CSE 421 – Embedded Machine Learning
Q1: Estimating Future Temperature Values

Can Sarp Aridogan
Ahmet Renda Sendag

1. Introduction

In this study, future indoor temperature values are estimated using an embedded machine learning approach. The application follows the methodology presented in Embedded Machine Learning with Microcontrollers (Section 7.6). The goal is to train a lightweight regression model on a personal computer and deploy it on an STM32 microcontroller.

2. Dataset

The SML2010 dataset is used in this work. Indoor temperature measurements (T1) are selected as the target variable. NEW-DATA-1.T15.txt is used for training and NEW-DATA-2.T15.txt is used for testing. The original 15-minute sampled data is downsampled by a factor of 4 to obtain hourly measurements.

3. Data Preprocessing

A sliding window approach is applied to convert the time series into a supervised learning problem. The previous 5 temperature values are used as input features to predict the next temperature value. All input features are normalized using z-score normalization based on the training data.

4. Model

A Linear Regression model is selected due to its simplicity and low computational cost. The model estimates the output temperature using a weighted sum of normalized inputs and a bias term. The trained model parameters are transferred directly to the STM32 platform.

5. Results

Model performance is evaluated using Mean Absolute Error (MAE) and Root Mean Square Error (RMSE). The obtained results are MAE = 0.167 and RMSE = 0.204, showing accurate future temperature prediction.

6. Embedded Implementation

The trained model is implemented on the STM32F746G-DISCO development board using C. The embedded application applies normalization, regression computation, and outputs predicted values via UART.

7. Conclusion

This study demonstrates that a simple linear regression model can be efficiently deployed on a microcontroller. The approach achieves accurate results while maintaining low computational complexity, making it suitable for embedded systems.