$\alpha \in \{\frac{1}{2}, 1, 2\}$ $\quad p^S(Y)^\alpha$

Examples for amortized SBI

① robot arm ⇒ see earlier

② Epidemiology: SIR model [Kermack & McKendrick 1927]

- compartments:
  - Susceptible S: people who are healthy but can get infected
  - Infected I: —"— ill and can transmit disease
  - Recovered R: —"— healthy again and now immune

  in the most simple model variant
- simplifying assumptions:
  - stationary dynamics: behavior of virus/bacteria and people does not change over time (no mutations, no countermeasures)
  - only consider averages over all people in each compartment
    ⇒ all people in same compartment are considered identical
  - simplest variant 3 compartments $S\ I\ R$, $S + I + R = N$ population size

- design model:
  - a healthy individual meets on average λ_1 people per day
    (assume that N is so big that λ_1 is independent of N)
  - infected people are not isolated, but meet others as usual

    $\Rightarrow$ a fraction $\frac{I}{N}$ of the λ_1 meetings is potentially dangerous
  - a fraction λ_2 of all the dangerous meetings actually leads to transmission
  - if we observe a reduction in infection, we cannot distinguish if
    people became more cautious ($\lambda_1 \downarrow$) or virus is less infectious ($\lambda_2 \downarrow$)

    $\Rightarrow$ can only recover $\lambda = \lambda_1 \lambda_2$

    $\Rightarrow$ number of new infections per day: $\lambda \frac{I}{N} \cdot S$ $\quad$ λ: infection rate
  - nobody dies, infected people recover after δ days on average

    $\Rightarrow$ number of people to recover per day: $\frac{1}{\delta} I = \mu I$ $\quad$ μ: recovery rate
  - diagram $\quad \boxed{S} \xrightarrow{\lambda} \boxed{I} \xrightarrow{\mu} \boxed{R}$

- write the dynamics as a system of ordinary differential equations (ODEs)

$$\frac{dS(t)}{dt} = -\lambda \cdot \frac{I(t)}{N} \cdot S(t)$$ minus: infected people leave compartment S

$$\frac{dI(t)}{dt} = \lambda \frac{I(t)}{N} S(t) - \mu I(t)$$ minus: recovered leave comp. I

$$\frac{dR(t)}{dt} = \mu I(t)$$ immunity of R $\hat{=}$ no one leaves comp. R

- can divide all equations by $N \Rightarrow [S] = \frac{S}{N}$ etc.

$$\frac{d[S(t)]}{dt} = -\lambda [I(t)] \cdot [S(t)], \quad \frac{d[I(t)]}{dt} = \lambda [I(t)][S(t)] - \mu [I(t)], \quad \frac{d[R(t)]}{dt} = \mu [I(t)]$$

- adding of three equations proves correct normalization

$$\frac{d[S(t) + I(t) + R(t)]}{dt} = 0 \qquad [S(t) + I(t) + R(t)] = \text{const.} \overset{!}{=} 1$$

- to solve equations, must define initial conditions, e.g. values at $t = 0$

$I_0 = [I(t=0)] = \text{const}$ initial number of infected (when disease is first detected)

$[R(t=0)] = 0$

$[S(t=0)] = 1 - I_0$

$\Rightarrow$ given I_0, λ, μ we can solve the ODEs for any time $t>0$ by "integration"

- simplest method: Euler forward method: use discrete time steps Δt and approximate: $[S(t+\Delta t)] = [S(t)] - \lambda [I(t)] [S(t)] \cdot \Delta t$

  $\left[\frac{dS}{dt} = \lim_{\Delta t \to 0} \frac{S(t+\Delta t) - S(t)}{\Delta t} \right.$, now stop lim at finite $\left. \Delta t \right]$

  $[I(t+\Delta t)] = [I(t)] + (\lambda [I(t)] [S(t)] - \mu [I(t)]) \Delta t$

  $[R(t+\Delta t)] = [R(t)] + \mu [I(t)] \Delta t$

- theory say that Euler forward is a good approximation if Δt is small enough

  $\Delta t \le \min_t \left(\frac{2}{\lambda [I(t)] [S(t)]}, \frac{2}{|\lambda [I(t)] [S(t)] - \mu [I(t)]|}, \frac{2}{\mu [I(t)]} \right)$ (I'm not 100% sure)

- more sophisticated solvers (e.g. Euler backward, Runge-Kutta methods) allow larger time steps

- define observables X
  - report on every day number of new infections and newly recovered
  - observations are not perfect $\Rightarrow$ observation model
    - reporting delay. $\Delta S^{obs}(t) = f\left(\Delta S^*(t-L)\right)$ L: delay
    - underreporting: $\Delta S^{obs} \sim \rho \, \Delta S^*$ ρ: fraction of detected cases
    - noise
  - exact values $\Delta S^*(t) = S^*(t-\Delta t) - S^*(t)$

$$\Delta R^*(t) = R^*(t) - R^*(t-\Delta t)$$

  - measured values $\widetilde{\Delta S}(t) = \Delta S^*(t-L) \cdot \varepsilon_S$ $\varepsilon_S \sim N(\rho, \sigma^2)$

relative error, because multiplicative

$$\widetilde{\Delta R}(t) = \Delta R^*(t-L) \cdot \varepsilon_R \qquad \varepsilon_R \sim N(\rho, \sigma^2)$$

(for simplicity, we assume that observation parameters are equal for $\widetilde{\Delta S}$ and $\widetilde{\Delta R}$)

• full simulation: $Y = [I_0, \lambda, \mu, L, \varrho, \sigma^2] \sim p^S(Y)$

(for synthetic data) usually, independent prior $p^S(Y) = \underbrace{p^S(I_0)\, p^S(\lambda)\, p^S(\mu)\, p^S(L)\, p^S(\varrho)\, p^S(\sigma^2)}_{\text{choose according to epidemiological prior knowledge}}$

$X = \phi(Y, \eta)$ ODE + reporting noise

$$\eta = [(\varepsilon_S(\Delta t), \varepsilon_R(\Delta t)), (\varepsilon_S(2\Delta t), \varepsilon_R(2\Delta t)), \ldots, (\varepsilon_S(T), \varepsilon_R(T))]$$

$$X = [(\Delta\hat{S}(\Delta t), \Delta\hat{R}(\Delta t)), (\Delta\hat{S}(2\Delta t), \Delta\hat{R}(2\Delta t)), \ldots, (\Delta\hat{S}(T), \Delta\hat{R}(T))]$$

N, T are hyperparameters or sampled $N \sim p^S(N)$, $T \sim p^S(T)$

$\Rightarrow$ use simulation to create a large $TS = \{(Y_i \sim p^S(Y), X_i = \phi(Y_i, \eta \sim p^S(\eta)))\}_{i=1}^{M}$

[recall hidden Markov model:

$$\begin{tikzcd} \arrow[r] & \boxed{s_t} \arrow[r] \arrow[d] & \boxed{s_{t+1}} \arrow[r] \arrow[d] & \boxed{s_{t+2}} \arrow[r] \arrow[d] & {} \\ & \boxed{x_t} & \boxed{x_{t+1}} & \boxed{x_{t+2}} & \end{tikzcd}$$

$p(s_t \mid s_{t-1}) = \delta(s_t - \text{Euler})$

- traditional non-probabilistic parameter fitting: least squares

$$\hat{Y} = \arg\min_{Y} \mathbb{E}_{\eta \sim p^s(\eta)} \left[\| X^{obs} - \phi(Y, \eta) \|^2 \right]$$

real data

$\phi(\hat{Y}, \eta)$ should reproduce real observations

disadvantages: - non-linear least squares might get stuck in a bad local optimum

- disregards the uncertainty and ambiguity in Y

at best, we get $\hat{Y} = \arg\max_{Y} p^s(Y \mid X^{obs})$ ← true posterior of simulation

but not the full $p^s(Y \mid X^{obs})$

- amortized SBI learns a generative neural network for $p(Y \mid X^{obs}) \approx p^s(Y \mid X^{obs})$ using a large TS of synthetic data ($M = 10\,000+$)

- full algorithm:

1. define the simulation $\phi(Y, \eta)$ and priors $p^s(Y)$ and $p^s(\eta)$
2. use simulation to generate synthetic TS
3. set up architecture of generative neural network

X (or $\log X$)

$\downarrow$

$\boxed{\text{summary network}}$ ← any feature detector, for time series with varying length T

⇒ recurrent network (RNN, LSTM, transformer)

map variable length X to fixed length $h(X)$

$\downarrow h(x)$ learned features of X

$\hat{y} \sim p(Y \mid h(x)) \leftarrow \boxed{\text{cNF}} \leftarrow z \sim \mathcal{N}(0, \mathbb{I})$

(4) train cNF & summary network jointly using NLL loss

$$\hat{p}, \hat{h} = \arg\min_{p,h} \frac{1}{M} \sum_{i=1}^{M} -\log p(Y_i \mid h(X_i))$$

⇒ $h(x)$ will become optimally informative for $p(Y \mid h(x))$

(5) validate $\hat{p}, \hat{h}$ (check calibration, sensitivity etc.) using a synthetic test set

⇒ $\hat{p}, \hat{h}$ faithfully represent the simulation posterior

(6) infer $\hat{p}(Y \mid h(X^{obs}))$ for real data

(7) check for potential simulation gap (≙ simulation is unrealistic, "model misspecification")