ALB Spread Simulation

Lora Murphy

The ALB spread simulation takes a set of trees with an initial ALB infestation, and models insect spread for a desired length of time.

Required setup

The APHIS simulation is performed in R, and takes advantage of R's spatial analysis capabilities. (More on R and spatial data at https://cran.r-project.org/view=Spatial.)

To get set up: 1. Install the latest version of R. (https://cran.r-project.org/) 2. If you are working on a computer running Windows, install RTools, which can be found under the R Binaries section of the website above. Follow the directions to make sure R and RTools can find each other. 3. Install the additional packages "sp" and "sf". Be sure to use dependencies=TRUE when you do: "install.packages(c("sp","sf"), dependencies=TRUE)" 4. Additionally, the packages "dplyr", "raster" and "magick" may be required for certain types of output.

How to run a simulation

Open the script called "Simulation.R". Adjust the settings you see there as desired, and run in R. The other scripts do the heavy lifting. They will take their settings from "Simulation.R" and you shouldn't need to adjust them to run the model.

If you make an adjustment to one of the scripts in an area besides the settings, let Lora know so it can be included in future iterations.

Input

The input is a dataframe of trees that are potential ALB victims. All are assumed to be Acers. The dataframe should have the following fields, with these exact case-sensitive names (there can be other fields, they'll be ignored):

- "x", "y": location coordinates. Point of origin doesn't matter, this will only be used to calculate distances between points. SPECIFY THE LINEAR UNITS BELOW (feet, meters).
- "mean_noforestdist": the mean distance to the forest edges in each of the 8 cardinal and intercardinal directions, in m. If the tree isn't in a forest landcover, this is 0.
- "dbh": tree's DBH in cm

OPTIONAL: - "infested": integer, 0 being uninfested, 1 being infested. If this field is included and there is at least 1 tree with a 1 status, the trees marked as 1 will be the seed of the outbreak. If either this field is not included or all the values are 0, a select number of trees will be randomly assigned to be the seed of the outbreak (number settable below).

The "mean_noforestdist" field can be calculated with the script "Get non forest landcover in 8 directions.R."

Output

A dataframe that is a copy of the input dataframe, with additional fields for dates of infestation and removal ("year_infested" and "year_removed").

Risk model: determining which trees are infested

Uninfested trees have a certain probability of being infested each timestep, as a function of number of and distance to post-emergence infested source trees and various tree characteristics (see below).

Source trees are trees which are infested, post-emergence, and have not been removed. A tree which is infested is pre-emergence until the lag period has passed (see biology settings of the model). Pre-emergence trees do not infest other trees. The lag period could be set to 0 to remove the pre- and post-emergence distinction.

A tree's risk is calculated according to the assigned risk model (currently only Long Island's model ("LI") is supported).

LI risk model

The Long Island model assesses tree risk as a function of source pressure, DBH of tree, number of nearby Acer neighbors, and landcover.

The probability that a tree will be infested is:

$$prob = \mu * source.term * dbh.term * density.term * distance.term$$

where μ is a parameter representing the upper limit of infestation probability.

$$source.term = \frac{1}{1 + \left(\dfrac{sp}{\alpha}\right)^{\gamma}}$$

where sp is a measure of source pressure and α and γ are parameters.

$$sp = \sum_{i=1}^{N} \beta * e^{(dsn*(distance_i^{\delta}))}$$

for N source trees within the maximum distance of 5280 feet, where dsn, β, and δ are parameters and $distance_i$ is the distance from the source tree to the ith neighbor.

$$dbh.term = \frac{1}{1 + \left(\dfrac{DBH}{b_1}\right)^{b_2}}$$

where DBH is tree's DBH, in cm, and b_1 and b_2 are parameters.

$$density.term = \frac{1}{1 + \left(\dfrac{N_{acer}}{c_1}\right)^{c_2}}$$

where N_{acer} is the number of Acer neighbors within 30 m of the target (will be calculated and updated by the script) and c_1 and c_2 are parameters.

$$distance.term = \frac{1}{1 + \left(\dfrac{dist_{lc}}{e_1}\right)^{e_2}}$$

where $dist_lc$ is the mean distance to no-forest landcover and e_1 and e_2 are parameters.

Tree growth

In the case of a long simulation, growth may change a tree's risk over time. This script can grow trees using a linear function of DBH:

$$DBH_{t+1} = DBH_t + (m * DBH_t + b)$$

where this year's DBH is DBH_{t+1}, last year's DBH is DBH_t, m is the slope of growth, and b is the intercept. Either m or b can be set to 0.

Model flow

1. Calculate this year's probability of infestation for potential hosts, based on the chosen risk model.
2. Choose the trees that get infested this year. For each tree, compare a random number to its probability to make the choice.
3. For those trees that get infested, record the year they get infested to both track time until emergence, and generally track spread in output.
4. Apply management. As parameters, 4 different probabilities are defined for detection and removal: post-emergence tree in surveyed area, pre-emergence tree in surveyed area, post-emergence tree NOT in surveyed area, pre-emergence tree NOT in surveyed area.
5. Decide what's being surveyed: identify trees that were detected in the previous year. Define a circle of radius 1.5 miles around each detected tree: all areas so marked are surveyed areas. (This is limited to 500 trees, for computer memory reasons.) If no trees were detected in the previous year, no surveys will be done.
6. Identify all post-emergence trees within surveyed areas, and use a random number against the probability of detection of these trees to choose which are detected and removed.
7. If the removal budget has not been met, identify all pre-emergence infested trees within surveyed areas. Use a random number on each, compared to the detection probability, to choose which got detected and removed.
8. Repeat steps b and c, with non-surveyed areas and post- and pre-emergence trees. All detected trees will be removed, up to the budget.
9. Apply tree growth to all live trees.

```
library(sp)
library(rgeos)


# Location of the source R script files "Sim_master.R" and "LI_model.R". These
# don't have to be in the same place as the main script, so you can set up
# your workspace the way you wish.
source_directory <- ""
```

```r
# Where to put output
output_directory <- ""
# Output name. This will be used as the root for various files that may be
# produced
output_root <- "testrun"


#-------------------------------------------------------------------------------#
# Load input tree file according to desired method; let it be called
# "tree_dat". If it is from a workspace and already named something else,
# you can rename it here.
#-------------------------------------------------------------------------------#
data_directory <- "C:/users/lora/documents/Projects/APHIS2/Simulation"
setwd(data_directory)
tree_dat <- readRDS("LI_trees.rds")
#-------------------------------------------------------------------------------#



#-------------------------------------------------------------------------------#
# Run settings:
#-------------------------------------------------------------------------------#
# Year to start simulation at - this will count from here. It doesn't matter
# what it is as long as it's an integer.
start_year <- 1

# How long, in years, to run simulation
sim_length <- 10



# Linear units for coordinates: one of "feet" or "meters"
linear_unit <- "feet"

#-------------------------------------------------------------------------------#
# Biology settings:
#-------------------------------------------------------------------------------#
# Lag period between oviposition and emergence, in years
lag <- 1

# Model desired: Currently only "LI" supported
model <- "LI"

# Tree growth slope:
growth_slope <- 0.01
# Tree growth intercept:
growth_intercept <- 0



#-------------------------------------------------------------------------------#
# Management settings:
#-------------------------------------------------------------------------------#
# Probability that a post-emergence tree in a unit being surveyed is detected
# and removed
prob_surv_post_detect <- 0.9
```

```r
# Probability that a pre-emergence tree in a unit being surveyed is detected
# and removed
prob_surv_prem_detect <- 0.7

# Probability that a post-emergence tree in a unit NOT being surveyed is
# detected. This is also what will be used to find the initial tree in an
# infestation
prob_nosurv_post_detect <- 0.05

# Probability that a pre-emergence tree NOT in a survey unit will be detected.
# This is also what will be used to find the initial tree in an infestation.
prob_nosurv_prem_detect <- 0

# Budget. The max number of trees that the budget will allow to be removed
# per year. This is the simplistic beginning, assuming all trees cost the same.
# This could be elaborated into a cost function and total budget if warranted.
max_trees_removed_per_year <- 100000000

# Year to begin any management. Allows for a spin-up period. Management is not
# guaranteed to start this year - depending on probabilities, the
# infestation may yet go unnoticed for a while longer.
year_to_begin_management <- 1

# Number of trees to randomly select for infestation to start the outbreak
# in the first year. THIS IS IGNORED IF THERE ARE ANY TREES ALREADY INFESTED
# IN THE INPUT DATASET.
num_initial_outbreak <- 0
#---------------------------------------------------------------------------#
```