# Machine Learning Project

# On

# Predicting House Prices by

# Using Different Models


Submitted By:

Name - Chhaya Gupta

Admission No.- 24MT0524

Course - M.Tech Data Analytics

# Report: Predicting California Housing Prices

## 1. Introduction

This project aims to predict housing prices in California using various machine learning models. The dataset used is the California Housing Dataset, which contains features such as median income, house age, and population. The models implemented include Linear Regression, Decision Tree Regressor, and Neural Networks, with a focus on model evaluation and comparison.

## 2. Data Preprocessing & Exploratory Data Analysis (EDA)

- Dataset Loaded using fetch_california_housing().
- Data Cleaning: No missing values were found.
- Feature Scaling: StandardScaler was applied to normalize numerical features.
- EDA Techniques Used:
    - Correlation Matrix to identify relationships between variables.
    - PairPlot to visualize feature interactions.
    - Histograms to analyze target variable distribution.
    - BoxPlots to detect outliers.
    - ScatterPlots for feature-target relationship analysis.

## 3. Machine Learning Models

### 3.1 Linear Regression

- Model trained using LinearRegression().

- Performance:
    - Mean Squared Error (MSE): 0.56
    - R-squared: 0.57 (Poor performance)
- Due to low accuracy, alternative models were explored.

## 3.2 Decision Tree Regressor

- Model trained using DecisionTreeRegressor().
- Performance:
- MSE: 0.49
    - R-squared: 0.62
- Showed an improvement over Linear Regression.
- Hyperparameter Tuning: GridSearchCV was used to find the best model parameters.
- Optimized Decision Tree Performance:
- MSE: 0.39
    - R-squared: 0.70

## 3.3 **Neural Network**

- Implemented using PyTorch with multiple layers.
- Initial Neural Network Performance:
- MSE: 0.27
    - R-squared: 0.80 (Best model so far)
- Enhanced Model:

- Used dropout and batch normalization to check for further improvements.
- However, no significant performance gain was observed.

## 4. Model Comparison

| Model | MSE | R-Squared |
|---|---|---|
| Linear Regression | 0.56 | 0.57 |
| Decision Tree | 0.49 | 0.62 |
| Optimized Decision Tree | 0.39 | 0.70 |
| Neural Network | 0.27 | 0.80 |

## 5. Visualizing Model Performance

- Bar Plots comparing MSE and R-Squared for all models.
- Scatter Plots for actual vs predicted values (Decision Tree & Neural Network).
- Feature Importance Bar Plot for the Decision Tree Model.

## 6. Conclusion

- Neural Network performed the best, achieving an R-squared of 0.80.
- Decision Tree provided competitive results after hyperparameter tuning.
- Linear Regression was the weakest performer.
- Future Work: Further improvements can be explored using additional feature engineering and advanced deep learning architectures.

This project highlights the impact of choosing the right model and tuning its parameters for optimal performance in predictive tasks.