Introduction to Artificial Intelligence
Learning Goals

Alice Gao

November 3, 2021

1 Introduction to Artificial Intelligence

- Describe each of the four definitions of AI.
- Compare and contrast the four definitions of AI.
- Give a few reasons why we chose the Rational Agent definition rather than the other three definitions.

2 Search

2.1 Uninformed Search

- Formulate a real world problem as a search problem.
- Trace the execution of and implement uninformed search algorithms (Breadth-first search, Depth-first search, Iterative-deepening search).
- Given an uninformed search algorithm, explain its space complexity, time complexity, and whether it has any guarantees on the quality of the solution found.
- Given a scenario, explain whether and why it is appropriate to use an uninformed algorithm.

2.2 Heuristic Search

- Describe motivations for applying heuristic search algorithms.
- Trace the execution of and implement the Lowest-cost-first search, Greedy best-first search and A* search algorithm.
- Describe properties of the Lowest-cost-first, Greedy best-first and A* search algorithms.
- Design an admissible heuristic function for a search problem. Describe strategies for choosing among multiple heuristic functions.
- Describes strategies for pruning a search space.

2.3 CSP

- Formulate a real-world problem as a constraint satisfaction problem.
- Trace the execution of the backtracking search algorithm.
- Verify whether a constraint is arc-consistent.
- Trace the execution of the AC-3 arc consistency algorithm.
- Trace the execution of the backtracking search algorithm with arc consistency.
- Trace the execution of the backtracking search algorithm with arc consistency and with heuristics for choosing variables and values.

2.4 Local Search

- Describe the advantages of local search over other search algorithms.
- Formulate a real world problem as a local search problem.
- Verify whether a state is a local/global optimum.
- Describe strategies for escaping local optima.

- Trace the execution of greedy descent, greedy descent with random restarts, simulated annealing, and genetic algorithms.
- Compare and contrast the properties of local search algorithms.

3 Machine Learning

3.1 Introduction to Machine Learning

- Identify reasons for building an agent that can learn.
- Describe different types of learning.
- Define supervised learning, classification, and regression.
- Define bias, variance, and describe the trade-off between the two.
- Describe how prevent over-fitting by performing cross validation.

3.2 Decision Trees

- Describe the components of a decision tree.
- Construct a decision tree given an order of testing the features.
- Determine the prediction accuracy of a decision tree on a test set.
- Compute the entropy of a probability distribution.
- Compute the expected information gain for selecting a feature.
- Trace the execution of and implement the algorithm for learning a decision tree.
- Construct decision trees with real-valued features.
- Construct a decision tree to prevent over-fitting.

3.3 Artificial Neural Networks

- Describe motivations for using a neural network model.
- Describe the simple mathematical model of a neuron.
- Describe desirable properties of an activation function.
  Give examples of activation functions and their properties.
- Distinguish feed-forward and recurrent neural networks.

- Learn a perceptron that represents a simple logical function.
- Determine the logical function represented by a perceptron.
- Explain why a perceptron cannot represent the XOR function.
- Construct a 3-layer neural network that represents the XOR function.
- Explain the steps of the gradient descent algorithm.
- Explain how we can modify gradient descent to speed up learning and ensure convergence.
- Describe the back-propagation algorithm including the forward and backward passes.
- Compute the gradient for a weight in a multi-layer feed-forward neural network.
- Describe situations in which it is appropriate to use a neural network or a decision tree.

4 Reasoning Under Uncertainty

4.1 A Review of Probability Theory

- Calculate prior, posterior, and joint probabilities using the sum rule, the product rule, the chain rule and Bayes' rule.

- Given a probabilistic model,
  determine if two variables are unconditionally independent, or conditionally independent given a third variable.

- Give examples of deriving a compact representation of a joint distribution by using independence assumptions.

- Describe components of a Bayesian network.

- Compute a joint probability given a Bayesian network.

- Explain the independence relationships in the three key structures.

4.2 Bayesian Networks

- Determine whether an independence relationship holds by applying d-separation.

- Given a Bayesian network and an order of the variables, construct a Bayesian network that correctly represents the independence relationships among the variables.

4.3 The Variable Elimination Algorithm

- Explain how we can perform probabilistic inference more efficiently using the variable elimination algorithm.

- Define factors. Manipulate factors using operations restrict, sum out, multiply and normalize.

- Describe/trace/implement the variable elimination algorithm for calculating a prior or a posterior probability given a Bayesian network.

- Explain how the elimination ordering affects the complexity of the variable elimination algorithm.
- Identify the variables that are irrelevant to a query.

4.4 Hidden Markov Models

- Construct a hidden Markov model given a real-world scenario.
- Explain the independence assumptions in a hidden Markov model.
- Calculating the filtering probability for a time step in a hidden Markov model.
- Describe the justification for a step in the derivation of the filtering formulas.
- Calculate the smoothing probability for a time step in a hidden Markov model.
- Describe the justification for a step in the derivation of the smoothing formulas.
- Describe the forward-backward algorithm.

5 Decision Making under Uncertainty

5.1 Decision Networks

- Model a one-off decision problem by constructing a decision network containing nodes, arcs, conditional probability distributions, and a utility function.
- Determine the optimal policy of a decision network by computing the expected utility of every policy.
- Determine the optimal policy of a decision network by applying the variable elimination algorithm.
- Given a decision network with a single decision, determine the optimal policy and the expected utility of the optimal policy by enumerating all the policies.
- Given a decision network with a single decision, determine the optimal policy and the expected utility of the optimal policy by applying the variable elimination algorithm.
- Given a decision network with sequential decisions, determine the optimal policy and the expected utility of the optimal policy by applying the variable elimination algorithm.

5.2 Markov Decision Processes

- Describe motivations for modeling a decision problem as a Markov decision process.
- Describe components of a fully-observable Markov decision process.
- Describe reasons for using a discounted reward function.
- Define the policy of a Markov decision process.
- Give examples of how the reward function affects the optimal policy of a Markov decision process.
- Trace the execution of and implement the value iteration algorithm for solving a Markov Decision Process.

- Trace the execution of and implement the policy iteration algorithm for solving a Markov Decision Process.