1 Background-ASCII 1-3

1.1 The Observation Operator

Recall that in general, an observation operator is an arbitrary noninvertible projection from the space of sequences (the state-space transitions and the times they occur). In other words, in the notation of our paper, it is a non-invertible function of $(\vec{\sigma}, \vec{\tau})$. (In math-speak, if we take our observation operator to be deterministic, the combination of our observation operator with the SFT net is a "unifilar hidden markov process model"). In normal-person-speak, we are attempting to find the model that descirbes the transmission of messages among nodes. We know that what we observed (the message times) was generated by applying the observation operator to a more complete model. In other words, the observation operator hides from us some variables of the underlying model. Furthermore, this observation operator is not invertible so we can't just apply it to what we observed as an attempt to get the underlying model.

1.2 Likelihood issues

Our likelihood function gives the likelihood of the sequence of fully-specified state-space variables (see Eq. 17 in our paper). This includes message times, the infection times as well as the content of the message.

So to calculate the likelihood of an observation (whatever it is conditioned on), we must integrate the expression in Eq. 17 over the degrees of freedom that the observation operator projects out. In general, a horrible problem. In our case, this translates into integrating over the infection times.

1.3 Our Special Scenario

However the scenario that Brian constructed is special ...

In this scenario, what the observation operator gives is the time-stamps at which nodes emit messages. All other information - states of nodes at all times, and contents of messages at all times - is projected out.

Moreover, we know what the (stationary) Poisson rate of such emissions would be from any node v if v were infected, and what the rate would be if v were not infected. So we can write down the likelihood of any sequence of emissions generated by v if v were infected, and the likelihood if v were not.

We also make the assumption that once a node gets infected, it stays infected.

2 The Likelihood-ASCII 4-8

2.1 If we knew the infection times...

. We can write down in closed form the likelihood of a sequence of (time-stamps of) emissions from an arbitrary node v across all time, given that v is uninfected up to z_v, when it gets infected.

$$P(data|z) = \prod_{v \in V} \frac{e^{-\lambda_{v1} z_v} (\lambda_{v1} z_v)^{k_{v1}}}{k_{v1}!} \times \frac{e^{-(\lambda_{v2})(T-z_v)} (\lambda_{v2})(T-z_v)^{k_{v2}}}{k_{v2}!} \quad (1)$$

where k_{v1} is the number of messages emitted from node v before z_v, k_{v2} is the number of messages emitted from node v between z_v and T, λ_{v1} is homogenous poisson transmission rate for node v when it is uninfected and λ_{v2} is the homogenous poisson transmission rate for node v when it is infected.

Note that this is conditioning on *part* of what it hidden in the output of the observation operator, but not everything. Formally, to write down this likelihood we are actually integrating over the values of the other hidden variables concerning the messages from v that we cannot observe, namely the specifications of whether those messages contain malware or not. But for us that integration is trivial.

2.2 The problem

However the likelihood that we want to calculate is different than equation **??**. For example, in the simplest scenarios we might want to test, we would want the likelihood of our entire observation sequence either conditioned on the premise that one particular node v' is infected at t = 0 or that no nodes are ever infected. Note that in this likelihood we do NOT specify the times of infection for the other nodes for the case that v' is infected at t = 0. In fact, we don't even specify m, the number of nodes that get infected by time T.

However, we can use equation **??** as part of the likelihood. To see how, note that if we knew the infection times for all nodes v in the net that get infected, z_v, we would be done. In other words, in an N-node SFT net, where we can have between 1 and N nodes get infected in the time interval [0, T], it suffices to specify a point z in the union of spaces $\cup_{1 \leq M \leq N} R^M$ to fix the likelihood of our data. Alas, we do not know that vector z.

6) Therefore to calculate $P(data|\text{net infected at } t = 0)$, we need to do an integral, integrating out the vector z. We can decompose this integral as follows:

$$\int dz P(d, z|\text{net infected at } t = 0) =$$

$$\int dz P(d|z, \text{net infected at } t = 0) P(z|\text{net infected at } t = 0) \quad (2)$$

Again, the first term on the RHS is just equation **??** and is trivial to compute. What to do about the second?

2.3 Calculating P(z)

SUBSTANTIAL CHANGES 2/21/14

One issue that presented itself is that sometimes, all nodes do not get infected within the observation window. To deal with this problem, we have to consider all possible sequences of infection times, including those sequences that do not include some of the nodes.

More formally, let D be the the set of all sequence of infection times consistent with the cybernet, M is the size of such a sequence, the elements of such a sequence form an M-vector s^M, and z^M is a vector in R^M all of whose components lie in [0, T]. (Note that we are no longer calling the set D a DAG based on the discussion on Feb 20, 2014. It turned out that a graphical representation of node infection orderings was not natural).

What makes the analysis slippery is the that the sizes of the latter two random variables (s^M and z^M) are set by the value of the first random variable. In other words, if we knew the infection sequence, we would know the size of s^M and z^M, To clarify things, let's transform to a new coordinate system in which both of those latter two random variables are N-dimensional:

To do this, augment the index set of the cybernet nodes, $I = \{1, ..., N\}$, to include a special "null" value, $*$. Label that augmented set (which has $N + 1$ elements) by $I*$. Then for all M, map each $s^M \in I^M$ to $s \in (I*)^N$, where each s_i is an element of of I if $i <= M$, and all remaining components s_j equal $*$.

Similarly augment the space $\tau = [0, T]$ to include a special "null" value, %. Label that augmented space by $\tau\%$. Then for all M, map each $z^M \in \tau^M$ to $\bar{z} \in (\tau\%)^N$, where each $\bar{z_i}$ is an element of τ if $i \leq M$, and all other components of z equal %. (Aside: I use the symbol $\bar{z}$, since z already means something in the first coordinate system.)

Note that just as the analysis for the second coordinate system we were not interested in integrating all of τ^M, but only the subvolume where $z_1 < z_2 < ...$, so in the this coordinate system we will not be interested in integrating over all of $[\tau\%]^N$, but only over the subvolume (where until we hit the first component of $\bar{x}$ with a%), $\bar{z_1} < \bar{z_2} < ...$

In this coordinate system, we can decompose the integral in equation **??** as

$$\sum_M \sum_s \int d\bar{z} [P(d|M, s, \bar{z}) P(\bar{z}, s, M|\text{net infected at t} = 0)\ ] \tag{3}$$

where there is an implicit delta function forcing $\bar{z}$ and s to have $(N - M)\%$'s and $*$'s, respectively.

At this point we must do something novel, due to the random variable M. In particular, we cannot do as we were doing and start by evaluating something like $P(\bar{z_1}, s_1|M, \text{net infected at t} = 0\ )$. The reason is that knowing the total number of nodes that will get infected before T distorts the probability that s_1 is the first node to get infected, and also distorts the probability that $\bar{z_1}$ is the time it gets infected. So calculating $P(\bar{z_1}, s_1|M, \text{net infected at t} = 0\ )$ is actually quite hard.

One way forward is as follows:

$$P(\bar{z}, s, M|\text{ net infected at t = 0 }) =$$
$$P(M|\bar{z}, s,\text{ net infected at t = 0 }) \times P(\bar{z}, s|\text{ net infected at t = 0 }) =$$
$$\delta(M = \text{ the number of non-* components of } s)\times$$
$$P(\bar{z}, s|\text{ net infected at t = 0 }) \quad (4)$$

We can do an iterative expansion of $P(\bar{z}, s|\text{net infected at t = 0 })$, as

$$P(\bar{z_1}, s_1|\text{ net infected at t = 0 })\times$$
$$P(\bar{z_2}, s_2|\bar{z_1}, s_1,\text{ net infected at t = 0 }) \times ...$$
$$P(\bar{z_N}, s_N|....,\text{ net infected at t = 0 })$$

where each of those terms is evaluated in terms of a Poisson processes as follows.

Use the network topology to figure out the set of edges exiting s_1 and write it as $C(1)$. Also define $\lambda(i, j)$ as the Poisson rate constant for an infectied message going from an infected node $v(i)$ to a non-infected node $v(j)$ and $v(j)$ making a transition to being infected when that message arrives. Also define $\lambda(1)$ as $\sum_{j\in C(1)} \lambda(1, j)$, i.e., the sum of the rate constants over all edges exiting $v(1)$. Then

$$P(s_2, \bar{z_2}|s_1.z_1 = 0) = \lambda(1)e^{-\lambda(1)(\bar{z_2}-\bar{z_1})} \times \frac{\lambda(1, 2)}{\lambda(1)} \quad (5)$$

as in the Gillespie algorithm – the expression on the RHS equals the probability that the first transition among all the nodes that are connected to s_1 occurs at the time $\bar{z_2}$, times the probability that it is node s_2 that makes that transition).

After cancellation we have

$$P(s_2, \bar{z_2}|s_1, \bar{z_1} = 0) = \lambda(1, 2)e^{-|C(1)|\lambda\cdot(z_{\{v(2)\}}-z_{\{v(1))\}}} \quad (6)$$

where λ is the sum of the (homogenous) infection message traffic rate.

Note that $\lambda(1, 2) = 0$ if there is no edge going from s_1 to s_2. If it were not for this fact, our formula for the conditional distribution $P(s_2, \bar{z_2}|s_1, \bar{z_1} = 0)$ would not be normalized.

Similarly, define $C(2)$ as the set of edges exiting either s_1 or s_2, and define $\lambda(2) = \sum_{j\in C(2)} \lambda(2, j)$, i.e., the sum of the rate constants over all edges exiting s_1 and s_2 (I'm pretty sure that the possibility of a node being connected to both $v(2)$ and $v(1)$ doesn't change the fact that this is the correct sum.) Then use our assumption of homogenous (infected node) rate constants to write

$$P(s_3, \bar{z_3}|s_2, \bar{z_2}...0) = K(3)\lambda e^{\lambda(2)(\bar{z_3}-\bar{z_2})} \quad (7)$$

where $K(3)$ equals 1 or 2, depending on whether under the network topology one or both of s_1 and s_2 are connected to s_3, and λ is the homogenous rate constant.

We can keep iterating to evaluate for all nodes that get infected. Then for nodes that do not get infected we simply use the CDFs whose rate is defined as above but the time is given by $T - z_M$.

Note that only a tiny fraction of the points in R^N are physically possible. E.g., we can't have a node v get infected at t, and then a node v" get infected at $t" > t$, and no other nodes ever get infected, if due to the network topology the only way v" can get infected is from v via a bottleneck node v' lying between v and v". This will be reflected in the likelihood function - all disallowed points in R^N will be have likelihood zero, and furthermore, the likelihood function will be properly normalized to account for the contorted shape of the set of allowed points.

3 Calculation notes

9) Unfortunately, due to the contorted shape of the subset of R^N of $\bar{z}$'s that are actually allowed (given the network topology) discussed above, I don't think we can do the sum-integral to give our likelihood in closed form. In fact, even if we fix $\{s\}$, I don't think we can do the associated integral in closed form. For the same reason, simple sampling MC with a uniform distribution over $[0, T]^m$ (where m is the number of nodes that get infected in $[0, T]$, specified by $\{s\}$) may be quite inefficient.

The new coordinate system may allow us to do our calculations much more simply. For example, now, for every vector $\{s_i\}$ in the set of allowable infection sequences, the integrand (in the new set of variables) is never zero for any of the $\bar{z}$'s, so even something like importance sampling MC, rather than MCMC, should be possible.

NEW on 2/22/14

So under simple sampling we are interested in approximating

$$\sum_M \sum_s \int d\bar{z} stuff(M, s, \bar{z}) =$$
$$N \sum_M q(M) V(M) \sum_s q(s|M)(T^M/M!) \int d\bar{z} q(\bar{z}|M) P(data|\bar{z}, s) P(s) P(\bar{z}|s) \tag{8}$$

where $V(M)$ is the number of infection sequences where M elemnts are not %, all three distributions q are uniform over the allowed ranges of their arguments, and "$stuff(M, s, z-bar)$" is the (normalized) product of likelihoods discussed above that gives $P(d, M, s, \bar{Z}|$net infected at t = 0).

To do simple sampling we would randomly sample those three distributions, proceeding from $q(M)$ to $q(s|M)$ to $q(\bar{z}|M)$. For each M we would then sum the

associated values of $stuff(M, s, \bar{z})$ for all pairs $(\bar{z}, s)$. We would then multiply that sum by $V(M)T^M/M!$ We would then sum all such products, and multiply by N, and be done.

Note that we can instead generate samples from a uniform distribution over infection sequences - we just need to use importance sampling (!). This would mean introducing the usual correction factors.

The proposal distribution over M that is implicit in uniform sampling over infection sequences is $V(M)/|V|$. So if we use that proposal distribution, we must multiply by $q(M)/[V(M)/V] = V/NV(M)$ in our integrand. Doing that cancels the $V(M)$ term on the RHS of Eq. **??**, divides by N, and multiplies by $|V|$. Nothing else changes.

4 (Uniform) Importance Sampling over $[0, T]$

The pseudo code is as follows:

- $V \leftarrow \{v_i\}$ (The set containing all nodes)
- $S \leftarrow$ set of all allowable infection orderings, including the orderings where some nodes did not get infected. For example an element $s \in S$ might be $\{v_2, *, *, *\}$
- $K \leftarrow 20,000$ (Number of samples of per infection sequence)
- $lhood \leftarrow 0$
- **FOR** $s \in S$ (For each ordering)
  - $plist \leftarrow []$ (Store all samples)
  - $M \leftarrow \text{len}(s)$ (The number of nodes that are infected)
  - $normconst \leftarrow \frac{T^M}{M!}$
  - **FOR** k in range(K)
    - $\bar{z} \leftarrow$ **sort**(**rand**($[0, T]^M$)) (Sample infection times)
    - $pz_{infected} \leftarrow P(\bar{z}, s|\text{net infected at } t = 0)$
    - $pdata \leftarrow P(d|z, s, \text{ net infected at } t = 0)$ (as in ASCII)
    - $ptotal \leftarrow pdata \cdot pz$
    - $plist$ **append** $ptotal$
  - $lhood += normcons * \textbf{mean}(plist)$
- **return** $lhood$

Note that we are not explicitely sampling the q in equation **??**. However, this can be easily changed to use a non-uniform importance sampling approach.

4.1 MCMC over $[0, \infty]$

- $s_0 \leftarrow$ **sample**(S) (Sample an ordering)
- $\bar{z}_0 \leftarrow$ **SORT**(**rand**$([0,T]^N)$) (Sample infection times)
- $K \leftarrow 500,000$ (Number of MCMC samples)
- $\sigma^2 \leftarrow 500$ (Set standard deviation of proposal)
- $probs \leftarrow []$ (List to store integrand values)
- for k in **range**(K)
  - $\bar{z}_1 \leftarrow \bar{z}_0 + \overrightarrow{\mathcal{N}(0, \sigma^2)}$ (Draw proposed infection time)
  - (The new ordering is implied by the draw of $\bar{z}_1$
  - **If** $\frac{P(\bar{z}_1, s_1 | \text{net infected})}{P(\bar{z}_0, s_0 | \text{net infected})} > \mathbf{U}(0,1)$ (MH STEP)
    - $s_0 \leftarrow s_1$ (Move to the new infection ordering)
    - $\bar{z}_0 \leftarrow \bar{z}_1$ (Move to the new infection times)
  - **else**
    - $s_0 \leftarrow s_0$ (Stay at previous infection ordering)
    - $\bar{z}_0 \leftarrow \bar{z}_0$ (Stay at previous infection times)
  - $probs$ **append** $P(d|\bar{z}_0, s_0, \text{net infected at } t = 0)$
- $lhood \leftarrow$ **mean**$(probs)$
- **return** $lhood$

This does not take into account sampling over s since as we discussed before, the last implementation did not satisfty detailed balance.