CALIFORNIA HOUSING PRICE PREDICTION

OVERVIEW

This project develops a deep learning model to predict housing prices across California using neighbourhood-level demographic and geographic information. The model predicts the median house value of a given census block based on a range of socioeconomic and environmental factors. All of which are in tabular form.

METHODOLOGY

The dataset used was obtained from scikit-learn's California Housing dataset, which contains over 20,000 observations derived from the 1990 U.S. Census data. Each observation represents a small block of houses in California and includes features like average number of rooms, population, and location. The target label is median house value.

Following some subpar training runs, several engineered features were introduced to improve performance: geographical encodings and population density. This improved the model's ability to identify spatial and socioeconomic patterns. Skewed variables such as population and average room count were log-transformed to make their distributions more uniform.

Data was split 70/15/15 between training/validation/testing respectively.

MODEL DETAILS

The project uses Multilayer Perceptron (MLP): a form of fully connected neural network suited for tabular data.

Throughout development, multiple aspects of the model and training pipeline were refined. Various network depths, activation functions, and regularization methods were tested to find the most stable configuration. The final architecture adopted the SiLU activation function (a smoother alternative to ReLU) and dropout regularization to prevent overfitting. Optimization was performed using the Adam optimizer with a fixed learning rate of 5×10^{-4}. The model was trained for 300 epochs with early experimentation also exploring stochastic weight averaging (SWA) and variable patience settings for learning-rate adjustment.

To ensure reproducibility and model robustness, multiple training runs were performed using different random seeds, confirming the model's consistency across runs. The model with the lowest validation error was selected for final testing.

EVALUATION

Below are the performance metrics used to calculate the efficacy of the model during validation and testing.

Mean Absolute Error - average $ difference between predicted and true house value	Root Mean Squared Error - punishes larger errors more, highlighting potential outliers

RESULTS

Below are the results from testing. The model instantiation used was from training run 5, which had the highest performing validation metrics.

test_MAE: 0.3792	test_RMSE: 0.5515