

User Manual for Running Basic Simulations of the Stochastic Compartmental Model ‘OCTA’ for SARS-CoV-2

SIMID Group

April 12, 2024

Introduction

This user manual provides instructions on how to run simulations of the SARS-CoV-2 spread in Belgium using the stochastic compartmental model developed by the SIMID group. This document is aimed at users who have a basic understanding of R programming and are interested in modeling infectious diseases.

Prerequisites

Before running the simulations, ensure that you have:

1. R installed on your system (at least version 4.1.0 is recommended).
2. The required R packages installed. Run the `install.R` script from the project’s main directory to install all necessary packages.
3. The project’s repository from GitHub or Zenodo.

Getting Started

To start a simulation, navigate to the project directory and ensure all input data and configuration files are correctly placed under the `data/` folder.

Running Simulations

Open the R script

To get started, open the `projections_wave1.R` script located in the `R/` folder. This script is responsible for running the model for the first COVID-19 wave in Belgium.

Setting the Working Directory in R

It is essential to set the working directory to the main folder of the project to ensure that all scripts function correctly. The main folder is the directory that contains the `R`, `data`, `doc`, and other related project folders. To set the working directory, use the `setwd()` function in R.

```
# Set the working directory to the main project folder  
# Replace `path_to_project_folder` with the actual path to your project's main folder.  
setwd("path_to_project_folder")
```

Ensure that this path points to the root of the project where all major directories (R, data, doc, etc.) are located. Setting the correct working directory is crucial for the proper execution of the scripts, as they often assume that file paths are relative to this location.

Clear the workspace

The script starts by clearing the R environment to ensure that no residual data or variables interfere with the simulation.

```
rm(list=ls())
```

Load dependencies

It then loads necessary libraries, user-defined functions, and data.

```
source('R/main_vaccination.R')
```

Configure simulation parameters

- Set the output sub-directory in output/ to save the simulation results:

```
output_tag <- 'model_wave1'
```

- Specify the configuration file for parameters:

```
chains_param_files <- 'data/config/MCMCmulti_20230119_d730_belgium_ve00_customQ.csv'
```

- Set the number of parameter sets, stochastic realizations, and the simulation time horizon:

```
num_param_sets <- 5  
num_stochastic_real <- 3  
num_days_sim <- 150
```

Execute the model

Call the main function to run the simulations with the specified settings:

```
projections_vaccination(output_tag=output_tag,  
                        chains_param_files=chains_param_files,  
                        scen_tag='wave1',  
                        num_param_sets=num_param_sets,  
                        num_stochastic_real=num_stochastic_real,  
                        num_days_sim=num_days_sim  
                        )
```

After the simulation completes, check the output directory for results and figures.

Simulating Different Social Contact Patterns

This section explains how to simulate different social contact patterns in the stochastic compartmental model. It is essential for modeling the impact of changes in human interactions due to public health policies or behavioral changes during an epidemic. The steps below are also provided in the `projections_contacts.R` script located in the `R/` folder.

Load Contact Data

Start by loading the contact data from the stored R data file.

```
contact_data <- readRDS('data/social_contact_data.rds')
contact_data$label <- 'CoMix'
names(contact_data)
```

The `contact_data` variable now contains the social contact matrices and related data. The `label` field is set to 'CoMix', indicating the dataset's origin.

Scenario Option 1: Re-use Contact Matrices

In this scenario, you can re-use existing contact matrices by extending them with data from a specific stage. This approach simulates the continuation of social behaviors from an earlier phase.

```
# Extend the existing contact matrices
contact_data$db_C_sim <- rbind(contact_data$db_C_sim[1:6,], contact_data$db_C_sim[1,])

# Set the new date for the extended data
contact_data$db_C_sim$date[nrow(contact_data$db_C_sim)] <- '2020-06-01'

# Label the data as 'prepandemic'
contact_data$label <- 'prepandemic'

# Review the modified contact data
summary(contact_data$db_C_sim)
```

Scenario Option 2: Use New Contact Matrices

For this scenario, you introduce new contact matrices that reflect altered social behaviors, such as reduced contacts due to social distancing measures.

```
# Create new matrices by reducing contacts in the CoMix5 matrices
contact_data[['C_summer2020_asy']] <- contact_data$C_CoMix5_asy * 0.5
contact_data[['C_summer2020_sy']] <- contact_data$C_CoMix5_sy * 0.25

# Integrate these new matrices into the simulation dataset
contact_data$db_C_sim <- rbind(contact_data$db_C_sim[1:8,],
                               c('2020-07-01', 'C_summer2020', 8))

# Label the new scenario as 'summer2020'
contact_data$label <- 'summer2020'
```

In both scenarios, you manipulate the `contact_data` object to represent different assumptions about how people interact during different stages of the pandemic. These changes in the contact matrices directly

influence the model's simulations, showcasing the potential impacts of social measures or changes in public behavior. To execute the model with these new settings, refer back to the “Execute the model” section of this manual, ensuring that the modified `contact_data` is loaded into your simulation environment.

Simulating Different Vaccine Uptake Scenarios

This section describes how to run simulations considering various vaccine uptake scenarios, which can significantly affect the outcomes of the stochastic compartmental model. The process includes selecting specific vaccine uptake files based on scenario needs. The steps below are also provided in the `projections_uptake.R` script located in the `R/` folder.

Select Vaccine Uptake Files

To begin, specify the directory containing the vaccine uptake files and use a pattern to select specific files of interest. This example selects files related to booster doses:

```
# Select files related to booster vaccine uptake
vacc_files <- dir('data/uptake/uptake_vac_voc/', pattern = 'uptake_booster.*csv', full.names = TRUE)
```

The `vacc_files` variable will contain the full path to each file matching the specified pattern, allowing the model to access these during the simulation.

Call the Simulation Function

With the vaccine uptake files and any necessary parameter adjustments set, call the main simulation function to run the model:

```
# Execute the model with specified vaccine uptake scenarios
projections_vaccination(output_tag      = output_tag,
                        chains_param_files = chains_param_files,
                        scen_tag         = scen_tag,
                        num_param_sets   = num_param_sets,
                        num_stochastic_real = num_stochastic_real,
                        num_days_sim     = num_days_sim,
                        vacc_files       = vacc_files,
                        adjusted_parameters = adjusted_parameters)
```

This function call incorporates the selected vaccine uptake files and any adjusted parameters into the model's execution, allowing for detailed exploration of different vaccine strategies. After running the simulations, examine the output to understand the impact of different vaccine uptake scenarios on the model predictions.

Adjust Model Parameters

Depending on the scenario, you might need to adjust specific model parameters. For example, if simulating a scenario omitting the Omicron variant, you might adjust the start point of the variant's introduction:

```
# set introduction of the Omicron VOC after the simulation horizon (i.e. omit this VOC)
adjusted_parameters <- data.frame(VOC_omicron_start = 1e3)

# Execute the model with specified vaccine uptake scenarios
projections_vaccination(output_tag      = output_tag,
                        chains_param_files = chains_param_files,
                        scen_tag         = scen_tag,
                        num_param_sets   = num_param_sets,
                        num_stochastic_real = num_stochastic_real,
                        num_days_sim     = num_days_sim,
                        adjusted_parameters = adjusted_parameters)
```

Support

For further assistance, please contact the project contributors listed in the README file.