scfmDriver

Module Author

09 March 2018

This module is not meant to run in isolation, so no example code chunk is provided. A minimal "standalone" example is provided in the file **scfmDemo.Rmd** which is associated with the **scfm** modules.

A fire regime is a "quantitative description of the fires characteristics of a given place and time" *(add citation)*. A [landscape] fire model can be regarded as a toolkit for simulating fires on a given or a theoretical landscape. There are many different fire model toolkits, and at least as many ways of describing a fire regime. The problem is how to bridge the descriptions and the toolkits so that a given tool kit may simulate a given description.

This module illustrates a solution to this problem. The solution is a fire regime *driver module*. It takes as input a given representation of a fire regime in terms of paramaters such as mean fire size or ignition rate. It outputs another set of parameters for a given fire model, such that the input regime is simulated as fully as the fire model's capacity permits. The driver has no access to the raw data that may have been used to estimate the fire regime paramaters, only to the paramaters as provided. Similarly, it has no access to the state of the fire model's component modules. It can only interact with them by maintaining a list of their parameters, in this case **scfmPars**. It is up to the modules to consult this list at appropriate times, normally as part of their **init** events.

A spcialised driver is required for every unique regime::toolkit pair. The present driver module **scfmDriver** drives the toolkit documented in **threeStageFire.Rmd**. The fire regime is described in terms of fire frequency and a size distribution. Frequency is modelled as homogeneous spatial Poisson process. Size distrbution is partitioned into two segments $<$ cell size and $\geq$ cellsize. The first segment is modelled as the probability of the latter event, of being larger than a single cell size. This is sometimes referred to as the "escape probability" especially in cases where a cell size of the same order as fire size control objectives of fire management agencies. The second component is modelled by a shifted truncated Pareto distribution, in terms of the expected and maximum fire size. The full paramater set is defined in section **Input data**, below.

Plotting

Nothing is plotted.

Saving

Nothing is saved

Input data

1. **scfmRegime** , a list of fire regime paramaters to emulate

- **ignitionRate** , rate parameter for a Poisson model in units of fires ha¯1year¯1
- **pEscape** , the probability that a fire will grow larger than the size of a single cell
- **xBar** mean or expected fire size (ha)
- **emfs** estimated maximum fire size (ha) based on shifted truncated Pareto it to data
- **xMax** maximum observed fire size in case **emfs** is non-credible

2. **landscapeAttr** relevent features of the landscape on which simulations will execute

- **cellSize** : cell size in ha (from input raster);
- **nFlammable** : number of flammable cells;,
- **burnyArea** : product of the two preceding quantities;
- **nNbrs** : frequency table of the counts of flammable neighbours for flammable cells $\{0, 1, \ldots, 8\}$

3. **cTable2** a table of calibration data linking percolation spread probability to mean fire size.

Input item **cTable2** is loaded from the modules **data** directory by the **.inputObects** event.

Output data

The parameters for a three stage stochastic landscape fire mode based on Bernoulli ignitions and percolation fire spread, such that running with these parameters will emulate the fire regime characteristcs specified in the inputs.

The list **scfmPars**, containing:

- **pIgnition** the parameter to scfmIgnition
- **naiveP0** non-landscape specific version of **p0** provided for experimental access only
- **p0** the initial spread parameter for module scfmEscape
- **pSpread** a parameter to scfmSpread and spread()
- **maxBurnCells** a parameter to scfmSpread to limit size in spread().

Anticipated linkages to other modules

scfmRegime is expected to be provided by the eponymous module, with parameters estimated or caclulated from empirical data for the study region.

landscapeAttr is a list that is expected to be provided by module **scfmLandcoverInit**, calculated from the input raster.

The intended use of the ouput object **scfmPars** is illustrated in **threeStageFire.Rmd**

Theory

pIgnition

The input parameter **ignitionRate** is the parameter of a spatial Poisson process. It is the expected number of fires to arrive per ha$^-1$year$^-1$. Given the initial assumptions, it follows from the additive property of Poisson distributions, that the number of fires to arrive per cell per year is also a Poisson random variable with rate paramater $\lambda = \mathsf{ignitionRate} \cdot \mathsf{cellSize}$. The target landscape fire model treats periodic cell-level ignition as a Bernoulli process with parameter p.

If $\lambda \ll 1$ then the probability of more than 1 fire starting in a cell is negligible, and we may approximate the annual Poisson fire arival process with Bernoulli process with probability $p = \lambda$. The expected number of fires per cell is the same. Given that there are n flammable cells on the landscape, the expected number of fires under the Poisson process is $n\lambda$. The number of fires resultant from n independent trials of a $\mathsf{Bernoulli}(p)$ process is distributed as $\mathsf{Binomial}(n, p)$ with expected value $np = n\lambda$. Thus the expected number of fires per cell and per landscape are conserved under this approximation.

process Variances

We now consider the cell and landscape level variances under the intended Poisson model and the Bernoulli cell level approximation.

The vaiance of a $\mathsf{Poisson}(\lambda = p)$ process is λ. The variance of a $\mathsf{Bernoulli}(p)$ process is $p(1-p) = p - p^2$. When p is small ($p \ll 1$), this square term vanishes, and the Poisson and Bernoulli cell-level variances are essentially identical. The situation is a little different for the landscape level variances in annual fire count. The "true" landscape-level variance is $n\lambda$. The simulated variance is determined by the variance of a $\mathsf{Binomial}(n, p)$ random variable, which is equal to $np(1-p) = n(p - p^2)$ which is close to $n\lambda$. The simuated interannual variance will be smaller than under the Poisson model by a factor of $(1-p)$, no matter the size of the landscape. Again, for $p \ll 1$ the cell level Bernoulli approximation should be adequate.

naive p_0

The empirical value **pEscape** $= p_e$ is an estimate of the probability that a detected fire in the given spatial-temporal frame will exceed the size of a landscape cell. In a spatially discrete system, this is modelled by the event of a fire ignition failing to spread to any neighbouring cell. To model this probability, we may separate the fire spread process into two stages: the initial stage of escape from the cell of origin, and any subsequent spread events. Call this initial spread probability p_0. In a neutral homogeneous landscape, where every cell has 8 neighbours, the event of not escapeing requires that 8 independent spread events fail to occur. From this, ss noted in **scfmEscape.Rmd**, we can derive p_0 from the empirical estimate p_e of the escape probability as:

$$p_0 = 1 - (1 - p_e)^{\frac{1}{8}}$$

p_0: initial spread probability, landscape corrected.

The naive estimate of p_0 assumes that all cells where fires start have 8 neighbours. This is not true on natural landscapes. Let N be the number of flammable cells in the study region, and let $n_i, i = 0, \ldots 8$ be the number of flammable cells in the landscape with exactly i flammable neighbours. The $w_i = n_i/N$ is the proportion of flammable cells having i flammable neighbours. If we assume that ignition probability is independent of the number of flammable neighbours and the initial spread probabilities p_0 are spatially homogeneous, than the probability that a randomly chosen fire will escape is given by the left hand size of the following equation, and for a given escape probability p_e we can find the value of p_0 that would result in such p_e by solving the equation for p_0.

$$\sum_{i=0}^{8} \left(w_i \cdot (1 - (1 - p_0)^i)\right) = p_e$$

This can probably only be done numerically. We do it by using an optimiser to minimize

$$\left|\sum_{i=0}^{8} \left(w_i * (1 - (1 - p_0)^i)\right) - p_e\right|$$

The data necessary for this calculation are provided in the input object **landscapeAttr**. We gloss over the fact that the empirical value of p_e is only an estimate of the landscape-level property leading to the solution for p_0, so there maybe substantial error bounds around this point estimate.

pSpread

This is estimated from mean or expected fire size using a calibration curve generated by simulation. This calibration is an unederstimate because it does not account for the suppression fire spread by non-flammabale areas on a landscape, such as waterbodies, or other non-vegetated areas. Correcting this *grave* deficiency is work in progress.

maxBurnCells

This is derived from the estimated maximum fire size sipmply by dividing by the landscape cell size. This is provided as a parameter to **SpaDES.tools::spread()**. It may default to the number of pixels in the landscape.