14-0: **Rule Learning**

- Previously, we've assumed that background knowledge was given to us by experts.
  - Focused on how to use that knowledge.
- Today, we'll talk about how to acquire that knowledge from observation.
- Focus on learning propositional rules
  - $sunny \wedge warm \rightarrow PlayTennis$
  - $cool \wedge (rain \vee strongWind) \rightarrow \neg PlayTennis$

14-1: **Learning**

- What does it mean for an agent to learn?

14-2: **Learning**

- What does it mean for an agent to learn?
- Agent acquires new knowledge
- Agent changes its behavior
- Agent improves its performance measure on a given task

14-3: **Learning Agents**

- A learning agent has a *performance element* and a *learning element*.
  - The performance element is what an agent uses to decide what to do.
  - This is what we've studied up to now.
- The learning element is what allows the agent to modify the performance element.
  - This might mean adding or changing rules or facts, modifying a heuristic, changing a successor function
  - In order to modify its behavior, an agent needs information telling it how well it is performing.
  - This information is called *feedback*.

14-4: **Deduction vs. Induction**

- Up to now, we've looked at cases where our agent is given general knowledge and uses this to solve a particular problem.
  - Exactly two people like Homer, Suck always cleans a room, etc.
  - This general-to-specific reasoning is known as *deduction*.
  - Advantage: deduction is sound, assuming your knowledge is correct.

14-5: **Deduction vs. Induction**

- Sometimes, you may not have general information about a problem.
- Instead, you might have *data* about particular instances of a problem.
- The problem then is to figure out a general rule from specific data.
- This is called *induction* - most learning is an inductive process.

- Problem: induction is not sound.

14-6: **Example**

- Consider the problem of an agent deciding whether we should play tennis on a given day.
- There are four observable percepts:
  - Outlook (sunny, rainy, overcast)
  - Temperature (hot, mild, cool)
  - Humidity (high, low)
  - Wind (strong, weak)
- We don't have a model, but we do have some data about past decsions.
- Can we induce a general rule for when to play tennis?

14-7: **Types of Learning Tasks**

- There are essentially three categories of learning tasks, each of which provides different feedback.
- They vary in the amount of information that is available to our learning algorithm.
- Supervised learning.
  - In this case, an external source (often called a teacher) provides the agent with *labeled examples*
  - Agent sees specific actions/cases, along with their classification.
- D2 was Sunny, mild, high humidity and weak wind. We played tennis.

14-8: **Types of Learning Tasks**

- Unsupervised Learning
  - In this case, there is no teacher to provide examples.
  - The agent typically tries to find a "concept" or pattern in data.
  - Statistical methods such as clustering fall into this category
  - Our agent might be told that day1, day 4 and day 7 are similar and need to determine what characteristics make these days alike.

14-9: **Types of Learning Tasks**

- Reinforcement Learning
  - This is a particular version of learning in which the agent only receives a *reward* for taking an action.
  - May not know how optimal a reward is.
  - Will not know the "best" action to take
  - Our agent might be presented with a Sunny, Hot, Low humidity, Strong wind day and asked to choose whether to play tennis.
  - It chooses 'yes' and gets a reward of 0.3
  - Is 0.3 good or bad?

14-10: **Supervised Learning**

- Supervised learning is one of the most common forms of learning.

- Agent is presented with a set of labeled data and must use this data to determine more general rules.
- Examples:
  - List of patients and characteristics: what factors are correlated with cancer?
  - What factors make someone a credit risk?
  - What are the best questions for classifying animals?
  - Whose face is in this picture?
- This is the form of learning we will spend most of our time on.

14-11: **Classification**

- the particular learning problem we are focusing on is sometimes known as *classification*
  - For a given input, determine which class it belongs to.
- Programs that can perform this task are referred to as *classifiers*

14-12: **The Learning Problem**

- We can phrase the learning problem as that of estimating a function $f$ that tells us how to classify a set of inputs.
- An example is a set of inputs $x$ and the corresponding $f(x)$ - the class that $x$ belongs to.
  - $<< Overcast, Cool, Low, Weak >, playTennis >$
- We can define the learning task as follows:
  - Given a collection of examples of $f$, find a function $H$ that approximates $f$ for our examples.
  - $H$ is called a *hypothesis*.

14-13: **Induction**

- We would like $H$ to *generalize*
  - This means that $H$ will correctly classify unseen examples.
- If the hypothesis can correctly classify all of the training examples, we call it a *consistent* hypothesis.
- Goal: find a consistent hypothesis that also performs well on unseen examples.
- We can think of learning as search through a space of hypotheses.

14-14: **Inductive Bias**

- Notice that induction is not sound.
- In picking a hypothesis, we make an educated guess about how to classify unseen data.
- The way in which we make this guess is called a *bias*.
- All learning algorithms have a bias; identifying it can help you understand the sorts of errors it will make.
- Examples:
  - Occam's razor
  - Most specific hypothesis.
  - Most general hypothesis.
  - Linear function

14-15: **Observing Data**