# CS 337 (Spring 2019): Class Test 4

Instructor: Shivaram Kalyanakrishnan

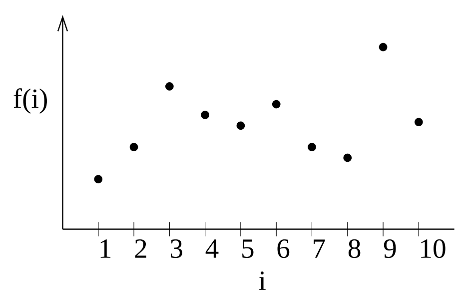
9.15 p.m. – 10.30 p.m., April 15, 2019, 101/103/105 New CSE Building

Total marks: 20

**Note.** Provide brief justifications and/or calculations along with each answer to illustrate how you arrived at the answer.

**Question 1.** Consider performing hill-climbing on the local search instance shown below. There are 10 possible solutions—1 through 10 in sequence—each with one neighbour on the left and one on the right (except 1, which only has a right-neighbour, 2; and 10, which only has a left-neighbour, 9). The version of hill-climbing applied is *greedy*, meaning that the neighbour with the highest fitness is picked at each stage, breaking ties uniformly at random.

The fitness values $f(i)$ of each solution $i$ are shown in the figure. The aim of hill-climbing is to maximise fitness. For the questions below, you do not need to use numerical fitness values: instead refer to the fitness of solution $i$ as $f(i)$. Your answers can be in terms of $f(\cdot)$.

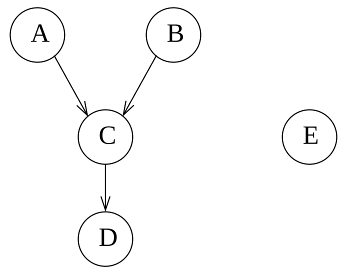



1a. If a run of hill-climbing is initialised by picking a starting solution uniformly at random from $\{1, 2, \ldots, 10\}$, what is the expected fitness of the solution returned? [2 marks]

1b. Suppose $n$ such runs are conducted (that is, with $n$ random starts), and the solution with the maximum fitness encountered across all $n$ runs is returned. What is the expected fitness of the solution returned? [3 marks]

**Question 2.** Suppose $x$, $y$, and $z$ are drawn independently and uniformly at random from $[0, 1]$. What is the expectation of $\max\{x, y, z\}$? [5 marks]

**Question 3.** This question pertains to the Bayes Net specified in the figure below. The variables are $A$, $B$, $C$, $D$, and $E$. Each variable is Boolean, taking values denoted in lower case with or without negation. The tables only show the (conditional) probabilities associated with the non-negated value.




| $\mathbb{P}\{a\}$ |
|---|
| 1/2 |

| $\mathbb{P}\{b\}$ |
|---|
| 1/3 |

| $A$ | $B$ | $\mathbb{P}\{c\|A,B\}$ |
|---|---|---|
| $a$ | $b$ | 1 |
| $a$ | $\neg b$ | 0 |
| $\neg a$ | $b$ | 1/2 |
| $\neg a$ | $\neg b$ | 0 |

| $C$ | $\mathbb{P}\{d\|C\}$ |
|---|---|
| $c$ | 1/4 |
| $\neg c$ | 3/4 |

| $\mathbb{P}\{e\}$ |
|---|
| 4/5 |

3a. Suppose an experimenter draws a sample $(a_s, b_s, c_s, d_s, e_s)$ based on the joint probability distribution of $(A, B, C, D, E)$ encoded by the Bayes Net. What is the probability that $c_s = c$? [3 marks]

3b. If the sample obtained above does indeed have $c_s = c$, what is the probability that it also satisfies $b_s = b$ and $e_s = e$? [2 marks]

**Question 4.** This question asks you to provide a useful tool to answer d-separation queries in Bayes Nets.

Assume you are provided a Bayes Net with a set of nodes $N = \{1, 2, \ldots, n\}$. An adjacency matrix $P[\,][\,]$ specifies the directed connections: for $i, j \in N$,

$$P[i][j] = \begin{cases} \text{1 if i is a parent of j, and} \\ \text{0 otherwise.} \end{cases}.$$

A subset of nodes $G \subset N$ are *given*: that is, their values are fixed.

Let $A$ and $I$ be the sets of all *active* and *inactive* triples, respectively, in the Bayes Net, taking the nodes in $G$ as given. Provide pseudocode to compute $A$ and $I$. The computations can share code. You may represent a triple as $(i, j, k)$, where $i, j, k \in N$. [5 marks]

# Solutions

**1a.** From points 1, 2, 3, and 4, the solution reached is 3, with fitness $f(3)$. From points 5, 6, and 7, the solution reached is 6, with fitness $f(6)$. From points 8, 9, and 10, the solution reached is 9, with fitness $f(9)$. Since the starting solution is picked uniformly at random, the expected fitness is

$$\frac{4}{10}f(3) + \frac{3}{10}f(6) + \frac{3}{10}f(9).$$

**1b.** Let $p = \frac{4}{10}$ denote the probability of being initialised at 1, 2, 3, or 4 on any given run. Let $q = \frac{3}{10}$ denote the probability of being initialised at 5, 6, or 7 on any given run. Let $r = \frac{3}{10}$ denote the probability of being initialised at 8, 9, or 10 on any given run. Clearly the answer returned after $n$ runs is either 3, 6, or 9, which are the only local maxima.

Observe that $f(9) > f(3) > f(6)$. 6 is returned after $n$ runs if and only if each these runs were initialised at 5, 6, or 7. The probability of this occurrence is $q^n$.

3 is returned after $n$ runs if and only if each of the runs is initialised at 1, 2, 3, 4, 5, 6, or 7, and at least one run is initialised at 1, 2, 3, 4. The probability of this occurrence is $(p+q)^n - q^n$.

9 is returned if and only if 3 and 6 are not returned; the corresponding probability is $1-(p+q)^n$.

The expectation of the fitness of the solution returned is therefore

$$q^n f(6) + ((p+q)^n - q^n)f(3) + (1-r^n)f(9) = 0.3^n(f(6) - f(3) + f(9)) + 0.7^n f(3).$$

**2.** $\max\{x, y, z\}$ is either $x$ or $y$ or $z$ depending on their relative order. We decompose it accordingly to calculate the expectation, which is

$$\begin{aligned}
&\int_{x=0}^{1}\int_{y=0}^{1}\int_{z=0}^{1} \max\{x,y,z\} dzdydx \\
&= \int_{x=0}^{1}\int_{y=0}^{x}\int_{z=0}^{x} xdzdydx + \int_{x=0}^{1}\int_{y=0}^{x}\int_{z=x}^{1} zdzdydx + \int_{x=0}^{1}\int_{y=x}^{1}\int_{z=0}^{y} ydzdydx + \int_{x=0}^{1}\int_{y=x}^{1}\int_{z=y}^{1} zdzdydx \\
&= \int_{x=0}^{1}\int_{y=0}^{x} x^2 dydx + \int_{x=0}^{1}\int_{y=0}^{x} \frac{1-x^2}{2} dydx + \int_{x=0}^{1}\int_{y=x}^{1} y^2 dydx + \int_{x=0}^{1}\int_{y=x}^{1} \frac{1-y^2}{2} dydx \\
&= \int_{x=0}^{1} x^3 dx + \int_{x=0}^{1} \frac{x-x^3}{2} dx + \int_{x=0}^{1} \frac{1-x^3}{3} dx + \int_{x=0}^{1} \left(\frac{1-x}{2} - \frac{1-x^3}{6}\right) dx \\
&= \frac{1}{4} + \frac{1}{2}\left(\frac{1}{2} - \frac{1}{4}\right) + \frac{1}{3}\left(1 - \frac{1}{4}\right) + \frac{1}{2}\left(1 - \frac{1}{2}\right) - \frac{1}{6}\left(1 - \frac{1}{4}\right) \\
&= \frac{3}{4}.
\end{aligned}$$

**3a.** The probability is

$$\begin{aligned}
\mathbb{P}\{c\} &= \sum_{aa\in A}\sum_{bb\in B}\sum_{dd\in D}\sum_{ee\in E}\mathbb{P}\{aa,bb,c,dd,ee\} \\
&= \sum_{aa\in A}\sum_{bb\in B}\sum_{dd\in D}\sum_{ee\in E}\mathbb{P}\{aa\}\mathbb{P}\{bb\}\mathbb{P}\{c|aa,bb\}\mathbb{P}\{dd|c\}\mathbb{P}\{ee\} \\
&= \sum_{aa\in A}\sum_{bb\in B}\mathbb{P}\{aa\}\mathbb{P}\{bb\}\mathbb{P}\{c|aa,bb\} \\
&= \frac{1}{2}\cdot\frac{1}{3}\cdot 1 + \frac{1}{2}\cdot\frac{2}{3}\cdot 0 + \frac{1}{2}\cdot\frac{1}{3}\cdot\frac{1}{2} + \frac{1}{2}\cdot\frac{2}{3}\cdot 0 \\
&= \frac{1}{4}.
\end{aligned}$$

**3b.** The probability is

$$\begin{aligned}
\mathbb{P}\{b\wedge e|c\} &= \frac{1}{\mathbb{P}\{c\}}\sum_{aa\in A}\sum_{dd\in D}\mathbb{P}\{aa,b,c,dd,e\} \\
&= \frac{1}{\mathbb{P}\{c\}}\sum_{aa\in A}\sum_{dd\in D}\mathbb{P}\{aa\}\mathbb{P}\{b\}\mathbb{P}\{c|aa,b\}\mathbb{P}\{dd|c\}\mathbb{P}\{e\} \\
&= \frac{\mathbb{P}\{b\}\mathbb{P}\{e\}}{\mathbb{P}\{c\}}\sum_{aa\in A}\mathbb{P}\{aa\}\mathbb{P}\{c|aa,b\} \\
&= \frac{16}{15}\left(\frac{1}{2}\cdot 1 + \frac{1}{2}\cdot\frac{1}{2}\right) = \frac{4}{5}.
\end{aligned}$$

**4.** We provide an $O(n^3)$ algorithm to identify all active and inactive triples. In principle the computation can be further optimised. Also, it is a matter of convention whether one counts $(i, j, k)$ and $(k, j, i)$ as the same or as different triples. We assume that once a master list is created, the required convention can be implemented.

The solution is in two parts. First, a separate computation is undertaken to identify which nodes have "given" descendants. This information is needed for deciding if common effect triples are active. The second part of the solution merely applies known rules for determining if different types of triples are active or inactive.

```
//First we determine for each node whether it has a "given" descendant (including itself).
//We require this information for "common effect" type of triples.
For i ∈ N:
    hasGivenDescendant[i] ← False.
    If i ∈ G:
        hasGivenDescendant[i] ← True.

Repeat n times:
    For i ∈ N:
        For j ∈ N:
            If P[i][j] and hasGivenDescendant[j]:
                hasGivenDescendant[i] ← True.

//Next we implement the logic for identifying active and inactive triples.
I ← [ ].
A ← [ ].
For i ∈ N:
    For j ∈ N:
        For k ∈ N:

            If P[i][j] and P[j][k]//Causal chain
                If j ∈ G:
                    Append (i, j, k) to I.
                Else:
                    Append (i, j, k) to A.

            If P[j][i] and P[j][k]//Common cause
                If j ∈ G:
                    Append (i, j, k) to I.
                Else:
                    Append (i, j, k) to A.

            If P[i][j] and P[k][j]//Common effect
                If hasGivenDescendant[j]:
                    Append (i, j, k) to A.
                Else:
                    Append (i, j, k) to I.
```