

A framework to share healthcare simulations on the web using free and open source tools and python

Thomas Monks ^{1,2}  Alison Harper ¹ 

¹University of Exeter Medical School

²Alan Turing Institute

This work is licensed under a Creative Commons “Attribution 4.0 International” license.



Definitions

Free and Open Source Software (FOSS)

- 0 The freedom to run the program as you wish, for any purpose.
- 1 The freedom to study how the program works, and change it so it does your computing as you wish.
- 2 The freedom to redistribute copies so you can help your neighbour.
- 3 The freedom to distribute copies of your modified versions to others. By doing this you can give the whole community a chance to benefit from your changes.

Personal learning editions of commercial simulation software are not FOSS

If you are using AnyLogic PLE then "*The right to use the Software, Model(s) or Model Output for any other purposes, including commercial purposes or research, is strictly prohibited.*"

A Review of DES FOSS python packages



Dagkakis and Heavey(2016)

- SimPy (Team SimPy, 2020)
- Pysimulator - No recent maintenance
- SciPySim - No recent maintenance

Updates:

- Ciw (Palmer et al. 2019)
- Salabim (van der Ham, 2018)
- De-sim (Goldberg and Karr, 2020)

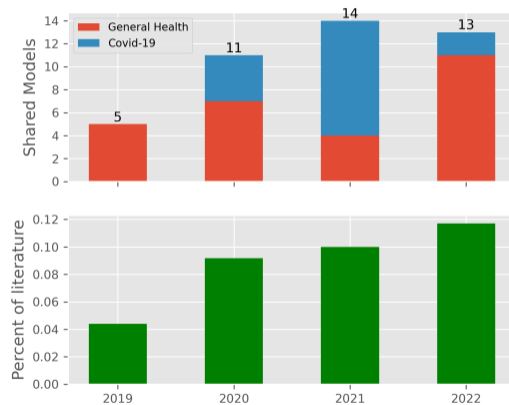
Our definition: DES on the web

A DES model that has been deployed to a remote server, is accessed via a public web URL, and can be executed, in some manner, with varying processing power, without local installation of any software or components.

Motivation

Healthcare DES computer model sharing practice (Working paper, 2023)

Group	Overall	Shared n (%)
Total	485	43 (8.9)
COVID-19	59	16 (27.1)
FOSS	88	26 (29.5)



Commercial software has cloud solutions

Home Board | Epidemic Model

Epidemic
2 years ago

Experiment 1
Variation

Experiment
Simulation

Agent Based Epidemic Model



Developers: AnyLogic
Categories: Healthcare
Application areas: Demographics
Simulation models: Epidemics
Tags: Epidemics

This is an agent based model of spread of contagious disease. The problem

- * We are considering a population of 10000 people. They live in the area
- * A person knows everybody who lives closer than 1 kilometer to him.
- * Initially 10 random people are sick and infectious, and everybody else is susceptible.
- * If an infectious person contacts a susceptible person, the latter gets infected.
- * Having been infected, a person does not immediately become infectious. The people in the latent period are called exposed.
- * The infectious duration after the latent period (i.e. the duration of the infection) is 100 days.
- * During the infectious phase a person on average contacts 5 people he is infectious to.
- * when the person recovers, he gets immune to the disease, but not forever.

The output of the model is the number of infectious people over time.

The terminology and overall structure of that problem is taken from the SIR (Susceptible-Exposed-infectious-recovered) model. SIR problem is a classic problem in systems dynamics. We, however, are adding details that at space and communication dependent on space and uniformly distributed. Agent based approach is its naturalness: we may not know how to derive global dynamics of the disease and can easily model it at individual level.

You can vary the parameters of the model on the fly and watch the disease spread.

The model was created in AnyLogic simulation software / Healthcare

Write a comment

Like

Agent Based Epidemic Model

10000 people



Latency (min)

Latency (max)

Infectious (min)

Infectious (max)

Immune (min)

Immune (max)

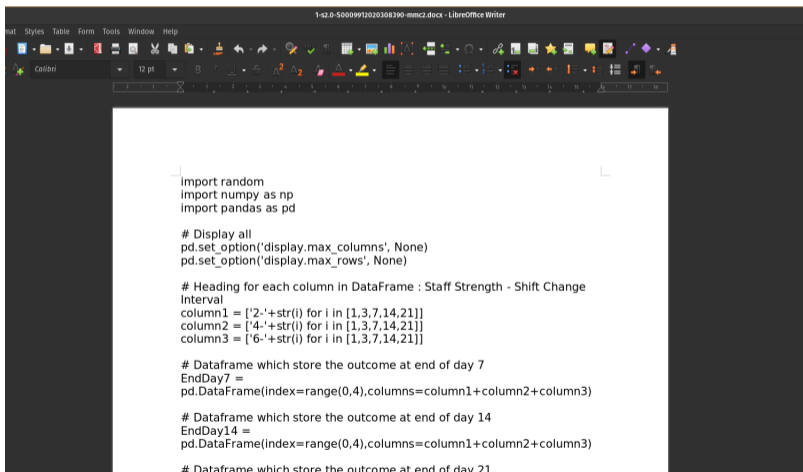
Infection probability

Contacts per day

infectious Exposed Susceptible Immune

Paused

A FOSS model in a Word document

A screenshot of the LibreOffice Writer application window. The title bar reads "1-42.0-S0009912020308390-mm2.docx - LibreOffice Writer". The menu bar includes "File", "Edit", "Format", "Tools", "Window", and "Help". The toolbar contains various icons for file operations, editing, and formatting. The font is set to "Calibri" and the size is "12 pt". The main document area contains the following Python code:

```
import random
import numpy as np
import pandas as pd

# Display all
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

# Heading for each column in DataFrame : Staff Strength - Shift Change Interval
column1 = ['2-'+str(i) for i in [1,3,7,14,21]]
column2 = ['4-'+str(i) for i in [1,3,7,14,21]]
column3 = ['6-'+str(i) for i in [1,3,7,14,21]]

# Dataframe which store the outcome at end of day 7
EndDay7 =
pd.DataFrame(index=range(0,4),columns=column1+column2+column3)

# Dataframe which store the outcome at end of day 14
EndDay14 =
pd.DataFrame(index=range(0,4),columns=column1+column2+column3)

# Dataframe which store the outcome at end of day 21
```

Our objectives for FOSS DES and Python

- 1 Outline a straightforward framework for deploying a simulation developed in Python on the web for users of varying technical skills;
- 2 Provide an applied simulation example implementing our framework;
- 3 Provide guidance for modellers to **begin** sharing models built using FOSS via the web.
- 4 See paper for discussion for very specific advice regarding web-app design.

A framework for sharing python models

Two classes of user

Users with simulation and/or python knowledge

For example, researchers, or health service analysts with training.

Users with no python knowledge

But do have some familiarity with using software!

A summary of the two approaches

Users with simulation and/or python knowledge

FOSS licensed, interactive, and executable scientific notebooks using BinderHub;

Users with no python knowledge

A web app (browser based) model front end built via StreamLit and deployed via Streamlit's Community Cloud

A summary of the two approaches

Users with simulation and/or python knowledge

FOSS licensed, interactive, and executable scientific notebooks using BinderHub;

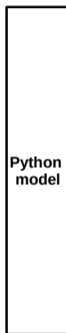
Users with no python knowledge

A web app (browser based) model front end built via StreamLit and deployed via Streamlit's Community Cloud

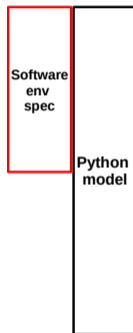
All Users

Supports enhanced model documentation and open working using Github Pages and an (interactive) Jupyter Book.

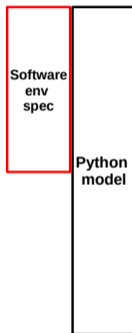
Code a model!



Add dependency management

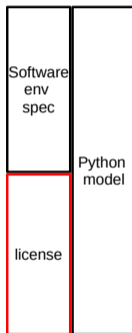


Add dependency management



```
binder > ! environment.yml
1  name: deploy_st
2  channels:
3    - conda-forge
4  dependencies:
5    - matplotlib=3.3.4
6    - numpy=1.19.2
7    - pandas=1.2.3
8    - pip=21.0.1
9    - python=3.8.12
10   - scipy=1.6.1
11   - simpy=4.0.1
12   - pip:
13     - streamlit
14     - treat-sim==0.1.0
```

Add an open license

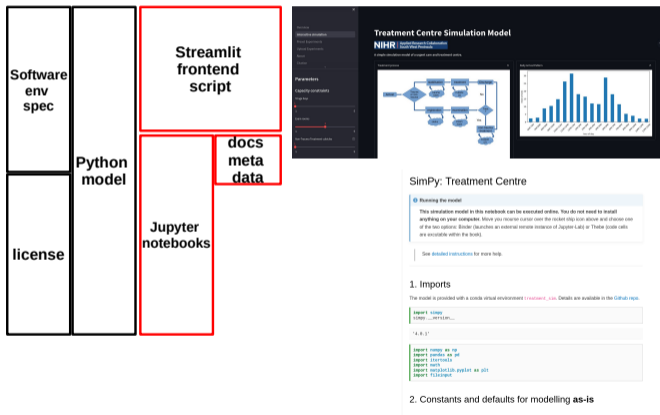


Permissive (e.g MIT, BSD-3)

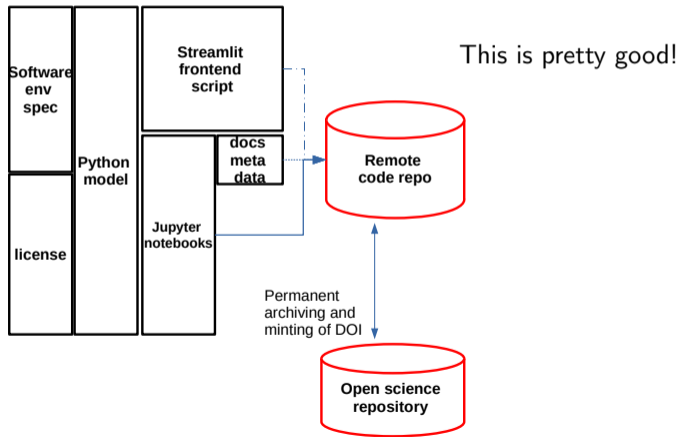
Copyleft (e.g. GPL-3, LGPL)

<https://choosealicense.com>

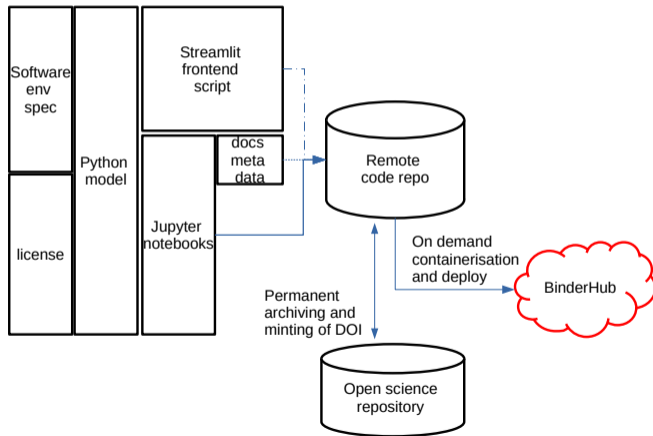
Notebooks, Streamlit App + Meta Data



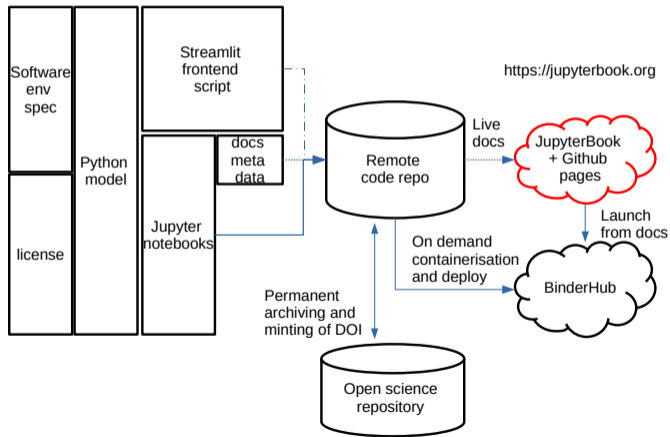
Deploy to a remote repository and mint a DOI



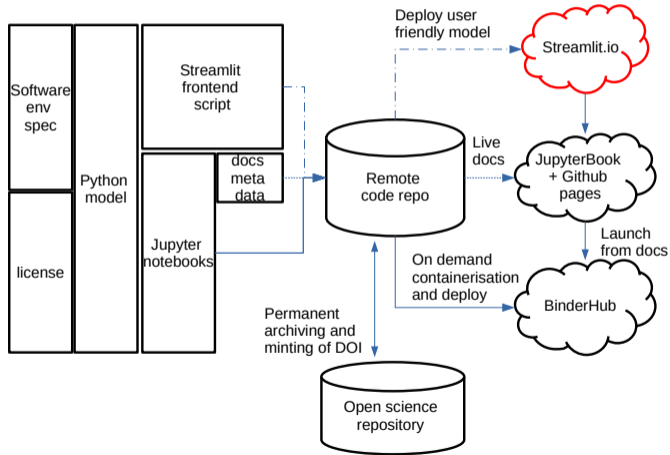
Binderhub



Jupyter Book hosted on GitHub Pages



Streamlit community cloud



Code for StreamLit Simulation Interface

- https://github.com/TomMonks/treat_sim_streamlit
- Cite as:
 - Monks T, & Harper A. (2023).
Treatment Centre Simulation StreamLit
Example: v1.1.0 (v1.1.0). Zenodo.
<https://doi.org/10.5281/zenodo.7561882>



Code for JupyterBook

- <https://github.com/TomMonks/treatment-centre-sim>
- Cite as:
 - Monks, T. & Harper, A. (2022). Treatment Centre Simulation Jupyter Book v1.0.0. Zenodo. <https://doi.org/10.5281/zenodo.6833526>



pythonhealthdatascience.com: BinderHub Tutorial

- `https://www.pythonhealthdatascience.com/content/03_mgt/04_binder/01_binder.html`
- Cite as:
 - Monks, Thomas. (2022). Python for health data science: a hands-on introduction v2.0.0. Zenodo. `https://doi.org/10.5281/zenodo.7107920`

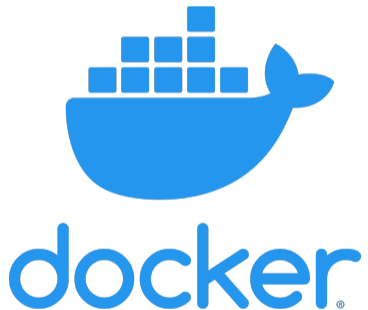


Discussion

Strengths and contributions

- Previewing python models to potential users;
- Sharing runnable code with an organisation that cannot install python;
- Sharing **easy to run code/models** with early career researchers;
- Making the results of a publication repeatable.

Limitations



Extensions to R?



Thank you for listening. Questions?

This work is licensed under a Creative Commons “Attribution 4.0 International” license.

