Inquiry

- What might happen?
- What is most likely to happen?
- When is it hardest to plan precisely?

Reading: Epp, Ch. 9; handouts
6. Apply the basic notions of combinatorics and discrete probability, including by proof

Subtopic outcomes

6.1 Solve a problem in permutations and combinations*
6.2 Describe the relationship between combinatorics and intractable problems
6.3a Describe basic concepts of probability theory*
6.3b Prove a theorem in probability theory*
6.4 Describe and apply Bayes’ Theorem
6.5 Describe a computational application of probability theory
1. Combinatorics and counting

• What events are possible?
• How many arrangements exist of symbols or numbers?
• How are difficult problems solved?
• How is communication kept secure?

Possibility trees

• A series of events that each has a finite number \( n \) of alternative outcomes may be diagrammed by a possibility tree, which is \( n \)-ary

• *Theorem*: A series of \( k \) events, each with \( n \) possible outcomes, has \( n^k \) distinct paths from root to leaf of its possibility tree

• *Example*: a four-character PIN number with 36 possibilities for each character has \( 36^4 \) possible values
**State-space search**

- **State space**: A set of possible arrangements of values, e.g.:
  - Board configurations in board games
  - Paths in a graph
  - Arrangements of items in a knapsack
  - Assignments of truth values in a formula
- **Transitions** may be defined from state to state, which we can express as a *tree* of exponential size

**Multiplication rule**

- Suppose an operation has \( k \) steps
- Suppose steps 1 .. \( k \) have \( n_1, \ldots, n_k \) ways to be performed
- Then there are \( \prod_{i=1}^{k} (n_i) \) ways to perform the operation
- From this it follows that there are \( O(2^k) \) ways to perform it
6. Combinatorics and discrete probability

Pigeonhole principle

• (Intuition) If \( n \) pigeons enter \( m \) pigeon holes, and if \( n > m \), then at least one hole must have at least two pigeons
• (Formal) **Theorem:** If \( |A| > |B| \) then \( f : A \rightarrow B \) cannot be injective; i.e., \((\exists a, b \in A, a \neq b) f(a) = f(b)\)
• **Example:** at least two people in Framingham have the same last-four, because there are 10K last-4s and more than 10K persons in Framingham
• **Corollary:** Any function from an infinite set to a finite one is non-injective

Permutations and combinations

• **Permutations:** The possible orderings of elements of a set
• **Combinations:** The unordered subsets of a set
• Our interest is to *count* permutations and combinations in order to determine *possibilities*, so as to compute *probabilities*
Counting elements of disjoint sets

- For a finite set $A$ partitioned as $A_1, A_2, \ldots, A_k$, $|A| = \sum_{i=1}^{k} |A_i|$

- **Theorem**: for finite disjoint sets $A, B$, with $B \subseteq A$, $|A - B| = |A| - |B|$

- **Proof**:
  - if $B \subseteq A$, then $B \cap A - B = \emptyset$ (partition of $B$)
  - so $|B| + |A - B| = |A|$
  - hence $|A - B| = |A| - |B|$

- Also, $|A \cup B| = |A| + |B| - |A \cap B|$

Permutations

- **Definition**: Orderings of $n$ objects taken $k$ at a time, without repetition

- There are $(n!)$ permutations for $n$ objects

- **Example**: There are $5! = 120$ ways to order the letters A, B, C, D, E

- $k$-permutations ($P(n,k)$): Orderings of $n$ objects taken $k$ at a time

- There are $(n! / (n-k)!)$ $k$-permutations of $n$ objects

- **Example**: there are $P(6, 3) = 120$ different ways to throw a die such that only 1, 2, or 3 show
Combinations

- **Definition:** the number of ways to select from \( k \) objects at a time, taken from a set of \( n \) objects, without order or repetition
- \( C(n, k) = \frac{n!}{(n-k)! k!} = \frac{P(n, k)}{k!} \)
- **Example:** There are \( C(36, 6) \) ways to play the lottery where 6 numbers are chosen out of 36
- \( C(n, k) \) is also written \( \binom{n}{k} \) (“\( n \) choose \( k \)’’)
- Note that \( n! = n(n-1)(n-2) \times \ldots \times 2 \)

Binomial coefficients

- \( C(n, k) \) is also called a *binomial coefficient*, is computed by Pascal’s Triangle, and is defined by the following recurrence, called *Pascal’s formula*:
  \[
  C(n, k) = \begin{cases} 
  1 & \text{if } k = 0 \text{ or } k = n \\
  C(n-1, k-1) + C(n-1, k) & \text{otherwise}
  \end{cases}
  \]
- **Binomial theorem:** \((a + b)^n = \sum_{k=0}^{n} C(n, k) a^{n-k} b^k\)
Binomial theorem

- Gives an expression for \((a + b)^n\) for any \(a, b, n\)
- **Theorem:**
  \[(a + b)^n = \sum_{k=0}^{n} C(n, k) a^{n-k} b^k\]
- Hence the values \(C(n, k)\) are the **binomial coefficients**

Counting partitions

- **Stirling numbers** \(S_{n,r}\) denote the number of ways a set of size \(n\) may be partitioned into \(r\) subsets
- \(S_{n,r} = S_{n,r-1} + S_{n-1,r}\)
### Multisets

- A **multiset** is an \( r \)-combination with repetition allowed.
- **Theorem**: the number of multisets of size \( r \) from a set of \( n \) elements is \( C(r + n - 1, r) \).
- Summary of ways of choosing \( k \) elements out of \( n \) elements:

<table>
<thead>
<tr>
<th></th>
<th>ordered</th>
<th>unordered</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repetition allowed</td>
<td>( n^k )</td>
<td>( C(k + n - 1, k) )</td>
</tr>
<tr>
<td>No repetition</td>
<td>( P(n, k) )</td>
<td>( C(n, k) )</td>
</tr>
</tbody>
</table>

### Cryptography and modular arithmetic

- RSA cryptography uses two keys: **public** (for encryption) and **private** (for decryption).
- The encryption and decryption algorithms use modular arithmetic on very large numbers, including properties of exponents and other number-theoretic theorems.
- Important theorem: \( x^{a+b} = x^a x^b \).
- Public key includes a value \( pq \) and a value \( e \); decrypting requires private key, which includes prime values \( p, q \), very hard to obtain from \( pq \).
Subtopic outcome

6.1 Solve a problem in permutations and combinations*

2. Intractability

- What is a hard problem?
- What is complexity?
- Is there an algorithm that solves SAT in reasonable time?
- Are there solvable problems that are not worth the trouble to solve?
Combinatorial explosion

• Many computational problems are subject to *combinatorial explosion*

• For most state-space-search problems, the number of possible solutions is exponential in problem size

• For some problems, typical for AI, no easy way exists to prune the number of candidate solutions that require examination

• These problems are called *intractable*

Constraint and optimization problems

• *Constraint problem*: To find some value that satisfies a set of constraints or conditions

• *Constraint problem examples:*
  – Search
  – Pattern matching
  – Sort of a set of records

• *Optimization problem*: to find maximum or minimum valued solution to a constraint problem, among all solutions

• Many optimization problems are intractable
Constraint satisfaction problems

- A set $X$ of variables $x_1, x_2, ..., x_n$
- A set $C$ of constraints $C_1, C_2, ..., C_m$, that each specifies acceptable combinations of values for a certain subset of $x$
- A variable assignment to $X$ that does not violate any of $C$ is consistent or legal
- A variable assignment to all of $X$ is complete
- Desired problem solution: a complete consistent variable assignment

Satisfiability problem (SAT)

- Given a formula $\phi$ in propositional logic, does a set of variable assignments exist that satisfies $\phi$ (makes $\phi$ true)?
- Examples:
  (a) $p \land q \land r$
  (b) $(p \land q) \lor \neg q$
  (c) $p \land \neg(q \lor \neg q)$
- How large a truth table would be required?
- SAT is the set of formulas that are satisfiable
Set partition

- Given a finite set \( S \subseteq \mathbb{N} \), can \( S \) be partitioned into two sets, \( A \) and \( B \), s.t. \( \sum_{x \in A} x = \sum_{y \in B} y \)?
- How many subsets does \( S \) have, hence how many partitions?
- *This problem, like SAT, has a brute-force \( O(2^n) \) solution*

Hamiltonian-cycle problem

- *Problem:* Given a graph \( G = \langle V, E \rangle \), is there a path in \( V^* \) that starts and ends with the same vertex and passes through every vertex exactly once?

- *One solution* is to generate and test every possible sequence of \( |V| + 1 \) vertices to see if it is a path
  - Complexity of the test operation: \( O(n) \)
  - Complexity of generate-and-test: \( O(2^n) \)
Observations about these problems

- Each (SAT, set partition, Hamiltonian cycle) seems to require generating and checking a very large set of candidate solutions.
- Checking a given candidate solution is easy enough.
- But the number of candidates is $O(2^n)$.
- For $n > 100$, the time required could be too much for practical purposes.

Complexity and intractability

- *Complexity of a problem* is the time function of the fastest algorithm that solves the problem.
- *Example*: No search algorithm is better than $O(n)$ for arbitrary collections, so the complexity of the search problem is $\Omega(n)$.
- Some problems are *decidable* but take too long to solve in practice.
- Those *intractable* problems may be characterized mathematically.
Intractable problems

- Some problems have no known polynomial time \((O(n^k))\) solutions for any constant \(k\).
- These are considered intractable because for sufficient \(n\), they “might as well take forever”.
- Exponential-time examples: Hanoi, password guessing, understanding English.
- Others are called \(NP\)-complete problems: solutions are checkable, but not known to be obtainable, in polynomial time.

Complexity classes of problems

- Definition: \(TIME(f(n))\) is the set of problems/languages with solution (acceptance algorithms) of time \(O(f(n))\) for input size \(n\).
- Examples:
  - Sorting is in \(TIME(n \lg n)\) and in \(TIME(n^2)\), because \(O(n \lg n)\) sorts exist.
  - Search for shortest path in a graph is in \(TIME(n^2)\) – Dijkstra algorithm.
  - SAT is in \(TIME(2^n)\).
  - Searching a balanced BST is in \(TIME(\lg n)\).
6. Combinatorics and discrete probability

Polynomial time complexity

- $P = \bigcup_{k \in \mathbb{N}} \text{TIME}(n^k)$
- That is, $P$ is the set of problems decidable in $O(n^k)$ time (polynomial time), where $n$ is the size of the problem and $k$ is a constant

- Examples of problems in class $P$:
  - Searching a collection
  - Sorting an array
  - Generating a graph reachability matrix

- It is not known whether SAT is in $P$, but no one knows a $P$-time algorithm to solve it

Exponential time

- $EXP = \text{EXPTIME} = \text{TIME}(2^n)$

- $SAT$ seems to be in $\text{EXPTIME}$ but not in $P$, because it seems that to solve $SAT$, $O(2^n)$ candidate solutions need to be generated and tested

- Problems that are in $EXP$ but not in $P$ are considered intractable
Reducibility

- Problem B is *reducible* to problem A iff a solution to A enables a solution to B
- Intuitively, B is at least as hard as A
- Example: multiplication is reducible to addition
- If B is *undecidable* and B is reducible to A, then A is undecidable
- By showing that the SAT is reducible to problem P, we can show that P is intractable too

NP-completeness

- Problems with solutions that can be checked in P time, but at least as hard as SAT, are called *NP-complete*
- Example: Traveling Salesperson
- If *any* of these has a polynomial-time solution, then *all* do
- Most researchers think NP-complete problems to have no polynomial-time solutions
**Intractability, NPC, and EXPTIME**

- Certain problems are thought or known to have only exponential-time, \( O(2^n) \), solutions
- \((EXPTIME – P)\) and NPC are problems considered intractable
- Problems of planning, scheduling, routing, drawing inferences, understanding language, etc., are in general intractable
- *What to do:* Replace intractable problem with a simpler one, e.g., one with a probabilistic or approximate solution

**Summary**

- Applying the multiplication rule with \( n \) factors generates *combinatorial explosion*
- So the number of steps in exhaustive state-space search is often exponential in the problem size
- *Intractable problems* are ones that are of exponential complexity or to which SAT is reducible
6.2 Describe the relationship between combinatorics and intractable problems

3. Discrete probability

- What will happen in the future?
- What do we expect to happen?
- Can we precisely predict uncertain events?
- How strongly do we believe a claim, when we’re not sure?
Random processes

- *Discrete random process*: one whose outcome is from a set of discrete possibilities that are not predictable with certainty
- *Examples*: tossing a coin, playing lottery, or rolling dice are random processes
- *Probability theory* expresses uncertainty as belief in assertions, quantified in [0..1]

Basic concepts

- *Sample space*: All possible results of an experiment
- *Discrete probability* assumes finite sample space
- *Event*: one particular result; a subset of the sample space
- For event $E$ in sample space $S$,
  \[ P(E) = \frac{|E|}{|S|} \]
Uniform probability space

• A probability space \( S \) is a set of possible outcomes of an experiment

• Example: \( S \) for a die throw is \{1, 2, 3, 4, 5, 6\}

• Let \( |S| = n \) for probability space \( S \)

• Theorem: uniform probability function \( P : S \to \mathbb{R} \) is defined \( P(x) = (1/n) \) for any \( x \) in \( S \)

• Example: Using fair die, \( P(3) = 1/6 \), because there are 6 possible events, all equally likely

More concepts

• Compound event: sets of events

• Probability of an event: the ratio of event’s number of outcomes in the to the size of the sample space; \( 0 \leq P(x) \leq 1 \)

• Probability measure: “A mathematical measure on a probability space that can take on values between 0 and 1, with 0 corresponding to the empty set and 1 to the entire space.”
Law of large numbers

• “A theorem that describes the result of performing the same experiment a large number of times.
• “According to the law, the average of the results obtained from a large number of trials should be close to the expected value, and will tend to become closer as more trials are performed.”

Kolmogorov’s axioms

For sample space $S$ and events $A, B \subseteq S$,

1. $(\forall A) \ 0 \leq P(A) \leq 1$
2. $P(S) = 1, \ P(\emptyset) = 0$
3. $P(A \cup B) = P(A) + P(B) - P(A \cap B)$

• *Useful*: it is irrational to have beliefs that violate the axioms because they will result in poor bets
• *Theorems* that follow: $P(A \cup A^c) = 1$; $P(A^c) = 1 - P(A)$
\[ P(A \cup B) = P(A) + P(B) - P(A \cap B) \]

**Intuition:**
- \(P(A \cup B)\) is probability of shaded area at top right
- \(P(A \cap B)\) is probability of shaded area at center right
- \(P(A) - P(A \cap B)\) is shown bottom right
- Clearly, the sum of \(P(A)\) and \(P(B)\), minus the intersection’s probability, is \(P(A \cup B)\)

**Theorems**

Let \(S\) be a finite probability space, with events \(A, B \subseteq S\)

1. **Monotonicity:** \(A \subseteq B \rightarrow P(A) \leq P(B)\)
   
   *Example:* In cards, \(P(\text{Heart}) \leq P(\text{Red})\)

2. **Complementary event:**
   
   \(P(\neg A) = 1 - P(A)\)
   
   *Example:* \(P(\spadesuit) = 1 - P(\heartsuit, \diamondsuit, \clubsuit)\) in cards
   (because \(\frac{1}{4} = 1 - \frac{3}{4}\))
Probability of events

• For event $E$, $P(E) =$
  (number of ways $E$ can occur) ÷
  (# possible outcomes)

• Example: Probability of rolling 9 with two dice is
  $P((3, 6) \lor (4, 5) \lor (5, 4) \lor (6, 3)) ÷$
  $P((1, 1) \lor (1, 2) \lor \ldots \lor (6, 6))$
  $= 4 / 36 = 1/9$

Partial additivity

• Theorem: $P(A \cup B) = P(A) + P(B) - P(A \cap B)$

• Example:
  $P(\heartsuit \cup Queen) = \frac{1}{4} + \frac{1}{13} - \frac{1}{52}$

• The probability of $A$ or $B$ happening is
  the sum of the probabilities of each,
  discounting the chance of both)
Independent events

- *Intuition:* Independent events can have no effect on each other or overlap with each other
- *Formally:* Events $A$ and $B$ are independent iff $P(A \cap B) = P(A) \cdot P(B)$
- Single coin tosses and die rolls are independent
- *Example:* For draw of cards, $P(\heartsuit)$ is independent of $P(J \text{ or } Q \text{ or } K)$
- For non-independent events, notion of conditional probability is used, i.e., probability of $E_1$ given $E_2$

Mutual independence of three events

- Events $A$, $B$, $C$ are independent iff
  - They are pairwise independent
  - $P(A \cap B \cap C) = P(A) \cdot P(B) \cdot P(C)$
- Events $A_1$, $A_2$, ..., $A_n$ are mutually independent iff for any subset $B$ of $A_1$, $A_2$, ..., $A_n$,
  $$P(B_1 \cap B_2 \cap \ldots \cap B_m) = P(B_1) \cdot P(B_2) \cdots P(B_m)$$
Additivity under disjunction

Theorem:

• Let $S$ be a finite probability space, $A, B \subseteq S$

• Suppose $A, B$ are disjoint events, i.e., $A \cap B = \emptyset$

• Then
  
  $P(A \cup B) = P(A) + P(B)$

Example: $P(\heartsuit \text{ or } \diamondsuit) = P(\heartsuit) + P(\diamondsuit) = \frac{1}{2}$

Expected values

• For $n$ equally likely outcomes of a random process, where $a_k$ is the value of the $k$th outcome, the expected value of the process is
  
  $\sum_{k=1}^{n} a_k p_k$, where $p_k$ is the probability of $a_k$

• Examples:
  
  – In coin toss, expected value is $(0.5)$
    (heads $= 1$, tails $= 0$
  – In die throw, expected outcome is
    $(1 + 2 + 3 + 4 + 5 + 6) / 6 = 3.5$
  – Expected time for linear search is $(n / 2)$
**The binomial distribution**

- Probability of $k$ successes (the binomial distribution function):
  \[ B(n, k; p) = C(n, k) p^k (1 - p)^{n-k} \]
- *Example*: The probability of having 4 or more heads in 6 coin throws is
  \[ \sum_{k=4}^{6} B(6, 4; 0.5) \]

---

**Discrete random variables**

- *Definition*: A function from a finite sample space to a finite set of outcomes
- *Example*:
  - Let random variable $\chi$ (“Chi”) be the sum of scores for two dice.
  - Then $\chi$ takes the value 1 in no case, 2 in 1 case, 3 in 2 cases \{ (1,2), (2,1) \}, etc.
Random variables

• **Definition:** A random variable is a function \( f : S \rightarrow 2^S \) that gives the number of ways each outcome in the sample space \( S \) can occur

• A random variable helps describe the likelihood of outcomes

• **Kinds:** Boolean, discrete, continuous

• **Example:** random variable for throw of two dice:

\[
\begin{array}{cccccccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 5 & 4 & 3 & 2 & 1
\end{array}
\]

Random distributions

• Probability that a random variable takes a given value is the probability of the set of outcomes where that holds,

\[
P(\chi = k) = P(\{ s \in S \mid \chi(s) = k \})
\]

• Probability distribution function, \( f_\chi(x) \), maps from outcomes to their probabilities

• Examples:
  - Uniform distribution (flat graph)
  - Gaussian distribution (“normal curve”)

\[
\begin{array}{cccccccccccc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 5 & 4 & 3 & 2 & 1
\end{array}
\]
### Prior probability

- **Prior (unconditional) probability** $P(\alpha)$: degree of belief in the absence of other information
- **Joint probability distribution**: grid of probabilities of all combinations chosen from sets of random variables, e.g., weather, traffic
- **Probability density function**: probability distribution of a continuous variable

### Conditional probability

- $P(A \mid B) = P(A \cap B) \div P(B)$
- **Interpretation**: The probability of event $A$, given event $B$, is the probability that both will occur, divided by the probability of $B$
- **Example**: Given that the first of two coin tosses is heads, what’s the chance of two heads?
  $P(c_1 = c_2 = H \mid c_1 = H) = P(H-H \text{ and } c_1 = H) / P(c_1 = H) = \frac{1}{4} / \frac{1}{2} = \frac{1}{2}$
- $P(A \cap B) = P(A \mid B) \times P(B)$
- $P(B) = P(A \cap B) \div P(A \mid B)$
Subtopic outcomes

6.3a Describe basic concepts of probability theory*

6.3b Prove a theorem in probability theory*

4. Bayes’ Theorem

• What is evidence good for?
• What are techniques to put evidence to use?
Inverse probability

• *Intuition:* given some knowledge of an object, and some statistics about the population containing the object, what else can we surmise about the object?

• *Example:* Suppose we know 2/3 of the numbered cards in a pile are red, and ¼ of the JQK cards are red, and ¾ of all the cards are JQK. If a card randomly drawn is red, then by inverse probability we can calculate the probability that it is a J, Q, or K.

Bayes’ Theorem

• Established law applying inverse probability

• By Thomas Bayes, pub. 1763

• Helps relate cause and effect by showing how we can learn probability of causes by understanding an effect

• Let $H$ be a set of hypotheses $h_1, h_2, ...$, explaining evidence $E$

• *Theorem:* $P(h_i \mid E) = P(E \mid h_i) P(h_i) \div P(E)$
Applications of Bayes’ Theorem

- By the theorem, some medical screening tests may be useful but more accurate results may be needed to diagnose a disease, because such tests may yield false positives or negatives

- *Example:* Suppose 0.5% of people have a disease, and a test has false positive rate of 3% and false negative rate of 1%

- Then under Bayes’ theorem, 99.995% of negative results are correct, but only 14% of persons with positive results actually have the disease

A traffic scenario

- Bayesian networks reflect *multiple causalities*

- *Example:* Why is traffic heavy, given evidence of orange barrels or flashing lights?

- Accidents cause heavy traffic and cause emergency vehicles to arrive; these vehicles cause flashing lights

- Construction causes heavy traffic and causes orange barrels to be placed

- *Evidence* is traffic, barrels and/or flashing lights; *cause* is accident or construction
Bayesian net for traffic problem

- The unlabeled BBN below reflects causal relations

![Bayesian network diagram]

- To explain traffic slowdown as due to construction or accident, use evidence of orange barrels (B), bad traffic (T), and flashing lights (L)

Example of Bayesian inference

- Suppose we know the following a priori:

<table>
<thead>
<tr>
<th>Constr</th>
<th>Traffic</th>
<th>Probability</th>
</tr>
</thead>
<tbody>
<tr>
<td>t</td>
<td>t</td>
<td>.3</td>
</tr>
<tr>
<td>t</td>
<td>f</td>
<td>.2</td>
</tr>
<tr>
<td>f</td>
<td>t</td>
<td>.1</td>
</tr>
<tr>
<td>f</td>
<td>f</td>
<td>.4</td>
</tr>
</tbody>
</table>

- Then \( P(C \mid T) = P(C \land T) \div P(C \land T) + P(\neg C \land T) \)
  \[ = .3 \div (.3 + .1) = 0.75 \]

- Adding orange-barrels evidence, if available, will increase likelihood of the explanation that construction is the cause
Speech recognition

- Bayesian probabilistic inference is used
- Where *words* is a sequence of words, *signal* is sound, we want to maximize \( P(\text{words} \mid \text{signal}) = \alpha P(\text{signal} \mid \text{words}) P(\text{words}) \)
- *Acoustic model*: \( P(\text{signal} \mid \text{words}) \)
- *Language model*: \( P(\text{words}) \)
- HMM for toe-mah-toe | toe-may-toe: [pic 9]

Probabilities of success

- Assume *n* trials in an experiment where the result may be success or failure, and the probability of success in each trial is *p*
- Each way of having *k* successes has probability \( p^k(1 - p)^{n-k} \)
- There are \( C(n, k) \) ways to have *k* successes
6.4 Describe and apply Bayes’ Theorem

5. Computational applications

- What’s the right action in an uncertain situation?
- Will it rain?
- Is belief worthwhile?
Cases

- Hashing; randomized algorithms for video games
- Average-case analysis of algorithms
- Planning under uncertainty in AI
- *Games*: Making play more realistic by generating unpredictable events
- Evolutionary computation
- Bounded rationality, Markov processes, modal logic

Application of stochastic methods

Some applications:

- *Diagnostic reasoning*, because cause-effect relationship is not always obvious
- *Natural language processing*, because semantics are fuzzy or ambiguous
- *Planning*, because of uncertainty of future events and cause-effect relationships
- *Learning*, because conclusions to draw from experience are ambiguous and probabilistic
Modal logic

- Modal operators reflect belief:
- **Unless**: \( p(x) \rightarrow q(x) \) unless \( r(x) \)
- Operator \( Ab \): \( p(x) \) unless \( ab \ p(x) \rightarrow q(x) \), where \( ab \) refers to abnormal instance of \( p \), e.g., bird with broken wing
- Operator \( M \): \( (\forall x) \) good-student\((x) \land M \) study-hard\((x) \rightarrow graduates(x) \), where \( M \) means “is consistent with”, i.e., the fact that \( x \) studies hard is consistent with what we know

Approximation and randomized algorithms

Why compute an approximation rather than an exact result?

- Exact solution algorithm may take too much time (*Example*: shortest tour through \( n \) cities)
- Exact results may have infinitely large representation (*Example*: square root)
- Approximate solution may be used as part of a larger exact algorithm
Acting under uncertainty

- *Rational decisions* under uncertain information depend on
  - Relative importance of multiple goals
  - Probabilities of achieving goals by alternative actions
- *Diagnosis*: knowledge only provides a *degree of belief* in [0..1)
- Degree of belief is expressed using probability theory

Bounded rationality

- Notion suggested by Herbert Simon, 1972, as alternative to classical rationality assumption of economic theory
- *Argument*: Humans have limited knowledge and resources for decision making
- Alternative goal to optimality: *satisficing* (good enough)
- *Rational agent*: one that chooses actions that yield maximum expected utility averaged over all outcomes
Markov chains

- Probability of being in a given state at a given time is dependent on state at previous times
- *First-order Markov chain* is one where probability of present state depends *only* on previous state
- *Example*: weather at any location

Example: Weather

- Let states be \{sunny, cloudy, rainy\}
- Let transitions be as follows:
  
  \[
  \begin{array}{ccc}
  \text{sun} & \text{cloud} & \text{rain} \\
  \text{sunny} & .4 & .5 & .1 \\
  \text{cloudy} & .2 & .5 & .3 \\
  \text{rainy} & .1 & .3 & .6 \\
  \end{array}
  \]

- First-order Markov model (right):
Querying a Markov model

- **Example problem:** If it’s rainy today, what is the probability that it will be rainy two days from now?
- **Solution:** Following the Markov model on previous slide, find
  
  \[ P(r, r, r) + P(r, s, r) + P(r, c, r) \]
  
  \[ = (0.6)(0.6) + (0.1)(0.1) + (0.3)(0.3) = 0.46 \]

Markov decision processes

- Defined by initial state \( s_0 \), transition model \( T(s, a, s') \), and reward function \( R(s) \)
- A solution specifies a *policy* \( \pi(s) \): what agent should do given any state of environment
- Policies have *expected utilities*: utility of possible environment histories generated by it
- Optimal (maximal-utility) policy is sought
- Future rewards may be discounted in deciding expected utility
Evolutionary computation

- State space is explored comparing a population of states rather than one at a time
- Populations of solutions to a problem are generated starting with a random guess
- Fitness tests are applied
- A new generation is chosen and altered using random mutation and crossover
- The process iterates, yielding better and better solutions

Subtopic outcome

6.5 Describe a computational application of probability theory
References

