

stoPET v1.0: A stochastic potential evapotranspiration generator for simulation of climate change impacts (stoPET v1.0 User Guide)

Dagmawi Teklu Asfaw¹, Michael Bliss Singer^{2,3,4}, Rafael Rosolem^{5,6}, David MacLeod¹, Mark Cuthbert^{2,7}, Edison Quichimbo Miguitama², Manuel F. Rios Gaona², Katerina Michaelides^{1,4,6}

¹ School of Geographical Sciences, University of Bristol, Bristol, UK

² School of Earth and Environmental Sciences, Cardiff University, Cardiff, UK

³ Water Research Institute, Cardiff University, Cardiff, UK

⁴ Earth Research Institute, University of California Santa Barbara, Santa Barbara, USA

⁵ Department of Civil Engineering, University of Bristol, UK

⁶ Cabot Institute for the Environment, University of Bristol, Bristol, UK

⁷ School of Civil and Environmental Engineering, The University of New South Wales (UNSW), Sydney, Australia

Correspondence to: Dagmawi Teklu Asfaw (d.t.asfaw@bristol.ac.uk)

1. How to install stoPET v1.0

stoPET is comprised of two Python scripts (run_stoPET.py and stoPET_v1.py) and four netCDF4 parameter files (dpetdt.nc, hpet_slope.nc, monthly_cont_percentage.nc, stopet_parameters.nc); the total size of the file is ~520 MB. To install stoPET, download the code and the necessary parameter files from [10.6084/m9.figshare.19665531](https://doi.org/10.6084/m9.figshare.19665531) into a location where you have read and write permissions.

stoPET requires a working installation of Python 2.7+ or Python 3.7+ with the following modules installed: netCDF4, numpy, scipy, (and datetime, os, sys; these three latter embedded already in Python). An environment dependency file (environment.yml) is provided with the model script.

The instructions below assume that Python can be called on the command line - i.e. that the user is using a unix or linux-based command prompt, e.g., CMD, PowerShell, BASH, ZSH... etc. The stoPET running script, i.e., “run_stoPET.py”, can be called from within a Python IDE by typing `%run run_stoPET.py`. Type `python3 run_stoPET.py` (`python run_stoPET.py` - for Python 2.7) if you are outside a Python IDE.

2. How to run stoPET v1.0

To run stoPET, make sure all the python scripts (run_stoPET.py and stoPET_v1.py) and the parameter files are in the same directory. Once all the files are in the same directory, open the run_stoPET.py script and provide the required arguments before running the script.

The user should modify the run_stoPET.py file for their environment (see below), change the directory into the directory containing the parameter files and stoPET_v1.py script and execute the following command:

python3 run_stoPET.py (or **python run_stoPET.py**) or **%run run_stoPET.py**

N.B the commands above are given as an example. The user can choose any Python IDE (IPython, PyCharm, Spider, Anaconda ...etc.) and run the script accordingly

Config file: run_stoPET.py

run_stoPET.py is the script used to run the stoPET model. This script allows the user to edit all the variables required to run the model. Not all variables are relevant to all parts of the system. Irrelevant variables are disregarded.

The variables in run_stoPET.py are as follows:

Variable	Explanation	Use
datapath	The directory that contains the parameter netCDF files and the stoPET_v1.py script.	stoPET_v1.py
outputpath	The directory where the output of the model will be placed.	stoPET_v1.py
runtype	This is a string input with two options. ‘regional’ or ‘single’ . It will tell the model whether the PET is generated for a single point or an area of a specified region (or rectangle).	stoPET_v1.py
startyear	The year from which the user wants to start the PET time series to start.	stoPET_v1.py stoPET_wrapper_singlepoint(startyear, endyear, latval, lonval, locname)[function]
endyear	The last year requested by the user for the PET time series.	stoPET_v1.py stoPET_wrapper_singlepoint(startyear, endyear, latval, lonval, locname)[function]
latval	The latitude of the single point where the user wants the PET.	stoPET_v1.py stoPET_wrapper_singlepoint(startyear, endyear, latval, lonval, locname)[function]
lonval	The longitude of the single point where the user wants the PET.	stoPET_v1.py stoPET_wrapper_singlepoint(startyear, endyear, latval, lonval, locname)[function]
locname	Any name to be given as a string which will be used in the file name of the final PET outcome of the location.	stoPET_v1.py stoPET_wrapper_singlepoint(startyear, endyear, latval, lonval, locname)[function]
latval_min	This is for running stoPET on an area. The minimum latitude of the region.	stoPET_v1.py stoPET_wrapper_regional(startyear, endyear, latval_min, latval_max, lonval_min, lonval_max, locname, ens_num)[function]
latval_max	The maximum latitude of the region.	stoPET_v1.py

		stoPET_wrapper_regional(startyear, endyear, latval_min, latval_max, lonval_min, lonval_max, locname, ens_num) ...[function]
lonval_min,	The minimum longitude of the region.	stoPET_v1.py stoPET_wrapper_regional(startyear, endyear, latval_min, latval_max, lonval_min, lonval_max, locname, ens_num) ...[function]
lonval_max	The maximum longitude of the region.	stoPET_v1.py stoPET_wrapper_regional(startyear, endyear, latval_min, latval_max, lonval_min, lonval_max, locname, ens_num) ...[function]
number_ensm	This is the number of ensembles the user wishes to run. It is kept 1 by default but if the user wants to have multiple runs the number should be given by this variable.	stoPET_v1.py stoPET_wrapper_regional(startyear, endyear, latval_min, latval_max, lonval_min, lonval_max, locname, ens_num) ...[function]
tempAdj	This is an integer number with values 1, 2, or 3. Each of these numbers represents what method to use for the model to account for temperature adjustment on future PET. Method 1 = 1, Method 2 = 2, Method 3 = 3 Refer to the paper for the description of each method.	
deltat	This variable represents the user-defined temperature increase expected in the area of interest. The value will only be used if Method 2 (tempAdj = 2) is selected as the temperature adjustment technique. Otherwise, this value will be ignored by the model.	
udpi_pet	This variable is the user-defined percentage increase of PET. The value should be from 0 to 100. This will be used to adjust the estimated PET by the user-provided percentage if Method 1 (tempAdj = 1). Otherwise, this value will be ignored by the model.	

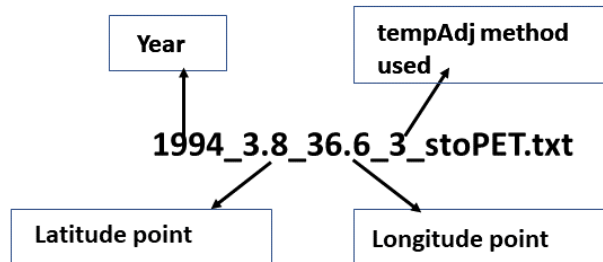
PET estimation file: stoPET_v1.py

stoPET_v1.py is the script that contains the required functions to estimate the PET for a single point or a defined region. The script comprises eight functions well described within the script itself. The two core functions used to run stoPET and generate the PET are **stoPET_wrapper_singlepoint** (***) and **stoPET_wrapper_regional** (***). The first one is used to generate PET at a single point, whereas the second one is used to generate a regional PET.

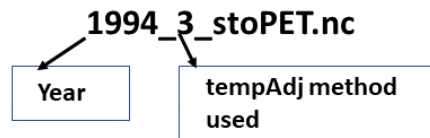
The output of these functions will be written in the output directory provided, where the model creates a new named directory (<outputpath>+<locname>+'_E'+ <number_ensm> +'_stoPET/').

For the single point run, stoPET generates text files (<year>_<latval>+'_'<lonval>+'_'<tempAdj>+stoPET.txt and (<year>_<latval>+'_'<lonval>+'_'<tempAdj>+AdjstoPET.txt). The first file is the PET generated without accounting for any adjustment for temperature. The second file is the adjusted PET based on the user's choice (tempAdj). If one just wants to avoid any temperature change adjustment in the model, just use the first file output and ignore the second file.

An example of a file name output from a single point run for year=1994, latval=3.8, lonval=36.6 and tempAdj=3 is given below.



For a regional run, the file will be stored within the output directory folder where the model creates a new folder with a name <outputpath>+<locname>+'_E'+ <number_ensm> +'_stoPET/'. The file names are based on the year values specified by the user (startyear, endyear). The file name is <year>+'_'<tempAdj>+'_stoPET.nc for the files without any temperature adjustment, and <year>+'_'<tempAdj>+'_AdjstoPET.nc for the PET generated accounting for temperature adjustment. An example of a file output is given below.



Each year's file will have a four dimensional array (days, hours, latitude, longitude). The variable name for the stochastically generated PET is 'pet' within the netCDF files.