

## Model overweging

- Long short term memory (LSTM): (simple 1-2 layers) Niek
- Prophet (Meta/Facebook): <https://facebook.github.io/prophet/> Daan
- Temporal Convolutional Network (TCN): David
- Lineaire regressie aanpak: Daniel
- GRU?
- Transformer lite: (andere model architectuur, data hungry).

## Resources

- <https://scikit-learn.org/stable/>

## Nature van de data

De waarden zijn time dependent, we mogen dus geen toekomst waarde gebruiken om retroactief waarden te bepalen omdat deze waarden niet indepedent zijn. Daarom gebruiken we een split waarvan de 'nieuwste' waarden test zijn (in een percentage split).

Er is geen zekerheid of de test data op 9 Mei dezelfde gemiddelde heeft, de aanname is dat alleen het patroon van (klein naar groot) voorkomt in die uiteindelijke test data.

## Normaliseren van de data

Als de uiteindelijke test data erg afwijkt rond het gemiddelde uit de training dan faalt dat op de normalisatie.

## Data mining

Eventueel beter de data zelf begrijpen om aannames te nemen om een model zodanig in te richten voor een specifiek doeleinde.

## Complicaties

'BlockingTimeSeriesSplit' als aanpak voor splitsen van de training test data voor lokaal trainen. De grootte van de window kan daarnaast ook nog aangepast worden om te kijken vanaf welke grootte trainingsdata je genoeg hebt.

Voor  $b$  moeten we in overweging nemen de aantal stappen die we terug redeneren. Dit is een parameter om te optimaliseren. Daarnaast de eerste paar punten zullen geen 'geschiedenis' hebben om the kunnen zien, NaN rows droppen (hoeveelheid dropped optimaliseren), afweging omdat het ten koste gaat van de trainings data.

Correlatieanalyse op de 'lags' om te kijken welke lags het meest belangrijk is. Eruit halen wat de importance is van lags in deze correlatie matrix. De aanname dat hoog aantal lags misschien verkill worden. Hoog positieve en negatieve waarden dragen bij, maar zodra ze samenvloeien (naar 0) dan zijn deze niet belangrijk. De afweging is namelijk hoe meer lags hoe minder de training data wordt.

### **AI statement**

We ask ChatGPT to recommend models for especially low sample count data.