.

Appendix for "Improving Parameter Estimation of Epidemic Models: Likelihood Functions and Kalman Filtering"

Tianyi Li *Sloan School of Management, MIT; DSME, CUHK Business School*

Hazhir Rahmandad *Sloan School of Management, MIT*

John Sterman *Sloan School of Management, MIT*

A: Covariance Matrices for Kalman Filtering

For data variance, consider standard Gaussian measurement noise around signal x: $x_d \sim N(x, \sigma_d^2)$. Assume $x_d = x(1 + n_D \mathrm{U})$ where U is uniform white noise on [-0.5,0.5]. So data variance:

$$\sigma_d^2 = x^2 n_D^2 \sigma_U^2 = x^2 n_D^2 \cdot 0.0833, \tag{1}$$

where 0.0833 is the variance of U. n_D is to be estimated; x corresponds to measured variables: ER, IR, or RR. This is for the variance-scaled case; for the variance-unscaled case, $\sigma_d^2 = c^2 n_D^2 \cdot 0.0833$, where constant c is derived from prior tests, similar to the formulation in GS (see *Scaling of variance*).

For drive variance, variances on stocks come from the driving noise in the three flow terms ER, IR, RR. The stochastic system is:

$$\begin{cases} \dot{S}(t) = -ER + \epsilon_{ER} \\ \dot{E}(t) = ER - IR + \epsilon_{ER} + \epsilon_{IR} \\ \dot{I}(t) = IR - RR + \epsilon_{IR} + \epsilon_{RR} \\ \dot{R}(t) = -RR + \epsilon_{RR}. \end{cases} \tag{2}$$

We investigate two options for the specific noise structure. Again, x corresponds to measured variables: ER, IR, or RR.

(A) The noise structure is inherited from the data generation process: $Poisson[x(1 + n_R \mathrm{AU})]$ where AU is the autocorrelated noise constructed from U. So drive variance:

$$\sigma_p^2 = Var(Poisson[x(1 + n_R \mathrm{AU})]) = x(1 + n_R \mathrm{AU}). \tag{3}$$

n_R is to be estimated. Note we use uniform AU while set the amplitude open to vary the noise level.

(B) Assume we inherit no knowledge from the data generation process and consider standard Gaussian process noise. Similar to the Gaussian measurement noise, we have $x(1 + n_R \mathrm{U})$. So drive variance

$$\sigma_p^2 = x^2 n_R^2 \sigma_U^2 = x^2 n_R^2 \cdot 0.0833. \tag{4}$$

n_R is to be estimated. Note here we may alternatively assume AU instead of U, i.e., autocorrelated Gaussian process noise (still different from Option (A) as no Poisson process is assumed). In this case only $\sigma_U^2 \to \sigma_{AU}^2$; nothing changes besides a constant.

In Kalman filtering, the ratio of data variance to drive variance matters in determining the Kalman gain.

For Option (B),

$$\frac{\sigma_d^2}{\sigma_p^2} = \frac{x^2 n_D^2 \sigma_U^2}{x^2 n_R^2 \sigma_U^2} = (\frac{n_D}{n_R})^2 .. \tag{5}$$

Only one degree of freedom exists, so we only need to estimate either n_D or n_R. We estimate n_D in our experiments.

For Option (A),

$$\frac{\sigma_d^2}{\sigma_p^2} = \frac{x^2 n_D^2 \sigma_U^2}{x(1 + n_R \mathrm{AU})}, \tag{6}$$

which could not break down to a function of n_D/n_R, so we need to estimate both n_D and n_R.

Variances on the four stocks are linked to variances on the flows ($\sigma_p^2 \to \sigma_{ER/IR/RR}^2$):

$$\begin{cases} \Sigma_S = \sigma_{ER}^2 (\Delta t)^2 \\ \Sigma_E = (\sigma_{ER}^2 + \sigma_{IR}^2)(\Delta t)^2 \\ \Sigma_I = (\sigma_{IR}^2 + \sigma_{RR}^2)(\Delta t)^2 \\ \Sigma_R = \sigma_{RR}^2 (\Delta t)^2 \end{cases} \tag{7}$$

where Δt is the simulation time step. The four terms are the diagonal elements of Σ_x. For the covariance between stocks, there is $corr(S, E) = corr(E, I) = corr(I, R) = 1$ since the driving noises come from the flow terms. Thus the following off-diagonal elements in Σ_x are non-zero:

$$\begin{cases} \Sigma_{SE} = \Sigma_{ES} = \sigma_{ER}^2 (\Delta t)^2 \\ \Sigma_{EI} = \Sigma_{IE} = \sigma_{IR}^2 (\Delta t)^2 \\ \Sigma_{IR} = \Sigma_{RI} = \sigma_{RR}^2 (\Delta t)^2. \end{cases} \tag{8}$$

Initial drive variances $\Sigma_{S/E/I/R} = \sigma_0$ are set at a relatively large value (1E6) in order that model series are drawn close to data relatively quickly. Initial off-diagonal elements of Σ_x are 0.

For Option (A), we need to include three extra stocks to the model to account for noise autocorrelation. The drive variances of these noise stocks are exogenous. There is no covariance between these new stocks with the four original stocks, or between themselves.

Gaussian log-likelihood function under multi-variate Kalman Filtering

For multi-variate Kalman Filtering, the Gaussian log-likelihood function (10) is amplified to the matrix form. Noting (13) and (14), the observed variables $\boldsymbol{y} = \boldsymbol{D}\boldsymbol{x} + \Sigma_y$ follow the multi-Gaussian distribution, i.e., $\boldsymbol{y}_d \sim N(\boldsymbol{y}, \boldsymbol{\Sigma}_d)$, where the covariance $\boldsymbol{\Sigma}_d = \boldsymbol{D}\hat{\boldsymbol{\Delta}}\boldsymbol{D}^T + \Sigma_y$. Therefore, for each data point along the time series, the multivariate likelihood function is:

$$\begin{aligned} -\frac{1}{2}\left(\frac{x - x_d}{\sigma_d}\right)^2 - \log(\sigma_d) &\Rightarrow -(\boldsymbol{y} - \boldsymbol{y}_d)\Sigma_d^{-1}(\boldsymbol{y} - \boldsymbol{y}_d)^T - \log(|\boldsymbol{\Sigma}_d|) \\ &\Rightarrow -(\boldsymbol{D}\boldsymbol{x} - \boldsymbol{D}\boldsymbol{x}_d)\Sigma_d^{-1}(\boldsymbol{D}\boldsymbol{x} - \boldsymbol{D}\boldsymbol{x}_d)^T - \log(|\boldsymbol{\Sigma}_d|). \end{aligned} \tag{9}$$

Payoffs are accumulated along the time series, with $\boldsymbol{x}$ (thus $\boldsymbol{y}$) and $\hat{\boldsymbol{\Delta}}$ (thus $\boldsymbol{\Sigma}_d$) updated at each t.

B: Different Strength of Noise Autocorrelation

At Stage 4, we generate very noisy data series and use them in estimation. The noise level is drastically amplified ($n_{D/R} = 2$), and the series show low-frequency but high-amplitude fluctuations around the signals (Figure S1). The noisiness of our synthetic data may exceed the noise level of actual epidemics datasets (e.g., Figure S5). We further tested different strength of noise autocorrelation in the data: noise correlation time is 2, 5 (for the results in Table 3 and Figure 9), or 8 days. Tests show that the estimation performance of studied schemes largely converges and is generally invariant to this change (Table S1; sensitivity analysis results when noise correlation time = 2 or 8 are almost identical to Figure 9 and are not shown).

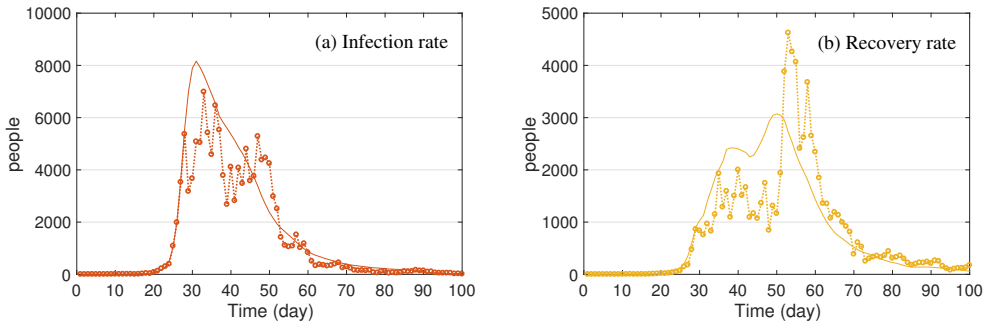

Figure S1: Sample noise level of synthetic data used in Stage 4 experiments. Showing the true signals (solid) and signals with noise (dotted) on (a) infection rate IR and (b) removal rate RR.

Est. Scheme	c	λ_E	λ_I
Stage 4: Noise correlation time: 2 days			
#GS	5.124 (0.790)	9.802 (2.272)	**10.279** (2.042)
#NB	5.135 (0.846)	10.084 (2.010)	10.694 (1.991)
#KFSA	**5.054** (2.669)	**9.988** (5.551)	12.043 (5.475)
#KFSB	5.455 (1.556)	9.062 (4.404)	11.172 (4.996)
Stage 4: Noise correlation time: 5 days			
#GS	5.174 (0.995)	**10.021** (4.151)	**10.955** (3.576)
#NB	**5.121** (1.037)	10.707 (3.375)	11.320 (3.408)
#KFSA	5.446 (2.736)	8.270 (5.000)	11.429 (5.186)
#KFSB	5.622 (2.041)	9.248 (5.474)	11.888 (6.022)
Stage 4: Noise correlation time: 8 days			
#GS	5.269 (1.126)	**9.940** (4.320)	**10.504** (3.924)
#NB	5.200 (1.058)	10.365 (3.694)	11.329 (3.739)
#KFSA	**5.022** (2.694)	8.861 (4.962)	11.570 (4.947)
#KFSB	5.520 (1.894)	9.993 (5.613)	11.723 (6.427)

Table S1: Estimation performance of selected schemes under different levels of noise autocorrelation in the data (Stage 4). Noise correlation time is 2, 5, or 8 days. Best results at each stage are shown in bold fonts.

C: Uncorrelated Measurement Noise

For sanity checks, we replicate Stage 4 experiments with a new dataset where the measurement noise is assumed to be uncorrelated, i.e., U instead of AU in (4), which could still be viewed as correlated noise but with infinitesimal correlation time. The other conditions are the same as in Stage 4 (e.g., noise correlation time = 5 days). The performance of investigated schemes is in general invariant to this change (Table S2 and Table 3), with KFSB performing slightly better this time, since now the noise structure is closer to its standard multivariate i.i.d. Gaussian buildup.

Est. Scheme	c	λ_E	λ_I
Stage 4: Uncorrelated measurement noise			
#GS	5.114 (0.964)	10.176 (3.767)	10.815 (2.596)
#NB	**5.106** (1.029)	10.703 (3.254)	11.226 (2.976)
#KFSA	4.643 (2.761)	9.599 (5.592)	11.248 (5.104)
#KFSB	5.338 (1.563)	**9.983** (5.182)	**10.760** (4,482)

Table S2: Estimation performance of selected schemes with uncorrelated measurement noise (Stage 4 replica).

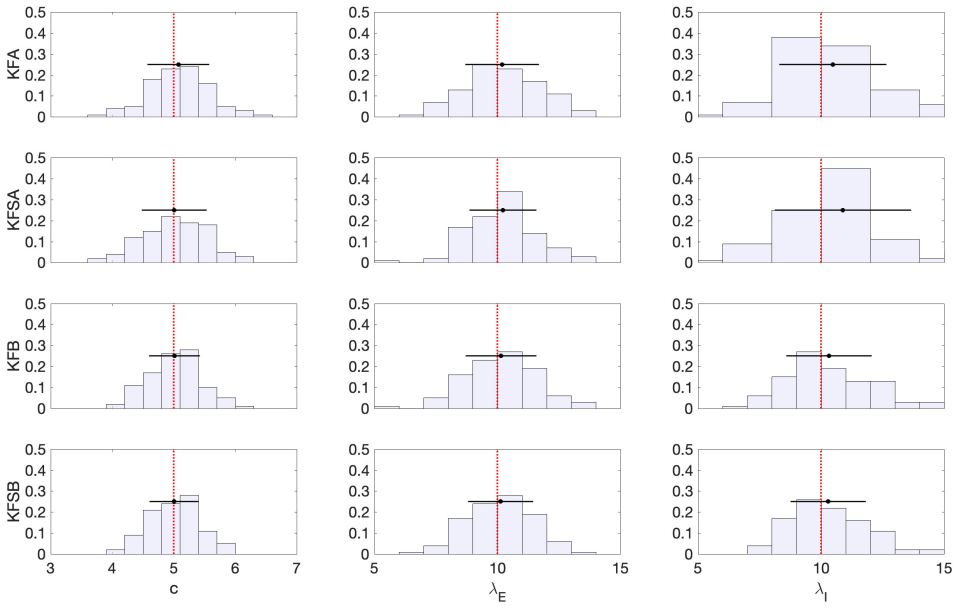

Figure S2: Estimation performance at Stage 2. Histograms of 100 instances of parameter estimates. ground-truth values are shown in red bars. Black dots/bars show mean/standard deviation of estimates over all instances.

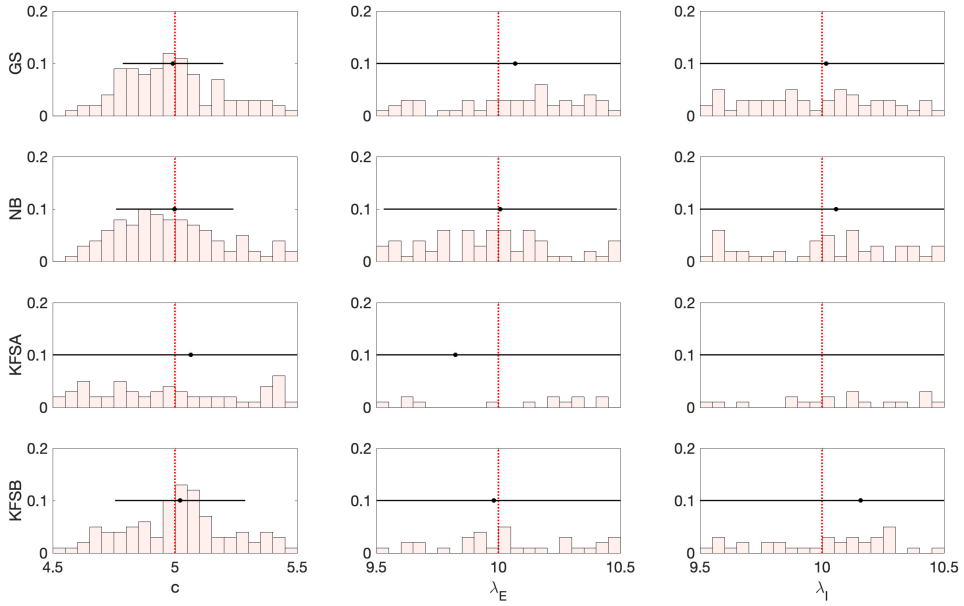

Figure S3: Estimation performance at Stage 3 (first scenario). Histograms of 100 instances of parameter estimates. ground-truth values are shown in red bars. Black dots/bars show mean/standard deviation of estimates over all instances.

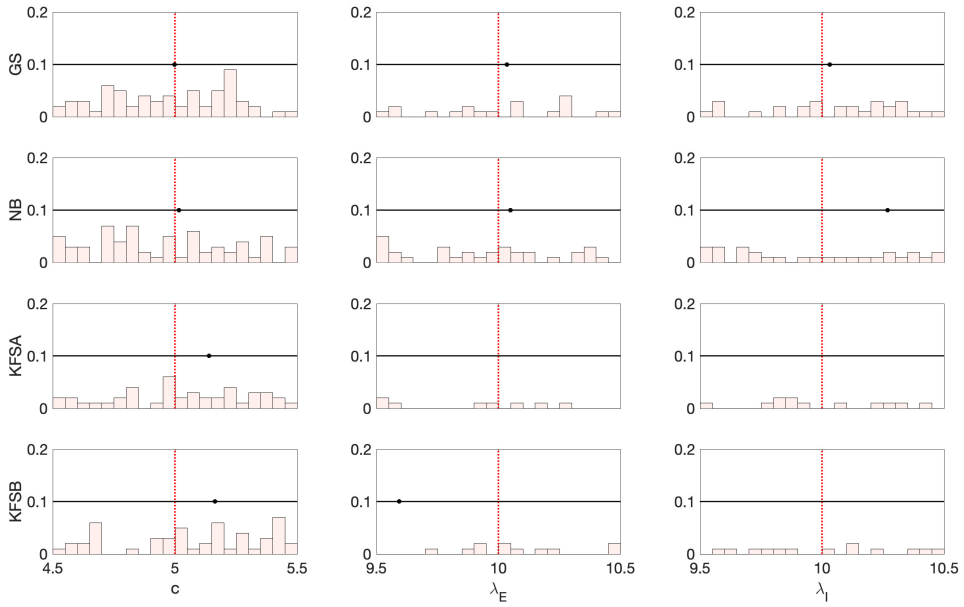

Figure S4: Estimation performance at Stage 4. Histograms of 100 instances of parameter estimates. ground-truth values are shown in red bars. Black dots/bars show mean/standard deviation of estimates over all instances.

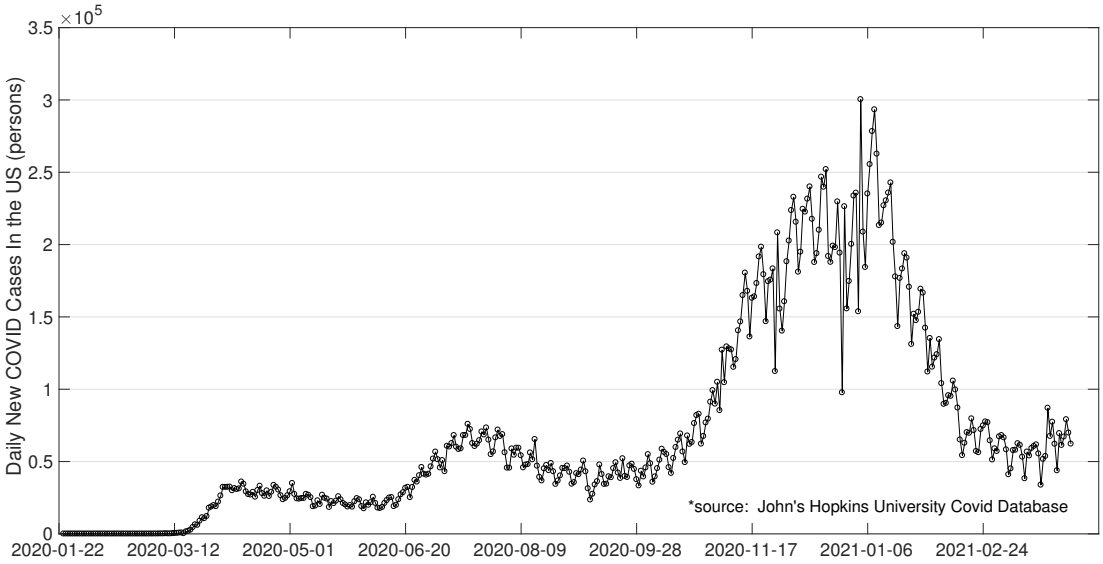

Figure S5: Daily new COVID cases in the US (2020-01-22 to 2021-04-03). Source: John's Hopkins University COVID database.

D: Details of Model Setup and Experimental Procedures

D1: Data generation Model

The model is used to generate simulated time series that serve as the datasets for the followed model estimation. Stochasticity is added to the model through white noise, autocorrelated noise and the Poisson process (see main text).

List of model parameters

For noise amplitudes, the large the value is, the noisier the generated time series are. In Stage 1-3 experiments, we use an amplitude 0.5; in Stage 4 experiments, we use an amplitude 2 to generate very noisy data (see Figure S1).

Parameter	Unit	Value	Category
initial susceptible (S0)	people	100000-2	initial condition
initial exposed (E0)	people	2	initial condition
initial infected (I0)	people	0	initial condition
initial recovered (R0)	people	0	initial condition
contact rate	1/day	(estimated)	epidemiological
infectivity	dimensionless	0.1	epidemiological
incubation period	day	(estimated)	epidemiological
infection period	day	(estimated)	epidemiological
Noise amplitude ER	dimensionless	>0 (set by modeler)	noise structure
Noise amplitude IR	dimensionless	>0 (set by modeler)	noise structure
Noise amplitude RR	dimensionless	>0 (set by modeler)	noise structure
Noise amplitude ED	dimensionless	>0 (set by modeler)	noise structure
Noise amplitude ID	dimensionless	>0 (set by modeler)	noise structure
Noise amplitude RD	dimensionless	>0 (set by modeler)	noise structure
Autocorrelated Noise STD	dimensionless	(1/24)^0.5	noise structure
Noise correlation time	day	2/5/8	noise structure
TIME STEP	day	0.125	system
SAVE PER	day	1	system

Table S1: Parameters of data generation model

Model equations

Key model equations are explained in the main text. Extra equations used in the model:

$$\text{Total population } P = S + E + I + R = S0 + E0 \tag{10}$$

White noise (i.e., Noise W in the Vensim model, U in the main text) is uniform over [-0.5, 0.5]; the autocorrelated noise (i.e., Noise P in the Vensim model, AU in the main text) is constructed from the white noise:

$$\begin{aligned} \frac{dNoise\ P}{dt} &= \frac{Noise\ W - Noise\ P}{Noise\ correlation\ time} \\ Noise\ W &= Noise\ P\ STD(24\frac{Noise\ correlation\ time}{TIME\ STEP})^{0.5} * U[-0.5, 0.5] \end{aligned} \tag{11}$$

Reported Datasets

In the model, we maintain the options of reporting either the flow data (new cases on each day) or the stock data (cumulative cases on each day). For each option, we have three scenarios of data availability: perfect reporting, medium reporting and poor reporting.

Type	Availability	Model Variables
stock	perfect: E_d, I_d, R_d	Exposed Data, Infected Data, Removed Data
	medium: I_d, R_d	Infected Data, Removed Data
	poor: I_d	Infected Data
flow	perfect:ER_d, IR_d, RR_d	Exposure Rate Data, Infection Rate Data, Removal Rate Data
	medium: IR_d, RR_d	Infection Rate Data, Removal Rate Data
	poor: IR_d	Infection Rate Data

Table S2: Data reporting options

D2: Estimation Model

The model is used to conduct estimation experiments, using simulated datasets generated from the data generation model. Multiple likelihoods (G, GS, GL, P, NB), two types of reported data (stock data & flow data) and four variants of Kalman filtering are opted in the estimation scheme.

List of model parameters

Parameter	Unit	Value	Usage
initial variance (S,E,I,R)	people^2	1e6	Kalman filtering
initial variance (Noise stocks)	dimensionless	1e6	Kalman filtering
scaling constant - stock	people	1000	unscaled Gaussian
scaling constant - flow	people/day	1000	unscaled Gaussian
min variance (σ_{min}) - S	people^2	~100	scaled Gaussian
min variance (σ_{min}) - F	(people/day)^2	~10000	scaled Gaussian
r	dimensionless	(estimated)	negative binomial
Noise amplitude D (n_D)	dimensionless	(estimated)	Kalman filtering
Noise amplitude R (n_R)	dimensionless	(estimated)	Kalman filtering

Table S3: Parameters of estimation model

Initial drive variances of stock variables are required in Kalman filtering. A large value is often used to quickly drift the model towards the data. A scaling constant (discussed in the main text) determined from a priori runs is used for the variance-unscaled Gaussian likelihoods. For variance-scaled Gaussian, a minimum variance σ_{min} is used (discussed in the main text). Values of these two system parameters are different for using stock data or flow data in estimation and are determined from prior runs. r is used in negative binomial likelihoods and is estimated in the experiments; results suggest that a value around 0.1 is desirable. Two noise levels n_D and n_R are estimated.

Note that conceptually, a σ_{min} also applies to constant-variance Gaussian likelihoods to prevent corner solutions. The lower bound is embodied in the search range of n_D during optimization, whose minimum value $\underline{n_D}$ ensures that $\underline{n_D}\sigma_0 > \sigma_{min}$. In the experiments we make sure that this boundary value is not so small as to be falsely chosen by the optimization engine.

Model equations

Model formulations are explained in the main text. Different payoff functions are constructed manually (i.e., as policy payoffs) instead of using default Vensim formulations, for different likelihoods (G stands for Gaussian, S stands for variance-scaling, P stands for Poisson, NB stands for negative binomial), for the usage of either flow data (noted by F in variable names) or stock data (noted by S in variable names), and for each data-availability condition (noted by EIR, IR or I in variable names; EIR stands for ER_d, IR_d, RR_d, IR stands for IR_d, RR_d, and I stands for IR_d only). Kalman filtering formulations are explained in the Appendix, which have two options in the process noise structure.

Overall, for each estimation scheme, we specify the following options:

Feature of the estimation scheme	Options		
data-availability condition	EIR	IR	I
type of datasets	F	S	
type of likelihood function	G	P	NB
log transform	Y	N	
scaling of variance	Y	N	
Kalman filtering	A	B	

Table S4: Options of the estimation scheme

Optimization

Three model parameters (c, D_E, D_I) and certain auxiliary parameters (r, n_D, n_R) are estimated. The upper bound and lower bound of the search range are determined at proper values, centering at the ground-true values. Multiple initial parameter values are tested and estimation results are largely invariant to initial search points.

Estimated Parameter	Lower Bound	Upper Bound	Initial Value	Ground truth
contact rate (c)	1	20	3	5
incubation period (D_E)	1	20	5	10
infection period (D_I)	1	20	5	10
r	0.1	10	2	-
Noise amplitude D (n_D)	0	20	0.1	-
Noise amplitude R (n_R)	0	20	0.1	-

Table S5: Optimization parameters

Tests suggest that the following system parameters (for two optimizers, respectively) ensure the convergence of results with a relatively economical computational burden.

$$\begin{aligned} &Vensim\ Powell\ optimizer: RESTART\ MAX = 10(with\ Kalman\ Filtering)/40(no\ Kalman\ Filtering) \\ &Vensim\ MCMC\ optimizer: MCLIMIT = 5000, MCBURNIN = 3000 \end{aligned} \tag{12}$$

The convergence of MCMC chains can be indicated by the potential scale reduction factor (PSRF), and in particular its multivariate version, although the metrics do not always guarantee convergence. Typically, a PSRF near 1 suggests the convergence of MCMC, and values larger than 1.2 signify that more time is required to converge to steady state. Confirming the convergence of MCMC chains is key to reliable estimation.

D3: Sample Results from Gaussian Estimation Schemes

Sample parameter estimates from four estimation schemes (Figure S6): least squares on stocks, least squares on flows, Gaussian with log transform, and variance-scaled Gaussian. The aggregated absolute difference with respect to the ground truth ($||\Delta||_1$, summing over all three parameters) is reported for each scheme. At each row, the best result is in bold font.

Parameter	Ground truth	least sq. (stock)	least sq. (flow)	log trans.	var-scaled
c	**5**	5.57	5.34	**5.32**	5.42
D_E	**10**	12.27	10.78	10.87	**9.76**
D_I	**10**	9.72	11.30	10.77	**10.11**
$\|\|\Delta\|\|_1$	**0**	3.12	2.42	1.96	**0.77**

Figure S6: Sample Results from Gaussian Estimation Schemes.

D4: Notes on the Negative Binomial Estimator

When using the negative binomial estimator, the choice of r needs some special attention. Look closely at the likelihood function (bottom row of equation (10) in the main text): at around $r = x/e$, the U-shape likelihood function reaches its trough. Therefore, for a series of different x, to keep uniform monotonicity, one chooses a small rather than a large value for r. This is consistent with its definition as the number of cases missing in the reported data (so r is not supposed to be as large as in the higher end of a group of U-curves). However, when we let r be small, the likelihood function is approximated to:

$$x_d log(\frac{x}{x+r}) + r log(\frac{r}{x+r}) + log(\Gamma(x_d + r)) - log(\Gamma(x_d)) - log(\Gamma(r)) \Longrightarrow r log(\frac{r}{x}) - 1. \tag{13}$$

In this case, data x_d is not playing a role, and to yield a large likelihood one seeks $x \sim 0$, which may lead to incorrect optima. This is similar to the discussion on σ_{min}: since along the U-shape of the likelihood function, the largest likelihood is obtained at the lower bound of r at this region of r (i.e., $r \ll x \sim x_d$), we need to make sure that r is not too small. Per these two considerations (i.e., r should be neither too large nor too small), in the experiments we found that r around 0.1 is a good choice; a larger or smaller r often lead to inferior results. It is thus also reasonable to take r out of optimization and use the fixed value; tests suggest that whether putting r in optimization or not generally has little influence on the performance of negative binomial estimators.

D5: Automation and Model Documentation

Estimation is automated using Python scripts which conduct the same experiment for multiple runs where the simulated data series are generated with different random seeds. The scripts are fully parameterized and are compiled into i/o format. For the purpose of reproducing the current study, one does not need to access the main scripts, and all experimental setups and variants could be made in *params.py*.

Figures and statistics are drawn/computed with Matlab. Scripts automatically take in python output files and produce the two major types of figures in the main text (i.e., Figure 3 and 4 types). Complete models (in Vensim .mdl format) and scripts (python and Matlab) are publicly available at https://github.com/TimothyLi0123/LRS_Estimation.

Acknowledgements

We thank Tom Fiddaman and Ventana Systems for providing the Vensim simulation and optimization platform. Complete models and scripts of this study are publicly available at *https://github.com/TimothyLi0123/*.