



# **Cartographie de l'occupation du sol 2020 sur l'île de la Réunion**



# Objectif:

Réaliser une cartographie de l'occupation des sols par classification supervisée. Celle-ci sera composée de 12 différentes typologies d'occupation du sol.

## Zone d'étude:

Ouest de l'île de la Réunion



# Présentation des données

---

Images Sentinel:

- 12 images au niveau 2A (correction radiométriques, corrections géométriques)
- affiner avec une résolution pour chacune des 12 bandes à 10m

Données vecteurs:

- des polygones représentant les 12 classes à partir de vérité terrain ou photo-interprétation



# Principe de la classification:

Chaque pixel d'une image appartient à une classe d'occupation. On classe les pixels en fonction de leurs signatures spectrales. En analysant la valeur radiométrique de chaque bande, on arrive à les classer. Par une méthode de classification, on détermine les limites de chaque classe.

La classification s'entraîne sur un jeu d'apprentissage et qui, une fois pertinente, permet de classer l'ensemble du jeu de données.

# Base de données utilisée:

La base de données regroupant une liste de polygones au format shapefile a été découpé en 2 catégories:

Les deux jeux se recoupent. Une partie de la vérité terrain est exclue de l'apprentissage

- pour le jeu d'apprentissage
  - regroupe env 300 objets
- pour le jeu de validation
  - regroupe env 400 objets

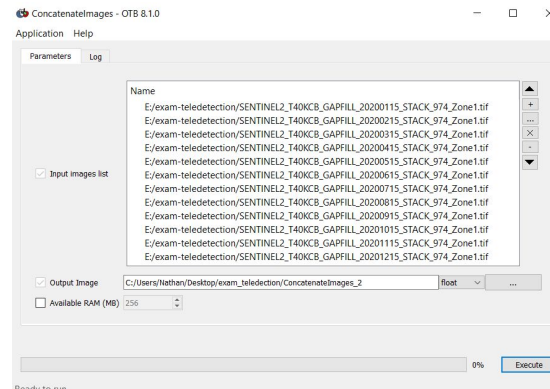
Check: on prend bien les 12 classes dans les dans les 2 jeux.



# Démarche et outils:

Utilisation de OTB et de ces outils:

- **ConcanetelImage**: regrouper les 12 images de la série et ses bandes en 1 image multi-bandes (120 bandes)
  - déjà ré-échantillonner (10m)
  - respect de l'ordre chronologique
  - autre solution: fusion des bandes grâce à l'outil de QGIS
- création d'un **fichier statistique**:
  - statistique globale (moyenne/écart-type) pour chacune des bandes.

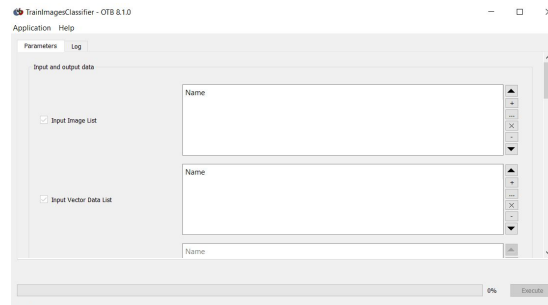


---

# Outils OTB:

## TrainImageClassifier:

- saisie des données pour l'apprentissage
- créer un modèle d'apprentissage
- créer une matrice de confusion
- saisie de l'image sur laquelle est réalisée la classification



## ImageClassifier:

- représenter le modèle de la classification sous forme d'une image raster
-

# Plan d'expérimentation:

- Test de 3 méthodes de classifications supervisées
- paramètre de base: analyse du kappa index + les résultats détaillés par classe
- détermination de la meilleure méthode
- amélioration des paramètres pour cette outil
- ajout de données dans le jeu d'apprentissage
- création du résultat final
- qualité du produit



# Test 1: bayes

Méthode de Bayes: les échantillons de l'image sont classés en fonction de leur distance de la moyenne de la classe. On assigne chaque échantillon à la classe dont il est le moins éloigné.

Paramètres utilisés: 1000 pixels en apprentissage/1000 pixels en validation

Résultat: génère un modèle d'apprentissage et une matrice de confusion

résultat du Kappa index trop faible

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [6] vs all: 0.784483

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [6] vs all: 0.455

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [6] vs all: 0.575949

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [7] vs all: 0.75388

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [7] vs all: 0.34

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [7] vs all: 0.468642

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [8] vs all: 0.629885

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [8] vs all: 0.822

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [8] vs all: 0.713232

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [9] vs all: 0.638677

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [9] vs all: 0.753

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [9] vs all: 0.691143

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [10] vs all: 0.882883

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [10] vs all: 0.882

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [10] vs all: 0.882441

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [11] vs all: 0.805031

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [11] vs all: 0.512

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [11] vs all: 0.625917

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Precision of class [12] vs all: 0.40544

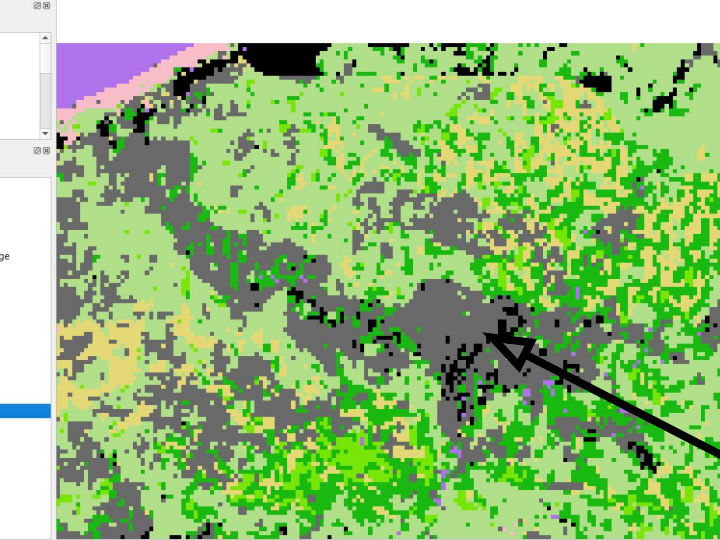
2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Recall of class [12] vs all: 0.788438

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: F-score of class [12] vs all: 0.535505

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: Global performance, Kappa index: 0.641202

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: mapOfIndicesValid[0] = 1

2022-12-19 16:46:40 (INFO) TrainImagesClassifier: mapOfIndicesValid[1] = 2



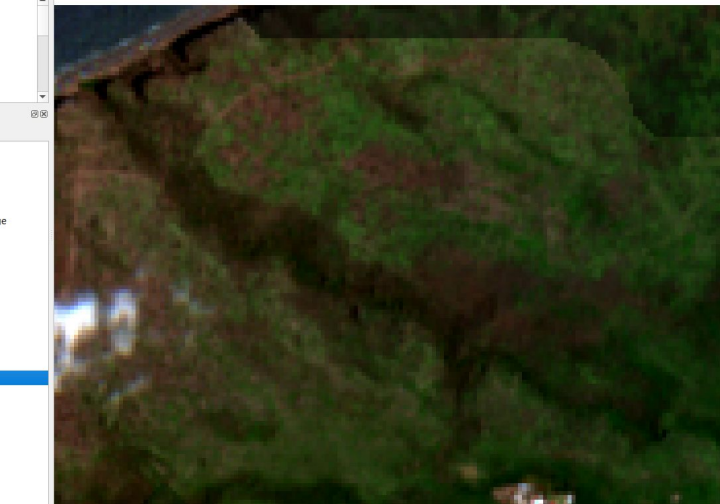
## illustration

Mauvais résultat dans l'ensemble des classes:

espace complètement végétalisé représenté par la classification en friche.

- demande de retravailler les classes:

Friche/eau/paturage/roche/espace boisé



# Test 2: Random Forest

Méthode de RF: conçue pour les jeux de données hétérogènes. Utilisé pour les approches multi-capteurs, ce qui n'est pas le cas ici.

Paramètres utilisés:

maximum depth of the tree=100/minimum number of samples in each node =10/termination criteria for regression tree=1/cluster=10/ size of the randomly=5/maximum number of trees in forest = 200

Résultat:

résultat du Kappa index:0.72

```
2-20 10:04:27 (INFO) TrainImagesClassifier: Precision of class [7] vs all: 0.643564
2-20 10:04:27 (INFO) TrainImagesClassifier: Recall of class [7] vs all: 0.643564
2-20 10:04:27 (INFO) TrainImagesClassifier: F-score of class [7] vs all: 0.643564

2-20 10:04:27 (INFO) TrainImagesClassifier: Precision of class [8] vs all: 0.864583
2-20 10:04:27 (INFO) TrainImagesClassifier: Recall of class [8] vs all: 0.821782
2-20 10:04:27 (INFO) TrainImagesClassifier: F-score of class [8] vs all: 0.84264

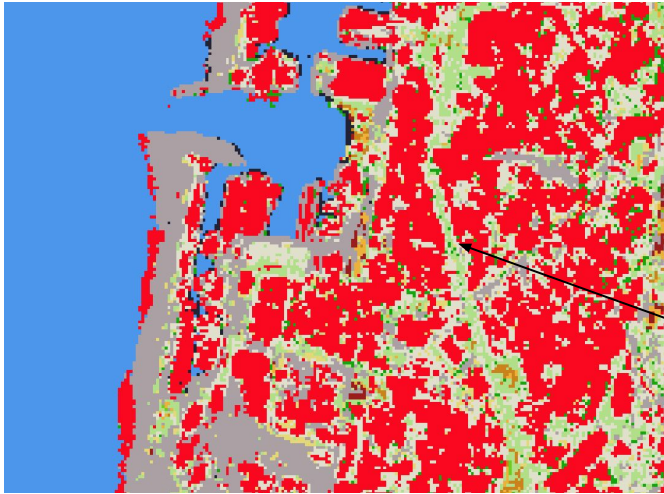
2-20 10:04:27 (INFO) TrainImagesClassifier: Precision of class [9] vs all: 0.721154
2-20 10:04:27 (INFO) TrainImagesClassifier: Recall of class [9] vs all: 0.742574
2-20 10:04:27 (INFO) TrainImagesClassifier: F-score of class [9] vs all: 0.731707

2-20 10:04:27 (INFO) TrainImagesClassifier: Precision of class [10] vs all: 0.988636
2-20 10:04:27 (INFO) TrainImagesClassifier: Recall of class [10] vs all: 0.861386
2-20 10:04:27 (INFO) TrainImagesClassifier: F-score of class [10] vs all: 0.920635

2-20 10:04:27 (INFO) TrainImagesClassifier: Precision of class [11] vs all: 0.898876
2-20 10:04:27 (INFO) TrainImagesClassifier: Recall of class [11] vs all: 0.792079
2-20 10:04:27 (INFO) TrainImagesClassifier: F-score of class [11] vs all: 0.842105

2-20 10:04:27 (INFO) TrainImagesClassifier: Precision of class [12] vs all: 0.634921
2-20 10:04:27 (INFO) TrainImagesClassifier: Recall of class [12] vs all: 0.792079
2-20 10:04:27 (INFO) TrainImagesClassifier: F-score of class [12] vs all: 0.704846

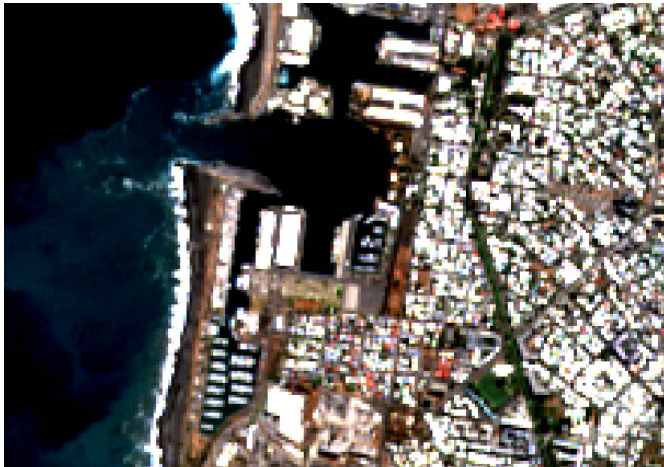
2-20 10:04:27 (INFO) TrainImagesClassifier: Global performance, Kappa index: 0.725473
2-20 10:04:27 (INFO) TrainImagesClassifier: mapOfIndicesValid[0] = 1
```



De nombreux classes qui posent problèmes, il faudrait retravailler les classes:

landes savanes(7)/espace boisé(6)/ arboriculture(5)/  
maraîchage(2)/espace articielle(11)/friche(12)

classé en maraîchage au milieu de la ville?



	1	2	3	4	5	6	7	8	9	10	11	12
1	80	1	3	2	8	2	1	0	0	0	0	4
2	2	70	5	1	11	1	3	0	1	0	0	7
3	4	4	65	6	7	4	5	1	3	0	0	2
4	0	0	0	101	0	0	0	0	0	0	0	0
5	7	19	10	1	56	3	1	0	3	0	0	1
6	1	0	2	0	8	65	8	0	11	0	0	6
7	2	0	2	3	0	2	65	1	6	0	3	17
8	0	0	1	6	0	0	1	83	1	0	6	3
9	0	0	1	0	0	12	5	1	75	1	0	6
10	0	0	2	0	7	0	1	3	0	87	0	0
11	0	0	0	14	0	0	0	7	0	0	80	0
12	3	0	1	0	1	1	12	2	1	0	0	80



# Test3: Support Vector Machine

**Méthode SVM:** représentation des données dans un hyperplan. il crée autant de plan que de catégorie. Trouve les frontières qui séparent le jeu de données. La frontière doit maximiser sa distance avec les points les plus proches de la frontière. Ces points d'entraînements sont les vecteurs supports

Paramètre: 2000 pixels en apprentissage/1000 en validation/

Bound sample = 1 (limiter le nombre d'échantillon par classe pour l'apprentissage)

C = 1

Résultat: Kappa index de 0.78

	1	2	3	4	5	6	7	8	9	10	11	12
1	75	4	7	0	5	2	2	0	0	1	2	3
2	1	76	5	0	11	0	2	0	1	1	2	2
3	1	3	67	4	12	4	5	1	1	0	3	0
4	0	0	2	200	0	0	0	0	0	0	0	0
5	2	6	6	0	80	3	1	1	1	0	0	1
6	2	1	2	0	1	77	3	0	10	1	0	4
7	0	1	4	2	2	2	72	4	0	1	3	10
8	1	0	4	0	0	0	1	89	2	0	4	0
9	0	0	3	1	1	8	7	2	72	0	4	3
10	0	0	0	0	4	3	0	0	3	89	2	0
11	0	0	1	4	0	1	0	3	1	0	91	0
12	1	0	0	0	2	1	8	4	4	0	0	81

retravailler:

- espace boisé (6)
- lande/savane(7)
- maraîchage (3)
- pâturage (2)

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Precision of class [9] vs all: 0.757895

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Recall of class [9] vs all: 0.712871

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: F-score of class [9] vs all: 0.734694

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Precision of class [10] vs all: 0.956989

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Recall of class [10] vs all: 0.881188

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: F-score of class [10] vs all: 0.917526

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Precision of class [11] vs all: 0.81982

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Recall of class [11] vs all: 0.90099

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: F-score of class [11] vs all: 0.858491

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Precision of class [12] vs all: 0.778846

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Recall of class [12] vs all: 0.80198

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: F-score of class [12] vs all: 0.790244

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: Global performance, Kappa index: 0.781278

2022-12-19 18:06:36 (INFO) TrainImagesClassifier: mapOfIndices[valid0] = 1



# Amélioration du jeu d'apprentissage ?

Idée: ajouter des polygones aux classes:

- 1) pâturages
- 2) landes et savanes
- 3) espace boisé

## Problème n°1:

- risque de faire une mauvaise photo-interprétation et créer des objets qui ne correspondent pas à l'occupation réelle du sol sur le terrain.



---

# Amélioration du jeu d'apprentissage ?

ajout de polygones pour certaines classes: exemple pour “espace boisé”

2023-01-06 16:29:48 (INFO) TrainImagesClassifier: Precision of class [6] vs all: 0.768293

2023-01-06 16:29:48 (INFO) TrainImagesClassifier: Recall of class [6] vs all: 0.623762

2023-01-06 16:29:48 (INFO) TrainImagesClassifier: F-score of class [6] vs all: 0.688525

avant

2023-01-06 16:53:21 (INFO) TrainImagesClassifier: Precision of class [6] vs all: 0.742268

2023-01-06 16:53:21 (INFO) TrainImagesClassifier: Recall of class [6] vs all: 0.712871

2023-01-06 16:53:21 (INFO) TrainImagesClassifier: F-score of class [6] vs all: 0.727273

après

Les résultats ne sont pas convaincants

Problème n°2:

Risque de surapprentissage: déconseillé dans les modèles tel que le  
Support Vector Machine

---

# Matrice de Confusion

## Modification des paramètres du modèle SVM:

paramètre C: 3 (minimise les erreurs par rapport au jeu d'apprentissage)

nombre de classe en apprentissage: 2000

nombre de classe en apprentissage: 1000

bound sample: passage de 1 à 0: ce qui a radicalement changé le résultat du kappa index global. Le SVM est efficace sur des petits jeux de données. Les résultats obtenus semblent donc logiques.

Kappa index = 0,92

	1	2	3	4	5	6	7	8	9	10	11	12
1	952	16	6	0	21	2	1	0	1	0	0	1
2	12	905	12	0	48	7	6	0	0	0	0	10
3	0	4	980	0	14	1	1	0	0	0	0	0
4	0	0	0	98	2	0	1	0	0	0	0	0
5	12	22	17	0	898	36	6	0	5	0	1	3
6	4	4	16	0	45	834	33	1	54	8	0	1
7	6	6	10	0	7	15	886	11	25	1	5	28
8	0	1	25	0	0	0	3	950	2	0	17	2
9	1	2	1	0	3	37	11	3	929	3	2	8
10	1	0	1	0	1	5	0	0	7	982	3	0
11	0	0	0	6	0	0	0	28	1	0	965	0
12	0	6	2	0	1	15	44	7	13	0	0	725

2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [1] vs all: 0.962245  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [1] vs all: 0.943  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [1] vs all: 0.952525  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [2] vs all: 0.925888  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [2] vs all: 0.912  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [2] vs all: 0.918892  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [3] vs all: 0.920978  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [3] vs all: 0.979  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [3] vs all: 0.949103  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [4] vs all: 0.932039  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [4] vs all: 0.950495  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [4] vs all: 0.941176  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [5] vs all: 0.86699  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [5] vs all: 0.893  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [5] vs all: 0.879803  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [6] vs all: 0.879098  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [6] vs all: 0.858  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [6] vs all: 0.868421  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [7] vs all: 0.899295  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [7] vs all: 0.893  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [7] vs all: 0.896136

2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [8] vs all: 0.954865  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [8] vs all: 0.952  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [8] vs all: 0.95343  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [9] vs all: 0.897016  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [9] vs all: 0.932  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [9] vs all: 0.914174  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [10] vs all: 0.986948  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [10] vs all: 0.983  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [10] vs all: 0.98497  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [11] vs all: 0.977778  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [11] vs all: 0.968  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [11] vs all: 0.972864  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Precision of class [12] vs all: 0.94357  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Recall of class [12] vs all: 0.884379  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: F-score of class [12] vs all: 0.913016  
  
2023-01-16 17:23:41 (INFO) TrainImagesClassifier: Global performance, Kappa index: 0.92089



## Classification supervisée: île de la Réunion



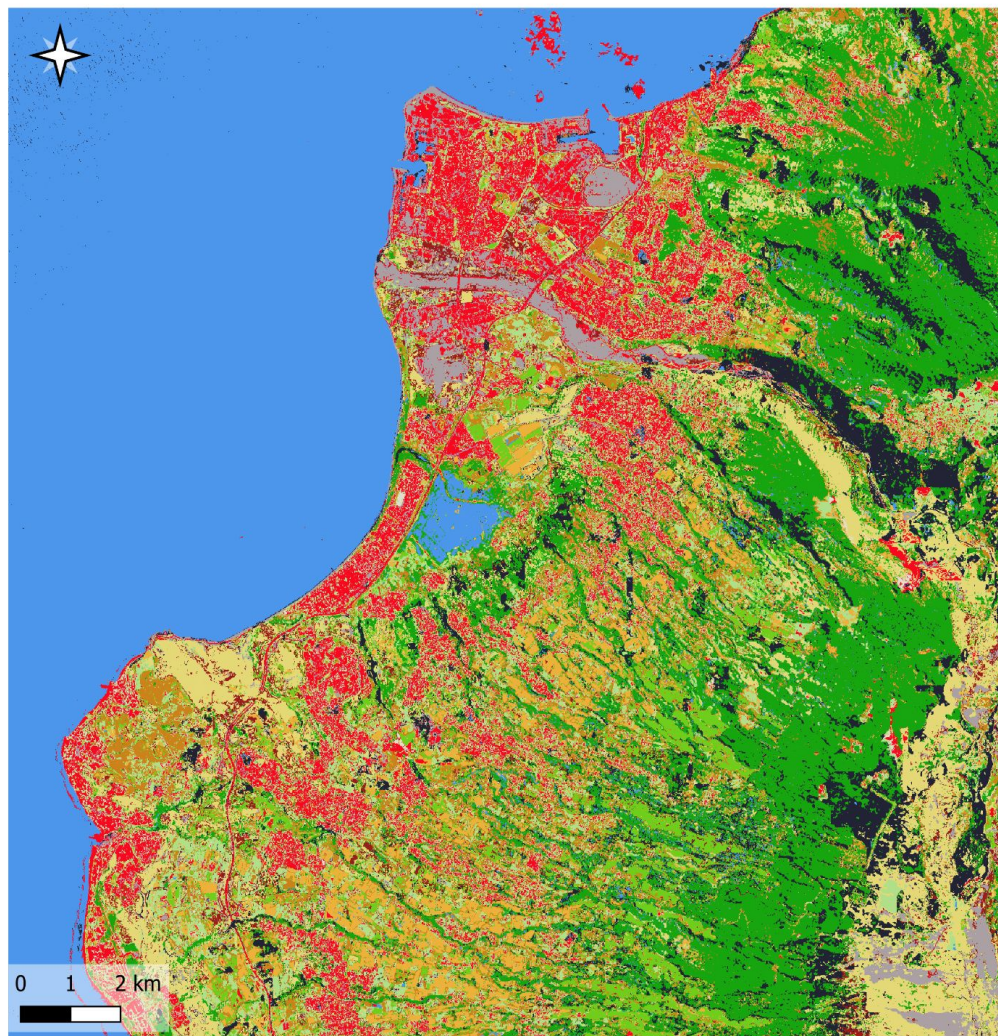
Classes d'occupation du sol:

- 1- canne à sucre
- 2- paturage et fourrage
- 3 - maraichage
- 4 - culture sous serre ou ombrage
- 5 - arboriculture
- 6 - espace boisé
- 7- lande et savane
- 8- rocher et sol nu naturel
- 9- ombre due au relief
- 10- eau
- 11- espace artificialise
- 12- friche

Méthode:  
Support Vector Machine

Données:  
CIRAD/Maison de la télédétection

Réalisation:  
N.FOLMER 2023







**Comparison:**

# Evaluation du produit

A decorative graphic in the top right corner consisting of several overlapping circles in various shades of gray, creating a modern, abstract background element.

Objectif qualité: précision globale sup à 0.85%

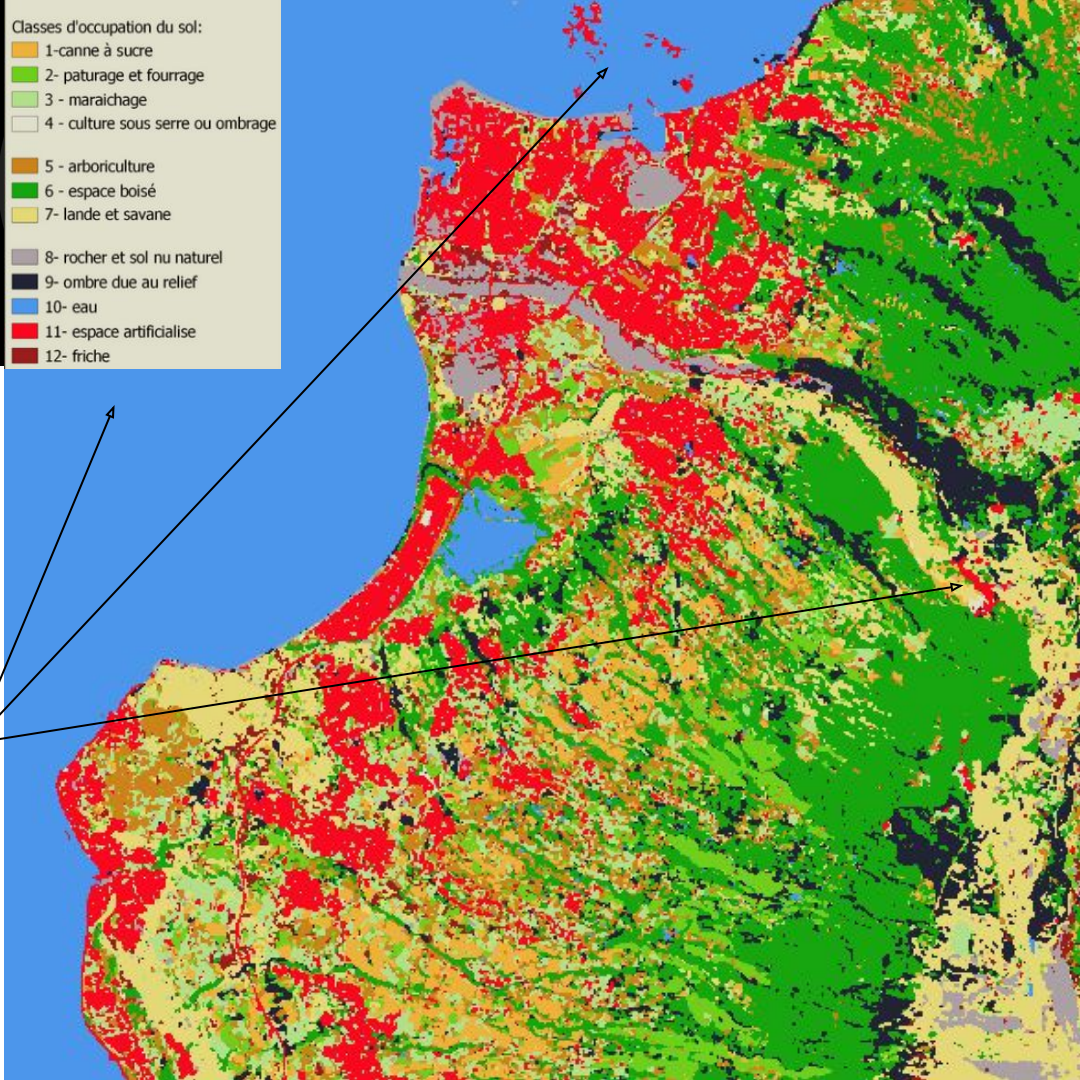
précision comparable pour chaque classe sup à 70%



# Post-classification: réduction de l'effet poivre et sel

utilisation d'un outil  
morphologique:  
"Classification Map  
regulation"  
Réduit le nombre de  
pixels isolés

suppression et/ou  
réduction des pixels  
classés en espace  
artificielles



---

---

# Bibliographie

Guide OrfeoToolBox

<https://larevueia.fr/support-vector-machines-svm/>

---