**Mall Customer Segmentation Model**

This project applies **K-Means Clustering** to segment mall customers based on their behaviors. Using the mall\_customers.csv dataset, the goal is to find optimal customer groupings to support targeted marketing strategies.

**Project Overview**

* **Technique**: K-Means Clustering
* **Dataset**: Mall customers dataset (includes features like age, annual income, and spending score)
* **Goal**: Group customers into meaningful clusters for marketing insights
* **Evaluation Metric**: Silhouette Score (used to assess clustering performance)

**Key Results**

* Multiple values of k were tested to evaluate cluster quality using silhouette scores.
* The **optimal number of clusters was found to be 6**, based on the highest silhouette score.
* KMeans++ initialization was used to smartly position centroids by considering data distribution patterns.

| **Clusters (k)** | **Silhouette Score** |
| --- | --- |
| 3 | 0.312 |
| 4 | 0.405 |
| 5 | 0.397 |
| **6** | **0.452** |
| 7 | 0.437 |
| 8 | 0.374 |

**Visualizations**

* Silhouette Scores plotted against different values of k to choose the optimal number of clusters.
* Cluster visualization to interpret customer segmentation visually.

**Technologies Used**

* Python 3
* Pandas, NumPy
* Matplotlib, Seaborn
* Scikit-learn (for KMeans, silhouette\_score)

**How to Run**

1. Clone this repository or download the notebook.
2. Make sure you have Python and the required libraries installed.
3. Load the dataset: mall\_customers.csv
4. Run the notebook classification\_model.ipynb or mall\_segmentation.ipynb
5. Experiment with different values of k or use init='k-means++' as shown:

from sklearn.cluster import KMeans

kmeans = KMeans(n\_clusters=6, init='k-means++', random\_state=42)

kmeans.fit(X)

**Conclusion**

The use of KMeans++ improved centroid initialization, and silhouette analysis helped in selecting the best k. With k=6, the clustering results were most consistent and meaningful for customer segmentation.

**Folder Structure**

📁 mall-customer-segmentation/

│

├── mall\_customers.csv

├── mall\_segmentation.ipynb

├── README.md

**✍️ Author**

Ali Reza Mohseni

# Real Estate Price Prediction

This project aims to predict real estate property prices using various regression models including \*\*Linear Regression\*\*, \*\*Decision Tree\*\*, and \*\*Random Forest\*\*. It uses the `scikit-learn` library and demonstrates essential machine learning processes like training/testing, model evaluation, and serialization with `pickle`.

---

## Project Structure

- `final.csv` - The dataset used for training and evaluation.

- `model\_training.ipynb` - Jupyter Notebook with full model training and evaluation.

- `RE\_Model` - A serialized (`.pkl`) version of the trained Decision Tree model.

- `tree.png` - Decision tree visualization saved as an image.

- `README.md` - Project overview and instructions.

---

## Models Used

### ✅ Linear Regression

- Baseline model

- Trained on feature-engineered dataset

- Evaluated using \*\*Mean Absolute Error (MAE)\*\*

### ✅ Decision Tree Regressor

- `max\_depth=3`, `max\_features=10`

- Visualized using `sklearn.tree.plot\_tree`

- Exported with `pickle`

### ✅ Random Forest Regressor

- `n\_estimators=200`

- Criterion: `absolute\_error`

- Improved generalization over single tree

---

## Libraries Required

- `pandas`

- `numpy`

- `matplotlib`

- `scikit-learn`

- `pickle`

You can install all dependencies using:

```bash

pip install pandas numpy matplotlib scikit-learn

Authur: Ali Reza Mohseni