1 Search

Parts of a search problem: start state, transition model, goal test, path cost

What search algorithms have in common: can be thought of as building trees

1. Root node is annotated with start state. Initial fringe is root node.
2. Successively expand nodes (pop them off the fringe, adding children to fringe).
3. When you expand (not enqueue!) a goal node, you’re done.

Dimensions of variation

1. Whether you remember states you’ve visited before
2. How you decide which node to expand
   • Uninformed: breadth-first, depth-first, uniform cost, iterative deepening
   • Informed: you have a function $f(n)$ defined for all nodes of search tree
     - Expand lowest $f(n)$
     * $A^*$: $f(n) = g(n) + h(n)$ ($h$ estimates forward cost; $h$ only a function of state!)
     * Greedy (sometimes also called best-first): $f(n) = h(n)$ ($h$ as above)
     - Beam search (like greedy, but prune paths)

Admissibility: $h(n)$ never more than true best cost from $n$. Implies $A^*$ is optimal (but remember caveat about graph search).

Consistency: for all edges $n \rightarrow p$ in search tree, $h(n) \leq cost(n, p) + h(p)$. Implies admissibility. Also implies $A^*$ visits nodes in order of increasing $f$.

Constructing heuristics: bigger is better; try true cost for a relaxed problem; try max of multiple heuristics.

Local Search: hill-climbing (gets stuck on local minima, ridges), random restarts, stochastic beam search, simulated annealing (Monte Carlo descent: $p \propto e^{-\Delta h/T}$), genetic algorithms

2 Constraint Satisfaction Problems

Parts of a CSP: variables $X_i$, domains $D_i$, constraints $C_i$ dealing with subsets of $\{X_i\}$

Concepts: CSPs as search, incremental vs. complete-state formulations, constraint graphs

Fail-fast/minimum remaining values (MRV) heuristic: assign the variable with the smallest number of values it could take on (while preserving consistency). Degree heuristic.

Least-constraining value heuristic: assign a value to $X_i$ that minimizes some measure of how constrained the problem is. Example: maximize minimum over neighbors $V$ of $X_i$ of $|\text{consistent values for } V|$.
Forward checking: prune domains when a variable is assigned

Arc consistency on \((X,Y)\): prune domain of \(X\) (initially and/or anytime domain of \(Y\) is reduced)

\(k\)-consistency of a CSP: given any consistent assignment to a subset (of size \(k-1\)) of variables, it is possible to consistently assign any \(k\)th variable. 2-consistency is arc-consistency. 3-consistency is called path-consistency. A CSP is strongly \(k\)-consistent if it is \(j\)-consistent for \(j\) from 1 to \(k\).

Cycle cutset: a set of nodes which, if removed, make a graph into a tree

Methods exploiting structure of constraint graphs
Constraint graphs that are trees can be solved in \(O(n)\) time. In cutset conditioning, we loop over consistent assignments to a cutset, solving the tree CSP inside the loop. In tree decomposition, we build a junction tree that represents the constraints and solve the resulting tree CSP.

3 Games

Definition of a game: initial state, successor, terminal test, utility function on terminal states.

Zero-sum: the sum of the utilities for each players is 0 in all outcomes.

Concepts: game tree, strategy, minimax value of a node in a game tree, ply, expectiminimax.

If the players play perfectly, the minimax value of a game tree node will be the ultimate outcome of a game that reaches that node.

Evaluation function: a function which estimates actual node minimax values to cut down on search.

Alpha-beta pruning is a method of speeding up minimax search. \(\alpha\) and \(\beta\) are values which are computed after the minimax value of any node \(n\) is found.

\(\alpha\) is the maximum of (previously computed) minimax values at nodes \(u\) such that (1) the parent of \(u\) represents a choice by the player who prefers high outcomes (2) the parent of \(u\) is on the path from the root of the tree to \(n\).

\(\beta\) is the minimum of (previously computed) minimax values at nodes \(u\) such that (1) the parent of \(u\) represents a choice by the player who prefers low outcomes (2) the parent of \(u\) is on the path from the root of the tree to \(n\).

If \(\alpha > \beta\), we prune the parent of \(n\). Note that \(\alpha\) is always increasing when we proceed down a path of the game tree, and \(\beta\) is always decreasing when we proceed down a path.

4 Logic

Parts of a logic: syntax (a definition of well-formed sentences), semantics (a way to establish the truth of a given sentence in a given world).

Entailment: \(\alpha \models \beta\) means that in all worlds where \(\alpha\) is true, \(\beta\) is also true. Decidable in propositional logic, semi-decidable in first-order logic.

Logical equivalence: \(\alpha \equiv \beta\) means that \(\alpha\) entails \(\beta\) and vice versa.

Inference: the process of generating new sentences from old ones. This is done by a set of inference rules such as \(\frac{\alpha}{\beta}\), which says that if you have determined \(\alpha\) and \(\beta\) to be true, you may conclude that \(\gamma\) is true. It is desirable for inference rules to be sound (they never generate false sentences from true premises) and complete (they are capable of generating all true sentences). There are sound and complete sets of inference rules for propositional and first-order logic, but not for more powerful logics (Godel’s incompleteness theorem). Sometimes people write \(\alpha \vdash \beta\) to indicate that \(\beta\) may be derived from \(\alpha\) using a given set of inference rules.
Propositional logic: sentences are formed from True, False, symbols, ¬, ∧, ∨, ⇒, and ⇔.

Satisfiability of \( \alpha \): \( \alpha \) is true in some model. Validity of \( \alpha \): \( \alpha \) is true in all models.

\( KB \models \alpha \) iff \( KB \models \alpha \) is valid

CNF (conjunctive normal form): conjunction of disjunction of literals (e.g. \((A \lor B \lor C) \land (D \lor E \lor F) \land \ldots\)).

Resolution: a procedure for determining satisfiability. May be used to decide entailment in first-order logic.

Syntax of first order logic:

- Things which form sentences: connectives, quantifiers, predicates (sometimes including equality) applied to terms
- Terms: constants, variables, functions applied to terms

DeMorgan’s rules: For sentences \( \alpha \) and \( \beta \), \(- (\alpha \lor \beta) \equiv \neg \alpha \land \neg \beta\) and \(- (\alpha \land \beta) \equiv \neg \alpha \lor \neg \beta\).

5 Probability

Probabilistic models always involve a sample space \( \Omega \), which can be thought of as the set of all possible outcomes of some experiment. The model also specifies a function \( P(\omega) \), which gives the probability of each outcome \( \omega \) in \( \Omega \). \( \sum_{\omega \in \Omega} P(\omega) \), the probability of the outcome being any element of \( \Omega \), must be 1.

Random variables are just functions which assign a value to each outcome in a sample space. They are usually written with capital letters and can be thought of as measurements of outcomes. For example, if the sample space is the set of all people, one could define a random variable \( H \) which measures height. If \( k \) is a person, \( H(k) \) is his/her height.

An event is a collection of outcomes of an experiment which have some property in common. For example, if the sample space is the set of people, \( F \) might be the set of females. Randomly picking a person may or may not result in the event that they’re female. We write \( P(F) \) for the probability that the event occurs.

If \( A \) and \( B \) are events, we can define new events \( A \lor B \), \( A \land B \), and \( \bar{A} \). \( P(A, B) \) always means \( P(A \lor B) \).

The conditional probability of event \( A \) occurring given that event \( B \) occurs, written \( P(A | B) \), is

\[
P(A | B) = \frac{P(A \land B)}{P(B)}.
\]

If \( X \) is a random variable and \( x \) a value, consider the event that the measurement \( X \), performed on an outcome of a random experiment, equals \( x \). This event is written \( X = x \). We can write \( P(X = x) \) for the probability that \( X = x \) occurs. Often people abbreviate this to \( P(X) \). \( P(X = x) \), viewed as a function of \( x \), is called the distribution of \( X \).

A joint distribution of multiple random variables is a function giving the probability that they take on particular combinations of values. Example: \( P(X = x, Y = y) \).

A conditional distribution is a function giving a conditional probability where the events are equality tests of random variables. Example: \( P(X = x | Y = y) \).

Often people will define a probabilistic model by simply listing some random variables and providing a full joint distribution over all of them. (What are the outcomes and the function \( P(\omega) \) in that case?) To calculate smaller joint distributions it then becomes necessary to marginalize over unused variables. For

---

1 This is technically a lie. \( P \) is usually defined as a function whose domain is a subset of the powerset of \( \Omega \), and where \( P(A) \) represents the probability that the outcome falls in \( A \). The probabilistic model is a measure space with measure \( P \). For a discrete \( \Omega \), this technicality may be ignored. We assume a discrete \( \Omega \) for the rest of this introduction, though all the results generalize.

2 Be sure you understand the difference between \( X \) and \( x \): \( X \) is a name for something about outcomes that we can measure. \( x \) is a variable which ranges over possible results of that measurement.
example, if the full joint is $P(X = x, Y = y, Z = z)$, then we can compute a smaller joint distribution $P(X = x, Y = y)$ as follows:

$$P(X = x, Y = y) = \sum_z P(X = x, Y = y, Z = z).$$ (2)

People call such a distribution a **marginal** distribution to emphasize that it was computed by marginalization.\(^3\)

The words *marginal*, *conditional*, and *joint* don’t refer to disjoint categories. $P(X, Y | Z)$ could be the joint probability of $X$ and $Y$, conditioned on $Z$, marginalized over some other variable $W$.

**Chain rule:** $P(A, B) = P(A|B)P(B)$

**Bayes’ rule:** $P(A|B) = \frac{P(A|B)P(B)}{P(B)}$. For random variables, we can equivalently write $P(X|Y) = \frac{P(Y|X)P(X)}{P(Y)}$.

The typical application of this rule is estimating a random variable $X$ which we can’t observe when we believe it has a causal effect on $Y$ and we can observe $Y$. In that case $P(Y|X)$ is sometimes referred to as the **likelihood** (of the observed data given various possibilities for the unobserved variable), $P(X)$ is referred to as the **prior probability distribution** of $X$, and $P(Y)$ is a normalization constant. $P(X|Y)$ is referred to as the **posterior probability** of $X$ (i.e. our belief about $X$ after we’ve observed its influence on $Y$).

Events $A$ and $B$ are called **independent** if $P(A, B) = P(A)P(B)$. This is equivalent to $P(A|B) = P(A)$.

Random variables $X$ and $Y$ are called **independent** if $P(X, Y) = P(X)P(Y)$. This is equivalent to $P(X|Y) = P(X)$.

**Naive Bayes model:** a model with one “cause” variable $C$ and a number of “evidence” variables $E_i$. $P(C, E_1, E_2, \ldots)$ is assumed to be $P(C) \prod_i P(E_i|C)$. Usually we observe the $E_i$ and want to guess the value of $C$.

When a probabilistic model for a variable $X$ has parameters $\theta$, we sometimes write $P(X; \theta = a)$ for the probability distribution of $X$ when $\theta$ takes on a particular value $a$.

Often we want to estimate $\theta$ given some values of $X$ (call them $x_1, x_2, \ldots, x_n$) sampled from the distribution. The **maximum likelihood estimate** (MLE) for $\theta$ is $\arg\max_{\theta} \prod_i P(X = x_i; \theta)$.

\(^3\)The word *marginalization* derives from the pre-spreadsheet accountant’s practice of writing sums of columns or rows of a table in the margin.