Forest Fires in Portugal – Bayesian Networks

Uzi Smadja - 313673782 - uzi.smadja@gmail.com

Bernie Howard - 203039417- bernie.howard@gmail.com

1 Introduction

The goal of this project is to analyze forest fire data from the Montesinho natural park from the Trás-os-Montes northeast region of Portugal (Figure 2). The park contains a high flora and fauna (plants & animals) diversity. In the simplest case, an expert defines a Bayesian network, which is then used to perform inference.

In this case the network structure and the parameters of the local distributions are learned from the data itself. Following the learning of the structure and distributions, we intend to evaluate and compare between two of the inference algorithms learned in class. We will explore the difference between exact inference and approximate inference. The query we will explore is the relation of the size of fires to the humidity and temperature.

2 Fire Weather Index

The forest Fire Weather Index (FWI) is the Canadian system for rating fire danger. It consists of six components as illustrated in Figure 1: Fine Fuel Moisture Code (FFMC), Duff Moisture Code (DMC), Drought Code (DC), Initial Spread Index (ISI), Buildup Index (BUI), FWI

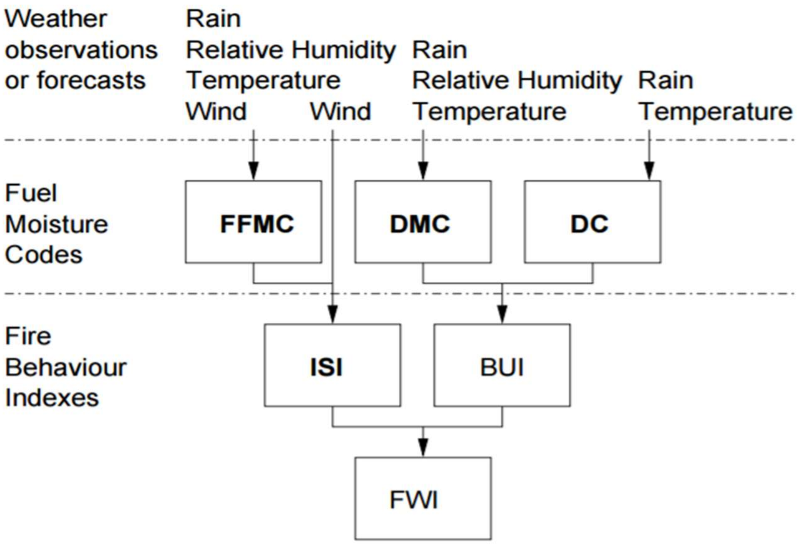


Figure 1 Fire Weather Index structure

The first three are related to fuel codes: FFMC denotes the moisture content surface litter and influences ignition and fire spread, while the DMC and DC represent the moisture content of shallow and deep organic layers, which affect fire intensity. The ISI is a score that correlates with fire velocity spread, while BUI represents the amount of available fuel. The FWI index is an indicator of fire intensity and it combines the two previous components. Although different scales are used for each of the FWI elements, high values suggest more severe burning conditions. Also, the fuel moisture codes require a memory (time lag) of past weather conditions: 16 hours for FFMC, 12 days for DMC and 52 days for DC.

3 Dataset

3.1 General description

The data used in the experiment was collected from January 2000 to December 2003 and originated from two sources. The first database was collected by the inspector who is in charge of monitoring the Montesinho fire occurrences. Several features were recorded on a daily basis, every time a forest fire occurred. They include, the time, date, spatial location within a 9×9 grid (x and y axis of Figure 2), the type of vegetation involved, the six components of the FWI system (see Figure 1) and the total burned area. The second database was collected by the Bragança Polytechnic Institute and contains several weather observations (e.g. wind speed) that were recorded over a 30 minute period by a meteorological station located in the center of the Montesinho park.

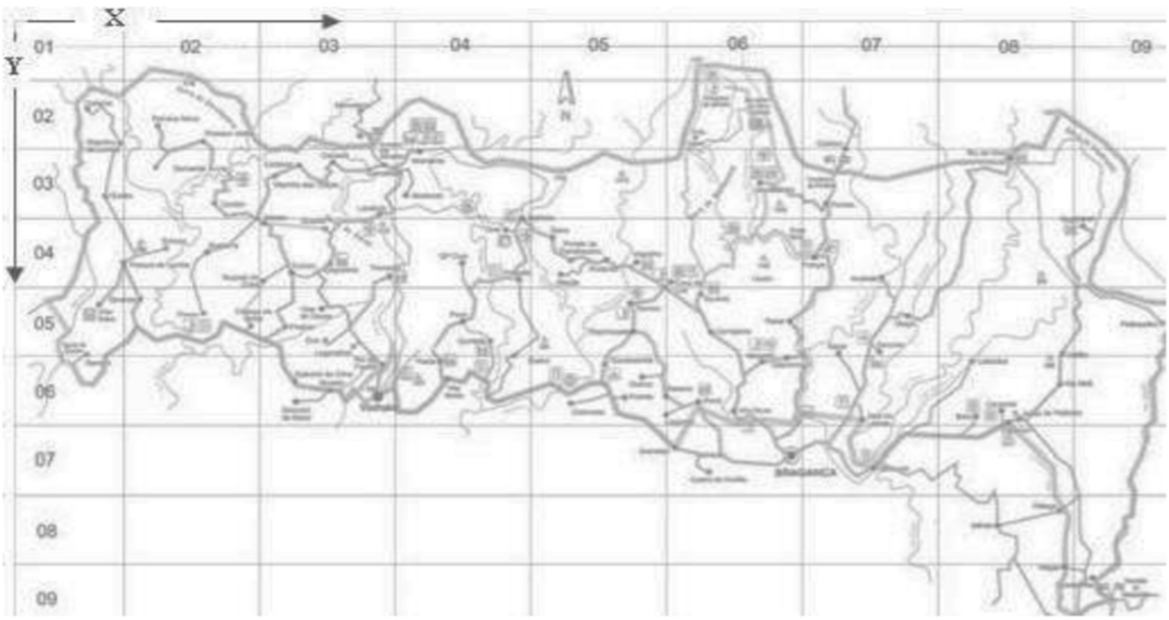


Figure 2 The map of the Montesinho natural park

3.2 Raw data description

As mentioned above, the dataset contains 517 entries, each entry consists of the following features:

- Longitude and latitude coordinates as shown in Figure 2
- Month and day of the week of the forest fire occurred
- Weather condition: temperature, humidity(RH), wind and occurrence of rain during the day
- FFMC, DMC, DC, ISI as defined earlier
- Area of the forest fire

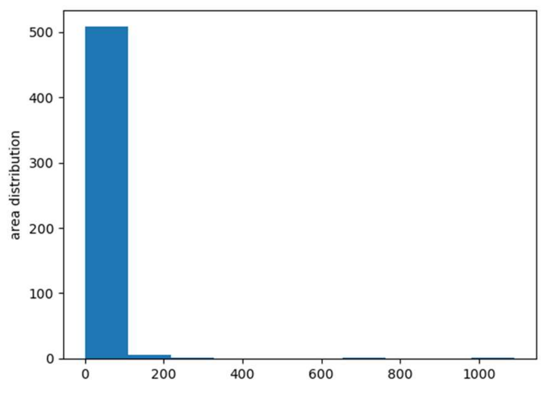
In addition, we created another small database, each row within it contains an "experts" information about a certain fire event that could happen "by logic" - For example, a large fire in the middle of the forest, during a hot dry summer day with no rain.

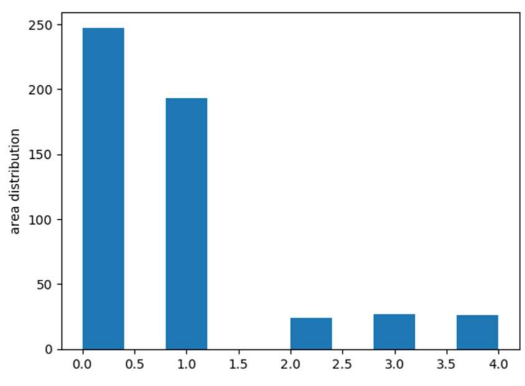
3.3 Refined data description

In order to adapt our continuous features to the discrete approaches learned in class, the features were converted to discrete values, as detailed below.

- X, Y, day, month: already discrete and each value represents a different spot in time or space. Therefore, these features were kept as is.
- Temperature, wind, RH, FFMC, DC – were given one of 5 values (0,1,2,3,4). The calculation was done by splitting the difference between the highest and lowest values of each feature into 5 equally sized ranges.
- DMC, ISI, rain, area – The previous approach could not be applied to these features as they are poorly distributed, causing some values to be entirely absent from the dataset. Consequently, we chose here to adapt the size of each range of values was adapted using the mean and standard deviation of each feature. This allowed us to better represent all the possible different values.

One of the features that needed careful attention was 'area', which represents the size of the burned area. Even after careful quantization, the probabilities were not distributed in a Gaussian manner. The following graphs show this feature value first in its raw form (left side) and its new form after quantization (right side).





4 Bayesian Network Structure Learning

To simulate a Bayesian graph, we created a graph class containing the node classes as mentioned above. Each node contains a set of parents. As seen Figure 1, some parent child node dependencies are already known, but not all of them (i.e. X and Y coordinates aren't represented in Figure 1).

Our goal here us to compute the maximum likelihood estimation (MLE) of the model given the data above, meaning $p(m|d)$. As learned, $p(m|d) \propto p(m)p(d|m)$, while $p(d|m) = \int p(d|\Theta)p(\Theta|m)d\Theta$. A grave assumption (prior) lies here (which we will attend to in the result section below), which is that the model structures are distributed uniformly, so $argmax_{m\in M}\, p(d|m)$ was the goal searched.

As shown in lecture no. 10, and while assuming global and local parameter independence,

$$p(d|m) = \prod_{i=1}^{n} \prod_{j=1}^{q_j} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

While denoting N_{ijk} as the number of cases when $X_i = x_i^k$ and $Pa_i = pa_i^j$

r_i - number of states of X_i

q_i - number of instances of the parents of X_i

$\alpha_{ij} = \sum_{k=1}^{r_i} \alpha_{ijk}$, $N_{ij} = \sum_{k=1}^{r_i} N_{ijk}$, and $\Gamma(\alpha) = (\alpha - 1)!$

We made a small change, and instead we did a log-likelihood estimation, denoted as

$$p(d|m) = \sum_{i=1}^{n} \sum_{j=1}^{q_j} \log_{10}\left(\frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})}\right) + \sum_{k=1}^{r_i} \log_{10}\left(\frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}\right)$$

Finding the optimal graph model is a NP-Hard problem, so we addressed the problem in an easier way, yet logical way. We created an initial graph as a base for our calculations and assumptions (shown in fig. 3). We implemented two simple graph-changing algorithms, both changing the graph by adding edges, removing edges and switching directions of directed edges.

The first mechanism consists of manually entering add/remove/switch commands into a configuration file that is fed as input to our graph modification program. The second mechanism consists of randomly manipulating the edges of the graph. Both methods can of course damage the DAG characteristic of the graph, so a DFS is being executed at every step.

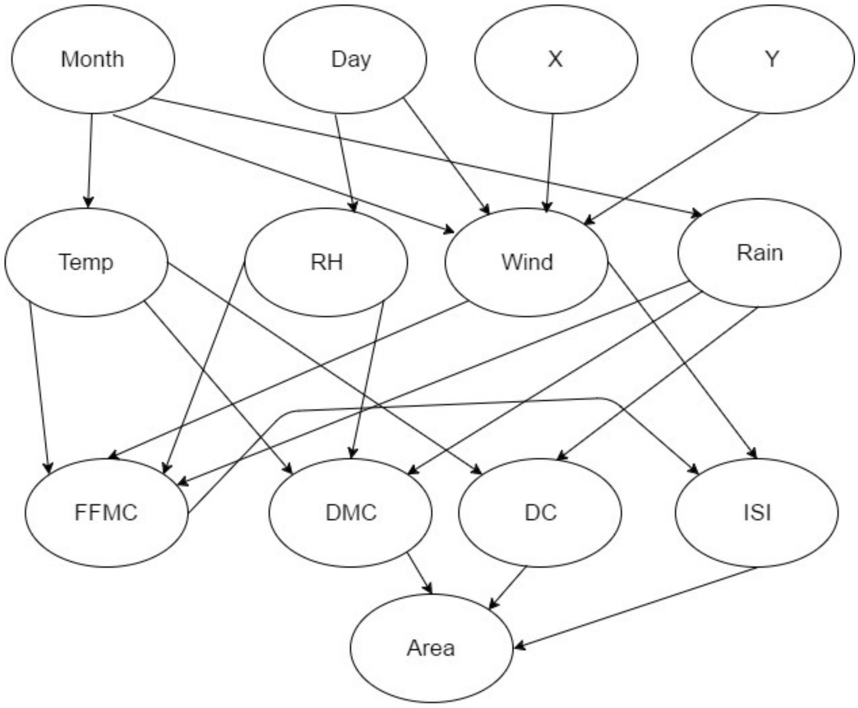


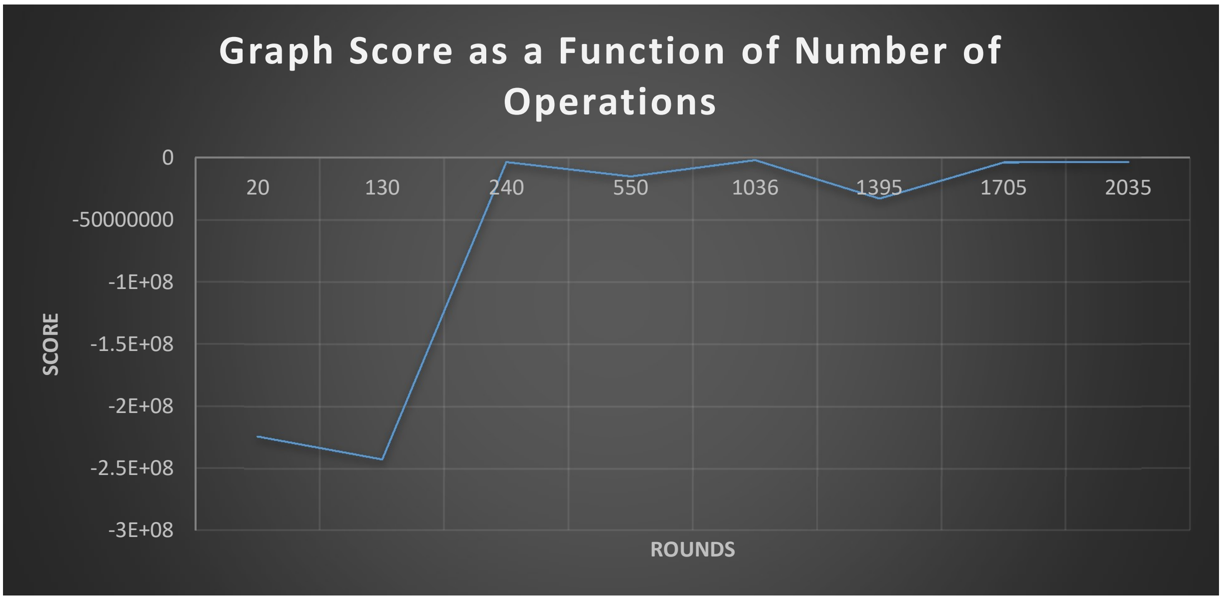
Fig. 3 Initial graph

4.1 Structure learning results

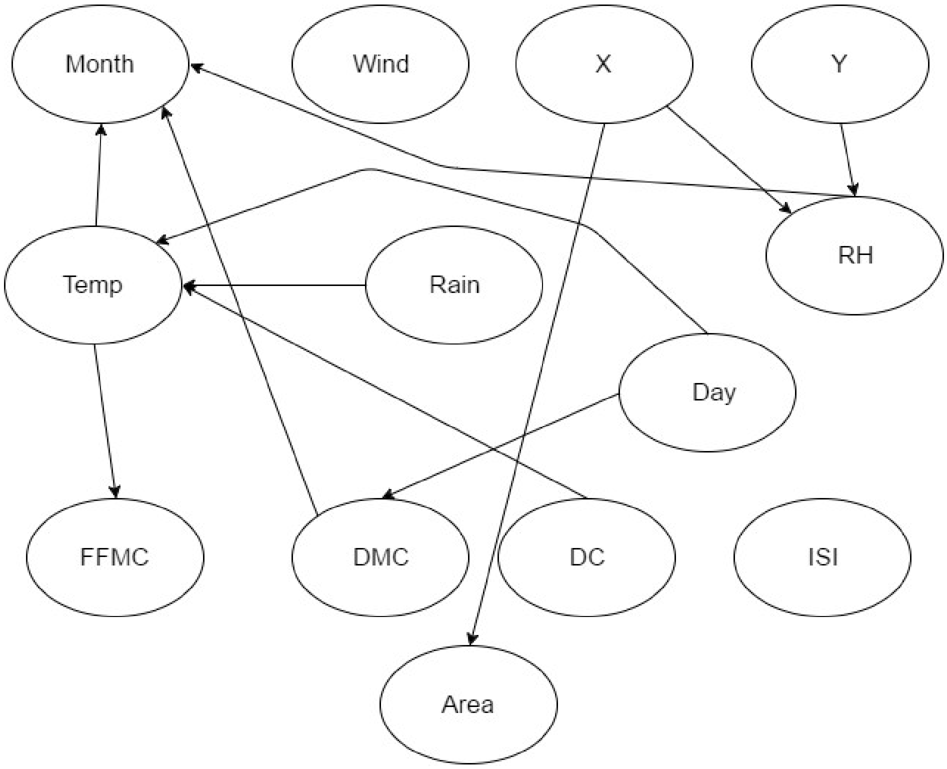
We detail the result of our second graph manipulation mechanism, which implements random modifications. The auto generator function gets two arguments, I and R (iterations and rounds).

For each round j, $j \in [R]$, we conduct a series of random changes in the model: $2 \cdot I$ add operations, I removals and I edges swap. After each operation, the maximal-scored-model is saved and passed for further changes.

We tested the algorithm for different increasing I and R values, and got the following graph representing score as a function of number of operations $(r \cdot (3i + 1))$. Remembering that due to proximity to zero, we cannot represent the real score, each score is in fact 10^{score}.



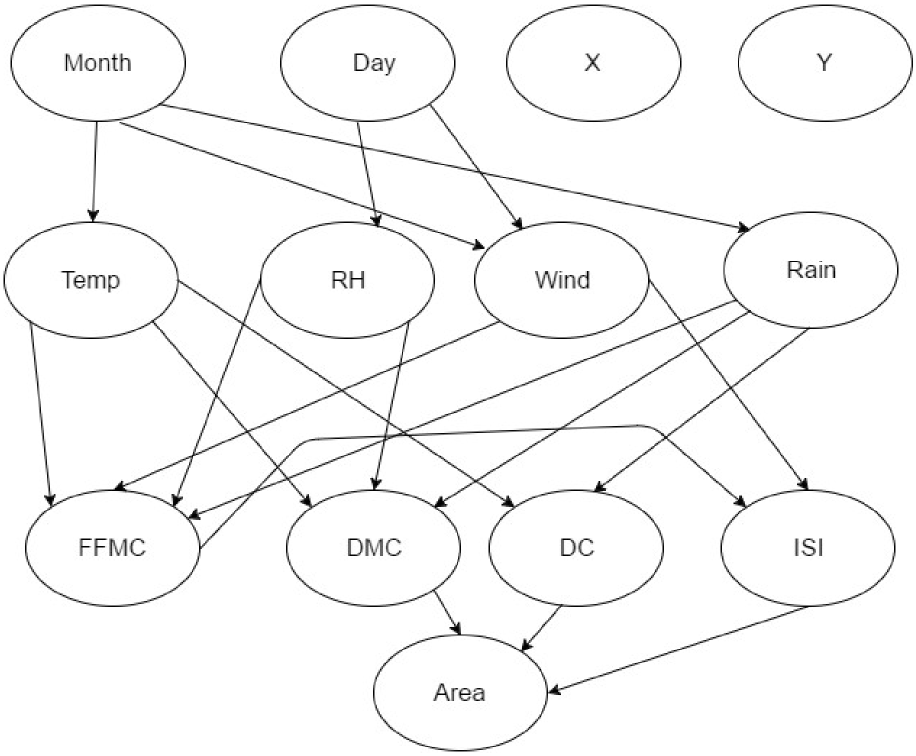
The score in grows with the increasing number of iterations, but we notice that the generated graph doesn't necessarily make sense, and strange dependencies may occur. For instance, the month of the fire is shown as being dependent on the humidity rather than the opposite. Such strange dependencies can be explained by the uniformity of the graph probability assumed earlier. In fact, the probablity of a graph with a dependency like the one presented above is close to zero.



Considering these issues, we approached the problem from a different prespective, taking into consideration the initial graph. We manually created subtle changes in the graph and checked the score at each stage.This approach is far more accurate in our opinion, and we did see that swapping edges in a non logical way did harm the model score.

4.2 Final Graph

The final graph created with manual changes on which the inference algorithms lean on, is the same graph as seen in Figure 3, with some minor changes. We discovered that the best model disregards the impact of 'X' and 'Y' nodes on 'Wind' node, meaning that the model without these edges is more exact – and there is no impact of a location on wind strength at this location.



5 Inference – exact vs. approximate

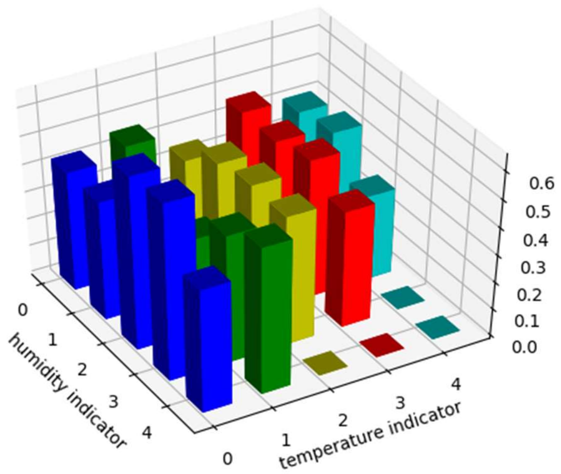
After learning the graph structure, we can now define a proper query and use two inference algorithms to answer it. In this work, we want to verify whether there is a relation between the burned area and the humidity (RH) and temperature levels. In other words, we wish to know: $P(area|RH = h, temp = t)$. In each of the algorithms, we set the value of the two evidences (humidity and temperature) and examine results over all possible values.

5.1 Exact inference – variable elimination

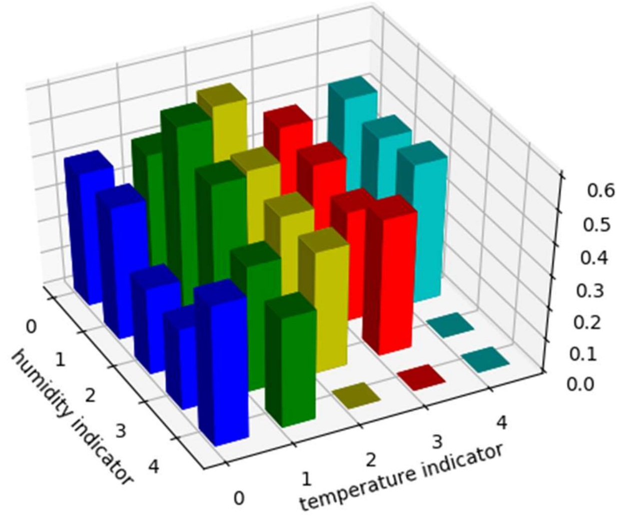
Since finding an ideal order for the variable elimination is an NP-hard problem, we settled on an order chosen manually, with respect to the graph. The order of elimination is: day, month, rain, wind, FFMC, DMC, DC, ISI.

The following graphs show the probability of a fire occurring for each of the five fire levels, based on all the combinations of humidity and temperature.

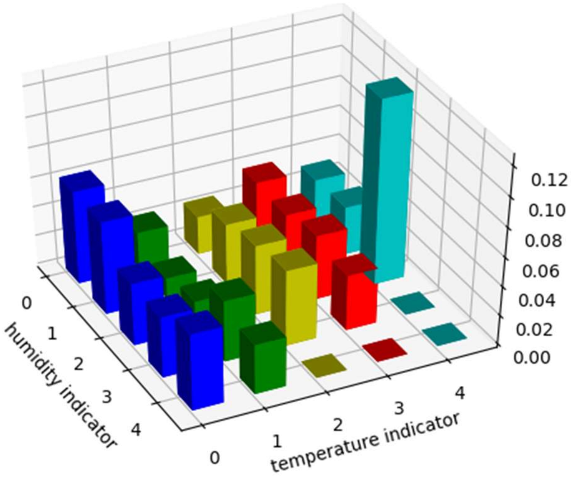
- $P(area = 0|RH = h, temp = \ \ )$



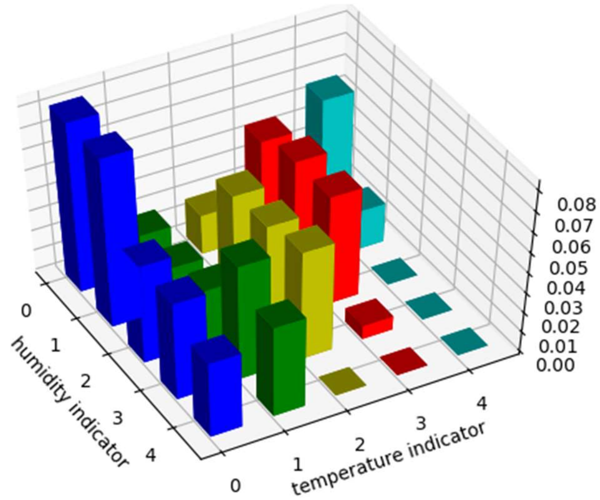
- $P(area = 1|RH = h, temp \ \ \ \ )$



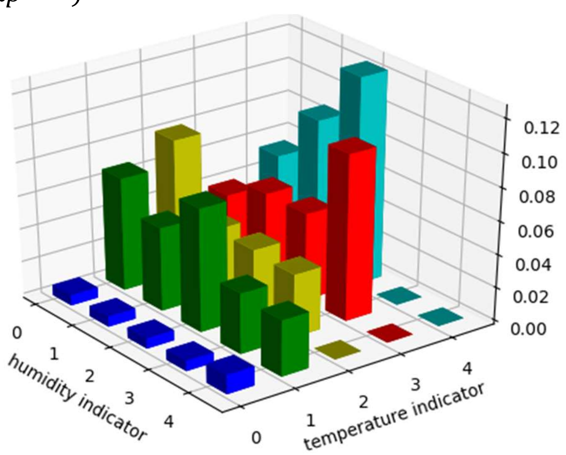
- $P(area = 2|RH = h, temp \quad)$



- $P(area \quad 3|RH = h, temp \quad)$



- $P(area = 4|RH = h, temp = \ )$



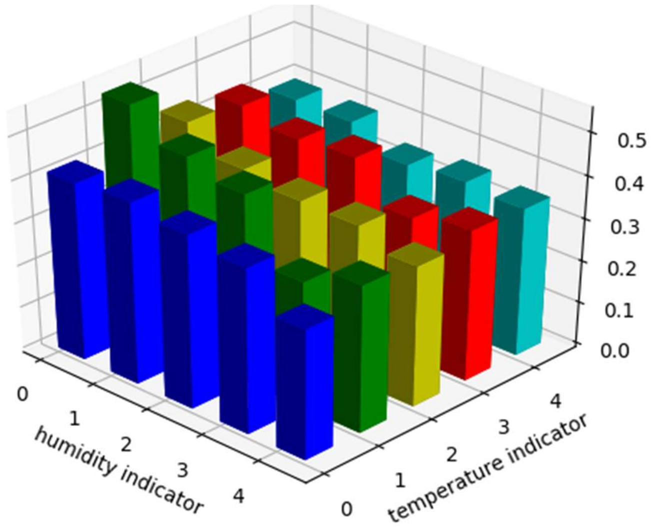
5.1.1 Discussion of results

There are some combinations of 'RH' and 'temp' that do not exist in any of the samples of the dataset. As a result, some probabilities amount to 0. As seen in the five graphs shown above, the combinations (3,4), (4,2), (4,3), (4,4) of (RH, temp) are all 0.
Looking at the results, we can deduct that large fires occur mostly when the temperature is high and when the humidity is low, which confirms basic intuition. However, small and medium fires occur mostly when the humidity is low and have less correlation with the temperature.

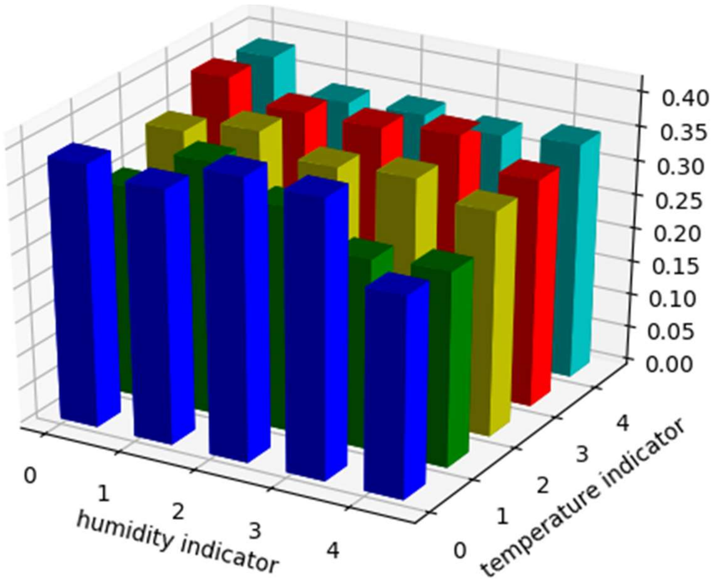
5.2 Approximate inference – Gibbs sampling

We used the Gibbs sampling algorithm as taught in class in order to obtain a sequence of observations approximated from a specified multivariate probability distribution.

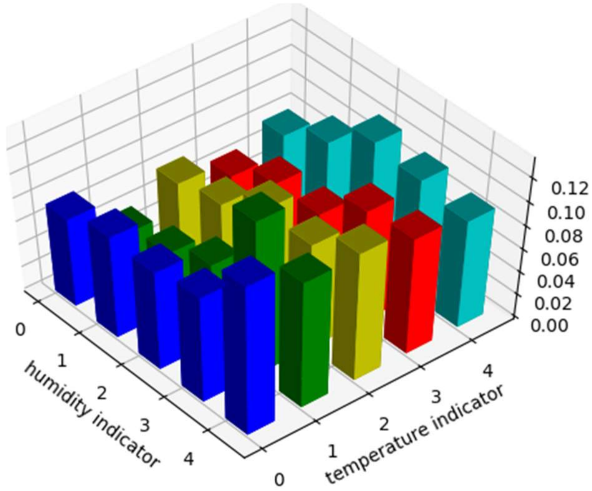
- $P(area = 0|RH = h, temp$



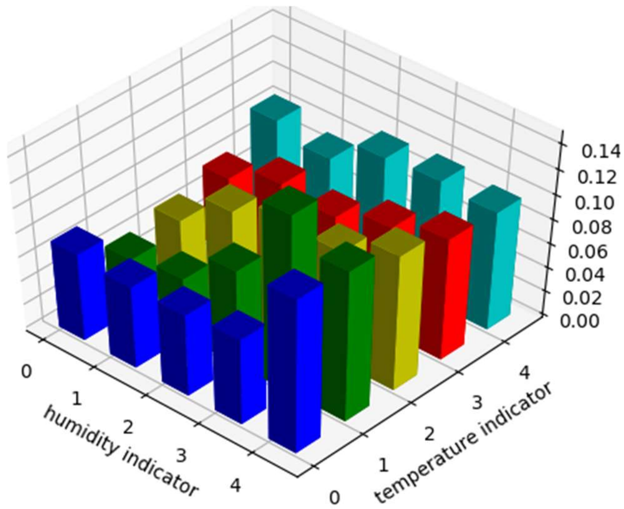
- $P(area = 1|RH = h, temp = t)$



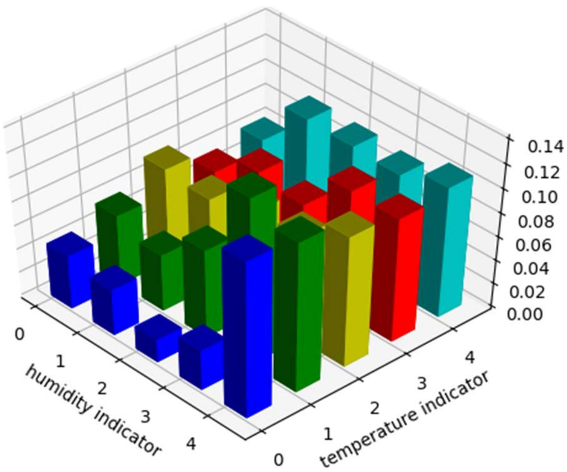
- $P(are \quad 2|RH = \quad, temp = \quad)$



- $P(area = 3|RH = h, temp \quad)$



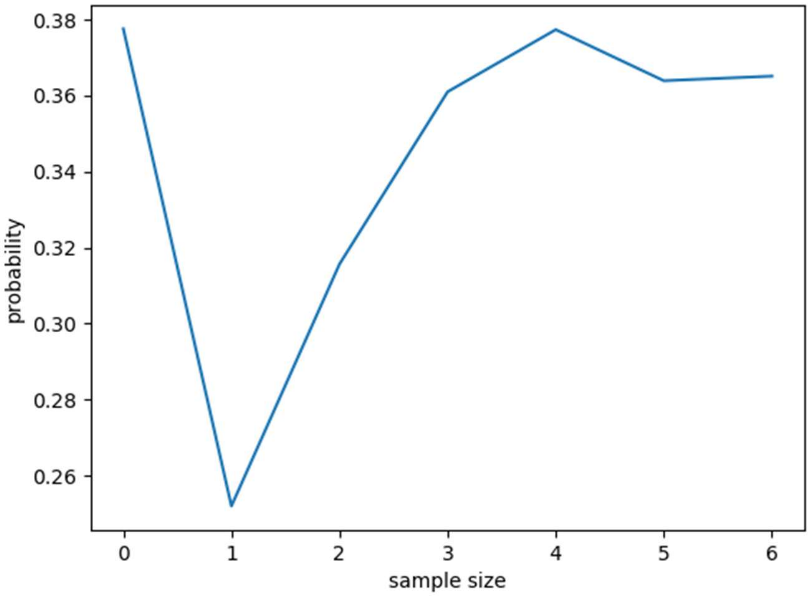
- $P(area = 4|RH = h, tem \quad)$



5.2.1 Discussion of results

From the results of the Gibbs sampling we can deduct that large fires occur mostly when the temperature is high. However, small and medium fires occur mostly when the humidity is low and the temperature is high.

The number of iterations chosen for the Gibbs sampling algorithm is 10000. The following convergence graph demonstrates the convergence for $P(ar \quad 1|RH = 3, temp = 3)$ as an example.



Sample size values represent indices of [10, 50, 100, 500, 1000, 5000, 10000]

6 Conclusions and discussion

The first step in the project was a most probable model containing the edges mentioned above. This problem is NP-Hard (Chickering, 1995), and looping in the loop described in section 4.1 for a significant number of rounds resulted in a graph with almost no edges. As result we tried the different approach mentioned above, a greedy and manual one. As described, with the assumption of uniformity of models, the greedy (random) method produced the best results, but they didn't always made sense. The manual edge removals did help us to discover that some edges did indeed had no affect one another, without damaging the wholeness of the graph model. As result we decided do inference on the new exact model.

Looking at the two inference approaches we can clearly see the advantages and disadvantages in both. The exact inference approach (variable elimination), produces the exact probabilities of the given query. However, as mentioned in section 5.1.1, if the data is not large enough, the algorithm cannot handle these gaps and fails to converge on the result. Moreover, the algorithm requires some basic understanding of Bayesian Networks to pick an elimination order that won't take too much time to compute – as the task of finding an optimal order is an NP-hard problem. On the other hand, the Gibbs sampling method, which is easy to implement, produces results for all possible combinations of the query and seamlessly deals with gaps and missing samples. We can see that the results of the Gibbs sampling are close to the variable elimination but not exactly the same. As shown in section 3.3, there are two factors which occur, to a certain degree, in our dataset:

- Islands of high-probability states, with no paths between them
- Single island of high-probability state. This can happen even when all states have a nonzero probability.

The query we tried to answer is $P(area|RH = h, temp = t)$, which gives us information about the size of the burned area in a fire, based only on the humidity and temperature. After comparing the two approaches we can conclude that although the two methods of inference are not identical, there is a strong correlation between the humidity, temperature and the size of burned area. These results sit well with our general knowledge and prediction of the results, as high humidity means a lower chance of fire and higher temperature means higher chance of fire.

7 References

[1] [D. Heckerman, D. Geiger, D.M. Chickering, 1995] Learning Bayesian Networks: The Combination of Knowledge and Statistical Data

[2] [J. Pearl, 1988] Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference

[3] https://en.wikipedia.org/wiki/Gibbs_sampling

[4] http://www.lx.it.pt/~asmc/pub/talks/09-TA/ta_pres.pdf