## **Artificial Intelligence and Machine Learning**

Project Report

Semester-IV (Batch-2022)

**Case Study**: - Predicting Addition with Linear Regression

[Url:-](about:blank) https://drive.google.com/file/d/1lMxT4tVnuIXzAAic7tXJomrLzKTF22tb/view?usp=sharing

A red and white sign

Description automatically generated with low confidence

**Supervised By: Submitted By:**

Rajeev Thakur Vipash Arora

Roll Number: -2210990964

Group - 14

**Department of Computer Science and Engineering**

**Chitkara University Institute of Engineering & Technology,**

## **Chitkara University, Punjab**

**\*\*Case Study: Predicting Addition with Linear Regression\*\***

* **Objective:**

The objective of this analysis is to predict the addition of two numbers using a Linear Regression model.

* **\*\*Methods Used:\*\***
  + Pandas library
  + Matplotlib library
  + Scikit-learn library
  + Joblib library
  + Tkinter library
* **Methods:**

- `read\_csv()`: Reads a CSV file and converts it into a data frame.

- `scatter()`: Plots a scatter plot.

- `train\_test\_split()`: Splits the dataset into train and test sets.

- `LinearRegression()`: Initializes the Linear Regression model.

- `fit()`: Trains the Linear Regression model.

- `score()`: Calculates the coefficient of determination (R^2) of the prediction.

- `predict()`: Predicts the target variable using the trained model.

- `DataFrame()`: Creates a DataFrame from specified data.

- `dump()`: Saves the model to a file.

- `load()`: Loads the model from a file.

- `Tk()`: Initializes the Tkinter window.

- `Label()`: Creates a label widget.

- `Entry()`: Creates an entry widget for user input.

- `Button()`: Creates a button widget with a callback function.

* **\*\*Analysis Steps:\***\*

1. \*\*Data Loading and Visualization:\*\*

- Read the data from the CSV file using `read\_csv()` method.

- Visualize the relationship between features and target variable using scatter plots.

2. \*\*Data Preparation:\*\*

- Store feature matrix in `X` and response (target) in vector `Y`.

3. \*\*Train/Test Split:\*\*

- Split the data into train and test sets using `train\_test\_split()` method.

4. \*\*Model Building and Evaluation:\*\*

- Initialize the Linear Regression model.

- Train the model using the training data.

- Evaluate the model's performance on both training and test sets using `score()` method.

- Predict the target variable for test data using `predict()` method.

- Create a DataFrame to compare actual and predicted values.

5. \*\*Save and Load Model:\*\*

- Save the trained model to a file using `dump()` method from Joblib library.

- Load the saved model using `load()` method for future predictions.

6. \*\*Create User Interface with Tkinter:\*\*

- Create a Tkinter window for user input.

- Add entry fields for users to input two values.

- Implement a button to trigger the addition prediction.

- Display the predicted addition result.

* **\*\*Conclusion:\*\***

The Linear Regression model trained on the provided dataset achieved satisfactory results in predicting addition. The model can be utilized for further predictions and integrated into a user-friendly interface using Tkinter.

This structured approach facilitates a comprehensive understanding of the steps involved in building, evaluating, and deploying a machine learning model for predicting addition.