Dynamic Programming and Optimal Control HS18
Ignacio Castañeda

Probabilities

Expectation

$\mathbb{E}[X] = \int_\Omega xf(x)dx = \int_\omega xP[X=x]dx$
$\mathbb{E}_{Y|X}[Y] = \mathbb{E}_Y[Y|X]$
$\mathbb{E}_{X,Y}[f(X,Y)] = \mathbb{E}_X \mathbb{E}_{Y|X}[f(X,Y)|X]$
$\mathbb{E}_{Y|X}[f(X,Y)|X] = \int_{\mathbb{R}} f(X,y)p_{Y|X}(y)dy$

Variance & Covariance

$Var(X) = \mathbb{E}[(X-\mathbb{E}[X])^2] = \mathbb{E}[X^2]-\mathbb{E}[X]^2$
$\text{Var}[X+Y] = \text{Var}[X] + \text{Var}[Y] \quad X Y iid$
$\text{Var}[\alpha X] = \alpha^2 \text{Var}[X]$
$Cov(X,Y) = \mathbb{E}[(X-\mathbb{E}[X])(Y-\mathbb{E}[Y])]$

Conditional Probabilities

$P[X|Y] = \frac{P[X,Y]}{P[Y]}$, $P[\overline{X}|Y] = 1 - P[X|Y]$

Distributions

$\mathcal{N}(x|\mu,\sigma^2) = 1/(\sqrt{2\pi\sigma^2})\exp^{-(x-\mu)^2/(2\sigma^2)}$
$\mathcal{N}(x|\mu,\Sigma) = \frac{1}{(2\pi)^{2D/}|\Sigma|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu)^T\Sigma^{-1}(\mathbf{x}-\mu)}$
$\text{Exp}(x|\lambda) = \lambda e^{-\lambda x}$
$\text{Ber}(x|\theta) = \theta^x(1-\theta)^{(1-x)}$
Sigmoid: $\sigma(x) = 1/(1+\exp(-x)))$

Problem Statement

Dynamics

$x_{k+1} = f_k(x_k,u_k,w_k),\ k = 0,\dots,N-1$

Cost Function

$C(x) = g_N(x_N) + \sum_{k=0}^{N-1} g_k(x_k,u_k,w_k)$

Expected Cost

$\mathbb{E}_{X_1,U_0,W_0|x_o}[C(X)]$

Open/Closed Loop Control

Open Loop Control

All control inputs given x_0
N_u^N different strategies

Closed Loop Control

Optimal policy $\pi^* = (\mu_0,\dots,\mu_{N-1})$ given x_0.
Closed Loop Expected Cost: $J_\pi(x)$
$N_u^{N_x(N-1)+1}$ different strategies

Dynamic Programing Algorithm

Initialization

$J_N(x) = g_N(x),\ \forall x \in S_N$

Recursion

$J_k(x) = min_{u\in U_k(x)} \mathbb{E}_{w_k|x,u}[g_k + J_{k+1}]$
$\forall x \in S_k,\ k = N-1,\dots,0$

Convert To Standart Form

Time Lags

$x_{k+1} = f_k(x_k,x_{k-1},u_k,u_{k-1},w_k)$
$y_k = x_{k-1},\ s_k = u_{k-1},\ \widetilde{x} = (x_k,y_k,s_k)$

$$\widetilde{x}_{k+1} = \begin{bmatrix} f_k(x_k,y_k,u_k,s_k,w_k) \\ x_k \\ u_k \end{bmatrix} = \widetilde{f}_k(\widetilde{x}_k,u_k,w_k)$$

Correlated Disturbances

$w_k = C_k y_{k+1},\ y_{k+1} = A_k y_k + \xi_k$
A_k, C_k given, ξ_k i.r.v.
$\widetilde{x}_k = (x_k,y_k)$

$$\widetilde{x}_{k+1} = \begin{bmatrix} f_k(x_k,u_k,C_k(A_k y_k + \xi_k)) \\ A_k y_k + \xi_k \end{bmatrix} = \widetilde{f}_k(\widetilde{x}_k,u_k,\xi_k)$$

Forecasts

TODO

Infinite Horizon Problems

$J_N(x) = 0,\ \forall x \in S$
$l = N-k,\ V_l = J_{N-l}$
$\lim_{N\to\infty} V_l(x) = J(x)$

Bellman Equation

$J(x) = min_u \mathbb{E}_{w|x,u}[g(x,u,w) + J(f(x,u,w))]$
$\forall x \in S$

Stochastic Shortest Path

Time invariant transition probabilities and there is a cost-free termination state. BE yields optimal cost-to-go and optimal stationary policy. Unique solution.
$J^*(i) = min_u(q(i,u) + \sum_{j=1}^n P_{ij}(u)J^*(j))\ \forall i \in S$

Value Iteration

Arbitrary initialization until it converges.
$V_{l+1}(i) = min-u(q(i,u) + \sum_{j=1}^n P_{ij}(u)V_l(j))\ \forall i \in S$
Stop when $\|V_{l+1}(i) - V_l(i)\|,\ \forall i \in S$
Complexity $(O)(n^2p)$ per iteration
Requires (generally) inifnite iterations.

Policy Iteration

Initialize with a proper policy μ^0
Stage 1: Solve following linear system for μ^h
$J_{\mu^h}(i) = q(i,\mu^h(i)) + \sum_{j=1}^n P_{ij}(\mu^h(i))J_{\mu^h}(j))\ \forall i \in S$
Stage 2: Obtain new policy μ^{h+1} satisfying
$\mu^{h+1}(i) = argmin_u(q(i,u) + \sum_{j=1}^n P_{ij}(u)J_{\mu^h}(j))$
$\forall i \in S$
Iterate between 1 and 2 until
$J_{\mu^{h+1}}(i) = J_{\mu^h}(i)\ \forall i \in S$
Complexity $(O)(n^2(n+p))$ per iteration
Worst case: p^n iterations

Linear Programming

maximize $\sum_{i\in S_+} V(i)$ subject to
$V(i) \le q(i,u) + \sum_{j=1}^n P_{ij}(u)V(j)\ \forall u \in U(i)\ \forall i \in S$

Discounted Problems

Stage costs are discounted exponentially. Equivalent to solving a SSP problema with a virtual termination state (0)
$P_{ij}(u) = \alpha\widetilde{P}_{ij},\ u \in U$
$P_{i0}(u) = 1-\alpha,\ u \in U$
$P_{0j}(u) = 0,\ u = stay$
$P_{00}(u) = 1,\ u = stay$
$g(x,u,w) = \alpha^{-1}\widetilde{g}(x,u,w)\ g(x,u,0) = 0$
$g(0,stay,0) = 0$

Shortest Path Problem

Path: $Q = (i_i,\dots,i_q)$
Length: $J_Q = \sum_{h=1}^{q-1} c_{i_h,i_{h+1}}$
Optimal Path: $Q^* = \arg\min_{Q\in\mathbb{Q}_{S,T}} J_Q$
There cannot be a negative cycle in the graph

Deterministic Finite State Problema (DFS)

Equivalent to SP
$N = |V|$
$S_0 = s,\ S_N = T,\ S_k = V$
$x_{k+1} = u_k$
$g_N(T) = 0,\ g_k(x_k,u_k) = c_{x_k,u_k}$
Solve with DPA, $J_k(i)$ is the optimal cost of getting from i to T in $N-k$ moves.
$J_N(T) = 0$
$J_{N-1}(i) = c_{iT}$
$J_k(i) = \min_j(c_{ij} + J_{k+1}(j))$

Hidden Markov Models

Can be converted in a SP problem. We use artificial nodes s and T and we use the negative log likelihood as the weights.

Label Correcting Algorithm

1. Place node s in OPEN, set $d_s = 0$ and $d_j = \infty\ \forall j$

2. Remove a node i from OPEN and execute step 3 for all children of j of i (nodes you can go from i)

3. If $d_i + c_{i,j} < d_j$ and $d_i + c_{i,j} < d_T$, set $d_j = d_i + c_{i,j}$ and set i to be parent of j. If $j \neq T$, put j in OPEN

4. If OPEN is not empty, go to step 2

Removing nodes from OPEN:
Depth-First Search: LIFO (Stack)
Breadth-First Search: FIFO (Queue)
Best-First Search: Node with best label (Dijkstra)

A*-algorithm

Modification of LCA. In step 3, for node j to be admited in OPEN, it must fulfill
$d_i + c_{i,j} + h_j < d_T$ where h_j is a lower bound constructed from problem knowledge.

Continuous Time Deterministic Optimal Control

$\dot{x}(t) = f(x(t),u(t)),\ 0 \le t \le T$
$J_\mu(t,x) = h(x(T)) + \int_t^T g(x(\tau),u(\tau))d\tau$

Hamilton-Jacobi-Bellman Equation

$0 = \min_u$