Model AI Assignments 2012

Todd W. Neller
Gettysburg College
tneller@gettysburg.edu

Laura E. Brown and **John Earnest** and **Jason Hiebel**
Michigan Technological University
{lebrown, jwearnes, jshiebel}@mtu.edu

Douglas Turnbull
Ithaca College
dturnbull@ithaca.edu

Abstract

The Model AI Assignments session seeks to gather and disseminate the best assignment designs of the Artificial Intelligence (AI) Education community. Recognizing that assignments form the core of student learning experience, we here present abstracts of three AI assignments from the 2012 session that are easily adoptable, playfully engaging, and flexible for a variety of instructor needs. Assignment specifications and supporting resources may be found at http://modelai.gettysburg.edu.

Vitro - A Simulation and Visualization Framework to Engage Learning: Reversi Model Assignment Exemplar - Laura E. Brown, John Earnest, and Jason Hiebel

The classic two-player board game of Reversi (a.k.a. Othello™) is presented as an entertaining domain in which adversarial search is studied. Students are expected to implement an agent using the minimax and alpha-beta pruning algorithms and to explore evaluation functions. They are given a project codebase (Vitro) and Java source code to setup the Reversi domain and implement a naïve greedy agent. A tic-tac-toe domain is also provided for testing purposes, illustrating concepts at a smaller scale, and ensuring students' solutions are generalizable.

Vitro is a simulation and visualization framework designed for use by instructors and students for in-class lecture aids, distributed demonstrations, and assignments. The Reversi assignment within Vitro provides an intuitive visual mechanism for students to explore their agents' behavior. Reversi is the first of a series of programming assignments all using the Vitro codebase; additional classic and entertaining domains are implemented within the project (N-puzzle, Wumpus, Sokoban, Lunar Lander, etc.) allowing for future assignments to be created covering a wide range of AI topics. The Reversi assignment was evaluated in a pilot study. Students enjoyed the visual interface and opportunities to study agents' behavior, giving positive reviews.

Music Genre Classification - Douglas Turnbull

In this model AI assignment, students explore the engaging topic of music genre classification while developing an understanding of important supervised machine learning concepts. We provide a small music data set consisting of 150 songs that have been divided into 6 genres (e.g. jazz, rock, techno). Each song is represented by a bag-of-feature-vectors where each vector describes the timbre, or "color", of the sound for a short segment of audio data. If we think about each feature vector as being a point in a timbre space, then we can think of a song being a cloud of points in this same timbre space. Furthermore, we can think of many songs from a particular genre as occupying a region in this space. We will use a multivariate Gaussian probability distribution to model this region of timbre for each genre. When we are given a new unclassified song, we calculate the probability of the song's bag-of-audio-feature-vectors under each of the six Gaussian genre models. We then predict that the genre with the highest probability. We can evaluate the accuracy of our Gaussian classifier by comparing how often the predicted genre matches the true genre for songs that were not originally used to train the Gaussian genre models.

Solving the Dice Game Pig: an introduction to dynamic programming and value iteration - Todd W. Neller

Pig is a folk jeopardy dice with simple rules: Two players race to reach 100 points. Each turn, a player repeatedly rolls a die until either a 1 is rolled or the player holds and scores the sum of the rolls (i.e. the turn total).

Although Pig is simple to describe, the optimal policy for play is far from trivial. Using the computation of the optimal solution as a central challenge problem, we introduce dynamic programming and value iteration methods, applying them to similar problems using the Java language.

The object of this project is to give the student a deep, experiential understanding of dynamic programming and value iteration through explanation, implementation examples, and implementation exercises. Students are expected to:

- Demonstrate the need for dynamic programming through Fibonacci number computation.

- Study the details of a dynamic programming Java solution to a simple Pig variant with an acyclic state space.
- Apply the same dynamic programming technique to one of three novel exercises.
- Study the details of a value iteration Java solution to a simple coin variant of Pig called Piglet.
- Apply the same value iteration technique to one of three novel exercises.