Prediction in Classification Task using Bayesian Networks

1. Introduction

The objective of this assignment is to design, train and test Bayesian Network in order to make predictions.
The given dataset *"leukemia.dat"* consists of 73 examples, each one of them with 5 features and a label (the target); the features are the binary state of a gene (active or inactive) and the label (AML/ALL) representing the two possible types of leukemia.

The first step among all is to define the structure of the Bayesian Network; three different approaches have been tried out:

(1) using *NPC* algorithm
(2) using *Greedy Search-and-Score* algorithm
(3) using the fixed *Naive Bayes* structure

In the first two cases the structure is learnt from data, in the third one it's fixed in advance.

2. Procedure

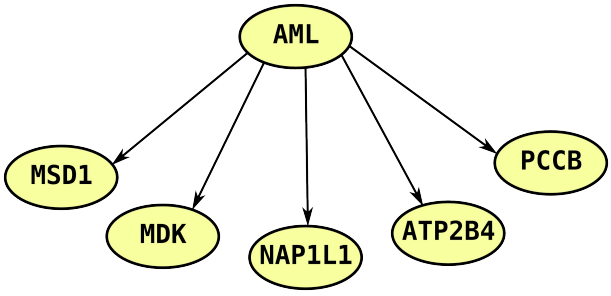
Next up is the parameter learning step: the dataset has been divided in two smaller ones (training set [80%] and test set [20%]). The parameters are estimated using the training part of the data and since the size of the dataset is quite small, some low fixed prior (0.1) has been added to the model in order to avoid seemingly impossible configurations that can be observed in the test dataset.

(1) NPC

The NPC algorithm is a constraint-based approach to the structural learning of the network. It's based on statistical tests done on each pair of features, and the edges are added if the two variables are conditionally dependent (according to the input data). In order to obtain a Directed Acyclic Graph the direction of the links are assigned after the construction of the so called "skeleton" that consists of solely undirected links.

Confusion Matrix

Pred\Actual	**Yes**	**No**
Yes	4	2
No	1	8



Error rate: 13.33

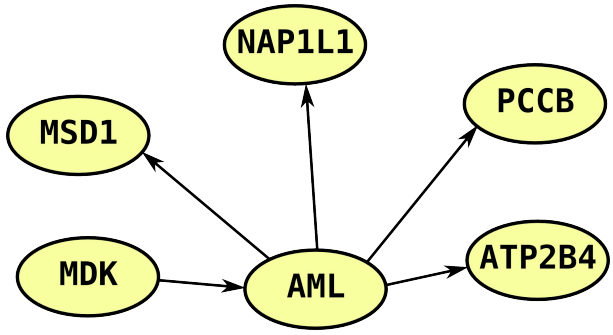
(2) Greedy Search-and-Score

This method constructs the network trying out different configurations and returning the one that scores “better”. Obviously a score function has to be defined and an important property of it should be the decomposability, simplifying the computation of the score even with small changes (such as the one done by the searching algorithm: add, remove and invert an edge).

Confusion Matrix

Pred\Actual	**Yes**	**No**
Yes	4	1
No	1	9

Error rate: 13.33



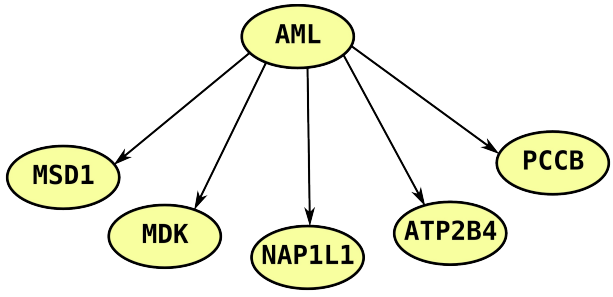
(3) Fixed Naive Bayes structure

The Naive Bayes approach makes a simplistic assumption: every feature is independent of each other given the class (the target label). From the d-separation rules we know that the wanted property is given by the tail-to-tail configuration, where the class is the father of all the features and no other edges are present.

Confusion Matrix

Pred\Actual	**Yes**	**No**
Yes	4	1
No	1	9

Error rate: 13.33



3. Analysis

Due to the small size of the dataset and the lack of background knowledge about the topic, it's hard to conclude any certain statement about the performances of the different structures of BN and the goodness of the underlying model regarding this application