Estimation of Asymptomatic Population Based On Stochastic Epidemiological Modelling

EE691: Research and Developmental Project

Syomantak Chaudhuri, 170070004
Guide: **Prof. Vivek Borkar**

11 December 2020

Electrical Engineering Department
Indian Institute of Technology Bombay

Contents

Introduction

Aim

Aim: Estimation of asymptomatic population in an epidemic based on the data given of the symptomatic patients.
Given:

1. Number of new symptomatic patients each day
2. Number of recovered symptomatic patients each day
3. Total number of symptomatic patients (or initial symptomatic population)

Focus: Analyzing the performance of of our modeling technique.

Conventional SIR Epidemiological Model

Some assumptions of Kermack-McKendrick model

- Population size is fixed
- Instantaneous infection, no incubation period
- Homogeneous population

The model has 3 compartments

- S : the population susceptible to the epidemic
- I : the population which is currently infected
- R : the population recovered from the epidemic

The Dynamics of SIR

Figure 1: SIR Compartment Connectivity

$$
\begin{aligned}
\dot{S}(t) &= -\beta S(t) I(t)/N \\
\dot{I}(t) &= \beta S(t) I(t)/N - \gamma I(t) \\
\dot{R}(t) &= \gamma I(t)
\end{aligned}
$$

The Kermack-McKendrick model is deterministic. Stochastic modeling is more realistic but is *hard*!

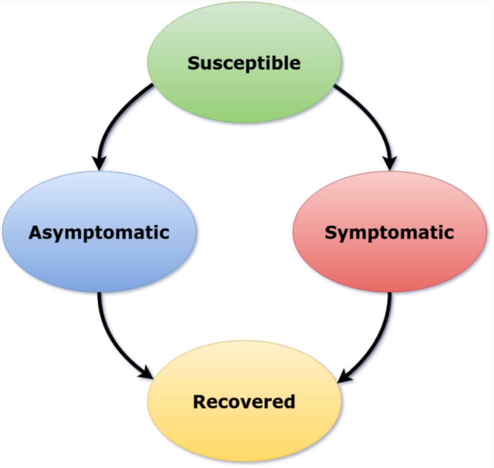
Methodology

The 4 compartment model

We propose a 4 compartment model - SIAR - for epidemics like COVID-19.

- **Compartment S (susceptible)** - at risk of contracting the virus.
- **Compartment I (symptomatic)** - contracted the virus and currently symptomatic.
- **Compartment A (asymptomatic)** - contracted the virus but asymptomatic.
- **Compartment R (recovered)** - recovered and not at risk of infecting others.

Connectivity of S,I,A,R



Dynamics of SIAR

$$\dot{S}(t) = -\lambda S(t)(\alpha I(t) + A(t))/N, \tag{1}$$

$$\dot{I}(t) = \lambda\mu S(t)(\alpha I(t) + A(t))/N - \gamma I(t), \tag{2}$$

$$\dot{A}(t) = \lambda(1-\mu)S(t)(\alpha I(t) + A(t))/N - \beta A(t), \tag{3}$$

$$\dot{R}(t) = \gamma I(t) + \beta A(t) \tag{4}$$

We take $\beta = \gamma$

Discretizations

For a compartment X on a particular day

- Divide each day into T intervals
- In each interval, a chunk of size $\Delta_X = X/D$ moves out of the compartment with probability P_X
- Chunk size and P_X set based on estimates of S, I, A, R at the beginning of the day
- $T = D$

What is P_X?

- P_X is set based on the differential equations.
- For compartment S, expected rate of outflux $\frac{\Delta S}{\Delta t} = P_S \Delta_S / T$.
- From the differential equations, $\frac{\Delta S}{\Delta t} = \lambda S(\alpha I + A)/N$.
- Get $P_S = \lambda(\alpha I + A)/N$.
- Similarly, $P_A = P_I = \gamma$, $P_R = 0$.
- Approximation: P_X and Δ_X calculated using S, I, A at the start of the day

Influx and Outflux

Consider the S compartment. On any day,

- Sampling T values of Bernoulli P_S random variables $\equiv$ Binomial(T, P_S).
- $D_S \sim Bin(T, P_S)$.
- Each of these D_S chunks move into I compartment with probability μ and otherwise into A compartment.
- Let D_{SI},D_{SA} denote number of chunks moving into I and A compartments from S respectively.
- $D_{SI} \sim Bin(D_S, \mu)$, $D_{SA} = D_S - D_{SI}$.
- Similarly, $D_{IR} \sim Bin(T, P_I)$ and $D_{AR} \sim Bin(T, P_A)$.

Tracking the population

We can update the population based on the random variables D.

$$S_{n+1} = S_n - \Delta_S^n D_S^n \tag{5}$$

$$I_{n+1} = I_n - \Delta_I^n D_{IR}^n + \Delta_S^n D_{SI}^n \tag{6}$$

$$A_{n+1} = A_n - \Delta_A^n D_{AR}^n + \Delta_S^n D_{SA}^n \tag{7}$$

$$R_{n+1} = R_n + \Delta_I^n D_{IR}^n + \Delta_A^n D_{AR}^n \tag{8}$$

Notation

- Let the given data be for N days. Let us represent the given data as $X^N = (X_0, X_1, .., X_{N-1})$ and each X_i has 2 components - SI transitions ($X_{i,0}$) and IR transitions ($X_{i,1}$) between day i to $i+1$.
- $X^k := \{X_0, X_1, .., X_{k-1}\}$
- Let Z^N be the trajectory of the hidden variables, ie, $Z_n = \{D^n_{SA}, D^n_{AR}\}$ for $n = 0, .., N-1$.
- Z^N, X^N together define the entire trajectory completely (initial condition implicit).
- Let $(\lambda, \alpha, \mu, \gamma)$ be $\vec{\theta} \in [0,1]^4$

Progressing Along the Trajectory

On day i, given X^i, Z^i,

- We have some estimates of $\hat{S}_i$ and $\hat{A}_i$ (and exact I_i).
- Get corresponding chunk sizes Δ_S^i, Δ_I^i, and Δ_A^i and probabilities.
- Trajectory of S and A unknown
- Z^N, X^N data equivalent to S, I, A evolution.
- For EM algorithm, we can use Z^N as hidden variable.

SAEM Algorithm

Algorithm 1: SAEM

random initialize $\vec{\theta}^{(0)}$;

zero initialize $Q^{(0)}(\vec{\theta})$;

while *Not Converged* **do**

$z^N \sim P(Z^N | X^N, \vec{\theta}^{(t)})$;

$Q^{(t)}(\vec{\theta}) \leftarrow (1-\gamma_t) Q^{(t-1)}(\vec{\theta}) + \gamma_t \log(P(z^N, X^N | \vec{\theta}))$;

$\vec{\theta}^{(t+1)} \leftarrow \vec{\theta}^{(t)} + \eta_t \nabla_{\vec{\theta}} Q^{(t)}(\vec{\theta})$;

end

Log-likelihood Calculation Step - SAEM

Given X^N and a sample Z^N,

$$\log P(Z^N, X^N|\vec{\theta}) = \sum_{i=0}^{N-1} \log(P(Z_i, X_i|Z^i, X^i, \vec{\theta}))$$

$$P(Z_i, X_i|Z^i, X^i, \vec{\theta}) = P_{i1}(d_{SA}, d_{SI})P_{i2}(d_{SA}, d_{SI})P_{i3}(d_{IR})P_{i4}(d_{AR})$$

$$P_{i1} := \binom{T}{d_{SA} + d_{SI}} (P_S)^{d_{SA}+d_{SI}} (1 - P_S)^{T-d_{SI}-d_{SA}} \tag{9}$$

$$P_{i2} := \binom{d_{SI} + d_{SA}}{d_{SI}} (\mu)^{d_{SI}} (1 - \mu)^{d_{SA}} \tag{10}$$

$$P_{i3} := \binom{T}{d_{IR}} P_I^{d_{IR}} (1 - P_I)^{T-d_{IR}} \tag{11}$$

$$P_{i4} := \binom{T}{d_{AR}} (P_A)^{d_{AR}} (1 - P_A)^{T-d_{AR}} \tag{12}$$

Log-likelihood Calculation Step - SAEM (Cont.)

As T is large, we can approximate these by

$$\log(P_{i1}) \approx -T\,\mathrm{D}\left(\frac{d_{SA}+d_{SI}}{T}\middle\|P_S\right) \tag{13}$$

$$\log(P_{i2}) \approx -(d_{SA}+d_{SI})\,\mathrm{D}\left(\frac{d_{SI}}{d_{SA}+d_{SI}}\middle\|\mu\right) \tag{14}$$

$$\log(P_{i3}) \approx -T\,\mathrm{D}\left(\frac{d_{IR}}{T}\middle\|P_I\right) \tag{15}$$

$$\log(P_{i4}) \approx -T\,\mathrm{D}\left(\frac{d_{AR}}{T}\middle\|P_A\right) \tag{16}$$

Sampling Step - SAEM

In the sampling step of Stochastic Approximation variant of EM algorithm, we need to draw samples z^N from a rather complex distribution

$$z^N \sim \tilde{P}(Z^N|X^N, \vec{\theta}^{(t)})$$

$$P(Z^N|X^N, \vec{\theta}^{(t)}) = \prod_{i=0}^{N-1} P(Z_i|Z^i, X^N, \vec{\theta}^{(t)}) \tag{17}$$

$P(Z_i|Z^i, X^N, \vec{\theta}^{(t)})$ is quite complex to compute as the future X_n values can affect the distribution. Since the trajectories can be quite long, this relation is not feasible to analyze. We instead make an approximation -

$$P(Z_i|Z^i, X^N, \vec{\theta}^{(t)}) \approx P(Z_i|Z^i, X^{i+1}, \vec{\theta}^{(t)}) \tag{18}$$

Sampling Step - SAEM (Cont.)

We can now easily sample from the approximate distribution. Define

$$P_c^i := \frac{P_S^i(1-\mu^{(t)})}{1-\mu^{(t)}P_S^i}$$

Then for random variables $Z_{i,0} = D_{SA}^i$ and $Z_{i,1} = D_{AR}^i$, we can write the probabilities as (dropping the sub/super-script i for the RHS terms)

$$\begin{aligned}
P(Z_i|Z^i, X^{i+1}, \vec{\theta}^{(t)}) &= P_{i1}(d_{SA})P_{i4}(d_{AR}) \\
P_{i1}(d_{SA}) &:= \binom{T-d_{SI}}{d_{SA}} (P_c)^{d_{SA}} (1-P_c)^{T-d_{SI}-d_{SA}} \qquad (19) \\
P_{i4}(d_{AR}) &:= \binom{T}{d_{AR}} (P_A)^{d_{AR}} (1-P_A)^{T-d_{AR}} \qquad (20)
\end{aligned}$$

Sampling Step - SAEM (Cont.)

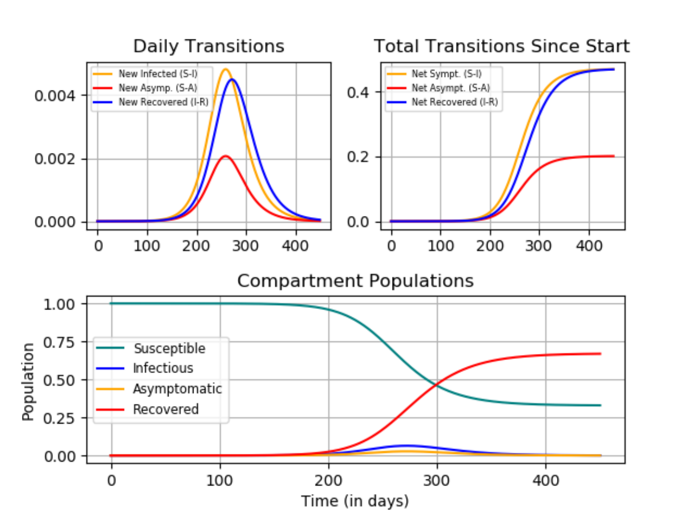
In all, the basic sampling strategy described below

- At any day i, get the P_X and Δ_X values based on Z^i, X^{i+1}.
- Get Z_i by independently sampling D^i_{SA} and D^i_{AR} based on the binomial distribution.
- Move ahead in the trajectory to day $i+1$.

Results

The Dataset

Gsenerated the data based on the differential equations 1-4 with $\lambda = 0.2$, $\alpha = 0.4$, $\mu = 0.7$, and $\gamma = 0.07$. The initial condition used was $S = 10^6$, $I = 0$, $A = 1$, and $R = 0$.



Initial Condition and Initialisation

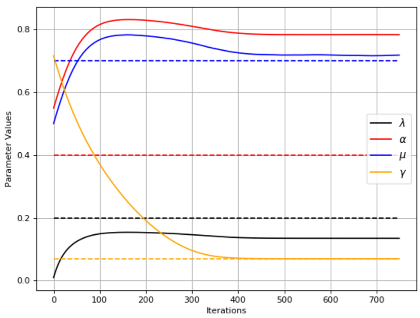
For the initial condition,

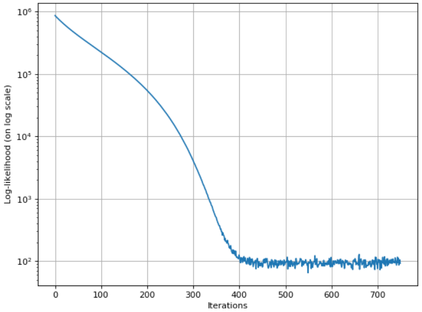
- P_S, P_C extremely small at the beginning and the end of the epidemic when I value is less.
- $P_S = \lambda(\alpha I + A)/N$
- During sampling $D_{SA} = 0$ always because of this
- Overcame this by initialising the technique with day 220 as $t = 0$ and running till day 310.

For the initialisation of parameters,

- Bad initialisation causes error due to distribution approximation
- $mu^{(0)} \approx 1$ or $\lambda^{(0)} \approx 0$ can fix this

While γ and μ converge to their true values, α and μ converge to other values.

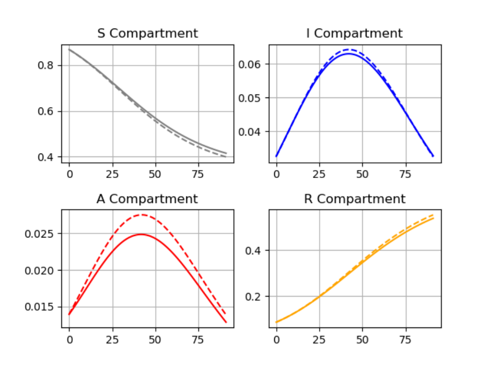




Results

$$RMSfE(\hat{X}^H) = \sqrt{\frac{1}{H}\sum_{t=1}^{H}\left(\frac{X_t - \hat{X}_t}{X_t}\right)^2}$$

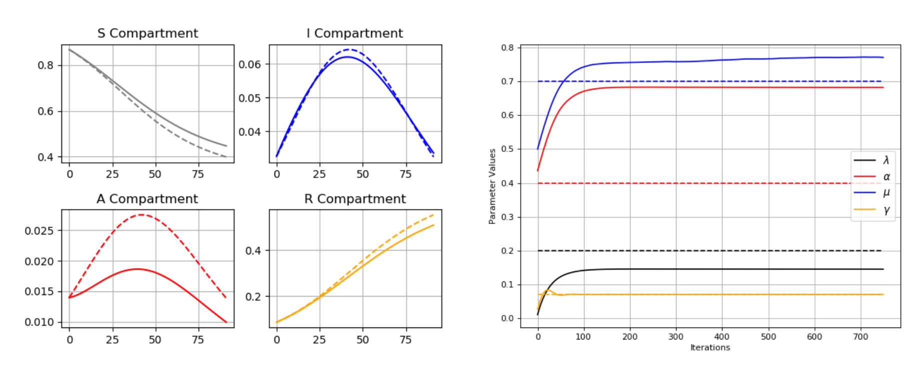
$$NCC(\hat{X}^H) = \frac{\sum_{t=1}^{H}(X_t - \bar{X})(\hat{X}_t - \bar{\hat{X}})}{\left\|X - \bar{X}\right\|_2 \left\|\hat{X} - \bar{\hat{X}}\right\|_2}$$



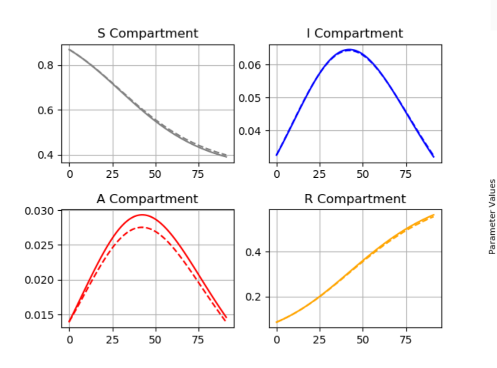
	RMSfE (in %)	NCC (in %)
S	2.4	99.9
I	1.2	99.9
A	8.2	99.7
R	2.1	99.9

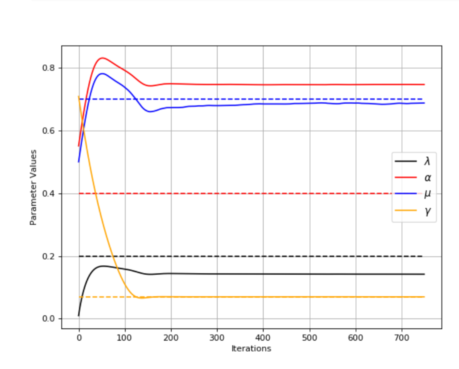
Initialisation Sensitivity

The technique is sensitive to the initial condition and this can be attributed to the the EM algorithm.



Initialisation Sensitivity (Cont.)

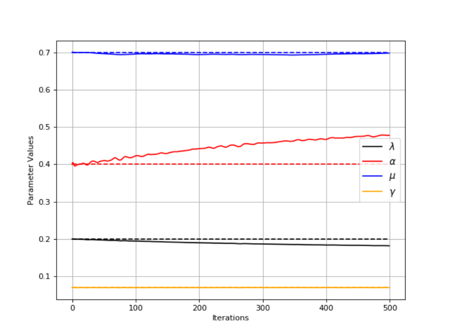


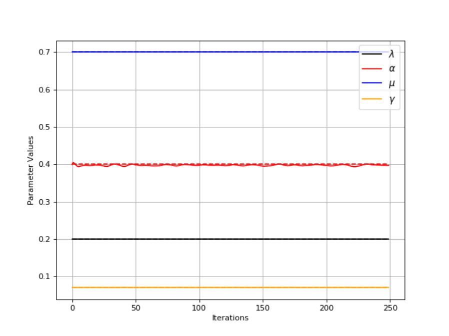


Optimization Starting At True Value

Initialising the parameters with their true values and optimizing -

- Slight variation in the other parameters cause a large change in α.





Conclusion

Explaining The Wrong Convergence

- For the dataset used, at any time t, $I/A \approx \mu/(1-\mu)$.
- $\dot{S} = SI\lambda\frac{\alpha\mu+1-\mu}{\mu}$
- For all the solutions obtained, $\lambda\frac{\alpha\mu+1-\mu}{\mu}$ was nearly the true value of 0.166.
- Can it be fixed by generating 3 datasets with different initial condition and simultaneously using all 3 for training?
- Limitations of EM algorithm

Further Remarks

- Is out initial condition really practical?
- Possible solution can be to use varying T, D based on the population in I compartment and estimated S compartment population.
- Instead of discretization, use Beta distribution to get fraction of compartment which moves out.

The End

Thank you!

Bibliography

- Agrawal, M., Kanitkar, M. & Vidyasagar, M. (n.d.). Modelling the spread of sars-cov-2 pandemic-impact of lockdowns & interventions. *The Indian journal of medical research.*
- Bagal, D. K., Rath, A., Barua, A. & Patnaik, D. (2020). Estimating the parameters of susceptible-infected-recovered model of covid-19 cases in india during lockdown periods. *Chaos, Solitons Fractals*, *140*, 110154. *https://doi.org/https://doi.org/10.1016/j.chaos.2020.110154*
- Cappé, O. & Moulines, E. (2009). On-line expectation–maximization algorithm for latent data models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, *71*(3), 593–613. *https://doi.org/https://doi.org/10.1111/j.1467-9868.2009.00698.x*

Bibliography (cont.)

- Dan, J. M., Mateus, J., Kato, Y., Hastie, K. M., Faliti, C. E., Ramirez, S. I., Frazier, A., Yu, E. D., Grifoni, A., Rawlings, S. A., Peters, B., Krammer, F., Simon, V., Saphire, E. O., Smith, D. M., Weiskopf, D., Sette, A. & Crotty, S. (2020). Immunological memory to sars-cov-2 assessed for greater than six months after infection. *bioRxiv*. `https://doi.org/10.1101/2020.11.15.383323`
- Dempster, A. P., Laird, N. M. & Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, *39*(1), 1–22. `https://doi.org/https://doi.org/10.1111/j.2517-6161.1977.tb01600.x`
- Greenwood, P. E. & Gordillo, L. F. (2009). Stochastic epidemic modeling. In G. Chowell, J. M. Hyman, L. M. A. Bettencourt & C. Castillo-Chavez (Eds.), *Mathematical and statistical estimation approaches in epidemiology* (pp. 31–52). Springer Netherlands. `https://doi.org/10.1007/978-90-481-2313-1_2`

Bibliography (cont.)

- Jank, W. (2006). Implementing and diagnosing the stochastic approximation em algorithm. *Journal of Computational and Graphical Statistics*, *15*(4), 803–829. `https://doi.org/10.1198/106186006X157469`
- Kermack, W. O. & McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. *Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character*, *115*(772), 700–721.
- Kingma, D. & Ba, J. (2014). Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

Bibliography (cont.)

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., … Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox & R. Garnett (Eds.), *Advances in neural information processing systems 32* (pp. 8024–8035). Curran Associates, Inc. *http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf*