learning_rate in [0.01, 0.05, 0.1]:
    for n_estimators in [50, 100, 200]:
        for max_depth in [3, 5]:
            model = GradientBoostingRegressor(
                learning_rate=learning_rate,
                n_estimators=n_estimators,
                max_depth=max_depth,
                random_state=42
            )
            model.fit(X_train, y_train)
            val_pred = model.predict(X_val)
            current_val_mse = mean_squared_error(y_val, val_pred)
            if current_val_mse < best_val_mse:
                best_val_mse = current_val_mse
                best_model = model
                best_params = {
                    'learning_rate': learning_rate,
                    'n_estimators': n_estimators,
                    'max_depth': max_depth
                }
```

```
Cross-Validation MSE Scores: [623.68049622 549.72394035 608.94628636 553.25093915 589.98016552]
Mean CV MSE: 585.12
Standard Deviation CV MSE: 29.48
```


Comparison of MSE Results for Gradient Boosting Regressor
Mean Squared Error (MSE)
800
700
600
500
400
300
200
100
0
MSE(Simple Model)
MSE(Tuned Model)
Test MSE(Tuned Model)
Evaluation Categories

# Experiments Conducted

1. Gradient Boosting with Regularization
2. Polynomial Features with Gradient Boosting
3. Using XGBoost Regressor
4. Feature Engineering with Gradient Boosting


Validation MSE Comparison Across Experiments
Validation MSE
700
675
650
625
600
575
550
525
500
Base Gradient Boosting
Regularized Gradient Boosting
Gradient Boosting + Poly Features
XGBoost
Gradient Boosting + Interaction Terms
Model Experiments


what's
next
?

ILLINOIS TECH

College of Computing

# Thank You