Terms

Agent-An entity that perceives and acts. Usually via sensors
Percept-Agent’s perceptual inputs
Percept Sequence-Complete history of everything agent has perceived.
Agent Funcion-Maps a percept sequence to an action. Implemented by an agent program
Performance Measure-Eval’s the behavior of agent in an environment.
Rational Agent-Agent selects an action that’s expected to maximize its performance measure for each percept sequence
Sample/State space-the set of all possible world configurations or states
State space graph - a mathematical representation of the problem, each state is a vertex on a graph

Task Environments

PEAS:Performance, Environment, Actuators, Sensors
Fully Observable, Partially Observable, Unobservable - How much agent observes
Deterministic vs Stochastic - Predetermined vs randomness
Discrete vs. Continuous - Finitely actions vs unlimited action choices
Benign vs Adversarial - No opponent vs opponent or objective that contradicts
Single agent vs. multi agent - ex. Word puzzle vs chess
Episodic vs Sequential - decisions don't affect future vs decisions could affect all future decisions
Static vs Dynamic - environment is constant vs environment can change

←— Ex. Packman Example, solution was (10*12)*(4)*(2^30)*(12^2)
(Grid for packman)(Packman orientation)(Food on or off)(Ghost grid)

Agent Types

Simple Reflex - Selects and action based on present only
Ex. Vacuum only has data on if location is clean or not
Model Based Reflex - Action based on past and present
Ex.Vacuum knows what it cleaned
Goal Based - Action on past, present, and future consideration, works towards pre-specified goal. Hypothesized consequences.
Ex. Vacuum will try and find optimal path to clean with least number of actions
Utility based - Past, present, and future. And quantifies the benefit of various actions. Utility scores to states
Ex. Gameboard with the diamond
Learning agent - Can predict, learns through feedback and examples.
Ex. Spam filters

Bayes Theorem

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)}$$

A, B = events
$P(A|B)$ = probability of A given B is true
$P(B|A)$ = probability of B given A is true
$P(A), P(B)$ = the independent probabilities of A and B

EX.

$P(A|D) = P(D|A)P(A)/P(D)$

Problem 4

$P(A|D) = \frac{P(D|A)P(A)}{P(D)}$

$P(A) = .35$, $P(D|A) = .015$ these are given

$P(D) = \sum_x P(D|x)P(x) = P(D|A)P(A) + P(D|B)P(B) + P(D|C)P(C)$
$= .015 \times .35 + .01 \times .35 + .02 \times .30$
$= .00525 + .0035 + .006$
$= .01475$

$P(A|D) = \frac{.015 \times .35}{.01475} = \frac{.00525}{.01475} = \sim .356$

MDP

Policy-Solution to an MDP pi(s) is action recommended by policy pi in state s.
Optimal Policy - has highest expected utility.
Policy Improvement - Calc new policy pi_i+1 using one step look ahead based on utility values calculates
Passive learning vs Active - Learning how good policy is by doing Upi(s) vs adapting and improving pi
Reinforcement learning - uses exploration, exploitation, and rand

→ Optimal policy in s is then: action

$$\pi^*(s) = \arg\max_{a \in A(s)} \sum_{s'} P(s' \mid s, a)\, U(s')$$

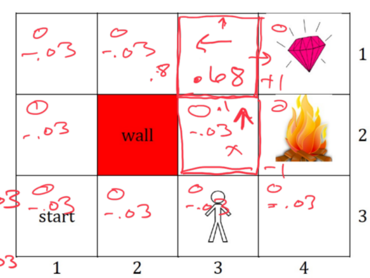
$$U_{i+1}(s) = R(s) + \gamma \max_{a \in A(s)} \sum_{s'} P(s'|s,a)\, U_i(s')$$

Ex.

Example: Find U_0 and U_1 for all states. ϕ

$U_0 = \phi$ for all states
$U_1 = R(s)$ for all states
$U_2(1,3) = -.03 + .9$
→ $.8 \times 1 + .1 \times -.03 + .1 \times -.03$
↑ $.8 \times -.03 + .1 \times -.03 + .1 \times 1$
← $.8 \times -.03 + .1 \times -.03 + .1 \times -.03$
↓ $.8 \times -.03 + .1 \times -.03 + .1 \times +1$

$U_2(1,3) = -.03 + .9 \max[.794, .073, -.03, .073] = .68$



Search
Search Problem Consists of state space, transition model, actions, initial state, goal test, and solution.
EX. Taveling the US northeast from chicago to philadelphia

State Space - cities
Transition Model - Driving between cities
Actions - driving
Initial state - Chicago
Goal test - are we in philadelphia?
Solution - Sequence of cities from chi to phyl

Probability

The **joint probability** of $V = v, C = c$, $p(V = v, C = c)$, is the probability that the card value is v and the card color is c simultaneously.

The **conditional probability** of $V = v$ given $C = c$, $p(V|C)$, is the probability that the card value is v if you know that the card color is c. This is given by

$$p(V|C) = \frac{p(V,C)}{p(C)}$$

Probability Chain Rule: We can use the product rule (twice) to show that for 3 rar variables A, B, and C:

$$p(A,B,C) = p(C|A,B)\,p(A,B) = p(C|A,B)\,p(B|A)\,p(A)$$

Marginal Probability

Law of Total Probability: $p(V) = \sum_c p(V|C = c)p(C = c)$

Joint -> Marginal Distributions

$P(T, W)$

T	W	P
hot	sun	0.4
hot	rain	0.1
cold	sun	0.2
cold	rain	0.3

$$P(t) = \sum_w P(t, w)$$

$P(T)$

T	P
hot	0.5
cold	0.5

$$P(w) = \sum_t P(t, w)$$

$P(W)$

W	P
sun	0.6
rain	0.4

Markov Models

Markov model - a chain-structured Bayesian network

EX.

Example: Is it raining? Let X_t denote the event that it is raining on day t

$$X_1 \to X_2 \to X_3 \to X_4 \to X_5 \to \ldots$$

Recall: Causal chain Bayes net forms conditional independence

- Past and future are independent of the present.
- State at $t + 1$ only depends on state at t
- The CPTs give the transition probabilities from one state to another for this.

Markov property (first-order)

CPT - Can also branch

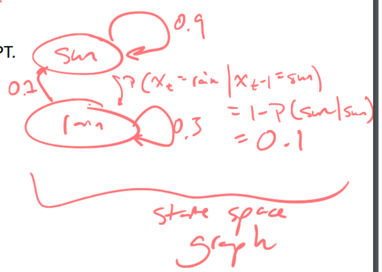
Note:found the rain,sun and sun,rain by doing 1-P(ss or rr)

Example: Suppose we want to forecast the weather. From historical data, we know that in our town, if the current day was sunny, then the following day was also sunny 90% of the time, and that if the current day was rainy, then the following day was also rainy 30% of the time.

Draw the state space graph and specify the CPT.

CPT

t	$t+1$	$P(X_{t+1} \mid X_t)$
sun	sun	0.9 ← Given
sun	rain	0.1
rain	sun	0.2
rain	rain	0.3 ← Given



$P(X_t = \text{rain} \mid X_{t-1} = \text{sun}) = 1 - P(\text{sun}|\text{sun}) = 0.1$

state space graph

Q-Learning - Bruh idk

Temporal Difference Q-learning:

discount factor

$$Q_{i+1}(s,a) = Q_i(s,a) + \alpha\,[R(s) + \gamma \max_{a'} Q_i(s',a') - Q_i(s,a)]$$

HMM & Viterbi

Day	Person 1	Feeling	Person 2	Feeling	Person 3	Feeling
Day 1	Healthy	Good	Healthy	Good	Healthy	Good
---	---	---	---	---	---	---
Day 2	Cold	Tired	Healthy	Good	Healthy	Good
---	---	---	---	---	---	---
Day 3	Cold	Tired	Healthy	Tired	Cold	Good
---	---	---	---	---	---	---
Day 4	Cold	Tired	Healthy	Tired	Cold	Tired
---	---	---	---	---	---	---
Day 5 (current)	Healthy	Tired	Healthy	Good	Cold	Tired

Solution——> Given Health = 1 Cold = 0

Filtering - Compute the belief state given the evidence observed so far

Prediction - Compute the distribution over the future state given all evidence up to current state

Smoothing - Computer belief state given the evidence over a previous state using all evidence

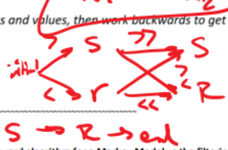
Hidden Markov Models - Viterbi:

- Viterbi algorithm: Given a sequence of observations, find the most likely state sequence that generated those observations.
  - *Generate trellis of state transitions and values, then work backwards to get path*
  - *$\max(P(e_t|X_t)P(X_t|X_{t-1})P(X_{t-1}))$*
  - *$\text{argmax}_{x_{1:t}} P(X_{1:t} \mid e_{1:t})$*

key strokes

x_1 x_2

p[ ][pred]



max[p[i]]

↳ p[i][pred]

initial → S → R → end

Practice Questions:

1. What is the difference between the mini-forward algorithm for a Markov Model vs the filtering step of an HMM?

Generate trellis of state transitions and values

max(P(e_t|X_t)P(X_t|X_t-1)P(X_t-1)

Argmax_x1:t P(X_1:t|e_1:t)

Transitions:

From	To	Prob
Healthy	Healthy	5/7
Healthy	Cold	2/7
Cold	Healthy	1/5
Cold	Cold	4/5

Emissions:

Evidence	State	Prob
Good	Healthy	6/9 = 2/3
Tired	Healthy	3/9 = 1/3
Good	Cold	1/6
Tired	Cold	5/6

Day 1

$$P(x_1) = \sum_{x_0} P(x_1 | x_0) P(x_0)$$
$$= [\tfrac{5}{7} \cdot 1 + \tfrac{1}{5} \cdot 0,\ \tfrac{4}{5} \cdot 0 + \tfrac{2}{7} \cdot 1]$$
$$= [\tfrac{5}{7}, \tfrac{2}{7}]$$
$$P(x_1 | \text{good}) = \alpha P(\text{good} | x_1) P(x_1)$$
$$= \alpha[\tfrac{2}{3} \cdot \tfrac{5}{7},\ \tfrac{1}{6} \cdot \tfrac{2}{7}]$$
$$= \alpha[\underset{H}{0.47}, \underset{C}{0.048}]$$

$$\alpha = \frac{1}{0.47 + 0.048} = 0.5176$$

$$\frac{0.47}{0.5176}, \frac{0.048}{0.5176}$$

$$= [\underset{H}{0.90}, \underset{C}{0.09}]$$

Day 2

Day 2

$$P(x_2) = \sum_{x_1} P(x_2 | x_1) P(x_1)$$
$$= [\tfrac{5}{7} \cdot 0.90 + \tfrac{1}{5} \cdot 0.09,\ \tfrac{4}{5} \cdot 0.09 + \tfrac{2}{7} \cdot 0.90]$$
$$= [0.66, 0.324]$$

$$P(x_2 | \text{tired}) = \alpha P(\text{tired} | x_2) P(x_2)$$
$$= \alpha[\tfrac{1}{3} \cdot 0.66, \tfrac{5}{6} \cdot 0.324]$$
$$= \alpha[0.2178, 0.2744]$$
$$\alpha = \frac{1}{0.2178 + 0.2744} = 1/.4918$$
$$= \frac{0.2178}{0.4918}, \frac{0.274}{0.4918}$$
$$= [0.44, 0.55]$$

Day 3

Day 3

$$P(x_3) = \sum_{x_2} P(x_3 | x_2) P(x_2)$$
$$= [\tfrac{5}{7} \cdot 0.44 + \tfrac{1}{5} \cdot 0.55,\ \tfrac{4}{5} \cdot 0.55 + \tfrac{2}{7} \cdot 0.44]$$
$$= [0.424, 0.565]$$
$$P(x_3 | \text{tired}) = \alpha P(\text{tired} | x_3) P(x_3)$$
$$= \alpha[\tfrac{1}{3} \cdot 0.424, \tfrac{5}{6} \cdot 0.565]$$
$$= \alpha[0.139, 0.47]$$
$$\alpha = \frac{1}{0.139 + 0.47} = 1/0.609$$
$$= \frac{0.139}{0.609}, \frac{0.47}{0.609}$$
$$= [\underset{H}{0.23}, \underset{C}{0.77}]$$

FINAL