

Network Model Extensions

Network Modeling for Epidemics

Day 5

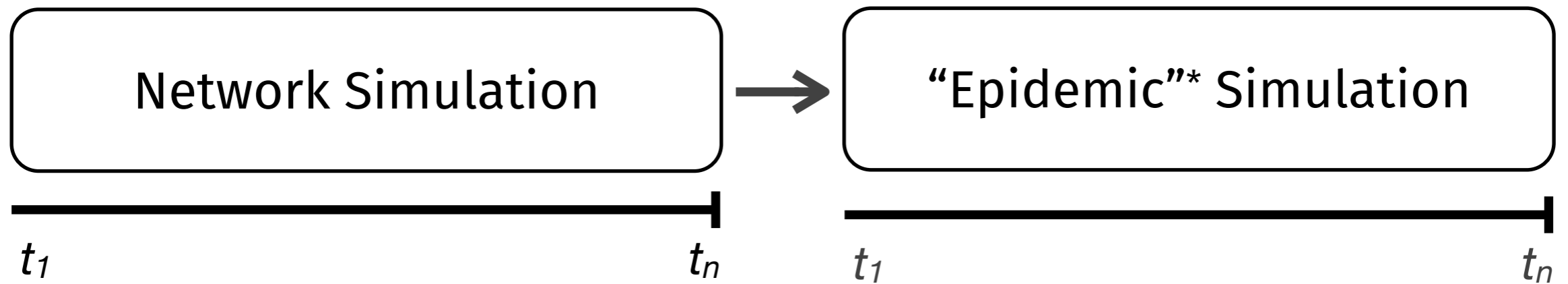


Outline for Rest of Week

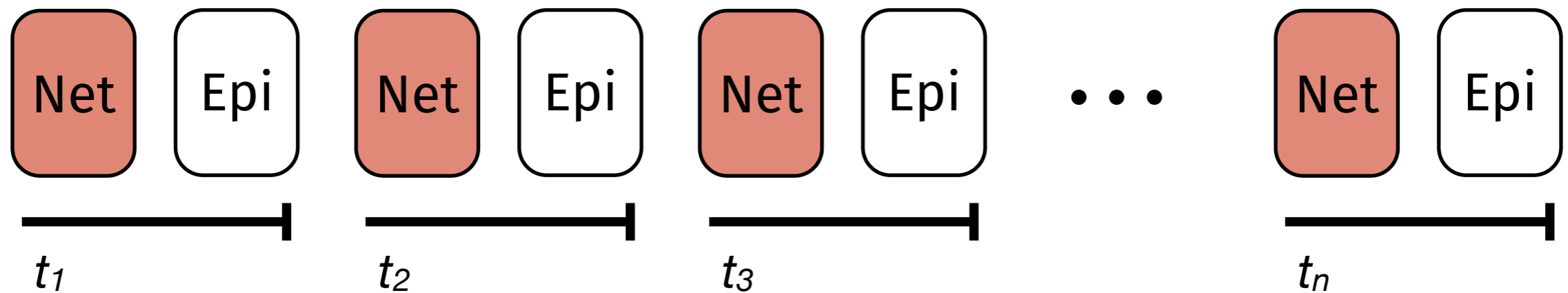
- Wednesday
 - Modeling epidemics + networks = modeling epidemics over networks
 - Core assumption: no feedback of epidemiology on networks
 - One important implication: closed populations
 - Still feedback: *network structure* \implies *epidemiology* and *incidence* \implies *prevalence*
 - Built-in **epidemiology** types (SI, SIR, SIS)
 - Working with nodal attributes, with heterogeneity in network structure and epidemiological parameters
- Thursday
 - Feedback: epidemiology \implies network structure
 - Vital dynamics, “sero-sorting” (edge formation based on changing nodal attributes)
 - Simple vaccine intervention
 - Built-in **epidemiology** types (SI, SIR, SIS), then getting started with extensions
- Friday
 - Getting comfortable with extensions
 - Building a network-based extension model for COVID, step-by-step...

Model Feedback

Models without Feedback

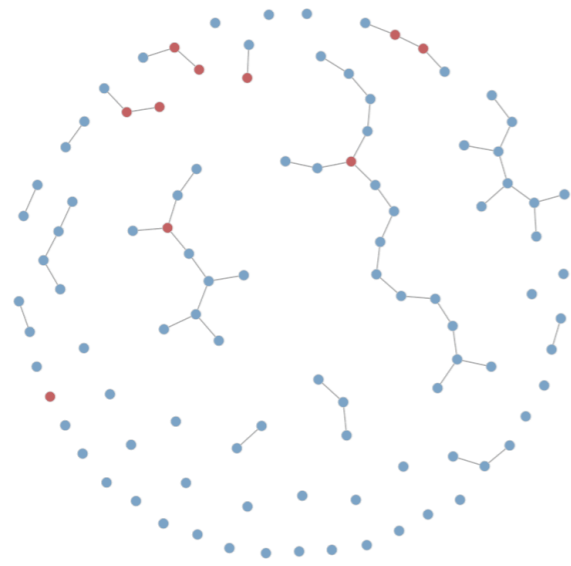


Models with Feedback



"Epidemic" = biological, behavioral, demographic, etc., changes*

EpiModel Extensions

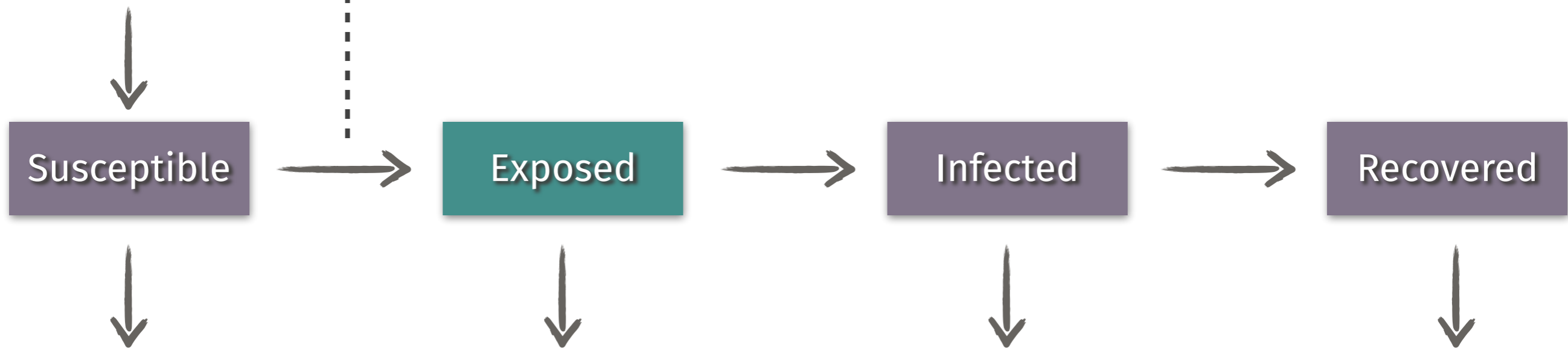


Modifiable

Basic structure of states and flows

Modifiable

Epidemic parameters
Dynamic network structure



The EpiModel Extension API

1. Modules have associated function with standard inputs and outputs
 - Inputs are `dat` and `at`, outputs are `dat`
2. Modules use the accessor `get_/set_` functions to read and write sublist data to the `dat` object; See `help("net-accessor")`
 - epi summary stats updated with single value defined for current time step
 - nodal attributes updated, with entry for all nodes on network
3. Modules defined and called into `EpiModel::netsim` through `control.net` settings
 - Each defined module parameter ends in `.FUN`
 - `type` control setting should be set to `NULL` for any extension models
 - Standard modules — handling `dat` initialization, network resimulation, and network data updates — are not intended to be edited by end users (but may be as necessary)

The EpiModel Extension API

4. Nodal attributes may be initialized on the network as prior to TERGM estimation with `set_vertex_attribute`
 - Applies even if attributes are not called during TERGM estimation
5. Models with departures (e.g., mortality) must update nodal attributes
 - `active` must be set to `0` for all departing nodes
 - `exitTime` must be set to `at` for all departing nodes
6. Models with arrivals (e.g., births) must append nodal attributes for incoming nodes
 - `append_attr` used in arrival module to set new nodal attributes

Schedule for Friday

(approximate)

| Session | Type | Title | Start (PST) | End (PST) |
|---------|------|---|-------------|-----------|
| 1 | Lec | Overview | 8:00 | 8:10 |
| 2 | Lec | Applied Network Models | 8:10 | 8:40 |
| 3 | Disc | Split discussion of HIV vs COVID models | 8:40 | 9:20 |
| | | break | 9:20 | 9:30 |
| 4 | Tut | Adding COVID Demog | 9:30 | 10:15 |
| 5 | Lab | Exper COVID Demog | 10:15 | 11:00 |
| | | lunch | 11:00 | 12:00 |
| 6 | Tut | Adding Asympt, Interv | 12:00 | 12:45 |
| 7 | Lab | Exper COVID Interven | 12:45 | 1:45 |
| | | break | 1:45 | 1:55 |
| 8 | Lec | Final considerations | 1:55 | 2:15 |
| 9 | Disc | Discussions, consultations | 2:15 | 3:00 |