# template\_params.py

import numpy as np

import pandas as pd

import os

# set up the landscape

b4 = np.vstack([np.linspace(1, 0, 50) for \_ in range(50)])

af = np.vstack([np.linspace(1, 0.5, 50) for \_ in range(50)])

stable = np.vstack([np.linspace(1, 0, 50) for \_ in range(50)])

K = np.ones((50,50))

**# NOTE: time\_steps.CSV USED TO SET CLIMATE CHANGE**

**# TO START AT TIME STEP 2500**

**# AND FINISH AT TIME STEP 2750**

if os.getcwd().split('/')[1] == 'home':

steps = pd.read\_csv(('/home/deth/Desktop/CAL/research/projects/sim/'

'ch2/climate\_change\_adaptation\_and\_genomic\_arch/sim/'

'time\_steps.CSV'))

else:

steps = pd.read\_csv(('/global/scratch/users/drewhart/'

'ch2/climate\_change\_adaptation\_and\_genomic\_arch/sim/'

'time\_steps.CSV'))

# set time when environmental change begins

change\_T = int(steps[steps['name']=='start']['num'].values[0])

# set time when environmental change ends

T = int(steps[steps['name']=='end']['num'].values[0])

# show the landscape, for debugging, if requested

debug\_landscape = False

if debug\_landscape:

import matplotlib.pyplot as plt

fig = plt.figure()

ax1 = fig.add\_subplot(221)

im1 = ax1.imshow(b4, cmap='spring', vmin=0, vmax=1)

plt.colorbar(im1)

ax3 = fig.add\_subplot(223)

im3 = ax3.imshow(af, cmap='spring', vmin=0, vmax=1)

plt.colorbar(im3)

ax2 = fig.add\_subplot(222)

im2 = ax2.imshow(stable, cmap='winter', vmin=0, vmax=1)

plt.colorbar(im2)

ax4 = fig.add\_subplot(224)

im4 = ax4.imshow(K, cmap='autumn', vmin=0, vmax=1)

plt.colorbar(im4)

plt.show()

# This is a parameters file generated by Geonomics

# (by the gnx.make\_parameters\_file() function).

# :::::: ::: :: :::::::::::#

#:::::: :::: ::: :: :: :: ::::::::::: ::#

#::::::::: :: :: :::::::::::::::::::::::::#

#:::::::::: :::::::::: :::::: :::::::: ::#

# : :::: :: :::: : :: :::::::: : :: : #

# GGGGG :EEEE: OOOOO NN NN OOOOO MM MM IIIIII CCCCC SSSSS #

# GG EE OO OO NNN NN OO OO MM MM II CC SS #

# GG EE OO OO NN N NN OO OO MMM MMM II CC SSSSSS #

# GG GGG EEEE OO OO NN NNN OO OO MM M MM II CC SS #

# GG G EE OO OO NN NN OO OO MM MM II CC SSS #

# GGGGG :EEEE: OOOOO NN NN OOOOO MM MM IIIIII CCCCC SSSSS #

# : :::::::: :::::::::: :: :: : : #

#: ::::: :::::: ::: ::::::: #

# ::: ::::: :: ::::: #

# :: :::: #

# #

# :: :: ::: #

params = {

###############################################################################

###################

#### LANDSCAPE ####

###################

'landscape': {

##############

#### main ####

##############

'main': {

#x,y (a.k.a. j,i) dimensions of the Landscape

'dim': (50,50),

#x,y resolution of the Landscape

'res': (1,1),

#x,y coords of upper-left corner of the Landscape

'ulc': (0,0),

#projection of the Landscape

'prj': None,

}, # <END> 'main'

################

#### layers ####

################

'layers': {

#layer name (LAYER NAMES MUST BE UNIQUE!)

'shift': {

#-------------------------------------#

#--- layer num. 0: init parameters ---#

#-------------------------------------#

#initiating parameters for this layer

'init': {

#parameters for a 'defined'-type Layer

'defined': {

#raster to use for the Layer

'rast': b4,

#point coordinates

'pts': None,

#point values

'vals': None,

#interpolation method {None, 'linear', 'cubic',

#'nearest'}

'interp\_method': None,

}, # <END> 'defined'

}, # <END> 'init'

#---------------------------------------#

#--- layer num. 0: change parameters ---#

#---------------------------------------#

#landscape-change events for this Layer

'change': {

0: {

#array or file for final raster of event, or directory

#of files for each stepwise change in event

'change\_rast': af,

#starting timestep of event

'start\_t': change\_T,

#ending timestep of event

'end\_t': T,

#number of stepwise changes in event

'n\_steps': T-change\_T,

}, # <END> event 0

}, # <END> 'change'

}, # <END> layer num. 0

#layer name (LAYER NAMES MUST BE UNIQUE!)

'stable': {

#-------------------------------------#

#--- layer num. 1: init parameters ---#

#-------------------------------------#

#initiating parameters for this layer

'init': {

#parameters for a 'defined'-type Layer

'defined': {

#raster to use for the Layer

'rast': stable,

#point coordinates

'pts': None,

#point values

'vals': None,

#interpolation method {None, 'linear', 'cubic',

#'nearest'}

'interp\_method': None,

}, # <END> 'defined'

}, # <END> 'init'

}, # <END> layer num. 1

#layer name (LAYER NAMES MUST BE UNIQUE!)

'K': {

#-------------------------------------#

#--- layer num. 2: init parameters ---#

#-------------------------------------#

#initiating parameters for this layer

'init': {

#parameters for a 'defined'-type Layer

'defined': {

#raster to use for the Layer

'rast': K,

#point coordinates

'pts': None,

#point values

'vals': None,

#interpolation method {None, 'linear', 'cubic',

#'nearest'}

'interp\_method': None,

}, # <END> 'defined'

}, # <END> 'init'

}, # <END> layer num. 2

#layer name (LAYER NAMES MUST BE UNIQUE!)

'move': {

#-------------------------------------#

#--- layer num. 2: init parameters ---#

#-------------------------------------#

#initiating parameters for this layer

'init': {

#parameters for a 'defined'-type Layer

'defined': {

#raster to use for the Layer

'rast': np.ones((50,50)),

#point coordinates

'pts': None,

#point values

'vals': None,

#interpolation method {None, 'linear', 'cubic',

#'nearest'}

'interp\_method': None,

}, # <END> 'defined'

}, # <END> 'init'

}

#### NOTE: Individual Layers' sections can be copy-and-pasted (and

#### assigned distinct keys and names), to create additional Layers.

} # <END> 'layers'

}, # <END> 'landscape'

###############################################################################

###################

#### COMMUNITY ####

###################

'comm': {

'species': {

#species name (SPECIES NAMES MUST BE UNIQUE!)

'spp\_0': {

#-----------------------------------#

#--- spp num. 0: init parameters ---#

#-----------------------------------#

'init': {

#starting number of individs

'N': 1000,

#carrying-capacity Layer name

'K\_layer': 'K',

#multiplicative factor for carrying-capacity layer

'K\_factor': 2.5,

}, # <END> 'init'

#-------------------------------------#

#--- spp num. 0: mating parameters ---#

#-------------------------------------#

'mating' : {

#age(s) at sexual maturity (if tuple, female first)

'repro\_age': 0,

#whether to assign sexes

'sex': False,

#ratio of males to females

'sex\_ratio': 1/1,

#whether P(birth) should be weighted by parental dist

'dist\_weighted\_birth': False,

#intrinsic growth rate

'R': 0.5,

#intrinsic birth rate (MUST BE 0<=b<=1)

'b': 0.5,

#expectation of distr of n offspring per mating pair

'n\_births\_distr\_lambda': 1,

#whether n births should be fixed at n\_births\_dist\_lambda

'n\_births\_fixed': True,

#radius of mate-search area

'mating\_radius': 5,

#whether individs should choose nearest neighs as mates

'choose\_nearest\_mate': False,

#whether mate-choice should be inverse distance-weighted

'inverse\_dist\_mating': False,

}, # <END> 'mating'

#----------------------------------------#

#--- spp num. 0: mortality parameters ---#

#----------------------------------------#

'mortality' : {

#maximum age

'max\_age': None,

#min P(death) (MUST BE 0<=d\_min<=1)

'd\_min': 0,

#max P(death) (MUST BE 0<=d\_max<=1)

'd\_max': 1,

#width of window used to estimate local pop density

'density\_grid\_window\_width': None,

}, # <END> 'mortality'

#---------------------------------------#

#--- spp num. 0: movement parameters ---#

#---------------------------------------#

'movement': {

#whether or not the species is mobile

'move': True,

#mode of distr of movement direction

'direction\_distr\_mu': 0,

#concentration of distr of movement direction

'direction\_distr\_kappa': 0,

#mean of distr of movement distance

'movement\_distance\_distr\_param1': 0.25,

#variance of distr of movement distance

'movement\_distance\_distr\_param2': 0.5,

#movement distance distr to use ('lognormal','levy','wald')

'movement\_distance\_distr': 'wald',

#mean of distr of dispersal distance

'dispersal\_distance\_distr\_param1': 0.5,

#variance of distr of dispersal distance

'dispersal\_distance\_distr\_param2': 0.5,

#dispersal distance distr to use ('lognormal','levy','wald')

'dispersal\_distance\_distr': 'wald',

}, # <END> 'movement'

#---------------------------------------------------#

#--- spp num. 0: genomic architecture parameters ---#

#---------------------------------------------------#

'gen\_arch': {

#file defining custom genomic arch

'gen\_arch\_file': None,

#num of loci

'L': 1000,

#value to use for fixed starting allele freqs (None to draw)

'start\_p\_fixed': 0.5,

#whether to start neutral locus freqs at 0

'start\_neut\_zero': True,

#genome-wide per-base neutral mut rate (0 to disable)

'mu\_neut': 0,

#genome-wide per-base deleterious mut rate (0 to disable)

'mu\_delet': 0,

#shape of distr of deleterious effect sizes

'delet\_alpha\_distr\_shape': 0.2,

#scale of distr of deleterious effect sizes

'delet\_alpha\_distr\_scale': 0.2,

**#NOTE: MAIN SCRIPT OVERRIDES THE FOLLOWING TWO PARAMS**

**#TO SET RECOMBINATION RATES TO A FIXED VALUE OF**

**#0.5, 0.05, OR 0.005 FOR INDEPENDENT,**

**#WEAK, OR STRONG LINKAGE VALUES**

#alpha of distr of recomb rates

'r\_distr\_alpha': 1000,

#beta of distr of recomb rates

'r\_distr\_beta': 1e3,

#whether loci should be dominant (for allele '1')

'dom': False,

#whether to allow pleiotropy

'pleiotropy': False,

#custom fn for drawing recomb rates

'recomb\_rate\_custom\_fn': None,

#number of recomb paths to hold in memory

'n\_recomb\_paths\_mem': int(1e4),

#total number of recomb paths to simulate

'n\_recomb\_paths\_tot': int(1e5),

#num of crossing-over events (i.e. recombs) to simulate

'n\_recomb\_sims': 100\_000,

#whether to generate recombination paths at each timestep

'allow\_ad\_hoc\_recomb': False,

#whether to save mutation logs

'mut\_log': False,

#whether to jitter recomb bps, to correctly track num\_trees

'jitter\_breakpoints': False,

#whether to use tskit (to record full spatial pedigree)

'use\_tskit': True,

#time step interval for simplication of tskit tables

'tskit\_simp\_interval': 100,

'traits': {

#-------------------------#

#---trait 0 parameters ---#

#-------------------------#

#trait name (TRAIT NAMES MUST BE UNIQUE!)

'trait\_0': {

#trait-selection Layer name

'layer': 'shift',

#polygenic selection coefficient

'phi': 1,

**#NOTE: MAIN SCRIPT CHANGES NEXT PARAM TO 4, 20, OR 100**

**#FOR LOW-REDUNDANCY SCENARIOS OF DIFF. POLYGENICITY,**

**#OR 8, 40, OR 200 FOR HIGH-REDUNDANCY SCENARIOS**

#number of loci underlying trait

'n\_loci': 50,

#mutation rate at loci underlying trait

'mu': 0,

#mean of distr of effect sizes

'alpha\_distr\_mu' : 0,

#variance of distr of effect size

'alpha\_distr\_sigma': 0,

#max allowed magnitude for an alpha value

'max\_alpha\_mag': None,

#curvature of fitness function

'gamma': 1,

#whether the trait is universally advantageous

'univ\_adv': False

}, # <END> trait 0

#-------------------------#

#---trait 1 parameters ---#

#-------------------------#

#trait name (TRAIT NAMES MUST BE UNIQUE!)

'trait\_1': {

#trait-selection Layer name

'layer': 'stable',

#polygenic selection coefficient

'phi': 1,

**#NOTE: MAIN SCRIPT CHANGES NEXT PARAM TO 4, 20, OR 100**

**#FOR LOW-REDUNDANCY SCENARIOS OF DIFF. POLYGENICITY,**

**#OR 8, 40, OR 200 FOR HIGH-REDUNDANCY SCENARIOS**

#number of loci underlying trait

'n\_loci': 50,

#mutation rate at loci underlying trait

'mu': 0,

#mean of distr of effect sizes

'alpha\_distr\_mu' : 0,

#variance of distr of effect size

'alpha\_distr\_sigma': 0,

#max allowed magnitude for an alpha value

'max\_alpha\_mag': None,

#curvature of fitness function

'gamma': 1,

#whether the trait is universally advantageous

'univ\_adv': False

}, # <END> trait 1

#### NOTE: Individual Traits' sections can be copy-and-pasted (and

#### assigned distinct keys and names), to create additional Traits.

}, # <END> 'traits'

}, # <END> 'gen\_arch'

}, # <END> spp num. 0

#### NOTE: individual Species' sections can be copy-and-pasted (and

#### assigned distinct keys and names), to create additional Species.

}, # <END> 'species'

}, # <END> 'comm'

###############################################################################

###############

#### MODEL ####

###############

'model': {

**# NOTE: NEXT PARAM OVERRIDDEN BY MAIN SCRIPT**

#total Model runtime (in timesteps)

'T': 100000,

#min burn-in runtime (in timesteps)

'burn\_T': 30,

#seed number

'num': None,

###############################

#### iterations parameters ####

###############################

'its': {

#num iterations

'n\_its': 1,

#whether to randomize Landscape each iteration

'rand\_landscape': False,

#whether to randomize Community each iteration

'rand\_comm': False,

#whether to randomize GenomicArchitectures each iteration

'rand\_genarch': True,

#whether to burn in each iteration

'repeat\_burn': False,

}, # <END> 'iterations'

####################################

#### data-collection parameters ####

####################################

'data': {

'sampling': {

#sampling scheme {'all', 'random', 'point', 'transect'}

'scheme': 'all',

#sample size at each point, for point & transect sampling

'n': 1000,

#coords of collection points, for point sampling

'points': None,

#coords of transect endpoints, for transect sampling

'transect\_endpoints': None,

#num points along transect, for transect sampling

'n\_transect\_points': None,

#collection radius around points, for point & transect sampling

'radius': None,

#when to collect data

'when': [change\_T-1,

int((change\_T-1+T-1)/2),

T-1],

#whether to save current Layers when data is collected

'include\_landscape': False,

#whether to include fixed loci in VCF files

'include\_fixed\_sites': True,

},

'format': {

#format for genetic data {'vcf', 'fasta'}

'gen\_format': ['vcf'],

#format for vector geodata {'csv', 'shapefile', 'geojson'}

'geo\_vect\_format': 'csv',

#format for raster geodata {'geotiff', 'txt'}

'geo\_rast\_format': 'geotiff',

#format for files containing non-neutral loci

'nonneut\_loc\_format': 'csv',

},

}, #<END> 'data'

} # <END> 'model'

} # <END> params