Network Effect: True Spread through NYC

...

The Data Hounds - Deepak Ravi and Joanne Chen

Problem - Informing Government Policy

- Certain communities are being affected worse than others due to COVID19
- A **reopening plan needs to account for network effects** of how different communities are being affected and how the virus travels
- Achieving herd immunity or snuff-out before a second wave

How can we tackle this?

We need local focus but we live in the world's most connected city

Granular forecasting, not just statistics

A place to test theories

How can we tackle this?

We need local focus but we live in the world's most connected city

Granular forecasting, not just statistics

A place to test theories

NETWORK

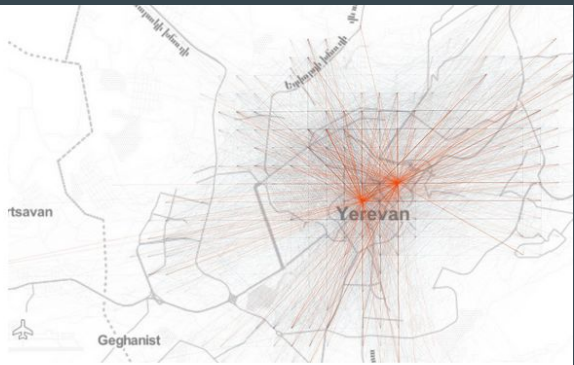
Inspiration

Modelling the coronavirus epidemic in a city with Python

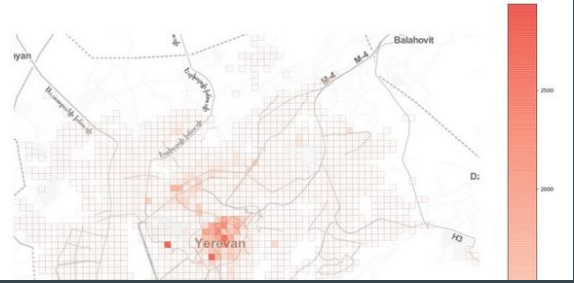
Are cities prepared for epidemics?

Gevorg Yeghikyan Follow
Feb 4 · 10 min read ★

The recent 2019-nCoV Wuhan coronavirus outbreak in China has sent shocks through financial markets and entire economies, and has duly triggered panic among the general population around the world. On 30 January 2020, 2019-nCoV was even designated a global health emergency by the World Health Organization (WHO). At the time of this writing, no specific treatment verified by medical research standards has yet been discovered. Moreover, some key epidemiological metrics such as the basic



Further, if we look at the total inflow to the grid cells, we see a more or le
monocentric spatial organisation with some cells with high daily inflow
located off the center:



Data

Census Tracts

MTA turnstile data

Taxi Trips

CDC Social Vulnerability Index (SVI)

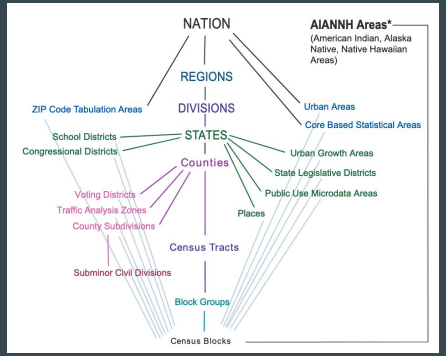
NYC COVID testing data

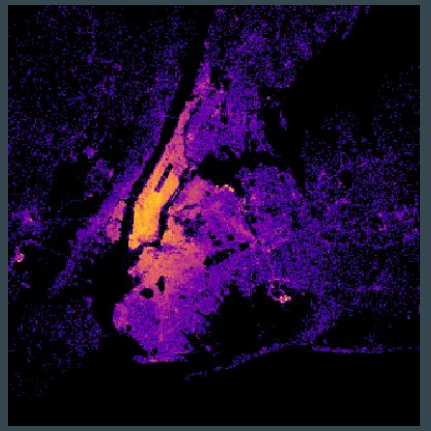
Where?

Who?

How?

Why?

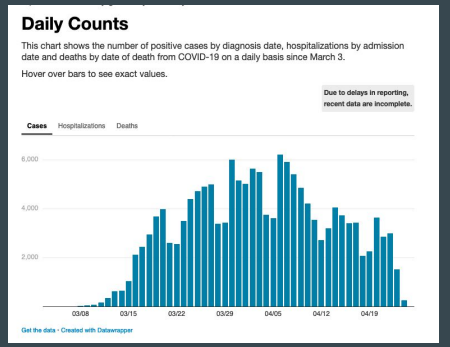




Journal of Homeland Security and Emergency Management

Volume 8, *Issue* 1 2011 *Article* 3

A Social Vulnerability Index for Disaster Management

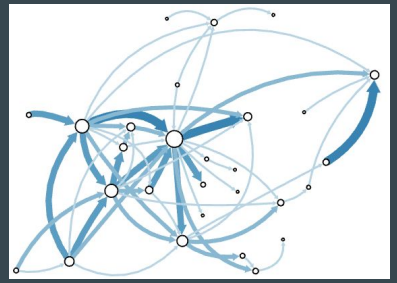


Too Much Data

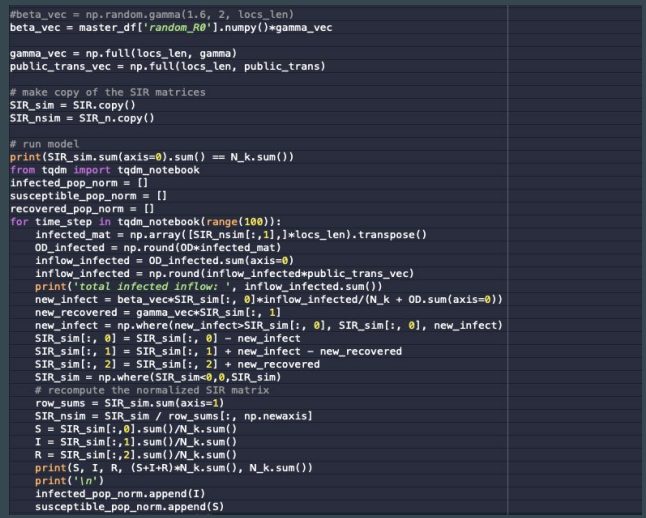
2166 Census Tracts → 4,691,556 Flows

Origin-Destination Matrix

SIR Model



	2	3	4	5	6	7	8	9
0	0	0	864.959	0	0	0	0	0
1	1003.26	570.512	0	1091.61	0	269.458	0	0
2	0	829.707	0	0	0	0	0	0
3	1047.46	0	0	1485.72	0	0	0	0
4	0	0	0	0	545.563	0	590.535	372.56
5	0	844.931	0	0	0	886.895	0	0
6	0	0	707.174	0	0	0	995.112	0
7	0	0	0	624.998	0	0	0	0
8	0	0	466.561	0	851.182	0	0	604.385
9	0	0	357.991	0	0	0	493.975	0
10	0	0	626.813	0	0	0	785.5	394.726
11	0	0	0	0	0	333.412	0	0



```
#beta_vec = np.random.gamma(1.6, 2, locs_len)
beta_vec = master_df['random_R0'].numpy()*gamma_vec

gamma_vec = np.full(locs_len, gamma)
public_trans_vec = np.full(locs_len, public_trans)

# make copy of the SIR matrices
SIR_sim = SIR.copy()
SIR_nsim = SIR_n.copy()

# run model
print(SIR_sim.sum(axis=0).sum() == N_k.sum())
from tqdm import tqdm_notebook
infected_pop_norm = []
susceptible_pop_norm = []
recovered_pop_norm = []
for time_step in tqdm_notebook(range(100)):
    infected_mat = np.array([SIR_nsim[:,1],]*locs_len).transpose()
    OD_infected = np.round(OD*infected_mat)
    inflow_infected = OD_infected.sum(axis=0)
    inflow_infected = np.round(inflow_infected*public_trans_vec)
    print('total infected inflow: ', inflow_infected.sum())
    new_infect = beta_vec*SIR_sim[:, 0]*inflow_infected/(N_k + OD.sum(axis=0))
    new_recovered = gamma_vec*SIR_sim[:, 1]
    new_infect = np.where(new_infect>SIR_sim[:, 0], SIR_sim[:, 0], new_infect)
    SIR_sim[:, 0] = SIR_sim[:, 0] - new_infect
    SIR_sim[:, 1] = SIR_sim[:, 1] + new_infect - new_recovered
    SIR_sim[:, 2] = SIR_sim[:, 2] + new_recovered
    SIR_sim = np.where(SIR_sim<0,0,SIR_sim)
    # recompute the normalized SIR matrix
    row_sums = SIR_sim.sum(axis=1)
    SIR_nsim = SIR_sim / row_sums[:, np.newaxis]
    S = SIR_sim[:,0].sum()/N_k.sum()
    I = SIR_sim[:,1].sum()/N_k.sum()
    R = SIR_sim[:,2].sum()/N_k.sum()
    print(S, I, R, (S+I+R)*N_k.sum(), N_k.sum())
    print('\n')
    infected_pop_norm.append(I)
    susceptible_pop_norm.append(S)
```

1 The Basic Reproduction Number in a Nutshell

The basic reproduction number, $\mathcal{R}_0$, is defined as the expected number of secondary cases produced by a single (typical) infection in a completely susceptible population. It is important to note that $\mathcal{R}_0$ is a dimensionless number and not a rate, which would have units of time^{-1}. Some authors incorrectly call $\mathcal{R}_0$ the "basic reproductive rate."

We can use the fact that $\mathcal{R}_0$ is a dimensionless number to help us in calculating it.

$$\mathcal{R}_0 \propto \left(\frac{\text{infection}}{\text{contact}}\right) \cdot \left(\frac{\text{contact}}{\text{time}}\right) \cdot \left(\frac{\text{time}}{\text{infection}}\right)$$

More specifically:

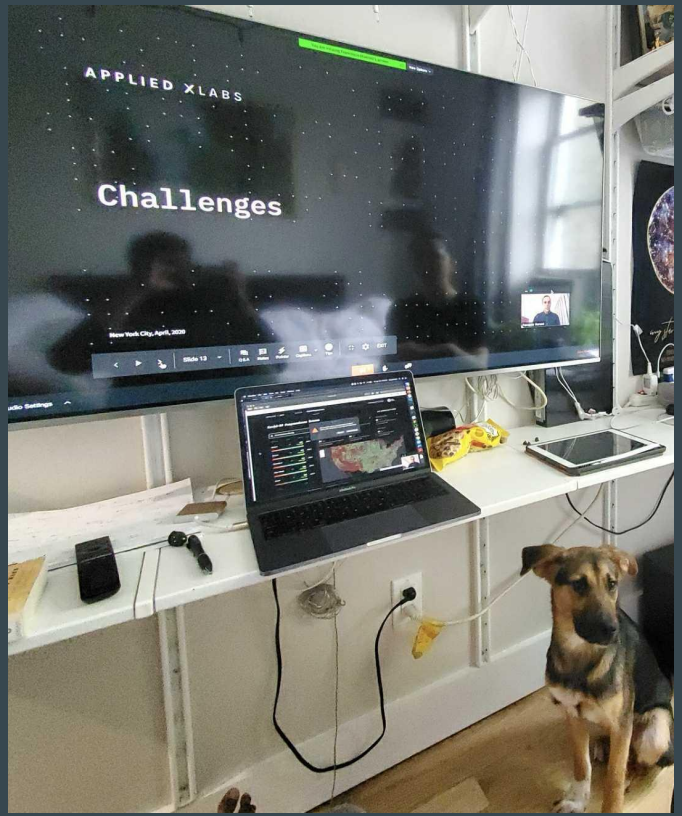
$$\mathcal{R}_0 = \tau \cdot \bar{c} \cdot d \tag{1}$$

where τ is the transmissibility (i.e., probability of infection given contact between a susceptible and infected individual), $\bar{c}$ is the average rate of contact between susceptible and infected individuals, and d is the duration of infectiousness.

Next Steps

- More Computation Time
- Smarter Flow
- Scenario Planning
  - Full Open
  - Staggered Opening
- Expansion

Thank you



Our data hound