Predicting Plankton Production from Satellite Data

Technical Paper CS400
Department of Computer Science
University of Cape Town

Robert Curtis
crtrob001@cs.uct.ac.za

Richard Fenn
fnnric001@cs.uct.ac.za

Damon Oberholster
obrdam001@cs.uct.ac.za

ABSTRACT

Estimates of plankton primary production are essential to understanding the functioning of the marine ecosystem and the possible impacts of climate change of the marine food web. Sub-surface chlorophyll is an excellent predictor plankton production, but collection of sub-surface chlorophyll data is slow. Surface data, however, can quickly be obtained via satellite. A method is therefore needed to predict sub-surface data using only surface information. Previous research in this field involved the use of self-organising maps (SOMS) to predict plankton-profiles. These SOMS are, however, hard to interpret and not very precise. The system proposed used Bayesian networks to predict sub-surface chlorophyll based on satellite data and other environmental factors. Bayesian Networks are comprised of two parts: a learning engine and inference engine. The learning engine finds patterns in historical data and the inference engine takes these patterns as input and predicts likely trends. An Investigation was undertaken to determine the use of topic maps for representing Bayesian network structure and beliefs. These topic maps needed to be visualised in an intuitive manner. A hyperbolic tree visualization was investigated as an alternative to static visualizations.

The accuracy of predictions was limited by the use of Gaussian approximations to define the predicted profile, but the use of EM to create new profiles should give far better results in future. It was found that the topic maps provided a useful mechanism for passing the Bayesian network information between the inference engine and the interface. The hyperbolic visualisation of Bayesian networks was at least as easy to use as static representations.

1. INTRODUCTION

This project aims to find a new method to predict sub-surface chlorophyll information, based on Bayesian networks. Because chlorophyll is present in plankton, it provides an excellent indication of the amount of plankton present. The predictions will be based on a number of environmental factors that are known at a point in the ocean for a given time and space (depth of ocean floor, season, and region) and also data obtained from satellite (surface chlorophyll and temperature).

Collection of sub-surface data can only be achieved through the use of shipboard measurements. This process is time-consuming and provides poor coverage. There is, however, an archive of 10 years worth of sub-surface ship readings. By combining the environmental and satellite data with the archive of ships' sub-surface readings, a single training set can be created in order to train a Bayesian network. Once the training has taken place, the Bayesian network can be supplied with environmental data, as well as the satellite data for a given day, and the sub-surface chlorophyll information predicted. This can be done on a per-pixel basis, providing large-scale sub-surface information in real-time.

The sub-surface chlorophyll information is predicted in the form of a *chlorophyll profile*, which is simply a continuous function of chlorophyll with respect to depth. This profile gives an indication of what depth the sub-surface peak occurs. While this peak often occurs near the surface, currents (called upwellings) can cause peaks to occur below the surface over time. This has been well researched in areas such as the Agulhas bank and off the west coast of South Africa. [1] When an upwelling of cold water occurs, it often brings lots of nutrients to the surface. The surface layer is well lit, and therefore a plankton *bloom* (a very high concentration) develops at the surface. The density of the plankton at the surface blocks light to lower layers, preventing sub-surface plankton growth. As the surface bloom diminishes (due to nutrient depletion), more light is let through to the lower layers, and so a sub-surface peak occurs.

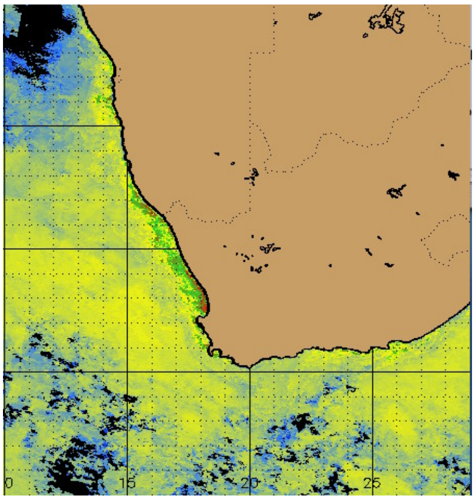


Figure 1: Integrated chlorophyll

Once a chlorophyll profile has been predicted, it can be used to estimate *plankton primary production* at a given point in the ocean. This is done by integrating the profile, and combining it with light field satellite images. (Result shown in **Figure 1**)

Estimates of primary production are essential to understanding the functioning of the marine ecosystem and the possible impacts of climate change of the marine food web. [1]

2. BACKGROUND

2.1. Related Work

Other artificial intelligence techniques have been used to classify both satellite data, and sub-surface ship readings. [2] This involved the use of self-organizing maps (SOM), a type of artificial neural network, to extract inter-annual and seasonal variations observed in both thermal and chlorophyll satellite images. The SOMs were also used to classify the archive of ships' sub-surface readings into 15 categories

Another technique involved fitting each sub-surface ship reading to four-parameter shifted Gaussian curve. [1] A generalised linear model was used to generate correct parameters for the Gaussian curve, hence predicting the chlorophyll profile. The linear model made use of environmental and satellite images much like those used in this project.

2.2 Bayesian Networks

A Bayesian network is defined by Murphy [3] as a *directed* graphical model, where a graphical model is defined by Jordan [4] as a "Marriage between probability theory and graph theory", providing a natural tool for dealing with uncertainty and complexity.

Using this representation, Bayesian networks provide an intuitive interface through which humans can easily model interacting sets of variables.

The following example outlines the basic principles behind Bayesian representation and probabilistic inference.

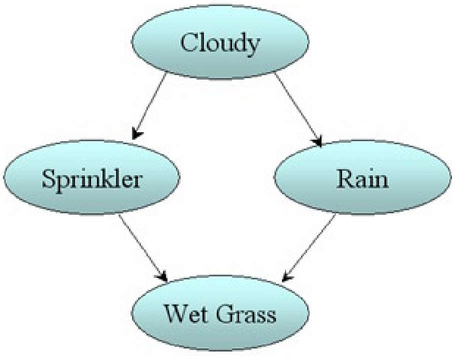
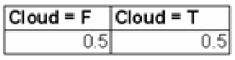


Figure 2: Basic Bayesian network

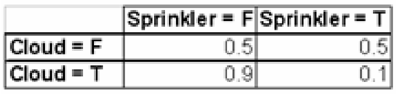
Each node has the following associated conditional probability tables (CPTs):

Cloudy



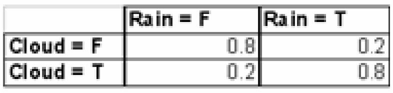
Cloud = F	Cloud = T
0.5	0.5

Sprinkler



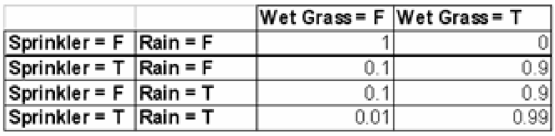
	Sprinkler = F	Sprinkler = T
Cloud = F	0.5	0.5
Cloud = T	0.9	0.1

Rain



	Rain = F	Rain = T
Cloud = F	0.8	0.2
Cloud = T	0.2	0.8

Wet Grass



		Wet Grass = F	Wet Grass = T
Sprinkler = F	Rain = F	1	0
Sprinkler = T	Rain = F	0.1	0.9
Sprinkler = F	Rain = T	0.1	0.9
Sprinkler = T	Rain = T	0.01	0.99

The joint probability of each node can be calculated using the chain rule, combined with the conditional independence relationships in the Bayesian network. This formula is:

$$P(C, S, R, W) = P(C) * P(S \mid C) * P(R \mid C, S) * P(W \mid C, S, R)$$

Now assume evidence is obtained: the sprinkler is known to be on. The sprinkler node can now be instantiated to the "true" state. Conditional queries, such as the probability of the sprinkler (S=t) is on, given that the grass is wet (WG = t), can be calculated by using the following formula:

$$P(S=t\,|\,WG=t)=\frac{P(S=t,WG=t)}{P(WG=t)}$$

$$=\frac{\sum_{c,r}P(C=c,S=t,R=r,W=t)}{P(WG=t)}$$

$$=\frac{(0.0396+0.0495+0.009+0.18)}{(0.0396+0.0495+...+0.018+0+0)}$$

$$=\frac{0.2791}{0.6471}=0.4298$$

There is therefore a 42.98% probability of the sprinkler being on, given that the grass is wet.

The CPTs used by the Bayesian network are created by learning algorithms. There are a number of different algorithms that can be used, depending on whether there are hidden variables. These are discussed in detail later.

2.3 Topic Maps

Topic maps are an ISO standard for describing knowledge structures, and link these structures to the underlying resources that they represent [7]. They provide a means of connecting the realms of knowledge representation and information management [8].

2.4 Hyperbolic Trees

Hyperbolic trees are a distortion based (focus and context) technique for visualising large graph structures. They utilise hyperbolic geometry to attain focus and context. This is achieved by laying the graph onto a hyperbolic plane and then mapping that plane to a circular display region. Hyperbolic geometry is a non-Euclidian geometry. In non-Euclidian space, parallel lines diverge. This is an important property because it means that the circumference of a circle grows exponentially with increasing size of radius. This results in exponentially more space as one moves away from the centre of the circle. Therefore, nodes at the centre of the circle appear larger (and are in focus) and nodes at the edges appear smaller. This allows the viewer to always see at least 3 child nodes ahead of the current centred node. [10], [9].

3. SYSTEM OVERVIEW

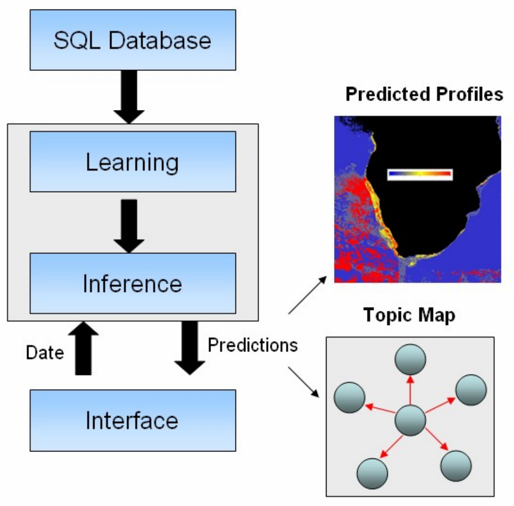


Figure 3: Overall system design

3.1. SQL Database

The database contains final training set for the learning engine. A large part of the project involved the creation of this data set, as many years of sub-surface data needed to be combined with satellite chlorophyll and temperature data.

3.2 Learning Engine

The learning engine runs through the database and creates the conditional probability tables required by the inference engine. Two Learning algorithms were implemented. These were the Maximum Likelihood Estimation algorithm and the Expectation Maximisation algorithm. Each of these will be explained in the section 4.

3.2 Inference Engine

The inference engine loads the CPTs generated by the learning engine and performs inference for the dates provided by the interface. This results in an image of profiles being generated, and a topic map representing the state of the network at a particular point.

3.3 Bayesian Structure

Before any learning or inference can take place, the network structure needs to be modelled. This determines how the various variables are incorporated into the Bayesian network. The following structure was used for the MLE algorithm:

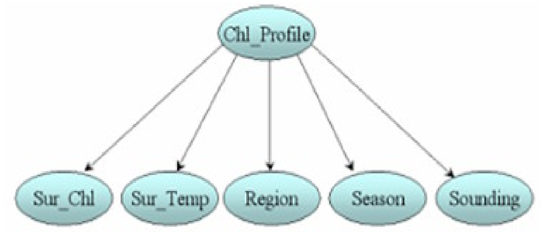


Figure 4: Structure used for MLE

When using MLE, the predicted profile is 1 of 10 clustered 4-parameter, shifted Gaussian curves. (Mentioned in the related work section) These clusters were created using the EM algorithm.

The EM algorithm needs to cluster using the sub-surface information, and therefore needs this additional information. The associated structure is as follows:

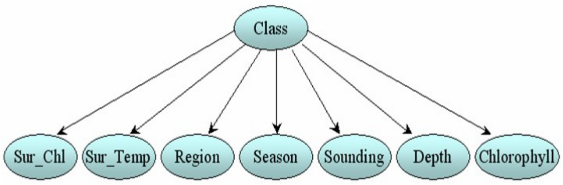


Figure 5: Structure for EM

Each of the variables has the following meaning:

- Sur_Chl – The surface chlorophyll obtained from the satellite image
- Sur_Temp – The surface temperature obtained from the satellite image
- Region – The ocean area was divided into four regions, each with slightly different chlorophyll behaviour patterns
- Season – May→October: Winter, November→Februrary: Summer
- Sounding – The depth of the ocean floor
- Depth – The depth of the reading
- Chlorophyll – The amount of chlorophyll at a given depth

3.4 Interface

The interface loads the images and generated by the inference engine, and also displays the generated topic map as a hyperbolic tree.

4. LEARNING ENGINE

4.1 MAXIMUM LIKELIHOOD ESTIMATE

The Maximum Likelihood Estimation (MLE) algorithm can be used when the structure of the Bayesian Network is already known and the dataset is complete.

The following description of the MLE is adapted from the example in [5].

Assume we buy a bag of chocolate. The chocolates in the bag are mixture of either Cadbury or Nestle. The Bayesian network for this structure is shown in Figure 6.

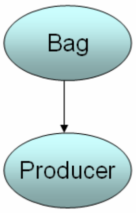


Figure 6: Basic network

The proportions of each producer's chocolate in the bag are completely unknown. If we wanted to calculate the proportion of Nestle chocolates in the bag, the MLE algorithm would be used.

Let the proportion of Nestle be denoted by *θ*. Now, suppose we select N chocolates from the bag where *n* is the number of Nestle chocolates selected and *c* = N – *n* is the number of Cadbury chocolates selected. Under the assumption that all observations are independent and identically distributed, the likelihood of the dataset is given by:

$$\mathrm{P}(d \mid h_\theta) = \prod_{j=1}^{N} P(d_j \mid h_\theta) = \theta^n (1-\theta)^c$$

Where h_θ is a non-observed hypothesis about the data and *d* represents the dataset.

We can simplify it by taking the logarithm of the equation. This breaks the equation up into the sum of three terms, where each of the terms contains a single parameter. The equation then looks as follows:

$$L(d \mid h_\theta) = \log P(d \mid h_\theta) = n \log \theta + c \log(1-\theta)]$$

In order to obtain θ, we take the derivative with respect to the parameter of interest (in this case the parameter is *n)* and set it to zero:

$$\frac{\partial L}{\partial \theta} = \frac{n}{\theta} - \frac{c}{1-\theta} = 0 \Rightarrow \theta = \frac{n}{n+c} = \frac{n}{N}$$

It can now be seen that the solution for θ is simply the number of Nestle chocolates taken from the Bag, divided by the total number of chocolates taken from the bag.

4.2 Expectation Maximisation Algorithm

The EM algorithm is an iterative algorithm used when the Network structure is known, but the data is only partially observable (i.e. not all nodes are evidence nodes). This happens in many real world problems that have hidden variables (nodes that are not observable in the data).

The EM algorithm is also best explained with an example. Consider the example used for the MLE algorithm, but this time assume that 2 bags (Bag1 and Bag2) of chocolates have been bought and mixed together so that we no longer know which bag each chocolate comes from.

It can be seen that the Bag node is now a hidden variable because once the chocolates have been mixed together; we no longer know which bag each chocolate came from. Assume that we want to find the probability, θ, of a selected chocolate coming from bag1.

In the fully observable case we would be able to estimate *θ* directly from the observed counts of the number of chocolates from bag1. Because the Bag node is hidden, we need to calculate the expected counts instead.

The expected count of the number of chocolates that come from Bag 1 can be calculated as *the sum, over all the chocolates, of the probability that the chocolate came from bag 1* [5]. The notation for the expected count is: *N*(Bag = 1).

Therefore:

$$\theta^{(1)} = \frac{N(Bag=1)}{N} = \sum_{j=1} \frac{P(Bag=1 \mid \text{Producer}_j)}{N}$$

At first, we do not know the probability of a selected chocolate coming from each bag. We therefore start by randomly assigning probabilities.

For each iteration of the algorithm, we recalculate the probabilities using an inference algorithm. This example uses Bayes rule :

$$\theta^{(1)} = \frac{1}{N}\sum_{j=1}^{N} \frac{P(producer_j \mid Bag=1)P(Bag=1)}{\sum_i P(producer_j \mid Bag=i)P(Bag=i\ )}$$

For each iteration, the probabilities will tend closer and closer to the best 'fit' for the data.

5. INFERENCE ENGINE

The inference engine implemented was Judea Pearl's message passing algorithm, also known as Pearl's belief propagation algorithm. While this is an efficient algorithm for probabilistic inference (polynomial time), it assumes that the network is singly-connected (a polytree). However, accurate approximate results have been obtained in multiply-connected networks [6].

The algorithm is based on the notion of propagating evidence[1] through the network. These messages travel both up and down (in order to accommodate both top-down and bottom-up inference) and are labeled λ and π messages respectively. The following figure gives some details with regards to notation.

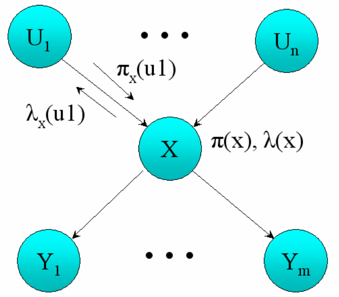


Figure 7: Notation used for belief propagation

Using the conditional independence relationships of a polytree, the following recursive expressions for computing messages can be derived:

(1) The equation to calculate the *belief* array of a certain node (used after calculating local π and λ messages after a round of message passing):

$$P(x \mid e) = \alpha\, \pi(\mathrm{x})\, \lambda\, (\mathrm{x})$$

(2) The equation used to calculate the local π value of a node, after receiving all π messages from a round of message passing:

$$\pi(x) = \sum_{u1..un} P(x \mid u1..un) \prod_{i=1}^{n} \pi_x(u_i)$$

(3) As with equation **(2)**, only this is for λ value:

$$\lambda(x) = \prod_{j=1}^{m} \lambda_{Y\,j}(x)$$

(4) The equation used to calculate the π value a node sends to its child:

$$\pi_{Y\,j}(x) = \alpha \prod_{k \neq j} \pi(x)\lambda_{Y\,k}(x)$$
$$= \alpha \frac{P(x \mid e)}{\lambda_{Y\,j}(x)}$$

(5) The equation used to calculate the λ value a child node sends to its parents:

[1] Knowing that a certain variable (or a number of variables) is in a certain state. It is often said a node is *instantiated* with evidence.

$$\lambda_{Y_j}(x) = \sum_{yj}[\lambda(y_j)\sum_{v1}^{vp} P(y_j \mid x, v_1..v_p)\prod_{k=1}^{p}\pi_{y_j}(v_k)]$$

Where $V_1..V_p$ are causes of Y_j other than X, and α is a normalizing factor, which is calculated after finding π(x)λ(x).

Every node contains an associated CPT, and belief, π, and λ array. The CPTs are generated by the learning engine, and loaded from text files at runtime. When the network is first created, the following boundary conditions can be applied:

- Root node's π values are set to the same values found in the CPT
- All[2] node's λ arrays are set to all (1,1,…,1,1)

When the interface provides a date, the inference engine loads the appropriate satellite images. For every pixel, each node is instantiated to the appropriate state, and π and λ messages passed (equations 4, 5). The *Profile* node can then calculate its local λ, π and then belief values (equations 3, 2 and 1 respectively). The profile with the greatest associated belief is then the profile predicted for that point.

6. INTERFACE

6.1. Overview

Designing an interface for the plankton prediction system involved a user-centred design approach.

An alternative mechanism to the static graph visualisation technique used by most other Bayesian prediction programs was needed. This mechanism was required to provide the user with an intuitive and easy-to-use mechanism for viewing the Bayesian network.

The Bayesian network structure needed to be passed between the inference engine and the interface in order for it to be visualised. For this purpose, topic maps are used to represent and store the Bayesian network structure and beliefs.

Given that the Bayesian network is stored in topic map, the problem of visualizing the Bayesian networks is extended to visualizing topic maps, given that the networks are stored in topic maps.

Based on research about visualizing large graphs, the technique that was chosen to visualise the topic maps (and in turn the Bayesian network) is a hyperbolic tree visualisation.

6.2 Testing and Evaluation

6.2.1 Plankton Prediction Interface

The study to deduce the usability and functionality of the plankton prediction interface was a qualitative study based on interviews and pluralistic walkthroughs of the system.

Participants
The sample consisted of 5 people (n=5, 1 stakeholder, 4 students).

Equipment

- A computer with the plankton prediction software installed.
- Notepad and pen

Method
Each participant was interviewed individually. Given that the system is mainly used to visualise the information predicted, the users were asked questions about where they thought they would find and access certain information. The users were asked to run the system for a particular date and then answer questions/tasks about the interface. Having completed tasks, the users were asked questions about how easy the interface was to use and what they would like changed.

6.2.2 Hyperbolic Tree Representation of Bayesian Networks

In order to evaluate the usability of the hyperbolic tree representation of Bayesian networks, a comparative study between the hyperbolic representation and the static representation (used in BayesiaLab) of Bayesian networks was performed. A combination of qualitative and quantitative methods was used to test the usability of each of the system.

Participants
The sample consisted of 16 science students (n=16, 11 males, 5 females). The
age of the participants ranged from 20 to 28.

Equipment
• Computer with BayesiaLab program and hyperbolic tree representation program.
• Notepad and pen
• Stopwatch

Method
The participants were tested individually. The 16 participants were sampled into 2 different groups of equal size (8 participants in each group).

[2] The boundary condition is actually only leaf nodes, but if all nodes are un-initialised, all λ values are can be set to (1,1,…,1,1)

Group 1 was required to use BayesiaLab first and then use hyperbolic tree representation. Group 2 first used hyperbolic tree representation and then BayesiaLab. The advantage to this approach is that all the participants used each interface. Therefore, the participants could make meaningful comparisons between the interfaces.

The same Bayesian network was used for each interface.

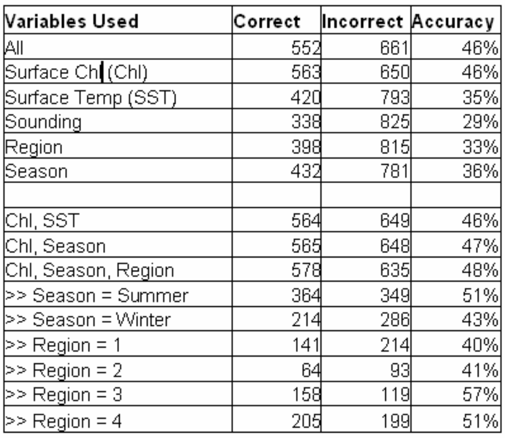
The experiment was split into 4 sections as follows:

1. The users were given a basic description of Bayesian networks.
2. Conceptual model extraction of each interface. Once the users had developed a basic understanding of Bayesian networks, the users were asked questions about what they thought represented cause and effect relationships and what represented states. This was used to determine users understanding of the Bayesian network visualisation.
3. Recording performance time on tasks. The users were given a set of tasks to perform. The tasks involved finding the percentage chance of a state of a node occurring (task type 2), and which nodes affected or were affected by a particular node (task type 1). This was used to determine which interface was easier to use, based on time.
4. A post task interview. An interview was conducted to deduce how the participants felt about each interface. This qualitative measure is used to determine which interface was easier to use.

7. RESULTS

7.1 Accuracy of Predictions (MLE)

In order to test the correctness of the final prediction, data from the actual training set was used to instantiate each node. Inference was then used to predict a profile, which was compared to the actual profile listed in the training set. If these values matched, the predicted profile was correct. Additionally, by using different combinations of variables, one can determine what effect each variable has on the accuracy of the prediction. The results from these tests have been shown below.



Variables Used	Correct	Incorrect	Accuracy
All	552	661	46%
Surface Chl (Chl)	563	650	46%
Surface Temp (SST)	420	793	35%
Sounding	338	825	29%
Region	398	815	33%
Season	432	781	36%
Chl, SST	564	649	46%
Chl, Season	565	648	47%
Chl, Season, Region	578	635	48%
>> Season = Summer	364	349	51%
>> Season = Winter	214	286	43%
>> Region = 1	141	214	40%
>> Region = 2	64	93	41%
>> Region = 3	158	119	57%
>> Region = 4	205	199	51%

Table 1: Accuracy using results from MLE algorithm, with different variables

The predictions when using all variables is less than 50%, which is fairly disappointing. Also, the prediction is more accurate by removing the SST and sounding variables, which is fairly unexpected.

The key reason for these results is the use of clustering on parameters that define a Gaussian approximation to an actual profile. The fact that additional variables can reduce the accuracy of the solution probably means the clusters generated were not significantly related.

Although results above are worse than expected, the images resulting from inference over an entire map appear fairly promising. (Figure 8)

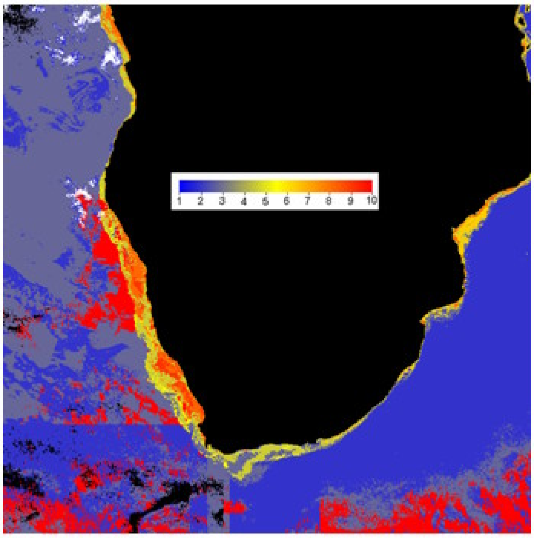
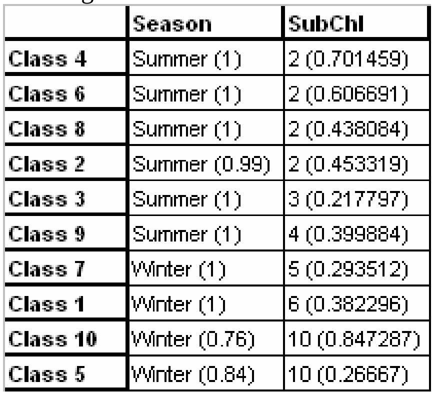


Figure 8: Predicted profiles

While it is fairly difficult to know if a predicted profile at one point on a given day is correct, the general results do agree with previous research: areas with a combination of high surface chlorophyll, and temperature of 12-15° C, show plankton profiles with large integral values. Note that areas which do not fall in the training regions can not be expected to return accurate results. This can be seen by the large red areas that occur far away from the coast.

7.2 EM Clusters

The EM algorithm was intended to create new clusters, each with an associated profile. The aim was to infer a profile for each class, by instantiating the class node, and predicting a sub-surface amount for each depth interval. However, this could not be done because both depth and sub-surface chlorophyll is dependant on the class. The clusters obtained are therefore difficult to interpret. It was decided that forward-inference would be used on each class to determine patterns in the clusters. A subset of the resulting table is shown below.



	Season	SubChl
Class 4	Summer (1)	2 (0.701459)
Class 6	Summer (1)	2 (0.606691)
Class 8	Summer (1)	2 (0.438084)
Class 2	Summer (0.99)	2 (0.453319)
Class 3	Summer (1)	3 (0.217797)
Class 9	Summer (1)	4 (0.399884)
Class 7	Winter (1)	5 (0.293512)
Class 1	Winter (1)	6 (0.382296)
Class 10	Winter (0.76)	10 (0.847287)
Class 5	Winter (0.84)	10 (0.26667)

Table 2: Some results from EM algorithm, each cell contains: State (Probability)

The most significant pattern mined is that all the low sub-surface chlorophyll values have been placed into clusters in summer and higher values in winter. The season also appears to be a highly significant variable when clustering, shown by the high probabilities.

7.3 Hyperbolic Tree Visualisation Evaluation

7.3.1 Conceptual Model Extraction

Both the static and hyperbolic interfaces were easily interpreted by the participants. The hyperbolic interface was neither better nor worse to understand.

7.3.2 Performance

The results from the experiment on performance show that the hyperbolic representation allowed for faster access/ viewing/ retrieval of probabilities associated with states.

Using the hyperbolic interface did not improve participants' ability to find connected nodes. However the participants performed equally well, using each of the interfaces for this task.

7.3.3 Interface Preference and Ease of Use

The results from the post task interviews indicated that the users predominantly found the hyperbolic interface easier to use and preferred it. However, this result is questionable, given that experimenter effect may have confounded this result.

It was found that the hyperbolic interface was at least as easy to use as the static representation used in BaysiaLab. However, it could not be conclusively proven that it was indeed significantly easier to use.

Given that the participants were science students, the results cannot be generalised to the general population. Furthermore, the interfaces were only tested on using one contrived Bayesian network and it is uncertain whether the same results would be achieved using a different network.

7.4 Interface Evaluation

In terms of ease of use, all the participants found the system relatively straightforward. The primary stakeholder, who had been involved in the design process found it easy to use. Furthermore, the primary stakeholder was able to interpret the information represented in the hyperbolic representation of the Bayesian network.

8. CONCLUSION

It has been shown that Bayesian networks can be used to predict chlorophyll profiles. Unfortunately, due to the use of Gaussian approximations, the accuracy of results was slightly less than 50%.

While these results are somewhat disappointing, the scope for future work in this field is immense. Regardless of accuracy, results for inference on a map revealed the prediction patterns which agreed with current research in the field. Furthermore, the EM algorithm did find some patterns in the data, such as different sub-surface chlorophyll amounts occurring in different seasons.

With the inclusion of more variables, and further use of the EM algorithm to generate profiles, the future of Bayesian networks appears promising in this field of research.

Topic maps are used to represent and store the Bayesian network structure and beliefs. The topic maps serve as a mechanism for passing information between the Bayesian network component of the system and the interface. If the system were to be expanded, topic maps could provide a useful mechanism for representing the Bayesian network to be passed between prediction systems.

9. FUTURE WORK

- The EM algorithm could be extended to check for *'overfitting*'. At the moment, the EM algorithm uses *'parametric learning'*. This is where the number of probability distributions across the hidden node are predefined. *'Nonparametric learning'* is where the number of these distributions are not predefined. While this makes the algorithm more complex, it allows the 'fitting' of the data to an optimal number of distributions.
- The structure of the network could be changed to show the causal relationship between the sub-surface data. At the moment it is assumed that this data is all conditionally independent. This is not entirely correct as sub-surface chlorophyll and temperature vary with regards to depth.
- There are other factors that influence the production of plankton. These include light, currents and wind. All of this data could be obtained from satellite data. Unfortunately we were unable to obtain this data in time, but if it is added to the dataset in the future, and the network is updated, the quality of the prediction will improve
- The coastal waters around South Africa were divided up into 4 regions when discretizing the data. These four regions could be further subdivided into known areas where there are noticeable differences in the characteristics of the environment variables. This would make the prediction for a given area more precise
- No discretization algorithm was used when discretizing the data. In the future, implementing a mathematical discretization algorithm would improve the ranges that the data is divided up into.
- Due to the fact that the dataset that we received did not represent every month of the year, we were unable to create a Dynamic Bayesian network. With time, as the amount of historical data increases, it will be possible to convert from a 'Static' Bayesian Network, to a 'Dynamic' one.
- In the future, this project could be extended to represent parts of the ocean, other than just off the coast of South Africa. This could be useful in discovering global trends.

- Replicate the comparative study between the hyperbolic visualisation and static visualisation of the Bayesian network. However, the study should be conducted with a larger and more representative sample, as well as with different Bayesian networks.

- Investigate the notion of information scent (as researched by Pirolli [11]) in Bayesian networks, to further investigate the applicability of visualising Bayesian networks in hyperbolic trees.

- Representing Bayesian networks in topic maps in conjunction with conditional probability tables and beliefs.

[1] A.J. Richardson, N.F. Silulwane, B.A. Mitchell-Innes, F.A. Shillington. *A Dynamic Quantitative Approach for Predicting the Shape of Phytoplankton Profiles in the Ocean.* Science Direct, 59, 301-319, 2003

[2] A.J. Richardson, C. Risien, F.A. Shillington. *Using Self-Organizing Maps to Identify Patterns in Satellite Imagery*. Science Direct, 59, 223-239, 2003

[3] K. Murphy. *A Brief Introduction to Graphical Models and Bayesian*
Networks, 1998 [Online].
Available:
http://www.ai.mit.edu/~murphyk/Bayes/bintro.html

[4] M. Jordan. *Learning in graphical models.* MIT Press, 1998

[5] Russel, S. & Norvig, P. 1995, *Artificial Intelligence: A Modern Approach.*
Upper Saddle River, New Jersey: Upper Saddle River: Prentice Hall

[6] Y. Weiss, *Belief Propagation and Revision in Networks with Loops.* Massachusetts Institute of Technology, 1997

[7] Pepper, S. "The TAO of Topic Maps, Finding the Way in the Age of the Infoglut", site: http://www.ontopia.net/topicmaps/materials/tao.html

[8] Pepper, S. & Moore, G (eds). "XML Topic Maps(XTM) 1.0" 2001 http://www.topicmaps.org/xtm/1.0/

[9] Lamping,J., Rao, R & Piroll, Pi. "a focus+Context technique Based on
Hyperbolic geometry for visualizing Large Hierachies" ,1995, site:
http://www.acm.org/sigchi/chi95/proceedings/papers/jl_bdy.htm

[10] Lamping, J. and Rao, R. 1994. Laying out and visualizing large trees using a
hyperbolic space. In Proceedings of the 7th Annual ACM Symposium on User
interface Software and Technology (Marina del Rey, California, United States,
November 02 - 04, 1994). UIST '94. ACM Press, New York, NY, 13-14. DOI=
http://doi.acm.org/10.1145/192426.192430

[11] Pirolli, P., Card, S. K., and Van Der Wege, M. M. 2003. The effects of information scent on visual search in the hyperbolic tree browser. *ACM Trans. Comput.-Hum. Interact.* 10, 1 (Mar. 2003), 20-53. DOI= http://doi.acm.org/10.1145/606658.606660