

Package ‘AlphaSDM’

October 4, 2026

Title Species Distribution Models on 'AlphaEarth' Satellite Embeddings

Version 0.2.0

Description Fits species distribution models and maps habitat suitability at up to 10 m resolution from occurrence records alone, using the 'AlphaEarth' Foundations satellite embeddings (Brown et al. 2025) [doi:10.48550/arXiv.2507.22291](https://doi.org/10.48550/arXiv.2507.22291). The embeddings, 64 values per pixel per year from a geospatial foundation model, replace environmental layers, so none need to be sourced or aligned. Provides tools to format occurrence records, place pseudo-absences, train and evaluate an ensemble of machine learning models, and export habitat-suitability rasters. Sampling, model training and prediction all run on 'Google Earth Engine', which requires a free account for noncommercial use.

License MIT + file LICENSE

SystemRequirements Python (≥ 3.9) with the 'earthengine-api' module; 'reticulate' installs it automatically when needed.

Encoding UTF-8

Language en-US

RoxygenNote 7.3.3

Imports jsonlite, reticulate (≥ 1.41), sf, stats, utils

URL <https://james-longo.github.io/AlphaSDM/>,
<https://github.com/James-Longo/AlphaSDM>

BugReports <https://github.com/James-Longo/AlphaSDM/issues>

Suggests stars, withr, testthat ($\geq 3.0.0$), knitr, rmarkdown

VignetteBuilder knitr

Config/testthat/edition 3

NeedsCompilation no

Author James Longo [aut, cre, cph]

Maintainer James Longo <james.longo.birds@gmail.com>

Repository CRAN

Date/Publication 2026-10-04 09:00:07 UTC

Contents

calculate_classifier_metrics	2
clear_gee_credentials	3
evaluate_models	3
format_data	5
gee_clean_assets	6
gee_status	7
gee_tasks	7
generate_map	8
generate_pseudo_absences	9
setup_gee	12
Index	13

calculate_classifier_metrics
<i>Calculate discrimination and calibration metrics</i>

Description

Calculate discrimination and calibration metrics

Usage

calculate_classifier_metrics(scores_pos, scores_neg)

Arguments

- scores_pos Numeric suitability scores at presence points. NA is dropped.
- scores_neg Numeric suitability scores at background or absence points. NA is dropped.

Value

A named list: ‘cbi’, ‘auc_roc’, ‘auc_prg’, ‘tss’, ‘ba’ and ‘cor’, each a single number. Scores need only be on a common scale within one call, since every metric except ‘cor’ depends on the ranking alone. When either class is empty the list is filled with the no-skill values.

Examples

```
set.seed(1)
presences <- rbeta(50, 4, 2)
absences  <- rbeta(200, 2, 4)
str(calculate_classifier_metrics(presences, absences))
```

clear_gee_credentials	<i>Forget the saved Earth Engine project, and optionally sign out</i>
-----------------------	---

Description

Removes the project ID AlphaSDM saved. In an interactive session it then offers to delete the Earth Engine sign-in credentials as well; those are shared by every tool on this computer that uses Earth Engine, so they are kept unless you agree. Run [setup_gee](#) afterwards to reconnect.

Usage

```
clear_gee_credentials()
```

Value

Invisibly, TRUE.

Examples

```
## Not run:  
clear_gee_credentials()  
  
## End(Not run)
```

evaluate_models	<i>Evaluate SDM models on Alpha Earth embeddings</i>
-----------------	--

Description

Trains the model ensemble on Google Earth Engine and scores an independent set of coordinates. Training data must contain absences: real ones, or pseudo-absences from `[generate_pseudo_absences()]`. Presence-only input is rejected with directions.

Usage

```
evaluate_models(  
  data,  
  predict_coords,  
  scale = 10,  
  methods = c("svm", "rf", "gbt"),  
  bg_ratio = 1,  
  bg_replicates = TRUE,  
  params = list(),  
  gee_project = NULL  
)
```

Arguments

<code>data</code>	Data frame of training records with ‘longitude’, ‘latitude’, ‘year’ and a ‘present’ column (1 = presence; include 0 rows to supply real absences).
<code>predict_coords</code>	Data frame of records to score, with ‘longitude’, ‘latitude’ and ‘year’ (as from <code>[format_data()]</code>). Include a ‘present’ column to compute evaluation metrics.
<code>scale</code>	Embedding resolution in metres (default 10, the native resolution).
<code>methods</code>	Character vector of models to ensemble. Defaults to ‘c("svm", "rf", "gbt")’; also accepts ‘maxent’, ‘glm’ (logistic regression fitted server-side by IRLS with equal total class weights), ‘similarity’, ‘knn’, ‘cart’, ‘mindist’. MaxEnt and glm follow the regression-family recipe of Barbet-Massin et al. (2012): a large RANDOM pseudo-absence set suits them best (see ‘?generate_pseudo_absences’).
<code>bg_ratio</code>	Absence-to-presence ratio for the models that train on a balanced pool (rf, gbt and knn): their absences are thinned at random to ‘bg_ratio’ times the number of presences. Default 1. ‘NULL’ gives them every absence. svm and maxent always use every absence.
<code>bg_replicates</code>	Logical (default TRUE). Train the balanced-pool methods (rf, gbt, knn) on $k = \min(10, \text{ceil}(10000/\text{pool size}))$ replicate thinned subsets of the absences and average their predictions (Barbet-Massin et al. 2012, Table 1: several runs when few pseudo-absences are used). Requires ‘bg_ratio’ thinning to be active; methods on the full pool are never replicated.
<code>params</code>	Named list of settings for individual models, using the argument names of the Earth Engine classifier, for example ‘list(gbt = list(shrinkage = 0.01), svm = list(kernelType = "RBF"))’. Every model uses Earth Engine’s defaults except where Earth Engine needs a value or its default cannot work on these data: ‘rf’ uses 500 trees and ‘gbt’ 150, since Earth Engine requires a number, and ‘knn’ uses 15 neighbours, since Earth Engine’s single neighbour gives a two-value map. See the Earth Engine reference for ‘ee.Classifier.smileRandomForest’, ‘smileGradientTreeBoost’, ‘libsvm’, ‘amnhMaxent’, ‘smileKNN’ and ‘smileCart’ for every available setting.
<code>gee_project</code>	Optional Earth Engine project override (normally set via <code>[setup_gee()]</code>).

Value

A list containing ‘methods’, ‘model_metadata’, ‘point_predictions’ and, when ‘predict_coords’ has a ‘present’ column, per-model and ensemble ‘metrics’. For cross-validation, split the data yourself and call this once per fold with the fold’s holdout as ‘predict_coords’.

Examples

```
## Not run:
# `occ` holds presences and absences, e.g. from generate_pseudo_absences().
test <- sample(nrow(occ), round(nrow(occ) / 5))
fit <- evaluate_models(occ[-test, ], predict_coords = occ[test, ])
fit$metrics$ensemble

## End(Not run)
```

format_data

*Standardize occurrence records for AlphaSDM***Description**

Standardizes an input data frame for the AlphaSDM pipeline: renames the coordinate, year and presence columns to the package's lowercase names, drops everything else, and filters to the years Alpha Earth covers.

Usage

```
format_data(data, coords, year, presence = NULL, species = NULL, label = NULL)
```

Arguments

data	A data frame of survey records. Coordinates must be longitude and latitude in decimal degrees on WGS84 (EPSG:4326), the system GPS units and most occurrence databases (GBIF, eBird) report in. A data frame carries no CRS information, so nothing can be reprojected on your behalf: projected coordinates such as UTM metres are rejected, and coordinates in another geographic datum are not detectable and will be treated as WGS84. Reproject with 'sf::st_transform(x, 4326)' before formatting if needed.
coords	Character vector of length 2 naming the longitude and latitude columns, longitude first: 'c(longitude_col, latitude_col)'. Values must be decimal degrees on WGS84 (EPSG:4326); see Details.
year	A character string naming the year or date column. Dates are reduced to their year. Records outside the Alpha Earth window are dropped.
presence	Optional. Name of the presence column, holding 1 for presence and 0 for absence. Omit it to treat every record as a presence.
species	Optional. Name of the species column.
label	Optional. Short name for this data set, used only to label the console summary line, for example "Training" or "Evaluation".

Details

The embeddings are annual. Records dated outside the covered window are dropped and the number removed is reported; set them to the first covered year to keep them instead. The window is read from the Earth Engine collection, so it tracks each annual release. This means 'format_data()' needs a connection: run [setup_gee()] first.

Value

A data frame with 'longitude', 'latitude', 'year' and 'present' columns, ready for [evaluate_models()] or [generate_map()]. Rows are ordered presences first. Fewer rows come back than went in whenever records are dropped for coverage, missing values or duplication; each drop is reported as a message.

Examples

```
## Not run:
# Checks the years against the embeddings' coverage, so it needs Earth Engine.
records <- data.frame(lon = c(-111.05, -111.10), lat = c(32.25, 32.30),
                      year = 2022)
pres <- format_data(records, coords = c("lon", "lat"), year = "year")

## End(Not run)
```

gee_clean_assets	<i>Remove leftover AlphaSDM temporary assets</i>
------------------	--

Description

AlphaSDM writes temporary Earth Engine assets while it works and deletes them when it finishes. A run that is killed or loses its connection never reaches that cleanup, so the asset stays and counts against the project's storage quota. This removes those leftovers.

Usage

```
gee_clean_assets(older_than_hours = 48, dry_run = FALSE, project = NULL)
```

Arguments

older_than_hours	Keep assets younger than this. Default 48.
dry_run	If TRUE, report what would be deleted and delete nothing.
project	Earth Engine project id, or NULL to use the saved one.

Details

Only assets named by AlphaSDM ('alphasdm_...') are considered. An asset is kept if a task for it is still pending or running, or if it is newer than 'older_than_hours', so a job in progress in another session is not disturbed.

Value

The asset ids removed, or the ones that would be, invisibly.

Examples

```
## Not run:
gee_clean_assets(dry_run = TRUE) # list what would be removed

## End(Not run)
```

gee_status	<i>Report the Google Earth Engine connection status</i>
------------	---

Description

Prints whether the Earth Engine client is available, whether sign-in credentials exist and of which kind, which project is configured, and whether a live connection succeeds. To monitor running Earth Engine tasks, use [gee_tasks](#) instead.

Usage

```
gee_status(check_live = TRUE)
```

Arguments

check_live	If TRUE (default), make a small request to confirm that the credentials work, not only that they are on disk.
------------	---

Value

Invisibly, a named list of the status fields.

Examples

```
## Not run:
gee_status()

## End(Not run)
```

gee_tasks	<i>Show recent AlphaSDM Earth Engine tasks</i>
-----------	--

Description

Lists the export tasks AlphaSDM has started, with each task's state and age. Call it from a second R session to see what Earth Engine is doing while a long run is in progress. To check the connection itself, use `[gee_status()]`.

Usage

```
gee_tasks(active_only = TRUE, since_minutes = 180)
```

Arguments

active_only	TRUE shows only pending and running tasks; FALSE also lists recently finished ones.
since_minutes	Only include tasks created within this many minutes.

Value

A data frame of tasks with ‘description’, ‘state’ and ‘age_min’, invisibly. Also prints them.

Examples

```
## Not run:
gee_tasks(active_only = FALSE)

## End(Not run)
```

generate_map	<i>Generate an SDM suitability map</i>
--------------	--

Description

Trains the model ensemble on Google Earth Engine and exports a continuous habitat-suitability raster over an area of interest, one GeoTIFF per model plus the ensemble. The maps download directly from Earth Engine in tiles; only a map Earth Engine will not compute tile by tile goes through its batch system and Google Drive, which is slower.

Usage

```
generate_map(
  data,
  aoi,
  scale = 10,
  output_dir = getwd(),
  methods = c("svm", "rf", "gbt"),
  ensemble = TRUE,
  aoi_year = NULL,
  bg_ratio = 1,
  bg_replicates = TRUE,
  params = list(),
  gee_project = NULL
)
```

Arguments

data	Data frame of training records with ‘longitude’, ‘latitude’, ‘year’ and a ‘present’ column (1 = presence; include 0 rows to supply real absences).
aoi	Area of interest: a pre-built ‘ee.Geometry’, a list with ‘lon’/‘lat’/‘radius’, a path to a vector file readable by [sf::st_read()], or “bbox” for the bounding box of ‘data’ (presences and absences).
scale	Output resolution in metres (default 10).
output_dir	Directory to write the GeoTIFF(s) to.

methods	Character vector of models to ensemble. Defaults to 'c("svm", "rf", "gbt")'; also accepts 'maxent', 'glm' (logistic regression fitted server-side by IRLS with equal total class weights), 'similarity', 'knn', 'cart', 'mindist'. MaxEnt and glm follow the regression-family recipe of Barbet-Massin et al. (2012): a large RANDOM pseudo-absence set suits them best (see '?generate_pseudo_absences').
ensemble	Logical; also export the ensemble mean map (default 'TRUE').
aoi_year	Year of the embeddings to map. By default, the most common year in 'data'.
bg_ratio	Absence-to-presence ratio for the models that train on a balanced pool (rf, gbt and knn): their absences are thinned at random to 'bg_ratio' times the number of presences. Default 1. 'NULL' gives them every absence. svm and maxent always use every absence.
bg_replicates	Logical (default TRUE). Train the balanced-pool methods (rf, gbt, knn) on $k = \min(10, \text{ceil}(10000/\text{pool size}))$ replicate thinned subsets of the absences and average their predictions (Barbet-Massin et al. 2012, Table 1: several runs when few pseudo-absences are used). Requires 'bg_ratio' thinning to be active; methods on the full pool are never replicated.
params	Named list of settings for individual models, using the argument names of the Earth Engine classifier, for example 'list(gbt = list(shrinkage = 0.01), svm = list(kernelType = "RBF"))'. Every model uses Earth Engine's defaults except where Earth Engine needs a value or its default cannot work on these data: 'rf' uses 500 trees and 'gbt' 150, since Earth Engine requires a number, and 'knn' uses 15 neighbours, since Earth Engine's single neighbour gives a two-value map. See the Earth Engine reference for 'ee.Classifier.smileRandomForest', 'smileGradientTreeBoost', 'libsvm', 'amnhMaxent', 'smileKNN' and 'smileCart' for every available setting.
gee_project	Optional Earth Engine project override (normally set via [setup_gee()]).

Value

A named list of output file paths, with one '<method>_map' entry per model, plus 'ensemble_map' when more than one method is requested.

Examples

```
## Not run:
maps <- generate_map(occ, aoi = "bbox", scale = 30, output_dir = tempdir())
maps$ensemble_map

## End(Not run)
```

generate_pseudo_absences

Generate pseudo-absences for presence-only data

Description

Presence-only records cannot be modelled directly: every method in AlphaSDM needs absence or background data, and how that background is placed is a modelling decision with real consequences (Barbet-Massin et al. 2012, *Methods in Ecology and Evolution* 3:327-338). This function makes that decision explicit. It draws pseudo-absences inside ‘aoi’ under the strategy you choose, reports every threshold it used, and returns your presences and the new absences as one data frame ready for [evaluate_models()] or [generate_map()].

Usage

```
generate_pseudo_absences(
  data,
  aoi,
  strategy,
  n = 10000L,
  radius_m = NULL,
  env_threshold = NULL,
  aoi_year = NULL,
  scale = 10,
  seed = 0L,
  gee_project = NULL
)
```

Arguments

data	Formatted presence records from [format_data()] (standard ‘longitude’, ‘latitude’, ‘year’, ‘present’ columns, all presences). The workflow is format first, then add absences: <pre>pres <- format_data(obs, coords = c("lon", "lat"), year = "yr") data <- generate_pseudo_absences(pres, aoi = ..., strategy = ...) evaluate_models(data)</pre>
aoi	Where absences may be placed: an ‘ee.Geometry’, a ‘list(lon, lat, radius)’, a path to a vector file, or the string “bbox” to use the presence bounding box (an explicit choice, not a silent default; a bounding box is rarely the right availability frame for clustered records).
strategy	One of “random”, “disk”, “envelope”, “combined”. No default: this is the modelling decision.
n	Number of pseudo-absences (default 10000, Barbet-Massin et al. 2012; use about the presence count for “combined” feeding tree methods).
radius_m	Disk radius in metres; NULL estimates it from the embedding-autocorrelation range and reports it.
env_threshold	Mahalanobis envelope threshold; NULL uses the bias-corrected presence maximum and reports it.
aoi_year	Embedding year to read the pseudo-absences from. By default (‘NULL’) they are spread over the presences’ years in the same proportions, so presences and absences come from the same years’ embeddings. Give one year to place them all in that year.

scale	Sampling scale in metres (default 10).
seed	Integer seed for the draws.
gee_project	Optional Earth Engine cloud project.

Details

Strategies, following Barbet-Massin et al. (2012):

"random" Uniform over the AOI. Their recommendation for regression-style methods and MaxEnt (with 'n = 10000'). MaxEnt is not in the default ensemble for exactly this reason: give it its own random set and run 'methods = "maxent"' separately.

"disk" Their "2-degree-far": only beyond a distance from every presence. 'radius_m = NULL' estimates the distance at which embedding similarity to the presences decays to the regional baseline, and reports it; pass a number to choose it yourself.

"envelope" Their SRE, in embedding space: only outside the presence environmental envelope, measured as Mahalanobis distance to the presence cloud. 'env_threshold = NULL' uses the bias-corrected maximum distance among the presences themselves.

"combined" Both exclusions at once. Their recommendation for classification and machine-learning methods (rf, gbt) with 'n' near the number of presences; validated here on Bicknell's Thrush (real-absence AUC) and *Prunus africana* (Boyce index).

Supply is guaranteed: every strategy redraws until 'n' points with satellite coverage survive the active exclusions; "disk" and "combined" halve the radius stepwise rather than come up short (the envelope never relaxes, since points inside it are the likely false absences the strategy exists to avoid).

Value

A data frame with 'longitude', 'latitude', 'year', 'present' (your presences as 1, pseudo-absences as 0), ready for [evaluate_models()] or [generate_map()] directly, carrying the settings used in 'attr("pa_settings")'.

Examples

```
## Not run:
pres <- format_data(records, coords = c("lon", "lat"), year = "year")
occ <- generate_pseudo_absences(pres, aoi = "bbox", strategy = "combined",
                               n = nrow(pres))

table(occ$present)

## End(Not run)
```

setup_gee

Connect AlphaSDM to Google Earth Engine (one-time)

Description

Signs in to Google Earth Engine with your own Google account and saves your project ID, so later R sessions connect on their own. Run it once per machine; running it again when already connected does nothing.

Usage

```
setup_gee(project = NULL, force = FALSE, auth_mode = NULL)
```

Arguments

project	Earth Engine (Google Cloud) project ID, for example "my-ee-project". If NULL, the saved project is used, or you are asked for one.
force	If TRUE, sign in again even if valid credentials exist.
auth_mode	Earth Engine sign-in flow, passed to ee.Authenticate(). Leave NULL to use "localhost" (a browser click) where a browser is available and "notebook" (paste a code) where it is not. "gcloud" uses the gcloud command-line tool.

Details

You need a free Earth Engine account first: register at <https://earthengine.google.com/signup/>. Earth Engine is free for noncommercial, research, education and nonprofit use, and registration gives you the Cloud project ID to pass as project.

Signing in opens your browser, where you click **Allow**. The Earth Engine client stores the resulting credentials in its own configuration folder, as it does for every tool that uses Earth Engine. AlphaSDM saves only the project ID, in `tools::R_user_dir("AlphaSDM", "config")`.

The Earth Engine Python client (`earthengine-api`) is provided through **reticulate**, which sets up a Python environment for it the first time it is needed, unless you have pointed `reticulate` at a Python of your own.

Value

Invisibly, TRUE once connected.

Examples

```
## Not run:
# Needs an Earth Engine account and an interactive session.
setup_gee(project = "my-ee-project")

## End(Not run)
```

Index

`calculate_classifier_metrics`, [2](#)
`clear_gee_credentials`, [3](#)

`evaluate_models`, [3](#)

`format_data`, [5](#)

`gee_clean_assets`, [6](#)
`gee_status`, [7](#)
`gee_tasks`, [7](#), [7](#)
`generate_map`, [8](#)
`generate_pseudo_absences`, [9](#)

`setup_gee`, [3](#), [12](#)