May 17, 2020

PLANNING AS INFERENCE IN EPIDEMIOLOGICAL DYNAMICS MODELS

Frank Wood, Andrew Warrington, Saeid Naderiparizi, Christian Weilbach, Vaden Masran1,William Harvey1, Adam ´ Scibior, Boyan Beronov, and Ali Nasseri

The article discusses planning as inference as decision making tool in epidemics crisis (particulary COVID-19) . Current approaches of model-based decision making can be significantly improved . The probabilistic models are already developed by epidimioligists , but the inference can (computing the required conditional probabilities on latent varaiables) is done automatically by the proposed tools . The planning problem is formulated as inference problem .

Assumptions:

There exists some current population, and the status of its constituents with respect to disease status is only partially known. There exists a disease whose transmission properties may only partially be known, but whose properties cannot themselves be readily controlled. There exists a population dynamic that can, in part, be controlled. There exists a "policy goal" or target which we will refer to as the allowable, allowed, or goal set.

Approach :

- Define the latent and the control parameters and priors over them
- Define likelihoods on observed data and/or define constrains we want to keep
- Do inference on posterior of control values given Data and constrains
- Choose optimal control values

Model:

Global parameters (θ, η) controlled and uncontrolled and discrete time dependent variables X_t, Y_t, Z_t (Latent vars we are interested in , Observed vars , and latent vars we are not interested in, but essential for the model).
The following factorization is assumed :
$p(\theta, \eta, X_{0:T}, Y_{0:T}, Z_{0:T}) = p(\theta)p(\eta)p(X_0, Y_0, Z_0) \prod p(X_t, Y_t, Z_t|X_{t-1}, Y_{t-1}, Z_{t-1}, \theta, \eta)$ (η, θ are independent) .

Inference:

Compute $p(\eta, X_{0:T}|Y_{0:T}, \theta)$ or $p(X_{0:T}|\eta, Y_{0:T}, \theta)$ if η is known . The difficulty is that the simulator doesn't provide a capability to sample from it directly (And $Z_{0:T}$ can't be integrated out analytically) . PP tools provide a mechanism to compute it automatically .
In this case approximate Bayasian computation (ABC)is a suitable tool

Control as inference:

Instead of conditioning on observed data we condition on the desired state using binary (yes/no) auxiliary variables Y_t. Using rejection sampling : sample θ from prior , run full simulation , keep only θs when $Y_t = 1$. We can do it more sophisticated $p(\theta|p(\forall_t : Y_t = 1|\theta) > p_0)$ with nested Monte Carlo method .

Model predictive control:

Choosing the policy in time T given new data Y_T^{data} by making standart inference on latent variable(given new data) and then based on data and new belief state regarding X_t, Z_t performing control stage $p(\theta|Y_T^{aux} = 1)$. It's possible also to upgrade our model to perform a long term planning - finding a sequence of policies to achive the desirable goal ($\theta_0 ... \theta_t$) (beware of phenomen of reversed casuality)

Automation:

Evaluating each one of previously mentioned conditional distributions can be automated using PP tools .

Probabilistic programming :

The major challenge in designing useful PPL systems is the development of general-purpose inference algorithms that work for a variety of user-specified programs. Here the author give an example of PyProb that is compatible with existing stochastic simulators (David has objections regarding this way, why?).

Models:

The authors discusses two models that represent two types : compartmental (Macro) and agent-based(Micro) .
Compartmental models divide the population into groups (compartments) with respect to disease (susceptible (S), infectious (I), exposed (E), and recovered (R)) , This models are less useful as policy making tool unless the number of compartments is large enough (demographic info , spacial strata etc.).

Agent based models reflects system dynamics through discrete agents interaction (that mimic household, people etc)

Compartmental model :
SEIR model .

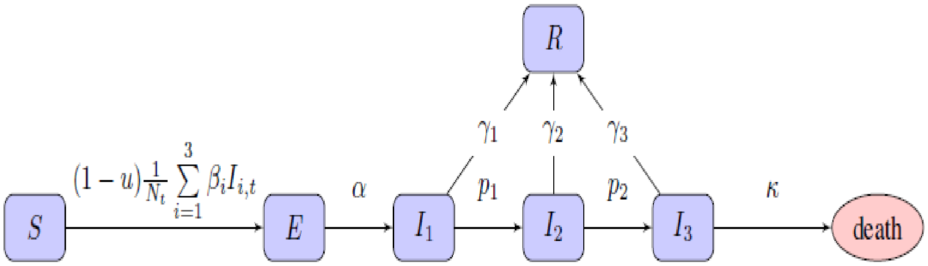

Figure 1: Flow chart of the SEI^{3}R model we employ. A member of the susceptible population S moves to exposed E after being exposed to an infectious person, where "exposure" is defined as the previous susceptible person contracting the illness. After some incubation period, a random duration parameterized by α, they develop a mild infection (I_1). They may then either recover, moving to R, or progress to a severe infection (I_2). From I_2, they again may recover, or else progress further to a critical infection (I_3). From I_3, the critically infected person will either recover or die.

Where dynamic parameters (Greek letters) are unknown , with it's own prior. In addition there is one control parametr $u \in \{0,1\}$. There is also equasions that describe spatial dynamics of the variables (that depends on parameters and u) : for example the expression above the first arrow .
Given exact values to the parameters make the model fully deterministic . **Policy goal:** $Y^{aux}_{0:T} = \mathbb{I}[(max_t(\sum_n I_{n,t})) < C]$.
To estimate the uncontrollable parameters point-wise , measurable quantities can be used .For example :

$$\text{incubation period} = \alpha^{-1} \tag{21}$$

$$\text{mild duration} = \frac{1}{\gamma_1 + p_1} \tag{22}$$

$$\text{severe duration} = \frac{1}{\gamma_2 + p_2} \tag{23}$$

$$\text{critical duration} = \frac{1}{\gamma_3 + \kappa} \tag{24}$$

$$\text{mild fraction} = \frac{\gamma_1}{\gamma_1 + p_1} \tag{25}$$

$$\text{severe fraction} = \frac{\gamma_2}{\gamma_2 + p_2} \cdot (1 - \text{mild fraction}) \tag{26}$$

$$\text{critical fraction} = 1 - \text{severe fraction} - \text{mild fraction} \tag{27}$$

$$\text{fatality ratio} = \frac{\kappa}{\gamma_3 + \kappa} \cdot (\text{critical fraction}). \tag{28}$$

Stochastic model can be much more useful for decision maker . For example define confidence intervals of 95% on βs and left hand side of equations 21 -24 .Sample from uniform distbution that defined over this intervals each t time unit. And then solve the equations 21-28 , and solve (simulate) the model .

Agent based model: FRED

FRED captures many particular characteristics of specific areas by simulating every region down to the resolution of households , workplaces etc . with discrete steps of one day .Following the recipe of control as inference we define : prior on controls in terms of FRED internals,how FRED params relates to η and how to condition on desired outcome $Y^{aux}_{0:T}$ (???) . The author describes how stocastic simulator can be seen as a probabilistic model and (using PP) can be re purposed to perform automatic inference . Given parametr file of $\eta\theta$ FRED produces sampling from $p(X_{0:T}, Z_{0:T}|\theta, \eta)$. The high level stages to convert FRED to PP model of PyProb are listed .

Experiments SEIR .

Author executed several simulation following what was described in the paper : deterministic ,stochastic simulation , Model predictive control and policy based control (find the optimal policy while there are two control variables) . It is pure python implementation (model inference and control) that implements inference using nested Monte Carlo

Experiments FRED . 5 controls are presented with uniform priors with intervals of interest (for example Hand washing compliance rate $\theta \sim N(0, 1)$ etc .) Also indication variable Y_t was presented (Whether infected population is less then 10% in day t). Then , by generating one million samples , the researchers inferred the posterior on θ by sampling them from prior , running the simulations , and keeping only the results with $Y_t = 1$ for 150 days . The simulation began with 10 randomly infected agents at time t.
What is also interesting that marginal distributions of single and pairs of control parameters was inferred . This is how the efficient policy can be derived (using some cost/utility function) . For example we can see that if we have high hand washing compliance then we need low shelter in place duration rate (isolation time).
Open source FRED simulator interfaced with PyProb
(PP system designed to perform inference on pre existing simulators) .