CSCE 633: Machine Learning

Lecture 31: Reinforcement Learning

Texas A&M University

11-4-19

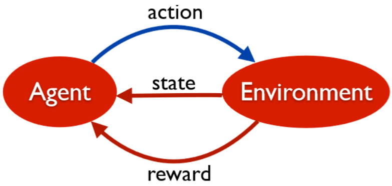
Goals of this lecture

- Reinforcement Learning
- Source: Mohri text and slides

Reinforcement Learning

- We have an Actions and Environment that do not passively collected labeled data
- The learner, called an Agent, gets two kinds of information to learn from: The current state of the environment, and a real-valued reward
- The objective of the learning problem is for the agent to maximize its reward
- It does this through finding the best course of actions - called a policy

Reinforcement Learning



- Exploration - search unknown states and actions to gain reward information
- Exploitation - search known states to optimize reward

Reinforcement Learning vs. Supervised Learning

- No fixed distribution that instances are drawn from
- Environment may not be fixed!
- Training and testing phases are mixed.
- Planning Problem: When the environment model is known - objective is to maximize reward
- Learning Problem: Environment model is unknown
- We will explore both

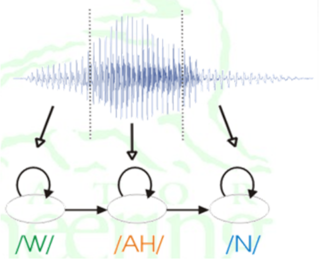
Applications

- Robot control e.g., Robocup Soccer Teams (Stone et al., 1999).
- Board games, e.g., TD-Gammon (Tesauro, 1995).
- Elevator scheduling (Crites and Barto, 1996).
- Ads placement.
- Telecommunications.
- Inventory management.
- Dynamic radio channel assignment.

Motivation

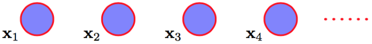
Why to model time-series?

- Many phenomena depict inherent dependencies between successive time points
- Examples
  - speech, DNA sequencing, industrial processes, human behavior
- Data samples are no longer considered iid

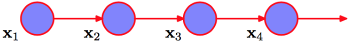


Discrete (Observable) Markov Model

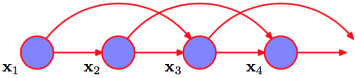
The simplest approach to modelling a sequence of observations is to treat them as independent, corresponding to a graph without links.



A first-order Markov chain of observations $\{\mathbf{x}_n\}$ in which the distribution $p(\mathbf{x}_n|\mathbf{x}_{n-1})$ of a particular observation $\mathbf{x}_n$ is conditioned on the value of the previous observation $\mathbf{x}_{n-1}$.



A second-order Markov chain, in which the conditional distribution of a particular observation $\mathbf{x}_n$ depends on the values of the two previous observations $\mathbf{x}_{n-1}$ and $\mathbf{x}_{n-2}$.



For the next slides, $q_t = x_t$ is the i^{th} observable sample of the sequence $Q = \{q_1, q_2, \ldots, q_T\}$

where $q_t \in \{S_1, \ldots, S_N\}$

Discrete (Observable) Markov Model

- N distinct states: $\{S_1, S_2, \ldots, S_N\}$
- Observable sequence: $Q = \{q_1, q_2, \ldots, q_T\}$
- $q_t = S_i$: at time t the system is at state S_i
- General Markov model: future state depends on current & previous

$$P(q_{t+1} = S_j | q_t = S_i, q_{t-1} = S_k, \ldots)$$

- 1st-order Markov model: future state depends only on current

$$P(q_{t+1} = S_j | q_t = S_i)$$

The future is independent of the past

Discrete (Observable) Markov Model

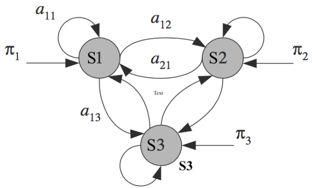
- Constant transition probability

$$\alpha_{ij} = P(q_{t+1} = S_j | q_t = S_i) \ , \quad \alpha_{ij} \geq 0 \ , \quad \sum_{j=1}^{N} \alpha_{ij} = 1$$

  Going from S_i to S_j has the same probability no matter when it happens

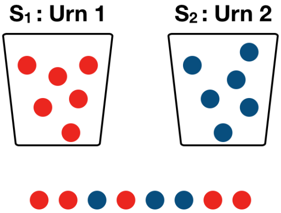
- Initial probability

$$\pi_i = P(q_1 = S_i) \ , \quad \sum_{j=1}^{N} \pi_i = 1$$



Observable Markov model

- The states are *observable*
    - We know q_t at any time t
- Transition matrix $\mathbf{A} = [\alpha_{ij}]$
- Initial probability vector $\boldsymbol{\pi} = [\pi_1, \ldots, \pi_N]$



$Q = \{q_1, q_2, \ldots, q_8\}$

Discrete (Observable) Markov Model

Observable Markov model: Two Basic Problems

1. Given a model $\lambda = \{\mathbf{A}, \boldsymbol{\pi}\}$, we would like to evaluate the probability of a given observation sequence $Q = \{q_1, \ldots, q_T\}$: $P(\mathbf{Q}|\mathbf{A}, \boldsymbol{\pi})$
2. Given a training set of observation sequencies, $\mathcal{X} = \{Q^k\}_{k=1}^{K}$, we would like to learn the model that maximizes the probability of generating $\mathcal{X}$: $P(\mathcal{X}|\mathbf{A}, \boldsymbol{\pi})$

Observable Markov model: Two Basic Problems

Problem 1

Given a model $\lambda = \{\mathbf{A}, \boldsymbol{\pi}\}$, we would like to evaluate the probability of a given observation sequence $Q = \{q_1, \dots, q_T\}$: $P(\mathbf{Q}|\mathbf{A}, \boldsymbol{\pi})$

$$P(\mathbf{Q}|\mathbf{A}, \boldsymbol{\pi}) = P(q_1) \prod_{t=2}^{T} P(q_t|q_{t-1}) = \pi_{q_1} \alpha_{q_1 q_2} \dots \alpha_{q_{T-1} q_T}$$

Observable Markov model: Two Basic Problems

Problem 2

Given a training set of observation sequencies, $\mathcal{X} = \{Q^k\}_{k=1}^K$, we would like to learn the model that maximizes the probability of generating $\mathcal{X}$: $P(\mathcal{X}|\mathbf{A}, \boldsymbol{\pi})$

$$\hat{\pi}_i = \frac{\#\text{sequences starting with } S_i}{\#\text{sequences}} = \frac{\sum_{k=1}^K \mathbb{I}(q_1^k = S_i)}{K}$$

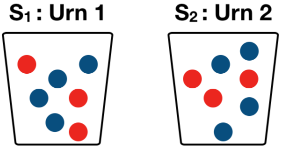
$$\hat{\alpha}_{ij} = \frac{\#\text{transitions from } S_i \text{ to } S_j}{\#\text{transitions from } S_i} = \frac{\sum_{k=1}^K \sum_{t=1}^{T-1} \mathbb{I}(q_t^K = S_i \text{ and } q_{t+1}^k = S_j)}{\sum_{k=1}^K \sum_{t=1}^{T-1} \mathbb{I}(q_t^k = S_i)}$$

Observable Markov model

- The states are *not* observable
- But when we reach a state, an observation occurs with *emission probability*

$$b_j(m) = P(O_t = v_m | q_t = S_j)$$

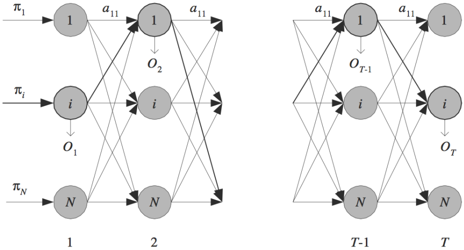
  $b_j(m)$: is the probability of observing value v_m in state S_j
- Example: each urn contains balls of different colors
  - $b_j(m)$ is the probability of drawing a ball of color m from urn j



Hidden Markov Model

Observable Markov model

- Hidden states: $\{S_1, \ldots, S_N\}$
- Observation symbols: $\{v_1, \ldots, v_M\}$
- State transition probabilities: $\mathbf{A} = [a_{ij}]$, $a_{ij} = P(q_{t+1} = S_j | q_t = S_i)$
- Observation probs: $\mathbf{B} = [b_j(m)]$, $b_j(m) = P(O_t = v_m | q_t = S_j)$
- Initial state probabilities: $\boldsymbol{\pi} = [\pi_i]$, $\pi_i = P(q_1 = S_i)$



Hidden Markov model: Three Basic Problems

1. Given a model $\lambda = \{\alpha_{ij}, \pi_i, b_j(m)\}$, we would like to evaluate the probability of a given observation sequence $O = \{O_1, \ldots, O_T\}$: $P(\mathbf{O}|\lambda)$
2. Given a model λ and observation sequence O, we would like to find out the state sequence $Q = \{q_1, \ldots, q_T\}$, that generates O with the highest probability: $Q^* = \max_Q P(Q|O, \lambda)$
3. Given a training set of observation sequencies, $\mathcal{X} = \{O^k\}_{k=1}^K$, we would like to learn the model that maximizes the probability of generating $\mathcal{X}$: $\lambda^* = P(\mathcal{X}|\lambda)$

Hidden Markov model: Problem 1

Given a model $\lambda = \{\alpha_{ij}, \pi_i, b_j(m)\}$, we would like to evaluate the probability of a given observation sequence $O = \{O_1, \ldots, O_T\}$: $P(\mathbf{O}|\lambda)$

- The probability of the state sequence is

$$P(Q|\lambda) = P(q_1)\prod_{t=2}^{T} P(q_t|q_{t-1}) = \pi_{q_1}\alpha_{q_1 q_2}\ldots\alpha_{q_{T-1}q_T}$$

- The joint probability is

$$P(O, Q|\lambda) = \pi_{q_1} b_{q_1}(O_1)\alpha_{q_1 q_2} b_{q_2}(O_2)\ldots\alpha_{q_{T-1}q_T} b_{q_T}(O_T)$$

- By marginalizing the joint

$$P(O|\lambda) = \sum_{Q} P(O, Q|\lambda)$$

  which is not practical sine there are N^T possible sequences Q
  $\rightarrow$ forward-backward procedure

Hidden Markov model: Problem 1

Forward Procedure

- We want to estimate

$$P(O|\lambda) = \sum_{Q} P(O, Q|\lambda)$$

- We use a temporary variable $a_t(i)$, called forward variable

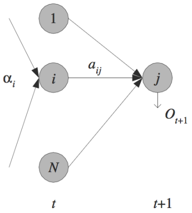
$$a_t(i) = P(O_1 \dots O_t, q_t = S_i|\lambda)$$

which is the probability of observing $\{O_1, \dots, O_t\}$ until time t and being in S_i at time t, given the model λ

Hidden Markov model: Problem 1

Forward Procedure

1. Initialize $a_1(i) = \pi_i b_i(O_1)$
2. Recursion
$$a_{t+1}(j) = \left[\sum_{i=1}^{N} a_t(i)\alpha_{ij}\right] b_j(O_{t+1})$$
3. Calculate probability of observation O (sum over all possible states)
$P(O|\lambda) = \sum_{i=1}^{N} a_T(i)$

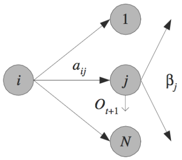


Backward Procedure

$\beta_t(i) = P(O_{t+1}, \ldots, O_T | q_t = S_i, \lambda)$: probability of being in S_i at time t and observing $\{O_{t+1}, \ldots, O_T\}$

1. Initialize $\beta_T(i) = 1$
2. Recursion

$$\beta_t(i) = \sum_{j=1}^{N} \alpha_{ij} b_j(O_{t+1}) \beta_{t+1}(j)$$



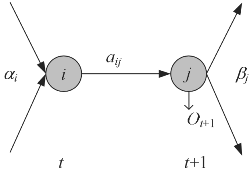
Hidden Markov model: Problem 2

Given a model λ and observation sequence O, we would like to find out the state sequence $Q = \{q_1, \ldots, q_T\}$, that generates O with the highest probability: $Q^* = \max_Q P(Q|O, \lambda)$

- Define the temporary variable $\gamma_t(i)$, as the probability of being in state S_i at time t, given O and λ

$$\gamma_t(i) = \frac{a_t(i)\beta_t(i)}{\sum_{j=1}^{N} a_t(j)\beta_t(j)}$$

- To find the sequence at time step t: $q_t^* = arg\max_i \gamma_t(i)$, BUT there is no transition info $\alpha_{ij} \rightarrow$ Viterbi algorithm



Hidden Markov model: Problem 2

Viterbi Algorithm

$\delta_t(i) = \max_{q_1 \ldots q_{t-1}} P(q_1 \ldots q_{t-1}, q_t = S_i, O_1, \ldots, O_t|\lambda)$: prob of most likely path at time t, after taking into account $\{q_1, \ldots, q_t\}$ and ending in S_j

1 Initialize $\delta_1(i) = \pi_i b_i(O_1)$

2 Recursion

$$\delta_t(j) = \max_i \delta_{t-1}(i)\alpha_{ij} b_j(O_t)\ , \ \ \psi_t(j) = arg\max_i \delta_{t-1}(i)\alpha_{ij}$$

3 Terminate

$$p^* = \max \delta_T(i)\ , \ \ q_T^* = arg\max_i \delta_T(i)$$

4 Path backtracking

$$q_T^* = \psi_{t+1}(q_{t+1}^*)\ , \ \ t = T-1, T-2, \ldots, 1$$

Hidden Markov model: Problem 3

Given a training set of observation sequencies, $\mathcal{X} = \{O^k\}_{k=1}^K$, we would like to learn the model that maximizes the probability of generating $\mathcal{X}$: $\lambda^* = P(\mathcal{X}|\lambda)$

- Find $\lambda = \{\alpha_{ij}, , \pi_i, b_j(m)\}$ that maximizes the likelihood
- The probability of being in S_i at time t and S_j at time $t+1$ is

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda) = \frac{a_t(i)\alpha_{ij} b_j(O_{t+1})\beta_{t+1}(j)}{\sum_k \sum_l a_t(k)\alpha_{kl} b_l(O_{t+1})\beta_{t+1}(l)}$$

- The probability of being in S_i at time t is $\gamma_t(i) = \sum_j \xi_t(i,j)$
- Baum-Welch algorithm (type of EM)

Hidden Markov model: Problem 3

Baum-Welch: E-step

Compute $\xi_t(i,j)$ and $\gamma_t(i)$ given λ

$$\xi_t(i,j) = \frac{a_t(i)\alpha_{ij} b_j(O_{t+1})\beta_{t+1}(j)}{\sum_k \sum_l a_t(k)\alpha_{kl} b_l(O_{t+1})\beta_{t+1}(l)}$$

$$\gamma_t(i) = \sum_j \xi_t(i,j)$$

Hidden Markov model: Problem 3

Baum-Welch: M-step

Compute λ given $\xi_t(i,j)$ and $\gamma_t(i)$

$$\hat{\alpha}_{ij} = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_K}\xi_t(i,j)}{\sum_{k=1}^{K}\sum_{t=1}^{T_K}\gamma_t(j)}$$

$$\hat{b}_j(m) = \frac{\sum_{k=1}^{K}\sum_{t=1}^{T_K}\gamma_t(j)\mathbb{I}(O_t = v_m)}{\sum_{k=1}^{K}\sum_{t=1}^{T_K}\gamma_t(j)}$$

$$\hat{\pi}_i = \frac{\sum_{k=1}^{K}\gamma_1^K(i)}{K}$$

Reinforcement Learning: Markov Decision Process

- Set of epochs $\{0, \cdots, T\}$
- a set of states S, possibly infinite!
- an initial state $s_0 \in S$
- Actions A, also possibly infinite
- Transition Probability $P(s'|s, a)$ which is the distribution over destination states $s' = \delta(s, a)$
- Reward Probability $P(r'|s, a)$ which is the distribution over rewards returned $r' = r(s, a)$

Takeaways and Next Time

- Next Time: More Reinforcement Learning