隐藏式马科夫模型

(Hidden Markov Models)

概率原理复习

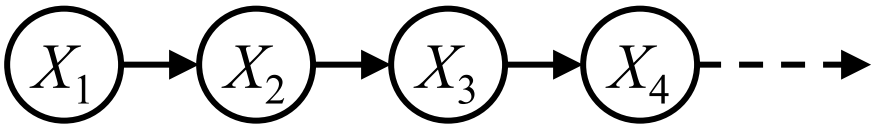
- Conditional probability $$P(x|y) = \frac{P(x,y)}{P(y)}$$
- Product rule $$P(x,y) = P(x|y)P(y)$$
- Chain rule
$$\begin{aligned} P(X_1, X_2, \ldots X_n) &= P(X_1)P(X_2|X_1)P(X_3|X_1, X_2)\ldots \\ &= \prod_{i=1}^{n} P(X_i|X_1, \ldots, X_{i-1}) \end{aligned}$$
- X, Y independent if and only if: $\forall x, y : P(x,y) = P(x)P(y)$
- X and Y are conditionally independent given Z if and only if: $X \perp\!\!\!\perp Y | Z$
$$\forall x, y, z : P(x, y|z) = P(x|z)P(y|z)$$

在时间或空间上的推理

- 通常，我们的推理是建立在一序列的观察值上面的
  - 语音识别
  - 机器人定位
  - 用户的注意力 User attention
  - 医疗监测
- 需要引入时间(或空间)到我们的模型里

马科夫模型（Markov Models）

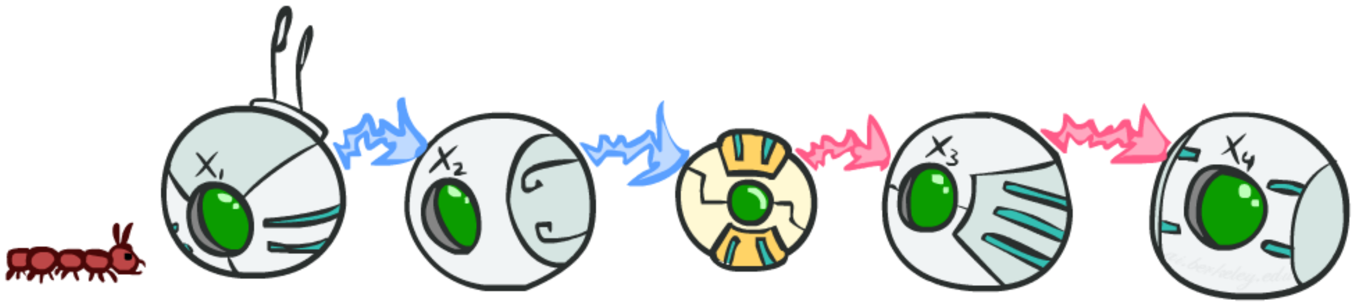
- X 在某个时刻的值是一个随机变量，也叫做一个 状态



$P(X_1)$ $\quad$ $P(X_t|X_{t-1})$ $\quad$ $P(X_t) = ?$

- 参数： 转移概率 或 动态参数，指定状态随时间的发展是如何演变的（参数还包括，初始状态概率分布）
- 稳定性假设：转移概率不随时间发生变化
- 与 MDP 的转移模型类似，但不存在行动上的选择

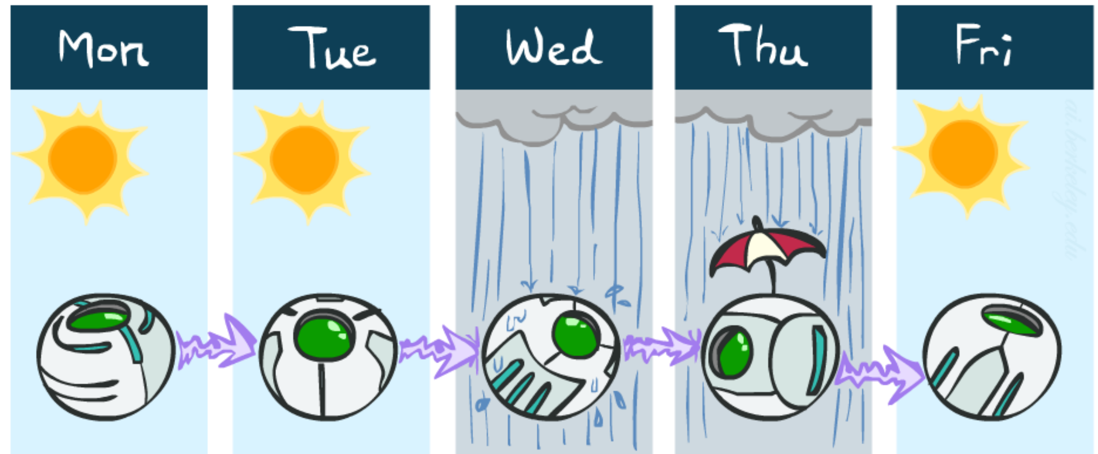
马科夫假设:条件独立性



- 条件独立性的基本性质:
  - 过去和未来给定现在是相互独立的
  - 每个时刻的状态只依赖于上一个时刻的状态
  - 这也叫做(一级) 马科夫属性
- A (growable) BN: 可以应用 贝叶斯网络上推理，当链的长度被剪裁到一个固定长度的时候

马科夫链举例：Weather

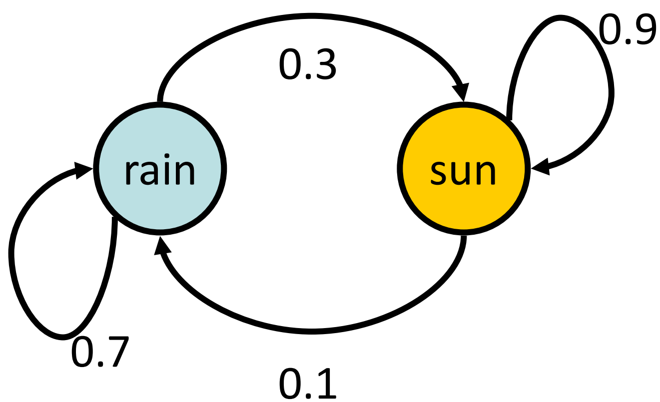
- 状态：$X = \{rain, sun\}$

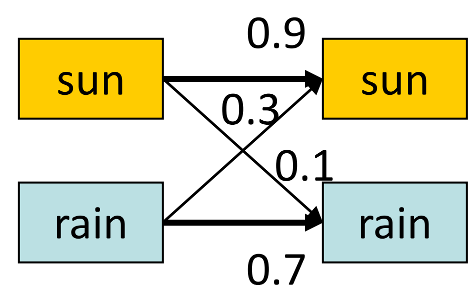


- 初始分布：1.0 sun
- CPT $P(X_t \mid X_{t-1})$：

两种用图的方式来表达 CPT

X_{t-1}	X_t	$P(X_t \mid X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7

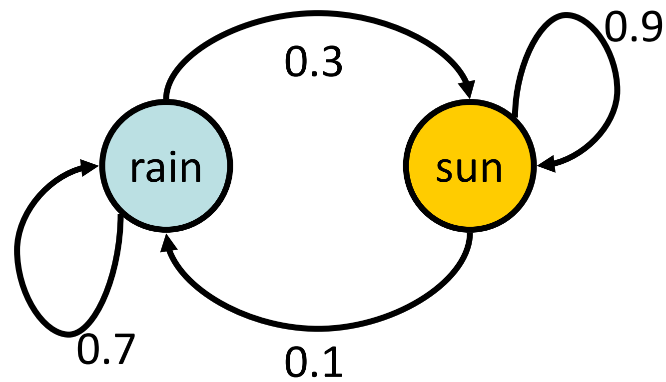




马科夫链举例：Weather

- 初始分布：1.0 sun

- 在过去一个时刻后天气的概率分布变成了什么？



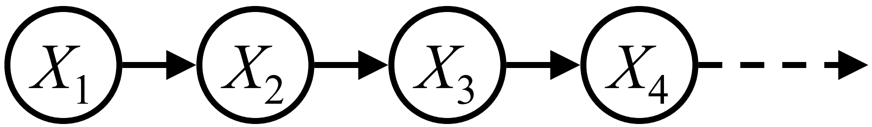
$$P(X_2 = sun) = \sum_{x_1} P(x_1, X_2 = sun)$$
$$= \sum_{x_1} P(X_2 = sun|x_1)P(x_1)$$

$$P(X_2 = \text{sun}) = P(X_2 = \text{sun}|X_1 = \text{sun})P(X_1 = \text{sun}) + P(X_2 = \text{sun}|X_1 = \text{rain})P(X_1 = \text{rain})$$

$$0.9 \cdot 1.0 + 0.3 \cdot 0.0 = 0.9$$

Mini-Forward 算法

- 问题：在某一天t， P(X) 是多少?

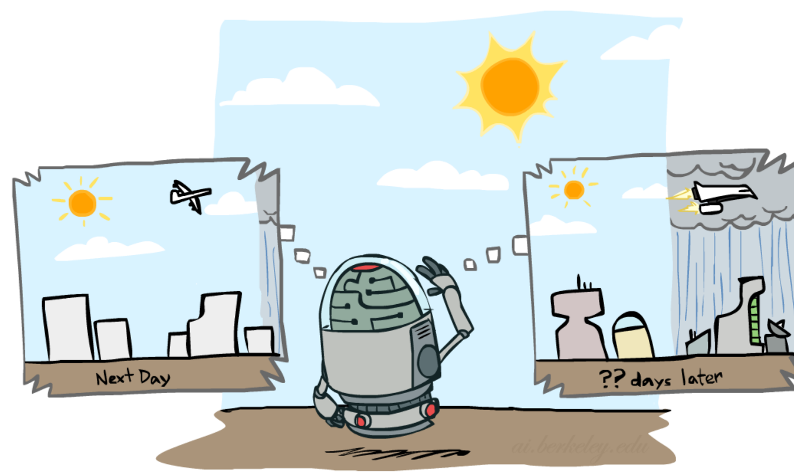


$$P(x_1) = \text{known}$$

$$P(x_t) = \sum_{x_{t-1}} P(x_{t-1}, x_t)$$

$$= \sum_{x_{t-1}} P(x_t \mid x_{t-1}) P(x_{t-1})$$

前向模拟 Forward simulation



运行Mini-Forward 算法的例子

- 从初始观察到 sun

$$\begin{pmatrix}1.0\\0.0\end{pmatrix}\quad\begin{pmatrix}0.9\\0.1\end{pmatrix}\quad\begin{pmatrix}0.84\\0.16\end{pmatrix}\quad\begin{pmatrix}0.804\\0.196\end{pmatrix}\Rightarrow\begin{pmatrix}0.75\\0.25\end{pmatrix}$$

$$\mathrm{P}(X_1)\quad \mathrm{P}(X_2)\quad \mathrm{P}(X_3)\quad \mathrm{P}(X_4)\quad \mathrm{P}(X_\infty)$$

- 从初始观察到 rain

$$\begin{pmatrix}0.0\\1.0\end{pmatrix}\quad\begin{pmatrix}0.3\\0.7\end{pmatrix}\quad\begin{pmatrix}0.48\\0.52\end{pmatrix}\quad\begin{pmatrix}0.588\\0.412\end{pmatrix}\Rightarrow\begin{pmatrix}0.75\\0.25\end{pmatrix}$$

$$\mathrm{P}(X_1)\quad \mathrm{P}(X_2)\quad \mathrm{P}(X_3)\quad \mathrm{P}(X_4)\quad \mathrm{P}(X_\infty)$$

- 从另一个初始分布 $\mathrm{P}(X_1)$:

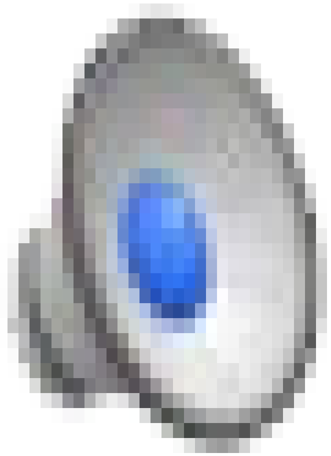
$$\begin{pmatrix}p\\1-p\end{pmatrix}\quad \cdots \Rightarrow\begin{pmatrix}0.75\\0.25\end{pmatrix}$$

$$\mathrm{P}(X_1)\qquad \mathrm{P}(X_\infty)$$

Video of Demo Ghostbusters Basic Dynamics

Video of Demo Ghostbusters Circular Dynamics

Video of Demo Ghostbusters Whirlpool Dynamics



静态分布
(Stationary Distributions)

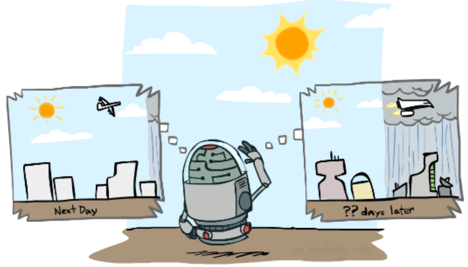
- 对于大部分马科夫链来说:
  - 初始分布的影响随时间的发展变得越来越小
  - 最终的分布是独立于初始分布的

- 静态分布:
  - 这个最终的分布叫做这个马科夫链的 静态分布 P_∞
  - 满足以下关系:

$$P_\infty(X) = P_{\infty+1}(X) = \sum_x P(X|x)P_\infty(x)$$

举例：静态分布

- 问题： P(X) =? 当 t = infinity



$$X_1 \rightarrow X_2 \rightarrow X_3 \rightarrow X_4 \dashrightarrow$$

$$P_\infty(sun) = P(sun|sun)P_\infty(sun) + P(sun|rain)P_\infty(rain)$$
$$P_\infty(rain) = P(rain|sun)P_\infty(sun) + P(rain|rain)P_\infty(rain)$$

$$P_\infty(sun) = 0.9P_\infty(sun) + 0.3P_\infty(rain)$$
$$P_\infty(rain) = 0.1P_\infty(sun) + 0.7P_\infty(rain)$$

$$P_\infty(sun) = 3P_\infty(rain)$$
$$P_\infty(rain) = 1/3P_\infty(sun)$$

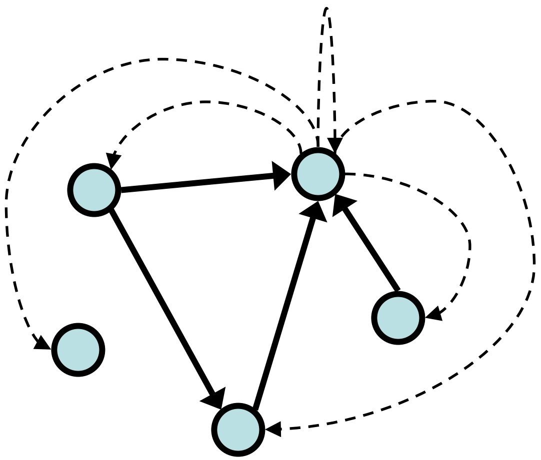
X_{t-1}	X_t	$P(X_t \mid X_{t-1})$
sun	sun	0.9
sun	rain	0.1
rain	sun	0.3
rain	rain	0.7

Also: $P_\infty(sun) + P_\infty(rain) = 1$

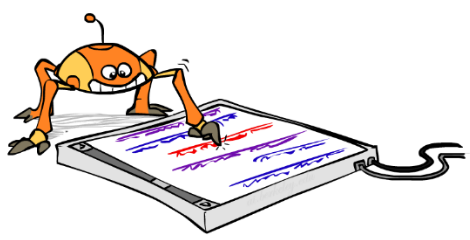
$$P_\infty(sun) = 3/4$$
$$P_\infty(rain) = 1/4$$

静态分布的应用：网页链接分析

- 网页图上的 网页等级排序 PageRank
  - 每个网页是一个随机状态的可能的值
  - 初始分布：均匀分布在所有网页上
  - 过渡转移：
    - 均匀跳转到一个随机的网页，概率为 c（虚线所示，没有显示全部）
    - 随机根据一个外向链接转移到下一个网页，概率为 1-c,(实线所示)



- 静态分布
  - 更多的时间会停留在访问频率高的网页上
  - 例如， 许多路径走到 Acrobat Reader 的下载页面
  - 比较有效地对抗垃圾链接
  - Google 1.0 搜索算法 返回的网页包括所有你输入的关键词，并已排序结果的降序显示出来
  - 现在所有的搜索引擎使用网页的链接分析，并结合许多其他的因素（随着时间的推移，等级 rank 实际上变得越来越不重要）



静态分布的应用：Gibbs 采样

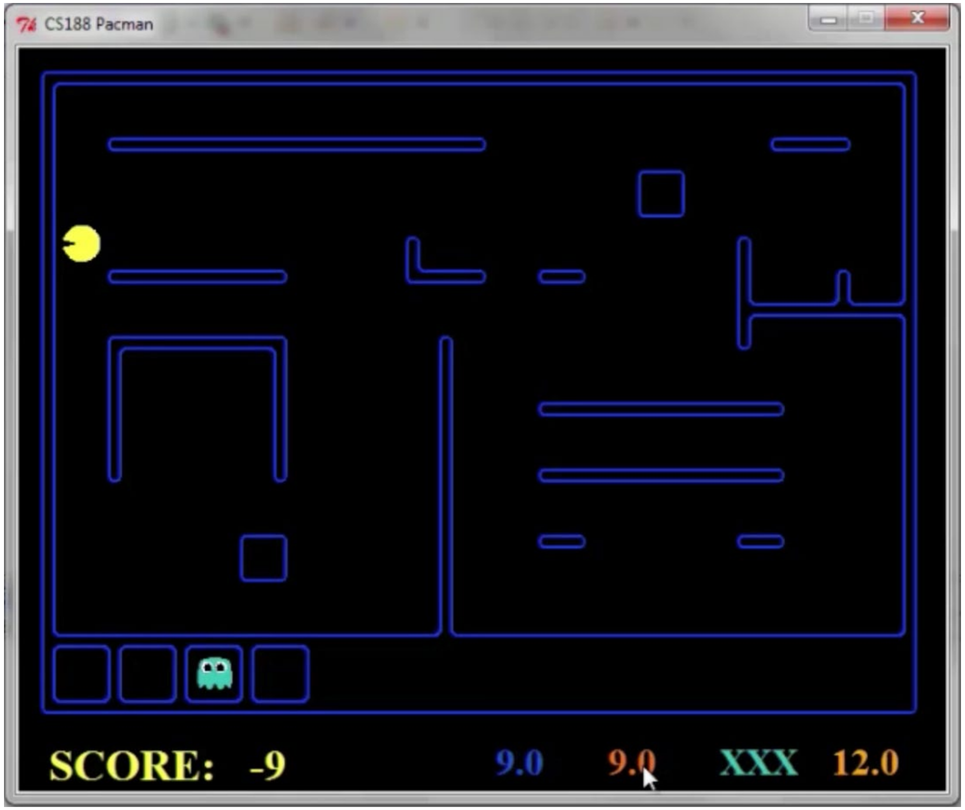
- 在所有隐藏和查询变量上的联合的实例化，即每一组赋值表示为一个状态：$\{X_1, \cdots, X_n\} = H \cup Q$
- 过渡转移：
  - 以概率为 1/n，重新采样变量 X_j 根据以下分布

    $$P(X_j \mid x_1, x_2, \cdots, x_{j-1}, x_{j+1}, \cdots, x_n, e_1, \cdots, e_m)$$

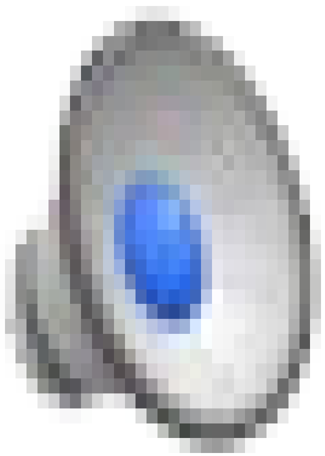
- 静态分布：
  - 条件概率分布 $P(X_1, X_2, \cdots, X_n | e_1, \cdots, e_m)$
  - 意味着当运行 Gibbs 采样一段长时间后，我们将会从我们所希望的概率分布上进行采样
  - 这个结论是可以证明的！

隐藏式马科夫模型
Hidden Markov Models

Pacman – Sonar (P4)

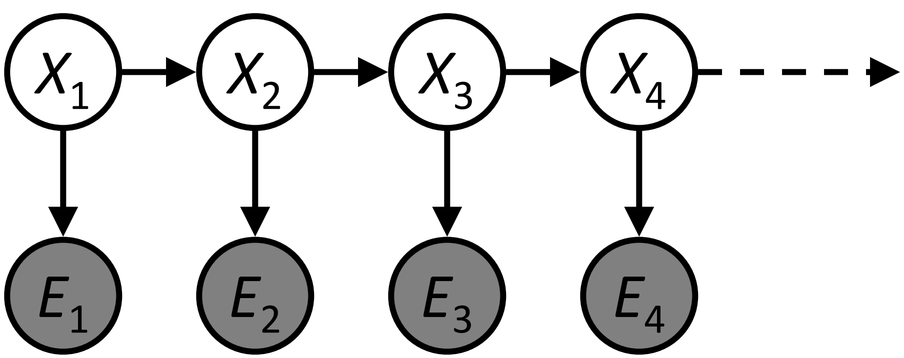


Video of Demo Pacman – Sonar (no beliefs)

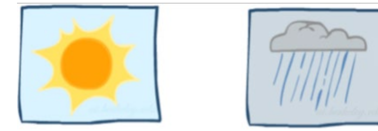


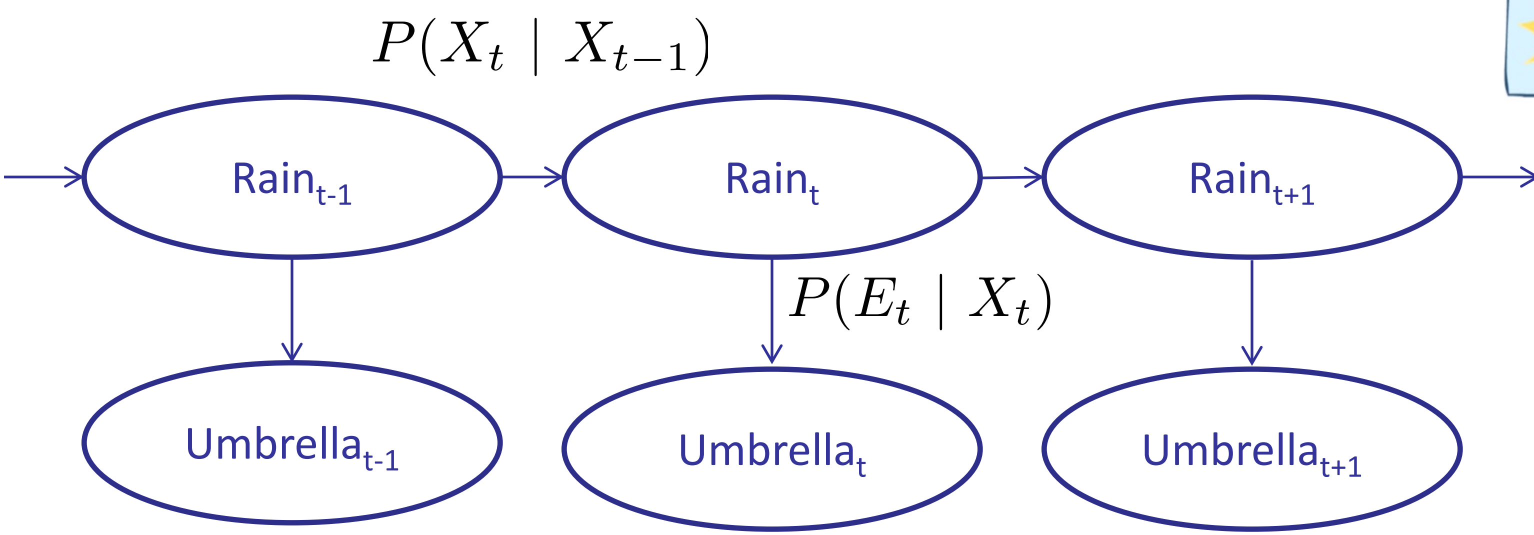
隐藏式马科夫模型

- 马科夫链对于大多数智能体来说不是非常有用
  - 需要用观察到的信息来更新 隐藏变量的置信度(belief)(概率分布)
- 隐式马科夫模型(HMMs)
  - 马科夫链构建在变量 X 上
  - 在每个时刻，你观察到输出结果（E的值），这个输出依赖于X



举例：Weather HMM





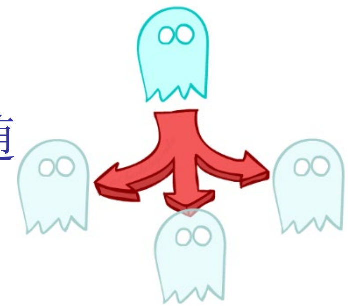
- HMM 的构成：
  - 初始的分布： $P(X_1)$
  - 转移模型： $P(X_t \mid X_{t-1})$
  - 释放/输出模型： $P(E_t \mid X_t)$

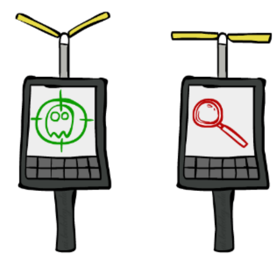
R_{t-1}	R_t	$P(R_t \mid R_{t-1})$
+r	+r	0.7
+r	-r	0.3
-r	+r	0.3
-r	-r	0.7

R_t	U_t	$P(U_t \mid R_t)$
+r	+u	0.9
+r	-u	0.1
-r	+u	0.2
-r	-u	0.8

举例：幽灵追捕者的 HMM

- $P(X_1)$ = 均匀分布
- $P(X|X')$ = 大体顺时针移动，但有时也随机向其他方向移动，或呆在原地不动
- $P(R_{ij}|X)$ = 与之前相同的传感器模型：红色表示靠近，绿色表示离得还远



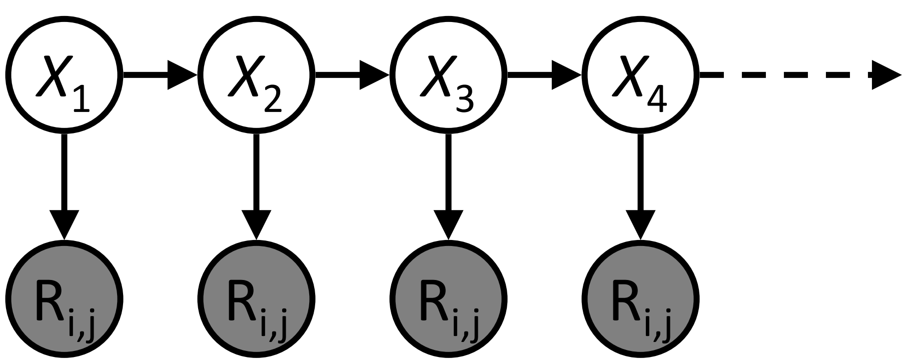


1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

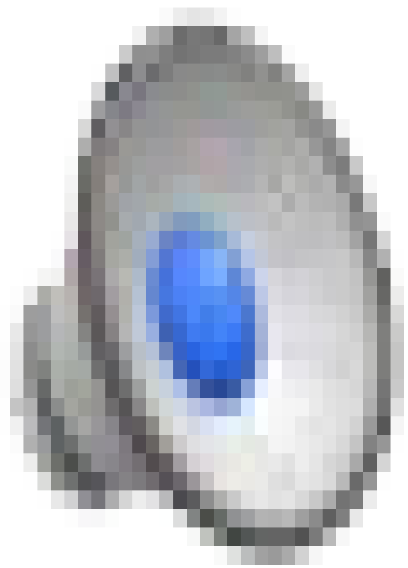
$P(X_1)$

1/6	1/6	1/2
0	1/6	0
0	0	0

$P(X|X'=<1,2>)$

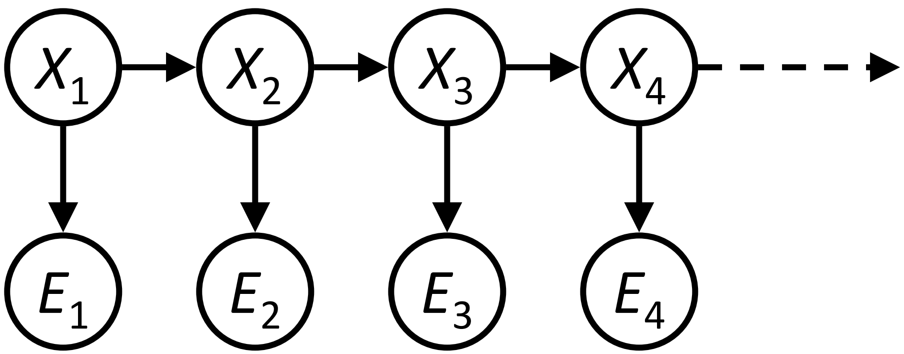


幽灵追捕者视频演示
Circular Dynamics -- HMM



条件独立性

- HMMs 有两个重要的独立性属性:
  - 马科夫隐式过程: 未来的状态依赖于过去的，仅通过当前的状态
  - 当前的观察值(Ei)独立于所有其他的状态，当给定当前的状态(Xi)



- 这意味着观察变量(E)之间是保证独立的吗?
  - [不是， 它们通过隐藏状态趋向于相互关联]

真实的 HMM 应用举例

- 机器人追踪定位:
  - 观察到的是距离传感器的读数 range readings (连续的数值)
  - 隐藏状态是机器人在地图上的位置 (连续的)

- 语音识别 HMMs:
  - 观察值是语音信号(连续数值)
  - 状态是在特定词语里的具体位置(字母或字母组合)

- 机器翻译 HMMs:
  - 观察值是单词 (成千上万的)
  - 状态是翻译选项

过滤 / 监测

- 过滤或监测(Filtering/monitoring)，指的是随时间的发展追踪某个状态的置信分布 $B_t(X) = P_t(X_t \mid e_1, \cdots, e_t)$

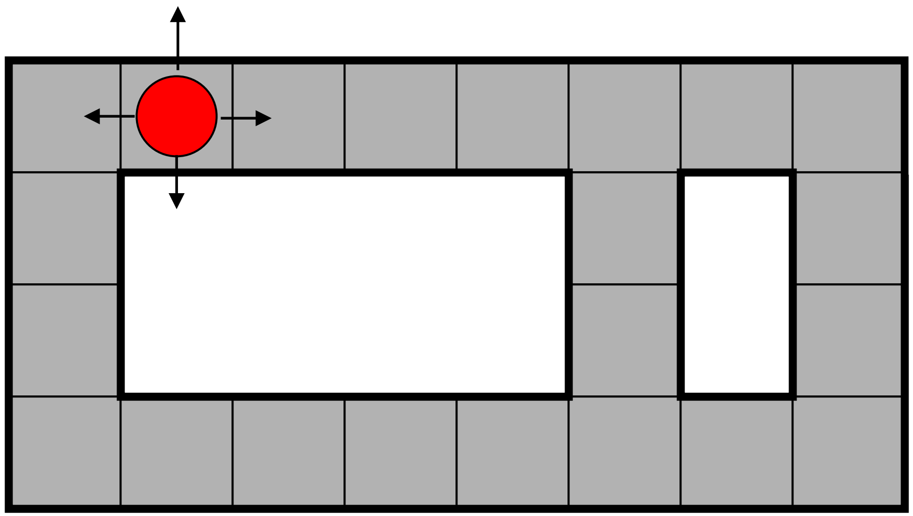
- 开始于一个初始的概率分布 $B_1(X)$，通常是均匀分布的

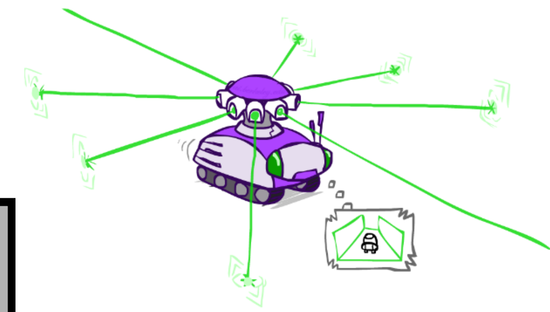
- 随着时间的推移，基于我们获取到的观察(observations)，我们更新 B(X)

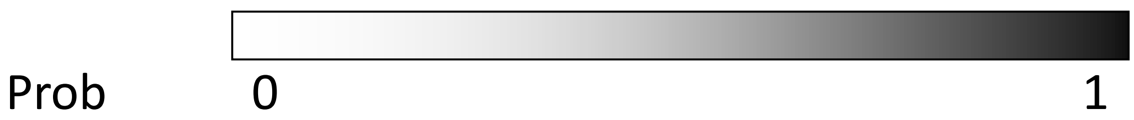
- 卡尔曼滤波 (Kalman filter) 发明于60’年代，最早实现的此类方法，在阿波罗登月计划中用于航天器的轨迹估计

举例：机器人定位追踪

Example from
Michael Pfeiffer





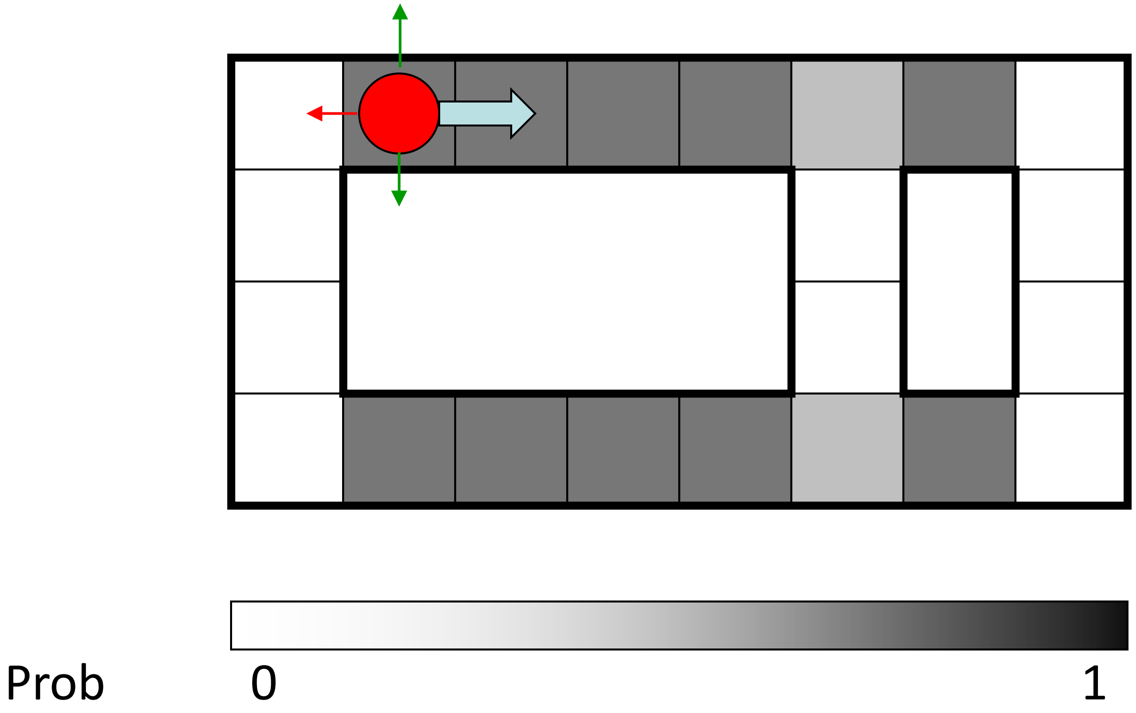


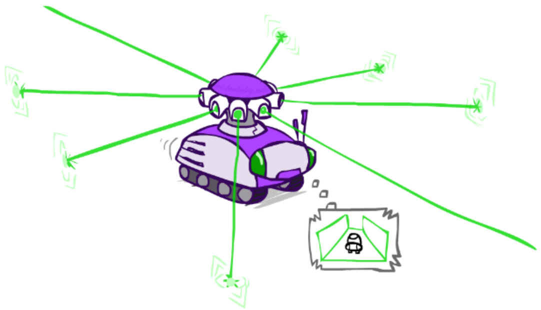
t=0

传感器模型：能够获取到在哪些方向上有一堵墙，误判不超过1

移动模型：存在一个小概率不会按指令执行相应的动作

举例：机器人定位追踪

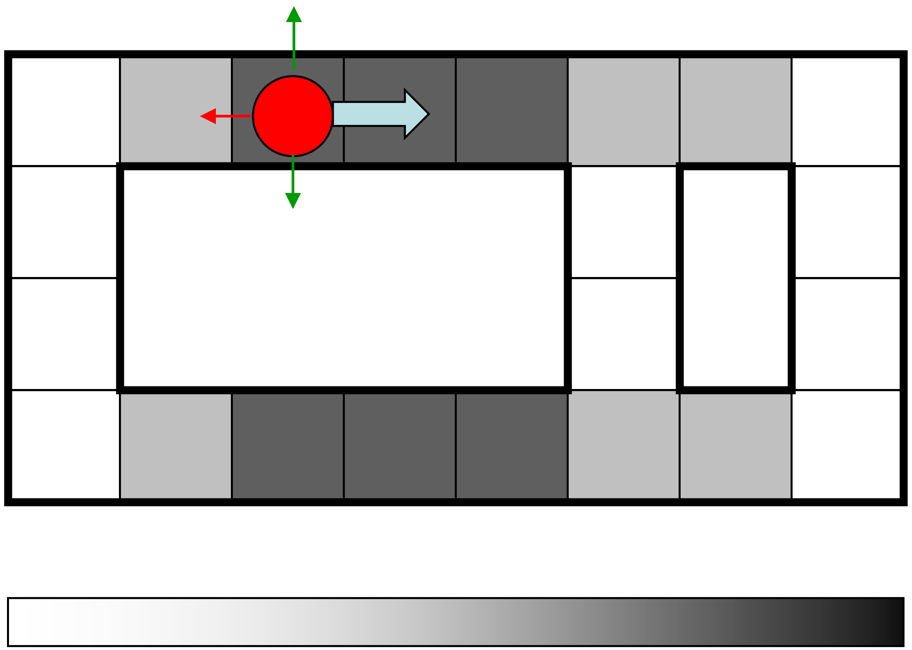




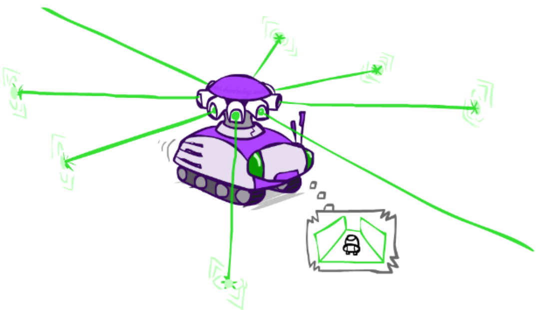
t=1

浅灰色：有可能的情况，因为探测可能会在某一个方向上有错误，但可能性相应比较小（1/4的概率）

举例：机器人定位追踪

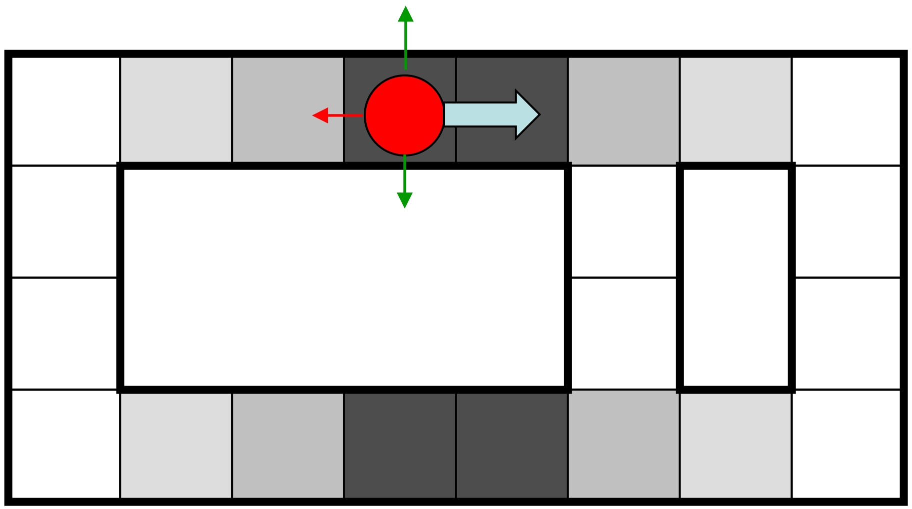


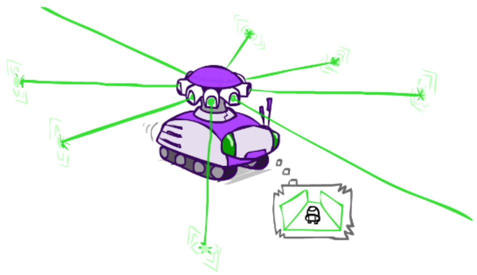
Prob 0 1

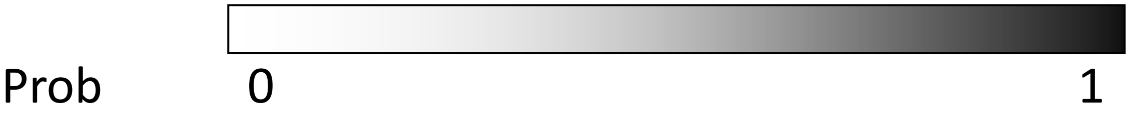


t=2

举例：机器人定位追踪

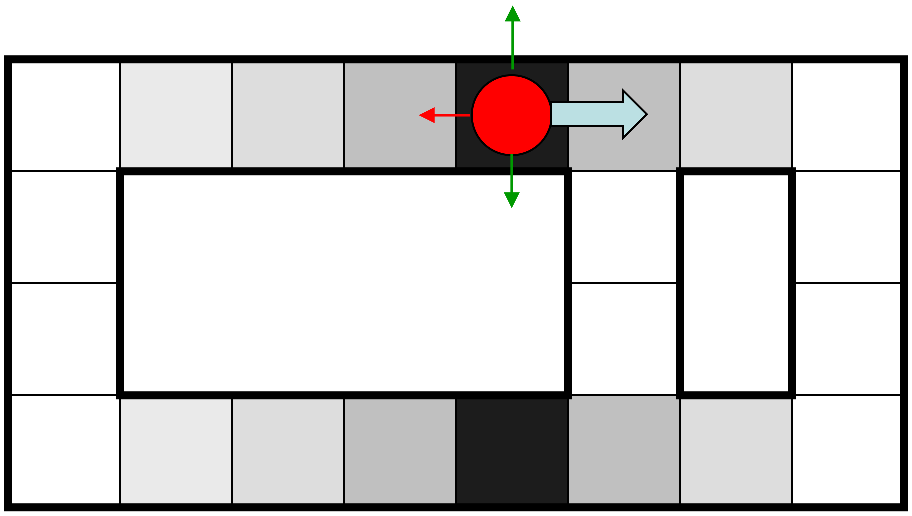


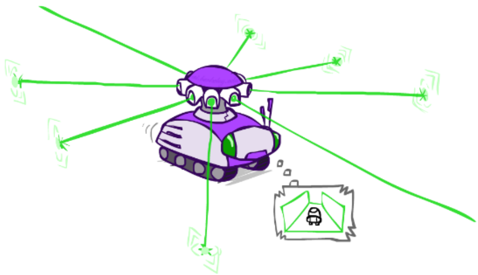


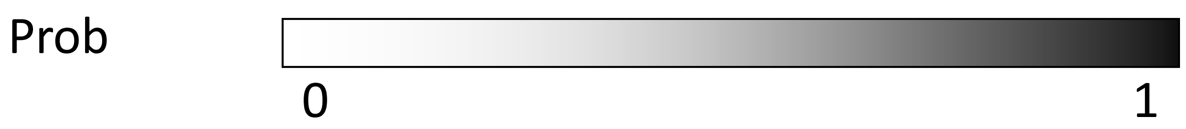


t=3

举例：机器人定位追踪

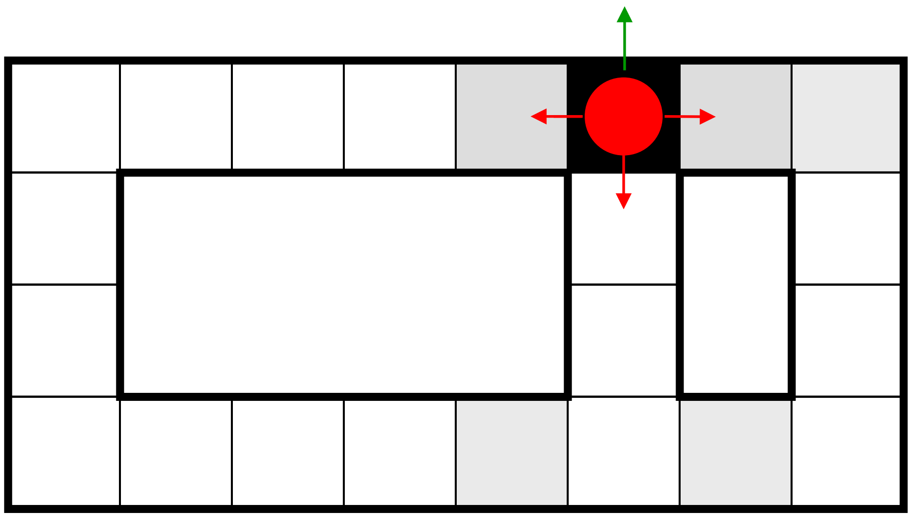


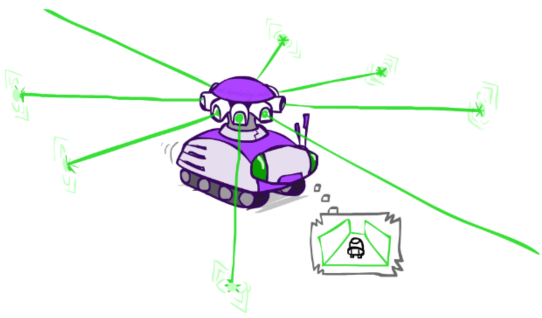


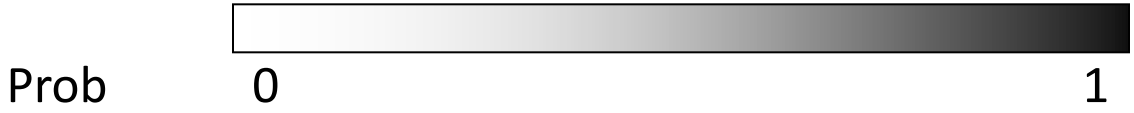


t=4

举例：机器人定位追踪







t=5

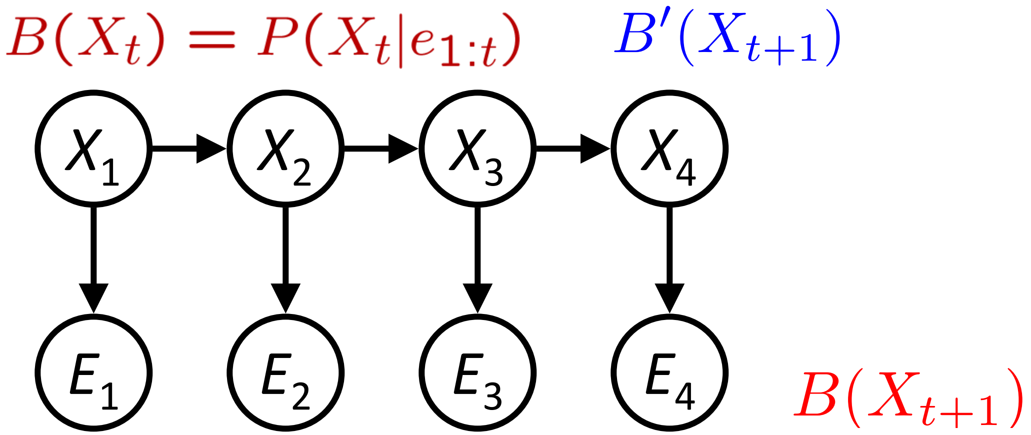
推理：给定观察到的找到状态的分布

- 在每一时刻我们获得传感器的观察值，并想知道：

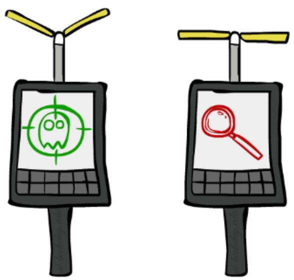
$$B_t(X) = P(X_t|e_{1:t})$$

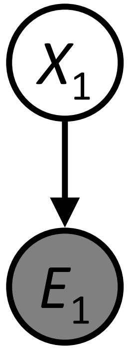
- 想法：从 $P(X_1)$开始 从 B_{t-1} 推导 B_t
  - 等价于，从 B_t 推导 B_{t+1}

两个步骤：时间推移 + 观察过滤

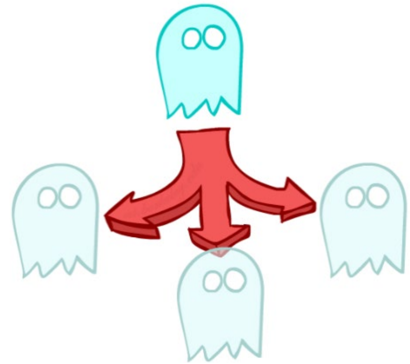


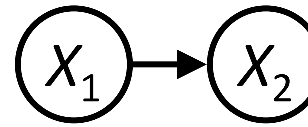
推理：基础情况





$$P(X_1|e_1)$$





$$P(X_2)$$

$$P(X_1|E_1 = e_1) = \frac{P(E_1 = e_1|X_1)P(X_1)}{\sum_{X_1} P(E_1 = e_1|X_1)P(X_1)}$$

$$= \alpha P(E_1 = e_1|X_1)P(X_1)$$

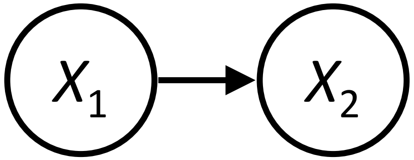
$$P(X_2) = \sum_{x_1} P(x_1, X_2)$$

$$P(X_2) = \sum_{x_1} P(X_2|x_1)P(x_1)$$

时间上的推移

- 假设当前的置信分布是 P(X | 给定所有当前的观察)

$B(X_t) = P(X_t|e_{1:t})$



- 然后，在下一个时刻里：

$$P(X_{t+1}|e_{1:t}) = \sum_{x_t} P(X_{t+1}, x_t|e_{1:t})$$

$$= \sum_{x_t} P(X_{t+1}|x_t, e_{1:t})P(x_t|e_{1:t})$$

$$= \sum_{x_t} P(X_{t+1}|x_t)P(x_t|e_{1:t})$$

- **或简要表示为:**

$$B'(X_{t+1}) = \sum_{x_t} P(X'|x_t)B(x_t)$$

- 基本想法：根据转移模型“推移”置信分布
  - 在标记“B”里，应注意时间t和所包括的观察值（哪一时刻的）

举例：时间上的推移

- 随时间的推移，不确定性逐渐 “积累”

<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	1.00	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

T = 1

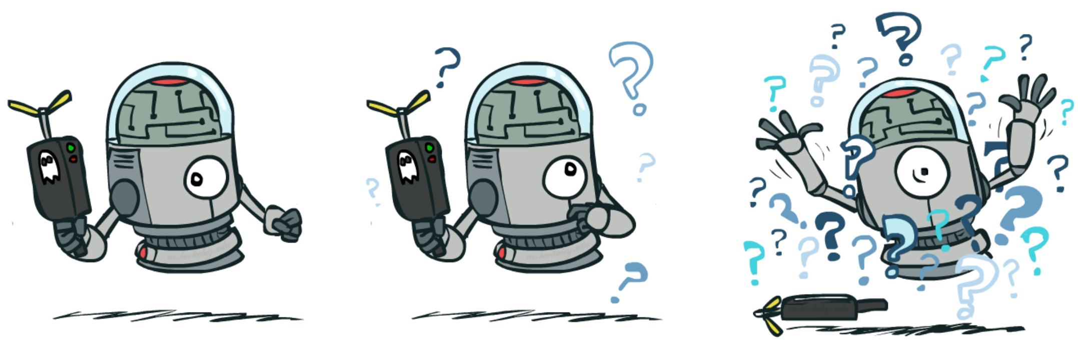
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01
<0.01	0.76	0.06	0.06	<0.01	<0.01
<0.01	<0.01	0.06	<0.01	<0.01	<0.01

T = 2

0.05	0.01	0.05	<0.01	<0.01	<0.01
0.02	0.14	0.11	0.35	<0.01	<0.01
0.07	0.03	0.05	<0.01	0.03	<0.01
0.03	0.03	<0.01	<0.01	<0.01	<0.01

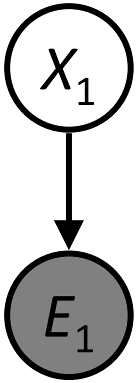
T = 5

(这里的转移模型：幽灵通常沿顺时针方向移动)



观察（更新）

- 假设当前的置信分布是 P(X | 之前的观察):



$$B'(X_{t+1}) = P(X_{t+1}|e_{1:t})$$

- 那么，获得当前时刻(t+1)观察值之后:

$$P(X_{t+1}|e_{1:t+1}) = P(X_{t+1}, e_{t+1}|e_{1:t})/P(e_{t+1}|e_{1:t})$$

$$\propto_{X_{t+1}} P(X_{t+1}, e_{t+1}|e_{1:t})$$

$$= P(e_{t+1}|e_{1:t}, X_{t+1})P(X_{t+1}|e_{1:t})$$

$$= P(e_{t+1}|X_{t+1})P(X_{t+1}|e_{1:t})$$

输出模型/观察模型

- 或简单表示为:

$$B(X_{t+1}) \propto_{X_{t+1}} P(e_{t+1}|X_{t+1})B'(X_{t+1})$$

举例：观察

- 获得观察值后，置信分布得到再权重变化更新，从而降低了不确定性

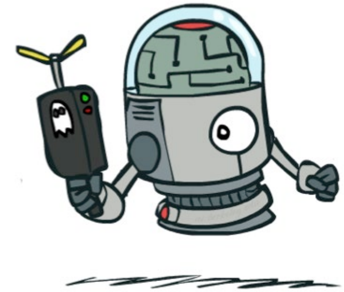
0.05	0.01	0.05	<0.01	<0.01	<0.01
0.02	0.14	0.11	0.35	<0.01	<0.01
0.07	0.03	0.05	<0.01	0.03	<0.01
0.03	0.03	<0.01	<0.01	<0.01	<0.01

观察之前

<0.01	<0.01	<0.01	<0.01	0.02	<0.01
<0.01	<0.01	<0.01	0.83	0.02	<0.01
<0.01	<0.01	0.11	<0.01	<0.01	<0.01
<0.01	<0.01	<0.01	<0.01	<0.01	<0.01

观察之后

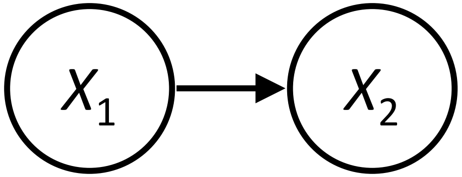
$$B(X) \propto P(e|X)B'(X)$$



在线置信度更新(Online Belief Updates)

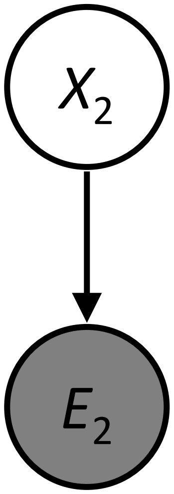
- 在每一时刻，开始于当前的置信分布 P(X | evidence)
- 随时间推移进行更新：

$$P(x_t|e_{1:t-1}) = \sum_{x_{t-1}} P(x_{t-1}|e_{1:t-1}) \cdot P(x_t|x_{t-1})$$



- 根据观察到的证据进行更新：

$$P(x_t|e_{1:t}) \propto_X P(x_t|e_{1:t-1}) \cdot P(e_t|x_t)$$



- 前向算法(forward algorithm) 每次执行这两步 (但不进行正规化)

前向推移算法
(The Forward Algorithm)

- 在每一时刻我们都获得观察证据，并想知道：

$$B_t(X) = P(X_t|e_{1:t})$$

- 可以推导出下面的更新过程：

如果我们想知道 P(x|e)可以在每一时刻进行正规化，或再最后一起做一次正规化

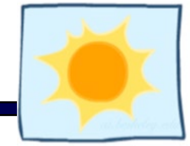
$$P(x_t|e_{1:t}) \propto_X P(x_t, e_{1:t})$$

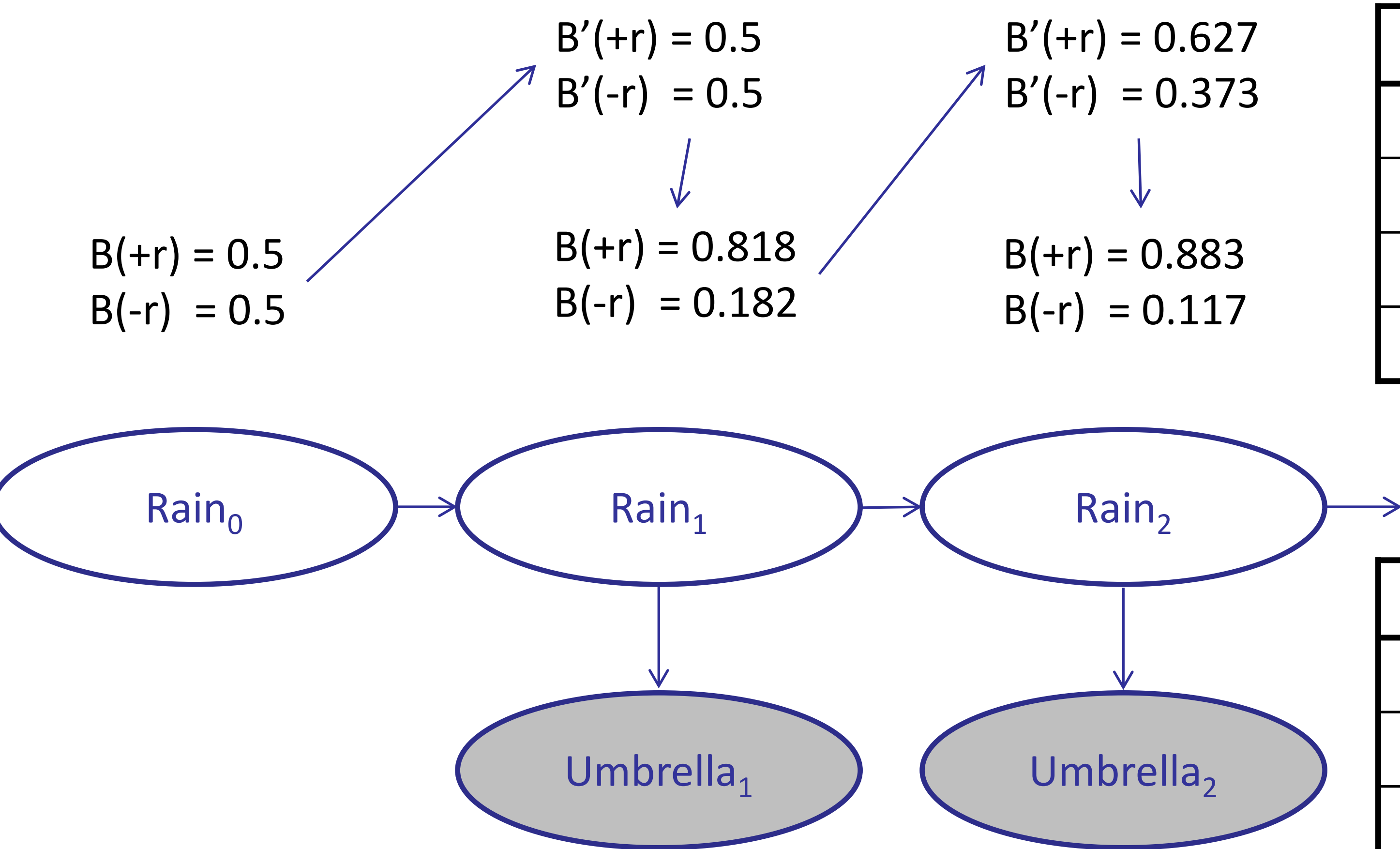
$$= \sum_{x_{t-1}} P(x_{t-1}, x_t, e_{1:t})$$

$$= \sum_{x_{t-1}} P(x_{t-1}, e_{1:t-1}) P(x_t|x_{t-1}) P(e_t|x_t)$$

$$= P(e_t|x_t) \sum_{x_{t-1}} P(x_t|x_{t-1}) P(x_{t-1}, e_{1:t-1})$$

举例：Weather HMM

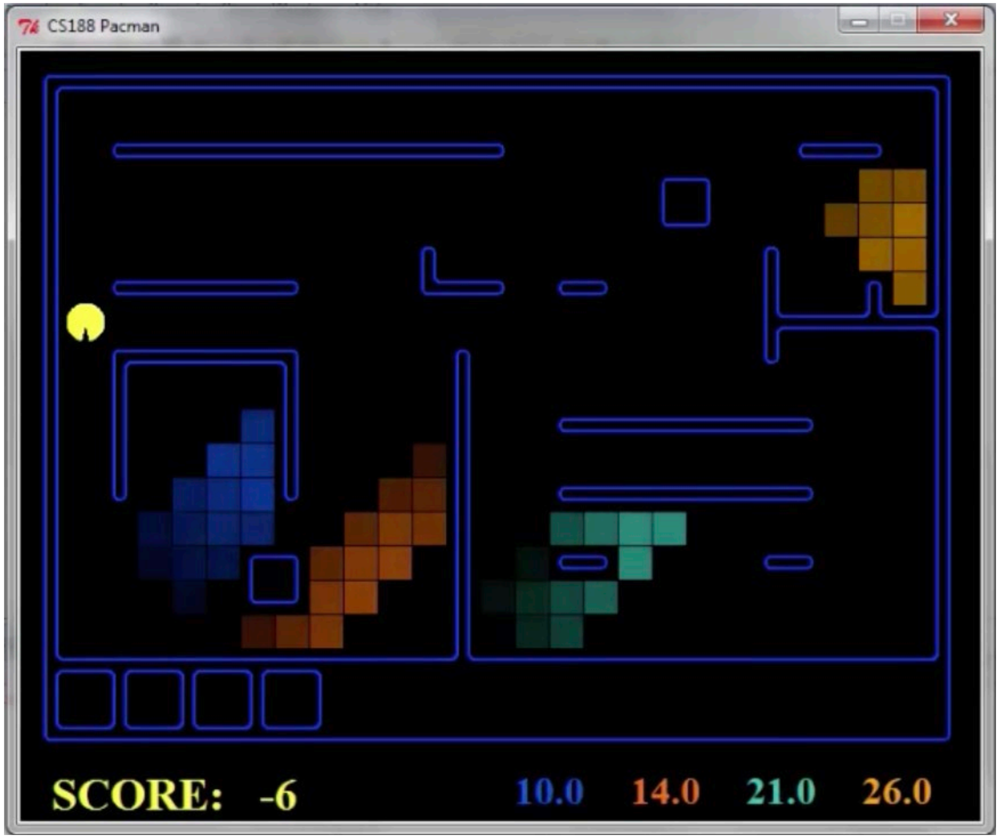




R_t	R_{t+1}	$P(R_{t+1}\|R_t)$
+r	+r	0.7
+r	-r	0.3
-r	+r	0.3
-r	-r	0.7

R_t	U_t	$P(U_t\|R_t)$
+r	+u	0.9
+r	-u	0.1
-r	+u	0.2
-r	-u	0.8

Pacman – Sonar (P4)



Pacman视频演示
声波定位幽灵（通过幽灵位置的置信分布）