# Training Day 13 Report

**Date:** 16 July 2025
**Topic:** Machine Learning Fundamentals - Linear Regression, Error Analysis, and KNN (K-Nearest Neighbors)

**Summary of the Day:**
Today's training session was focused on understanding and implementing key machine learning algorithms using Python. The mentor began by revisiting core data analytics concepts and moved into detailed discussions on supervised learning, particularly focusing on **Linear Regression** and **KNN (K-Nearest Neighbors)**. **Morning Session:**
The morning session began with an in-depth recap of **NumPy** and **Pandas**, explaining their interrelation and importance in handling multidimensional data arrays and dataframes. We explored how libraries like **scikit-learn** are built upon NumPy and how it provides ready-made functions for model training and testing. The session included practical demonstrations using datasets such as the **California Housing** and **Height-Weight Dataset** from Kaggle.

We implemented **Linear Regression** from scratch using scikit-learn and learned the following stages of model development: Importing libraries and loading datasets. Pre-processing and cleaning the data (checking for null values and understanding distributions). Splitting the dataset into training and testing subsets (typically 80/20 ratio). Training the model using model.fit(X_train, Y_train) and evaluating results using metrics like MSE (Mean Squared Error) and $R^2$ (Coefficient of Determination). We discussed the interpretation of the linear equation $\mathbf{Y = mX + C}$, where **m** is the slope and **C** the intercept, and plotted the regression line using Matplotlib. Practical exercises involved predicting house prices and body weights based on height data. **Afternoon Session:** The afternoon session covered the **K-Nearest Neighbors (KNN)** algorithm. The trainer introduced KNN as a simple yet powerful supervised learning algorithm applicable to both regression and classification tasks. We explored the working principle of KNN — how it classifies new data points based on their proximity to 'K' nearest neighbors using different distance metrics: Euclidean Distance Manhattan Distance Minkowski Distance Cosine Similarity Visualization exercises were performed to demonstrate classification of sample data points into distinct classes based on majority voting. The concept of choosing the optimal value of **K** was discussed along with the problems of **overfitting** (low K) and **underfitting** (high K).

**Key Learnings:** Understanding the difference between Linear and Polynomial Regression. Computation and interpretation of MSE, SSE, and $R^2$ values. Visualization and interpretation of training and test data results. Comparison between Regression and Classification approaches. Introduction to advanced regression models like Ridge and Lasso Regression. **Practical Implementation:**
Each participant implemented linear regression and KNN models in Google Colab using real datasets. The practical included: Loading and cleaning CSV files in Pandas. Splitting data into train/test subsets using train_test_split(). Plotting regression lines and scatter plots with Matplotlib. Testing prediction accuracy using scikit-learn's metrics module. Evaluating and tuning KNN using various K values to find the optimal model. **Conclusion:**
Day 13 provided an essential understanding of how machine learning models are structured, trained, and validated. The focus on practical coding and conceptual clarity helped reinforce understanding of key ML algorithms. The session concluded with a brief discussion on **model optimization techniques** and an introduction to **Ridge and Lasso Regression** to be covered in the upcoming session. **Overall Experience:**
The day was informative and engaging. The mix of theoretical explanation and coding practice deepened the understanding of machine learning workflow and data-driven modeling.