CS267A: Probabilistic Programming and Relational Learning **Spring 2020**

Homework 3: Due Friday May 1, 11:59PM

Instructions: Upload one file to CCLE: a PDF typeset using LaTeX containing your solutions (**including code**). Consider using package *listings* to embed the code. No late submissions will be accepted. See the syllabus for policies about collaboration and academic honesty.

A few things to watch out for when using Dice:

- We recommend directly using the compiled binaries of Dice instead of compiling it from the source. It can help minimize the errors you may run into. Linux and Mac users can access the binaries at https://github.com/SHoltzen/dice/releases. At this moment, Dice does not support Windows systems, and we recommend Windows users to install a Linux virtual machine or use Seaslab machines.
- Whenever using an integer, the range of the integer needs to be specified. For example, int(10,0) expresses integer 0 in the range of [0, 9].
- When doing operations over integers, the range of integers needs to match.
- When specifying a random variable taken over a distribution, the distribution has to be normalized (i.e. its corresponding parameters sum up to 1).
- The parameters of a distribution have to be non-negative float numbers (i.e. no rationals). For example, use 0.0 to express zero.

Problem 1

Probability of Satisfaction

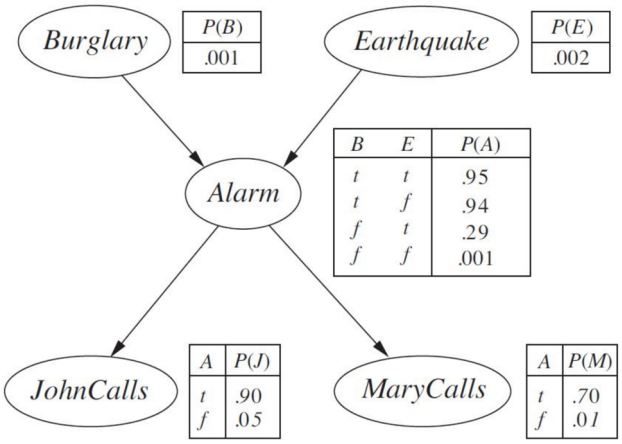
In the last homework, you were asked to implement a sampler to approximately evaluate the following expressions. This time, let us check how accurate the approximations are. To be specific, please use Dice to get the exact answer and attach your code along with your answer.

1. $(a \vee b \vee \neg c) \wedge (b \vee c \vee d \vee \neg e) \wedge (\neg b \vee \neg d \vee e) \wedge (\neg a \vee \neg b)$ with $\Pr(a) = 0.3, \Pr(b) = 0.6, \Pr(c) = 0.1, \Pr(d) = 0.8, \Pr(e) = 0.4$
2. $(\neg a \vee c \vee d) \wedge (b \vee c \vee \neg d \vee e) \wedge (\neg c \vee d \vee \neg e)$, with $\Pr(a) = 0.2, \Pr(b) = 0.1, \Pr(c) = 0.8, \Pr(d) = 0.3, \Pr(e) = 0.5$

Problem 2

Bayesian Network

In the lecture, we walked through the classic "earthquake, burglary" Bayesian network example. Please use Dice to model it, answer the following queries and provide your code along with your answer.



1. What is the probability that there is a burglary or earthquake given both John and Mary call?
2. What is the probability that there is an earthquake given no burglary and Mary calls?

Problem 3

Monty Hall problem

In a game show, a contestant is given the choice of three doors: one door has a car behind it, and the other two doors have goats. Suppose the contestant chooses the first door. Then the game host, who knows what is behind the doors, reveals one of the remaining two, showing it has a goat.

The contestant has the option to switch. In each of the following scenarios, what is the probability that the contestant wins the car after switching? Should the contestant switch? Please attach your Dice code along with your answer.

1. The car is equally likely to be behind any one of the three doors.
2. The car is behind the first door with probability 0.6, the second with probability 0.3, and the third with probability 0.1. For this specific sub-problem, one can assume the contestant always chooses the first door at the beginning.

Problem 4

Compilation

In the lecture, we introduced the pipeline of getting the result for a given query in Dice. One of its most critical steps is compilation. Please write down the Binary Decision Diagram the following Dice program would be compiled to? For your convenience, you can sketch out the answer on paper, and attach the photo or scan of it.

```
let c = flip 0.4 in
let x = if c then
  flip 0.6 else
```

```
  flip 0.4 in
let y = if c then
  flip 0.3 else
  flip 0.8 in
let z = if (c && x) then
  flip 0.2 else
  flip 0.7 in
z
```