User Manual - GISPrecip: Precipitation Classification and Estimation

GISPrecip Development Team (Gabriel Rocha, Júlia Gayotto, Salatiel Jordão and Thainá Lopes)

August 29, 2025

Abstract

This manual provides detailed instructions on how to use the GISPrecip plugin for precipitation classification and estimation in QGIS. It covers installation, interface description, workflow steps, and main features. The plugin follows the typical pattern recognition pipeline: preprocessing, feature extraction, and classification/regression. GISPrecip integrates satellite and radar precipitation data into QGIS, allowing users to train models, assess their performance, and generate precipitation classification or estimation in a geospatial context.

Contents

1 Introduction

GISPrecip was designed to support precipitation data analysis within QGIS by integrating machine learning methods into the geospatial environment. The plugin enables users to train models, evaluate them, and apply predictions directly on satellite and surface precipitation datasets. Provides evaluation and visualization capabilities directly within QGIS, promoting greater accessibility and speed in conducting meteorological studies.

1.1 Pattern Recognition Workflow

Following the general structure of a pattern recognition process [**?**], the plugin operates in three main stages:

1. **Preprocessing** – prepares the input data by normalizing scales, reducing noise, and improving consistency.
2. **Feature Extraction** – converts raw satellite and radar precipitation data into compact vectors, keeping only relevant and discriminative information.
3. **Classification/Regression** – applies supervised machine learning algorithms to assign precipitation classes or estimate precipitation values.

1.2 Technologies Used

The interface was developed with *Qt Designer* and integrated into QGIS. The backend is implemented in *Python*, relying on libraries such as:

- **Scikit-learn** – machine learning models and metrics.
- **NumPy** – numerical operations and array handling.
- **Xarray** – manipulation of multi-dimensional data (e.g., NetCDF).
- **netCDF** – manipulation of NetCDF files.
- **SciPy** – scientific computing, including optimization, statistics, and signal processing.
- **imbalanced-learn** – techniques for handling imbalanced datasets in classification problems.
- **PyQt** – GUI integration with QGIS.

2 Installation

This section provides detailed instructions on how to install and configure the GISPrecip plugin in QGIS. The process includes cloning the repository, setting up dependencies, and enabling the plugin within QGIS.

2.1 Technologies Required

The plugin is built using the following technologies:

- **QGIS**: Open-source GIS platform where the plugin is executed.
- **Python**: Backend implementation, leveraging PyQGIS for integration.

2.2 Step 1: Clone the Repository

Clone the GISPrecip repository from GitHub:

```
git clone git@github.com:RP-Group/GISPrecip.git
```

2.3 Step 2: Prepare the Plugin Folder

Compress the cloned folder into a `.zip` file, which will later be installed in QGIS.

2.4 Step 3: Virtual Environment and Dependencies

Before using the plugin, a Python virtual environment must be created and dependencies installed.

1. Create a virtual environment and activate it:

   ```
   python3 -m venv GISPrecip
   source GISPrecip/bin/activate
   ```

2. Install the required dependencies:

   ```
   pip install scikit-learn numpy netCDF4 scipy imbalanced-learn
   ```

2.5 Step 4: Configure QGIS Python Console

1. Open QGIS and navigate to `Plugins > Python Console`. 2. Open the Python editor. 3. Run the `requirements.py` script, updating the virtual environment path if necessary.

Example of path configuration:

```
import sys, os
venv_dir = os.path.expanduser("~/GISPrecip")
site_packages = os.path.join(venv_dir,"lib","python3.12","site-packages")
sys.path.append(site_packages)
print("Packages installed added.")
```

Wait for the confirmation message: `"Packages installed added."`

2.6 Step 5: Load the Plugin in QGIS

1. Open QGIS and go to `Plugins > Manage and Install Plugins`. 2. Select the option `Install from ZIP`. 3. Browse to the compressed plugin folder and click `Install Plugin`.

After installation, the GISPrecip plugin will be available in the QGIS menu.

2.7 Step 6: Start Using the Plugin

Once installed, the plugin interface can be launched, allowing the user to begin analyzing precipitation data through the `Model`, `Assessment`, and `Prediction` tabs.

3 User Interface

The plugin consists of three main tabs: **Model**, **Assessment**, and **Prediction**. Each tab includes a *log panel* at the bottom, where the plugin displays execution steps and processing messages.

3.1 Model Tab (Training)

This tab allows users to prepare and train machine learning models with satellite (GMI Data) and surface precipitation (radar) datasets.

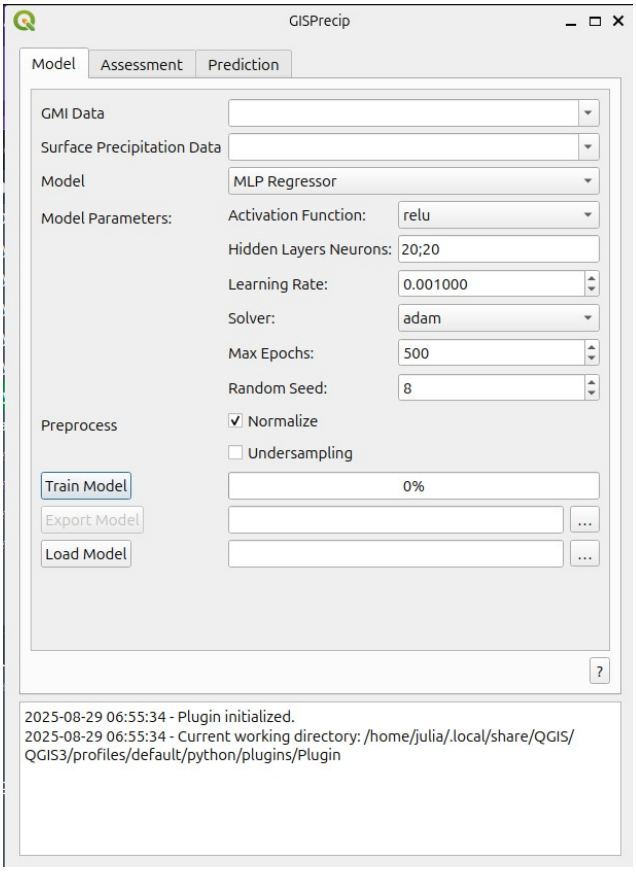

Figure 1: Model tab for training configuration.

3.1.1 Inputs

- **GMI Data** – satellite precipitation dataset.
- **Surface Precipitation Data** – ground-based radar precipitation dataset.

3.1.2 Model Selection

Users can select the algorithm:

- *Classification*: SVM, Random Forest, Decision Tree, AdaBoost.
- *Regression*: MLP Regressor.

3.1.3 Model Parameters

- **Activation Function** – activation in neural networks (e.g., relu, tanh).
- **Hidden Layers Neurons** – structure of the MLP (e.g., `20;20` means two hidden layers with 20 neurons each).
- **Learning Rate** – step size for weight updates.
- **Solver** – optimization algorithm (adam, sgd, etc.).
- **Max Epochs** – maximum training iterations.
- **Random Seed** – ensures reproducibility.

3.1.4 Preprocessing Options

- **Normalize** – standardizes data values for balanced training.
- **Undersampling** – optional reduction of majority classes for balanced datasets.

3.1.5 Training Actions

- **Train Model** – trains the selected algorithm with the provided data.
- **Export Model** – saves the trained model for later use.
- **Load Model** – loads a previously saved model.

3.2 Assessment Tab (Evaluation)

The *Assessment* tab enables users to test the trained model on validation data.

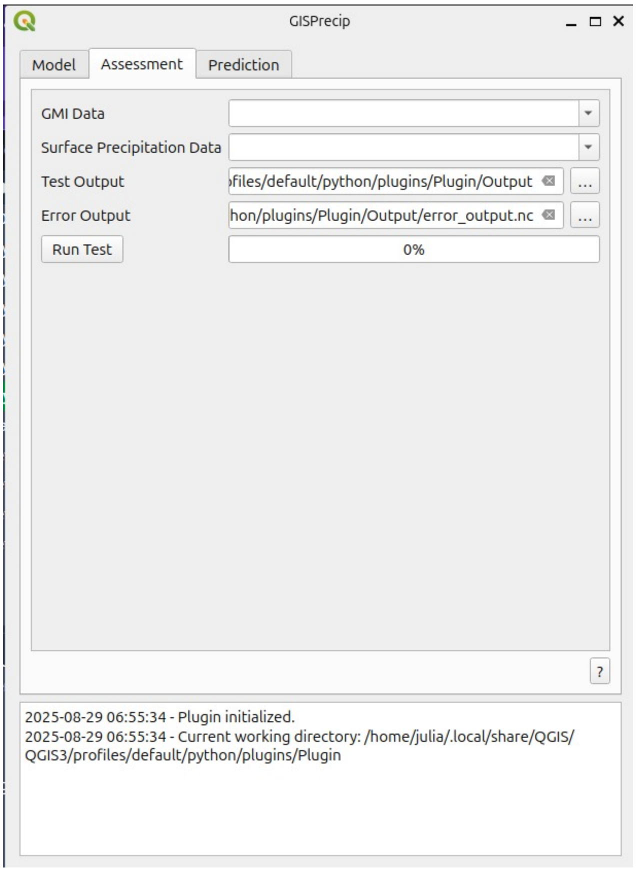

Figure 2: Assessment tab for model evaluation.

3.2.1 Inputs

- **GMI Data** – satellite precipitation dataset. This layer must be previously imported into QGIS so it can be selected within the plugin.
- **Surface Precipitation Data** – ground-based radar precipitation dataset. This layer must also be loaded into QGIS before being available for selection.

3.2.2 Outputs

- **Test Output** – stores computed evaluation metrics.
- **Error Output** – saves spatial distribution of prediction errors in raster format.

3.2.3 Execution

- **Run Test** – executes the model against validation data.

3.2.4 Evaluation Metrics

- **Regression**: Bias, MSE, MAE, SMAPE, linear correlation.
- **Classification**: Accuracy, Recall, F1-score, FAR (False Acceptance Rate), confusion matrix.

3.3 Prediction Tab (Classification or Estimation)

The *Prediction* tab applies trained models to new satellite datasets (without target labels).

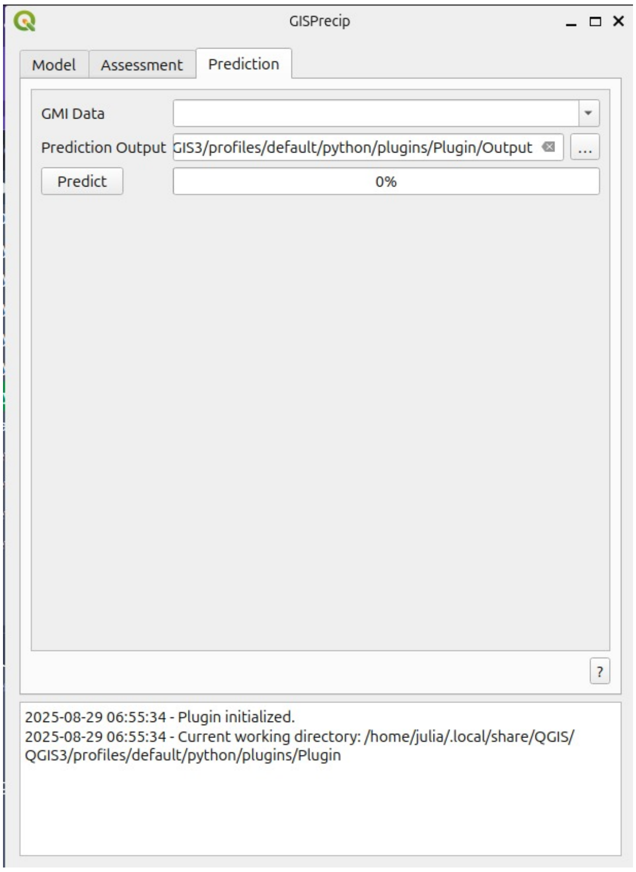

Figure 3: Prediction tab for generating new precipitation forecasts.

3.3.1 Inputs

- **GMI Data** – new satellite precipitation dataset.

3.3.2 Outputs

- **Prediction Output** – saves model classifications or estimations.

3.3.3 Execution

- **Predict** – applies the trained model to the input data and generates precipitation prediction layers.

The prediction layers can be exported and visualized directly in QGIS for further geospatial analysis.

4 Common Issues and Solutions

4.1 Data Loading Errors

Ensure that input files are supported. Avoid special characters or spaces in file paths.

4.2 Model Training Errors

Check that parameters are properly configured and sufficient system memory is available.

4.3 Assessment or Prediction Failures

Verify that compatible datasets are used and ensure consistency between input formats.

5 Conclusion

GISPrecip integrates geospatial datasets with machine learning techniques, providing QGIS users with a practical and accessible tool for precipitation classification and estimation. With its structured workflow (training, evaluation, prediction), it enhances the capacity to analyze, visualize, and interpret precipitation patterns across different regions.