Hidden Markov Models (HMM)
Prob. Basics:
- Cond. Prob.: 
  \[ P(A \mid B) = \frac{P(A, B)}{P(B)} \]
  - Bayes' Rule: 
  \[ P(A_i \mid B) = \frac{P(B \mid A_i) P(A_i)}{\sum_{i=1}^{n} P(B \mid A_i) P(A_i)} \]
- Joint Probability: 
  \[ P(A, B) \]
- Marginal Prob: 
  \[ P(A) = \sum_B P(A, B) \]
- Chain Rule: 
  \[ P(A, B, C) = P(A \mid B, C) \cdot P(B \mid C) \cdot P(C) \]
Robot moving from one grid to another

localize at $t_1$  localize at $t_2$  localize at $t_3$

$S_1$ $S_2$ $S_3$ $S_4$ ... $S_n$

True Robot motion

measurement

motion model

Motion model governs how robot moves from one grid location to another.

Example:

$P(S_i \rightarrow S_{i+1}) = 0.5$  
$P(S_i \rightarrow S_{i+2}) = 0.2$  
$P(S_i \rightarrow S_{i+j}, j \geq 3) = 0$  
$P(S_i \rightarrow S_i) = 0.2$  
$P(S_i \rightarrow S_{i-1}) = 0.1$  
$P(S_i \rightarrow S_{i-k}, k \geq 2) = 0$

Model the robot motion tracking problem as:

Estimate the true state and motion path (i.e., green)
Now goal is to measure the robot’s location at every time, and combine with the motion model (i.e., state transition probabilities) to infer the true state of the robot.

Modeled as a "State Transition Diagram"
Bayesian Filtering

\[ P(S_k | m_1:n) \]

**Good result to keep in mind**

\[ P(S_{1:n} | m_1:n) = \frac{P(S_{1:n}, m_1:n)}{P(m_1:n)} = \frac{P(m_n | m_1:n-1, S_{1:n}) P(m_{n-1} | m_1:n-2, S_{1:n}) \cdots P(m_1 | S_{1:n}) P(S_n | S_{1:n-1}) \cdots P(S_2 | S_1) P(S_1)}{P(m_1:n)} \]

\[ = P(m_n | S_n) P(m_{n-1} | S_{n-1}) \cdots P(m_1 | S_1) \cdot \prod_{i=2}^{n} P(m_i | S_i) P(S_i | S_{i-1}) \]
Now, we want $P(S_k | m_1:n)$

$$P(S_k | m_1:n) = P(S_k, m_1:n) = P(S_k, m_1:k, m_{k+1:n})$$

$$= P(m_{k+1:n} | S_k, m_1:k) \cdot P(S_k, m_1:k)$$

$$= P(m_{k+1:n} | S_k) \cdot P(S_k | m_1:k) \cdot P(m_1:k)$$

$$= P(m_{k+1:n} | S_k) \cdot P(S_k | m_1:k)$$

$$= P(S_k | m_1:k) \cdot P(m_{k+1:n} | S_k)$$

Forward (online)  
Backward (offline)

3. Let's look at the forward component

$$P(S_k | m_1:k) = \frac{P(S_k, m_1:k)}{P(m_1:k)}$$

$$= \sum_{S_{k-1}} P(S_k, S_{k-1}, m_1:k)$$

$$= \sum_{S_{k-1}} P(m_k | S_k, S_{k-1}, m_1:k-1) P(S_k, S_{k-1}, m_1:k-1)$$

$$= \sum_{S_{k-1}} P(m_k | S_k) \cdot P(S_k | S_{k-1}, m_1:k-1) P(S_{k-1}, m_1:k-1)$$

Call this $\alpha_k$. 
$\alpha_k = \sum_{S_{k-1}} P(m_k|S_k) P(S_k|S_{k-1}) P(S_{k-1}\text{ init} : 1:k-1)$

$\alpha_k = \sum_{S_{k-1}} P(m_k|S_k) P(S_k|S_{k-1}) \alpha_{k-1}$

dynamic program

Initial condition $= \alpha_1 = P(m_1|S_1) P(S_1)$ would be known.

For instance, $P(S_i)$ might be equal prob. for all states, while $P(m_1|S_1)$ is a probability derived from sensor data sheet.
\[ P(m_{k+1:n} | s_k) = P(m_{k+1:n}, s_k) \]
\[ = \frac{1}{P(s_k)} \sum_{s_{k+1}} P(m_{k+1:n}, s_k, s_{k+1}) \]
\[ = \frac{1}{P(s_k)} \sum_{s_{k+1}} P(m_{k+2:n} | s_{k+1}, s_k, m_{k+1}) \cdot P(s_{k+1} | s_k) \]
\[ = \sum_{s_{k+1}} P(m_{k+2:n} | s_{k+1}) \cdot P(m_{k+1} | s_{k+1}) \cdot P(s_{k+1} | s_k) \]
\[ \therefore \beta_k = \sum_{s_{k+1}} \beta_{k+1} \cdot P(m_{k+1} | s_{k+1}) \cdot P(s_{k+1} | s_k) \]

Again, dynamic program.
**PARTICLE FILTER**

0 → o → velocities

| 0 → o → |

Say given velocity distribution

![Velocity Distribution](image)

\[ f_v \]

\[ v_i \] vel

**Step 1**
Pick vel \( v_i \) for each particle \( i \)
Move the particle with this \( v_i \)
Now, weigh the particle with \( P(v_i) \)
from the PDF.

**Step 2**
Measure the location and update each particles probabilities

\[ \Psi_i = P(v_i) P(m_k | l_k) = \text{prob. of this location of the human} \]

\[ = \text{weight of this particle.} \]

\[ \Psi_i \]

| 0 → o → |

| 0 → o → |

| 0 → o → |

| 0 → o → |

| 0 → o → |

| 0 → o → |

| 0 → o → |

| 0 → o → |

| 0 → o → |

**Step 3**
Resample: Redistribute \( N \) particles in each of the \( N \) locations based on \( \Psi_i \), \( \forall i \in [1, N] \)
Step 4

Go to step 0

i.e., compute the transition to next step.
Kalman < Error Gaussian

Measurement is linear function

\[ m = f(\text{location}) \]

P.F. works well for multi-modal, e.g., ZEE

where particles disappear if gone through a wall.

HMM with 8 particles = P.F.

P.F. continuous space.