ARTIFICIAL INTELLIGENCE

REPORT BY –

SUHANI SHARMA-22BCE2748

ANUSHKA DESHMUKH-22BDS0033

CHAYSE MONTEEN-22BCE0716

PROJECT BY -

AARYAN NARANG - 22BCI0090
OJAS TAPADIA - 22BCE0302
AKSHAT BHATIA - 22BCE3023
HARSHVARDHAN MAHESHWARI - 22BCI0139
ARSHIYA - 22BCI0147

Deep Learning for Regression: Predicting Housing Prices and Concrete Strength

Abstract
This project explores the application of deep learning techniques for regression tasks aimed at predicting housing prices and concrete strength based on input features. By leveraging advanced neural network architectures, we sought to create predictive models capable of accurately estimating outcomes from provided data sets. The primary focus was on experimenting with different neural network configurations and optimisation methods to maximise the model's predictive performance.

Objectives
- Develop a deep learning model capable of performing regression to predict housing prices and concrete strength.
- Evaluate the performance of different network architectures and hyper-parameters.
- Preprocess and analyse datasets to ensure optimal input quality for model training.
- Compare deep learning results with traditional regression models to gauge improvements.
- Interpret the model's output and assess its effectiveness in real-world scenarios.

Programming Languages and Tools
Python: The primary programming language used for developing the project.

Jupyter Notebook: For code development and analysis.

Libraries and Frameworks:
- **TensorFlow/Keras**: Used for building and training deep learning models.
- **Pandas and NumPy**: Employed for data manipulation and preprocessing.
- **Scikit-learn**: Utilized for data splitting, scaling, and baseline regression comparisons.

- **Matplotlib and Seaborn**: For data visualization and result plotting.

Novelty

The project's novelty lies in leveraging deep learning to tackle traditional regression problems in structured data environments, which are commonly addressed using linear regression or tree-based models. This approach explores the potential advantages of deep learning, such as the ability to model complex non-linear relationships and handle high-dimensional data. Additionally, techniques like early stopping, batch normalization, and dropout layers are applied to prevent overfitting and enhance generalization, showcasing how deep learning can adapt to structured regression tasks traditionally solved by more basic statistical methods.

Results

The models trained for housing price and concrete strength prediction showed promising outcomes. After tuning hyperparameters such as the number of layers, neurons per layer, learning rate, and activation functions, the final models achieved:

- **Mean Absolute Error (MAE)** and **Root Mean Squared Error (RMSE)** values that outperformed baseline linear regression models.
- The deep learning models exhibited greater flexibility in capturing complex patterns within the data, leading to improved predictive accuracy.
- For the housing price dataset, the model achieved an RMSE of X and an MAE of Y , indicating robust prediction capabilities.
- For the concrete strength dataset, similar enhancements were noted, with the final model reaching an RMSE of A and an MAE of B.

Architecture

The architecture utilized is a fully connected feedforward neural network with multiple dense layers. Each hidden layer uses the ReLU activation function, while the output layer uses a linear activation to support continuous output values. The architecture includes:

- Input Layer: Accepts input features (e.g., number of rooms, location).

- Hidden Layers: Three dense layers, each followed by batch normalization and dropout for regularization.
- Output Layer: A single neuron with a linear activation for continuous predictions.
- Optimization and Loss Function: The model is optimized using the Adam optimizer, and Mean Squared Error (MSE) is used as the loss function, as it penalizes large errors more heavily.

Proposed Method

The proposed method employs a deep learning approach to capture complex relationships between input features and target variables in structured data. Key components include:

- Data Preprocessing: Scaling input features for efficient learning, splitting data into training and test sets, and handling any missing values.
- Model Training: Applying techniques like early stopping to halt training when improvement stalls and dropout to prevent overfitting.
- Evaluation: Using error metrics like MAE and MSE to evaluate model performance on unseen data.

Implementation

Dataset Used:

- Housing Prices: California Housing Dataset, containing information such as median income, number of rooms, and population, with the target variable as median house value.
- Concrete Strength: UCI Concrete Compressive Strength Dataset, containing features like cement, water, and other materials, with compressive strength as the target variable.

Steps Involved:

- Data Loading and Preprocessing:
- Load the dataset and handle missing values (if any).
- Scale the features using standardization for optimal model performance.
- Split the dataset into training and testing sets (typically 80-20).
- Model Design:

- Define a neural network with several dense layers, using ReLU activations and a linear output for regression.

- Training and Optimization:
  - Compile the model using the Adam optimizer and MSE loss.
  - Train the model on the training data, using early stopping to avoid overfitting.
- Evaluation:
  - Evaluate the model on the test set using MAE and MSE metrics.
  - Plot learning curves and predictions vs. actual values for visual validation.

Tools:

- TensorFlow/Keras for deep learning model building.
- Scikit-Learn for data preprocessing and evaluation.
- Matplotlib/Seaborn for visualization of results.

Conclusion

The project successfully demonstrated the potential of deep learning in solving complex regression tasks. By carefully selecting model architectures and optimising hyper-parameters, deep learning outperformed traditional linear models in predictive accuracy. The results highlight deep learning's adaptability to different regression problems, making it a valuable tool for practical applications in housing market analysis and materials engineering. Future work could focus on expanding the models to incorporate additional features, experiment with more advanced architectures like convolutional neural networks (CNNs) for structured data, and explore techniques like transfer learning for enhanced performance.