Review

* Bellman Optimality Equation

$$V^*(s) = R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V^*(s')$$

* Value Iteration

$$V_{k+1}(s) \leftarrow R(s) + \gamma \max_a \sum_{s'} P(s'|s,a) V_k(s')$$

$$\lim_{k\to\infty} \left[V_k(s)\right] \rightarrow V^*(s) \text{ for all states } s.$$

Reinforcement Learning

* What if $P(s'|s,a)$ and $R(s)$ are not known?
  Can we learn $\pi^*(s)$ or $V^*(s)$ from experience?

  Experience: $\begin{matrix} s_0 \\ r_0 \end{matrix} \xrightarrow{a_0} \begin{matrix} s_1 \\ r_1 \end{matrix} \xrightarrow{a_1} \begin{matrix} s_2 \\ r_2 \end{matrix} \rightarrow \ldots$

* Model-based approach
  - Explore world
    Estimate model $P_{ML}(s'|s,a)$ from experience.
    Hope that $P(s'|s,a) \approx P_{ML}(s'|s,a)$
    as agent gains more experience
    Compute $\hat{\pi}^*$ from $P_{ML}(s'|s,a)$.

* Disadvantage
  To store $P_{ML}(s'|s,a)$ is $O(n^2)$ for n states.
  Only care about $\pi^*(s)$ or $V^*(s)$ which are $O(n)$.
  Is it really necessary to estimate a model?

* Advantage

Model $P(s'|s,a)$ is useful for "task transfer," where rewards $R(s)$ or discount factor γ change but dynamics stay the same. Ex: robot navigation to different goal states.

Beyond CSE 150:

*Extension #1: Temporal difference methods

How to estimate $V^\pi(s)$ directly from experience?

$$
\begin{array}{llllllll}
\text{action} & & \pi(s_0) & & \pi(s_1) & & & \\
\text{state} & s_0 & \longrightarrow & s_1 & \longrightarrow & s_2 & \longrightarrow & \cdots \\
\text{reward} & r_0 & & r_1 & & r_2 & &
\end{array}
$$

Let $V_t(s)$ denote estimate at time t.

Initialize $V_0(s) = 0$ for all states s.

Temporal Difference Prediction:

$$V_{t+1}(s) = V_t(s) + \alpha\left[R(s_t) + \gamma V_t(s_{t+1}) - V_t(s_t)\right]$$

small learning rate $\alpha > 0$

estimate of $V^\pi(s)$ at time t is states s

Thm: $\lim_{t\to\infty} V_t(s) \rightarrow V^\pi(s)$ under certain conditions

*Extension #2: Large state space

* so far: implicit assumption that we can store $V^\pi(s)$ or $\pi(s)$ as lookup table.

* function approximation in RL
- storing $V^\pi(s)$ is impossible for backgammon (10^{50} states)
- parameterize $V^\pi(s, \vec{\theta})$ and estimate this function

* Extension #3: MDPs with undiscounted rewards

· suppose goal is to maximize (or evaluate)

$$\rho^\pi = \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} R(s_t)$$

Assume that ρ^π does not depend on initial states s.

Certain states have better trancients than others:

$$\tilde{V}^\pi(s) = E^\pi\left[\sum_{t=1}^{\infty}\left[R(s_t)-\rho^\pi\right] \,\middle|\, s_0=s\right]$$

$$\tilde{Q}^\pi(s,a) = E^\pi\left[\sum_{t=0}^{\infty}\left[R(s_t)-\rho^\pi\right] \,\middle|\, s_0=s,\ a_0=a\right]$$

* Extension #4: partially observable MDP's (POMDP's)

· POMDP's are to MDP's as HMM's are to Markov Models.

Ex: robot navigation
states: xy location
observations: sensors

· Model for POMDP's

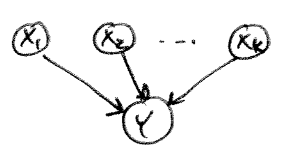
Transitions $P(s_{t+1}|s_t,a_t)$
Rewards $R(s_t)$
Observations $P(o_t|s_t)$

Experience:
Agent sees $o_1, o_2, \ldots, o_T$
not $s_1, s_2, \ldots, s_T$
Much harder than MDP.

---

* Compact representations of complex worlds;
balance power / expressiveness vs tractability

1) Noisy-OR CPT

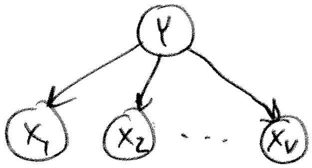


$X_i \in \{0,1\}$

$Y \in \{0,1\}$

$$P(Y=1|X_1,X_2,\ldots,X_k) = 1-\prod_{i=1}^{k}(1-p_i)^{X_i}$$

2) Naive Bayes model for document classification

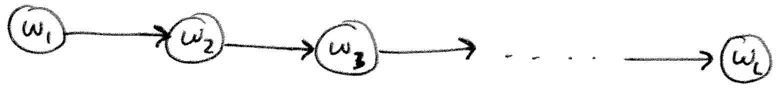


$Y \in \{1, \ldots, m\}$ topic

$X_i \in \{0, 1\}$ does i^{th} word in vocabulary appear in document?

3) Markov models of language

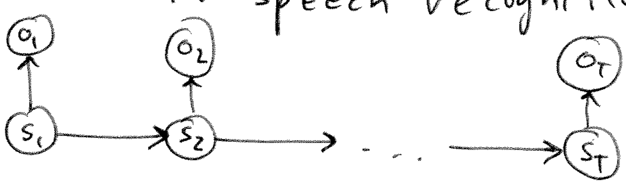
$W_\ell \in \{1, 2, \ldots, V\}$ ℓ^{th} word in sentence



bigram model

trigram model

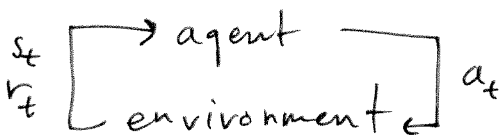
4) HMMs for speech recognition



acoustic observations of speech waveform

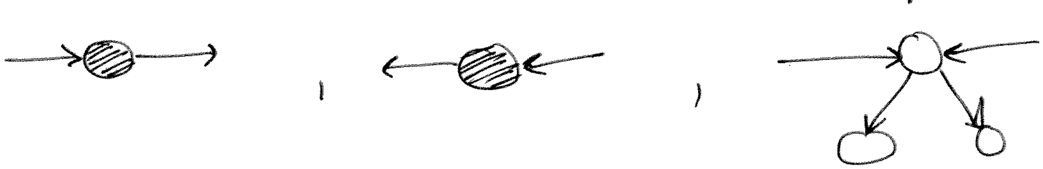
linguistic units

5) MDPs for planning

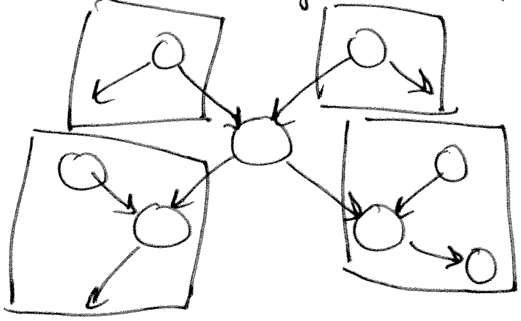


* Efficient Algorithms

1) conditional independence tests via d-separation



2) Polytree algorithm for inference



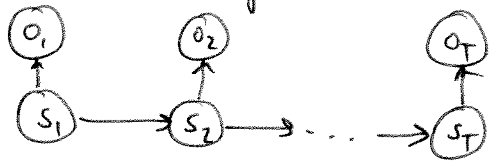
3) EM algorithm for ML estimation in hidden variable models

update: $P(X_i = x \mid pa_i = \pi) = \dfrac{\sum_t P(X_i = x, pa_i = \pi \mid V^{(t)})}{\sum_t P(pa_i = \pi \mid V^{(t)})}$

guarantee: monotonic convergence

$$\mathcal{L} = \sum_t \log\left(P(V^{(t)})\right)$$

4) Viterbi algorithm in HMMs



$$\underset{s_1, s_2, \ldots, s_T}{\operatorname{argmax}} P(s_1, s_2, \ldots, s_T \mid o_1, o_2, \ldots, o_T)$$

complexity $O(n^2 T)$ ← n = # hidden states, T = sequence length

Also in HMMs: forward & backward algorithms

5) Algorithms in MDPs

Policy Iteration $\quad \pi_0 \xrightarrow{\text{evaluate}} V^{\pi_0}(s),\ Q^{\pi_0}(s,a) \xrightarrow{\text{improve}} \pi_1 \longrightarrow$

Value Iteration $\quad V_{k+1}(s) \leftarrow R(s) + \gamma \max_a \sum_{s'} P(s' \mid s, a) V_k(s')$

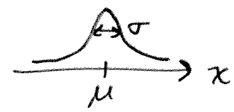
Simple algorithms, strong guarantees.

$\boxed{\text{Final Exam: Monday 3-6 pm}}$

Things we didn't cover

1) Continuous random variables

Ex: one dimensional gaussian



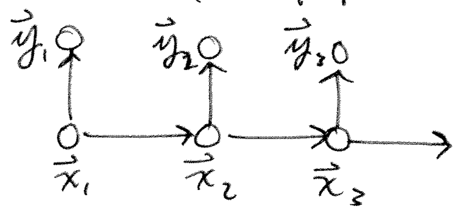
$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-(x-\mu)^2/2\sigma^2}$$

μ ← mean

multidimensional gaussian

"d" ↗ $$P(\vec{x}) = \frac{1}{(2\pi)^{d/2}|\Sigma|^{1/2}} e^{-(\vec{x}-\vec{\mu})^T \Sigma^{-1} (\vec{x}-\vec{\mu})}$$

Σ^{-1} ← inverse covariance matrix



observations $\vec{y}_t \in \mathbb{R}^m$

states $\vec{x}_t \in \mathbb{R}^n$

Ex: tracking missile from radar observations

2) Bayesian learning

In this course: ML estimation

choose parameters $\vec{\theta}$ to maximize $\log[P(\text{data}|\vec{\theta})]$

Problems: overfitting to small sample sizes

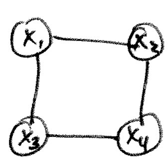
Ex: bias of coin from just 1 toss

Alternate Solution:

- choose prior distribution $P(\vec{\theta})$
- compute posterior distribution $P(\vec{\theta}|\text{data})$

3) Undirected graphical models

In this course: DAGs! Limitation: not all random variables have a natural ordering. Ex: pixels in an image



conditional independence relations, neighborhoods, Markov blankets, have different semantics.