Incorporating Deep Learning into System Dynamics:  
Amortized Bayesian Inference for Scalable Likelihood-free Parameter Estimation

## Replication instructions

Incorporating Deep Learning into System Dynamics:

Amortized Bayesian Inference for Scalable Likelihood-free Parameter Estimation

Replication instructions

To run the scripts, you need to have Vensim DSS, Python, and relevant Python packages installed on your computer, which will be listed below with their installation guidelines. Some functionalities like DLL are not available on macOS. DLL and some other packages (e.g., keyboard) require Administrative permissions to be installed and used. Follow the instructions below to install the packages required to run the codes:

1. Download and install Miniconda from https://docs.anaconda.com/free/miniconda/
2. Open a Command Prompt or a Terminal window that is connected to Conda
3. Use the command ‘pip install bayesflow’ to install BayesFlow and related packages
4. Use the following commands to install packages required for VST scripts
5. ‘pip install keyboard’
6. ‘pip install regex’
7. If you want to use the DLL functionality, download the VenPy package from https://github.com/VensimSoftware/venpy
8. Unzip the folder in a familiar location (e.g., Desktop) and, using the Command Prompt window, navigate to that directory (using the cd Desktop command, for example. You can ask ChatGPT or Bard if you struggle with these steps)
9. Go to the directory of the VenPy package, which is called ‘venpy-master.’
10. From within the directory in the command prompt window, use the command ‘pip install .’ to install the VenPy package

After doing these steps, you can run the scripts using the command line or an interactive environment such as Jupyter Notebook (you can navigate to the model’s folder in the command prompt window and use the command ‘jupyter notebook’ to open the interactive environment).

The folder includes:

* The Vensim models (Random Walk and SEIRb).
* Vensim Packaged Models (used with the DLL workflow).
* VST scripts.
* Vensim Optimization Control files.
* Python scripts.

For replication, the Vensim-related files do not need to be changed, although they can be modified through the Python scripts.

Each of the Python scripts includes several sections, some of which require adjustments to produce the desired outcomes. In the first cell, general settings such as parameter names, output variables, and model timelines are shown. Also, you can change the setting for using the DLL functionality from this cell. The first cell also includes the basic neural network settings used for the experiments. Comments are provided next to each hyperparameter, and their definitions and functionalities are explained in the main text of the article.

Other cells belong to importing the required libraries used for running the script and the functions to establish the BayesFlow framework, including but not limited to defining the summary network, inference network, generative model, amortizer, and trainer. Cell number 9 includes the conditions for experiments. You can choose which experiment to run by enabling and changing the flags (e.g., update\_flags(exp\_flags,[0,1,3,4,5,6,13,14],1)).

Cell number 10 includes the graphs for the experiments. To see the results, you must remove the comments from the graphs after enabling the experiment flags. Finally, the last cell includes the settings for a single run. You can modify the values and run the whole script to see the results.