

Model AI Assignments 2023

Todd W. Neller¹, Raechel Walker², Olivia Dias², Zeynep Yalçın³, Cynthia Breazeal², Matt Taylor², Michele Donini⁴, Erin J. Talvitie⁵, Charlie Pilgrim⁶, Paolo Turrini⁶, James Maher⁷, Matthew Boutell⁷, Justin Wilson⁷, Narges Norouzi⁸, and Jonathan Scott⁸

¹Gettysburg College, ²Massachusetts Institute of Technology, ³Wellesley College, ⁴Amazon Web Services, ⁵Harvey Mudd College, ⁶The University of Warwick, ⁷United States Air Force Academy, ⁸University of California, Santa Cruz

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 six AI assignments from the 2023 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>.

Analyzing the COMPAS Recidivism Algorithm - Raechel Walker, Olivia Dias, Zeynep Yalçın, Cynthia Breazeal, Matt Taylor, and Michele Donini

Data activism involves using data science to challenge power inequalities, such as racism. While we have created a five part data activism curriculum, we will discuss the few portions of the curriculum that emphasizes AI fairness. In this set of activities, students learn how to use data science to recognize, mitigate, and advocate for people that are disproportionately impacted by systemic inequality. Students execute the entire data science development process (data mining, data cleaning, data visualizations, and modeling). First, students will learn about the historical bias in arrest data. It is essential that students understand the nuances of systemic racism in the criminal justice system, so they can understand how AI can amplify racism. In the final assignment, students will analyze an infamous AI algorithm, called the Correctional Offender Management Profiling for Alternative Sanctions (COMPAS) recidivism algorithm. The project will teach them the critical thinking and programming skills required to challenge social injustice. This assignment aims to enlighten students about the importance of analyzing algorithms throughout the entire data science development process. These skills can be applied to a data set students are passionate about, such as predictive policing, and bias facial recognition.

Copyright © 2023, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

The WARLACS AI Assignments - Erin J. Talvitie

The WARLACS (Writing And Research in Liberal Arts Computer Science) AI assignments both give students practice applying a variety of AI ideas and scaffold the development of their writing and research skills, culminating in a final project that models the basic research process at a small scale. In this sequence of assignments, students apply heuristic search, adversarial search, and reinforcement learning ideas to challenging (but manageable), open-ended, and visually engaging problems. In addition to offering hands-on experience with AI applications, these projects model the basics of sound experimental design (e.g. averaging over multiple trials, performing ablation studies, designing counterexamples). In every project, students also write about their ideas, methods, and results in a variety of genres including informal emails, technical reports, grant applications, and paper reviews. The sequence of projects culminates in a large final project in which students experience a small-scale version of the basic research process in a topic of their own choosing, ultimately submitting a paper and presenting their work at a “conference” of their classmates.

Regret Matching Notebook - Charlie Pilgrim and Paolo Turrini

Students build their own regret matching algorithm and use it to solve for Nash equilibrium in a game of Rock Paper Scissors. The assignment is written as a Jupyter notebook that walks the students through building the algorithm in python, explaining each step in an interactive way. Students write the required elements of the algorithm by completing functions and running tests to verify that the code works correctly. They are then challenged to put these functions together to complete the algorithm and use it to solve Rock Paper Scissors. As an extra challenge, students are asked to visualise the learning process. Each of the last 3 years we have presented similar versions of the notebook to computer science Masters students as a one-hour seminar, with very good feedback. We include model solutions that can be released to students after the seminar.

Training Artificial Neural Networks to Beat StarCraft II - James Maher, Matthew Boutell, Justin Wilson

StarCraft II is a real-time strategy game, developed by Blizzard games, in which a player mines resources, builds a base, and fights a computer player or other human player. In 2019, Google's Deep Mind AI team created Alpha-Star, which achieved Grandmaster level playing StarCraft II against human opponents. We developed an undergraduate-level course project that utilizes the Python API (Python StarCraft II (`pysc2`)) developed for programmatic interactions with StarCraft II. We require three checkpoint turns to identify struggling students and to help these students past struggles in the initial stages of the project. The first assignment requires the students to become familiar with the `pysc2` environment and implement a scripted game mining resources, building marines, and sending the marines to random locations on the Simple 64 map. The second assignment requires the implementation of a Keras sequential neural network, which receives inputs about the game state and then predicts an ideal location to move the player's army. Finally, students optimize their neural networks and record these findings in a final report. Our project provides students a non-deterministic training environment where a neural network's predictions are immediately visible for analysis beyond only examining accuracy and precision numbers.

A 4-Module Sequence for Applied Deep Learning - Narges Norouzi

We designed a 4-module sequence for an undergraduate deep learning course. Modules are designed to give students hands-on experience with deep learning frameworks and concepts through working with datasets of varying types. The coverage for modules includes:

Module 1 introduction to linear regression analysis by working on the Facebook Metrics Dataset for modeling total user interactions. Students have also been introduced to non-linear datasets and how kernel methods and weight regularization can be used for non-linear modeling patterns.

Module 2 through working on the Fashion-MNIST dataset, students will

1. understand how to pre-process image data and conduct dimensionality reduction,
2. implement logistic regression from scratch, and
3. implement a neural network and observe the capacity of neural networks to learn a non-linearly separable decision boundary.

Module 3 focuses on designing convolutional neural networks and training them on the CIFAR-10 dataset. The modules also introduce transfer learning.

Module 4 students will work on training recurrent neural networks on the Reuters newswire classification dataset and will analyze their observations.

This 4-module sequence is easily adoptable by instructors and is implemented in a modular structure. The target audience of these modules is undergraduate CS or engineering students taking AI or ML courses that cover topics such as

1. linear regression,
2. logistic regression,
3. neural networks,
4. convolutional neural networks, and
5. recurrent neural networks.

The prerequisites for the course include programming, data structures, algorithms, and probability courses.

Local Search in Ackley Surface with Scaffolding - Jonathan Scott and Narges Norouzi

Teaching students different local search algorithm requires careful definition of a non-convex problem, appropriate scaffolding, and visualization. In this assignment, students will implement three optimization algorithms and examine their behavior on a hard-to-optimization function. The assignment uses an Ackley surface and scaffolding for students to implement

1. stochastic hill climbing with re-starts,
2. simulated annealing, and
3. local beam search.

Visualization is used for students to observe the behavior of each optimization algorithm on non-convex surfaces. Additional resources are provided to instructors to adapt the assignment to other non-convex optimization problems.