Bayesian network

<https://www.who.int/bulletin/volumes/90/6/11-086009.pdf>

Key Points:

* Modelled each district as either influencing or influenced
  + To calculate the conditional probabilities – either modeled as an outbreak or no outbreak (based on the recommended thresholds set by the WHO (10 per 100,000 inhabitants or 5 cases per 100,000 for smaller villages; alert threshold are 5 and 2 per 100,000 for larger and smaller villages respectively)
* Statistical tools allow us to determine unknown patterns that mathematical relationships which model the transmission of the disease don’t express
* Bayesian network made no allowance for time needed for inter-district influences to reveal themselves as changes in meningitis incidence
* Is it better to develop a Bayesian network for each year?
* How can we use re-sampling simulation techniques to simulate and replicate observations
  + Allows lack of large longitudinal records – can create and model as such

SCIR Model for Meningitis

<https://core.ac.uk/download/pdf/51291793.pdf>

* Typical SIR model = susceptible infected recovered

A screenshot of a cell phone

Description automatically generated

* R and alpha represent a rates – they’re positive
* I, S, and R represent populations

Mathematical model for the transmission of meningitis

<https://www.hindawi.com/journals/cmmm/2018/2657461/>

SIR model for meningitis

<https://dergipark.org.tr/en/download/article-file/814793>

SIS models

<https://institutefordiseasemodeling.github.io/Documentation/general/model-si.html>

<https://www.nature.com/articles/s41598-019-52351-x>

SIR model

<http://mathworld.wolfram.com/SIRModel.html>

SEIR Model

<https://institutefordiseasemodeling.github.io/Documentation/general/model-seir.html>

Specific Meningitis

<https://www.researchgate.net/publication/263741627_Climate_Change_and_Cerebrospinal_Meningitis_in_the_Ghanaian_Meningitis_Belt>

<http://bayesiandeeplearning.org/2018/papers/151.pdf>

Attention for Time Series Article

<https://towardsdatascience.com/attention-for-time-series-classification-and-forecasting-261723e0006d>

Attention Based Mechanisms for Time-Series Prediction

LSTNET

<https://arxiv.org/abs/1703.07015>

GitHub Implementation

<https://github.com/laiguokun/LSTNet/blob/master/models/LSTNet.py>

Visualizing LSTM activations

<https://towardsdatascience.com/visualising-lstm-activations-in-keras-b50206da96ff>

Autoregressive Neural Network for Time Series

<https://ai.facebook.com/blog/ar-net-a-simple-autoregressive-neural-network-for-time-series/>

<https://arxiv.org/abs/1911.12436?fbclid=IwAR1tGcAHg9h7TBg0UwIXCbqSypm9rulYY7B_nVF_xQz6MR9zdklsDgRllMY>

* Better for long-term dependencies
* Can perform well on small datasets (where traditional statistical tools usually perform
* Potentially pass the sequence through an AR-Net first to learn dependencies then pass the result into another model which incorporates latent feature representation of text/images

<https://arxiv.org/pdf/1809.04206.pdf>

<https://github.com/huseinzol05/Stock-Prediction-Models>

Hybrid between ARIMA and neural network

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.462.3756&rep=rep1&type=pdf>

<https://stats.stackexchange.com/questions/62693/what-arima-data-to-feed-to-neural-networks-in-hybrid-model-for-forecasting>

Some necessary theory:

Assume you’re trying to predict variable y from some data using an AIRMA model

* It’s reasonable to assume that Y = Lt + Nt where Lt is some linear correlation structure captured by the model and Nt is some non-linear structure
* An ARIMA model is very good at capturing the linear structure of the model thus the residuals = Y – forecast (making the residuals the non-linear part of the model)

\*In practice said to not work well

SIMILAR PROJECT:

<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6121625/>

**Promising way to combine outputs of different models: \*\***[**https://arxiv.org/pdf/1907.07464.pdf**](https://arxiv.org/pdf/1907.07464.pdf)

* Stacking predictions from multiple outputs
* Calculate p-values of suspicious spikes
  + Perform one-tailed test to capture a very large number of infections
* How do we do this?
  + For a fitted distribution p(x), calculate the value of observing ct (last recorded number of monthly cases) as that or higher by doing
  + Govern the sensitivity by the significance level i.e. lower significance level = more acute
  + Can obtain a probability distribution by
    - dividing forecasts at previous steps by their sum
    - calculate key values like mean and finding function of poisson/negative binomial distribution

Review of everything: <https://arxiv.org/pdf/1711.08960.pdf>

Review of probability to understand some of the ^ material <http://cs229.stanford.edu/section/cs229-prob.pdf>

Surveillance Wrapper Package for Python: <https://epysurv.readthedocs.io/en/latest/2_outbreak_detection.html>

Traditional ways of approaching this problem in the past:

* Window based approach e.g. EarsC1
  + Computes predictive distribution based on mean and std of last seven points
  + **Problems:** insensitive to trends and seasonality and recent outbreaks in historical data can contaminate it and reduce the sensitivity of algorithms
* Generalized Linear Models
  + Compute a predictive distribution for current week based on historical data and raise an alarm if an observed value is unlikely under the distribution (p-value) is below a certain significance level
    - Similar to what we are doing above with the hypothesis test
  + Usually Poisson or Negative Binomial distributions are used
  + Examples: Farrington algorithm and its improvement <https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.5595>
* Cusum-based Approach
  + Incorporates data from more than just last week (like above two) thus can pick up on outbreaks that build more slowly over time