Notation Guide

Before we jump into Markov Chains and their role in AI agents, let's get comfortable with the notation we'll use. Think of these symbols as a language for describing how systems change, how agents perceive and act, and how we predict what's next. They're designed to be intuitive yet precise, connecting math to real-world ideas like weather shifts or game moves. Each symbol captures a piece of the puzzle—states, transitions, observations, and decisions—and we'll build on them throughout the chapter.

What's This All About?

At its core, this notation tracks a system's states (what's happening), how they shift over time or actions, what an agent sees, and how it decides. It's flexible enough to model a robot navigating a room or a player strategizing in a game. We'll start with Markov Chains—where states are all we need—and later hint at how this grows into hidden states, actions, and beliefs. Ready? Here's the lineup:

Core Symbols

- **s**: Hidden States
  The underlying "truths" or conditions of a system—like the weather (sunny or rainy) or a robot's location (room A or B). These are what we're tracking or guessing. In visuals, they're bold (s) to grab your eye as the foundation of everything.

- *o*: Observations
  The clues or sensory data we get about states—like seeing clouds (hinting at rain) or hearing a beep (suggesting a position). They're italicized (o) in text to stand apart from states, since they're what we perceive, not the full truth.

- a: Actions
  Choices an agent makes to influence the system—like turning left or flipping a switch. They're underlined (a) in examples to spotlight decisions that shape what happens next.

- $t(s,s')$: Transition Probability
  The chance of moving from state s to s'—think "what's the next step?" It's a number between 0 and 1 (e.g., 0.7 chance of rain after sun), capturing how states evolve. For Markov Chains, this is the star of the show.

- $t(s,s',a)$: Action-Driven Transition Probability
  How likely s' follows s when action a is taken—like "if I turn right, what's next?" It adds control to transitions, hinting at decision-making we'll see in MDPs.

- $e(o|s)$: Emission Probability
  The likelihood of observing o given state s—answering "what do I see if this is true?" For example, a 0.9 chance of clouds if it's raining. This previews HMMs, where states hide behind observations.

- $b(s)$: Belief Distribution
  The agent's best guess about s, based on what it's seen—like "I'm 80% sure it's raining." It's a probability spread over states, bridging perception to action, and nods to POMDPs.

- $r(s,a)$: Reward
  The payoff for being in s and taking a—think "was that a good move?" Maybe +5 points for a win. It's key for goal-driven agents, setting the stage for MDPs.

How We'll Use Them

In this chapter, Markov Chains lean on s and $t(s,s')$ to model state shifts—like a game board's changing positions. We'll hint at how $e(o|s)$ hides states in HMMs, $t(s,s',a)$ and $r(s,a)$ add decisions in MDPs, and $b(s)$ handles uncertainty in POMDPs. Each symbol builds intuition for agents interacting with environments.

Compared to Classical Notations

Our notation is custom but echoes classics:

- Sutton & Barto (MDPs): Uses S for states, $P(s'|s,a)$ for transitions, and $R(s,a)$ for rewards. We simplify with s, t, and r, making transitions mnemonic ("t" for transition) and states lowercase for readability.

- Rabiner (HMMs): Has A for transitions, B for emissions, and π for initial states. Our t and e are similar but unified across concepts, avoiding extra letters.

- Standard Probability: Often $P(s_{t+1}|s_t)$ for transitions—we condense to $t(s,s')$ for brevity and agent focus. Ours is streamlined for students, blending agent intuition with math, while staying flexible for visuals (s, o, a) and future chapters.

[Image Placeholder: Diagram of s and $t(s,s')$ in a simple system—add your sketch here!]
'''