CS 182: Problem Set 4

Alan Turing

October 31, 2018

Introduction: Welcome to the fourth official homework for CS182! As you are hopefully already aware, this PDF comprises the written component of the second problem set. In addition to solving the problems found below, you will also need to complete the coding part of the assignment, found in the Github repo. Finally, we'd like to remind you that while you are allowed a partner for the coding part of the assignment, you are **NOT** allowed a partner for this and all future written components. All written work should be yours and yours alone. This being said, in addition to being able to ask questions at office hours, you are allowed to discuss questions with fellow classmates, provided 1) you note the people with whom you collaborated, and 2) you **DO NOT** copy any answers. Please write up the solutions to all problems independently.

Collaborators:

Problem 1 (The Coin Problem) – 3 Points: Ankit and Aidi decide to play a coin game to show how we can use HMMs for sequence analysis problems. Aidi tosses first, then they take turns based on rules described below. The game finishes when the subsequence "HTH" appears, and whoever last flips the coin wins. Each player can flip the coin for multiple turns in a row, and the rules for stopping and switching to the other partner are as follows:

1. Every time Aidi flips the coin, she also flips an extra unfair coin (P(H) = 0.3). She stops if the extra unfair coin lands heads. Otherwise, she keeps flipping the fair and extra biased coin (at the same time). The flips of the extra coin are not recorded.
2. Every time Ankit flips the coin, he only flips the fair coin until H appears (and all flips are recorded).

You're given a sequence of recorded coin fips. You'd like to infer the winner and the flips of each player.

Describe an HMM to model this game (draw a diagram with nodes rep and edges/arrows).

Hint: Make sure to draw an HMM model! **NOT** an FSM!

Solution 1:

Problem 2 (Typing Simulation) – 9 Points: For this problem, you will be playing a typing simulation. Let random variable E represent the observed key press, and X represent the hidden (intended) key press. We have a language with 4 letters (A, B, C, D), and a keyboard arranged as a circle.

A	B
C	D

At any time, the probability of hitting the intended key is 50%, and the probability of hitting the neighboring keys is 25%. For example, $P(E|X = \text{B})$:

0.25	0.5
0	0.25

We will construct a filtering model for constructing the belief state for this problem.

(a) (1 Point) Assuming a uniform prior distribution, calculate the condition probability table (CPT) of $P(X = x|E = e)$ for all x and e.

(b) (2 Points) Now let the prior distribution be:

x	P(X=x)
A	0.4
B	0.2
C	0.1
D	0.3

Calculate the CPT $P(X = x|E = e)$ for all x and e.

(c) (3 Points) Consider the following transition model for $P(X'|X)$:

	A’	B’	C’	D’
Begin	1	0	0	0
A	0.5	0.5	0	0
B	0	0.5	0.5	0
C	0.5	0	0	0.5
D	0.25	0.25	0.25	0.25

For this problem we are concerned with true (hidden) state sequences, as opposed to observations. What is the probability under this model of the sequence of letters "A B B C D"? How about "A A B A"? What is $P(X_3 = x|X_1 = \text{A}, X_2 = \text{B})$ for all x?

(d) (3 Points) Finally we consider the full filtering problem in which we compute $P(X_n|E_1, \dots, E_n)$. Let "A B B C D" be the sequence of observed key strokes. What is the current belief state of the model? That is compute $P(X_n = x|E_1 = \mathrm{A}, E_2 = \mathrm{B}, E_3 = \mathrm{B}, E_4 = \mathrm{C}, E_5 = \mathrm{D})$ for all x and $n = 2, 3, 4, 5$.

Hint:

$$P(X_n|E_1, ..., E_n) \propto P(E_n|X_n) \sum_{x_{n-1}} P(X_n|x_{n-1}) B(x_{n-1})$$

Solution 2:

Problem 3 (Robotic Motion Planning) – 4 Points: Describe using pseudocode an RRT-based planning algorithm that uses more than two trees. Make sure to consider issues such as the maximum number of allowable trees, when to start a tree, and when to attempt connections between trees.

What are the types of problems for which this algorithm would perform better than RRT or bi-directional RRT?

Solution 3:

Problem 4 (Ethics) – 4 Points: This question asks you to make an ethical argument about the implementation of an algorithm for autonomous vehicles, in the context of the following hypothetical scenario we discussed in class:

You are part of a programming team hired by the major's office of San Francisco. The mayor's office wants your team to design a set of autonomous emergency vehicles to be used in the city starting in 2050, when a law will be passed forbidding any non-AV (besides bicycles, skateboards, etc.) from driving in the city.

Your task is to **pick three of ethical values** that need to be taken into account when designing AV's (they do not need to be the ethical considerations discussed in class, but they can be.) Then, **answer the following two questions**:

1. Why did you choose these three particular ethical values?
   Think about questions like:
   - What are morally important aspects of the design problem?
   - What morally important features of the situation that the AV's need to represent?
   - What goal does the AV have?
2. For each of the ethical value: do you think that this ethical value should built into the algorithm? (For example, could environmental impact be captured as a cost function, or safety be built in as a constraint?)
   - If not, why not?
   - If yes: what approach would you take to building in this ethical value? (If you like: use the RRT algorithm as a starting point.)

We expect that:

- You focus on one or two points of discussion for each question.
- You provide reasons in support of your answers (i.e., explain why you chose your answer).
- You are clear and concise – stick to plain, unadorned language.
- You do not do any outside research.
- You demonstrate a thoughtful engagement with the questions.

Solution 4: