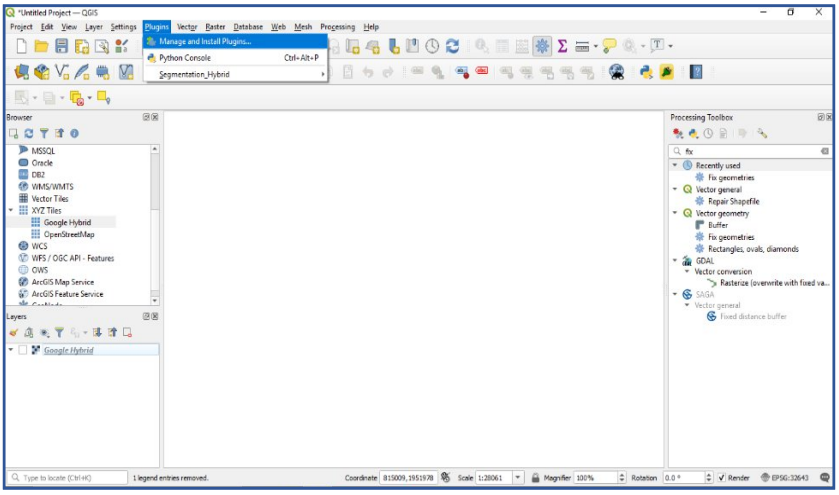
ML-CLAS : Plug-in (Quick Guide for TWDTW Algorithm)

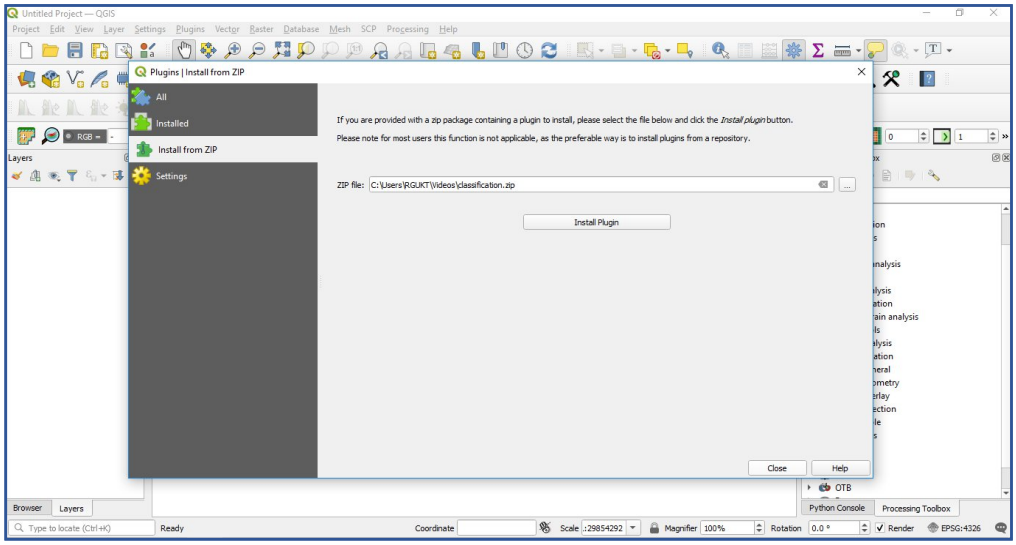
ML-CLAS (Machine Learning Classification for fragmented croplands) Crop Classification plug-in can perform parcel-based classification using segmented land parcels. It includes SVM _RF algorithms, where one can perform crop classification on Support Vector Machine (SVM) and Random Forest (RF) algorithms. Besides, another algorithm called TWDTW (Time Weighted Dynamic Time Warping) algorithm is also introduced in this plugin. Prior to run SVM / RF or TWDTW we require a complex and repetitive procedure to modify the Raster & Vector data as per requirement. However, the ML-CLAS plugin allows users to provide fewer inputs and produce desired outputs by achieving its default preprocess.

Installation of plugin :

1. Open QGIS. Go to **Plugins** in the Menu bar ⟶ click on **Manage and Install Plugins**.



2. Click on **Install from ZIP** in the right panel ⟶ Load the **ML-CLAS.ZIP** (file provided) Now click on **Install Plugin.**

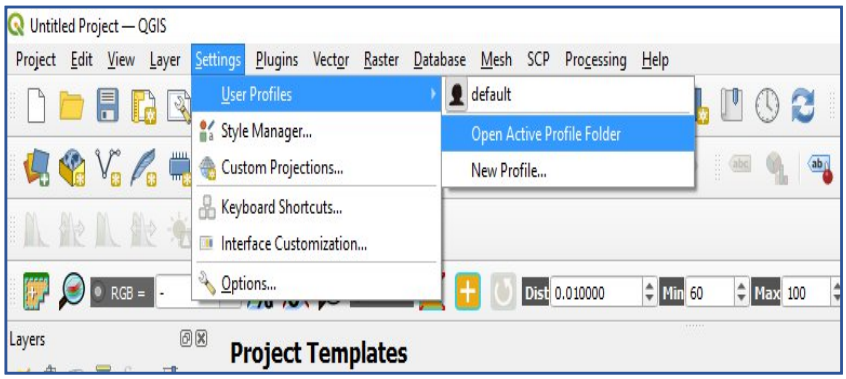


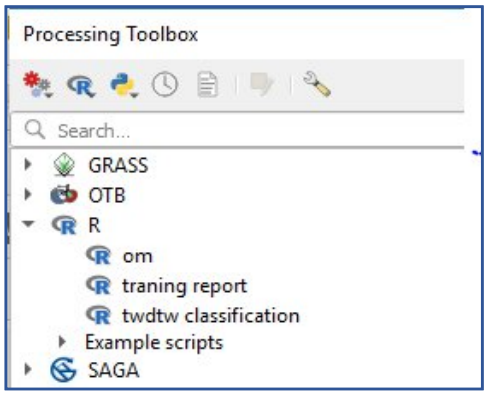
3. Finally, the plugin has installed and plugin  icon appears on the top panel bar.

4. Click on that plugin icon and new dialog box named **ML-CLAS** will open (as shown below).

Note:

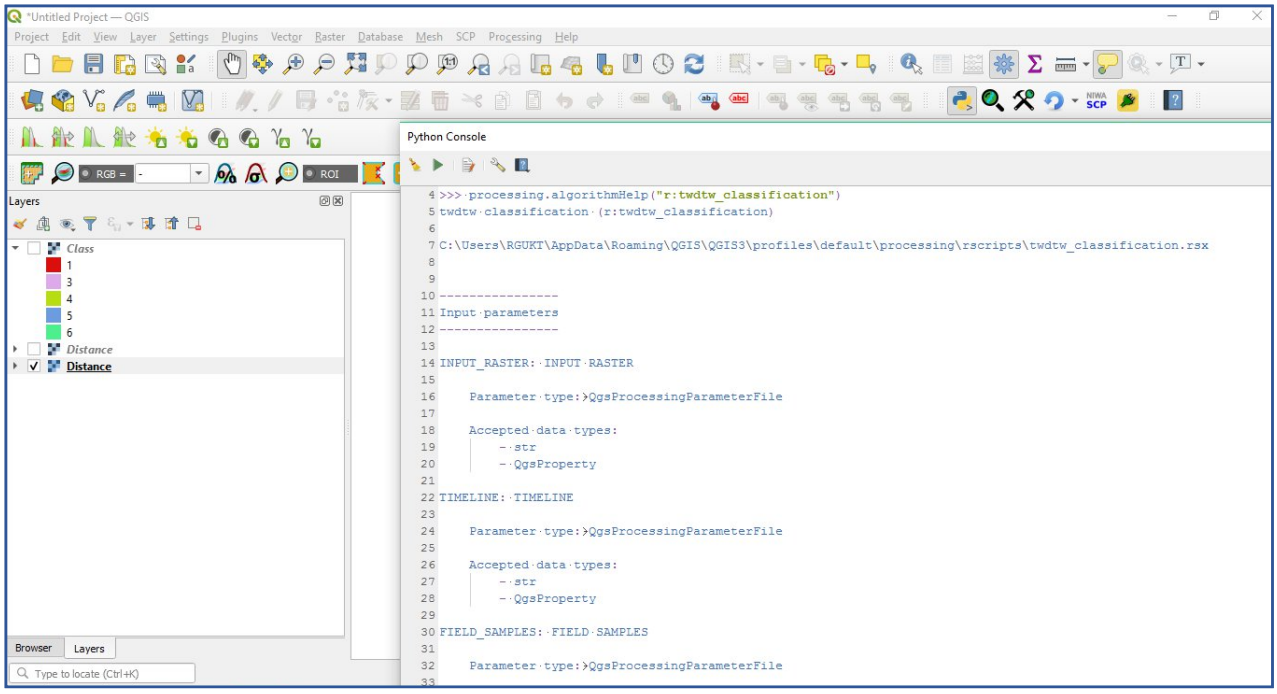
- Ensure **R processing** is successfully installed in **Processing Toolbox** in QGIS.
- Go to Menu bar **settings** ⟶ **User Profiles** ⟶ **Open Active Profile Folder.**
- Now open **processing** folder then open **rscripts** where **.rsx** files will appear.
- Copy the **twdtw classification.rsx** file provided in source file and paste in **rscripts** folder.
- Restart QGIS, click on **R processing** in processing toolbox and copied file will appear here as **twdtw classification.**
- Now you can run **TWDTW** in ML-CLAS plugin.



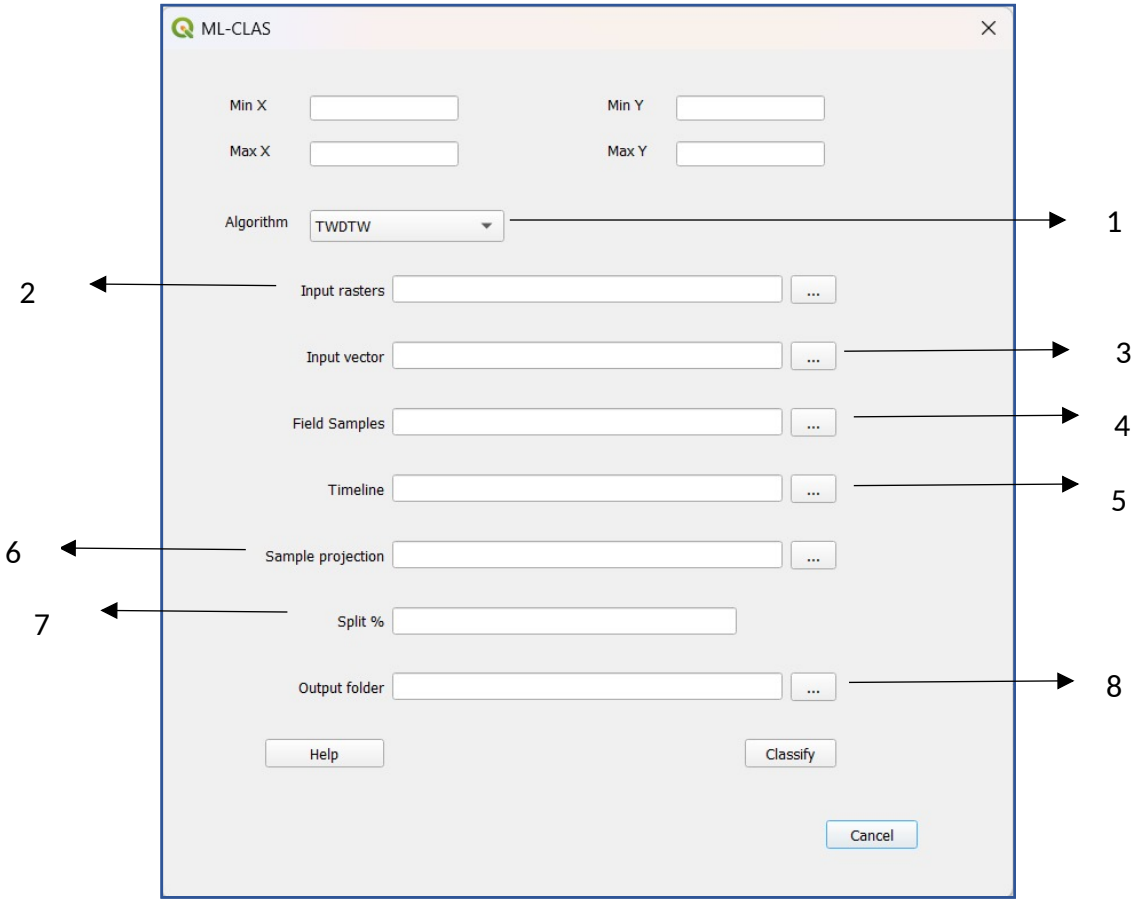


Verification:

- Before running the plugin verify the script loaded as processing algorithm by running following command in python terminal;
  processing.algorithmHelp("r:twdtw_classification")



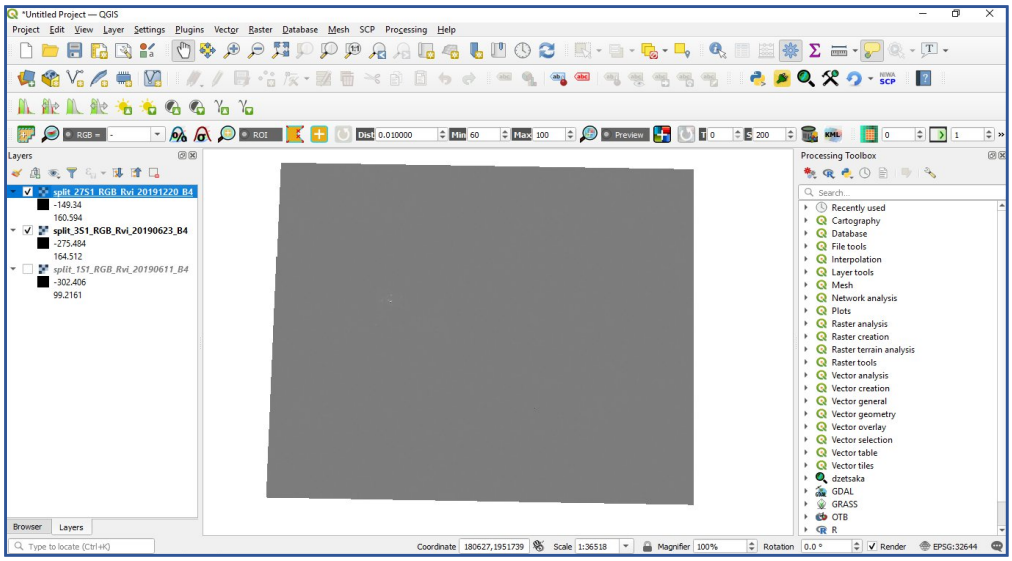
TWDTW inputs :



1. **Algorithm :**
   - ML-CLAS crop classification plugin contains another popular algorithm called **TWDTW**.
   - Users can select the TWDTW algorithm in dropdown menu for the crop classification.

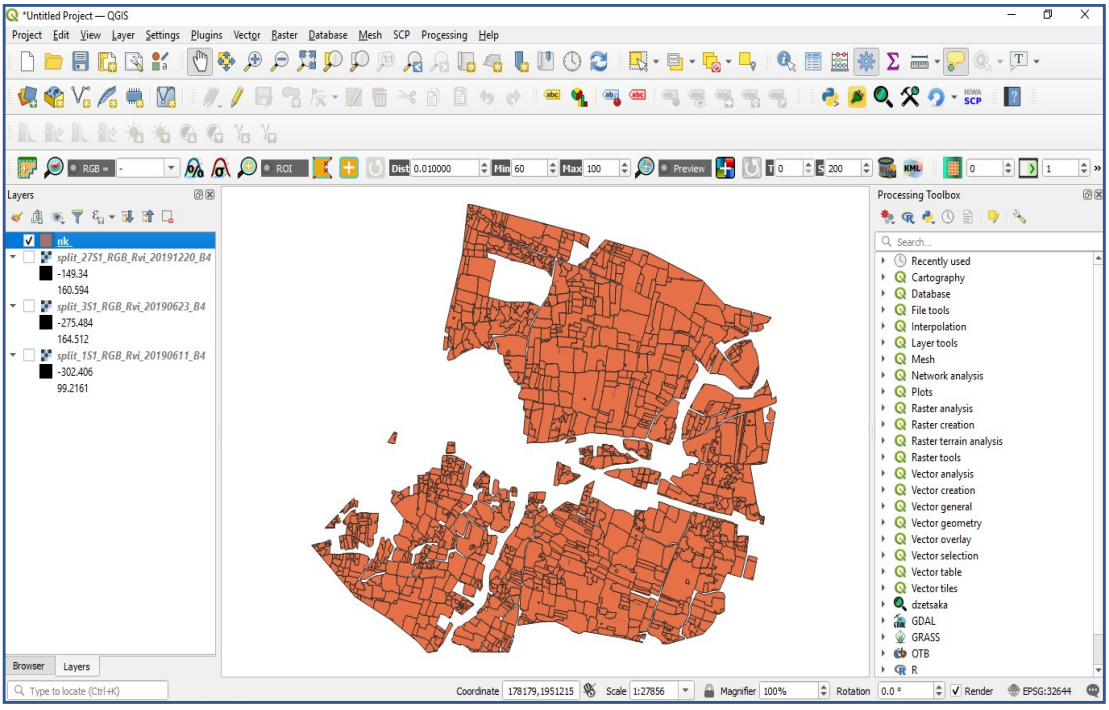
2. Input rasters:

- The .TIF/raster images of the satellite data is taken as input. User can select multiple satellite bands in single click.

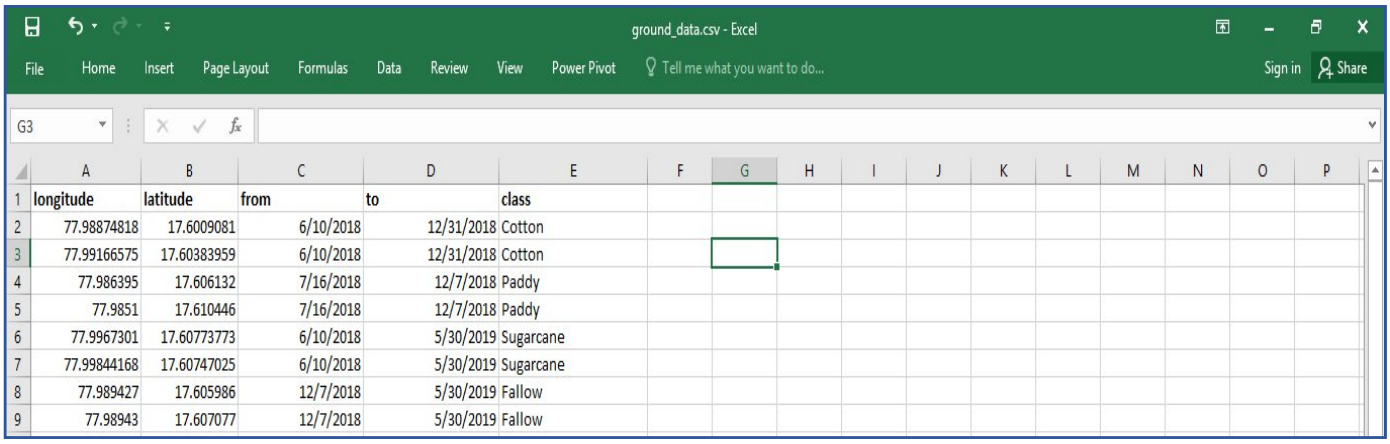


3. Input Vector:

- Segmentation should be done before the classification. So, the output segmented land parcels (i.e agricultural field boundaries) of our region of interest obtained from segmentation is considered as an input vector. It takes .**shp** file only



4. **Field samples:**
    - These are nothing but sample ground truth dataset obtains from field observations, which are provided to learn by model how to categorize new observations into other classes.
    - The input file must contain location details like latitude & longitude, data of sowing & harvesting and type of crop in that location.
    - Only **.csv** is taken as input.



	A	B	C	D	E
1	longitude	latitude	from	to	class
2	77.98874818	17.6009081	6/10/2018	12/31/2018	Cotton
3	77.99166575	17.60383959	6/10/2018	12/31/2018	Cotton
4	77.986395	17.606132	7/16/2018	12/7/2018	Paddy
5	77.9851	17.610446	7/16/2018	12/7/2018	Paddy
6	77.9967301	17.60773773	6/10/2018	5/30/2019	Sugarcane
7	77.99844168	17.60747025	6/10/2018	5/30/2019	Sugarcane
8	77.989427	17.605986	12/7/2018	5/30/2019	Fallow
9	77.98943	17.607077	12/7/2018	5/30/2019	Fallow

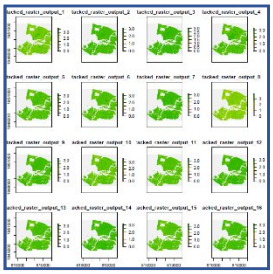
5. **Timeline:**
    - Date of acquisition of each satellite imagery must be saved as **"dd-mm-yyyy"** in a text file i.e., (**.txt**).
    - Remember date of acquisition must be matched with satellite imagery provided in **input raster** because, these dates will be used further to create temporal patterns by the classification model.
    - **Eg:** If we give 10 satellite images as input then the **txt** file should contain respective 10 dates of acquisition.

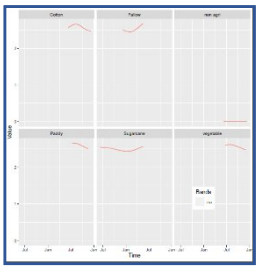
6. **Sample projection:**
    - Projection is a method by which curved surface of earth can be portrayed as flat surface.
    - Here user should provide projection file which is available in supporting files of Input vector.
    - Available format "**.prj** " (Eg:input_vector.prj)

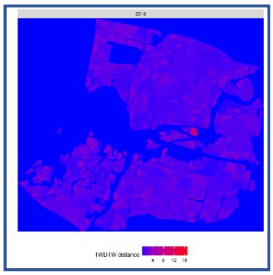
7. **Split % :**
    - The given data is divided into training data and validation data. Training data is a part of original data i.e., used to train the machine learning model whereas the validation data is used to check the accuracy of model.
    - So in classification plugin user should enter the percentage (%) of data for validation. Let's assume 40% is your testing data then enter 40.

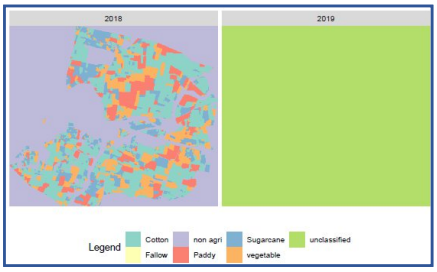
8. Output folder:

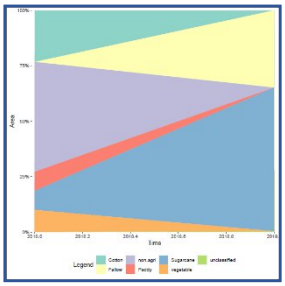
- Select the output folder location where you want to save your Classification files.
- Click on **classify** to run the plugin, once the plugin finished its operation following files will appear in output folder.
- From 1 to 8 images shown below will be available in a single document named **Rplots.pdf** in plugin output folder. Besides Confusion Matrix, Accuracy, Classification and Distance files also available in user accessible formats.

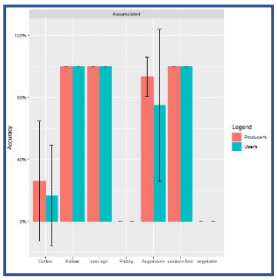


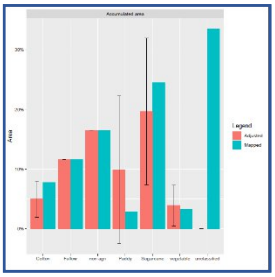


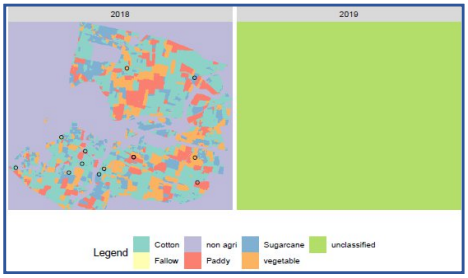


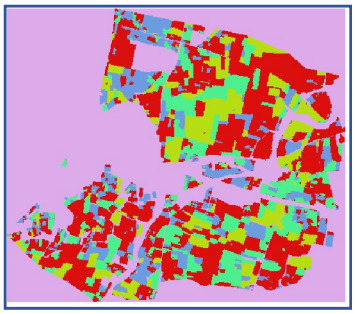












	cotton_hs	cotton_ls	nonagri	paddy_hs	paddy_ls	sugarcane	vegetable
cotton_hs	12	2	0	8	4	2	3
cotton_ls	1	11	0	1	4	1	1
nonagri	0	0	5	0	0	0	0
paddy_hs	2	3	0	14	1	0	1
paddy_ls	0	2	0	3	11	2	4
sugarcane	0	1	0	0	0	17	0
vegetable	3	1	0	1	0	2	15
unclassified	0	0	2	0	0	0	0
Total	18	20	7	27	20	24	24

```
An object of class "twdtwAssessment"
Number of classification intervals: 3
Accuracy metrics summary

Overall
Accuracy       Var        sd       ci*
    0.79      0.80      0.80      0.80

User's
             Accuracy  Var   sd  ci*
cotton_hs        0.73 0.65 0.76 0.72
cotton_ls        0.72 0.66 0.70 0.75
nonagri          1.00 0.00 0.00 0.00
paddy_hs         0.79 0.72 0.75 0.72
paddy_ls         0.66 0.70 0.70 0.72
sugarcane        0.78 0.77 0.72 0.71
vegetable        0.76 0.76 0.69 0.69
unclassified     1.00 0.00 0.00 0.00

Producer's
             Accuracy  Var   sd  ci*
cotton_hs        0.77 0.72 0.75 0.78
cotton_ls        0.69 0.66 0.73 0.71
nonagri          1.00 0.00 0.00 0.00
paddy_hs         0.69 0.70 0.69 0.76
paddy_ls         0.78 0.78 0.70 0.78
sugarcane        0.71 0.79 0.76 0.70
vegetable        0.68 0.67 0.67 0.66
unclassified     1.00 0.00 0.00 0.00
```