
SOFTWARE METAPAPER

Janus: A Python Package for Agent-Based Modeling of Land Use and Land Cover Change

K. E. Kaiser¹, A. N. Flores¹ and C. R. Vernon²

¹ Boise State University, US

² Pacific Northwest National Lab, US

Corresponding author: K. E. Kaiser (kendrakaiser@boisestate.edu)

Janus is an open source Python package for agent-based modeling (ABM) of land use and land cover change (LULCC). Many ABMs of LULCC have been created across platforms, some of which are not ideal for large scale, high resolution scenarios. This model provides a simple object-oriented framework for creating ABMs specific to LULCC. The organizational philosophy of the modeling framework is to create software objects (agents) that are associated with specific and contextual attributes which are isolated from where those agents exist in the spatial setting of the model, and still provide clear linkages between the agent, their environment, and other agents in the simulation. In this way, the framework allows for assembly of LULCC ABMs with low (programmatic) overhead, making the models extensible and providing clear mechanisms for integrating them with process-oriented biophysical models. Provided with Janus is a suite of geospatial data preprocessing tools that can use arbitrary land cover products as an input. Crop choice decisions are based on potential crop prices, these can be created synthetically, or drawn from integrated human-Earth systems models such as the Global Change Assessment Model. Janus is publicly accessible through GitHub and provides an example dataset for testing.

Keywords: Agent-based modeling; land use land cover change; agriculture; python

(1) Overview

Introduction

Agent-based models (ABMs) are a popular way to model coupled human and natural systems (CHANS) because of their flexibility in representing complexity in human decision-making and in how individuals can respond to, and exert control on, flows and stores of materials and energy represented by process-oriented models of biophysical systems. Agents can represent individuals or groups of individuals with common beliefs and values, they can interact with each other and with their environment, and learn from and adapt to changing environmental conditions [13, 5]. ABMs of land use and land cover change (LULCC) have been created to examine agriculture, conservation strategies, and urban expansion across the globe [1, 2, 15], yet many of these are challenging to use outside of the original system for which they were developed because of the location-specific nature of the data inputs and model structure.

There are many agent-based models and frameworks that have been developed for a variety of applications. These range from general-purpose ABM platforms like NetLogo, MASON, and RePast to ABMs that have been formulated and developed for very specific problems. These platforms and models span a number of programming languages

including (R, Java, C++, and Python) and some have developed entire windowed integrated development environments for creating, running, and analyzing ABMs (e.g., NetLogo, RePast). The Network for Computational Modeling in Social and Ecological Sciences (CoMSES Net, [12]) is a community portal that supports agent based modeling in social and ecological sciences, providing researchers the opportunity to contribute their ABM code, be it the entire modeling package or the subset of model scripts that require execution in a general-purpose ABM framework. A subset of ABMs have been developed and applied specifically to the problem of LULCC. Similar to the broader community of ABMs, these ABMs of LULCC range from sets of code oriented toward specific research questions and problems to more generic toolsets for modeling LULCC within a general-use ABM framework.

The need for the modeling framework reported here arises from a growing subfield of CHANS research that addresses MultiSector Dynamics (MSD). MSD research seeks to understand growing interdependencies and risks at the intersection of the energy, water, and land sector and, as such, recognizes LULCC as inextricably coupled to the dynamics of water and energy systems. As such, a goal of this model is to increase the flexibility of integrating various component models such as urban

population dynamics [14], land surface models (e.g. CLM; [8], surface and groundwater models (WRF-Hydro, [6]; ParFlow, [9]), transportation networks, infrastructure and energy development [10]. The interdependencies between these subsystems require either multi-model simulations or mechanisms for software coupling. Janus initiates this process by creating agents that can observe any number of environmental and social constraints while supporting a flexible library of decision-making options.

The model described here is intended to provide a framework for assembling ABMs of LULCC in ways that: (1) are grounded in data that characterizes both social and biophysical processes being represented, (2) can test alternative hypotheses about the role of social networking in potential spatiotemporal patterns of land use and land cover, and (3) can facilitate integration with models characterizing other sectors with which LULCC is connected, such as water supply systems, transportation, and energy distribution [11]. It was developed to facilitate the use of consistent county-scale demographic and land use (30 m) data that are available throughout the United States. Although this data is specific to the United States, any data source that permits the development of distributions of age and land status (e.g. owner, tenant) may be used. Demographics from sub-national datasets enables incorporation of local details about agents and their decision-making to be preserved in a regional scale model. The motivation of this model is to compare LULCC projections derived from global integrated human-Earth system models with ABM-derived projections that explicitly incorporate local socio-political and environmental constraints. It was developed to use projections from GCAM [3] but can be configured to take any crop profit times series and combination of land use categories.

Janus was developed in Python to facilitate use in a variety of ABM applications. The suite of preprocessing tools allows for streamlined and reproducible data inputs, while the post-processing tools allow for efficient

assessment of results (**Figure 1**). Explanation of how to run the model and details on the current decision-making process are explained in the Janus README. Although only one decision-making function is currently in place, the model allows these decision-making processes to be interchangeable. Additional attributes can be given to agents to tailor them to a specific location or set of environmental constraints.

Implementation and architecture

Janus follows a sequential workflow that includes preprocessing and the model run itself (**Figure 2**):

- Step 1: Gather input data, landcover data and a shapefile of counties in the area of interest. The county shapefile must have the county names under a field 'county'
- Step 2: GIS pre-processing (janus/preprocessing/get_gis_data.py)
- Step 3: Profit signal generation (janus/preprocessing/generate_synthetic_prices.py)
- Step 4: Declare variables (janus/example/config.yml)
- Step 5: Run Model (janus/model.py)

The Cropland Data Layer (CDL) may be downloaded using CropScape [7] (nassgeodata.gmu.edu/CropScape/) or the Aether Platform (pypi.org/project/aether/), and the county shapefiles can be found in the TIGER dataset [4]. Once these land cover and spatial extent files are placed in the data folder, use the get_gis_data.py to prepare the modeling domain and initial landcover dataset. In this step the user defines the scale of interest, the initial model year and the counties to include. A suite of GIS functions (geofxns.py) are then used to convert the CDL data to GCAM categories (or user defined categories in the "keyfile"), aggregate the data to a larger scale (1 km or 3 km), create a grid of polygons, and an extent grid. Although this preprocessing has been developed for CDL data, users could modify the code to use any land cover and land use classification product.

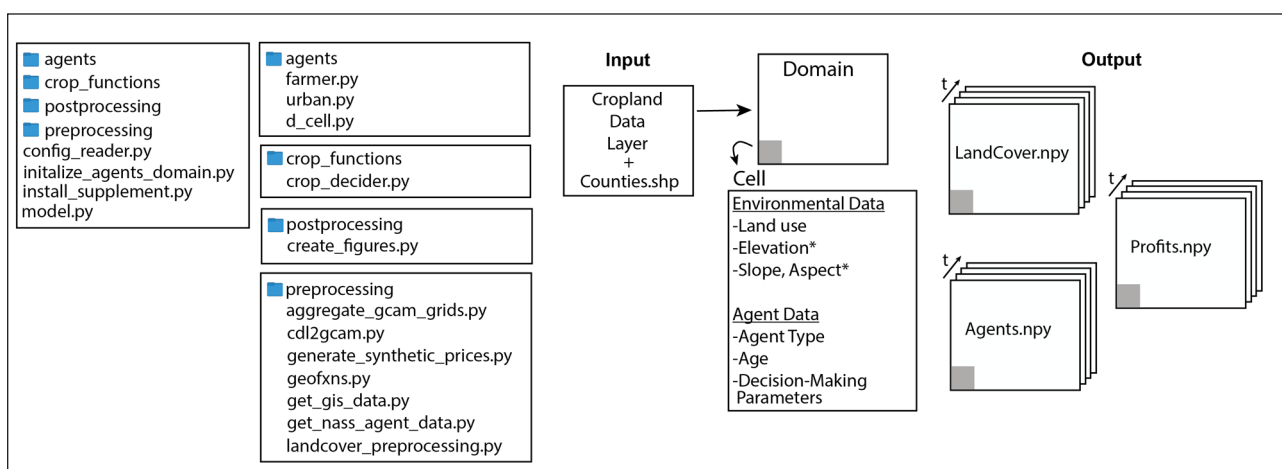


Figure 1: Architecture of the file system and data sets associated with Janus. * Layers are not included in the example dataset, but denote additional data that decision-making could be based upon in the future.

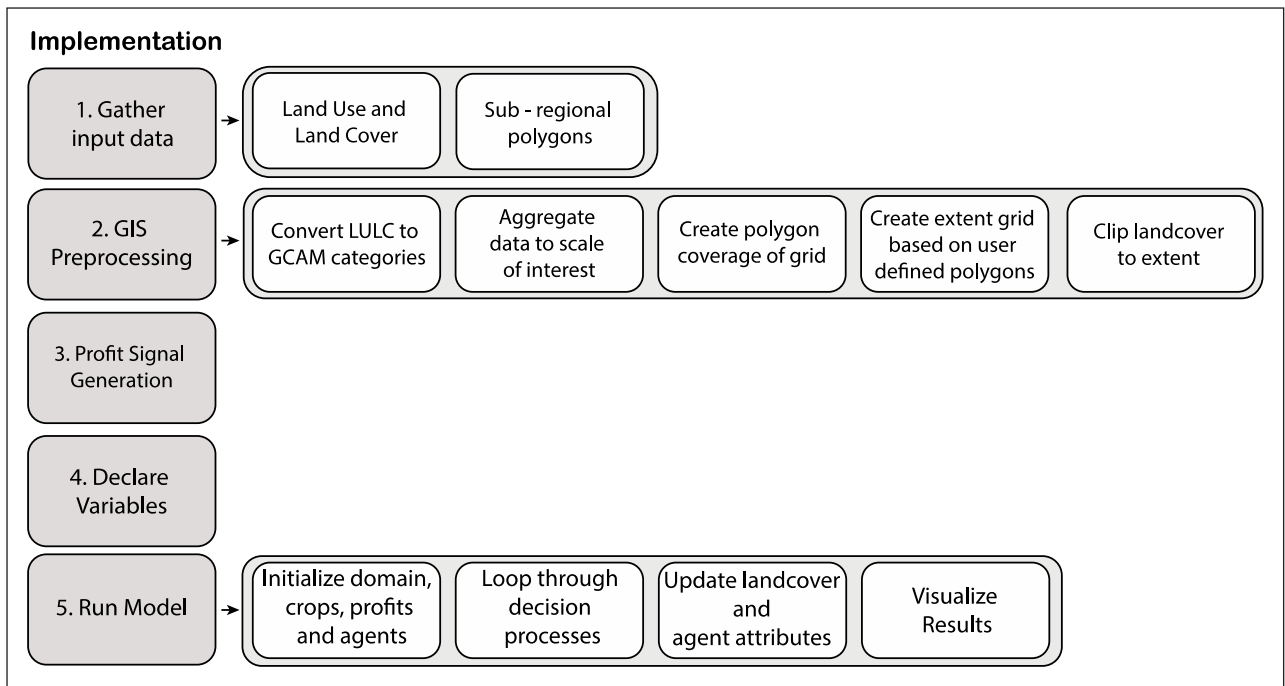


Figure 2: GIS preprocessing workflow, implementation and model workflow.

To run this model for different regions both inside and outside the U.S., the user would simply need to prepare the following:

- A shapefile containing polygon representations of their unit of interest (see https://github.com/LEAF-BoiseState/janus/tree/master/janus/tests/data/shp/countries_srb.shp).
- A CSV file providing a key from a user-specified source crop land data raster to GCAM land classes (see https://github.com/LEAF-BoiseState/janus/blob/master/janus/tests/data/CDL2GCAM_categories.csv). GCAM is a global model.
- Numpy arrays of the number of farmers in each age and tenure category if NASS data is not available (defined in `make_age_cdf` and `make_tenure_cdf` in `get_nass_agent_data.py`). These can then be entered into lines 147 and 149 in `model.py` and comment out lines 143 and 145 that call the NASS dataset.

Geospatial and demographic information are loaded and declared. Land cover data and grids may be generated or based on real data. The domain, crop choices, profit profiles and agents are initialized. Each agent type is an individual class and is populated and updated every time step. The number and type of agents used in the model can be modified by adding agents using an analogous class structure or by adding or changing attributes, the two that are currently available are Farmer and Urban agents (**Figure 3**).

Once initialized, the decision-making process is looped through each agent in each timestep. The functions that make up the decision-making process

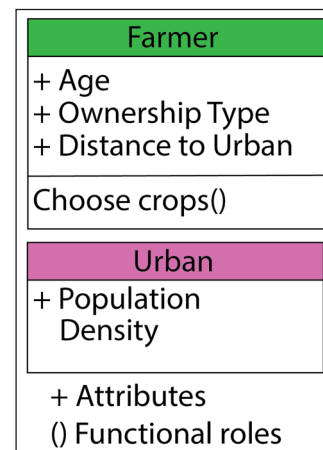


Figure 3: Class diagram of currently available agents in Janus.

are all included in the `crop_functions` directory, namely `crop_decider.py`. Within the crop decider a switching probability curve is created, and the profit comparison method the agent uses are defined (**Table 1**). After deciding whether to maintain their current crop or change to another, the land use decision is stored in the domain and agent attributes are updated accordingly. The resulting output are three 3-dimensional (time, x, y) NumPy arrays which contain the land cover, profit, and agent characteristics at each time step.

The following is the decision-making process implemented in Janus:

- LULCC decisions are made through a process by which farmer agents compare the profitability of current land use (i.e., crop choice) with potential alternatives.

- The difference between anticipated profit for alternative crop choices for the next year and the anticipated profit of the current land use for the next year are computed.
- For those alternatives for which there is an anticipated profit increase, a decision to adopt an alternative is made probabilistically.
 - Each agent is assigned behavioral characteristics that define the probability that the agent would choose the alternative as a function of the increase in profit associated with that alternative.
 - That probabilistic characteristic curve is parameterized using a beta distribution cumulative distribution function and, therefore, each agent is assigned parameters alpha and beta that describe the shape of the beta distribution.
 - The beta distribution was chosen for its relative simplicity and because it allows significant flexibility in describing agent behavior.
 - For instance, **Figure 4** shows two alternative agent behaviors given different values of the beta distribution. One agent exhibits a high probability of changing land use for a relatively modest increase in anticipated profit and the other exhibits a correspondingly low probability for switching land use at the same anticipated increase in profit.
- Based on the anticipated increase in profit associated with a particular alternative land use and the agent behavior, as defined by the parameters of the beta distribution, the value of the beta cumulative distribution function (CDF) is determined for that agent.
- The CDF value is compared to a uniform random number on the interval from 0 to 1 and, if the random number is less than the value of the CDF, the alternative crop is identified as a potential selection.
- This process of comparing anticipated profit between alternative crops and the current crops based on the associated agent characteristics captured by the beta distribution is repeated for all alternative crops for which profit is anticipated to increase.
- At the end of the comparison step, if there are no potential alternatives selected based on the stochastic decision process, the land use remains the same as the previous time step.
- If there is only one potential selection identified, then the land use will switch to that particular alternative land use.

Table 1: Description of the three available price functions and associated parameters.

Price function	Parameters	Notes
Linear ramp	Beginning price, ending price, noise variance	Generates linearly increasing, decreasing, or constant prices.
Step function	Beginning price, ending price, time fraction during simulation of steep change, noise variance	Generate step increases or step decreases
Sinusoidal function	Average price, amplitude of variation, number of periods during simulation, noise variance	Generates fluctuating prices or (using the number of periods simulated) a monotonically increasing or decreasing price

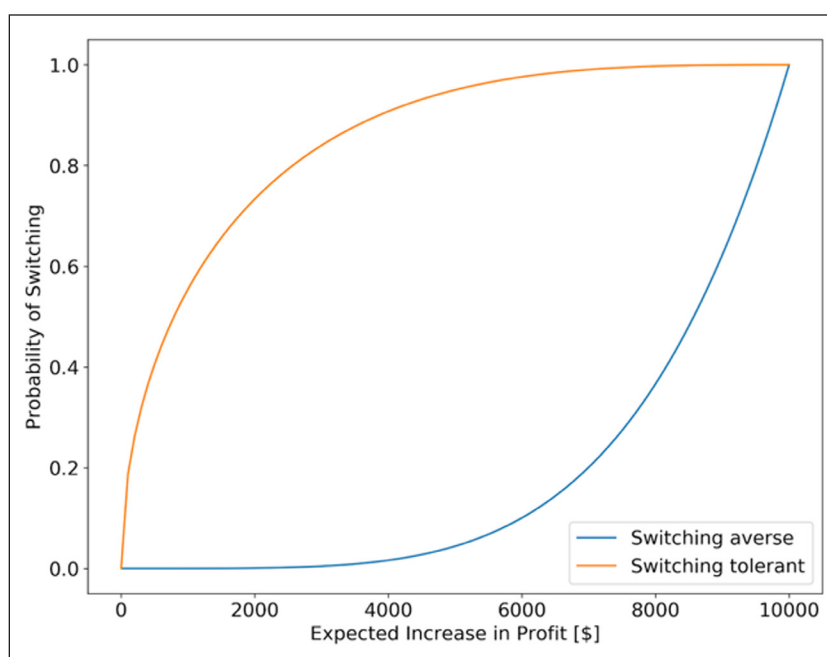


Figure 4: Example switching probability curves for two beta distributions that describe agent behavior.

- If multiple potential selections have been identified, the user can configure the model to select the alternative with the largest anticipated increase in profit or select randomly from all potential selections.

In most envisioned scientific applications, the input price signals would be derived from the output of a global integrated human-Earth systems model that is simulating global markets and associated crop prices under alternative scenarios, or from some other external data source that includes annual projections of a basket of alternative crops over some fixed time horizon. For purposes of functional testing of the model; however, we developed a simple script that can create an arbitrary number of synthetic price signals that can be used as input to the model. This synthetic price generator assumes that all synthetic price time series will be represented via either a: (1) linearly increasing or decreasing function, (2) a step increase decrease in price, or (3) a sinusoidally varying. Each of these functional forms is associated with specific parameters that the user specifies and that allow the user significant flexibility in the associated behavior of the prices that drive agent decisions.

Quality Control

Janus was developed with a robust testing suite that has been built to ensure unit performance and functional accuracy. Tests are triggered upon alteration to the remote repository through continuous integration using Travis CI (<https://travis-ci.org/>). We have also developed a suite of tests that are executed at runtime to provide informative feedback for any warnings and errors that may be raised. Users are also provided with an example data set for testing that can be installed automatically (see README). Users may raise issues on GitHub for additional support on using the software.

(2) Availability

Operating System

Mac OS X; Linux

Programming Language

Python >= 3.3

Dependencies

numpy>=1.11
 scipy>=0.18
 matplotlib>=1.3.1,<3.0
 pandas>=0.19
 geopandas>=0.5.0
 setuptools>=24.2.0
 rasterstats>=0.13.0
 gdal>=2.1
 joblib>=0.11
 rasterio>=1.0.8
 pycrs>=1.0.1
 shapely>=1.6.1
 nass>=0.1.1
 fiona>=1.7.13
 pyyaml>=3.12

Software location

Archive: Zenodo

Name: GitHub

Persistent Identifier: <https://doi.org/10.5281/zenodo.3763731>

Publisher: Kendra E. Kaiser

Version published: v1.0.1

Date published: 23 April 2020

Code repository

Name: GitHub

Identifier: <https://github.com/LEAF-BoiseState/janus/tree/v1.0.0>

License: BSD 2-Clause

Date published: 23 April 2020

Language

English

(3) Reuse

The code employs the use of docstrings throughout to ensure clarity about what each function does and alternative options that the user can declare. One of the main utilities of the model is the ability for users to automatically preprocess land cover data to larger scales and set how land categories are aggregated. We use CDL data in the example dataset, but other land cover data products could be used and aggregated with the preprocessing code with slight modifications to the code. Additional environmental data such as elevation and slope could also be added to include in decision-making (**Figure 1**).

The methods and functions in Janus are very well suited for future extension through addition of agents, agent attributes and decision-making processes. These additional agents could represent regulatory agents that constrain how and where land use changes occur. For example, in some places, agricultural land may be expanding, in order for this to occur, agents would need to be assigned to land parcels that have the potential to be developed, and those agents would need additional decision-making processes that would enable them to develop their land, or sell to an urban or agricultural agent.

Additional agent attributes that characterize how individuals might make decisions would more highly resolve local details about land use choices. Urban agents currently only have urban density as an attribute, additional attributes that characterize their values regarding agricultural land will enable additional decision-making options that include the ability for urban agents to buy agricultural land. Incorporation of social networks, and social learning strategies would lead to interesting experimental designs that lend themselves to alternative spatial patterns and decision-making outcomes.

In addition to modifying or expanding upon the existing code base, this model could be integrated with multisector models. Land use choices will be dependent on the availability of water resources, proximity to transportation for goods, and various nested scales of economic signals. The model structure is particularly suited to integrate with other gridded models.

Acknowledgements

This research was supported by the U.S. Department of Energy, Office of Science, as a part of research in MultiSector Dynamics, Earth and Environmental System Modeling Program. Pacific Northwest National Laboratory is operated by Battelle under contract DE-AC05-76RL01830.

Competing Interests

The authors have no competing interests to declare.

References

1. **Arneth, A, Brown, C and Rounsevell, M D A** 2014 Global models of human decision-making for land-based mitigation and adaptation assessment. *Nature Climate Change*, 4(7): 550–557. DOI: <https://doi.org/10.1038/nclimate2250>
2. **Blanco, V, Brown, C and Rounsevell, M** 2015 Characterizing forest owners through their objectives, attributes and management strategies. *European Journal of Forest Research*, 134(6): 1027–1041. DOI: <https://doi.org/10.1007/s10342-015-0907-x>
3. **Calvin, K**, et al. 2019 GCAM v5.1: representing the linkages between energy, water, land, climate, and economic systems. *Geosci. Model Dev.*, 12: 677–698. DOI: <https://doi.org/10.5194/gmd-12-677-2019>
4. **Data.gov** 2017 “TIGER, U.S. Current Count and Equivalent National Shapefile”. <https://catalog.data.gov/dataset/tiger-line-shapefile-2017-nation-u-s-current-county-and-equivalent-national-shapefile>.
5. **Groeneveld, J**, et al. 2017 Theoretical foundations of human decision-making in agent-based land use models – a review. *Environ. Model. Softw.*, 87: 39–48. DOI: <https://doi.org/10.1016/j.envsoft.2016.10.008>
6. **Gochis, D J, Yu, W and Yates, D N** 2015 The WRF-Hydro model technical description and user's guide, Version 3.0. *NCAR Technical document*. 120 pages. Available online http://www.ral.ucar.edu/projects/wrf_hydro/.
7. **Han, W, Yang, Z, Di, L and Mueller, R** 2012 CropScape: A Web service based application for exploring and disseminating US conterminous geospatial cropland data products for decision support. *Computers and Electronics in Agriculture*, 84: 111–123. DOI: <https://doi.org/10.1016/j.compag.2012.03.005>
8. **Lawrence, D M**, et al. 2019 The Community Land Model version 5: Description of new features, benchmarking, and impact of forcing uncertainty. *JAMES*. DOI: <https://doi.org/10.1029/2018MS001583>
9. **Maxwell, R M, Condon, L E and Kollet, S J** 2015 A high-resolution simulation of groundwater and surface water over most of the continental US with the integrated hydrologic model ParFlow v3. *Geosci. Model Dev.*, 8(3): 923–937. DOI: <https://doi.org/10.5194/gmd-8-923-2015>
10. **Moss, R H, Fisher-Vanden, K, Delgado, A, Backhaus, S, Barrett, C L, Bhaduri, B, Kraucunas, I P, Reed, P M, Rice, J S, Sue, I and Wing, C T** 2016 Understanding Dynamics and Resilience in Complex Interdependent Systems, Prospects for a Multi-Model Framework and Community of Practice. *Workshop Report, U.S. Global Change Research Program Interagency Group on Integrative Modeling*.
11. **Riahi, K**, et al. 2017 The Shared Socioeconomic Pathways and their energy, land use, and greenhouse gas emissions implications: An overview. *Glob. Environ. Chang.*, 42: 153–168. DOI: <https://doi.org/10.1016/j.gloenvcha.2016.05.009>
12. **Rollins, N D, Barton, M, Bergin, S, Janssen, M A and Lee, A** 2014 A Computational Model Library for publishing model documentation and code. *Environmental Modelling & Software*, 61: 59–64. DOI: <https://doi.org/10.1016/j.envsoft.2014.06.022>
13. **Rounsevell, M D A, Robinson, D T and Murray-Rust, D** 2012 From actors to agents in socio-ecological systems models. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 367(1586): 259–269. DOI: <https://doi.org/10.1098/rstb.2011.0187>
14. **Vahmani, P, Jones, A D and Patricola, C M** 2019 Interacting implications of climate change, population dynamics, and urban heat mitigation for future exposure to heat extremes Interacting implications of climate change, population dynamics, and urban heat mitigation for future exposure to heat extremes. *Environ. Res. Letters*, 14. DOI: <https://doi.org/10.1088/1748-9326/ab28b0>
15. **Valbuena, D, Verburg, P H and Brecht, A K** 2008 A method to define a typology for agent-based analysis in regional land-use research. *Agriculture, Ecosystems and Environment*, 128(27): 36. DOI: <https://doi.org/10.1016/j.agee.2008.04.015>


How to cite this article: Kaiser, K E, Flores, A N and Vernon, C R 2020 Janus: A Python Package for Agent-Based Modeling of Land Use and Land Cover Change. *Journal of Open Research Software*, 8: 15. DOI: <https://doi.org/10.5334/jors.306>

Submitted: 15 October 2019

Accepted: 29 May 2020

Published: 25 June 2020

Copyright: © 2020 The Author(s). This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited. See <http://creativecommons.org/licenses/by/4.0/>.

 *Journal of Open Research Software* is a peer-reviewed open access journal published by Ubiquity Press.

OPEN ACCESS 