Predictive Modeling of Housing Prices in California

Presented by: Shaik Azra Anisha

Artificial Intelligence & Machine Learning Capstone

June 2025

Dataset Used: California Housing Dataset (sklearn)

Problem Statement: Predicting Median House Values

This project aims to build a regression model to predict median house values in California districts using the California Housing dataset. The model will leverage socioeconomic and geographic factors to understand their impact on housing prices.

Objective

Predict median house values in California districts.

Data Source

California Housing dataset (sklearn).

Key Factors

Socioeconomic and geographic variables.

Key Features and Data Overview

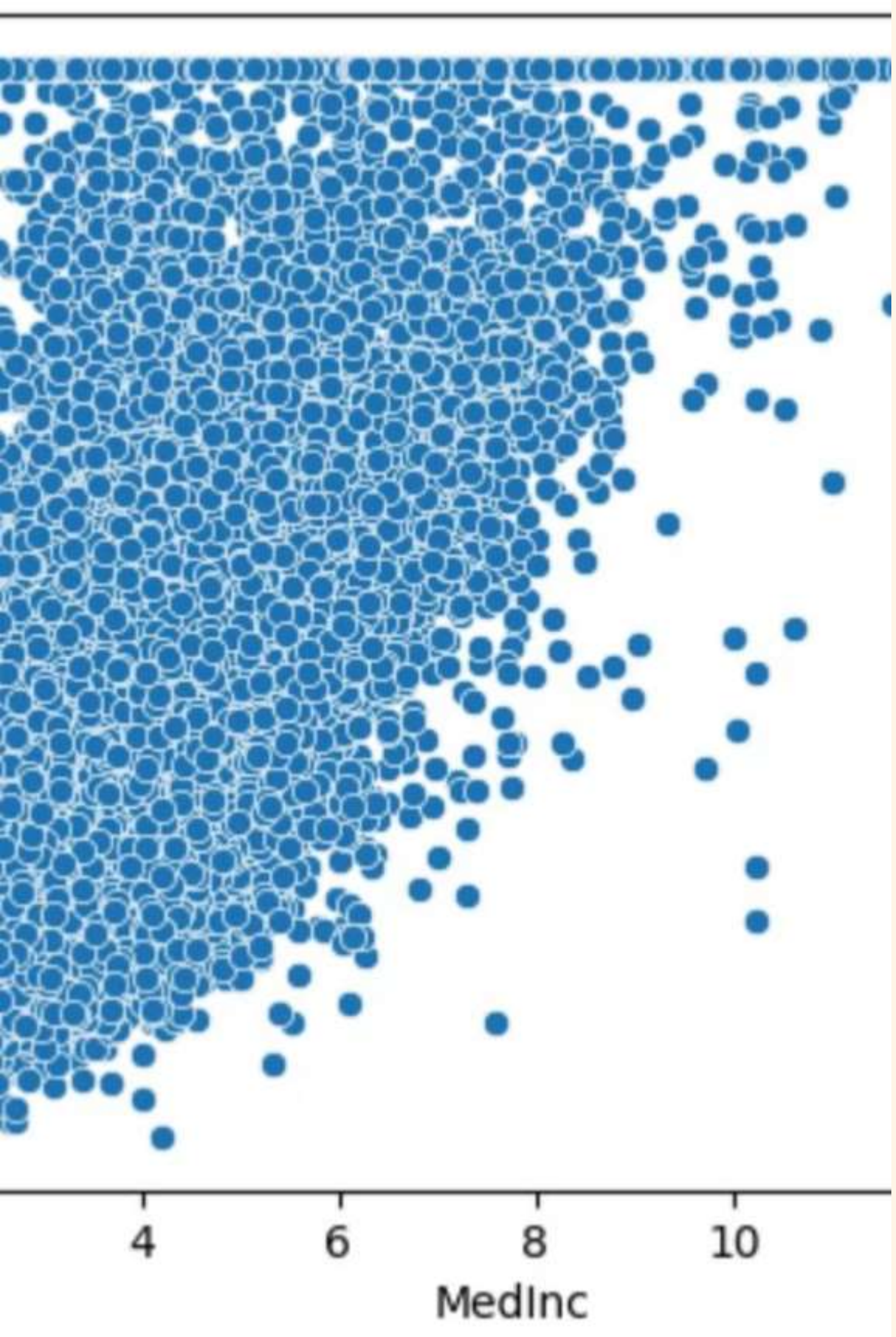
Selected Features

- MedInc (Median Income)
- HouseAge (Average age of houses)
- AveRooms (Average number of rooms)
- AveOccup (Average occupancy)
- Latitude & Longitude (Location)

Dataset Snapshot

MedInc	HouseAge	AveRooms
8.3252	41.0	6.984127
8.3014	21.0	6.238137

The dataset contains 20,640 entries with 9 columns, including 8 features and 1 target variable (MedHouseVal). No missing values were found, and all data types are numerical.



Exploratory Data Analysis & Preprocessing

Our EDA involved checking for missing values and duplicates, confirming data types, and visualizing feature correlations.

Data Integrity

No missing values or duplicate rows were found in the dataset.

Data Types

All features were confirmed to be numerical, as expected.

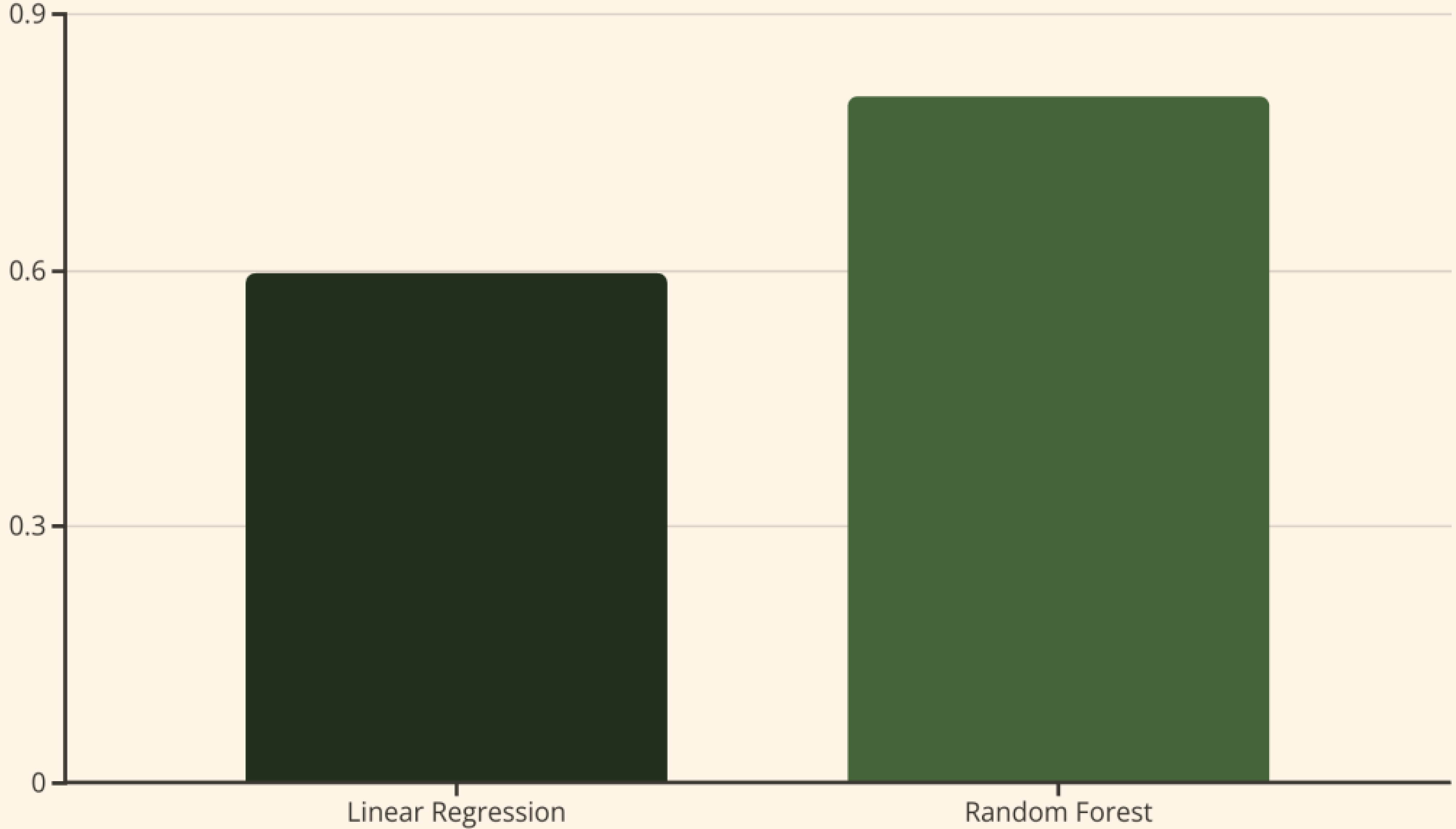
Visualizations

Heatmaps and scatter plots were used to analyze feature correlations, especially between Median Income and Median House Value.

Model Building & Evaluation

We trained Linear Regression and Random Forest Regressor models, comparing their performance using MSE, MAE, and R² scores. Random Forest showed superior performance.

Model Comparison (R² Score)

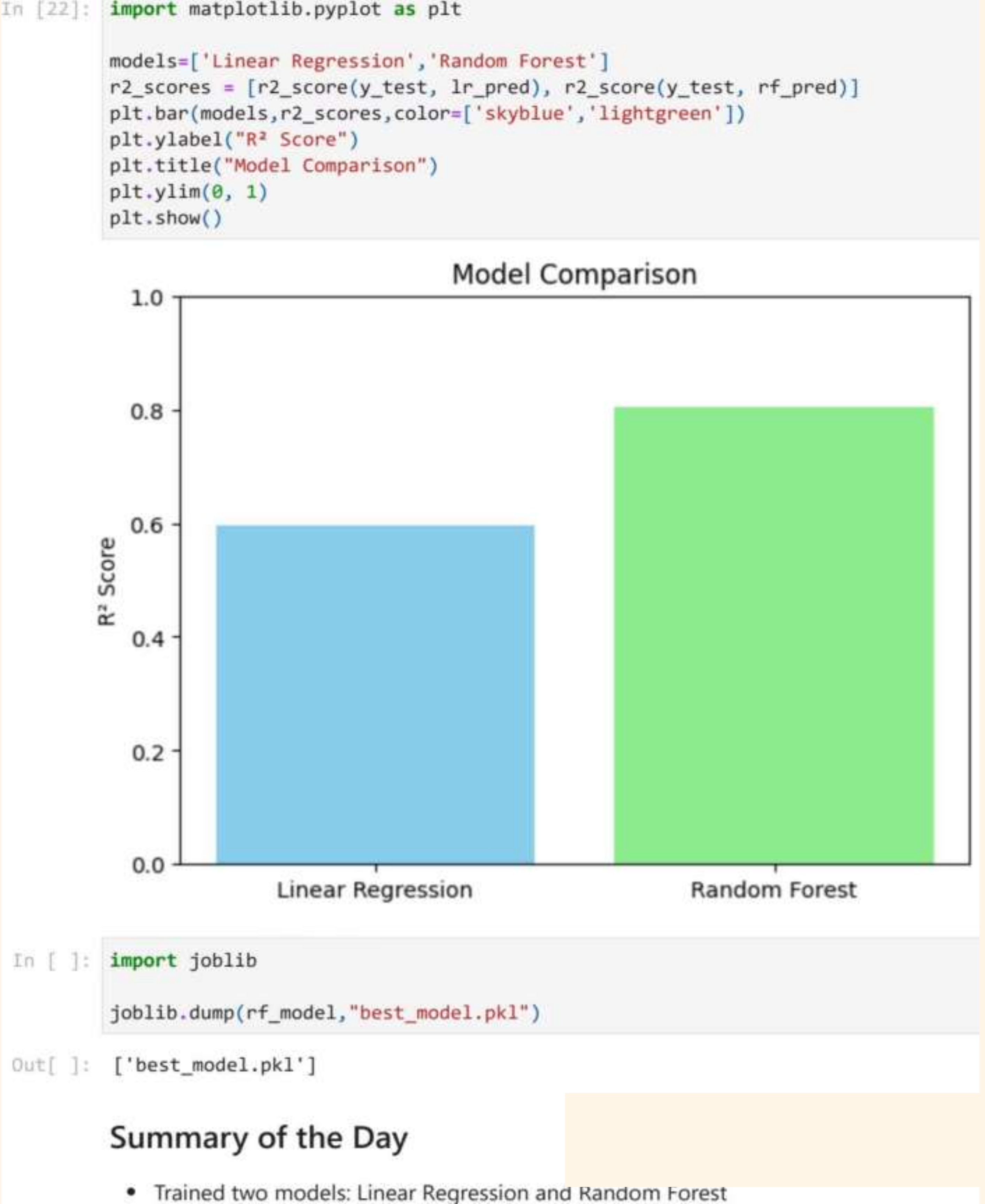


Key Findings

- Random Forest outperformed Linear Regression.

- Final R^2 score of ~0.80 on the test set, with a cross-validation R^2 of ~0.78.

- GridSearchCV was used for hyperparameter tuning.



In [22]:

```
import matplotlib.pyplot as plt

models=['Linear Regression','Random Forest']
r2_scores = [r2_score(y_test, lr_pred), r2_score(y_test, rf_pred)]
plt.bar(models,r2_scores,color=['skyblue','lightgreen'])
plt.ylabel("R² Score")
plt.title("Model Comparison")
plt.ylim(0, 1)
plt.show()
```

In [ ]:

```
import joblib

joblib.dump(rf_model,"best_model.pkl")
```

Out[ ]:

```
['best_model.pkl']
```

Summary of the Day

- Trained two models: Linear Regression and Random Forest thank

YOU