



[HOME](#)

[CHALLENGES](#)

|

# Challenge Data

## Land cover predictive modeling from satellite imagery

by Preligens

[Introduction](#)

[Public leaderboard](#)

[Private leaderboard](#)

[Submissions](#)

[Submit a solution](#)

Description

[Public leaderboard](#)



[Metrics](#)  
[Download](#)

[Course](#)

[Not registered](#)

[Add a course](#)

Dates

Started on Jan. 4, 2021

Files

[x\\_train](#)  
[input\\_coordinates](#)  
[y\\_train](#)  
[output](#)  
[x\\_test](#)  
[input\\_coordinates](#)  
[random](#)  
[a\\_random](#)  
[supplementary](#)  
[data-re](#)

Challenge context

Preligens

**Preligens** (ex-Earthcube) develops pioneering technology to empower analysts to take smart actions for a safer world. Its artificial intelligence solutions are based on an automated analysis of geospatial intelligence (GEOINT) workflows. Powering state-of-the-art machine learning and computer vision, Preligens aggregates information from multi-intelligence sources (Multi-INT), such as imagery (IMINT), electromagnetic (ELINT), or open source (OSINT), to provide its Defence and Intelligence customers with strategic knowledge regarding mission critical areas of interest ensuring a more insightful operating picture.

Land cover mapping

Land cover mapping is a fundamental problem in Earth observation. Land cover is the observed physical material covering the Earth’s surface. Accurately predicting the physical materials in an environment has many applications, such as analyzing the development of crops in agriculture and modeling the distribution of species in ecology.

Land cover mapping is a very relevant problem for us at Preligens, as it allows to find valuable information for understanding a terrain on a geographic site. One application of interest would be the automatic extraction of environmental attributes characterizing a site, such as the proportion of forest, water, or human-built structures.

The challenge

PRE

Contact

For any question, please contact the provider at

Challenge goals

## Goal

This challenge tackles the problem of land cover predictive modeling from aerial images taken from space. The goal of this challenge is to predict, for an input satellite image in which every pixel is assigned to a land cover class, the proportion of each class of land cover in the image.

## Data description

### Provided inputs and targets

#### Satellite images and segmentation masks

The input data contains 256x256 pixel images of four bands covering the visible spectrum as well as near-infrared (R-G-B-NIR). Those images are extracted from larger Sentinel-2 images over the European continent. Every pixel covers a 10 meters square area. In addition, a segmentation mask is provided for every image, where each pixel contains one land cover label, encoded as an integer. The land cover labels are drawn from the ten different classes that follow (the corresponding class label index is given in parenthesis):

- no\_data (0)
- clouds (1),
- artificial surfaces and construction (2),
- cultivated areas (3),
- broadleaf tree cover (4),
- coniferous tree cover (5),
- herbaceous vegetation (6),
- natural material surfaces (7),
- permanent snow-covered surfaces (8),
- water bodies (9).

The land cover data is a simplified version of a subset of the **S2GLC** dataset (The Sentinel-2 Global Land Cover (S2GLC) project was founded by the European Space Agency). The input images are 16-bits TIFF files and the land cover masks are 8-bits TIFF files.

Note about the “no\_data” and “clouds” classes:

- the “no\_data” class corresponds to pixels for which a annotation is missing; this class is not present in the dataset - its cardinality is zero - but it has a dedicated channel in the ground-truth masks by inheritance of original data in S2GLC;
- the “clouds” class is very rare and not informative as there is no direct correspondance in time between the images and the landcover ground-truth masks, it is rather equivalent to “no\_data”. The two classes “no\_data” and “clouds” are not taken into account in the evaluation metric.

Note also that there is a strong class imbalance present in the data, with dominant classes such as “cultivated areas” and very rare classes such as “permanent snow-covered surfaces” (images having all been taken during summer to get permanent

snow-covered surfaces).

## Class distribution vectors

The actual target to predict in this challenge are derived from the land cover pixel masks: they are vectors of class distribution (the fraction of each class in the image) defined at the image level, computed from all pixel labels. In the training set, both input images and land cover masks are provided. This means that participants can use the land cover masks as a supervisory signal during training. In the testing set, only input images are provided. The model needs to infer class distribution vectors from input images to be evaluated on the testing set. Therefore, the participants have the choice:

- to use the land cover pixel masks as “proxy” targets during training if they wish, to learn a semantic segmentation model as an intermediate function and only then predict the image-level class distribution vector;
- to train directly on the image-level labels that are the class distribution vectors, but the problem is probably harder to solve because of the weaker supervision signal.

The output class distribution vectors are saved in CSV format with: one column for the sample ID, one column for each class (totalizing 11 columns), and one row for every sample image in the set (either the training or testing set).

## Training and testing sets

The training set is composed of 18491 triplets of input images, segmentation masks and target image-level class vectors. The testing set is composed of 5043 input images. Training and testing samples have been separated in a way that the class cardinalities is kept relatively close across the sets.

This testing set has been halved randomly to construct the public and private test sets.

Every sample in the dataset has an unique sample ID, and TIFF files are named by their ID.

A folder containing the training images and masks, and the test images is given as a TAR GZIP archive under *supplementary files*, for a total size of ~7 Go for the compressed archive and ~14 Go when unpacked.  $x_{train}$  and  $x_{test}$  are CSV files containing just the sample ID of every image in the training and testing set respectively.  $y_{train}$  is a CSV file following the format of the output of the challenge, containing the ground-truth class distribution vectors for all samples in the training set.

## Submissions to the challenge

Participants should provide a predicted vector of class distribution for every image in the testing set. Submission CSV files have to follow the exact format described for the output targets.

## Evaluation metric

The metric used in this challenge to rank the participants is the **Kullback-Leibler divergence** for discrete distributions, defined by:

$$\mathbf{KL}(y, \hat{y}) = \sum_{i=1}^C (y_i + \epsilon) \log \left( \frac{y_i + \epsilon}{\hat{y}_i + \epsilon} \right)$$

where  $y$  is the ground-truth class distribution vector,  $\hat{y}$  is the predicted class distribution vector and  $C$  is the number of classes, excluding the classes “no\_data” and “clouds”. This measures how the predicted distribution is different from the ground-truth distribution, and therefore should be minimized. A small term  $\epsilon$  (fixed to  $10^{-8}$ ) is added to  $y_i$  and  $\hat{y}_i$  for smoothness around zero.

The classes “no\_data” and “clouds” are removed from the predicted vectors if they are present. The distribution vectors are re-normalized to sum to 1 on the remaining classes, in every case.

## Benchmark description

### Benchmark model for semantic segmentation : U-Net

The proposed benchmark model is a deep neural network trained on the “proxy” task of semantic segmentation of the land cover labels at the pixel level. This is a fully-convolutional network that follows the U-Net architecture (**Ronneberger et al 2015**). We use a relatively small version of U-Net that is composed of around 1 million parameters. Details of the model architecture, training scheme and hyperparameters will be provided in the GitHub repository.

The model is trained to predict class probabilities for every pixel in an image. After it has been trained on segmentation masks, we predict the most confident classes for every pixel and derive a class distribution vector at the image level from the predicted mask.

The evaluation score achieved by the benchmark model on the public testing set is 0.0593.

### GitHub repository

We provide a **GitHub repository** that accompanies the challenge. This repository contains the full Python code for reproducing the benchmark model. Participants can use the available code as a kick-starter for building their solution.

In addition the repository contains the following:

- A README with useful information covering code usage, as well as precise details of the model, training scheme and hyperparameters used. It will also give tips regarding the design of a deep learning solution.
- A Jupyter notebook to do some data visualization
- The implementation of an alternative baseline model: a fully-connected neural network that solves the "direct" task of predicting class distributions at the image level.

[I want to cancel my participation](#)

Supported by    
INSIGHT.DATA.CLARITY. Paris Artificial Intelligence Research InstitutE

[team](#) | [terms of use](#) | [legal notice and pr](#)