Chapter XIV - Probabilistic Reasoning in Time

14.1 Time and Uncertainty

Probabilistic reasoning is often concerned with how uncertainty evolves over time. Many real-world problems involve situations where the state of the world changes over time and where future states are uncertain, given the past.

Key Concepts:

- **Temporal Reasoning:** Temporal reasoning is the process of making inferences over time, often dealing with uncertainty about future states based on historical observations.
- **Dynamic Systems:** These systems change over time, and reasoning about such systems requires keeping track of how the system evolves under uncertainty.
- **Prediction and Estimation:** A central task in probabilistic reasoning in time is to predict future states or estimate the current state of the system, given past observations.

Example:

In weather forecasting, predicting tomorrow's weather involves considering the historical data of temperature, humidity, and pressure, while accounting for uncertainty in future conditions.

14.2 Inference in Temporal Models

Temporal models extend probabilistic reasoning to handle the evolution of uncertain information over time. In such models, the state at time t depends on the state at time $t-1$, and observations may also be dependent on past states.

Key Concepts:

- **Markov Assumption:** Temporal models often make a simplification known as the **Markov assumption**, which states that the future state depends only on the current state and not on the history of past states.
- **Dynamic Models:** Dynamic models describe how the system evolves over time, and they incorporate both the transition model (how states evolve) and the observation model (how observations are generated).
- **Temporal Inference:** The goal is to compute the posterior probability distribution over possible states at a particular time, given observations at that time and prior states.

Example:

In tracking the position of a moving vehicle, a temporal model would consider how the vehicle's position evolves over time and use sensor data (e.g., GPS) to update the belief about its current location.

14.3 Hidden Markov Models (HMMs)

A **Hidden Markov Model** (HMM) is a statistical model that represents systems that are assumed to follow the Markov process with unobserved (hidden) states. HMMs are widely used in time series data modeling, where the state of the system cannot be directly observed.

Key Concepts:

- **States:** The system is modeled as being in one of a finite set of hidden states at each time step.
- **Observations:** At each time step, an observation is made, which provides partial information about the state of the system.
- **Transition Probabilities:** The probability of transitioning from one state to another.
- **Emission Probabilities:** The probability of observing a certain observation given the current state.

Example:

In speech recognition, the sequence of spoken words is modeled as a sequence of states (e.g., phonemes), with the observed speech signal providing noisy observations of these states.

HMM Components:

1. **Initial State Distribution:** The probability distribution over the initial states.
2. **Transition Matrix:** The probability of moving from one state to another.
3. **Emission Matrix:** The probability of observing an observation given a state.
4. **Inference Problem:** Given the observed sequence, compute the most likely sequence of hidden states.

14.4 Kalman Filters

A **Kalman Filter** is an algorithm that provides estimates of the state of a linear dynamic system from noisy measurements. It is particularly useful when dealing with systems that are evolving over time, such as in tracking or navigation problems.

Key Concepts:

- **State Estimation:** The Kalman filter aims to estimate the true state of a system at each time step, given noisy observations.
- **Prediction and Update:** The Kalman filter operates in two steps: prediction (estimating the next state) and update (refining the estimate based on new observations).
- **Linear Systems:** Kalman filters are designed for systems that can be modeled with linear equations, with Gaussian noise in both the system model and the observations.

Example:

In GPS-based navigation systems, the Kalman filter can estimate the position and velocity of a vehicle, accounting for noisy GPS readings and the vehicle's motion dynamics.

Kalman Filter Equations:

1. **Prediction Step:**

   x^k=Ax^k−1+Buk−1\hat{x}$k = A\hat{x}${k-1} + Bu_{k-1}

   Where x^k\hat{x}_k is the predicted state at time kk, AA is the state transition matrix, and BB is the control input matrix.

2. **Update Step:**

   Kk=Pk−1HT(HPk−1HT+R)−1K_k = P_k^{-1} H^T (H P_k^{-1} H^T + R)^{-1}

   Where KkK_k is the Kalman gain, PkP_k is the state covariance matrix, and HH is the observation matrix.

14.5 Dynamic Bayesian Networks (DBNs)

Dynamic Bayesian Networks (DBNs) are extensions of Bayesian networks that model sequences of random variables, allowing for reasoning about temporal processes over time.

Key Concepts:

- **Dynamic Structure:** A DBN consists of a sequence of random variables that evolve over time. Each time step is modeled as a Bayesian network, and the structure captures the

dependencies between variables.

- **Temporal Dependencies:** The key advantage of DBNs is their ability to represent and reason about temporal dependencies, making them powerful for sequential decision-making problems.
- **Inference in DBNs:** Inference in DBNs typically involves computing the marginal likelihood of future states or updating beliefs about past states based on observations.

Example:

A DBN could be used to model the progression of a disease over time, where the health state of a patient at each time step depends on the previous state and the observed symptoms.

14.6 Keeping Track of Many Objects

When dealing with multiple objects in a dynamic environment, such as in tracking systems, it is necessary to reason about the states of many objects simultaneously. **Keeping track of many objects** involves managing the uncertainty associated with the positions, behaviors, or states of several objects over time.

Key Concepts:

- **Multiple Object Tracking (MOT):** Involves tracking several objects as they move or change over time, often using probabilistic models to handle the uncertainty in their positions and velocities.
- **Particle Filters:** Particle filters are a method used for tracking multiple objects, where each object is represented by a set of particles, each representing a possible state of the object.
- **Data Association:** In tracking multiple objects, it is important to correctly associate observations with the correct object, which may require solving a data association problem.

Example:

In autonomous vehicle navigation, tracking multiple pedestrians and vehicles in the environment can be modeled using particle filters, where each particle represents a possible position of the tracked object.

Exercises

1. **Hidden Markov Model Exercise:** Given a simple HMM with two hidden states (Rainy and Sunny) and observations (Wet and Dry), use the forward algorithm to compute the likelihood of a sequence of observations.
2. **Kalman Filter Exercise:** Implement a Kalman filter for a simple 1D tracking problem (e.g., tracking the position of a moving object) and estimate the object's position based on noisy observations.
3. **Dynamic Bayesian Network Exercise:** Create a DBN to model a robot's position over time, where each time step includes the robot's current position and sensor readings. Implement inference to predict the robot's future position based on past data.
4. **Multiple Object Tracking Exercise:** Given a set of noisy sensor readings for multiple objects in a 2D space, implement a particle filter to track the positions of these objects over time.