Accurately Estimating Total COVID-19 Infections using Information Theory

Jiaming Cui[1], Arash Haddadan[2], A S M Ahsan-Ul Haque[3], Jilles Vreeken[4], Bijaya Adhikari[5], Anil Vullikanti[2,3], and B. Aditya Prakash[1,*]

[1]College of Computing, Georgia Institute of Technology, Atlanta, GA 30332, US
[2]Biocomplexity Institute, University of Virginia, Charlottesville, VA 22904, US
[3]Department of Computer Science, University of Virginia, Charlottesville, VA 22904, US
[4]CISPA Helmholtz Center for Information Security, Saarbrücken 66123, Germany
[5]Department of Computer Science, The University of Iowa, Iowa City, IA 52242, US

Supplementary Information

[*]To whom correspondence should be addressed. E-mail: badityap@cc.gatech.edu

Supplementary Information

Table S1: **List of notations**

Symbol	Description
BaseInfer	Current methodology to estimate reported rate
MdlInfer	Our Minimum Description Length (MDL) framework to estimate reported rate
Calibrate	Calibration procedure used in BaseInfer and MdlInfer
O_{M}	Epidemiological models used in BaseInfer and MdlInfer
BaseParam	Baseline parameterizaton obtained by BaseInfer
MdlParam	MDL parameterization identified by MdlInfer
$\mathrm{SeroStudy}_{\mathrm{Tinf}}$	Total infections estimated by serological studies
$\mathrm{BaseParam}_{\mathrm{Tinf}}$	Total infections estimated by BaseInfer
$\mathrm{MdlParam}_{\mathrm{Tinf}}$	Total infections estimated by MdlInfer
ρ_{Tinf}	The performance metric comparing MdlInfer against BaseInfer in estimating total infections
NYT-Rinf	New York Times reported infections
$\mathrm{BaseParam}_{\mathrm{Rinf}}$	Reported infections estimated by BaseInfer
$\mathrm{MdlParam}_{\mathrm{Rinf}}$	Reported infections estimated by MdlInfer
ρ_{Rinf}	The performance metric comparing MdlInfer against BaseInfer in estimating reported infections
$\mathrm{Rate}_{\mathrm{Symp}}$	COVID-related symptomatic rate from symptomatic surveillance data
$\mathrm{BaseParam}_{\mathrm{Symp}}$	Symptomatic rate estimated by BaseInfer
$\mathrm{MdlParam}_{\mathrm{Symp}}$	Symptomatic rate estimated by MdlInfer
$\mathrm{SeroStudy}_{\mathrm{Rate}}$	Cumulative reported rate estimated by serological studies
$\mathrm{BaseParam}_{\mathrm{Rate}}$	Cumulative reported rate estimated by BaseInfer
$\mathrm{MdlParam}_{\mathrm{Rate}}$	Cumulative reported rate estimated by MdlInfer

In the supplementary materials, we first describe the datasets used in the paper. These include the *observed data* used for epidemiological model calibration. We then describe the epidemiological models used in the main paper in detail. Then, we elaborate on our Minimum Description Length optimization formulation and the two-step algorithm, both of which we had briefly described in the main paper. Finally, we present the results which were omitted from the main paper.

Data

New York Times reported infections [2]

This dataset (NYT-Rinf) consists of the time sequence of reported infections D_{reported} and reported mortality $D_{\text{mortality}}$ in each county across the U.S. since the beginning of the COVID-19 pandemic (January 21, 2020) to current. For each county, the NYT-Rinf dataset provides the date, FIPS code, and the cumulative values of reported infections and mortality. Here, we use the averaged counts over 14 days to eliminate noise.

Serological studies [5, 1]

This dataset consists of the point estimate and 95% confidence interval of the prevalence of antibodies to SARS-CoV-2 in 10 US locations every 3-4 weeks during March to July 2020. The serological studies use the blood specimens collected from population. For each location, CDC collects around 1800 samples approximately every 3-4 weeks. Using the prevalence of antibodies and the population, we can compute the estimated total infections and 95% confidence interval in the location. However, we cannot compare this number with the epidemiological model estimated total infections directly as mentioned in the main article Methods section. We account for this problem by comparing the serological studies numbers with the estimated total infections of 7 days prior to the first day of specimen collection period as suggested by the CDC serological studies work [5].

Symptomatic surveillance [8]

This dataset comes from Facebook's symptomatic survey [8]. The survey started on April 6, 2020 to current. As of January 28, 2021, there were a total of 16,398,000 participants, with the average daily participants number of 55,000. The survey asks a series of questions designed to help researchers understand the spread of COVID-19 and its effect on people in the United States. For the signal, they estimate the percentage of self-reported COVID-19 symptoms in population defined as fever along with either cough, shortness of breath, or difficulty breathing [8]. The dataset also includes weighted version which accounts for the differences between Facebook users and the United States population. In the experiments, we contrast the symptomatic rate trends inferred by our approach against the weighted data from the survey.

Epidemiological model

SAPHIRE model

We use the SAPHIRE model [4] as one epidemiological model O_{M} in our experiments. The compartmental diagram of SAPHIRE model is shown in Supplementary Fig. S1. It consists of 7 states shown in Supplementary Table S2.

The SAPHIRE model has 9 different parameters, which are listed in Supplementary Table S3. Note that only two parameters are calibrated, while the rest are fixed. Following its literature [4], we use Markov Chain Monte Carlo (MCMC) as the calibration procedure CALIBRATE for SAPHIRE.

In this article, we expect the epidemiological model to calibrate on both reported infections D_{reported} and candidate total infections D. We compute the newly reported infections and unreported infections as follows:

1. New reported infections= $\frac{rP}{D_p}$: $\frac{P}{D_p}$ represents the number of new infections from presymptomatic infections every day in O_{M}. Here, we assume r proportion of new infections every day will be that day's new reported infections.

2. New unreported infections = $\frac{(1-r)P}{D_p}$: Then, the $1-r$ proportion of new infections every day will be that day's new unreported infections.

SEIR+HD model

We also use the SEIR + HD model [6] as another epidemiological model O_{M} in our experiments. The compartmental diagram of SEIR + HD model is shown in Supplementary Fig. S2. It consists of the following 10 states shown in Supplementary Table S4:

The SEIR + HD model has 21 different parameters, which are listed in Supplementary Table S5. Note that only three parameters are calibrated, while the rest are fixed. Following its literature [6], we use iterated filtering (IF) as the calibration procedure CALIBRATE for SEIR + HD.

Similarly to SAPHIRE model, we still expect the epidemiological model to calibrate on reported infections D_{reported} and candidate unreported infections $D_{\text{unreported}}$. Specifically, we extend the calibration procedure to infer two more parameters: α and α_1 (proportion of new symptomatic infections that are reported). We compute the newly reported infections and unreported infections as follows:

1. New reported infections= $\alpha_1 \times (N_{I_P I_S} + N_{I_P I_M})$:

   $I_{\text{new sympt}} = N_{I_P I_S} + N_{I_P I_M}$ represents the number of new symptomatic infections every day in O_{M}. Here, we assume α_1 proportion of new symptomatic infections will be that day's new reported infections.

2. New unreported infections = $(1-\alpha_1) \times (N_{I_P I_S} + N_{I_P I_M}) + N_{E I_A}$:

   Then, the $1-\alpha_1$ proportion of new symptomatic infections and new asymptomatic infections every day will be that day's new unreported infections.

Methodology

Terms used in MDL cost

By calibrating the epidemiological model O_{M} on D_{reported}, we get the baseline paramterization (BASEPARAM) $\hat{\Theta}$:

$$\hat{\Theta} = \text{CALIBRATE}(O_{\text{M}}, D_{\text{reported}}) \tag{1}$$

By running the epidemiological model with $\hat{\Theta}$, O_{M} will output the estimated reported infections $D_{\text{reported}}(\hat{\Theta})$, estimated unreported infections $D_{\text{unreported}}(\hat{\Theta})$, and estimated total infections $D(\hat{\Theta}) =$

$D_{\text{reported}}(\hat{\Theta}) + D_{\text{unreported}}(\hat{\Theta})$. We can also calculate the reported rate $\hat{\alpha}_{\text{reported}}$ as follows:

$$\hat{\alpha}_{\text{reported}} = \frac{\sum D_{\text{reported}}(\hat{\Theta})}{\sum D(\hat{\Theta})} \tag{2}$$

Here, we sum over the daily sequence $D_{\text{reported}}(\hat{\Theta})$ and $D(\hat{\Theta})$ to calculate a scalar as the reported rate for MDL formulation.

Similarly by calibrating O_{M} on both D_{reported} and D, we get the candidate paramterization $\Theta^{'}$:

$$\Theta^{'} = \textsc{Calibrate}(O_{\text{M}}, (D, D_{\text{reported}})) \tag{3}$$

By running the epidemiological model with $\Theta^{'}$, O_{M} will output the estimated reported infections $D_{\text{reported}}(\Theta^{'})$, estimated unreported infections $D_{\text{unreported}}(\Theta^{'})$, and estimated total infections $D(\Theta^{'}) = D_{\text{reported}}(\Theta^{'}) + D_{\text{unreported}}(\Theta^{'})$. Similarly, we can calculate the reported rate $\alpha^{'}_{\text{reported}}$ as follows:

$$\alpha^{'}_{\text{reported}} = \frac{\sum D_{\text{reported}}(\Theta^{'})}{\sum D(\Theta^{'})} \tag{4}$$

With the calibration process, $\hat{\Theta}$, and $\Theta^{'}$ defined, we can next formalize the MDL cost.

Sender-receiver framework

Here, we use two-part sender-receiver framework based on Minimum Description Length (MDL) principle. The goal of the framework is to transmit the Data in possession of the Sender S to the receiver R using a Model. We do this by identifying the Model that describes the Data such that the total number of bits needed to encode both the Model and the Data is minimized. The number of bits required to encode both the Model and the Data is given by the cost function L, which has two components: (i) model cost L(Model): The cost in bits of encoding the Model, and (ii) data cost L(Data|Model): The cost in bits of encoding Data given the Model.

Model space: Other choice

In this work, the Data is D_{reported}. One idea for defining the Model space is to use $\hat{\Theta}$. With such a Model, the receiver R can easily compute first $D_{\text{reported}}(\hat{\Theta})$ given $\hat{\Theta}$. Then the sender S will only need to encode and send the difference between $D_{\text{reported}}(\hat{\Theta})$ and D_{reported} so that the receiver can recover the Data fully. However, this has the disadvantage that slightly different $\hat{\Theta}$ could lead to vastly different $D_{\text{reported}}(\hat{\Theta})$, and so the optimization problem will become hard to solve. To account for this, we propose Model as Model $= (D, \Theta^{'}, \hat{\Theta})$ as described in the main article, which consists of three components.

Model cost

With the model space Model $= (D, \Theta^{'}, \hat{\Theta})$, the sender S will send the Model to the receiver R in three parts: (i) first send $\hat{\Theta}$, (ii) next send $\Theta^{'}$ given $\hat{\Theta}$, and then (iii) send D given $\Theta^{'}$ and $\hat{\Theta}$. Therefore, the model cost $L(D, \Theta^{'}, \hat{\Theta})$ will also have three components

$$L(D, \Theta^{'}, \hat{\Theta}) = \textsc{Cost}(\hat{\Theta}) + \textsc{Cost}(\Theta^{'}|\hat{\Theta}) + \textsc{Cost}(D|\Theta^{'}, \hat{\Theta}) \tag{5}$$

Here, we will send the first component, $\hat{\Theta}$, directly, send the second component, $\Theta^{'}$ given $\hat{\Theta}$, via sending $\Theta^{'} - \hat{\Theta}$, and send the third component, D given $\Theta^{'}$ and $\hat{\Theta}$, via sending $\alpha^{\prime}_{\text{reported}} \times D -$

$D_{\text{reported}}(\hat{\Theta})$ (as described in the main article, both $\alpha'_{\text{reported}} \times D$ and $D_{\text{reported}}(\hat{\Theta})$ should be close to D_{reported}, and the receiver could recover the D using $\hat{\Theta}$, $\alpha'_{\text{reported}}$, and $D_{\text{reported}}(\hat{\Theta})$ since they have already been sent). We further write the model cost in Equation (5) as below:

$$L(D, \Theta^{'}, \hat{\Theta}) = \textsc{Cost}(\hat{\Theta}) + \textsc{Cost}(\Theta^{'} - \hat{\Theta}|\hat{\Theta}) + \textsc{Cost}(\alpha'_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})|\Theta^{'}, \hat{\Theta}) \tag{6}$$

Data cost

Give the $\textsc{Model} = (D, \Theta^{'}, \hat{\Theta})$ and model cost above, next we will send the $\textsc{Data}$ in terms of the $\textsc{Model}$. Here, the $\textsc{Data}$ is D_{reported}, and the data cost will have only one component:

$$L(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta}) = \textsc{Cost}(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta}) \tag{7}$$

Here, we will send it via $\frac{D-D_{\text{reported}}}{1-\alpha'_{\text{reported}}} - D(\Theta^{'})$ (as described in the main article, both $\frac{D-D_{\text{reported}}}{1-\alpha'_{\text{reported}}}$ and $D(\Theta^{'})$ should be close to the total infections D, and the receiver could recover the D_{reported} using D, $\alpha'_{\text{reported}}$, and $D(\Theta^{'})$ since they have already been sent), and we further write the data cost in Equation (7) as below:

$$L(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta}) = \textsc{Cost}(\frac{D - D_{\text{reported}}}{1 - \alpha'_{\text{reported}}} - D(\Theta^{'})|D, \Theta^{'}, \hat{\Theta}) \tag{8}$$

Total cost

The total cost is the sum of model cost $L(D, \Theta^{'}, \hat{\Theta})$ and data cost $L(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta})$:

$$\begin{aligned} L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta}) &= L(D, \Theta^{'}, \hat{\Theta}) + L(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta}) \\ &= \textsc{Cost}(\hat{\Theta}) + \textsc{Cost}(\Theta^{'}|\hat{\Theta}) + \textsc{Cost}(D|\Theta^{'}, \hat{\Theta}) + \textsc{Cost}(D_{\text{reported}}|D, \Theta^{'}, \hat{\Theta}) \\ &= \textsc{Cost}(\hat{\Theta}) + \textsc{Cost}(\Theta^{'} - \hat{\Theta}|\hat{\Theta}) + \textsc{Cost}(\alpha'_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})|\Theta^{'}, \hat{\Theta}) \\ &+ \textsc{Cost}(\frac{D - D_{\text{reported}}}{1 - \alpha'_{\text{reported}}} - D(\Theta^{'})|D, \Theta^{'}, \hat{\Theta}) \end{aligned} \tag{9}$$

Cost derivation

Next, we derive the cost for each component and give our encoding method explicitly:

1. $\textsc{Cost}(\hat{\Theta})$: We represent $\hat{\Theta}$ as a vector of real numbers. We describe our encoding later below.
2. $\textsc{Cost}(\Theta^{'} - \hat{\Theta}|\hat{\Theta})$: We will encode the difference of two vectors as a vector of real numbers.
3. $\textsc{Cost}(\alpha'_{\text{reported}} \times D - D_{\text{reported}}(\hat{\Theta})|\Theta^{'}, \hat{\Theta})$: Here, we encode the difference between the two time sequences: $\alpha'_{\text{reported}} \times D$ given $D_{\text{reported}}(\hat{\Theta})$.
4. $\textsc{Cost}(\frac{D-D_{\text{reported}}}{1-\alpha'_{\text{reported}}} - D(\Theta^{'})|D, \Theta^{'}, \hat{\Theta})$: Again, we encode it as a difference between the two time sequences: $\frac{D_{\text{unreported}}}{1-\alpha'_{\text{reported}}}$ given $D(\Theta^{'})$.

Next, we describe the encoding cost of real numbers, vectors, and the difference between two time sequences.

Encoding integers

To encode a positive integer n^+, we encode both the binary representation of integer n^+ as well as the length of the representation $\log_2 n^+$. Following [7], the cost in bits of encoding a single integer n is as follows:

$$\textsc{Cost}(n^+) = \log_2 c_0 + \log^*(n^+). \tag{10}$$

where $c_0 \approx 2.865$ and $\log^*(n^+) = \log_2 n^+ + \log_2 \log_2 n^+ + \cdots$ as described in [7]. There are infinite terms in $\log^*(n^+)$ function since after we encode a number, we always need to encode its length as another number, which could be repeated for infinite times. Additionally, if we want to transmit an integer that can be either positive or negative, we can add another sign bit and therefore the cost in bits for integers will be

$$\textsc{Cost}(n) = \textsc{Cost}(|n|) + 1. \tag{11}$$

Encoding real numbers

Note that most real numbers (e.g. π or e) need infinite number of bits to encode. Hence, we introduce a precision threshold δ. With threshold δ, we approximate a positive real number x^+ with x_δ which satisfies $|x^+ - x_\delta| < \delta$, and we encode x_δ instead. To encode x_δ, we encode both the integer part $\lfloor x^+ \rfloor$ as well as the fractional part $x_\delta - \lfloor x^+ \rfloor$. Hence the cost in bits of encoding a positive real number x^+ is as follows:

$$\textsc{Cost}(x^+) = \textsc{Cost}(\lfloor x^+ \rfloor) + \log_2 \frac{1}{\delta} \tag{12}$$

where $\lfloor x^+ \rfloor$ is the floor of x^+ and therefore is a integer, whose encoding cost is $\textsc{Cost}(\lfloor x^+ \rfloor) = \log_2 c_0 + \log^*(\lfloor x^+ \rfloor)$. Additionally, if we want to transmit a real number that can be either positive or negative, we can add another sign bit and therefore the cost in bits for real numbers will be

$$\textsc{Cost}(x) = \textsc{Cost}(|x|) + 1 \tag{13}$$

Encoding vectors

To encode a vector $\Theta = [\Theta[1], \Theta[2], \cdots, \Theta[n]]$, we encode every components one by one as real numbers. Hence the cost in bits of encoding a vector Θ is as follows:

$$\textsc{Cost}(\Theta) = \textsc{Cost}(\Theta[1]) + \textsc{Cost}(\Theta[2]) + \cdots + \textsc{Cost}(\Theta[n]) \tag{14}$$

Encoding the difference between two time sequences

To encode the difference $A - B = [A_{t_1} - B_{t_1}, A_{t_2} - B_{t_2}, \cdots, A_{t_n} - B_{t_n}]$ between two time sequences $A = [A_{t_1}, A_{t_2}, \cdots, A_{t_n}]$ and $B = [B_{t_1}, B_{t_2}, \cdots, B_{t_n}]$, we encode every components one by one as real numbers. Hence the cost in bits of encoding the difference is as follows:

$$\textsc{Cost}(A - B) = \textsc{Cost}(A_{t_1} - B_{t_1}) + \textsc{Cost}(A_{t_2} - B_{t_2}) + \cdots + \textsc{Cost}(A_{t_n} - B_{t_n}) \tag{15}$$

Problem statement

Now we have derived every cost involved in our problem, and we can finally state our problem as one of estimating the total infections D as follows: Given the time sequence D_{reported}, epidemiological model O_{M}, and a calibration procedure $\textsc{Calibrate}$, find D^* that minimizes the MDL total cost:

$$D^* = \arg\min_D L(D_{\text{reported}}, D, \Theta', \hat{\Theta}) \tag{16}$$

We will give the algorithm to find such D^* as follows:

Algorithm

Before presenting our algorithm to find D^*, we will first address the problem of searching D^* directly. Note that D^* is a time sequence of total infections instead of a scalar, naively searching D^* directly in large search space is intractable. Hence, we propose an alternate method: First, we can quickly find a good reported rate $\alpha^*_{\text{reported}}$ since we can constrain $D = \frac{D_{\text{reported}}}{\alpha_{\text{reported}}}$ to reduce the search space. Then we can search for the optimal D^* with $\alpha^*_{\text{reported}}$ from step 1 as constraints. Here, we write down our two-step search algorithm to find the D^* as follows:

1. Step 1: We do a linear search to find a good reported rate $\alpha^*_{\text{reported}}$, which serves as an initialization in the second step.
2. Step 2: Given the $\alpha^*_{\text{reported}}$ found in step 1, we use the Nelder-Mead [3] optimization to find the D^* that minimizes $L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$ with $\alpha^*_{\text{reported}}$ constraints.

Step 1: Find the $\alpha^*_{\text{reported}}$

In step 1, we search on α_{reported} to find the $\alpha^*_{\text{reported}}$ as follows:

$$\alpha^*_{\text{reported}} = \arg\min_{\alpha_{\text{reported}}} L(D_{\text{reported}}, \frac{D_{\text{reported}}}{\alpha_{\text{reported}}}, \Theta^{'}, \hat{\Theta}) \tag{17}$$

To be more specific, in the first step of our algorithm, we do a linear search on different $\alpha_{\text{reported}} = [0.01, 0.02, 0.03, \cdots, 0.99]$ and calibrate the O_{M} on $D = \frac{D_{\text{reported}}}{\alpha_{\text{reported}}}$, which means

$$\Theta^{'} = \textsc{Calibrate}(O_{\text{M}}, (\frac{D_{\text{reported}}}{\alpha_{\text{reported}}}, D_{\text{reported}})) \tag{18}$$

Then we pick the $\alpha^*_{\text{reported}}$ that corresponds to the lowest total cost as the $\alpha^*_{\text{reported}}$.

Step 2: Find the D^* given $\alpha^*_{\text{reported}}$

With $\alpha^*_{\text{reported}}$ inferred in step 1, we will next find the D^* that minimizes the total cost.

$$D^* = \arg\min_{D} L(D_{\text{reported}}, D, \Theta^{'}, \hat{\Theta}) \tag{19}$$

Since we have already found $\alpha^*_{\text{reported}}$ in step 1, we will only search the D^* that satisfies

$$\sum D^* = \frac{\sum D_{\text{reported}}}{\alpha^*_{\text{reported}}} \tag{20}$$

To search for the optimal D^*, we leverage the popular Nelder-Mead search algorithm [3].

We give the pseudo-code for MdlInfer as follows:

Algorithm 1 MdlInfer

Input: Epidemiological model O_{M}, calibration procedure Calibrate, reported infections time sequence D_{reported}.

1: Calibrate baseline parameterization $\hat{\Theta} = \textsc{Calibrate}(O_{\text{M}}, D_{\text{reported}})$
2: Step 1: Find $\alpha^*_{\text{reported}} = \textsc{GetAlpha}(O_{\text{M}}, \textsc{Calibrate}, D_{\text{reported}}, \hat{\Theta})$
3: Step 2: Find $D^* = \textsc{GetTotalInfections}(O_{\text{M}}, \textsc{Calibrate}, D_{\text{reported}}, \alpha^*_{\text{reported}}, \hat{\Theta})$

Output: Total infections D^*

We also give the pseudo-code for GetAlpha and GetTotalInfections:

Algorithm 2 GetAlpha (Step 1: Find the $\alpha^*_{\text{reported}}$)

Input: O_{M}, Calibrate, D_{reported}, $\hat{\Theta}$.

The array to save the MDL cost: CostArray = [ ]
for α_{reported} in the grid search space from 0.01 to 1 with step 0.01 **do**
  $D = \frac{D_{\text{reported}}}{\alpha_{\text{reported}}}$
  Calibrate candidate parameterization $\Theta' = \text{Calibrate}(O_{\text{M}}, (D, D_{\text{reported}}))$
  Save the MDL cost for α_{reported} in CostArray[α_{reported}] = $L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$
end for
Find the $\alpha^*_{\text{reported}} = \arg\min_{\alpha_{\text{reported}}}$ CostArray[α_{reported}]

Output: Reported rate $\alpha^*_{\text{reported}}$

Algorithm 3 GetTotalInfections (Step 2: Find the D^* given $\alpha^*_{\text{reported}}$)

Input: O_{M}, Calibrate, D_{reported}, $\alpha^*_{\text{reported}}$, $\hat{\Theta}$.

Find the $D^* = \arg\min_D L(D_{\text{reported}}, D, \Theta', \hat{\Theta})$. (using the Nelder-Mead algorithm).

Output: Total infections D^*

Why MdlInfer is better than BaseInfer

1. We introduce a latent variable D to help the calibration: With D and its corresponding $\alpha'_{\text{reported}}$, we can focus on a smaller search space by fixing the reported rate and get a better fit on D_{reported}. In contrast, BaseInfer ignores the D but directly searches in the whole parameter space for $\hat{\Theta}$, which is a complex problem in high-dimensional space since there are multiple parameters to estimate simultaneously. As shown in results section (B), we fit the observed D_{reported} better (measured by lower RMSE) in most regions.

2. We performed a principled model selection framework to search for the D. We carefully designed our MDL cost to minimize the discrepancy in fitting D_{reported}. This MDL cost ensures the generalizability of our learned D and $\alpha'_{\text{reported}}$ to avoid the overfitting of D_{reported}. Specifically, as we described in the methods section, we attempt different Ds that correspond to different reported rates in our algorithm. This can be viewed as a linear search on α_{reported}, which helps to avoid getting a locally optimal solution. Again, as shown in results section (B), good generalizability leads to better forecasts of future reported infections.

MdlInfer identifies the ground truth parameters better than BaseInfer: Synthetic experiments

Here, we use SAPHIRE and SEIR + HD model and its corresponding calibration procedure (Markov Chain Monte Carlo and iterated filtering) to showcase that MdlInfer identifies the ground truth parameters better than BaseInfer. Specifically, we use a synthetic parameterization Θ_{gt} to generate a synthetic reported infections curve, and then add different amounts of Gaussian noise to this reported infections curve. We use these noisy curves as the observed data D_{reported}, and then use both MdlInfer and BaseInfer to fit these curves. We hope that the MdlParam estimated by MdlInfer could be closer to the Θ_{gt} than BaseParam estimated by BaseInfer. As

shown in Supplementary Fig. S3, with increasing amount of Gaussian noise, both the root mean squared error (RMSE) between BaseParam and Θ_{gt}, and the RMSE between MdlParam and Θ_{gt} increase. However, the RMSE between MdlParam and Θ_{gt} increases slower than the RMSE between BaseParam and Θ_{gt}. This shows MdlInfer always identifies the ground truth parameters Θ_{gt} better than BaseInfer. It also shows that MdlInfer is more robust to noise than BaseInfer. We list the parameters in Supplementary Table S6 (for SAPHIRE model) and Supplementary Table S7 (for SEIR + HD model).

Using MdlInfer to generate uncertainty estimates

Here, we also use SAPHIRE and SEIR + HD model and its corresponding calibration procedure (Markov Chain Monte Carlo and iterated filtering) to showcase how to adapt the MdlInfer to generate uncertainty estimates for the inferred parameters. Note that we need multiple estimated parameters to generate uncertainty estimates and compute the mean value and standard error. Hence, we start from multiple $\hat{\Theta}_i$ by running BaseInfer multiple times. Recall that our MDL formulation builds a deterministic mapping from $\hat{\Theta}$ to Θ^*. Hence for each $\hat{\Theta}_i$, we could find the corresponding Θ_i^* using MdlInfer. Specifically, for each $\hat{\Theta}_i$ with probability Prob$[\hat{\Theta}_i]$, we can find the corresponding Θ_i^* with the same probability Prob$[\Theta_i^*]$. Although the corresponding Θ_i^* for different $\hat{\Theta}_i$ may overlap with each other (in fact, the ideal mapping is to map all $\hat{\Theta}_i$ to the 'ground-truth' Θ_{gt}), this mapping always exists and is deterministic. Hence, given multiple $\hat{\Theta}_i$ estimated by BaseInfer, we can find the corresponding Θ_i^*, and give the uncertainty estimates. Some examples are shown in Supplementary Table S8 (for SAPHIRE model) and Supplementary Table S9 (for SEIR + HD model). Here, we use a synthetic parameterization Θ_{gt} to generate a synthetic reported infections curve and then add 5% Gaussian noise to this reported infections curve. We use the curve with 5% Gaussian noise as the observed data D_{reported}. We run BaseInfer 10 times get multiple $\hat{\Theta}_i$, and then run MdlInfer with each $\hat{\Theta}_i$ to generate multiple Θ_i^*. Then we can use Θ_i^* to generate the uncertainty estimates. Note that the variance for both BaseParam and MdlParam are small.

Experimental setup

Here we describe our experimental setup in more detail and present results on additional testbeds.

Total infections

The Results section in the main paper refers to $\text{BaseParam}_{\text{Tinf}}$, which represents the cumulative total infections derived from the BaseInfer. It is computed as follows:

$$\text{BaseParam}_{\text{Tinf}} = \sum D(\hat{\Theta}) \tag{21}$$

Similarly, $\text{MdlParam}_{\text{Tinf}}$, which represents the cumulative total infections derived from MdlInfer, is computed as follows:

$$\text{MdlParam}_{\text{Tinf}} = \sum D(\Theta^*) \tag{22}$$

In Supplementary Fig. S4, we show additional results comparing the performance of MdlInfer and BaseInfer in estimating total infections. Here, MdlInfer (red) gives a closer estimation of total infections to serological studies (black) than BaseInfer (blue).

Reported infections

In Supplementary Fig. S5, we present additional results comparing the performance of MdlInfer and BaseInfer in forecasting future infections (forecast period). Here, MdlInfer (red) gives a closer estimation of reported infections (black) than BaseInfer (blue) on various geographical regions and time periods.

Symptomatic rate

The BaseInfer and MdlInfer also estimate the symptomatic rate $\text{BaseParam}_{\text{Symp}}$ and $\text{MdlParam}_{\text{Symp}}$ respectively. We compare these against the Facebook symptomatic surveillance data $\text{Rate}_{\text{Symp}}$.

We calculate $\text{BaseParam}_{\text{Symp}}$ from $\hat{\Theta}$ as follows:

$$\text{BaseParam}_{\text{Symp}} = \frac{I_S(\hat{\Theta}) + I_M(\hat{\Theta})}{N} \tag{23}$$

where $I_S(\hat{\Theta})$ is the number of infections in severe symptomatic state, $I_M(\hat{\Theta})$ represents the same in mild symptomatic state, and N is the total population in this area.

Similarly $\text{MdlParam}_{\text{Symp}}$ is computed as follows:

$$\text{MdlParam}_{\text{Symp}} = \frac{I_S(\Theta^*) + I_M(\Theta^*)}{N} \tag{24}$$

In Supplementary Fig. S6, we present additional results comparing MdlInfer and BaseInfer in estimating trends of symptomatic rate. Here, MdlInfer (red) gives a closer estimation of the trends of symptomatic rate (black) than BaseInfer (blue).

Cumulative reported rate

We also calculate the a dynamic reported rate from both BaseInfer and MdlInfer. Note that this cumulative reported rate is different from α_{reported} and $\alpha^*_{\text{reported}}$, which are two scalars used in MDL formulation. We calculate $\text{BaseParam}_{\text{Rate}}$ from BaseParam $\hat{\Theta}$ as follows:

$$\text{BaseParam}_{\text{Rate}} = \frac{\sum \text{NYT-Rinf}}{\sum D(\hat{\Theta})} \tag{25}$$

Similarly we calculate $\text{MdlParam}_{\text{Rate}}$ from MdlParam Θ^* as follows:

$$\text{MdlParam}_{\text{Rate}} = \frac{\sum \text{NYT-Rinf}}{\sum D(\Theta^*)} \tag{26}$$

Non-pharmaceutical interventions simulation

We also use the BaseInfer and MdlInfer to perform non-pharmaceutical interventions simulation on SEIR + HD model. Here, both the BaseInfer and MdlInfer are estimated on the observed period. Then on the future period, we will consider the following five scenarios of isolation:

1. Isolate reported infections: We isolate the α_1 fraction of severe symptomatic infections I_S and mild symptomatic infections I_M.

2. Isolate both reported infections and symptomatic infections: Note that some reported infections are included in the symptomatic infections. Here, we isolate all severe symptomatic infections I_S and mild symptomatic infections I_M.

3. Isolate 25% presymptomatic and asymptomatic infections: We isolate 25% of presymptomatic infections I_P, asymptomatic infections I_A, and all severe symptomatic infections I_S and mild symptomatic infections I_M.

4. Isolate 50% presymptomatic and asymptomatic infections: We isolate 50% of presymptomatic infections I_P, asymptomatic infections I_A, and all severe symptomatic infections I_S and mild symptomatic infections I_M.

5. Isolate 75% presymptomatic and asymptomatic infections: We isolate 75% of presymptomatic infections I_P, asymptomatic infections I_A, and all severe symptomatic infections I_S and mild symptomatic infections I_M.

The infectivity of the patients in isolation is reduced by 50%.

Sensitivity analysis

We also perform sensitivity experiments to inspect the robustness of our non-pharmaceutical interventions simulations for Minneapolis-Spring-20 in Supplementary Fig. S7. Here, we reduce the infectiousness of the isolated infections to 3 different values: 0.4, 0.5, and 0.6, and repeat simulations in each scenarios. Our results consistently show that only isolating reported or symptomatic infections is not enough to reduce the future reported infections. However, isolating both symptomatic infections and some fraction of asymptomatic and presymptomataic infections leads to reduction in reported infections in most settings.

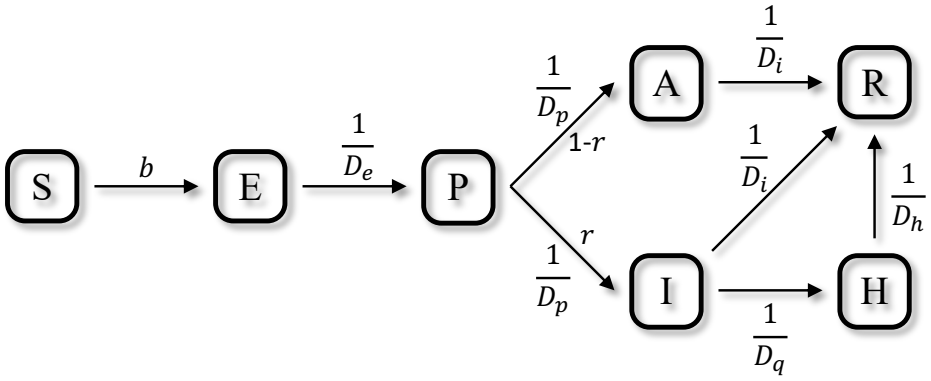

Figure S1: **Compartmental diagram of SAPHIRE model [4]**

Table S2: **States of SAPHIRE model**

State	Meaning
S	Susceptible
E	Exposed
P	Presymptomatic infectious
I	Ascertained infectious
A	Unascertained infectious
H	Isolation in hospital
R	Removed

Table S3: **Parameters of SAPHIRE model**

Parameter	Meaning	Value
b	Transmission rate of ascertained cases	Calibrated
r	Ascertainment rate	Calibrated
α	Ratio of transmission rate for unascertained over ascertained cases	0.55
D_e	Latent period	2.9
D_p	Presymptomatic infectious period	2.3
D_i	Symptomatic infectious period	2.9
D_q	Duration from illness onset to isolation	6
D_h	Isolation period	30
N	Population	Set

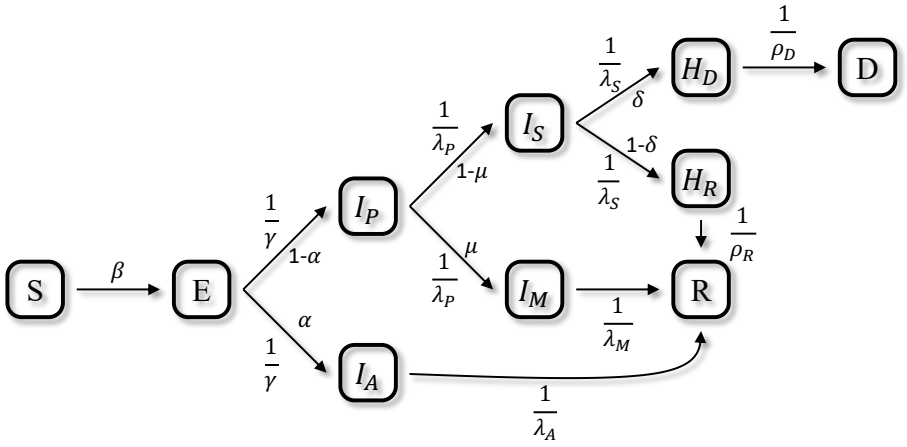

Figure S2: **Compartmental diagram of SEIR+HD model [6].**

Table S4: **States of SEIR+HD model**

State	Meaning
S	Susceptible
E	Exposed
I_P	Pre-symptomatic
I_S	Symptomatic, severe
I_M	Symptomatic, mild
I_A	Asymptomatic
H_D	Hospitalized, eventual death
H_R	Hospitalized, eventual recover
R	Recovered
D	Dead

Table S5: **Parameters of SEIR+HD model**

Parameter	Meaning	Value
C_A	Relative infectiousness of asymptomatic	0.425
C_P	Relative infectiousness of presymptomatic	1
C_M	Relative infectiousness of mild symptomatic	1
C_S	Relative infectiousness of severe symptomatic	1
γ	Preinfectious period	0.2857
λ_P	Presymptomatic duration	0.6667
λ_A	Infectious period for asymptomatic infections	0.1429
λ_S	Time from symptom onset to hospitalizations (severe)	0.1818
λ_M	Time from symptom onset to recovery (mild)	0.1818
ρ_R	Time from hospitalization to recovery	0.0667
ρ_D	Time from hospitalization to death	0.0752
N	Population	Set
Start date	Start date of the epidemic	Set
Work From Home start date	Work from home start date	Set
σ_{WFH}	Work from home proportion of contacts remaining	0.8125
E_0	Number of initial infections that began the epidemic	Calibrated
α	Proportion of infections that are asymptomatic	0.4875
δ	Mortality rate among hospitalizations	0.1375
μ	Proportion of symptomatic infections that require hospitalization	0.0656
β_0	Transmission rate in the absence of interventions	Calibrated
σ	The proportional reduction on β_0 under shelter-in-place	Calibrated

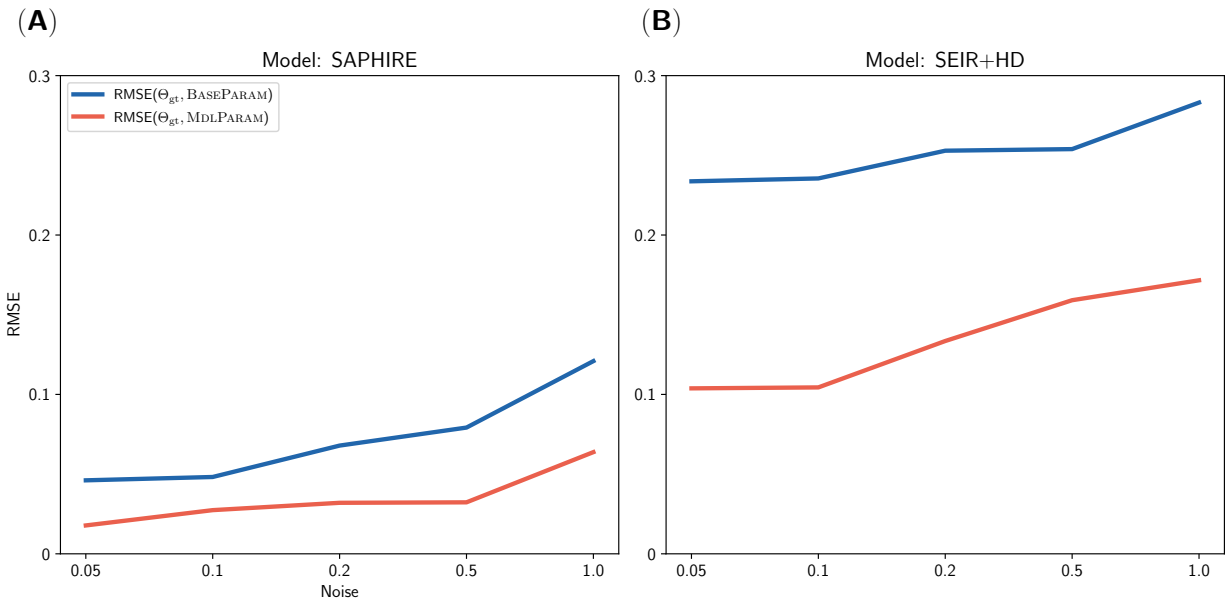

Figure S3: **MdlInfer identifies the ground truth parameters better than BaseInfer.** The red and blue curves represent the RMSE between MdlParam and Θ_{gt}, and RMSE between BaseInfer and Θ_{gt} when adding different amounts of Gaussian noise. Lower RMSE means better performance. (**A**) uses SAPHIRE model and its corresponding MCMC calibration procedure for both BaseInfer and MdlInfer, and (**B**) uses SEIR + HD model and its corresponding IF calibration procedure for both BaseInfer and MdlInfer.

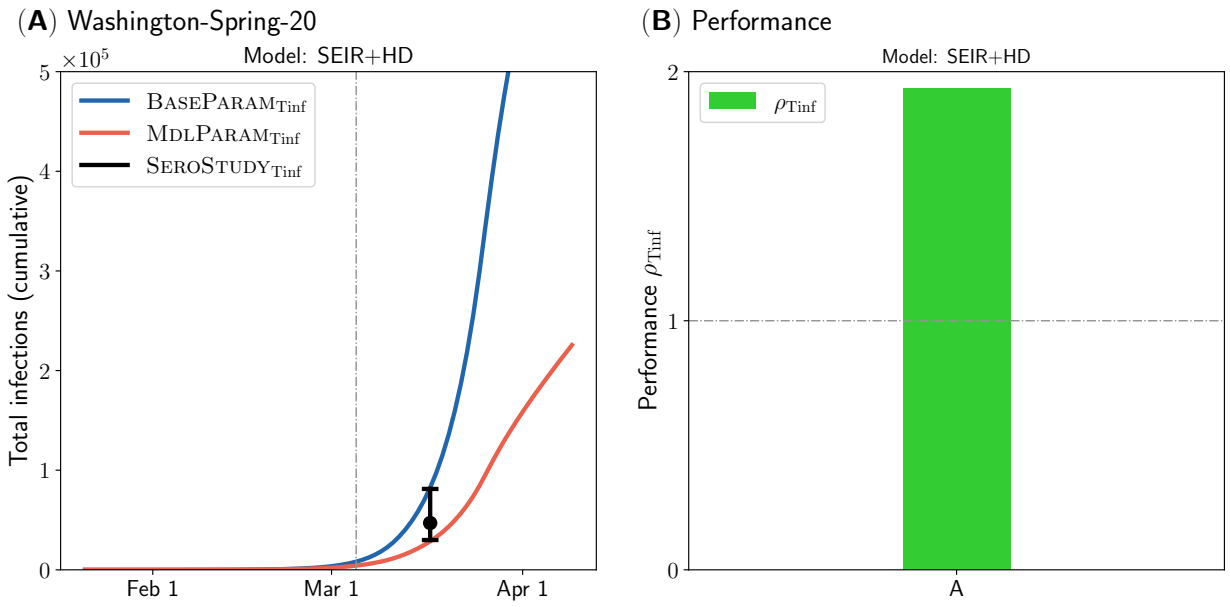

Figure S4: **MdlInfer (red) gives a closer estimation of total infections to serological studies (black) than BaseInfer (blue).** Note that both approaches try to fit the serological studies without being informed with them. (**A**) The red and blue curves represent MdlInfer's estimation of total infections, $\text{MdlParam}_{\text{Tinf}}$, and BaseInfer's estimation of total infections, $\text{BaseParam}_{\text{Tinf}}$, respectively. The black point estimates and confidence intervals represent the total infections estimated by serological studies [1, 5], $\text{SeroStudy}_{\text{Tinf}}$. (**B**) The performance metric, ρ_{Tinf}, comparing $\text{MdlParam}_{\text{Tinf}}$ against $\text{BaseParam}_{\text{Tinf}}$ in fitting serological studies is shown for (**A**) for SEIR + HD model. Here, the values of ρ_{Tinf} is 1.93.

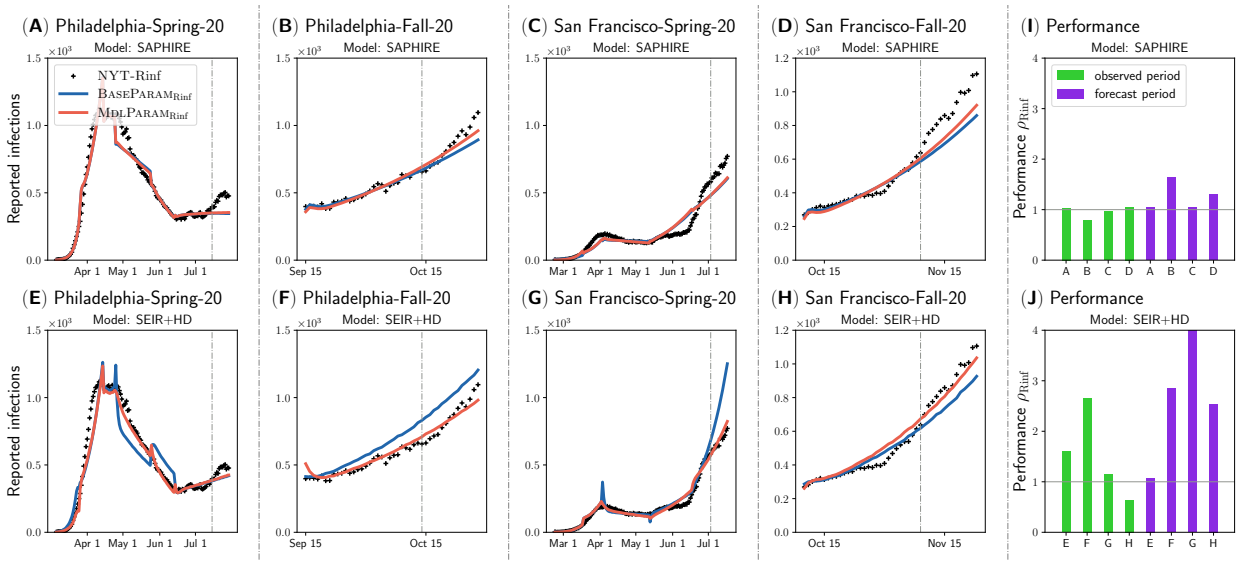

Figure S5: **MdlInfer (red) gives a closer estimation of reported infections (black) than BaseInfer (blue).** We use the reported infections in the observed period as inputs and try to forecast the future reported infections (forecast period). (**A**)-(**H**) The vertical grey dash line divides the observed period (left) and forecast period (right). The red and blue curves represent MdlInfer's estimation of reported infections, $\text{MdlParam}_{\text{Rinf}}$, and BaseInfer's estimation of reported infections, $\text{BaseParam}_{\text{Rinf}}$, respectively. The black plus symbols represent the reported infections collected by the New York Times (NYT-Rinf). (**A**)-(**D**) use SAPHIRE model and (**E**)-(**H**) use SEIR + HD model. (**I**)-(**J**) The performance metric, ρ_{Rinf}, comparing $\text{MdlParam}_{\text{Rinf}}$ against $\text{BaseParam}_{\text{Rinf}}$ in fitting reported infections is shown for each region. (**I**) is for SAPHIRE model in (**A**)-(**D**), and (**J**) is for SEIR + HD model in (**E**)-(**H**).

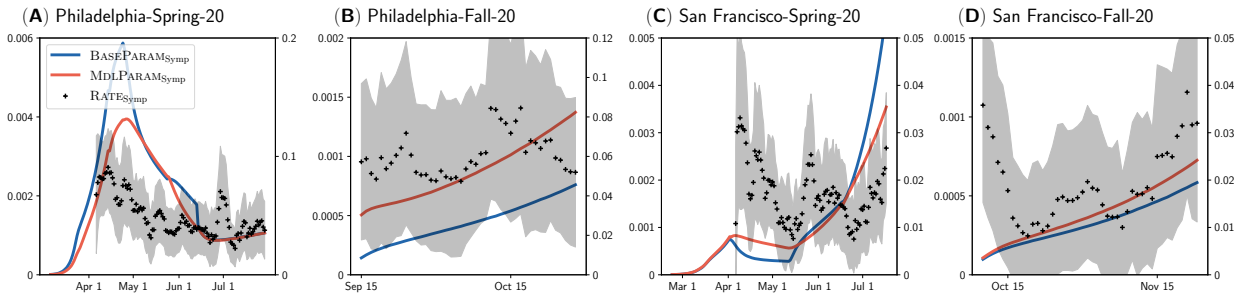

Figure S6: **MdlInfer (red) gives a closer estimation of the trends of symptomatic rate (black) than BaseInfer (blue).** (**A**)-(**D**) The red and blue curves represent MdlInfer's estimation of symptomatic rate, $\text{MdlParam}_{\text{Symp}}$, and BaseInfer's estimation of symptomatic rate, $\text{BaseParam}_{\text{Symp}}$, respectively. They use the y-scale on the left. The black points and the shaded regions are the point estimate with standard error for $\text{Rate}_{\text{Symp}}$ (the COVID-related symptomatic rates derived from the symptomatic surveillance dataset [8, 9]). They use the y-scale on the right. Note that we focus on trends instead of the exact numbers, hence $\text{MdlParam}_{\text{Symp}}$/$\text{BaseParam}_{\text{Symp}}$, and $\text{Rate}_{\text{Symp}}$ may scale differently.

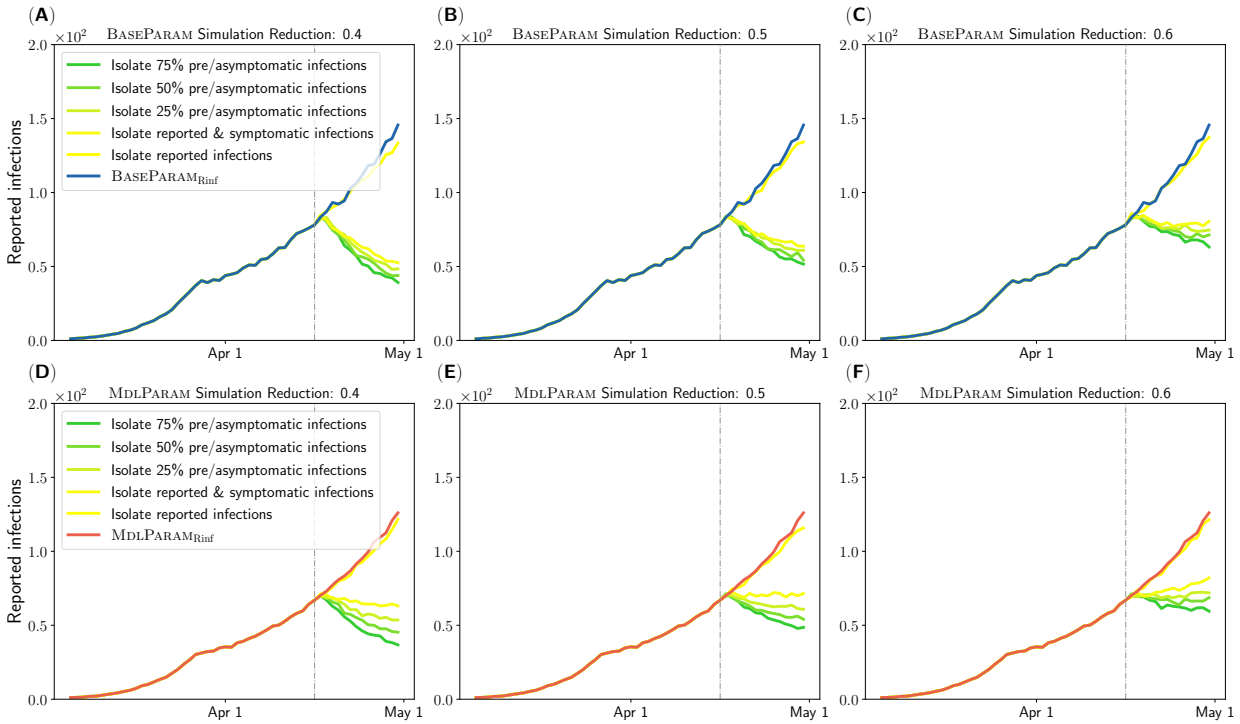

Figure S7: **Our non-pharmaceutical interventions simulation results are robust.** (**A**)-(**C**) The vertical grey dash line divides the observed period (left) and forecast period (right). The blue curve and other five curves represent the BaseInfer's estimation of reported infections for no NPI scenario and 5 different NPI scenarios described in the Results section. Here, we reduce the infectivity of these isolated infections to 40% in (**A**), 50% in (**B**), and 60% in (**C**) in future period. (**D**)-(**F**) The red curve and other five curves represent the MdlInfer's estimation of reported infections for no NPI scenario and 5 different NPI scenarios described in the Results section. Similarly, we reduce the infectivity of these isolated infections to 40% in (**D**), 50% in (**E**), and 60% in (**F**) in future period. The results are for Minneapolis-Spring-20 that we shown in main article Fig. 5.

Table S6: **Performance of BaseInfer and MdlInfer in identifying Θ_{gt} for SAPHIRE model**

Noise level	Parameter	BaseParam ($\hat{\Theta}$)	MdlParam (Θ^*)	Θ_{gt}
0.05	b	0.4552	0.4879	0.5
0.05	r	0.2473	0.2220	0.2
0.1	b	0.4524	0.4774	0.5
0.1	r	0.2487	0.2315	0.2
0.2	b	0.4398	0.4825	0.5
0.2	r	0.2749	0.2417	0.2
0.5	b	0.4109	0.4670	0.5
0.5	r	0.2678	0.2315	0.2
1.0	b	0.4356	0.5164	0.5
1.0	r	0.3583	0.2887	0.2

Table S7: **Performance of BaseInfer and MdlInfer in identifying Θ_{gt} for SEIR+HD model**

Noise level	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)	Θ_{gt}
0.05	β_0	0.6097	0.7189	0.8
0.05	α	0.0866	0.3110	0.5
0.05	α_1	0.2745	0.1834	0.2
0.05	σ	0.2239	0.2769	0.3
0.1	β_0	0.6207	0.8656	0.8
0.1	α	0.0745	0.6388	0.5
0.1	α_1	0.2810	0.3021	0.2
0.1	σ	0.2548	0.2022	0.3
0.2	β_0	0.5905	0.9530	0.8
0.2	α	0.0446	0.6663	0.5
0.2	α_1	0.2640	0.3354	0.2
0.2	σ	0.2806	0.2561	0.3
0.5	β_0	0.5971	1.0059	0.8
0.5	α	0.0426	0.7027	0.5
0.5	α_1	0.2770	0.3116	0.2
0.5	σ	0.2604	0.2271	0.3
1.0	β_0	0.5933	1.0020	0.8
1.0	α	0.0243	0.7032	0.5
1.0	α_1	0.3887	0.3539	0.2
1.0	σ	0.1735	0.1906	0.3

Table S8: **Using MdlInfer to generate uncertainty estimates for SAPHIRE model**

Running time	Parameter	BaseParam ($\hat{\Theta}$)	MdlParam (Θ^*)	Θ_{gt}
1	b	0.4529	0.4696	0.5
1	r	0.2505	0.2414	0.2
2	b	0.4539	0.4789	0.5
2	r	0.2599	0.2311	0.2
3	b	0.4526	0.4692	0.5
3	r	0.2504	0.2409	0.2
4	b	0.4527	0.4691	0.5
4	r	0.2504	0.2409	0.2
5	b	0.4538	0.4681	0.5
5	r	0.2496	0.2412	0.2
6	b	0.4520	0.4684	0.5
6	r	0.2516	0.2417	0.2
7	b	0.4508	0.4694	0.5
7	r	0.2528	0.2407	0.2
8	b	0.4527	0.4675	0.5
8	r	0.2504	0.2415	0.2
9	b	0.4541	0.4775	0.5
9	r	0.2499	0.2316	0.2
10	b	0.4530	0.4678	0.5
10	r	0.2503	0.2418	0.2
mean ± std	b	0.4528 ± 0.0009312	0.4705 ± 0.003879	0.5
mean ± std	r	0.2506 ± 0.0008981	0.2393 ± 0.003980	0.2

Table S9: **Using MdlInfer to generate uncertainty estimates for SEIR+HD model**

Running time	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)	Θ_{gt}
1	β_0	0.5971	0.9255	0.8
1	α	0.0426	0.6739	0.5
1	α_1	0.2770	0.2988	0.2
1	σ	0.2604	0.2834	0.3
2	β_0	0.5947	0.7342	0.8
2	α	0.0505	0.2510	0.5
2	α_1	0.2723	0.2406	0.2
2	σ	0.2720	0.2642	0.3
3	β_0	0.6286	0.7190	0.8
3	α	0.0676	0.2358	0.5
3	α_1	0.2757	0.2482	0.2
3	σ	0.2528	0.2704	0.3
4	β_0	0.6079	0.6995	0.8
4	α	0.0842	0.2360	0.5
4	α_1	0.2862	0.2350	0.2
4	σ	0.2846	0.2673	0.3
5	β_0	0.5959	0.9324	0.8
5	α	0.0406	0.5913	0.5
5	α_1	0.2847	0.2903	0.2
5	σ	0.2743	0.2561	0.3
6	β_0	0.6242	0.9111	0.8
6	α	0.0740	0.5576	0.5
6	α_1	0.2680	0.2893	0.2
6	σ	0.2383	0.2516	0.3
7	β_0	0.9313	0.8088	0.8
7	α	0.4831	0.4750	0.5
7	α_1	0.2797	0.2656	0.2
7	σ	0.2425	0.2655	0.3
8	β_0	0.6265	0.7963	0.8
8	α	0.0972	0.4459	0.5
8	α_1	0.2825	0.2673	0.2
8	σ	0.2561	0.2680	0.3
9	β_0	0.6690	0.7084	0.8
9	α	0.2084	0.2326	0.5
9	α_1	0.2709	0.2360	0.2
9	σ	0.2865	0.2660	0.3
10	β_0	0.6485	1.0348	0.8
10	α	0.0953	0.7135	0.5
10	α_1	0.2797	0.2896	0.2
10	σ	0.2304	0.2655	0.3
mean ± std	β_0	0.6524 ± 0.0957	0.8270 ± 0.1109	0.8
mean ± std	α	0.1244 ± 0.1280	0.4412 ± 0.1815	0.5
mean ± std	α_1	0.2777 ± 0.0057	0.2661 ± 0.0237	0.2
mean ± std	σ	0.2598 ± 0.0184	0.2658 ± 0.0080	0.3

Table S10: **Parameter list for Minneapolis-Spring-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Mar 6-Apr 16	b	0.5167	0.6485
Mar 6-Apr 16	r	0.1440	0.0825
Apr 17-May 21	b	0.4838	0.4707
Apr 17-May 21	r	0.1857	0.0853
May 22-Jun 20	b	0.2873	0.2894
May 22-Jun 20	r	0.1210	0.0788
Jun 61-Jul 14	b	0.4263	0.4413
Jun 61-Jul 14	r	0.1406	0.0996

Table S11: **Parameter list for South Florida-Spring-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Mar 6-Mar 25	b	1.2708	1.3448
Mar 6-Mar 25	r	0.0349	0.0895
Mar 26-Apr 4	b	0.5656	0.5400
Mar 26-Apr 4	r	0.0282	0.0694
Apr 5-Apr 28	b	0.3295	0.2974
Apr 5-Apr 28	r	0.0250	0.0672
Apr 29-May 28	b	0.0394	0.3596
Apr 29-May 28	r	0.0237	0.0688
May 29-Jun 27	b	0.7250	0.5773
May 29-Jun 27	r	0.0214	0.0682

Table S12: **Parameter list for Philadelphia-Spring-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Mar 6-Mar 25	b	1.3602	1.3434
Mar 6-Mar 25	r	0.1538	0.2358
Mar 26-Apr 14	b	0.5006	0.4890
Mar 26-Apr 14	r	0.1705	0.2684
Apr 15-Apr 24	b	0.3583	0.3512
Apr 15-Apr 24	r	0.1221	0.1930
Apr 25-May 24	b	0.3421	0.0980
Apr 25-May 24	r	0.3297	0.1576
May 25-Jun 13	b	0.3052	0.2923
May 25-Jun 13	r	0.0827	0.1378
Jun 14-Jul 15	b	0.3839	0.3679
Jun 14-Jul 15	r	0.0891	0.1499

Table S13: **Parameter list for San Francisco-Spring-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Feb 23-Mar 17	b	0.5270	0.9031
Feb 23-Mar 17	r	0.6531	0.2675
Mar 18-Apr 3	b	0.4834	0.5280
Mar 18-Apr 3	r	0.8883	0.17081
Apr 4-May 13	b	0.3141	0.3325
Apr 4-May 13	r	0.7888	0.1603
May 14-Jun 18	b	0.3741	0.4157
May 14-Jun 18	r	0.8364	0.1802
Jun 19-Jul 4	b	0.3530	0.1926
Jun 19-Jul 4	r	0.8322	0.3671

Table S14: **Parameter list for Minneapolis-Spring-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Mar 6-Apr 16	β_0	0.5190	0.6826
Mar 6-Apr 16	α	0.0266	0.4789
Mar 6-Apr 16	α_1	0.0970	0.0942
Apr 17-May 21	β_0	0.2116	0.2227
Apr 17-May 21	α	0.0441	0.1426
Apr 17-May 21	α_1	0.1862	0.1565
May 22-Jun 20	β_0	0.1181	0.1095
May 22-Jun 20	α	0.4169	0.06278
May 22-Jun 20	α_1	0.3154	0.1244
Jun 61-Jul 14	β_0	0.1562	0.1891
Jun 61-Jul 14	α	0.0309	0.0569
Jun 61-Jul 14	α_1	0.3697	0.1256

Table S15: **Parameter list for South Florida-Spring-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Mar 6-Mar 25	β_0	0.7996	1.0543
Mar 6-Mar 25	α	0.0891	0.3651
Mar 6-Mar 25	α_1	0.1979	0.1897
Mar 26-Apr 4	β_0	0.3185	0.6882
Mar 26-Apr 4	α	0.0184	0.7532
Mar 26-Apr 4	α_1	0.1985	0.2233
Apr 5-Apr 28	β_0	0.1176	0.0972
Apr 5-Apr 28	α	0.0719	0.1268
Apr 5-Apr 28	α_1	0.1721	0.0799
Apr 29-May 28	β_0	0.1286	0.1561
Apr 29-May 28	α	0.0803	0.0387
Apr 29-May 28	α_1	0.2015	0.0890
May 29-Jun 27	β_0	0.1955	0.2495
May 29-Jun 27	α	0.0489	0.0690
May 29-Jun 27	α_1	0.3379	0.0808

Table S16: **Parameter list for Philadelphia-Spring-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Mar 6-Mar 25	β_0	0.8777	1.0582
Mar 6-Mar 25	α	0.0665	0.4069
Mar 6-Mar 25	α_1	0.1481	0.1330
Mar 26-Apr 14	β_0	0.2500	0.2443
Mar 26-Apr 14	α	0.0147	0.0300
Mar 26-Apr 14	α_1	0.1688	0.2475
Apr 15-Apr 24	β_0	0.1642	0.1603
Apr 15-Apr 24	α	0.0347	0.0340
Apr 15-Apr 24	α_1	0.1432	0.2092
Apr 25-May 24	β_0	0.1798	0.1225
Apr 25-May 24	α	0.5218	0.0392
Apr 25-May 24	α_1	0.1857	0.2086
May 25-Jun 13	β_0	0.1576	0.1186
May 25-Jun 13	α	0.4397	0.2285
May 25-Jun 13	α_1	0.2282	0.2470
Jun 13-Jul 15	β_0	0.2263	0.2290
Jun 13-Jul 15	α	0.5223	0.5333
Jun 13-Jul 15	α_1	0.3152	0.3208

Table S17: **Parameter list for San Francisco-Spring-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Feb 23-Mar 17	β_0	0.5380	0.5161
Feb 23-Mar 17	α	0.0916	0.0793
Feb 23-Mar 17	α_1	0.1155	0.1178
Mar 18-Apr 3	β_0	0.2484	0.2415
Mar 18-Apr 3	α	0.1255	0.0753
Mar 18-Apr 3	α_1	0.1935	0.1839
Apr 4-May 13	β_0	0.2401	0.1481
Apr 4-May 13	α	0.7488	0.2411
Apr 4-May 13	α_1	0.3872	0.1800
May 14-Jun 18	β_0	0.1871	0.1991
May 14-Jun 18	α	0.1325	0.1465
May 14-Jun 18	α_1	0.1476	0.1559
Jun 19-Jul 4	β_0	0.2227	0.1856
Jun 19-Jul 4	α	0.0813	0.0331
Jun 19-Jul 4	α_1	0.1566	0.1725

Table S18: **Parameter list for Minneapolis-Fall-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Sep 10-Oct 9	b	0.4155	0.4047
Sep 10-Oct 9	r	0.1180	0.2877

Table S19: **Parameter list for South Florida-Fall-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Oct 15-Nov 14	b	0.4707	0.4373
Oct 15-Nov 14	r	0.0561	0.2372

Table S20: **Parameter list for Philadelphia-Fall-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Sep 16-Oct 15	b	0.3933	0.4008
Sep 16-Oct 15	r	0.2492	0.2474

Table S21: **Parameter list for San Francisco-Fall-20 for SAPHIRE model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Oct 10-Nov 9	b	0.4097	0.4179
Oct 10-Nov 9	r	0.2489	0.2369

Table S22: **Parameter list for Minneapolis-Fall-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Sep 10-Oct 9	β_0	0.1888	0.3883
Sep 10-Oct 9	α	0.0441	0.9145
Sep 10-Oct 9	α_1	0.9983	1.5852

Table S23: **Parameter list for South Florida-Fall-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Oct 15-Nov 14	β_0	0.2709	0.5404
Oct 15-Nov 14	α	0.1063	0.9473
Oct 15-Nov 14	α_1	0.4658	2.7961

Table S24: **Parameter list for Philadelphia-Fall-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Sep 16-Oct 15	β_0	0.2400	0.2883
Sep 16-Oct 15	α	0.2341	0.6342
Sep 16-Oct 15	α_1	1.1527	0.5294

Table S25: **Parameter list for San Francisco-Fall-20 for SEIR+HD model**

Time period	Parameter	BaseParam($\hat{\Theta}$)	MdlParam(Θ^*)
Oct 10-Nov 9	β_0	0.1987	0.4608
Oct 10-Nov 9	α	0.0823	0.7612
Oct 10-Nov 9	α_1	1.1611	1.0386

References

[1] Commercial laboratory seroprevalence survey data. https://www.cdc.gov/coronavirus/2019-ncov/cases-updates/commercial-lab-surveys.html.

[2] Coronavirus in the u.s.:latest map and case count. https://www.nytimes.com/interactive/2020/us/coronavirus-us-cases.html.

[3] Gao, F., and Han, L. Implementing the nelder-mead simplex algorithm with adaptive parameters. *Computational Optimization and Applications 51*, 1 (2012), 259–277.

[4] Hao, X., Cheng, S., Wu, D., Wu, T., Lin, X., and Wang, C. Reconstruction of the full transmission dynamics of covid-19 in wuhan. *Nature 584*, 7821 (2020), 420–424.

[5] Havers, F. P., Reed, C., Lim, T., Montgomery, J. M., Klena, J. D., Hall, A. J., Fry, A. M., Cannon, D. L., Chiang, C.-F., Gibbons, A., et al. Seroprevalence of antibodies to sars-cov-2 in 10 sites in the united states, march 23-may 12, 2020. *JAMA internal medicine 180*, 12 (2020), 1576–1586.

[6] Kain, M. P., Childs, M. L., Becker, A. D., and Mordecai, E. A. Chopping the tail: How preventing superspreading can help to maintain covid-19 control. *Epidemics 34* (2021), 100430.

[7] Lee, T. C. An introduction to coding theory and the two-part minimum description length principle. *International statistical review 69*, 2 (2001), 169–183.

[8] Reinhart, A., Brooks, L., Jahja, M., Rumack, A., Tang, J., Agrawal, S., Al Saeed, W., Arnold, T., Basu, A., Bien, J., et al. An open repository of real-time covid-19 indicators. *Proceedings of the National Academy of Sciences 118*, 51 (2021).

[9] Salomon, J. A., Reinhart, A., Bilinski, A., Chua, E. J., La Motte-Kerr, W., Rönn, M. M., Reitsma, M. B., Morris, K. A., LaRocca, S., Farag, T. H., et al. The us covid-19 trends and impact survey: Continuous real-time measurement of covid-19 symptoms, risks, protective behaviors, testing, and vaccination. *Proceedings of the National Academy of Sciences 118*, 51 (2021).