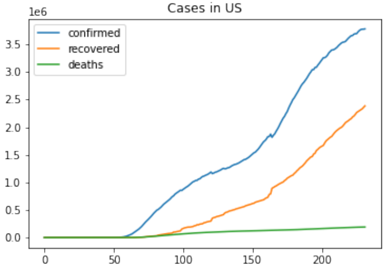
First-principles machine learning modelling of COVID-19

Che-Chia Chang

The COVID-19 Data

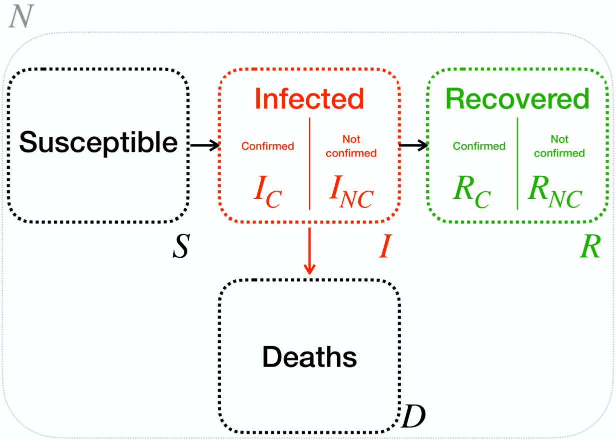
- Used data: COVID-19 Data Repository by the Center for Systems Science and Engineering (CSSE) at Johns Hopkins University (https://github.com/CSSEGISandData/COVID-19)
- Data includes number of:
  1. confirmed cases
  2. recovered cases
  3. deaths cases

  of each day for 188 countries.



The SIRD model

- To describe the data, we may consider the SIRD model to fit it.



- Note that here $N = S + I + R + D$.

The SIRD model

The original ODE system of the SIRD model is written as:

The SIRD model

$$\frac{dS}{dt} = -\frac{\beta IS}{N},$$
$$\frac{dI}{dt} = \frac{\beta IS}{N} - (\mu + \gamma)I,$$
$$\frac{dR}{dt} = \gamma I,$$
$$\frac{dD}{dt} = \mu I,$$

where β, γ, μ, are rates of infection, recovery, and mortality, respectively.

- Note that here β, γ, μ may be dependent of time t.

The discrete SIRD model

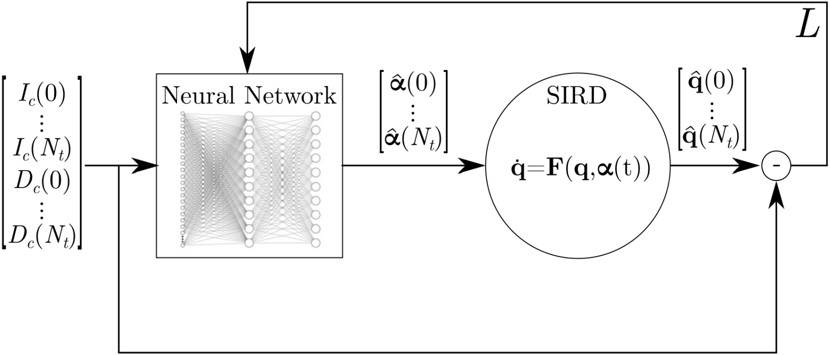
We solve the ode numerically using Euler's method with step size 1, which results in the discrete SIRD model:

The discrete SIRD model

$$
\begin{aligned}
S(t+1) &= S(t) - \frac{\beta(t)I(t)S(t)}{N}, \\
I(t+1) &= I(t) + \frac{\beta(t)I(t)S(t)}{N} - (\mu(t)+\gamma(t))I(t), \\
R(t+1) &= R(t) + \gamma(t)I(t), \\
D(t+1) &= D(t) + \mu(t)I(t),
\end{aligned}
$$

with $I(0)$ and $D(0)$ being the number of confirmed and deaths of the first confirmed day respectively, $S(0) = N - I(0) - D(0)$, $R(0) = 0$.

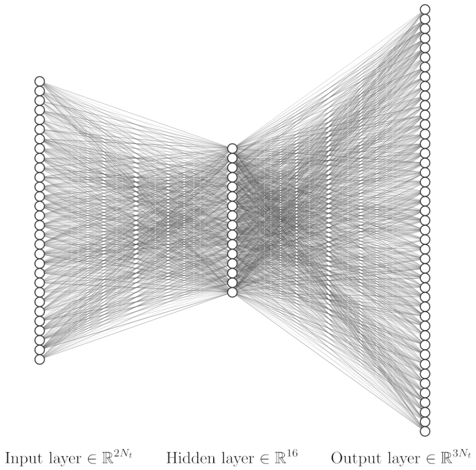
First-principles machine learning model



- $I_c(t)$: infected people = confirmed - recovered - deaths
- $D_c(t)$: deaths
- $\hat{\alpha}(t) = [\hat{\beta(t)}, \hat{\gamma(t)}, \hat{\mu(t)}]$, $\quad \hat{\mathbf{q}}(t) = [\hat{S}(t), \hat{I}(t), \hat{R}(t), \hat{D}(t)]$

The neural network

- A Fully connected neural network with 1 hidden layer and sigmoid activation. Input size : $2(N_t + 1)$, Output size: $3(N_t + 1)$.



Initialization

- We pre-train the neural network to output a constant guess of β_0, γ_0, μ_0.
- β_0, γ_0, μ_0 is found by minimizing

$$\sum_{t=0}^{\tilde{N}_t}(I_c(t) - \hat{I}(t))^2 + 100\sum_{t=0}^{\tilde{N}_t}(D_c(t) - \hat{D}(t))^2$$

  where $\tilde{N}_t$ is the time of initial exponential growth, using optimization package. (For USA, take $\tilde{N}_t = 70$)
- Pre-training uses mean-squared-error with Adam optimizer and step size 0.012, 2500 iterations.

Loss function and training

- The loss function L of the neural network is given by

$$L = \sum_{t=0}^{N_t} \Big((\log(I_c(t) - \log(\hat{I}(t)))^2 + (\log(D_c(t) - \log(\hat{D}(t)))^2 \Big) +$$

$$0.01 \frac{\log(\max(I_c))}{\max(I_c)} \sum_{t=0}^{N_t} \Big((I_c(t)) - \hat{I}(t))^2 + (D_c(t) - \hat{D}(t))^2 \Big) +$$

$$100 \frac{\log(\max(I_c))}{\max(\alpha_0)} \sum_{t=0}^{N_t - 1} (\hat{\beta}(t) - \hat{\beta}(t+1))^2 + (\hat{\gamma}(t) - \hat{\gamma}(t+1))^2$$

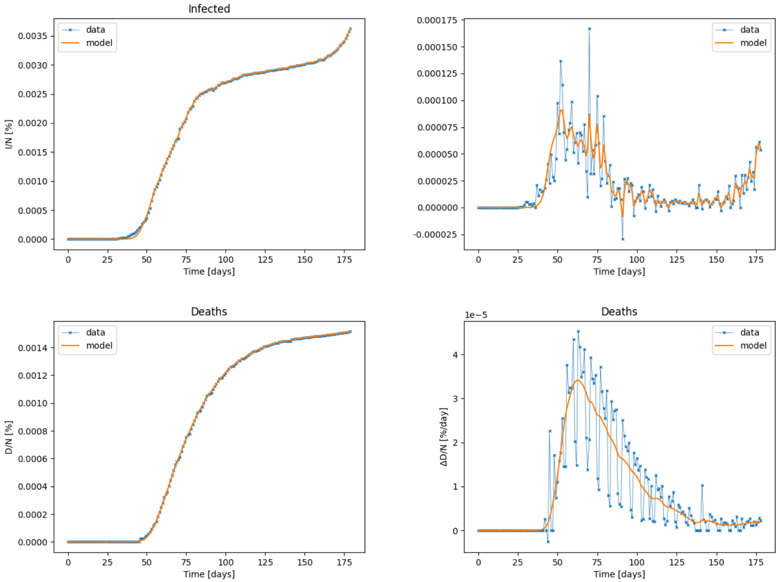
$$+ 100(\hat{\mu}(t) - \hat{\mu}(t+1))^2 +$$

$$100 \frac{\log(\max(I_c))}{\max(\alpha_0)} \Big((\hat{\beta}(0) - \beta_0)^2 + (\hat{\gamma}(0) - \gamma_0)^2 + 100(\hat{\mu}(0) - \mu_0)^2 \Big)$$

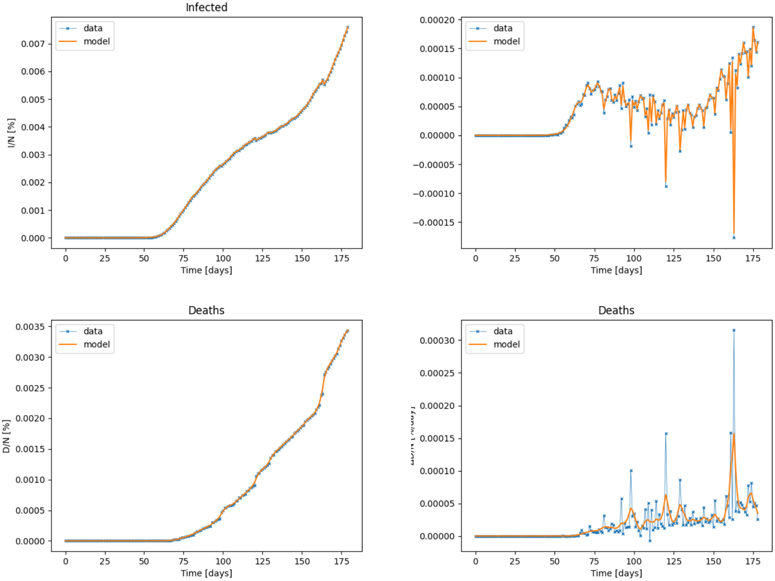
Loss function and training

- We'll train networks of the United Kingdom, Italy, Germany, France, Spain, Belgium, US, and China. The training only involves a single data.
- We train the network using Adam and learning rate 0.00005 with 20000 iterations and 0.00001 with 20000 iterations for $N_t = 179$.

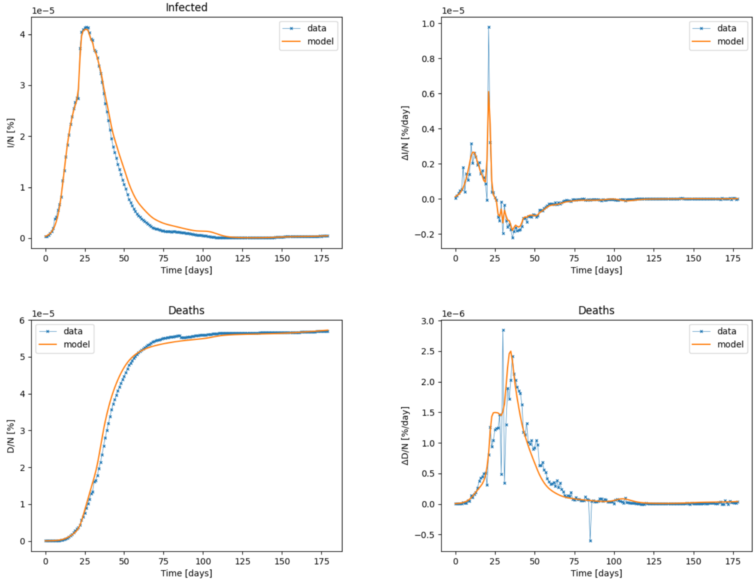
Fitting results(Belgium)



Fitting results(US)



Fitting results(China)



Parameter Results

Non-Machine Learning methods

Future works