Assignment 3: Probability

CS 6601

Due October 15 by 9:35 AM

Abstract

You will implement several Bayesian networks and sampling algorithms to gain a better understanding of probabilistic systems.

1 The Challenge

Many AI systems rely on probabilistic knowledge of the world, rather than absolute knowledge, to execute tasks efficiently: for example, motion planning in robots with unreliable sensors. One type of probabilistic system that is especially useful is the Bayesian network, which encodes a joint probability distribution among dependent variables as a network of conditional probabilities. Your challenge is to implement and test several of these networks, ultimately using a sampling method to approximate a probability distribution.

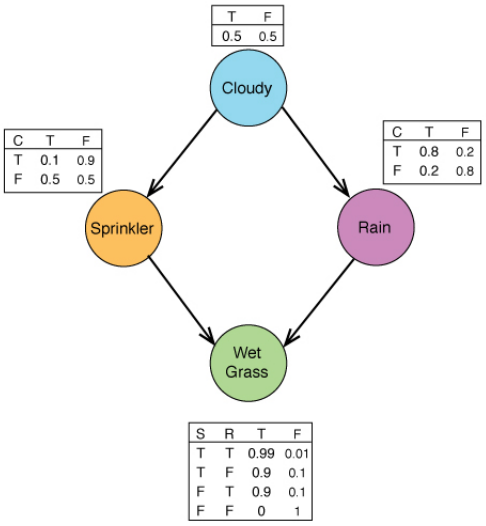

Figure 1: Example Bayesian network
(representing prediction for wet grass)

2 Your Assignment

Your task is to implement a few basic networks as well as several sampling algorithms.

You will do this in probability_notebook.ipynb, and there are tests along the way to help. Unlike previous assignments, we will not be grading on performance but rather on completion.

We will provide the following additional classes and files:

File	Description
probability_tests.py	a module containing partial tests for your algorithms
pbnt	a module to implement Bayesian networks (you'll mostly need BayesNode in Node.py and BayesNet in Graph.py)

This is meant to be a shorter assignment, so there won't be much testing required.

3 Grading

Each section of the assignment is associated with a number of points, as follows (out of 100 points total):

Warmup 1a: Build a basic Bayesian network representing a power plant. (10 points)

Warmup 1b: Answer a question about polytrees. (5 points)

Warmup 1c: Set the probabilities for the Bayes Net. (15 points)

Warmup 1d: Use inference to calculate several marginal probabilities within the Net. (10 points)

Exercise 2a: Build a Bayesian network representing a sports competition. (10 points)

Exercise 2b: Implement Metropolis-Hastings sampling. (15 points)

Exercise 2c: Implement Gibbs sampling. (15 points)

Exercise 2d: Compare the two sampling methods based on how quickly they converge. (15 points)

Exercise 2e: Answer a question about time complexity. (5 points)

4 Due date

This assignment is due on T-Square Tuesday October 15th by the start of class (9:35 AM). The deliverables for the assignment are:

- A filled out version of the iPython notebook provided. (probability_notebook.ipynb)

Please submit this in iPython notebook format - it makes grading much easier.

5 Resources

If you want to know more about how pbnt works, check out exampleinference.py and water() in pbnt/combined/ExampleModels.py.

You will need to run your code in iPython2, due to version issues in pbnt. There are instructions for how to do this within probability_notebook.py.

For more information on sampling probability distributions, see section 14.5 in Russell and Norvig (pp. 535-538 for Gibbs sampling).

Although you don't have to implement the inference algorithm that you'll use with your networks, you might be interested in knowing how it works. You can find details on pp. 529-530 of Russell and Norvig and here.

As always, TAs will hold office hours Monday, Tuesday, Thursday and Friday from 2:00 to 4:00 PM outside TSRB 241.