

# Report

## 1. Introduction

In this work, we analyze gridded meteorological data extracted from XML files, containing longitude–latitude grids and associated physical values. The task is two-fold:

1. **Classification:** Identify whether each grid cell is valid (label = 1) or invalid (label = 0).
2. **Regression:** Predict the temperature value for valid grid cells.

We build two neural networks — a **classifier** and a **regressor** — and train them using PyTorch. The final models are evaluated both numerically and visually, with spatial maps comparing ground truth and predictions.

## 2. Data Preprocessing

The XML file provides a  $67 \times 120$  grid (8040 points) with georeferenced values. The preprocessing pipeline includes:

- **Labeling:** Grid cells with value  $\leq -900$  are marked as invalid (label = 0), others as valid (label = 1).
- **Normalization:** Longitude and latitude are standardized using StandardScaler to stabilize training.
- **Splitting:**
  - Classification dataset: (Longitude, Latitude)  $\rightarrow \{0,1\}$ .
  - Regression dataset: (Longitude, Latitude)  $\rightarrow$  temperature (only for valid cells).
  - Each dataset is randomly split 80/20 into training and validation sets.

## 3. Model Architecture

## **Classifier**

A fully connected feedforward network:

- Input: 2 (Longitude, Latitude)
- Hidden layers:  $10 \rightarrow 16 \rightarrow 10$  units, ReLU activation
- Output: 2 classes (valid vs invalid)
- Loss: CrossEntropyLoss
- Optimizer: Adam, learning rate = 1e-3

## **Regressor**

A similar feedforward network for predicting temperature values:

- Input: 2 (Longitude, Latitude)
- Hidden layers:  $10 \rightarrow 16 \rightarrow 10$  units, ReLU activation
- Output: 1 (temperature in °C)
- Loss: MSELoss
- Optimizer: Adam, learning rate = 1e-3

Both models are lightweight but expressive enough to capture spatial variation in the grid.

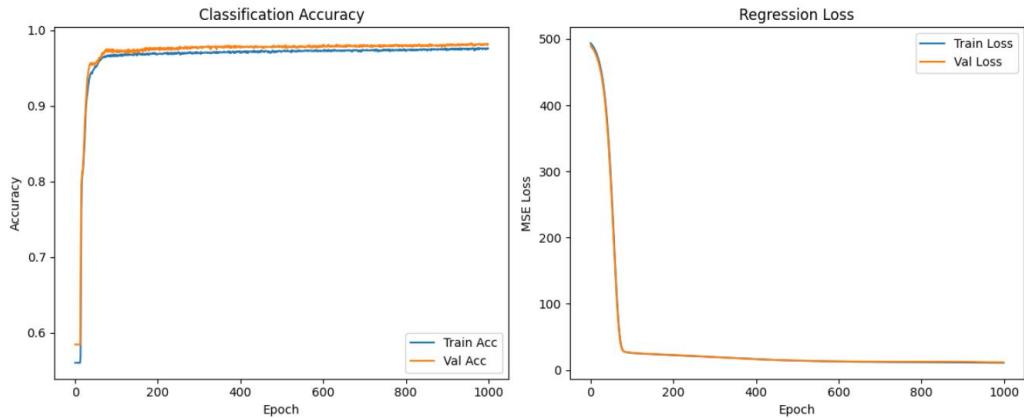
## **4. Training Procedure**

- Training runs for **1000 epochs**.
- Batch size = 1024.
- Classification: training and validation accuracy are tracked.
- Regression: training and validation MSE loss are tracked.

During training:

- The classification model quickly converges to >95% accuracy within a few hundred epochs.

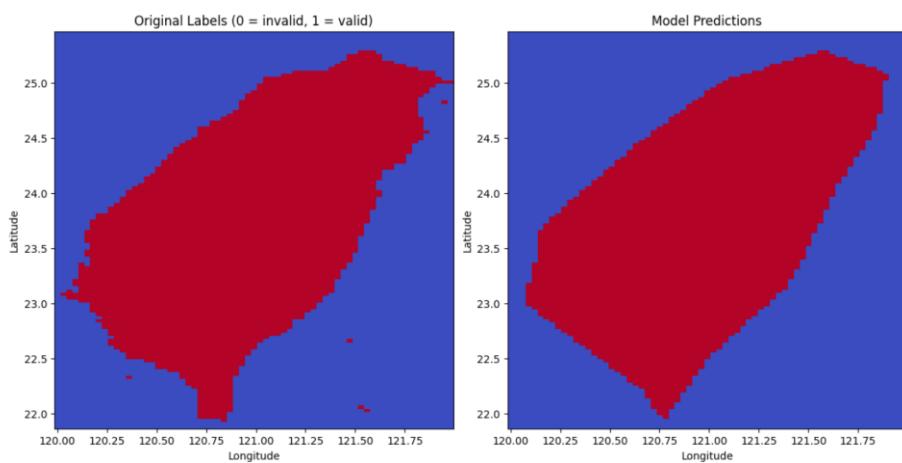
- The regression model's validation loss stabilizes after 500 epochs, indicating good generalization.



## 5. Results

### 5.1 Classification

- Accuracy:** 0.9772
- Confusion matrix:**
  - True Positive (TP): 3406
  - True Negative (TN): 4451
  - False Positive (FP): 94
  - False Negative (FN): 89
- Visualization:** Comparison between ground truth labels and predicted labels shows the model correctly identifies valid vs invalid regions, with most errors occurring near boundaries.



## 5.2 Regression

- **MSE Loss:** Training loss  $\approx 10.8386$ , Validation loss  $\approx 11.5542$ .
- **Scatter plot:** Predicted vs true temperatures align well along the diagonal, confirming good predictive accuracy.

