CS 6601 Assignment 3: Probability

Due October 9, 2016, by 11:59PM UTC-12 (Anywhere on Earth)

Notes:

- **Use Python 2.7 for this assignment.**
- **Submit the solution as a Python file : probability_solution.py on Bonnie as well as t-square.**
- **DO NOT CHANGE any function headers.**
- **If you are using the notebook provided to code, 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.ipnb.**

Abstract

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

Learning Objectives

Students should be able to understand the importance of Bayesian networks to represent conditional dependencies. Also, be able learn the sampling methods, Gibbs and Metropolis-Hastings and develop an intuition for their convergence criteria (very "researchy").

Evaluation

Evaluation is using the last submission on Bonnie.

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.

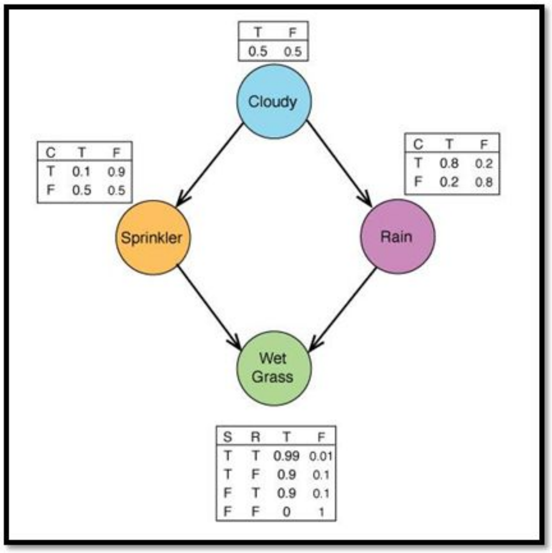

Figure 1: Example Bayesian network
(representing prediction for wet grass).

2. Your Assignment

Your task is to implement a few basic networks as well as the Gibbs sampling algorithm. There is a bonus section too where you need to implement Metropolis Hastings sampling and compare the results between the two. On completing the bonus section correctly, you will score up to 2% points on your final grade.

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

We have provided the following additional classes and files:

File/Folder	**Description**
probability_tests.py	To test the models you've built.
pbnt/combined	Module to implement Bayesian networks (you'll basically need BayesNode in Node.py and BayesNet in Graph.py). Also contains an example (ExampleModels.py) to help you get started.

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

3. Grading

BASIC TASK (100 points)

Warmup 1a: Build a basic Bayesian network representing a power plant. (10 points)
Warmup 1b: Answer a question about poly-trees. (5 points)
Warmup 1c: Set the probabilities for the Bayes Net. (10 points)
Warmup 1d: Use inference to calculate several marginal probabilities within the Net. (15points)

Exercise 2a: Build a small Bayesian network representing a sports competition. (10 points)
Exercise 2b: Calculate likelihoods for the 3rd match. (5 points)
Exercise 2c: Implement Gibbs sampling. (20 points)
Exercise 2d: Count the number of iterations it takes to converge to a stable distribution and return the estimated likelihood. (20 points)
Exercise 2e: Answer a question about time complexity. (5 points)

BONUS TASK (optional: up to 2% points added to final class grade)

Exercise 3a: Implement Metropolis-Hastings sampling and convergence. (20 points)
Exercise 3b: Compare the performance of the 2 sampling methods on the Sports network and answer the sampling question. (10 points)

4. Due date

This assignment is due on Bonnie and T-Square by **October 9th, at 11:59PM UTC-12** (Anywhere on Earth). The deliverable for this assignment is a **Python file** :

- probability_solution.py

5. Resources

IMPORTANT: If you want to know more about how pbnt works, check out exampleinference.py and water() in pbnt/combined/ExampleModels.py. Also here's a clone of the library : https://github.com/achille/pbnt.

Basics of Bayes nets and Conditional Probability:

- https://www.mathsisfun.com/data/probability-events-conditional.html
- https://ocw.mit.edu/courses/mathematics/18-05-introduction-to-probability-and-statistics-spring-2014/class-slides/

Gibbs Sampling and convergence:

- http://gandalf.psych.umn.edu/users/schrater/schrater_lab/courses/AI2/gibbs.pdf
- https://en.wikipedia.org/wiki/Gibbs_sampling
- https://www.youtube.com/watch?v=ol0l6aTfb_g
- Section 14.5 in Russell and Norvig (pp. 535-538 for Gibbs sampling).

Metropolis Hastings and convergence:

- https://www.cs.cmu.edu/~scohen/psnlp-lecture6.pdf
- http://www.bcs.rochester.edu/people/robbie/jacobslab/cheat_sheet/MetropolisHastingsSampling.pdf

Inference Algorithm (not needed for this assignment):

Although you don't have to implement the inference algorithm (Junction tree) 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.