PROBABILISTIC GRAPHICAL MODELS- Extra Project

Name: Kashyap Nagaraja UIN:126003913

Problem Statement

A gene regulatory network dataset is given. It has 20 genes. The problem is to find appropriate bayesian network structure which includes all the genes. The bayesian network is such that it is:

1. Singly connected graph
2. Maximum indegree ≤2
3. Maximum outdegree ≤2

Algorithm for creating the bayesian network

Phi-coefficient is a measure to quantify the association between two binary variables. It is the equivalent of correlation of numeric variables. The variables which are most correlated are connected.

We need to connect to a variable at most two parents and two children because of indegree and outdegree constraints. However it is possible that the same node(feature or variable) can be correlated well(in top two) with more than four variables. So our indegree and outdegree constraints are not taken care. Hence we need some sort of **priority ordering.** Using this ordering we can take care of the above conflict. We define and use **alpha scores** for this purpose. The full algorithm is given in next page.

Algorithm for finding bayesian network structure.

for all variables i :

alpha_score[i]=sum of correlations of i wrt all other(19) variables.

end

Sort the alpha_score vector in descending order. The variable whose alpha score is highest gets the highest priority and so on.

Put a correlation threshold ***δ*** *so that any two variables having correlation less than* ***δ*** *will not be connected.*

for all variables i in the descending order of alpha :

find ***2*** *children to node i by considering top 2 nodes having highest correlation.*

If any of these two nodes has already indegree of 2 then find the node with next best correlation. (For example if the node with second best correlation has already indegree of 2 then find the third best node and so on)

Connect a directed edge with i as parent and the above found 2 nodes as children.

end

The above procedure does not lead to fully connected graph and lead to ***p*** *disconnected sets(say). Hence we have to make* ***(p-1)*** *minimum extra connections to make the graph fully connected by connecting disconnected set i to set j with* ***i<j****. For this select the pair of nodes having highest correlation with first node from set* ***i*** *and second node from set* ***j*** *and remove the lowest correlation child node for the connecting node in set* ***i(****node within set* ***i).*** *Similarly remove lowest correlation parent node for the connecting node in set* ***j.***

The above algorithm gives a bayesian network which is fully connected and each node has indegree and outdegree ≤2.

The graph obtained is as shown below. The node **i** represents the variable $\mathbf{X_i}$.

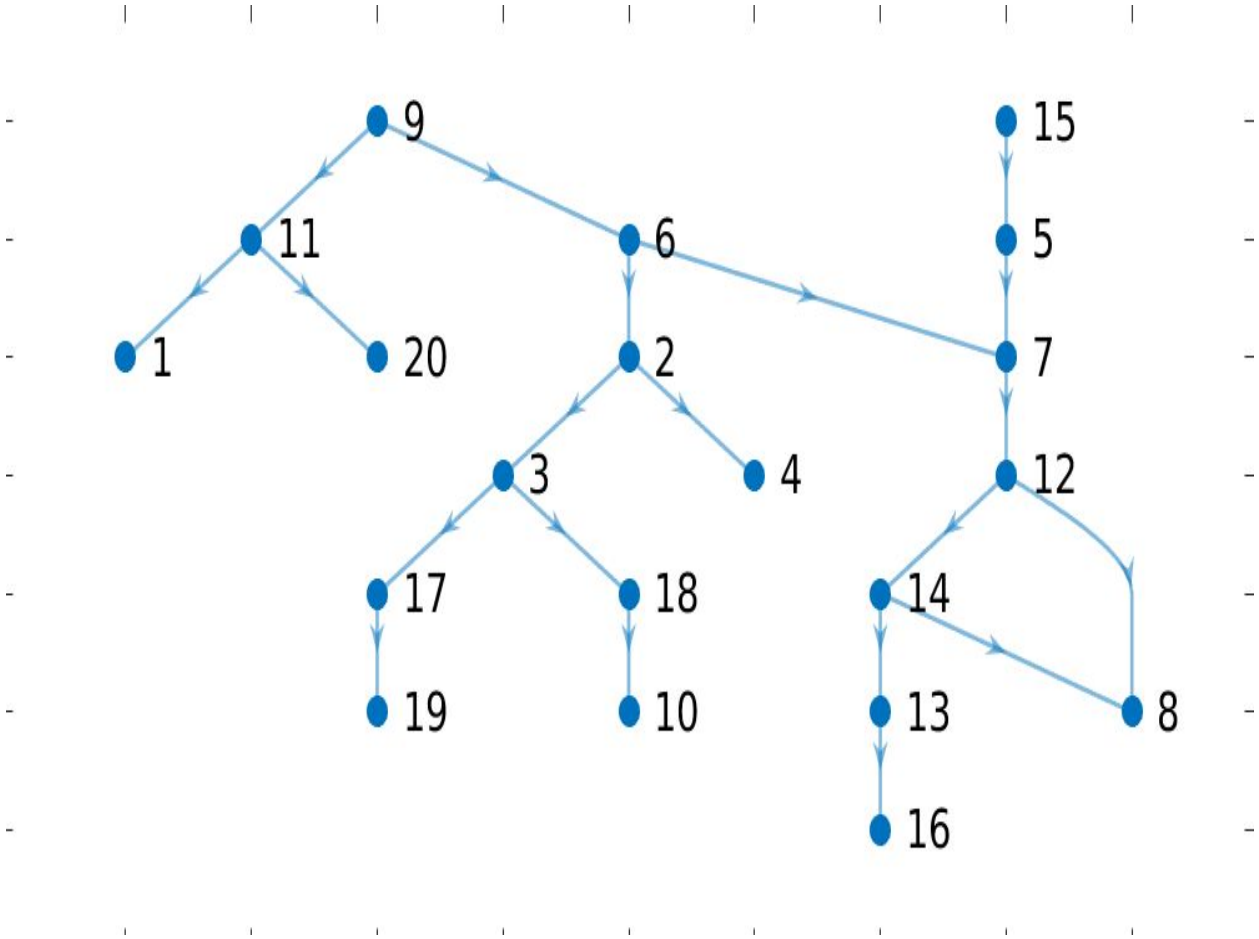

Fig 1: Bayesian network for the given data.

Results

1. **CPD Values**

   The CPD values for different variables are given in below :

$\mathbf{X_1}$	0.6955	0.6788		
$\mathbf{X_2}$	0.5063	0.4392		
$\mathbf{X_3}$	0.7706	0.2017		
$\mathbf{X_4}$	0.9063	0.1050		
$\mathbf{X_5}$	0.3340	0.3041		
$\mathbf{X_6}$	0.8419	0.1441		
$\mathbf{X_7}$	0.0809	0.0640	0.8547	0.8844
$\mathbf{X_8}$	0.4127	0.3817	0.9532	0.9670
$\mathbf{X_9}$	0.5696			
$\mathbf{X_{10}}$	0.1091	0.6370		
$\mathbf{X_{11}}$	0.4628	0.4710		
$\mathbf{X_{12}}$	0.7068	0.6728		
$\mathbf{X_{13}}$	0.2743	0.7636		
$\mathbf{X_{14}}$	0.7928	0.2619		
$\mathbf{X_{15}}$	0.4905			
$\mathbf{X_{16}}$	0.0946	0.8025		
$\mathbf{X_{17}}$	0.2740	0.7255		
$\mathbf{X_{18}}$	0.6920	0.2064		
$\mathbf{X_{19}}$	0.8660	0.0942		
$\mathbf{X_{20}}$	0.7500	0.1542		

Table1: Table for CPDs of different variables.

In the above table for a given variable X_i the probability that $P(X_i|\text{parents of } X_i)$ are calculated. For variables with one parent the first column represents parent=0 and second column represents parent=1.

For variables with two parents, the parent values are in order 00,01,10,11.

2. Summary about the predicted values, Actual values and the value of p_k^j

For most of the cases there is a **good variance** of the probability values. There are many cases where the predictor is very confident(p_k^j >0.8 or p_k^j<0.2) of the state(0 or 1) and many cases where it hangs around 0.5. Usually when the confidence is high the error rate is low.

For accessing the predicted value the variable **predicted_value** can be accessed for each row. Similarly for each the variable **conf_matrix** can be accessed for each row. Both these variables are from **Main_top_module.m.**

There is a folder called **predicted_values_for_test_data_sets_a_b_c_d** where all the predicted values(from column 11 to column 20) for all 4 files has been put.

3. Table with average prediction accuracy and expected prediction for each of the 20 genes.

The table showing average prediction accuracy for each of the 20 genes is as shown below:

Variable	test50a.txt	test50b.txt	test 50c.txt	test50d.txt
$\mathbf{X_{11}}$	0.4285714286	0.5714285714	0.5306122449	0.4285714286
$\mathbf{X_{12}}$	0.6530612245	0.7142857143	0.7755102041	0.693877551
$\mathbf{X_{13}}$	0.4489795918	0.5918367347	0.4489795918	0.387755102

X_{14}	0.4897959184	0.4081632653	0.4489795918	0.4897959184
X_{15}	0.5306122449	0.612244898	0.5918367347	0.4897959184
X_{16}	0.4285714286	0.5510204082	0.4489795918	0.4489795918
X_{17}	0.7551020408	0.7142857143	0.7346938776	0.7346938776
X_{18}	0.7551020408	0.693877551	0.6326530612	0.8367346939
X_{19}	0.7346938776	0.6734693878	0.6734693878	0.5510204082
X_{20}	0.5714285714	0.4285714286	0.5306122449	0.5306122449

Table2: Average prediction accuracy for each feature.

The table showing expected prediction accuracy for each of the 20 genes is as shown below:

Variable	test50a.txt	test50b.txt	test 50c.txt	test50d.txt
X_{11}	0.5325325325	0.5325325325	0.5325325325	0.5325325325
X_{12}	0.6957040056	0.6966133899	0.6971131342	0.696417984
X_{13}	0.5185206771	0.743483741	0.7496713224	0.7465775317
X_{14}	0.5765809883	0.7671187862	0.7581990047	0.7760092395
X_{15}	1	0.5104548089	0.5085296187	0.5130217291
X_{16}	0.5645660926	0.6755125155	0.6630909745	0.669301745
X_{17}	0.72574226	0.7257534661	0.72574226	0.7258094965
X_{18}	0.7396837757	0.7376105681	0.7396837757	0.7272445299
X_{19}	0.673011777	0.6722079439	0.673011777	0.6681887783
X_{20}	0.5285285285	0.5285285285	0.5285285285	0.5285285285

Table3: Expected prediction accuracy for each feature.

4. Table with actual overall prediction accuracy and expected overall prediction accuracy for all 4 files.

Filename	Actual accuracy	Expected Accuracy
test50a.txt	0.58	0.655
test50b.txt	0.6	0.659
test50c.txt	0.582	0.6576
test50d.txt	0.5692	0.657

Table4: Overall and expected prediction accuracies.

5. Comments and Discussion

As we can see the **overall predicted accuracy** hovers around 60% while the **expected accuracy** is around 65% . The following maybe the main reason for this level of accuracy.

1. We restricted indegree and outdegree to just two which may have neglected some of the key dependencies.

2. It is possible that we have lesser number of samples when compared to the number of features thereby overfitting the data.

3. The expected accuracy is more than the overall prediction accuracy. This suggests that the approximation we do when probability is greater than 0.5 to a particular state(up or down regulated) is introducing noise.

PREDICTION CHALLENGE

For the prediction challenge a challenge file named “challenge200.txt” is downloaded. Each row has several missing values and the given values are used as evidence. Using that the missing values are predicted.

Please find four files related to **prediction with 11,13,15,17** in the folder **Prediction_challenge.** Also find the consolidated file **ECEN_760_Nagaraja_Kashyap_challenge.txt.**

Please read the file **ECEN_760_Extra_Project_Kashyap_Nagaraja_Readme** for how to run the code.

****