# FAIKR/3 Report

## Pollen concentration research with Bayesian Networks and pgmpy library

Serban Cristian Tudosie[1]

[1]serban.tudosie@studio.unibo.it

August 2021

**Abstract– Pollen allergy has become more and more problematic and popular among humans in the recent years. An important role to cope with this problem is of course prevention. Knowing in advance what causes a rise in pollen concentration can be very beneficent in order to avoid irritation. This study aims at explaining what causes the rise of such pollen through the usage of Bayesian Networks.**

## 1 INTRODUCTION

The study is divided basically in two major parts. The first one deals with data gathering and web scraping. The second one takes into consideration the construction of a Bayesian Network, the inference on it (with two methods such as exact inference and approximate inference), and the final results.

## 2 DATASET

Major datasets for such type of study are almost all private, so the need to construct one is unavoidable. In order to gather weather and pollen data, the usage of `selenium` [1] is required since weather and pollen data are available on different sources. In particular the pollen concentration data is available in files that have monthly entries, so they have to be downloaded and merged automatically with weather data. The output dataset is thus saved with the following attributes:

| Date |
|---|
| Pollen Concentration |
| Max Temperature |
| Min Temperature |
| Max Wind |
| Min Wind |
| Wind Direction |
| Precipitations |

Table 1: Dataset Attributes

These are saved as continuous attributes and are only binned when the data is loaded and the bayesian network is constructed. This settings allows to easily change the number of bins, thus allowing a faster a posteriori grid search on the number of bins to choose for each attribute. The wind direction and date are categorical so they are not taken in consideration for the binning. The dataset considers only the city of Florence, Italy for dataset size simplicity and a faster testing.

## 3 NETWORK

A Bayesian Network has to be learnt from the data, this step is called structure learning and can be done with two different approaches: Search Based, and Constraint Based. I've

choose the Search Based approach since it has a great documentation on the pgmpy [2] library and is the most common. The search is paired with a scoring function that has to be optimized. Based on empirical results [3] for big datasets usually K2 score is a good choice, so I also opted out for it since usually weather and pollen data if taken over a large time and space windows can be very large.

By taking the averages for daily temperature and wind speed we get the following DAG as an output:

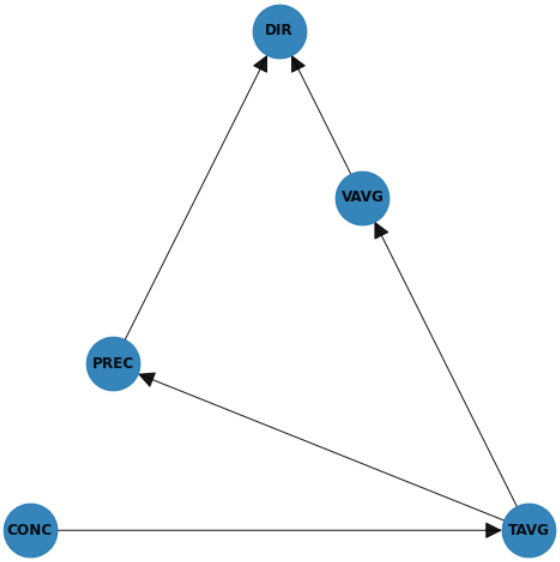


Figure 1: Network Structure

Now lastly on the network side is also required to find the Conditional Probability Distributions (CPDs). This is done through Maximum Likelihood Estimation where we maximize the probability of the data given the model; the likelihood function is thus represented by:

$$L = \prod_{i=1}^{n} \prod_{v \in V} P(x_v^i \mid x_{parent(v)}^i) \quad (1)$$

where: $n$ is the cardinality of the dataset, $V$ is the set of nodes in the model which is represented by a DAG and $P(x_v^i, x_{parent(v)}^i)$ are the parameters that we want to maximize. Usually we optimize the log likelihood since we end up with summations and an easier representation, more in: [4] and [5]

## 4 INFERENCE

With the CPDs found it is possible to perform inference and get to understand problem related information. Exact inference is the one I used the most in finding results since the dataset is not taken over a very big space window (only one location) and the graph has a simple topology (but still does not have the property of singly connectedness). Exact inference can be performed via marginalization but pgmpy provides the variable elimination alternative which is an improvement of the first one. Thus it is possible to perform queries that correspond for example to the following questions:

1. *Is the pollen concentration high when the temperature is warm ?*

2. *Is the pollen concentration high when the wind reaches very high speeds?*

We can get the output for these queries, but in order to understand the trend I have chosen to show the function of the probability of highest concentration of pollen by increasing temperature:

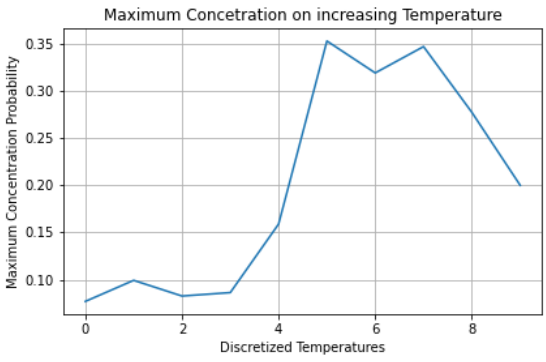


Figure 2: Temperature and Concentration

From this we can understand that the probability of having very high pollen concentrations is maximum when the temperature is mid/high.

Doing the same for increasing wind speeds values we find out that medium speed is the one in favor for high pollen concentration.

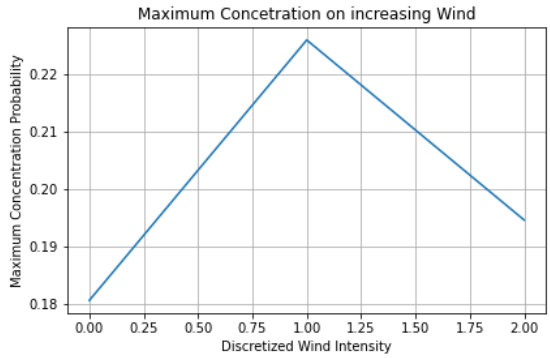


Figure 3: Wind and Concentration

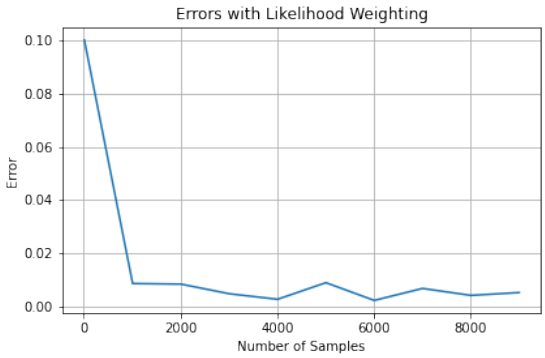


Figure 4: Absolute Errors

The complexity problem with exact inference stands in the worst case scenario, which is exponential in time and space. The solution is to use approximate inference, since even in this type of research if one has to consider many more nodes such as humidity, precipitation for previous days, time of the day and so on, the exact inference methods will have an impact in running time of the software. So I decided to also use approximate inference, in particular likelihood sampling. In order to do so one has to adapt the outputs of `likelihood_weighted_sample()` method from `pgmpy.BayesianModelSampling` does not output directly the probabilities. So a workaround is to implement an adapter function on the output of `likelihood_weighted_sample()` like so:

```
import numpy as np

def find_probs(samples, weights):
    n_vars = len(np.unique(samples))
    probs = np.zeros(n_vars)
    s_list = list(samples)
    for i, s in enumerate(s_list):
        probs[s] += weights[i]
    return probs / np.sum(weights)
```

This allows us to calculate the probabilities for each output of the given query by providing a list of samples and a list of weights that are returned by pgmpy function for likelihood. Using this on an increasing number of generated samples and calculating the error with respect to the variable elimination method gives us an idea of the convergence of approximate methods:

## 5 CONCLUSIONS

As we can see Bayesian Networks are a good tool to exploit in such cases since they allow a background understanding and explainability of the process. In this particular case further studies could include many more attributes that will allow a much deeper understanding of the phenomena. This has a great impact on the life quality of people with pollen allergies, by only avoiding a particular day of very high pollen concentration one has a chance at reducing cross induced inflammation. So by looking at weather forecasts and knowing the patterns to avoid it is possible to reduce in advance the inflammation. All thanks to a Bayesian Model. For further study, if the model rises in complexity, the usage of approximate inference should be considered. I believe that this approach alongside a robust weather predictor will provide a state of the art application for pollen avoidance.

**References**

[1] *Selenium Python Library*, /https://selenium-python.readthedocs.io/.

[2] *PGMPY Hill Ckimb Search*, /https://pgmpy.org/structure_estimator/hill.html.

[3] A. M. Carvalho, *Scoring functions for learning Bayesian networks*, /http://www.lx.it.pt/~asmc/pub/talks/09-TA/ta_pres.pdf.

[4] F. . M. O. Tommi Jaakkola, course materials for 6.867 Machine Learning, *Learning Bayesian Networks*, /https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-867-machine-learning-fall-2006/lecture-notes/lec22.pdf.

[5] H. Yu, "Bayesian Networks: Maximum Likelihood Estimation and Tree Structure Learning," /https://www.cs.helsinki.fi/group/cosco/\Teaching/Probability/2010/lecture12_BN6.4pg.pdf.