
# Weekly report IDCGP

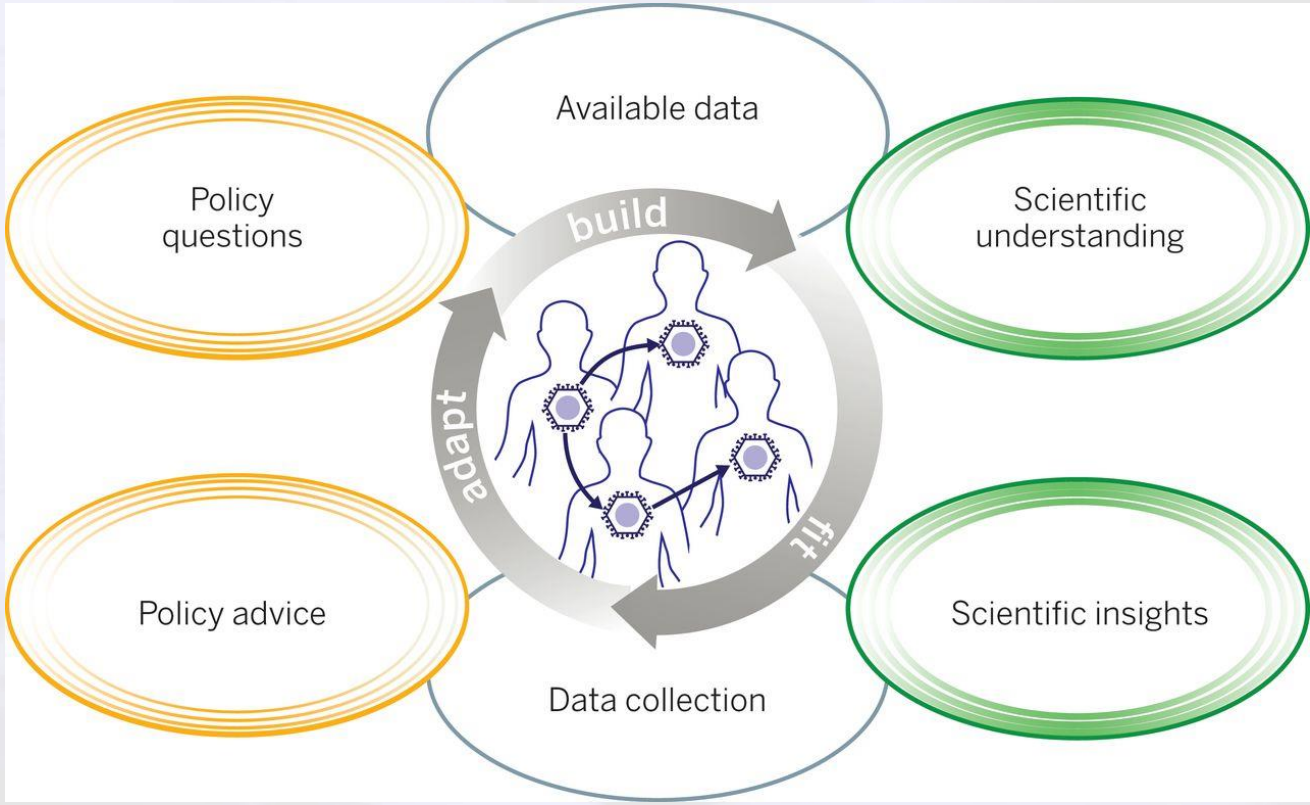
SOFTWARE ENGINEERING

YONGYU CHEN

YONGYU.CHEN@MAIL.RU

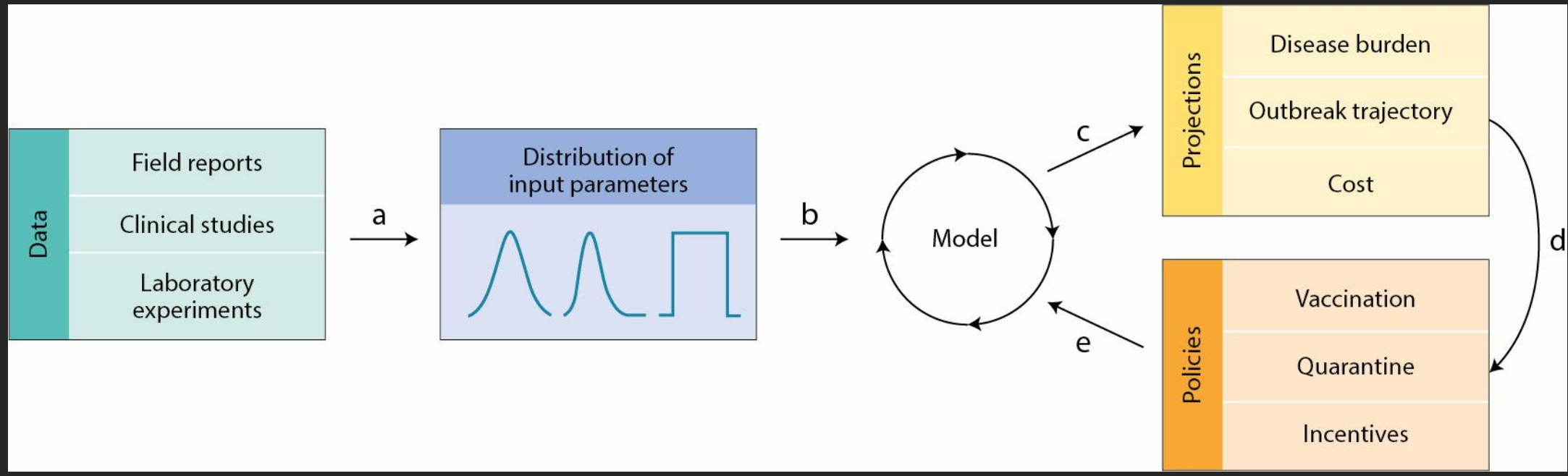
# Outline

- Session 1. Project goal, review
- Session 2. Project description
- Session 3. Architecture
- Session 4. Current results
- Session 5. Future plan



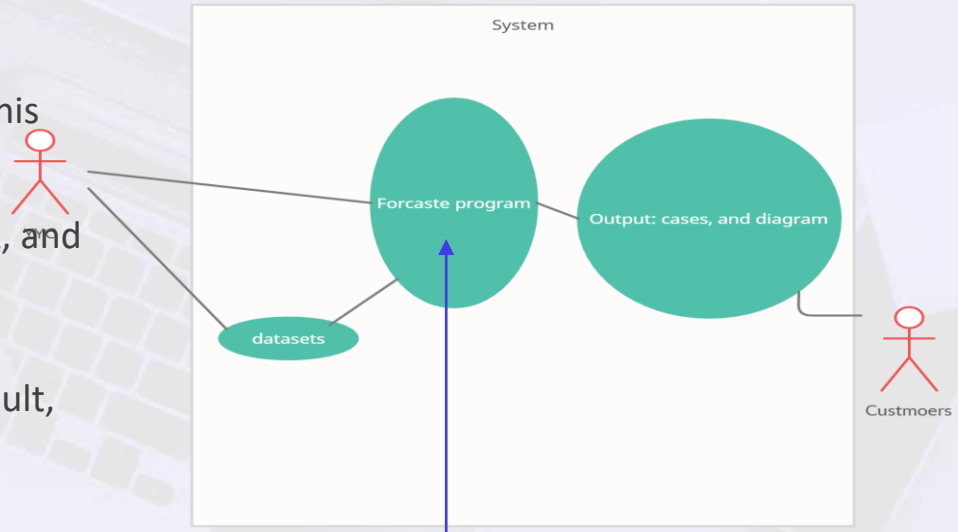
# Project goal, review

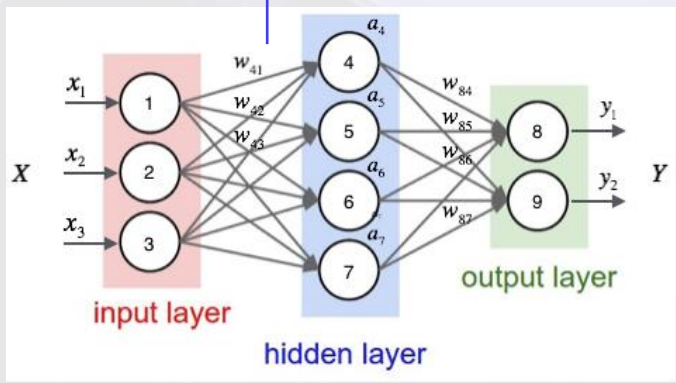
The software product is intended to predict characteristics of both seasonal epidemics and future pandemics. Accurate and timely infectious disease forecasts could aid public health responses by informing key preparation and mitigation efforts, for example covid-19.



# Project description

- We use the updated data sets from the official to forecast the pandemic of covid-19.
- The input is a 40+ dimensional data, including, date,total_cases,total_deaths,new_deaths,,total_cases_per_million,new_cases_per_million,new_cases_smoothed_per_million,total_deaths_per_million,new_deaths_per_million,new_deaths_smoothed_per_million,reproduction_rate,icu_patients,icu_patients_per_million,hosp_patients,hosp_patients_per_million,weekly_icu_admissions,weekly_icu_admissions_per_million,weekly_hosp_admissions,etc.
- The output is a 1-dim data, new case, but we may increase the dimension later, including the the data for new dates, but it will be hard to train this kind of network.
- We use 3 layers of neural network, and use the updated techniques to accelerate the training.
- We do not only care about the result, we also think deeply of the mechanism(most of which are unsolved), such as,
  - Why this techniques(deep, neural numbers, etc.) is better?
  - Why the loss stops to decrease?
  - Why should be consider for the datasets?
  - Why the results will be this?



# Architecture

CPU, x86
But arm supports


GPU, Nvidia, Cuda
AMD not supports


Pytorch, with cuda support, Based on Python

PyTorch

# Current results

1, We cleaned the data, which is an important step for machine learning, because the machine is not clear enough, we should try to make it more explicitly.
2, we selected the data what we want, changed the data into pytorch tensor, we then according to the dimension, we built a two layers  of neural network.

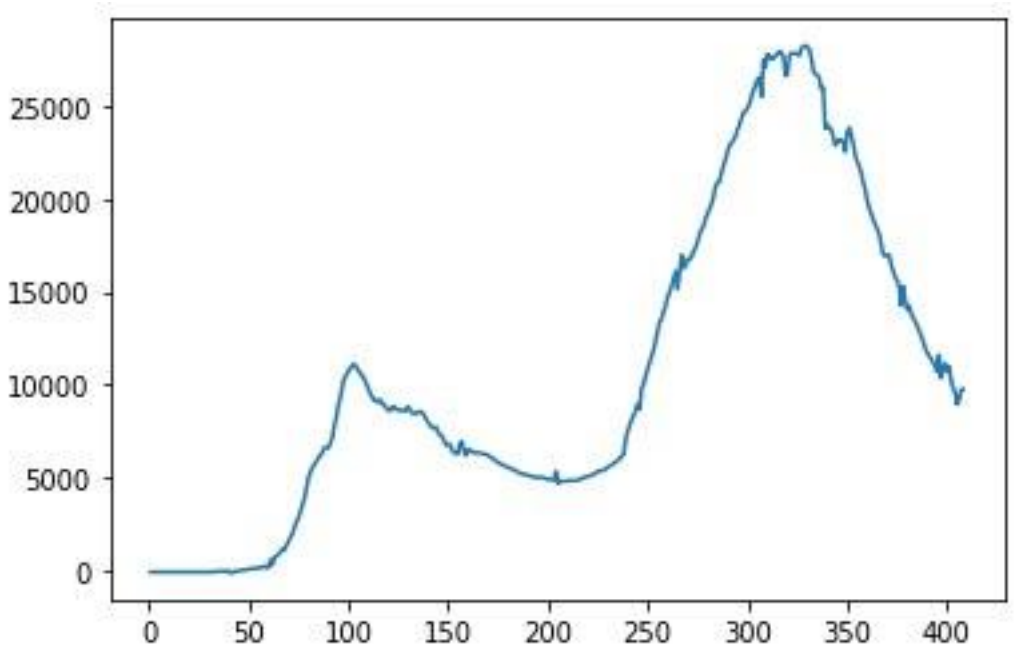
Firstly, we used the traditional method with numpy, and we found that, the loss is so large, and it's unacceptable.

Secondly, we changed the data into pytorch tensor, and we defined our autograd function, then we had a better result.

Thirdly, we added some optimization methods, the loss convergences more quickly, it became acceptable.

Lastly, we defined our own neural network, the results were worse again. We were still thinking about it.

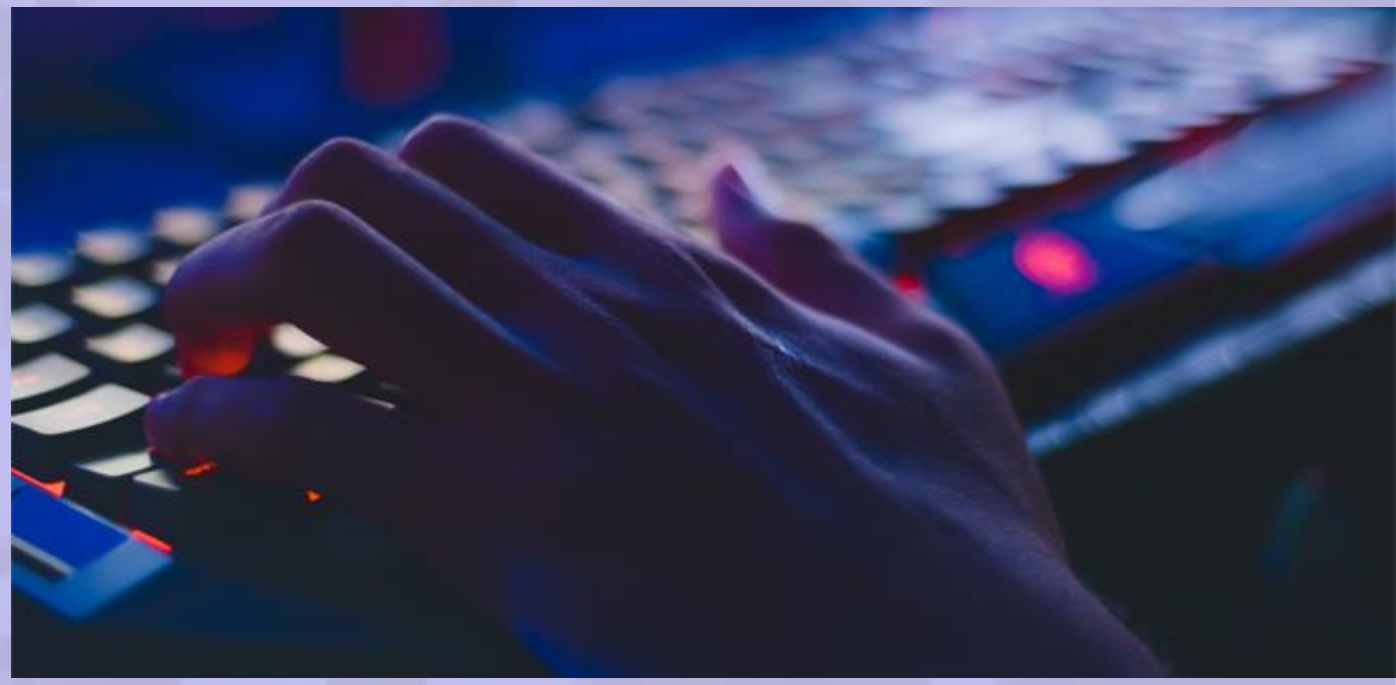
Loss is about 140 per day




The best result we got, using the existing data to predict the current and past cases

# Future plan

- We will continue to optimize it.
- And we started to predict the future cases, and we faced some difficult problems, we do not have such dimension data for future cases.
- We came up with some solutions, but it’s not easy to implement them, we will still try to do that.
- We will improve our graphical output, making it easier to understand.



# Thank You!

Man is only a reed, the weakest in nature, but he is a thinking reed.

Blaise Pascal