Build a DES with R Simmer

ZZhou

January 29, 2018

1 Introduction

Here is the text of your introduction.

2 Start from scratch

Here is the background of the case.

Figure 1: Model Diagram

2.1 Blueprint

We have a model structure in mind. Now think about inputs, events, interactions btw inputs and events, measures, strategies.

1. INPUTS

   Gene type: no finding, high (path, unknown,non-path), mod, low;

   Behavioral: modify behavior or not and whether behavioral change is beneficial or harmful

   Risk: beneficial or harmful change

2. EVENTS (with cost and QALY implication)

   Test: event? attribute?

   Adverse events

3. INTERACTIONS

   Chance of behavioral change differ by gene type

   Risks of adverse events differ by gene type, behavioral change

   Outcomes of adverse events differ by gene type and behavioral change

2.2 Code

Start coding. Some handy functions and tricks.

1. SET ATTRIBUTES

   Each simulation subject can be assigned a set of preset attribute values that carry and can be modified throughout the simulation.

   For example, we can set gene type as an attribute named "gene" in the "initialize_patient" function and draw values 1–6 based on a distribution with probabilites from the inputs list.

   One important status attribute in this model is whether a person is tested or not, so we can create an attribute called "aTest".

2. CREATE BRANCHES

   As noted in the diagram, there could be a few opportunities when simulation subjects go to different branches based on their attributes. To achieve that, we can use the "branch" function in a trajectory. For example, individuals with high penetrance gene finding will have different chances of modifying behavior depending on specific gene type. To reflect this in our model, we can split the trajectory into six branches and set whether to modify behavior accordingly.

3. REGISTER EVENTS

   DES processes events based on a time order. We can simulate time to a composite adverse event with a risk depending on gene type and behavioral change. An event can be depicted with two functions: one time function ("time_to_AE") and one event function ("AE_event"); the time function returns time to an event based on risk parameters from the list of inputs; the event function builds the trajectory to command what to do next. When an adverse event occurs, the event function records a counter called "AE" using the "mark" function (a wrapper of seize and release without timeout in between), which will reflect in simulation results and help count frequency of adverse events. Please note that "counter" and both event functions need to register in the "counters" and "event_registry" list respectively.