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

Thomas Monks [1,2] Alison Harper [1]

[1]University of Exeter Medical School

[2]Alan Turing Institute

Definitions

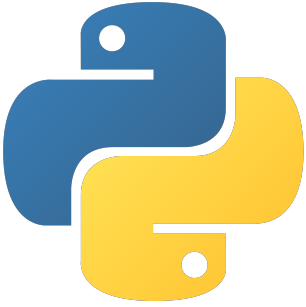
Free and Open Source Software (FOSS)

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)

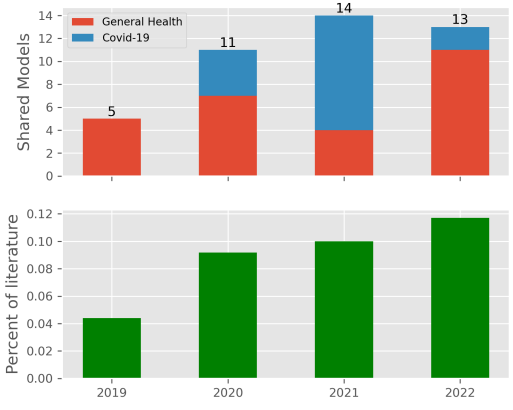
Our definition: DES on the web

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

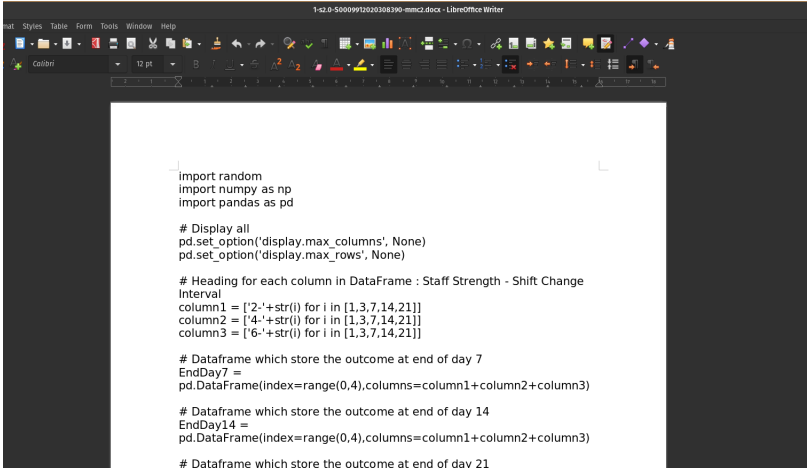
Healthcare DES computer model sharing practice (Working paper, 2023)

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

A FOSS model in a Word document



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;

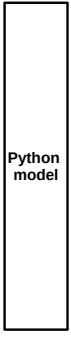
Users with no python knowledge

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

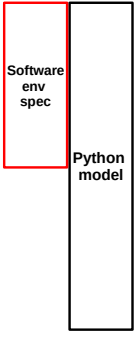
All Users

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

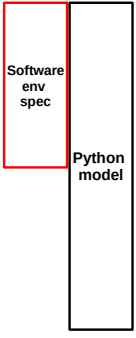
Code a model!



Add dependency management



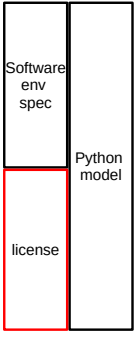
Add dependency management



binder > ! environment.yml

```yaml
name: deploy_st
channels:
  - conda-forge
dependencies:
  - matplotlib=3.3.4
  - numpy=1.19.2
  - pandas=1.2.3
  - pip=21.0.1
  - python=3.8.12
  - scipy=1.6.1
  - simpy=4.0.1
  - pip:
    - streamlit
    - treat-sim==0.1.0
```

Add an open license

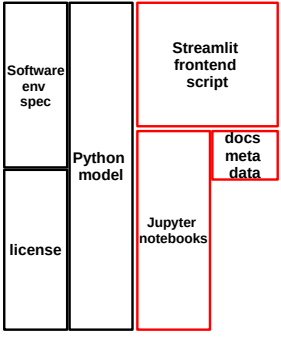


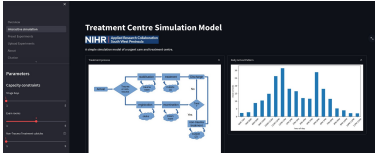
Permissive (e.g MIT, BSD-3)

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

`https://choosealicense.com`

Notebooks, Streamlit App + Meta Data





SimPy: Treatment Centre

Running the model

This simulation model in this notebook can be executed online. You do not need to install anything on your computer. Move you mouse cursor over the rocket ship icon above and choose one of the two options: Binder (launches an external remote instance of Jupyter-Lab) or Thebe (code cells are executable within the book).

See detailed instructions for more help.

1. Imports

The model is provided with a conda virtual environment `treatment_sim`. Details are available in the Github repo.

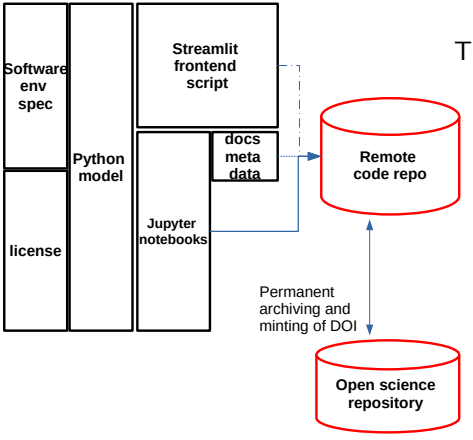
```
import simpy
simpy.__version__
```

```
'4.0.1'
```

```
import numpy as np
import pandas as pd
import itertools
import math
import matplotlib.pyplot as plt
import fileinput
```

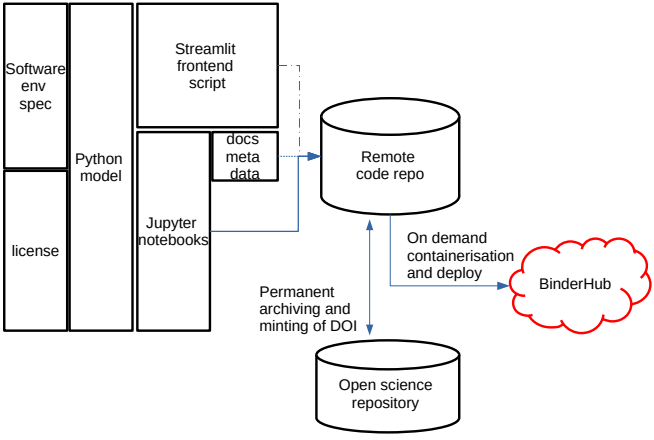
2. Constants and defaults for modelling **as-is**

Deploy to a remote repository and mint a DOI

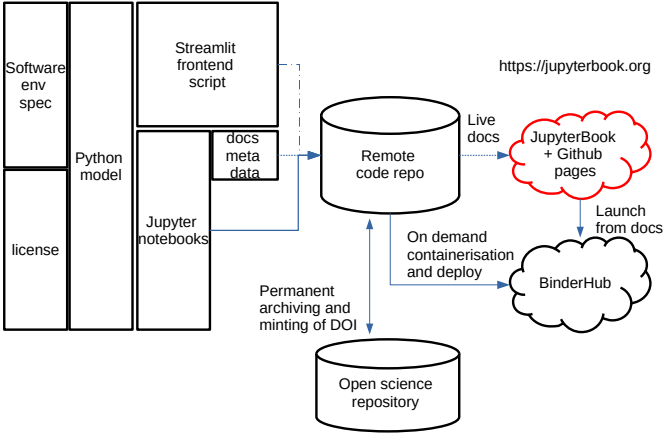


This is pretty good!

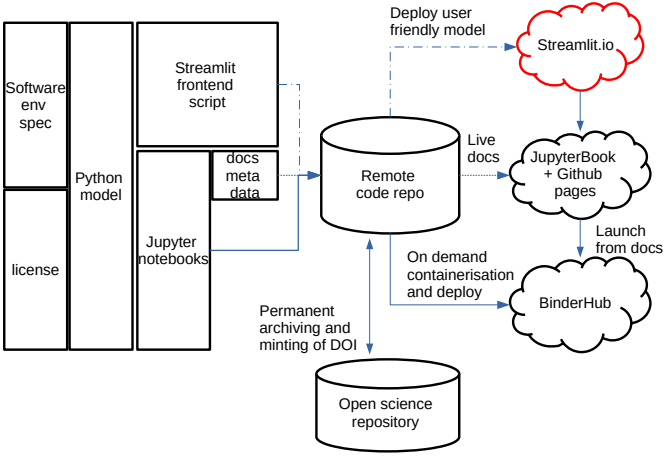
Binderhub


Software env spec
Python model
Streamlit frontend script
docs meta data
Jupyter notebooks
license
Remote code repo
On demand containerisation and deploy
BinderHub
Permanent archiving and minting of DOI
Open science repository

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

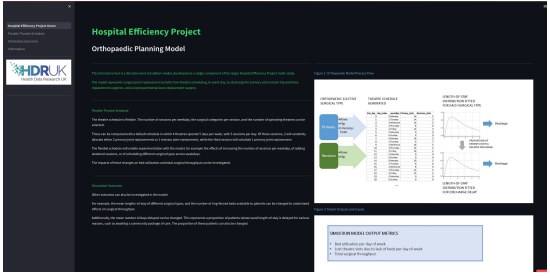
- https://www.pythonhealthdatascience.com/content/03_mgt/04_binder/01_binder.html
- Cite as:
  - Monks, Thomas. (2022). Python for health data science: a hands-on introduction v2.0.0. Zenodo. https://doi.org/10.5281/zenodo.7107920

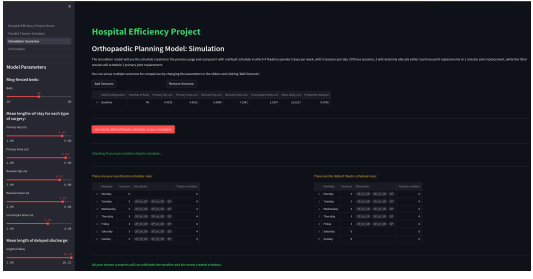
Discussion

Strengths and contributions

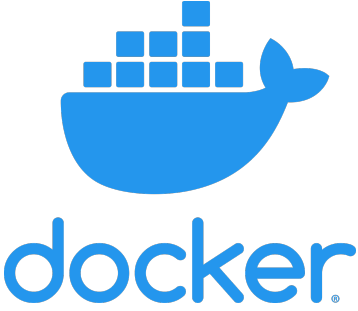
- Previewing python models to potential users;
- Sharing runnable code with an organisation that cannot install python;
- Sharing **easy to run code/models** with early career researchers;
- Making the results of a publication repeatable.

Strength: We use and refine this approach in practice!

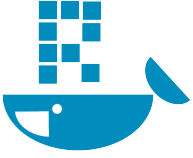




Limitations



Extensions to R?



Thank you for listening. Questions?