

Package: luh2impact (via r-universe)

September 28, 2026

Type Package

Title Process LUH2 Land Use Data for IMPACT Model Integration

Version 3.2.0

Description Tools for preparing land use inputs for the IMPACT model and its LUMEN land use downscaler. Two pipelines are provided: a pixel-level one that turns LUH2 land use states, planted forest, and protected area rasters into gridded baseline areas, spatial weights, and a GDX file for LUMEN; and a country-level one that fits FAOSTAT land use elasticities against cropland and GDP per capita and exports baseline areas, protected area floors, and biodiversity intactness coefficients as IMPACT model parameters. Plotting helpers are included for the model outputs.

License CC BY 4.0

Encoding UTF-8

URL <https://github.com/IFPRI/luh2impact>

BugReports <https://github.com/IFPRI/luh2impact/issues>

Depends R (>= 4.1)

Imports terra (>= 1.9.1), dplyr (>= 1.1.4), tidyr (>= 1.3.1), ggplot2 (>= 4.0.3), tidyterra (>= 0.6.1), DOORMAT (>= 3.14.1), magrittr (>= 2.0.3), data.table (>= 1.14.8), openxlsx (>= 1.9), ggrepel (>= 0.9.6), ggpubr (>= 0.6.0), grDevices, stats, utils

Suggests gamstransfer

Remotes IFPRI/DOORMAT

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.2

Repository <https://ifpri.r-universe.dev>

Date/Publication 2026-09-28 19:34:33 UTC

RemoteUrl <https://github.com/IFPRI/luh2impact>

RemoteRef HEAD

RemoteSha 32c3e1ed1e1dfa4e9e2e1d1c2c22448dd58c875c

Contents

luh2_bii_coeff_cty	2
luh2_bii_coeff_cty_pixel	4
luh2_build_pixels	6
luh2_compute_gdppc	8
luh2_crop_trend	10
luh2_export_elasticities_gdx	11
luh2_export_gdx	13
luh2_extract_year	14
luh2_fit_elasticities	15
luh2_landuse_elasticities	17
luh2_load	20
luh2_load_faoland	21
luh2_merge_planted	22
luh2_natfor_primary_share_cty	23
luh2_pa_area_cty	25
luh2_plot_bii	26
luh2_plot_elasticities	27
luh2_plot_results	28
luh2_pool_trend	30
luh2_suitability	31
luh2_urban_area_cty	32
luh2_write_tifs	33
luh2gdx	34
wdpa_process	36

Index	39
--------------	-----------

luh2_bii_coeff_cty	<i>Compute per-country BII coefficients per land pool from the literature</i>
--------------------	---

Description

Blends the FAOSTAT-derived primary-forest share from `luh2_natfor_primary_share_cty()` with primary and secondary forest biodiversity intactness coefficients from Leclere et al. (2018), Figure 3, so that natfor gets a country-specific coefficient instead of one flat global value. The other pools (crop, past, other, plant, urban) keep the flat coefficients already used in the LUMEN pixel implementation (`bii_coeff` in `lumen.gms`), because no validated country-level split exists for them yet - see `luh2_natfor_primary_share_cty()` for what was tried and abandoned.

Usage

```
luh2_bii_coeff_cty(
  fao_land_csv,
  sets_xlsx,
  year = 2021,
  bii_primary_forested = 1,
```

```

    bii_secondary_forested = 0.85,
    bii_other = 0.888,
    bii_past = 0.791,
    bii_plant = 0.672,
    bii_crop = 0.576,
    bii_urban = 0.65
)

```

Arguments

fao_land_csv	Path to the FAOSTAT Inputs_LandUse_E_All_Data_(Normalized).csv file.
sets_xlsx	Path to the IMPACT Sets.xlsx workbook (see luh2_load_faoland()).
year	Integer. Year used for the natfor primary-forest share. Default 2021.
bii_primary_forested	Numeric. BII coefficient for primary forest. Default 1.0.
bii_secondary_forested	Numeric. BII coefficient for secondary forest (approximate, see Details). Default 0.85.
bii_other, bii_past, bii_plant, bii_crop	Numeric. Flat BII coefficients matching the existing bii_coeff table in lumen.gms. Defaults 0.888, 0.791, 0.672, 0.576.
bii_urban	Numeric. Flat BII coefficient for urban land (approximate, see Details). Default 0.65.

Details

The idea behind the blend is that BII compares current abundance against the same ecosystem's own undisturbed state, so a country whose remaining natural forest is largely primary should not score the same as one whose forest is largely regrowth. Only natfor is treated this way here.

Two defaults are approximate and worth revisiting if better numbers turn up. `bii_secondary_forested` (0.85) is read visually off Figure 3's forested-panel secondary-vegetation bars, excluding Timber, which maps to plant rather than natfor. `bii_urban` (0.65) is likewise a visual read of the Urban bars; `lumen.gms` offers no precedent for it, since urban land is held fixed there and never enters the pixel BII calculation. Neither value has been digitized from the source figure.

Countries with no reported naturally regenerating forest (so `natfor_primary_share` is NA) fall back to the midpoint of `bii_primary_forested` and `bii_secondary_forested`, rather than being dropped or assumed to be all primary or all secondary.

Value

A data frame with columns `cty`, `fland` and `value` (BII coefficient, 0 to 1), one row per country and pool, ready to pass as the `bii_coeff` argument of [luh2_export_elasticities_gdx\(\)](#).

Author(s)

Abhijeet Mishra, Claude Code

References

Leclere, D. et al. (2018). Towards pathways bending the curve of terrestrial biodiversity trends within the 21st century. IIASA.

See Also

[luh2_bii_coeff_cty_pixel\(\)](#) for the imagery-derived alternative, which varies by country for every pool.

Other country pipeline functions: [luh2_bii_coeff_cty_pixel\(\)](#), [luh2_compute_gdppc\(\)](#), [luh2_export_elasticities\(\)](#), [luh2_fit_elasticities\(\)](#), [luh2_landuse_elasticities\(\)](#), [luh2_load_faoland\(\)](#), [luh2_natfor_primary_share_cty\(\)](#), [luh2_pa_area_cty\(\)](#), [luh2_urban_area_cty\(\)](#)

Examples

```
## Not run:
bii <- luh2_bii_coeff_cty(
  "data/FAOSTAT/Inputs_LandUse_E_All_Data_(Normalized).csv",
  "data/IMPACT/Sets.xlsx"
)

## End(Not run)
```

```
luh2_bii_coeff_cty_pixel
```

Compute per-country BII coefficients per land pool from pixel imagery

Description

Reads a pre-computed pixel-derived BII_COEFF(cty, fland) table, built from Nguyen's 2020 land use and biodiversity intactness rasters, as an alternative to the literature-based [luh2_bii_coeff_cty\(\)](#). Unlike that function, every pool (crop, urban, past, plant, natfor, other) gets a country-specific value taken from imagery, not just natfor.

Usage

```
luh2_bii_coeff_cty_pixel(
  pixel_csv = system.file("extdata", "pixel_bii_coeff_fraction_r090.csv", package =
    "luh2impact")
)
```

Arguments

pixel_csv	Path to the pixel-derived BII coefficient CSV. Defaults to the fraction-based table shipped with the package. Pass <code>system.file("extdata", "pixel_bii_coeff_r090.csv", package = "luh2impact")</code> to use the original HHLU-based one instead.
-----------	--

Details

The default table is built from Nguyen's Landuse_Fraction rasters (0.02 degree fractional coverage per class: agro, crop, past, prim, secd, urbn), which are pixel-for-pixel co-registered with the BII raster (identical size, pixel size, origin and extent), so each pool's coefficient is a fraction-weighted mean BII score. It superseded an earlier table built from the raw 500 m High-Resolution Harmonized Land Use (HHLU) raster through a custom cross-resolution alignment; the two were cross-checked and agree to correlations above 0.9999 on every pool, with a maximum absolute difference under 0.008, so the fraction-based version was kept as the faster and better-aligned path. Both tables ship with the package and either can be passed to `pixel_csv`.

`natfor` and `other` cannot be separated from this pixel data alone: Nguyen's primary and secondary vegetation classes are organized by successional and management-intensity stage rather than by cover type (primary vegetation is any low-disturbance land, whether forest, grassland, desert or tundra, and secondary vegetation mixes regenerating natural forest with commercial timber plantations), and no grouping of the raw HHLU classes tracked this package's FAOSTAT-derived `natfor`/other area split well - the best of six groupings correlated at about 0.35. Instead, the combined natural-land score is split back into the two pools assuming $\text{other_coeff} = 0.90 * \text{natfor_coeff}$, solved per country so that the LAND00-area-weighted average of the two still reproduces the observed combined value. Where other area vastly exceeds `natfor` area (the arid countries DZA, GRL, LBY, MLI, MRT, NER and TCD), that algebra would push `natfor` above the physical ceiling of 1.0; those seven have `natfor` capped at 1.0 and `other` re-solved from the mass balance alone, which the `natfor_capped` column of the CSV flags.

Two caveats on how far these numbers can be pushed. Nguyen's fitted BII model retained only primary vegetation, cropland and urban fractions as significant land use predictors, alongside human-impact covariates (roads, population density, travel time, geographic distance); pasture and agroforestry fractions did not survive variable selection. So `past` and `plant` here are real values read off the predicted BII surface at those locations, but their variation comes from the surrounding primary, cropland and urban composition and the human-footprint covariates rather than from a pasture or agroforestry effect of their own. `crop` and `urban` rest on directly fitted effects and are the most solid of the six pools. And a handful of countries have no pixels at all for `past` (one country) or `plant` (three); those rows are dropped rather than returned as NA, matching the way LAND00 simply omits a pool a country does not have.

Value

A data frame with columns `cty`, `fland` and `value` (BII coefficient, 0 to 1), one row per country and pool with data - the same shape as `luh2_bii_coeff_cty()`'s output, ready to pass as the `bii_coeff` argument of `luh2_export_elasticities_gdx()`.

CSV format

Both shipped tables have one row per IMPACT country, a `cty` column, a `bii_<pool>` column per pool, the `natfor` and `other` areas used in the mass-balance split, and the `natfor_capped` flag. The fraction-based table additionally carries `wsum_*` (summed class fractions) and the intermediate `bii_prim`, `bii_secd` and `bii_natural_combined` columns; the HHLU-based one carries `n_*` pixel counts instead. Only `cty` and the six `bii_<pool>` columns are read by this function. The scripts that build these tables from the raw rasters are not part of the package.

Author(s)

Abhijeet Mishra, Claude Code

See Also

[luh2_bii_coeff_cty\(\)](#) for the literature-based alternative.

Other country pipeline functions: [luh2_bii_coeff_cty\(\)](#), [luh2_compute_gdppc\(\)](#), [luh2_export_elasticities_gdx\(\)](#), [luh2_fit_elasticities\(\)](#), [luh2_landuse_elasticities\(\)](#), [luh2_load_faoland\(\)](#), [luh2_natfor_primary_share_cty\(\)](#), [luh2_pa_area_cty\(\)](#), [luh2_urban_area_cty\(\)](#)

Examples

```
bii <- luh2_bii_coeff_cty_pixel()
head(bii)
```

luh2_build_pixels	<i>Build the pixel-level data frame with weights</i>
-------------------	--

Description

Rasterizes the IMPACT regions shapefile onto the LUH2 grid, merges every raster input into one pixel-level data frame, rebalances the pixel areas so they add up to the IMPACT country totals, and computes the spatial weights LUMEN uses to place land use change within a country. This is the step that produces everything [luh2_export_gdx\(\)](#) writes.

Usage

```
luh2_build_pixels(
  luh,
  shares,
  crop_trend,
  natfor_trend,
  other_trend,
  cellarea,
  icwtr,
  cty_shp,
  landx0,
  pa_tif = NULL
)
```

Arguments

luh	A <code>SpatRaster</code> of historical land use shares returned by luh2_load() , used here to derive the suitability index.
shares	A <code>SpatRaster</code> of adjusted land use shares returned by luh2_merge_planted() .
crop_trend	A single-layer <code>SpatRaster</code> of crop slope returned by luh2_pool_trend() .

natfor_trend	A single-layer SpatRaster of natural forest slope returned by <code>luh2_pool_trend()</code> .
other_trend	A single-layer SpatRaster of other land slope returned by <code>luh2_pool_trend()</code> .
cellarea	A single-layer SpatRaster of cell areas in km ² .
icwtr	A single-layer SpatRaster of ice/water fraction.
cty_shp	Path to the IMPACT regions shapefile. Must contain a NEW_REGION field used as the country identifier.
landx0	A data frame of IMPACT land use results with columns cty, fland, yrs, value (1000 ha), e.g. LANDX0 read from an IMPACT solution GDX. Must contain the year 2021.
pa_tif	Path to a WDPA protected area raster (km ² per pixel) on the LUH2 grid, as produced by <code>wdpa_process()</code> . If NULL (default), pa_area is set to 0 for every pixel.

Details

Pixel areas are reconciled with the IMPACT control totals by iterative proportional fitting: up to 20 passes that distribute each country's gap for a pool across its pixels in proportion to the pool's current pixel area, cap each pixel at its available land, and rescale when the pools together overshoot. The loop stops early once the largest country level absolute gap falls below 1000 ha, and progress is printed for each pass. The control totals are taken from landx0 for 2021, which is hard-coded.

Two other fixed choices are worth knowing about. Pixels are assigned to countries with touches = TRUE, so border pixels go to a country instead of dropping out as NA, and South Sudan (SSD) is removed because it is not an IMPACT region. Protected area per pixel is capped at 90 percent of available land, since WDPA polygon edges can otherwise report more protected area than a 0.25 degree cell contains.

Weights are area times the relevant historical trend times suitability, normalised to sum to 1 within each country, separately for expansion and contraction of crop, natural forest and other land. Pasture is weighted by area and suitability only, and planted forest by area and the mean planted area of neighbouring cells, because neither has a trend layer.

Value

A data frame with one row per land pixel and a unique pid, containing:

pid, cty, x, y Pixel identifier, IMPACT country, and cell centre coordinates.

carea_km2, carea_kha, icwtr, land_frac, avail_kha Cell area, ice/water fraction, and land available for allocation (care_kha * land_frac less urban), in 1000 ha unless named km2.

crop_area, natfor_area, past_area, other_area, plant_area, urban_area Rebalanced pool areas in 1000 ha.

*_share, sharetotal The same pools as shares of cell area, recomputed after rebalancing.

pa_area Protected area in 1000 ha, capped as described above.

crop_slope, natfor_slope, other_slope, plant_prox Trend and proximity inputs to the weights, with NA set to 0.

suit_crop, suit_natfor, suit_past, suit_other Suitability index from `luh2_suitability()`.

w_crop_exp, w_crop_con, w_natfor_loss, w_natfor_gain, w_other_loss, w_other_gain, w_past, w_plant Allocation weights, each summing to 1 within a country.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other pixel pipeline functions: [luh2_crop_trend\(\)](#), [luh2_export_gdx\(\)](#), [luh2_extract_year\(\)](#), [luh2_load\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2_suitability\(\)](#), [luh2gdx\(\)](#), [wdpa_process\(\)](#)

luh2_compute_gdppc	<i>Compute IMPACT-country GDP per capita from SSP data, with a World Bank fallback</i>
--------------------	--

Description

Builds an annual GDP per capita series (thousand USD) per IMPACT country from the SSP GDP and population projections in an SSP scenario GDX, aggregated from SSP country level to IMPACT regions. IMPACT countries that the SSP mapping does not reach are backfilled from World Bank GDP per capita (PPP), with manual extrapolations for Afghanistan, Syria and Venezuela - see Details.

Usage

```
luh2_compute_gdppc(  
  ssp_gdx,  
  fao_country_map_csv,  
  sets_xlsx,  
  wb_gdppc_csv,  
  ssp = "SSP2",  
  year_min = 1995,  
  year_max = 2023  
)
```

Arguments

ssp_gdx	Path to the SSP scenario GDX file (must contain the OECD_GDP and POP symbols).
fao_country_map_csv	Path to a FAOSTAT country-code-to-ISO3 mapping CSV; columns 4 and 8 are read as the FAO country code and the ISO3 code.
sets_xlsx	Path to the IMPACT Sets.xlsx workbook (see luh2_load_faoland()).
wb_gdppc_csv	Path to the World Bank NY.GDP.PCAP.PP.CD GDP per capita (PPP) CSV, in the standard World Bank layout with four header lines and one column per year.
ssp	Character. SSP scenario to use. Default "SSP2".
year_min, year_max	Integer. Year range to retain from the World Bank series. Defaults 1995 and 2023.

Details

GDP is taken from the GDP|PPP variable of OECD_GDP and population from POP (both genders, total tranche), each summed to IMPACT region before dividing. Hong Kong, Macao and Taiwan are pooled with mainland China (FAO country code 351) so that the SSP series line up with the IMPACT region definition. Countries with no SSP entry take the World Bank series instead; the two sources are row-bound, so a country appears from one or the other, never both.

Manual CAGR-based backfills and forward-fills are applied after the World Bank merge, replicating analyst adjustments from the original elasticity workbook where World Bank coverage does not reach back to year_min (Afghanistan, Syria) or does not capture recent years cleanly (Venezuela, because of the hyperinflation-era swings):

- AFG: years year_min to 1999 back-filled at the 1999-2024 CAGR $(2201/813)^{(1/25)}-1$.
- SYR: years year_min to 2016 back-filled at the 2016-2022 CAGR $(4772/3265)^{(1/6)}-1$; 2023 set manually to 4800 (current USD, before scaling).
- VEN: years 2012 to 2023 forward-filled at a 55-year CAGR $(21241/13889)^{(1/55)}-1$.

These three adjustments are hard-coded, including their years and endpoint values, and are not re-derived from the data.

Value

A data frame with columns cty, year, value (thousand USD per capita), item (always "gdppc") and unit (always "000USD") - the same shape as `luh2_load_faoland()`'s output, so the two can be row-bound before fitting.

Requirements

Reads the SSP GDX with `DOORMAT::readGDX()`, which needs a working GDX backend.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: `luh2_bii_coeff_cty()`, `luh2_bii_coeff_cty_pixel()`, `luh2_export_elasticities`, `luh2_fit_elasticities()`, `luh2_landuse_elasticities()`, `luh2_load_faoland()`, `luh2_natfor_primary_share_cty`, `luh2_pa_area_cty()`, `luh2_urban_area_cty()`

Examples

```
## Not run:
gdppc <- luh2_compute_gdppc(
  ssp_gdx = "data/SSP/ssp.gdx",
  fao_country_map_csv = "data/FAOSTAT/country_codes.csv",
  sets_xlsx = "data/IMPACT/Sets.xlsx",
  wb_gdppc_csv = "data/WorldBank/API_NY.GDP.PCAP.PP.CD.csv"
)

## End(Not run)
```

luh2_crop_trend

Compute the per-pixel crop expansion trend

Description

Fits a per-pixel linear trend over time to the summed crop land use shares (c3ann, c4ann, c3per, c4per, c3nfx) and returns the slope, i.e. the change in crop share per year over the requested period.

Usage

```
luh2_crop_trend(luh, year_start, year_end)
```

Arguments

luh	A SpatRaster returned by luh2_load() .
year_start	Integer. First year of the trend period (e.g. 1990).
year_end	Integer. Last year of the trend period (e.g. 2015).

Details

This is the crop-only special case of [luh2_pool_trend\(\)](#), kept because it is convenient on its own. [luh2_pool_trend\(\)](#) is faster (it solves the OLS slope analytically instead of calling `stats::lm()` per pixel) and is what [luh2gdx\(\)](#) uses for all pools, crop included.

Value

A single-layer SpatRaster named "crop_slope", in share per year. Pixels that are NA in every year stay NA.

Author(s)

Abhijeet Mishra, Claude Code

See Also

[luh2_pool_trend\(\)](#) for the general, faster version.

Other pixel pipeline functions: [luh2_build_pixels\(\)](#), [luh2_export_gdx\(\)](#), [luh2_extract_year\(\)](#), [luh2_load\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2_suitability\(\)](#), [luh2gdx\(\)](#), [wdpa_process\(\)](#)

Examples

```
## Not run:
luh <- luh2_load("data/LUH2/states.nc")
crop_slope <- luh2_crop_trend(luh, year_start = 1990, year_end = 2015)

## End(Not run)
```

luh2_export_elasticities_gdx

Export land use elasticities and baseline areas to GDX

Description

Writes the fitted elasticities, their diagnostics, baseline FAOSTAT land areas and the optional protected area and biodiversity parameters into a single GDX file for the IMPACT land use module.

Usage

```
luh2_export_elasticities_gdx(
  results,
  faoland,
  output_gdx,
  base_year = 2021,
  extra_land00 = NULL,
  pa00 = NULL,
  bii_coeff = NULL,
  author = NULL
)
```

Arguments

results	The results data frame from <code>luh2_fit_elasticities()</code> .
faoland	A data frame returned by <code>luh2_load_faoland()</code> , used for the baseline land areas in <code>base_year</code> .
output_gdx	Path to write the output GDX file.
base_year	Integer. Year used for the LAND00 baseline. Default 2021.
extra_land00	Optional data frame with columns <code>cty</code> , <code>item</code> , <code>value</code> appended to LAND00 as additional baseline pools that are genuinely distinct land covers (e.g. <code>item = "urban"</code> from <code>luh2_urban_area_cty()</code>). These pools get no LANDELAS entry - they are fixed stocks, not fitted against cropland and income. Protected area does not belong here: it is not a distinct cover, it overlays natfor, past and other, so it goes through <code>pa00</code> instead. Default NULL.
pa00	Optional data frame with columns <code>cty</code> and <code>value</code> (aggregate protected non-crop land area in 1000 ha), e.g. from <code>luh2_pa_area_cty()</code> . Written as LANDPA00(<code>cty</code>), a floor on protected land rather than a LAND00 pool. Default NULL.
bii_coeff	Optional data frame with columns <code>cty</code> , <code>fland</code> and <code>value</code> (0 to 1), e.g. from <code>luh2_bii_coeff_cty()</code> or <code>luh2_bii_coeff_cty_pixel()</code> . Written as BII_COEFF(<code>cty</code> , <code>fland</code>). Default NULL.
author	Character. Recorded in the GDX as provenance. Defaults to the OS user running the session (<code>Sys.info()[["user"]]</code>) rather than the package maintainer, since the point is to record who generated this particular file once it is version controlled and several people can run the pipeline.

luh2_export_gdx	<i>Export pixel data and IMPACT totals to GDX</i>
-----------------	---

Description

Writes every set and parameter the LUMEN GAMS model reads into a single GDX file. This is the last step of the pixel pipeline; the file it writes is what LUMEN is run against.

Usage

```
luh2_export_gdx(df, landx0, output_gdx)
```

Arguments

df	A pixel-level data frame returned by <code>luh2_build_pixels()</code> .
landx0	A data frame of IMPACT land use results with columns <code>cty</code> , <code>fland</code> , <code>yrs</code> , <code>value</code> (1000 ha) - the same object passed to <code>luh2_build_pixels()</code> .
output_gdx	Path to write the output GDX file.

Details

Sets written: `pid` (pixels), `cty` (IMPACT countries), `fland` (the five allocated pools `crop`, `natfor`, `past`, `other`, `plant`), `yrs`, `coord`, and the `pid_cty` pixel-to-country map.

Parameters written, all areas in 1000 ha:

`luh2_base(pid,fland)` Baseline pixel area per pool.

`urban_area(pid)` Urban area per pixel, held fixed by LUMEN.

`carearea(pid)` Cell area.

`avail_kha(pid)` Land available for allocation, excluding urban land and ice/water.

`impact_total(cty,fland,yrs)` IMPACT control totals that the pixel allocation must reproduce.

`pid_coords(pid,coord)` Pixel centre coordinates, used to map the solution back to a raster in `luh2_write_tifs()`.

`w_crop_exp`, `w_crop_con`, `w_natfor_loss`, `w_natfor_gain`, `w_other_loss`, `w_other_gain`, `w_past`, `w_plant` (**all** (`pid`)) Allocation weights, each summing to 1 within a country.

`pa_area(pid)` **and** `pa_area_cty(cty)` WDPA protected area per pixel and its country total.

`suit(pid,fland)` Historical suitability index, 0 to 1.

Note that `urban` is written as its own parameter rather than as a member of `fland`, because LUMEN does not reallocate it.

Value

Invisibly returns `output_gdx`. Called for its side effect of writing that file.

Requirements

Needs the gamstransfer package, which is shipped with GAMS rather than installed from CRAN.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other pixel pipeline functions: `luh2_build_pixels()`, `luh2_crop_trend()`, `luh2_extract_year()`, `luh2_load()`, `luh2_merge_planted()`, `luh2_pool_trend()`, `luh2_suitability()`, `luh2gdx()`, `wdpa_process()`

luh2_extract_year	<i>Extract land use areas for a target year</i>
-------------------	---

Description

Subsets the LUH2 raster to a single year, multiplies the shares by cell area to get areas, and returns them together with the static grid layers (cell area, ice/water fraction, country code, forest/non-forest flag) that the rest of the pixel pipeline needs.

Usage

```
luh2_extract_year(luh, static_nc, year)
```

Arguments

luh	A SpatRaster returned by <code>luh2_load()</code> .
static_nc	Path to the LUH2 staticData_quarterdeg.nc file.
year	Integer. Target year (e.g. 2015).

Details

LUH2 layers are named <state>_<index>, where <index> counts years from the first year in the file. The index for year is derived from `terra::time(luh)[1]`, so the function works for any LUH2 release as long as the time axis is set. An error is thrown if the requested year is not in the file.

Value

A named list of SpatRaster objects:

luarea	Land use areas in km ² , one layer per LUH2 state, for year only (layer names have the year suffix stripped).
cellarea	Single layer. Grid cell area in km ² (carea).
icwtr	Single layer. Ice and water fraction of the cell (icwtr), between 0 and 1.

`ccode` Single layer. LUH2 country code per cell (`ccode`); 0 over ocean.

`fstnf` Single layer. Forest/non-forest flag (`fstnf`), 1 where the potential vegetation is forest, used to split natural land into `natfor` and other in [luh2_merge_planted\(\)](#).

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other pixel pipeline functions: [luh2_build_pixels\(\)](#), [luh2_crop_trend\(\)](#), [luh2_export_gdx\(\)](#), [luh2_load\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2_suitability\(\)](#), [luh2gdx\(\)](#), [wdpa_process\(\)](#)

Examples

```
## Not run:
luh <- luh2_load("data/LUH2/states.nc")
extracted <- luh2_extract_year(luh, "data/LUH2/staticData_quarterdeg.nc", year = 2015)
extracted$luarea

## End(Not run)
```

`luh2_fit_elasticities` *Fit land use elasticities to cropland and GDP per capita*

Description

Builds a per-country annual panel of land use areas and GDP per capita, fills interior gaps by linear interpolation, first-differences the logs, and fits one OLS regression per country and per non-cropland pool of that pool's log change on the log changes in cropland and GDP per capita.

Usage

```
luh2_fit_elasticities(faoland, gdppc, year_max = 2023, min_obs = 10)
```

Arguments

<code>faoland</code>	A data frame returned by luh2_load_faoland() .
<code>gdppc</code>	A data frame returned by luh2_compute_gdppc() .
<code>year_max</code>	Integer. Latest year to include in the panel. Default 2023.
<code>min_obs</code>	Integer. Minimum number of complete observations needed to fit a country's regression. Default 10.

Details

The fitted model, for each country and each of past, natfor, other and plant, is $dln_use \sim dln_crop + dln_gdppc$, where dln_x is the year-on-year change in $\log(x)$. The coefficients are therefore elasticities of that pool's area with respect to cropland area and income. They are descriptive associations within a country's own history, estimated without controls or instruments, and should not be read as causal effects.

Data handling worth knowing about: years missing inside a country's range are linearly interpolated (leading and trailing gaps are left alone); non-positive areas become NA before the logs are taken; only years present in both the FAOSTAT and GDP per capita inputs are used. Estimates are clipped to $[-4, 4]$ for cropland and $[-2, 2]$ for GDP per capita, which bounds the small-sample tails rather than describing the data.

Every country and pool appears in the output, including those that could not be fitted, distinguished by flag:

ok Fitted, both terms present.

insufficient_obs Fewer than min_obs complete observations; both elasticities set to 0.

model_failed `stats::lm()` errored; both elasticities set to 0.

dropped_term Fitted, but a term was dropped (collinearity), and the missing elasticity is reported as 0.

`luh2_export_elasticities_gdx()` writes all rows as LANDELAS_RAW_* but restricts LANDELAS, the parameter the model actually solves with, to ok rows.

Value

A list with two elements:

results One row per country and land use, with `elast_crop` and `elast_gdppc` (clipped as above), their standard errors (`se_crop`, `se_gdppc`) and p-values (`pval_crop`, `pval_gdppc`), `n_obs`, and `flag`. Standard errors and p-values are NA where no model was fitted.

wide_imputed The imputed panel the regressions were fitted to (cty, year, land, crop, past, natfor, plant, other, gdppc), kept so the inputs can be inspected.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: `luh2_bii_coeff_cty()`, `luh2_bii_coeff_cty_pixel()`, `luh2_compute_gdppc()`, `luh2_export_elasticities_gdx()`, `luh2_landuse_elasticities()`, `luh2_load_faoland()`, `luh2_natfor_primary_share_cty()`, `luh2_pa_area_cty()`, `luh2_urban_area_cty()`

Examples

```
## Not run:
fit <- luh2_fit_elasticities(faoland, gdppc)
table(fit$results$flag)
```

```
## End(Not run)
```

```
luh2_landuse_elasticities
```

Run the full FAOSTAT land use elasticity pipeline

Description

Convenience wrapper that loads FAOSTAT land use and GDP per capita data, fits per-country land use elasticities, optionally adds urban land, protected area and biodiversity coefficients, and writes the IMPACT input GDX. Each step is also exported and can be run on its own.

Usage

```
luh2_landuse_elasticities(
  fao_land_csv,
  sets_xlsx,
  ssp_gdx,
  fao_country_map_csv,
  wb_gdppc_csv,
  output_gdx,
  ssp = "SSP2",
  year_min = 1995,
  year_max = 2023,
  base_year = 2021,
  min_obs = 10,
  plot_dir = NULL,
  region_map_xlsx = NULL,
  region = "WLD",
  pa_tif = NULL,
  gdb_path = NULL,
  states_nc = NULL,
  static_nc = NULL,
  cty_shp = NULL,
  wdpa_year = 2021,
  urban_year = NULL,
  bii_year = NULL,
  bii_pixel_csv = NULL
)
```

Arguments

fao_land_csv	Path to the FAOSTAT Inputs_LandUse_E_All_Data_(Normalized).csv file.
sets_xlsx	Path to the IMPACT Sets.xlsx workbook. Must contain a Regions sheet mapping FAO country codes (FCTY) to IMPACT regions (CTY).

ssp_gdx	Path to the SSP scenario GDX file (must contain the OECD_GDP and POP symbols).
fao_country_map_csv	Path to a FAOSTAT country-code-to-ISO3 mapping CSV; columns 4 and 8 are read as the FAO country code and the ISO3 code.
wb_gdppc_csv	Path to the World Bank NY.GDP.PCAP.PP.CD GDP per capita (PPP) CSV, in the standard World Bank layout with four header lines and one column per year.
output_gdx	Path to write the output GDX file.
ssp	Character. SSP scenario to use. Default "SSP2".
year_min	Integer. Earliest year to retain. Default 1995.
year_max	Integer. Latest year to include when fitting. Default 2023.
base_year	Integer. Year used for the baseline land areas in the GDX. Default 2021.
min_obs	Integer. Minimum observations required per country regression. Default 10.
plot_dir	Optional directory to save the elasticity distribution plots. If NULL (default), no plots are produced.
region_map_xlsx, region	Passed to luh2_plot_elasticities() .
pa_tif	Optional path to an already-processed WDPA raster (km ² per pixel, the output of wdpa_process(..., output = "pixel")). Requires <code>cty_shp</code> . Takes precedence over <code>gdb_path</code> . Default NULL.
gdb_path	Optional path to the WDPA geodatabase (.gdb folder), used only when <code>pa_tif</code> is not supplied. Requires <code>static_nc</code> and <code>cty_shp</code> , and is considerably slower. Default NULL.
states_nc	Optional path to the LUH2 <code>states.nc</code> file. Requires <code>static_nc</code> and <code>cty_shp</code> . When supplied, urban area is added to <code>LAND00</code> . Default NULL.
static_nc	Path to the LUH2 <code>staticData_quarterdeg.nc</code> file. Required when <code>states_nc</code> , or <code>gdb_path</code> without <code>pa_tif</code> , is supplied.
cty_shp	Path to the IMPACT regions shapefile. Required when <code>pa_tif</code> , <code>gdb_path</code> or <code>states_nc</code> is supplied.
wdpa_year	Integer. Only PAs designated by this year are included when processing from <code>gdb_path</code> (see wdpa_process()). Default 2021.
urban_year	Integer. Year for the urban area aggregation. Defaults to <code>base_year</code> when NULL.
bii_year	Integer. Year used for the <code>natfor</code> primary-forest share (see luh2_bii_coeff_cty()). Defaults to <code>base_year</code> when NULL, and is ignored when <code>bii_pixel_csv</code> is supplied.
bii_pixel_csv	Optional path to a pixel-derived BII coefficient CSV (see luh2_bii_coeff_cty_pixel()). When supplied, <code>BII_COEFF</code> is built from that imagery-derived table instead of the literature-based luh2_bii_coeff_cty() . The table shipped with the package is at <code>system.file("extdata", "pixel_bii_coeff_fraction_r090.csv", package = "luh2impact")</code> . Default NULL.

Details

Only the elasticity core is mandatory. The optional pieces switch on when their inputs are supplied:

- Urban land (`item = "urban"` in `LAND00`, via `luh2_urban_area_cty()`) is added when `states_nc` is given. It is a fixed baseline cover and gets no fitted elasticity.
- Protected area is written as an aggregate `LANDPA00(cty)` floor, not as a `LAND00` item, because it is not a land cover of its own - it overlays `natfor`, `past` and `other`. Supply either `pa_tif` (fast, via `luh2_pa_area_cty()`) or `gdb_path` (slow, via `wdpa_process()`); `pa_tif` wins if both are given.
- BII coefficients (`BII_COEFF(cty, fland)`) are always computed and written, since `luh2_bii_coeff_cty()` needs only `fao_land_csv` and `sets_xlsx`, which are required anyway. Supply `bii_pixel_csv` to use the imagery-derived coefficients from `luh2_bii_coeff_cty_pixel()` instead.

Inputs are validated before any of the slow steps run: `cty_shp` is required whenever `pa_tif`, `gdb_path` or `states_nc` is supplied, and `static_nc` whenever `states_nc` or a bare `gdb_path` is.

Value

Invisibly, a list with results (the elasticity estimates from `luh2_fit_elasticities()`) and `output_gdx` (the path written).

Requirements

Needs the `gamstransfer` package, which is shipped with GAMS rather than installed from CRAN.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: `luh2_bii_coeff_cty()`, `luh2_bii_coeff_cty_pixel()`, `luh2_compute_gdppc()`, `luh2_export_elasticities_gdx()`, `luh2_fit_elasticities()`, `luh2_load_faoland()`, `luh2_natfor_primary_share()`, `luh2_pa_area_cty()`, `luh2_urban_area_cty()`

Examples

```
## Not run:
luh2_landuse_elasticities(
  fao_land_csv      = "data/FAOSTAT/Inputs_LandUse_E_All_Data_(Normalized).csv",
  sets_xlsx         = "data/IMPACT/Sets.xlsx",
  ssp_gdx           = "data/SSP/ssp.gdx",
  fao_country_map_csv = "data/FAOSTAT/country_codes.csv",
  wb_gdppc_csv      = "data/WorldBank/API_NY.GDP.PCAP.PP.CD.csv",
  output_gdx        = "outputs/landuse.gdx",
  states_nc         = "data/LUH2/states.nc",
  static_nc         = "data/LUH2/staticData_quarterdeg.nc",
  cty_shp           = "data/IMPACT/impact_regions.shp",
  pa_tif            = "data/WDPA/pa_area_km2.tif"
```

```
)
## End(Not run)
```

luh2_load

Load and clean the LUH2 states NetCDF

Description

Reads the LUH2 `states.nc` file and drops the two layers that are not land use shares (`secma`, secondary mean age, and `secmb`, secondary mean biomass), so that every remaining layer is a `<pool>_<year index>` share between 0 and 1.

Usage

```
luh2_load(states_nc)
```

Arguments

`states_nc` Path to the LUH2 `states.nc` file.

Value

A `SpatRaster` with one layer per land use state and year, named `<state>_<index>` where `<index>` is the 1-based year offset from the first year of the file (850 for the LUH2 historical release).

Author(s)

Abhijeet Mishra, Claude Code

See Also

[luh2_extract_year\(\)](#) to convert shares to areas for one year.

Other pixel pipeline functions: [luh2_build_pixels\(\)](#), [luh2_crop_trend\(\)](#), [luh2_export_gdx\(\)](#), [luh2_extract_year\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2_suitability\(\)](#), [luh2gdx\(\)](#), [wdpa_process\(\)](#)

Examples

```
## Not run:
luh <- luh2_load("data/LUH2/states.nc")
names(luh)[1:5]

## End(Not run)
```

luh2_load_faoland	<i>Load FAOSTAT land use data and map it to IMPACT countries</i>
-------------------	--

Description

Reads the FAOSTAT "Inputs: Land Use" bulk download, keeps the items used for elasticity estimation, maps FAO countries to IMPACT regions, and sums to IMPACT country level. This is the starting point of the country-level pipeline.

Usage

```
luh2_load_faoland(fao_land_csv, sets_xlsx, year_min = 1995)
```

Arguments

fao_land_csv	Path to the FAOSTAT Inputs_LandUse_E_All_Data_(Normalized).csv file.
sets_xlsx	Path to the IMPACT Sets.xlsx workbook. Must contain a Regions sheet mapping FAO country codes (FCTY) to IMPACT regions (CTY).
year_min	Integer. Earliest year to retain. Default 1995.

Details

Only records reported in 1000 ha are kept, and FAOSTAT items are renamed to the pool names used throughout the package:

land Land area - the country total, not one of the pools.

crop Cropland.

past Permanent meadows and pastures.

natfor Naturally regenerating forest.

plant Planted Forest.

other Other land.

The FAO country code to IMPACT region mapping is read from the Regions sheet of sets_xlsx (columns FCTY and CTY), and FAO countries with no IMPACT region are dropped rather than pooled into a residual. Several FAO countries can map to one IMPACT region, in which case their areas are summed.

Value

A data frame with columns cty, item, year, unit and value (1000 ha), one row per country, item and year.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: [luh2_bii_coeff_cty\(\)](#), [luh2_bii_coeff_cty_pixel\(\)](#), [luh2_compute_gdppc\(\)](#), [luh2_export_elasticities_gdx\(\)](#), [luh2_fit_elasticities\(\)](#), [luh2_landuse_elasticities\(\)](#), [luh2_natfor_primary_share_cty\(\)](#), [luh2_pa_area_cty\(\)](#), [luh2_urban_area_cty\(\)](#)

Examples

```
## Not run:
faoland <- luh2_load_faoland(
  fao_land_csv = "data/FAOSTAT/Inputs_LandUse_E_All_Data_(Normalized).csv",
  sets_xlsx    = "data/IMPACT/Sets.xlsx"
)
head(faoland)

## End(Not run)
```

luh2_merge_planted	<i>Merge a planted forest layer into the LUH2 shares</i>
--------------------	--

Description

Aggregates the LUH2 states into the pools used downstream, splits natural land into forest and non-forest, then carves the planted forest area out of the existing pools so that the shares still sum to at most 1. LUH2 has no planted forest class of its own, so the area has to be taken from somewhere rather than added on top.

Usage

```
luh2_merge_planted(luarea, cellarea, planted_tif, fstnf, cty_shp)
```

Arguments

luarea	A <code>SpatRaster</code> of land use areas in km ² , the <code>luarea</code> element of the list returned by luh2_extract_year() .
cellarea	A single-layer <code>SpatRaster</code> of cell areas in km ² , the <code>cellarea</code> element of the same list.
planted_tif	Path to a planted forest GeoTIFF giving planted forest area in km ² per cell on the LUH2 grid (e.g. an SDPT-derived <code>sdpt_global.tif</code>).
fstnf	A single-layer <code>SpatRaster</code> forest/non-forest flag (1 where the potential vegetation is forest), the <code>fstnf</code> element of the list returned by luh2_extract_year() .
cty_shp	Path to the IMPACT regions shapefile. Must contain a <code>NEW_REGION</code> field used as the country identifier; only used here to identify South Africa for the exemption described above.

Details

Pools are built from the LUH2 states as: $\text{crop} = \text{c3ann} + \text{c4ann} + \text{c3per} + \text{c4per} + \text{c3nfx}$; $\text{natfor} = \text{primf} + \text{secdf} + \text{primn} + \text{secdn}$; $\text{range} = \text{range} + \text{pastr}$; $\text{urban} = \text{urban}$; $\text{other} = \text{primn} + \text{secdf} + \text{secdn}$. natfor and other deliberately overlap at this point and are then separated with the LUH2 forest/non-forest flag: natfor is multiplied by fstnf and other by $1 - \text{fstnf}$. South Africa (ZAF) is hard-coded as an exception and keeps both pools unscaled; the reason for that exception is not recorded in the source.

Planted forest is then carved out in a fixed order: first from other , then whatever is left from range . Any residual planted share is capped at the space still free after all LUH2 pools, so it can be silently reduced where the grid is already full.

Value

A `SpatRaster` of adjusted shares (0 to 1) with one layer per pool: `crop_share`, `natfor_share`, `range_share`, `urban_share`, `other_share` and `planted_forest_share`. `luh2_build_pixels()` renames `range_share` to `past_share` and `planted_forest_share` to `plant_share`.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other pixel pipeline functions: `luh2_build_pixels()`, `luh2_crop_trend()`, `luh2_export_gdx()`, `luh2_extract_year()`, `luh2_load()`, `luh2_pool_trend()`, `luh2_suitability()`, `luh2gdx()`, `wdpa_process()`

Examples

```
## Not run:
luh      <- luh2_load("data/LUH2/states.nc")
extracted <- luh2_extract_year(luh, "data/LUH2/staticData_quarterdeg.nc", 2015)
shares   <- luh2_merge_planted(
  luarea      = extracted$luarea,
  cellarea    = extracted$cellarea,
  planted_tif = "data/SDPT/sdpt_global.tif",
  fstnf       = extracted$fstnf,
  cty_shp     = "data/IMPACT/impact_regions.shp"
)

## End(Not run)
```

luh2_natfor_primary_share_cty

Compute the FAOSTAT-native primary-forest share of the natfor pool

Description

For each IMPACT country, computes the share of natfor (naturally regenerating forest) that is primary forest, using items already present in the FAOSTAT bulk land use file - no LUH2 data needed. It follows the FAO/FRA definitional hierarchy: naturally regenerating forest is primary forest plus other naturally regenerated ("secondary") forest, so secondary forest is Naturally regenerating forest - Primary Forest.

Usage

```
luh2_natfor_primary_share_cty(fao_land_csv, sets_xlsx, year = 2021)
```

Arguments

fao_land_csv	Path to the FAOSTAT Inputs_LandUse_E_All_Data_(Normalized).csv file.
sets_xlsx	Path to the IMPACT Sets.xlsx workbook (see luh2_load_faoland()).
year	Integer. Year to compute the share for. Default 2021.

Details

The output is a share, not an area, and is meant to be used as a blending weight against natfor's existing FAOSTAT total (as [luh2_bii_coeff_cty\(\)](#) does), not as a substitute for it.

The shares were spot-checked against countries with known large tracts of intact forest (Brazil, Canada, DR Congo, Indonesia, Russia) and come out high relative to a global median near 1 percent, which is consistent with where intact forest actually remains.

Equivalent splits for past, other and crop were tried and dropped: FAOSTAT's pasture sub-items are unpopulated for most countries, and LUH2's primary/secondary non-forest classification did not give a trustworthy signal for other (Saudi Arabia, for instance, came out with a near-zero primary share for desert land that has never been converted). This function stays deliberately narrow to the one split that was validated.

Value

A data frame with columns:

cty IMPACT country code.

natfor_primary_share Share of natfor that is primary forest, 0 to 1, or NA where no naturally regenerating forest is reported.

flag_natfor "ok", or "primary_exceeds_natregen" where Primary Forest exceeds Naturally regenerating forest - a FAOSTAT inconsistency, in which case the share is clipped to 1.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: [luh2_bii_coeff_cty\(\)](#), [luh2_bii_coeff_cty_pixel\(\)](#), [luh2_compute_gdppc\(\)](#), [luh2_export_elasticities_gdx\(\)](#), [luh2_fit_elasticities\(\)](#), [luh2_landuse_elasticities\(\)](#), [luh2_load_faoland\(\)](#), [luh2_pa_area_cty\(\)](#), [luh2_urban_area_cty\(\)](#)

Examples

```
## Not run:
shares <- luh2_natfor_primary_share_cty(
  "data/FAOSTAT/Inputs_LandUse_E_All_Data_(Normalized).csv",
  "data/IMPACT/Sets.xlsx"
)
summary(shares$natfor_primary_share)

## End(Not run)
```

luh2_pa_area_cty	<i>Aggregate a pre-processed WDPA raster to IMPACT country totals</i>
------------------	---

Description

Aggregates an already-processed protected area raster (the pixel-level output of [wdpa_process\(\)](#), in km² per pixel) to IMPACT country totals. Use this rather than [wdpa_process\(\)](#) whenever a PA raster already exists, since it skips the slow geodatabase intersection.

Usage

```
luh2_pa_area_cty(pa_tif, cty_shp)
```

Arguments

pa_tif	Path to a protected area raster (km ² per pixel) on the LUH2 grid, e.g. as produced by wdpa_process(..., output = "pixel") .
cty_shp	Path to the IMPACT regions shapefile. Must contain a NEW_REGION field used as the country identifier.

Details

Pixels are assigned to countries by rasterizing the IMPACT shapefile with `touches = TRUE`. Protected area that falls outside the shapefile's coverage is dropped, so country totals from this function sit below published global protected area figures.

Value

A data frame with columns `cty` and `value`, protected area in 1000 ha. Ready to pass as the `pa00` argument of [luh2_export_elasticities_gdx\(\)](#).

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: `luh2_bii_coeff_cty()`, `luh2_bii_coeff_cty_pixel()`, `luh2_compute_gdppc()`, `luh2_export_elasticities_gdx()`, `luh2_fit_elasticities()`, `luh2_landuse_elasticities()`, `luh2_load_faoland()`, `luh2_natfor_primary_share_cty()`, `luh2_urban_area_cty()`

Examples

```
## Not run:
pa00 <- luh2_pa_area_cty("data/WDPA/pa_area_km2.tif",
                        "data/IMPACT/impact_regions.shp")

## End(Not run)
```

luh2_plot_bii

Plot LUMEN biodiversity intactness results

Description

Reads the pixel-level biodiversity intactness index (`bii_pixel`) from a LUMEN solution GDX and draws a three-panel figure for one country:

Left BII in 2021, sequential green palette.

Centre BII in 2050, on the same scale as the 2021 panel.

Right Change between 2021 and 2050, diverging red-blue palette centred at zero and scaled symmetrically.

Usage

```
luh2_plot_bii(output_dir, cty_shp, save_png = TRUE)
```

Arguments

<code>output_dir</code>	Path to the directory holding <code>solution_lu.gdx</code> . Its base name is used as the IMPACT country code and in the output file name.
<code>cty_shp</code>	Path to the IMPACT regions shapefile, used for the country outline. Must contain a <code>NEW_REGION</code> field.
<code>save_png</code>	Logical. If <code>TRUE</code> (default), the combined figure is written to <code>output_dir</code> as <code><cty>-bii.png</code> .

Details

BII is computed inside LUMEN from the per-pool BII coefficients and the solved pixel land use, so what is mapped here is the model's response to land use reallocation, not an observation. The country is taken from `basename(output_dir)`, and Alaska and Hawaii are cropped out for USA.

Value

Invisibly, the combined `ggpubr::ggarrange()` figure.

Requirements

Needs the gamstransfer package, which is shipped with GAMS rather than installed from CRAN.

Author(s)

Abhijeet Mishra, Claude Code

See Also

[luh2_bii_coeff_cty\(\)](#) and [luh2_bii_coeff_cty_pixel\(\)](#) for the coefficients the model uses.

Other output functions: [luh2_plot_elasticities\(\)](#), [luh2_plot_results\(\)](#), [luh2_write_tifs\(\)](#)

Examples

```
## Not run:
luh2_plot_bii("outputs/CHM", "data/IMPACT/impact_regions.shp")

## End(Not run)
```

luh2_plot_elasticities

Plot land use elasticity distributions

Description

Visualises the country-level elasticities from [luh2_fit_elasticities\(\)](#): a boxplot by driver and land use with the outlying countries labelled, and a histogram of the same values.

Usage

```
luh2_plot_elasticities(
  results,
  region_map_xlsx = NULL,
  region = "WLD",
  output_dir = NULL
)
```

Arguments

results	The results data frame from luh2_fit_elasticities() .
region_map_xlsx	Optional path to a region mapping workbook with columns cty and region, used to restrict the plot to one region. If NULL (default), every country in results is plotted.
region	Character. Region to keep when region_map_xlsx is supplied. Default "WLD".
output_dir	Optional directory in which to save elas_box.png and elas_hist.png. If NULL (default), nothing is written to disk.

Details

Only ok-flagged rows are plotted, and zero and missing elasticities are excluded, so the figures show the fitted distribution rather than the zeros that stand in for countries that could not be fitted. Outliers are labelled by country code using the usual 1.5 IQR rule, computed within each land use and driver.

Value

Invisibly, a list with the box and hist ggplot objects.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other output functions: [luh2_plot_bii\(\)](#), [luh2_plot_results\(\)](#), [luh2_write_tifs\(\)](#)

Examples

```
## Not run:
fit <- luh2_fit_elasticities(faoland, gdppc)
luh2_plot_elasticities(fit$results, output_dir = "outputs/plots")

## End(Not run)
```

luh2_plot_results	<i>Plot LUMEN land use results</i>
-------------------	------------------------------------

Description

Reads the GeoTIFFs written by [luh2_write_tifs\(\)](#) together with the LUMEN solution GDX in the same directory, and draws a set of maps of the solved land use for one country.

Usage

```
luh2_plot_results(output_dir, cty_shp, save_png = TRUE)
```

Arguments

output_dir	Path to the directory holding the lu_<pool>.tif files and solution_lu.gdx. Its base name is used as the IMPACT country code and in the output file names.
cty_shp	Path to the IMPACT regions shapefile, used for the country outline. Must contain a NEW_REGION field.
save_png	Logical. If TRUE (default), each figure is written to output_dir as <cty>-shares.png, <cty>-area.png, <cty>-areachange.png, <cty>-dominant.png and <cty>-dominantRGB.png.

Details

Five figures are produced, all restricted to the country whose code is `basename(output_dir)` and all comparing 2021 with 2050 (whichever of those years the solution contains):

shares Share of available land per pool, faceted by pool and year. Shares below 0.5 percent are dropped so the maps stay readable.

area The same facets showing area in 1000 ha, with the fill giving the share and transparency the area.

area change Change in area between 2021 and 2050 per pool, on a red-white-green diverging scale centred at zero.

dominant The pool with the largest share in each pixel, coloured by pool with transparency proportional to that share.

dominant RGB A share-weighted blend of the pool colours per pixel, so mixed pixels read as a mixed colour instead of being collapsed to a single winner.

Shares are computed against `avail_kha` from the solution GDX, not against the full cell area. Alaska and Hawaii are cropped out for USA so the contiguous states are not squeezed.

Value

Invisibly, a named list of ggplot objects: `shares`, `area` and `dominant`. The `area-change` and `RGB` figures are written to disk but not returned.

Requirements

Needs the `gamstransfer` package, which is shipped with GAMS rather than installed from CRAN.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other output functions: `luh2_plot_bii()`, `luh2_plot_elasticities()`, `luh2_write_tifs()`

Examples

```
## Not run:
luh2_write_tifs("outputs/CHM/solution_lu.gdx", "outputs/CHM")
luh2_plot_results("outputs/CHM", "data/IMPACT/impact_regions.shp")

## End(Not run)
```

luh2_suitability	<i>Calculate a land use suitability index from LUH2 historical data</i>
------------------	---

Description

Computes a suitability index for each land pool by averaging historical land use shares over a long time window. Pixels that have held a high share of a given land type over centuries are treated as more suitable for that land type in future projections, and the index is used to weight reallocation in [luh2_build_pixels\(\)](#) and exported to LUMEN as `suit`.

Usage

```
luh2_suitability(luh, year_start = 1715, year_end = 2015, subsample = 10)
```

Arguments

<code>luh</code>	A <code>SpatRaster</code> returned by luh2_load() .
<code>year_start</code>	Integer. Start year of the suitability window. Default 1715.
<code>year_end</code>	Integer. End year of the suitability window. Default 2015.
<code>subsample</code>	Integer. Step between the years actually read, to keep the computation tractable. Default 10 (every 10th year).

Details

Land pool definitions follow the LUH2 variable groupings:

- crop: `c3ann`, `c4ann`, `c3per`, `c4per`, `c3nfx`
- natfor: `primf`, `secdf`
- past: `pastr`, `range`
- other: `primn`, `secdn`

Within each year, variables are summed across pool members before averaging across years. `plant` and `urban` get no suitability layer: planted forest is placed by proximity instead (see [luh2_build_pixels\(\)](#)), and urban land is held fixed.

Unlike the other pixel functions, layer indices are derived from the hard-coded LUH2 historical epoch (`index = year - 849`) rather than from `terra::time(luh)`, so `year_start` and `year_end` must refer to a LUH2 file that starts in 850.

Value

A four-layer `SpatRaster`: `suit_crop`, `suit_natfor`, `suit_past`, `suit_other`. Values run from 0 to 1 and are the mean share of each pool over the sampled years.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other pixel pipeline functions: [luh2_build_pixels\(\)](#), [luh2_crop_trend\(\)](#), [luh2_export_gdx\(\)](#), [luh2_extract_year\(\)](#), [luh2_load\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2gdx\(\)](#), [wdpa_process\(\)](#)

Examples

```
## Not run:
luh <- luh2_load("data/LUH2/states.nc")
suit <- luh2_suitability(luh, year_start = 1700, year_end = 2015)
terra::plot(suit)

## End(Not run)
```

luh2_urban_area_cty	<i>Compute IMPACT-country urban area from LUH2 for a target year</i>
---------------------	--

Description

Aggregates the LUH2 urban layer to IMPACT country totals for a single year. Urban land is treated as a genuine land cover pool that is held fixed rather than reallocated, so it enters the model as a baseline stock in LAND00 with no fitted elasticity (see [luh2_export_elasticities_gdx\(\)](#)).

Usage

```
luh2_urban_area_cty(states_nc, static_nc, cty_shp, year = 2021)
```

Arguments

states_nc	Path to the LUH2 states.nc file.
static_nc	Path to the LUH2 staticData_quarterdeg.nc file.
cty_shp	Path to the IMPACT regions shapefile. Must contain a NEW_REGION field used as the country identifier.
year	Integer. Target year. Default 2021.

Value

A data frame with columns cty and value, urban area in 1000 ha.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other country pipeline functions: [luh2_bii_coeff_cty\(\)](#), [luh2_bii_coeff_cty_pixel\(\)](#), [luh2_compute_gdppc\(\)](#), [luh2_export_elasticities_gdx\(\)](#), [luh2_fit_elasticities\(\)](#), [luh2_landuse_elasticities\(\)](#), [luh2_load_faoland\(\)](#), [luh2_natfor_primary_share_cty\(\)](#), [luh2_pa_area_cty\(\)](#)

Examples

```
## Not run:
urban <- luh2_urban_area_cty(
  states_nc = "data/LUH2/states.nc",
  static_nc = "data/LUH2/staticData_quarterdeg.nc",
  cty_shp   = "data/IMPACT/impact_regions.shp",
  year      = 2021
)

## End(Not run)
```

luh2_write_tifs

*Write GeoTIFFs from a LUMEN solution GDX***Description**

Reads the solved pixel land use (lu_out) from a LUMEN solution GDX and writes one multi-layer GeoTIFF per land pool, with one layer per solution year. Files are named lu_<pool>.tif and written into output_dir.

Usage

```
luh2_write_tifs(gdx_path, output_dir)
```

Arguments

gdx_path	Path to the LUMEN solution GDX file (e.g. "outputs/CHM/solution_lu.gdx").
output_dir	Path to an existing directory where the GeoTIFFs are written. Existing files of the same name are overwritten.

Details

Rasters are built directly from the pixel coordinates stored in the GDX (pid_coords) and then resampled onto the standard 0.25 degree LUH2 grid with nearest-neighbour, which keeps the output aligned with the model inputs and avoids the NA holes that come from snapping points to a grid. One file is written for each of crop, natfor, past, other and plant; urban land is fixed in LUMEN and is not part of the solution.

Value

Invisibly, a named list of the SpatRaster objects written, one per pool, each with one layer per year named by the year.

Requirements

Needs the gamstransfer package, which is shipped with GAMS rather than installed from CRAN.

Author(s)

Abhijeet Mishra, Claude Code

See Also

[luh2_plot_results\(\)](#) and [luh2_plot_bii\(\)](#), which read the same outputs for plotting.

Other output functions: [luh2_plot_bii\(\)](#), [luh2_plot_elasticities\(\)](#), [luh2_plot_results\(\)](#)

Examples

```
## Not run:
luh2_write_tifs("outputs/CHM/solution_lu.gdx", "outputs/CHM")

## End(Not run)
```

luh2gdx

Run the full LUH2-to-GDX pixel pipeline

Description

Convenience wrapper that runs every step of the pixel pipeline in order and writes the LUMEN input GDX. Each step is also exported, so the pipeline can be run piece by piece when debugging or when only part of it needs to be redone.

Usage

```
luh2gdx(
  states_nc,
  static_nc,
  planted_tif,
  cty_shp,
  impact_gdx,
  output_gdx,
  year = 2015,
  year_start = 1990,
  year_end = 2015,
  pa_tif = NULL
)
```

Arguments

states_nc	Path to the LUH2 states.nc file.
static_nc	Path to the LUH2 staticData_quarterdeg.nc file.
planted_tif	Path to the planted forest GeoTIFF (e.g. sdpt_global.tif).
cty_shp	Path to the IMPACT regions shapefile. Must contain a NEW_REGION field used as the country identifier.
impact_gdx	Path to the IMPACT scenario GDX file, read for its LANDX0 land use results.
output_gdx	Path to write the output GDX file.
year	Integer. Target baseline year to extract from LUH2. Default 2015.
year_start	Integer. First year of the trend period. Default 1990.
year_end	Integer. Last year of the trend period. Default 2015.
pa_tif	Path to a WDPa protected area raster (km ² per pixel) produced by wdpa_process() . If NULL (default), protected area is set to zero.

Details

The steps, in order: [luh2_load\(\)](#); [luh2_pool_trend\(\)](#) three times, for crop, natural forest (primf + primn + secdf) and other land (secdn); [luh2_extract_year\(\)](#); [luh2_merge_planted\(\)](#); reading LANDX0 from impact_gdx with DOORMAT::readGDX(); [luh2_build_pixels\(\)](#); and [luh2_export_gdx\(\)](#). Progress is reported with messages as it goes.

Protected areas are not processed here. Pass an already-rasterized WDPa layer through pa_tif if protected area should be carried into LUMEN; see [wdpa_process\(\)](#) for building one.

Value

Invisibly returns the path to the written GDX file.

Requirements

Needs the gamstransfer package, which is shipped with GAMS rather than installed from CRAN.

Author(s)

Abhijeet Mishra, Claude Code

See Also

Other pixel pipeline functions: [luh2_build_pixels\(\)](#), [luh2_crop_trend\(\)](#), [luh2_export_gdx\(\)](#), [luh2_extract_year\(\)](#), [luh2_load\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2_suitability\(\)](#), [wdpa_process\(\)](#)

Examples

```
## Not run:
luh2gdx(
  states_nc = "data/LUH2/states.nc",
  static_nc = "data/LUH2/staticData_quarterdeg.nc",
  planted_tif = "data/SDPT/sdpt_global.tif",
  cty_shp = "data/IMPACT/impact_regions.shp",
  impact_gdx = "data/IMPACT/solution.gdx",
  output_gdx = "outputs/lumen_input.gdx",
  year = 2021,
  pa_tif = "data/WDPA/pa_area_km2.tif"
)

## End(Not run)
```

wdpa_process

Process WDPA protected areas to pixel or country level

Description

Loads the WDPA geodatabase, filters it to the ecologically meaningful IUCN categories and to designated terrestrial sites, and returns protected area either as km² per LUH2 pixel or as IMPACT country totals.

Usage

```
wdpa_process(
  gdb_path,
  static_nc,
  output_tif = NULL,
  year = 2021,
  output = "pixel",
  cty_shp = NULL
)
```

Arguments

<code>gdb_path</code>	Path to the WDPA geodatabase (.gdb folder).
<code>static_nc</code>	Path to the LUH2 staticData_quarterdeg.nc file, used for its ccode layer to match resolution, extent and CRS and to restrict the output to land.
<code>output_tif</code>	Path to write the output GeoTIFF. Required when output = "pixel", ignored otherwise.
<code>year</code>	Integer. Only PAs designated by this year (STATUS_YR <= year) are included. Default 2021.
<code>output</code>	Character. Either "pixel" (default) for a per-pixel PA area raster, or "cty" for a country-level summary data frame.
<code>cty_shp</code>	Path to the IMPACT regions shapefile. Required when output = "cty".

Details

Area is computed by intersecting the PA polygons with the grid cell polygons and summing the intersection areas, rather than by rasterizing with `touches = TRUE`: at 0.25 degree a cell covers roughly 700 km², so presence/absence rasterization would let a small reserve fill an entire cell. Per-cell area is then capped at the cell's own area.

Records are kept when the IUCN category is one of Ia, Ib, II, IV, V or VI, the realm is terrestrial or coastal, the status is Designated, Inscribed, Established or Adopted, and `STATUS_YR` is positive and no later than year:

- Ia - Strict Nature Reserve
- Ib - Wilderness Area
- II - National Park
- IV - Habitat/Species Management Area
- V - Protected Landscape/Seascape
- VI - Protected Area with Sustainable Use

Category III (Natural Monument) is excluded because those features are too small to resolve at 0.25 degree.

Value

If `output = "pixel"`, a single-layer `SpatRaster` of PA area in km² per pixel, also written to `output_tif`. If `output = "cty"`, a data frame with columns `cty` and `pa_area_km2`. Note the units: the country-level pipeline expects 1000 ha, so this is converted before it reaches `LANDPA00` (see [luh2_landuse_elasticities\(\)](#)), while [luh2_pa_area_cty\(\)](#) returns 1000 ha directly.

Known limitations

The polygon intersection is done in eighteen latitude bands and each band is cached as a GeoTIFF in a **hard-coded** directory, `C:/Local/Work/IFPRI/Landuse/WDPA/Processed Polygons`. The function will fail on a machine where that path does not exist, a band is skipped whenever its file is already there, and the final raster merges every `.tif` in that directory. Change year or the WDPA release and the cached bands must be deleted by hand, or the stale ones will be merged in. The WDPA layer name is likewise hard-coded (`WDPA_poly_Jul2026`) and has to be edited in the source for a different release.

With `output = "cty"`, only pixels inside the IMPACT regions shapefile are counted, so protected area in territories outside that coverage is dropped rather than reassigned. Country totals from this function are therefore lower than published global protected area figures.

Author(s)

Abhijeet Mishra, Claude Code

See Also

[luh2_pa_area_cty\(\)](#) to aggregate an existing PA raster to countries without redoing the intersection.

Other pixel pipeline functions: [luh2_build_pixels\(\)](#), [luh2_crop_trend\(\)](#), [luh2_export_gdx\(\)](#), [luh2_extract_year\(\)](#), [luh2_load\(\)](#), [luh2_merge_planted\(\)](#), [luh2_pool_trend\(\)](#), [luh2_suitability\(\)](#), [luh2gdx\(\)](#)

Examples

```
## Not run:
pa <- wdpa_process(
  gdb_path   = "data/WDPa/WDPa_Jul2026_Public.gdb",
  static_nc  = "data/LUH2/staticData_quarterdeg.nc",
  output_tif = "data/WDPa/pa_area_km2.tif",
  year      = 2021
)

## End(Not run)
```

Index

* country pipeline functions

luh2_bii_coeff_cty, 2
luh2_bii_coeff_cty_pixel, 4
luh2_compute_gdppc, 8
luh2_export_elasticities_gdx, 11
luh2_fit_elasticities, 15
luh2_landuse_elasticities, 17
luh2_load_faoland, 21
luh2_natfor_primary_share_cty, 23
luh2_pa_area_cty, 25
luh2_urban_area_cty, 32

* output functions

luh2_plot_bii, 26
luh2_plot_elasticities, 27
luh2_plot_results, 28
luh2_write_tifs, 33

* pixel pipeline functions

luh2_build_pixels, 6
luh2_crop_trend, 10
luh2_export_gdx, 13
luh2_extract_year, 14
luh2_load, 20
luh2_merge_planted, 22
luh2_pool_trend, 30
luh2_suitability, 31
luh2gdx, 34
wdpa_process, 36

luh2_bii_coeff_cty, 2, 6, 9, 12, 16, 19, 22,
24, 26, 33

luh2_bii_coeff_cty(), 4–6, 11, 18, 19, 24,
27

luh2_bii_coeff_cty_pixel, 4, 4, 9, 12, 16,
19, 22, 24, 26, 33

luh2_bii_coeff_cty_pixel(), 4, 11, 18, 19,
27

luh2_build_pixels, 6, 10, 14, 15, 20, 23, 30,
32, 35, 38

luh2_build_pixels(), 13, 23, 31, 35

luh2_compute_gdppc, 4, 6, 8, 12, 16, 19, 22,
24, 26, 33

luh2_compute_gdppc(), 15

luh2_crop_trend, 8, 10, 14, 15, 20, 23, 30,
32, 35, 38

luh2_export_elasticities_gdx, 4, 6, 9, 11,
16, 19, 22, 24, 26, 33

luh2_export_elasticities_gdx(), 3, 5, 16,
25, 32

luh2_export_gdx, 8, 10, 13, 15, 20, 23, 30,
32, 35, 38

luh2_export_gdx(), 6, 35

luh2_extract_year, 8, 10, 14, 14, 20, 23, 30,
32, 35, 38

luh2_extract_year(), 20, 22, 35

luh2_fit_elasticities, 4, 6, 9, 12, 15, 19,
22, 24, 26, 33

luh2_fit_elasticities(), 11, 19, 27

luh2_landuse_elasticities, 4, 6, 9, 12, 16,
17, 22, 24, 26, 33

luh2_landuse_elasticities(), 37

luh2_load, 8, 10, 14, 15, 20, 23, 30, 32, 35, 38

luh2_load(), 6, 10, 14, 30, 31, 35

luh2_load_faoland, 4, 6, 9, 12, 16, 19, 21,
24, 26, 33

luh2_load_faoland(), 3, 8, 9, 11, 15, 24

luh2_merge_planted, 8, 10, 14, 15, 20, 22,
30, 32, 35, 38

luh2_merge_planted(), 6, 15, 35

luh2_natfor_primary_share_cty, 4, 6, 9,
12, 16, 19, 22, 23, 26, 33

luh2_natfor_primary_share_cty(), 2

luh2_pa_area_cty, 4, 6, 9, 12, 16, 19, 22, 24,
25, 33

luh2_pa_area_cty(), 11, 19, 37, 38

luh2_plot_bii, 26, 28, 29, 34

luh2_plot_bii(), 34

luh2_plot_elasticities, 27, 27, 29, 34

luh2_plot_elasticities(), 18

luh2_plot_results, [27](#), [28](#), [28](#), [34](#)
luh2_plot_results(), [34](#)
luh2_pool_trend, [8](#), [10](#), [14](#), [15](#), [20](#), [23](#), [30](#),
[32](#), [35](#), [38](#)
luh2_pool_trend(), [6](#), [7](#), [10](#), [35](#)
luh2_suitability, [8](#), [10](#), [14](#), [15](#), [20](#), [23](#), [30](#),
[31](#), [35](#), [38](#)
luh2_suitability(), [7](#)
luh2_urban_area_cty, [4](#), [6](#), [9](#), [12](#), [16](#), [19](#), [22](#),
[24](#), [26](#), [32](#)
luh2_urban_area_cty(), [11](#), [19](#)
luh2_write_tifs, [27–29](#), [33](#)
luh2_write_tifs(), [13](#), [28](#)
luh2gdx, [8](#), [10](#), [14](#), [15](#), [20](#), [23](#), [30](#), [32](#), [34](#), [38](#)
luh2gdx(), [10](#)

stats::lm(), [10](#), [16](#)

wdpa_process, [8](#), [10](#), [14](#), [15](#), [20](#), [23](#), [30](#), [32](#),
[35](#), [36](#)
wdpa_process(), [7](#), [18](#), [19](#), [25](#), [35](#)