Redes Bayesianas 1

Matheus Cafalchio
Paulo Moreira

Teoria da Probabilidade

- Associa às sentenças um grau de crença numérico entre 0 e 1
  - Contudo, cada sentença ou é **verdadeira** ou é **falsa.** (O grau de crença é associado ao meu conhecimento sobre a sentença, e não sobre a sentença em si).
- Grau de crença(probabilidade):
  - a priori(incondicional): calculado antes do agente receber percepções
    - Ex. P(cárie) = 0.5
  - condicional: calculado de acordo com as **evidências** disponíveis
    - **evidências**: percepções que o agente recebeu até agora
    - Ex: P(cárie|dor de dente)= 0.8
      P(cárie|~dor de dente)= 0.3

Probabilidade condicional

- Probabilidade condicional (a posteriori) de A dado que B ocorreu é definida por:
  - $P(A|B) = \frac{P(A \wedge B)}{P(B)}$, quando P(B) > 0. **Regra do Produto**

  ou P(A ^ B) = P(A|B) * P(B)

- Possibilita inferência sobre uma proposição desconhecida A dada a evidência B

Independência

- Independência **absoluta**:
  - P(A|B) = P(A)
  - Exemplo: A = dor de dente e B=úlcera
    - Úlcera não causa dor de dente
- Independência **condicional:**
  - Seja X e Y condicionalmente independentes dado Z => P(X|Y,Z) = P(X|Z) ou P(X,Y|Z) = P(X|Z)
  - Independência condicional é crucial para o funcionamento eficaz de sistemas probabilísticos.

Regra de Bayes

- **Thomas Bayes** (1702 - 1761) **:** matemático inglês e pastor presbiteriano.
- Seu teorema pode ser derivado a partir da regra do produto:

  P(a ^ b) = P(a|b)P(b)
  P(a ^ b) = P(b|a)P(a)

  Igualando os membros da direita e dividindo por P(b) :

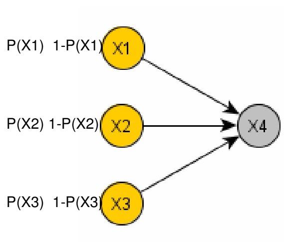
  $$\mathbf{P}(A|B) = \frac{\mathbf{P}(B|A)\mathbf{P}(A)}{\mathbf{P}(B)}$$

- Base de todos os sistemas modernos de IA para inferência probabilística, permite lidar com eventos condicionalmente independentes.

Estrutura de uma rede

- Um conjunto de variáveis aleatórias definem os nós da rede
- Um conjunto de setas ligam as variáveis, se uma seta liga X em direção a Y, então X é “pai” de Y
- Cada nó Y tem uma distribuição de probabilidade condicional P(Yi | pais(Yi))
- O grafo não tem ciclos orientados
- Cada nó possui uma tabela de probabilidade condicional que quantifica a influência dos pais sobre os filhos

Estrutura de uma rede



X1	X2	X3	P(X4)
V	V	V	P1
V	V	F	P2
V	F	F	P3
V	F	V	P4
F	V	V	P5
F	V	F	P6
F	F	V	P7
F	F	F	P8

TABELA DE PROBABILIDADE CONDICIONAL

Estrutura de uma rede

Tabela de probabilidade condicional (TPC)

Cada linha é um caso de condicionamento

Um caso de condicionamento é uma combinação possível de valores dos pais

$$P(X_1, \cdots, X_n) = \prod_{i=1}^{n} P(X_i \mid pais(X_i))$$

Contruindo uma rede

- Utilizando a regra do produto:

$$P(X_1,\ldots,X_n) = P(X_n \mid X_{n-1},\ldots,X_1)P(X_{n-1},\ldots,X_1)$$

- Repetimos o processo, reduzimos cada probabilidade conjuntiva a uma condicional mais uma conjunção menor

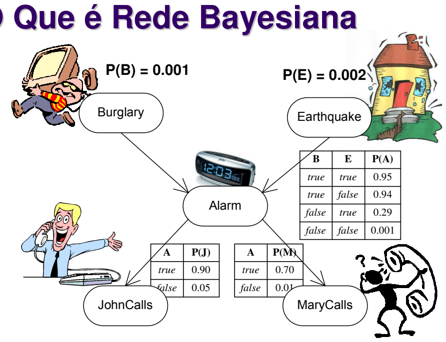
No final teremos:

$$\prod_{i=1}^{n} P(X_i \mid X_{i-1},\ldots,X_1)$$ REGRA DA CADEIA

Contruindo uma rede

- Escolher as variáveis pertinentes ao problema
- Ordem de inserção:
  1 Inserir os nós “raízes”
  2 Variáveis que eles influenciam
  3 Nós folhas (não influenciam nenhuma variável)
- Enquanto houver variáveis para representar:
  1 Adicionar um nó para ela, Xn
  2 Estabelecer pais(Xn) dentre os nós já exixtentes
  3 Definir a tabela de probabilidade condicional

O Que é Rede Bayesiana



B	E	P(A)
true	true	0.95
true	false	0.94
false	true	0.29
false	false	0.001

A	P(J)
true	0.90
false	0.05

A	P(M)
true	0.70
false	0.01

A Rede mostra valores de P(Xi | Parents(Xi))

Inferência ***Bayesianas***

Segundo [CASTILHO & GUTIERREZ, 1997] há três tipos distintos de algoritmos de inferência:

- **Exatos**
  - Enumeração
  - Eliminação de Variáveis
  - *Junction Tree*

  Obs: Inferência exata é intratável para redes grandes e muito conectadas

- **Aproximados**
  - *Forward Sampling* [RUSSEL & NORVIG, 2004]
  - *Likelihood Weighting* [FUNG & CHANG, 1990, RUSSEL & NORVIG, 2004]
  - *Gibbs Sampling* [GEMAN & GEMAN, 1984; RUSSEL & NORVIG, 2003]
  - *Metropolis-hasting*
- **Simbólicos**

PCS 2059 – Inteligência Artificial

Tema 6 - Redes Bayesianas e Inferência Exata

Alunos:

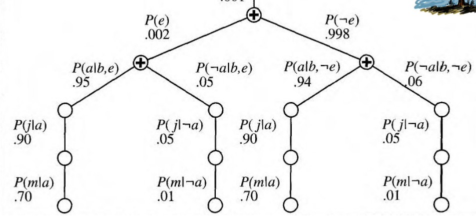
- Gabriel Iseppe Porto
- Raphael Petegrosso
- Victor Tseimazides

Inferência por Enumeração

$$P(b|j,m) = \alpha\, P(b) \sum_e P(e) \sum_a P(a|b,e) P(j|a) P(m|a)\,.$$

P(b | j, m) = α (P(b) . (P(e) . (P(a | b, e) . P(j | a) . P(m | a) + P(~a | b, e) . P(j | ~a) . P(m | ~a)) + P(~e) . (P(a | b, ~e) . P(j | a) . P(m | a) + P(~a | b, ~e) . P(j | ~a) . P(m | ~a)))

Inferência por Enumeração



P(b | j, m) = α . 0,00059224

Redes Bayesianas e Inferência Aproximada

Grupo 7: André Zan Ramos
João Paulo Fiori
Roberto Thiele

Likelihood Weighting

Evidência? não / sim

Gera amostra

Distribuição : P(E|C=n, ...

Número aleatório: 0,55

Amostra gerada: C = n

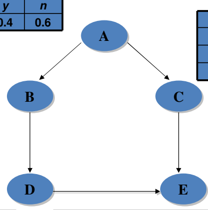
Atualiza peso

Peso = 0.07

Peso = Peso * P(B=n|A=y)

Peso = 0.07

	A	
	y	n
	0.4	0.6



	A	
C	y	n
y	0.7	0.4
n	0.3	0.6

	A	
B	y	n
y	0.3	0.8
n	0.7	0.2

Evidências:

B	D
n	y

Amostras geradas:

A	B	C	D	E
y	n	n	y	y

	B	
D	y	n
y	0.5	0.1
n	0.5	0.9

		E	
C	D	y	n
y	y	0.9	0.1
y	n	0.999	0.001
n	y	0.999	0.001
n	n	0.999	0.001

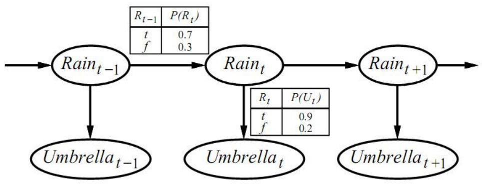
Contador das amostras geradas:

A		B		C		D		E	
y	n	y	n	y	n	y	n	y	n
0.07	0	0	1	0	0.07	1	0	0.07	0

Raciocínio probabilístico no tempo

Alexandre Shiroma – 5174737
Celso de Almeida Saad – 5123393
Márcio Yudi Sato - 5179979

Processos Estacionários e a suposição de Markov



- O modelo de transição: $P(Rain_t|Rain_{t-1})$.
- O modelo sensor: $P(Umbrella_t|Rain_t)$.

Inferência em modelos temporais

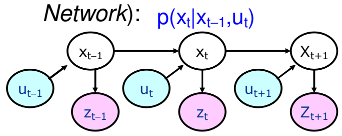
- **Filtragem:** computar o estado de crença atual, dada toda a evidência até ao momento, ou seja, computar $\mathbf{P(X_t|e_{1:t})}$.
- **Previsão:** computar distribuição posterior sobre algum estado futuro, dada toda a evidência até ao momento, ou seja, computar $\mathbf{P(X_{t+k}|e_{1:t})}$ para k>0
- **Suavização:** computar a distribuição posterior sobre algum estado passado, dada toda a evidência até ao momento, ou seja, computar $\mathbf{P(X_k|e_{1:t})}$ para $0 \le k < t$.

Lidando com a Incerteza

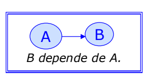
Robótica Probabilística

[Thrun, Burgard, Fox 2005]

- Representação do sistema estocástico dinâmico como uma DBN (*Dynamic Bayes Network*): $p(x_t|x_{t-1},u_t)$



$p(z_t|x_t)$

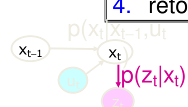


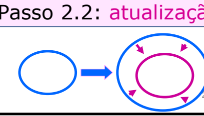
Crença **bel(x_t)**: reflete o conhecimento interno do agente a respeito do estado.

Principal algoritmo para calcular crença: Filtro de Bayes

1. Algoritmo filtro_Bayes (bel(x_{t-1}), u_t, z_t)
2. para todo $\mathbf{x_t}$ faça
   1. **bel'($\mathbf{x_t}$)** = $\int$ $\mathbf{p(x_t|x_{t-1},u_t)}$ bel(x_{t-1}). dx_{t-1}
   2. **bel ($\mathbf{x_t}$)** = η . $\mathbf{p(z_t|x_t)}$. bel'(x_t)
3. fim do para
4. retorne **bel($\mathbf{x_t}$)**

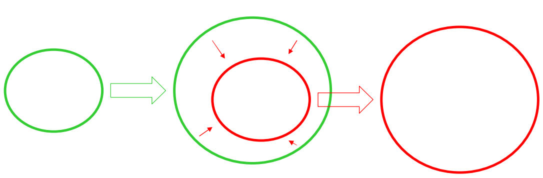
Passo 2.1: predição
Passo 2.2: atualização



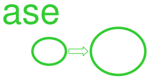


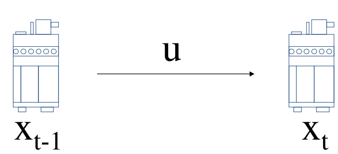
Bayesian Filtering

- Two phases:
  - 1. Prediction Phase
  - 2. Measurement Phase



1. Prediction Phase

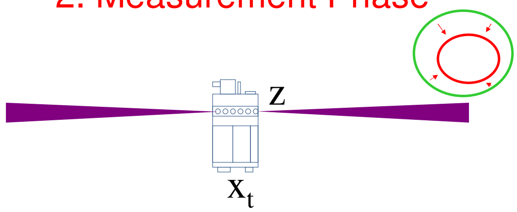




$$P(x_t) = \sum P(x_t|x_{t-1},u)\, P(x_{t-1})$$

Motion Model

2. Measurement Phase

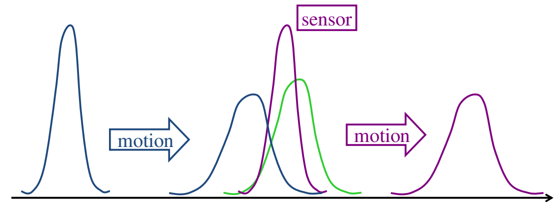


$$P(x_t|z) = k\ P(z|x_t)\ P(x_t)$$

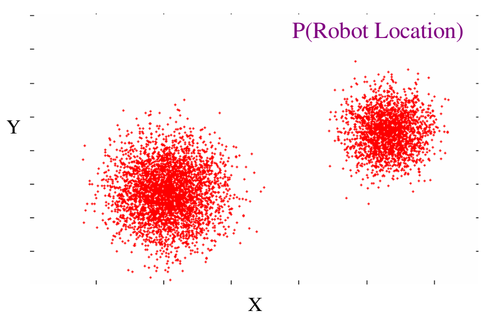
Sensor Model

Kalman Filter

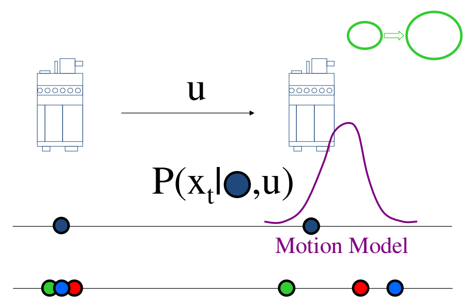
- Very powerful
- Gaussian, unimodal



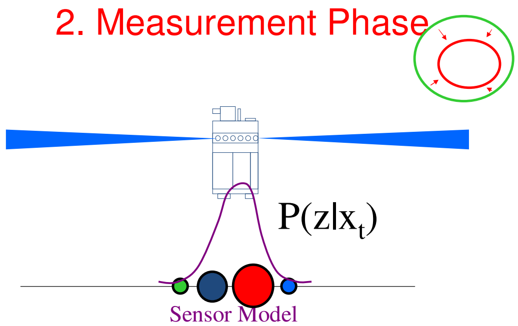
Sampling as Representation



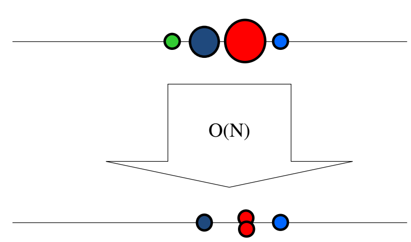
1. Prediction Phase



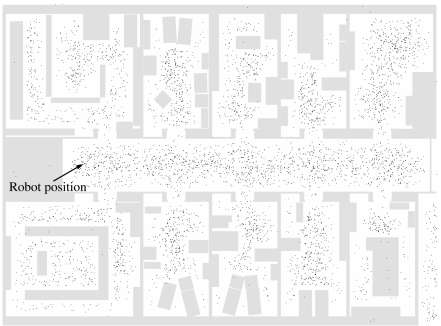
2. Measurement Phase



3. Resampling Step

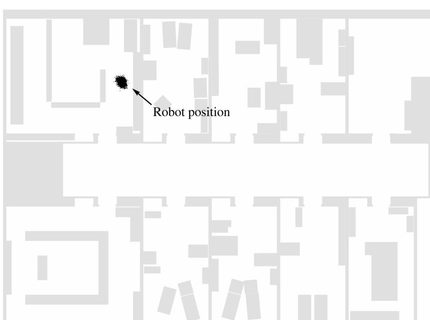


Global Localization



Global Localization (2)

Global Localization (3)



PROBLEMAS DE DECISÃO SEQÜENCIAL

PCS2428/2059 – Inteligência Artificial
Professora :
Anna Helena Reali Costa
Integrantes:
Andre Felipe Santos nºUSP: 3729471
Hernan da Cunha Martinez nºUSP: 5178342
Leandro Moreira da Costa nºUSP: 5178575
06/11/2008

MDP = (S,A,R,T)

Processo de Decisão de Markov (MDP):

- S: conjunto finito de estados ($s \in S$)
- A: conjunto finito de ações ($a \in A$)
- R: função reforço (eventualmente não-deterministica) $R: S \times A \rightarrow \Re$;
  - *r(s,a)* – recompensa obtida por executar *a* em *s*.
- T: função de transição entre estados (eventualmente não-deterministica) $T: S \times A \rightarrow S$;
  - t(s,a,s') – (probabilidade de) estando em *s* e executando *a*, ir para o estado *s'*.

Condição de Markov: Estado atual depende apenas do último par (*estado, ação).*

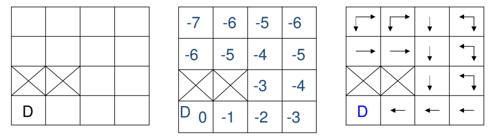
Política

- Solução para um MDP: determinar uma política.
- Qual política? Escolher aquela que produz a ***maior recompensa acumulada*** possível!
- Para explicar como determinar a política, usaremos o conceito de **função valor**.
  - Função valor: Indica o valor (numérico) de cada estado.

Função Valor: exemplo

- Ambiente discreto 4x4, com obstáculos.
- Agente deve alcançar posição destino D a partir de qualquer lugar do ambiente.
- Ações que o agente pode realizar: N, S, L, O.
- Penalidade por executar uma ação (qualquer) = -1
  - Melhor política => caminho mais curto.

Função valor: exemplo



Ambiente exemplo

Função valor ótima: indica as penalidades esperadas até atingir o destino, seguindo uma política ótima.

Política ótima obtida a partir da função valor ótima (melhores ações, para cada estado).

PD: algoritmo de iteração de valor

- Cálculo iterativo da função valor ótima.

  $V(s) \leftarrow \max_a (r_{s,a} + V(s'))$

  Repetir até V(s) estabilizar.

  Sendo:

  s – estado atual, s' – próximo estado,

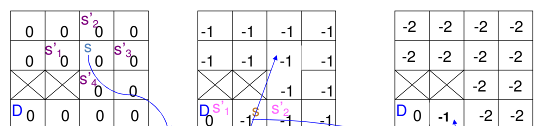
  $r_{s,a}$ – reforço recebido por executar a em s

  V(.) – valor

  Modelo determinístico e horizonte finito.

  OBS: uso da função de transição entre estados T.

Exemplo de cálculo de V(s)



$V(s) = \max_a ((r(s, O) + V(s'_1)),$
$(r(s, N) + V(s'_2)),$
$(r(s, L) + V(s'_3)),$
$(r(s, S) + V(s'_4)))$
$= \max_a ((-1+0), (-1+0), (-1+0), (-1+0))$
$= -1$

$V(s) = \max_a ((r(s, O) + V(s'_1)),$
$(r(s, L) + V(s'_2)))$
$= \max_a ((-1+0), (-1+(-1)))$
$V(s) = -1$

Política ótima

- $V^\pi(s_t)$: valor acumulado (descontado) conseguido por seguir a política arbitrária π a partir de um estado arbitrário s_t ($a_t = \pi(s_t)$, $a_{t+1} = \pi(s_{t+1})$,...):

$V^\pi(s_t) \equiv r_t + \gamma r_{t+1} + \gamma^2 r_{t+2} + \gamma^3 r_{t+3} + \ldots \equiv \sum_{i=0 \to \infty} \gamma^i r_{t+i}$

(discounted cumulative reward)

- **Política ótima**: $\pi^* \equiv \text{argmax}_\pi V^\pi(s)$ $(\forall s)$
  - $V^*(s)$: função valor da política ótima π^*.

Política ótima

- A ação ótima *a* no estado *s* é aquela que maximiza a soma da recompensa imediata somada ao valor V* do estado sucessor imediato, descontado de γ:

$\pi^*(s) \equiv \text{argmax}_a [r(s,a) + \gamma V^*(s')]$

Com: $\pi^*(s)$ – ação ótima no estado s;

$V^*(s')$ – máxima recompensa acumulada descontada obtida ao iniciar a partir do estado s' (seguindo política ótima).