Different Forecasting Models for COVID-19

Ann Gerber, Tobias Hofmänner, Bernhard Kratzwald, Mathias Gassner

What to expect

We have analyzed, implemented, tested and compared four different forecasting models:

- SIRD-Model
- Linear-Regression
- Auto-regressive model
- Neural network

SIRD-Model Recap

$$\frac{dS}{dt} = -\frac{\beta IS}{N},$$

$$\frac{dI}{dt} = \frac{\beta IS}{N} - \gamma I - \mu I,$$

$$\frac{dR}{dt} = \gamma I,$$

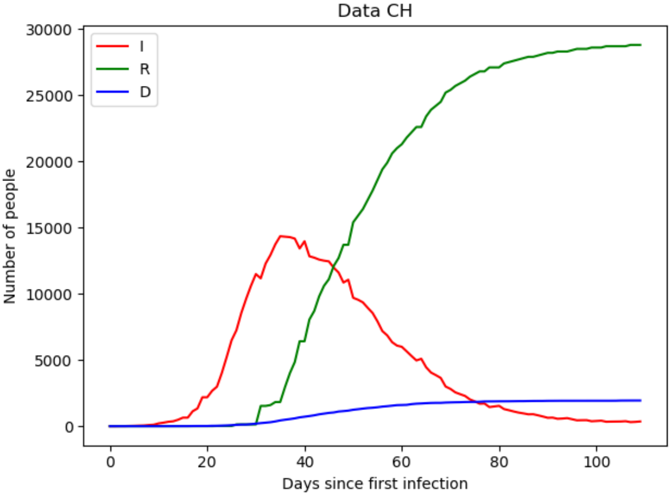
$$\frac{dD}{dt} = \mu I,$$

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

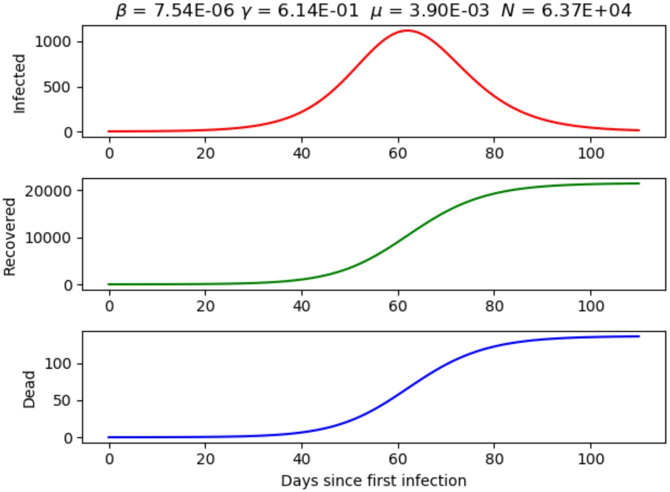
SIRD-Model Problem and Approach

- **Problem:** SIRD Models are very hard to fit. I.e how do we get the parameters β, γ and μ?
- **Approach:** Use a Neural Network to not predict the data directly but to estimate the parameters.
- **Advantage:** We can create the training data ourselves. => CPU is the limit not the amount of data we gathered from countries.
- **Did it work?:** No not really...

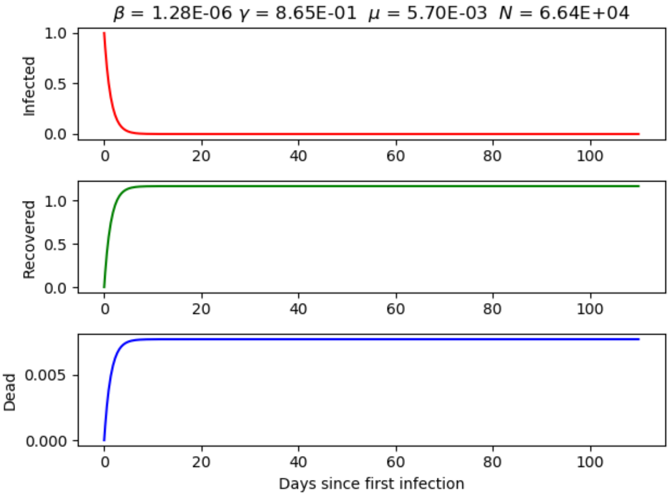
SIRD-Model Creating a Training Set



SIRD-Model Creating a Training Set



SIRD-Model Creating a Training Set



SIRD-Model Problems with my Approach and Outlook

- **Problem 1:** Hard to create a training set
- **Problem 2:** SIRD might not be the best model for COVID-19 (Does not take into account: Incubation period and implemented measures)
- **Do we think it could work?**: Yes. The key is to get a better feeling for the SODE in order to create a better training set.

Ordinary Least Squares Linear Regression (OLS)

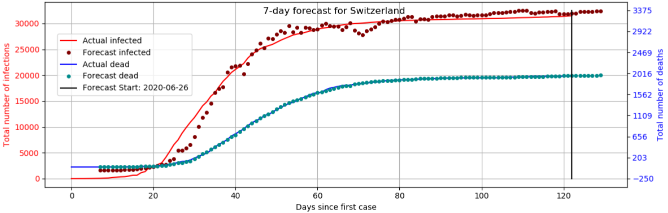
Fit the coefficients to **minimize the residual sum of squares** between the observed values in the dataset, and the targets predicted by the linear approximation

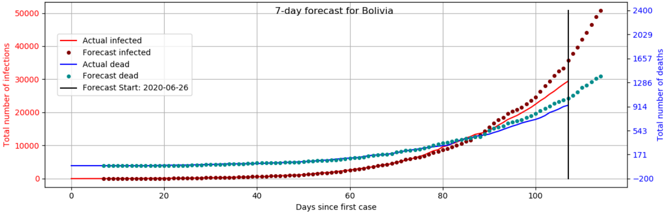
Ordinary Least Squares Linear Regression (OLS)

- Model of the type
  $I(t^*) = \alpha_I R(t^* - \Delta t) + \beta_I D(t^* - \Delta t) + \gamma_I I(t^* - \Delta t)$
  $\rightarrow$
  y_prediction = I(t^*)
  X_input = [R($t^* - \Delta t$), D($t^* - \Delta t$), I($t^* - \Delta t$)]
- Re-train the model to obtain predictions for I, R and D
- Advantage: Can take various input parameters into account
- Disadvantage: Fitted coefficients are time-independent

Ordinary Least Squares Linear Regression (OLS)

Performance





Arima Model with *skits*

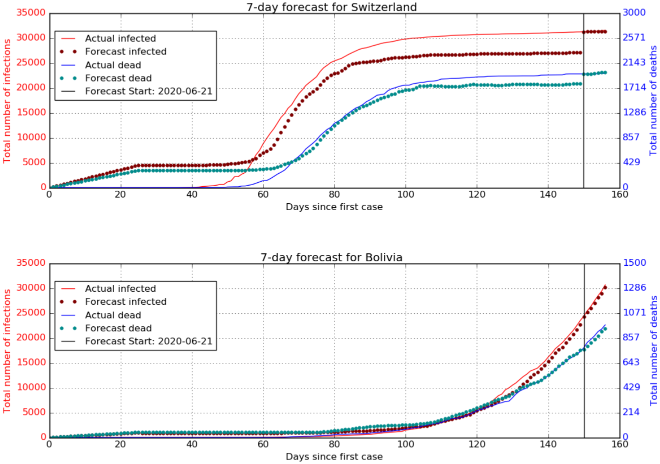
- **ARIMA**: **A**uto **R**egressive **I**ntegrated **M**oving **A**verage
  - **What?** Class of models that 'explains' a given time series based on its own past values
  - **Why?** Any time series (non-seasonal, not white noise) can be modeled with ARIMA models
  - **How?** Implementation with SciKit-learn and skits libraries
- $y_t = c + \phi_1 \Delta y_{t-1} + \phi_2 \Delta y_{t-2} + \ldots + \phi_p \Delta y_{t-p}$
- $X = \begin{pmatrix} \Delta y_0 & \Delta y_1 & \Delta y_2 & \Delta y_3 \\ \Delta y_1 & \Delta y_2 & \Delta y_3 & \Delta y_4 \\ \Delta y_2 & \Delta y_3 & \Delta y_4 & \Delta y_5 \\ \Delta y_3 & \Delta y_4 & \Delta y_5 & \Delta y_6 \end{pmatrix}, y = \begin{pmatrix} y_4 \\ y_5 \\ y_6 \\ y_7 \end{pmatrix}, \quad \Delta y_i = y_i - y_{i-1}$
- **skits**: **S**cikit-learn-**I**nspired **T**ime **S**eries models
  - **What?** library for time series modeling with similar API to SciKit-learn
  - **Why?** composability, manageability
  - **How?** pipelines, preprocessors, regressors

Arima Model with *skits*

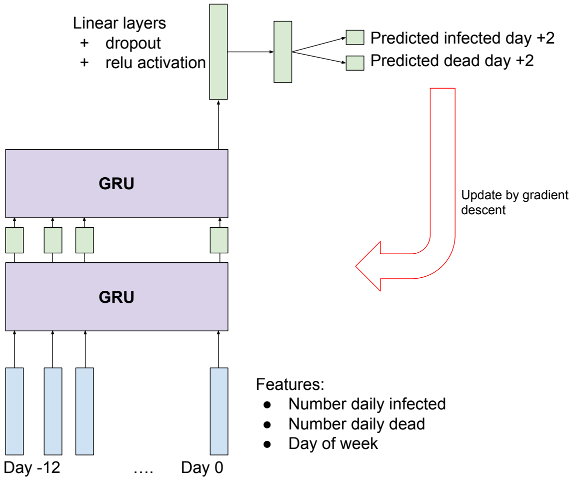
- **ARIMA**: **A**uto **R**egressive **I**ntegrated **M**oving **A**verage
  - **What?** Class of models that 'explains' a given time series based on its own past values
  - **Why?** Any time series (non-seasonal, not white noise) can be modeled with ARIMA models
  - **How?** Implementation with SciKit-learn and skits libraries
- **skits**: **S**ci**k**it-learn-**I**nspired **T**ime **S**eries models
  - **What?** library for time series modeling with similar API to SciKit-learn
  - **Why?** composability, manageability
  - **How?** pipelines, preprocessors, regressors

Arima Model with *skits*

Performance



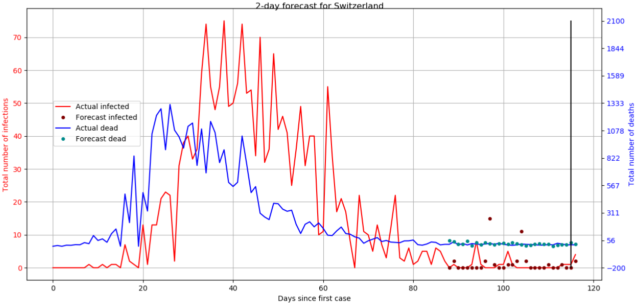
GRU Model: Predictions with Neural Networks



GRU Model: Training and prediction

1. Generate training data for past days
   - X = feature vector for past 12 days
   - y = ground-truth in 2/5/14 days
2. Train network on all countries
3. Fine-tune network for every country specifically
4. Make prediction

GRU Model: Two-day prediction for Switzerland



Quantitative Comparison

- Comparison of 2-day prediction over all countries
- We compare the MAE and MSE relative to the actual number of cases

	Daily Infected		Daily Deaths	
	Relative MSE	Relative MAE	Relative MSE	Relative MAE
OLS	0.934	0.203	1.259	0.254
ARIMA	0.008	0.034	0.005	0.028
GRU	1.875	0.902	0.287	0.354

Qualitative Comparison of the models

Brainstorming comparison, advantages & disadvantages of different models

- Ordinary Least Squares Linear Regression
  - Negative: fitted constants are time-independent
  - Positive: could include various parameters, e.g. population,...
  - Combination with Auto Regression?
- ARIMA Model
  - Negative: Predictions based only on previous values
  - Positive: No parameter tuning
- Neural Network
  - Negative: Dependent on hyperparameters
  - Negative: Difficult for interpretation
  - Positive: Predict both outcomes in one model
  - Positive: Flexible w.r.t. to additional features

Thank you for your Attention!

Questions?

GitHub: `https://gitlab.ethz.ch/kratzwab/covid-datathon`