Data Mining & ML

lewiswilson1497

November 2020

Contents

1 File management, data pre-processing, transformation and selection

test

1.1 File Management

1.2 Data pre-processing

1.3 Transformation and selection

2 Naive Bayesian Networks

3 Complex Bayes nets

Describe & analyse the problem. Show all experiments complete with graphs and tables. Discuss produced software quality & discuss interesting properties of the data and algorithms

3.1 Building Bayse Networks

Bayse networks represent probablistic directed acyclic graphs that define the relationshps between conditional dependencies and random variables. A naive Bayesian network can be represented in a Bayse network where the node representing the probability distribution of the class is the only parent of all other nodes and no other edges exist in the network. By adding additional edges (so long as the graph remains acyclic) we can represent causal relations between random variables.

We decided to approach this task using both Weka and Python, with the intention of verifying our results against the other. Attempting to build the network both ways gave us solid insights into the problems that would have to be solved to produce a Bayse Network. We decided to use the pgmpy library to build our Bayse Networks in python, this immediately presented us with 2 computational complexity problems:

1. Building all the conditional probability factors.
2. Learning the optimal edges.

We handled the first problem by discretizing the greyscale values using equal width and frequency binning (see section 1.3) When using Weka to compute Baysian networks we observed that Weka would perform extremely aggressive binning of the greyscale values often discretizing down to only 2 bins. This had a profound effect on the speed of learning the parameters and edges.

3.2 Algorithms & Data

3.3 Experimental Results

4 Clustering