SIRD Model for Disease Simulation and Prediction

Krutal Patel, Faiza Shaik
Statistical Modeling and Analysis with Python

JOHNS HOPKINS UNIVERSITY

JOHNS HOPKINS WHITING SCHOOL of ENGINEERING

The Problem

1. Covid-19 outbreak in a small town in India:

Population N = 10,000

2. N = S(I) + I(t) + R(t) + D(t)

S(t) : Susceptible

I(t) : Infected

R(t) : Recovered

D(t) : Deceased

GOAL:

Predict the number of cases that might occur in the next 60 days.

Estimate values of the parameters Β, Κ, γ

Step 1: Determine the Likelihood

- The Normal Distribution was selected to model the Likelihood function.
- When constructing likelihood, assume SIRD model is accurate
  - Any deviations from SIRD are a result of noise with an intensity of σ
- Thus, the number of individuals infected on any day t given by:

$$y_t = I(t) + \mathrm{N}(0, \sigma)$$

- Where I(t) comes from integration of the SIRD model and N(0,σ) represents the error
- The likelihood of observing y_t on day t can be expressed as the PDF

$$\mathcal{L}(y_i \mid \beta, \gamma, \kappa, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-(y_t - I(t))^2}{2\sigma^2}}$$

- The product of this function at each y_i, from the observed data, will provide the Likelihood function at state u_i given that the values of $\beta, \gamma, \kappa, \sigma$ are updated for each state.

Step 2: Determine the Prior and Bounds

Arbitrarily choose the Normal Distribution to model each of the parameters given historical data

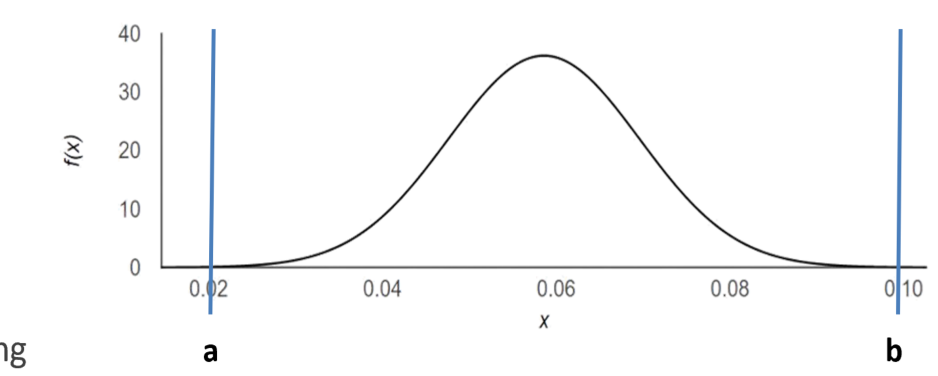
Process

1.Compute mean and std of each parameter's data

2. Determine appropriate bounds

3.Correct normal distribution by multiplying by constant k

Example for Gamma Parameter



Determining Distribution for Sigma

1.Challenges with determining the bounds and distribution for sigma

2. Considered Context: Population of 10,000

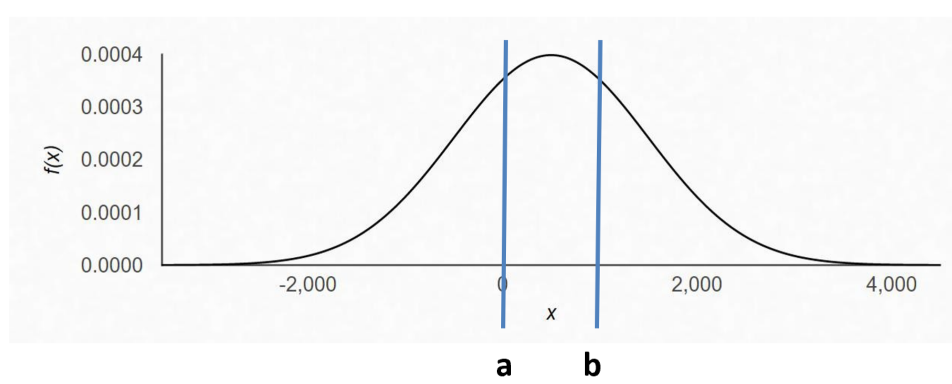
3.Use Prior for sigma such that probabilities are almost uniform, and range is wide!

Our Approach

Assume sigma is Normally distributed with mean 500 and std 1000

Multiply by constant k to reduce distribution to bounds [0,1000]

Sigma: Normal(500,1000)



Step 3: Determine the Posterior

From Bayesian Inference:

Posterior = Prior x Likelihood

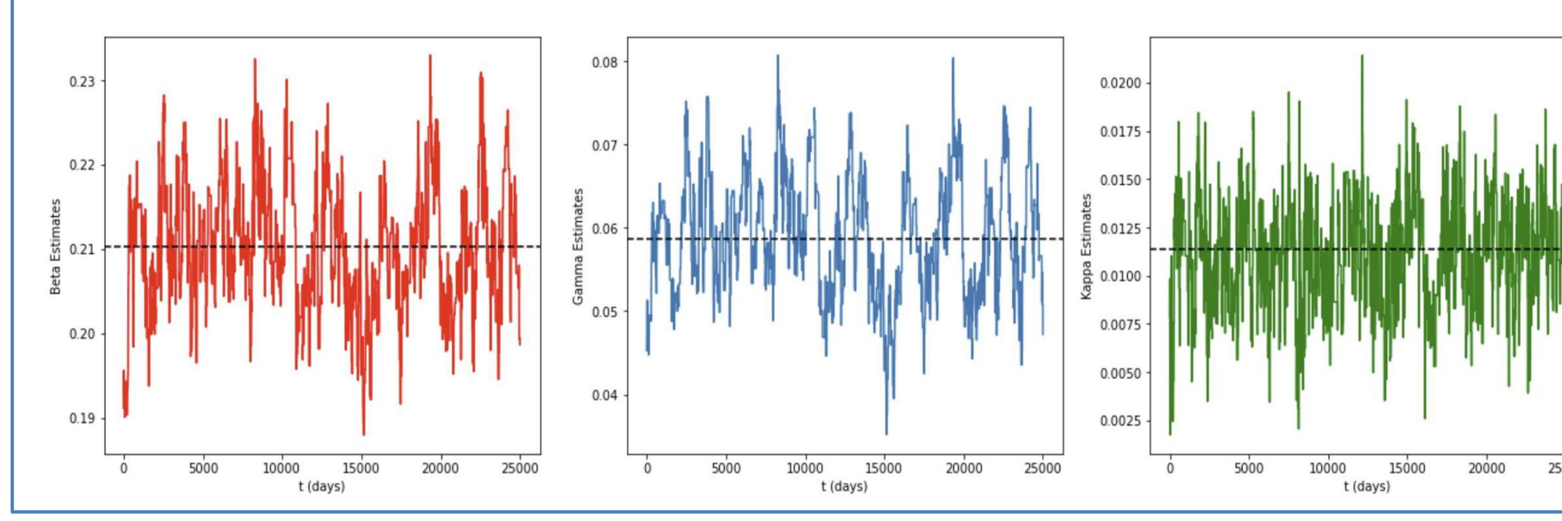
```python
def get_pi_val (u):
    prior = get_prior(u)
    likelihood = get_likelihood_val(u)
    return prior*likelihood
```

Step 4: MCMC loop!

N_steps = 25,000

U_i = randomly generated initial state given bounds

Acc_factor = 0.1093



90% Confidence Intervals

The 90% confidence intervals for the parameters were:

- **Beta:** (0.184, 0.227)
- **Gamma:** (0.0386, 0.0762)
- **Kappa:** (0.00249, 0.0212)
- **Sigma:** (15, 95.389)

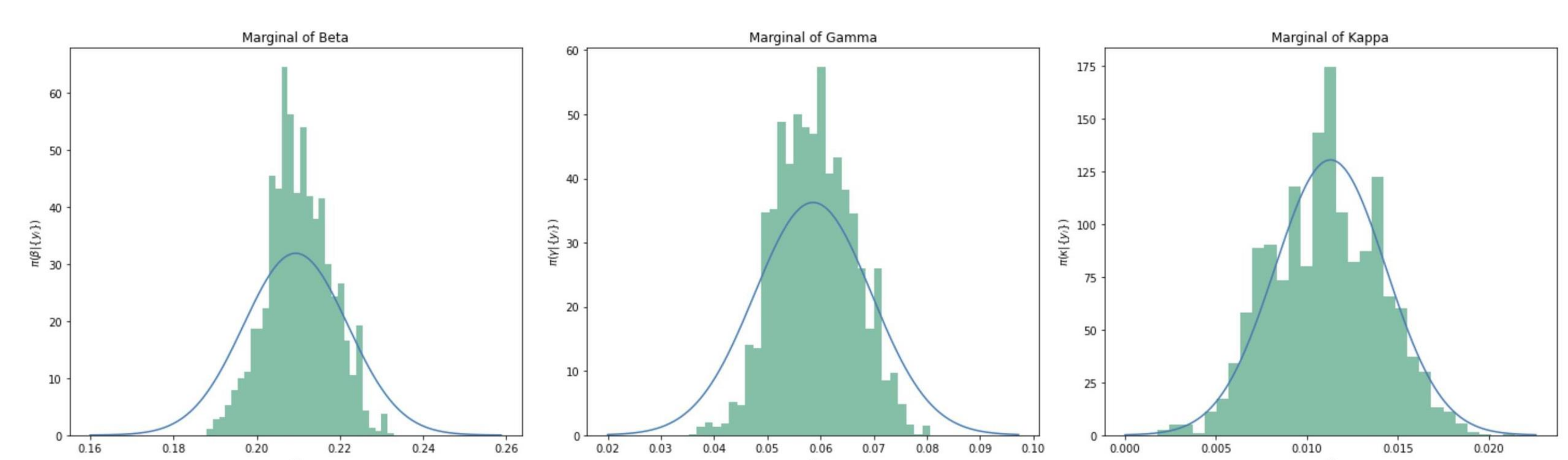
The estimate for each population parameter is within the 90% confidence interval, suggesting that the estimates are plausible

Beta = 0.210, Gamma = 0.0587,

Kappa = 0.0114, Sigma = 68.431

Posterior Histograms

Comparing Marginal Distributions of each parameter with chosen prior distributions.



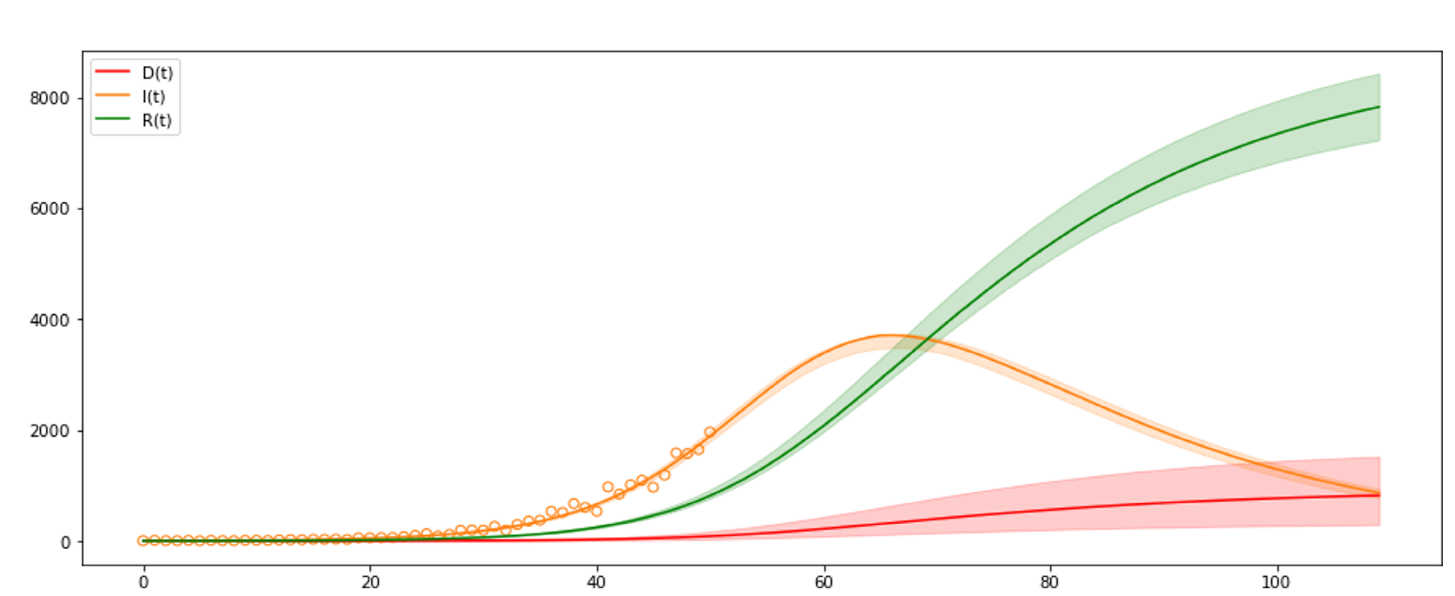
Part 4: Predicted Outcomes

- Projection indicates a peak in infections at Day 65

Recommendations to Policy Makers:

- Reduce the number of cases at the peak
  - Enforce mask mandates to slow spread
  - Put in place stay-at-home and social distancing policies to minimize contact between healthy and infected individuals
- If these recommendations are not followed, it is seen that after Day 80 there is a time period of decline in infection rates, but an increase in recovery rates
  - The increase in recover can be attributed to herd immunity.

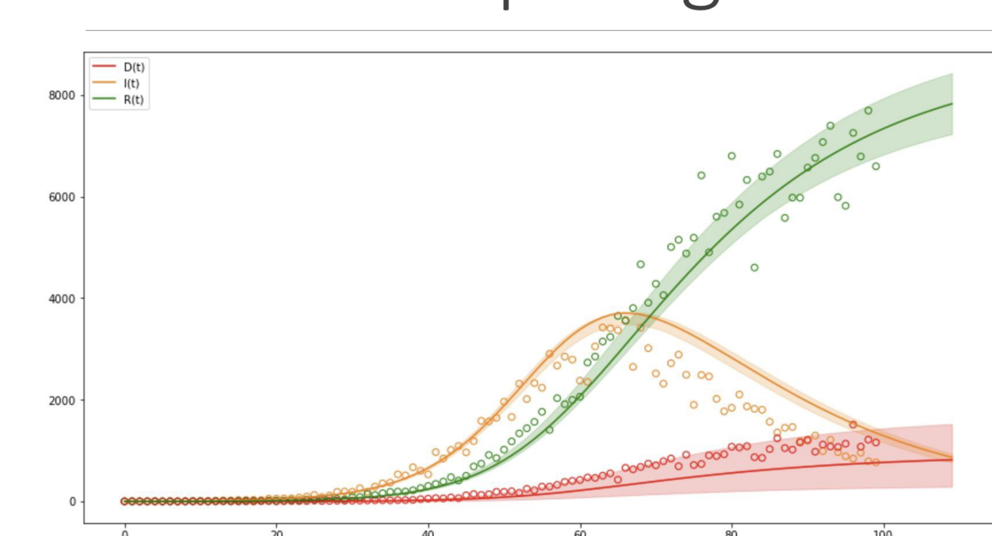
Part 4: Predicted Outcomes



Future projection of I(t), R(t), and D(t) computed using 100 parameter sets sampled from the posterior distribution.

The envelope is computed from the maximum and minimum value of 100 trajectories at each time step

Part 5: Comparing Predictions



Accuracy Rates within displacement errors

At t = 50 days, our model predictions

Given the context, we accepted this model since providing only a slight overestimate of the infections is ideal since we don't want to underestimate the severity of Covid-19 spread and infection

Acknowledgements

Team Project as part of 540.382 course at Johns Hopkins University.

Thank you to Dr. Edison for project idea, guidelines and support in the course.

https://pubs.acs.org/doi/10.1021/acs.iecr.0c04754