Final Year Project

Toby Devlin

November 7, 2017

Contents

Chapter 1

Introduction

General discussion on what game theory is and what the PD is. what this topic is and how it fits into the big picture. The Prisoners Dilemma is a classic game theory topic... Its important to game theory because... creating general sequences to strategies is important because... models in the real world that follow certain strategies... how we can leverage these to peruse goals.

1.1 Background

fill this section with background on game theory in when I do the course?

1.1.1 Iterated Prisoners Dilemma

The Prisoners Dilemma is a well known game theory problem based on the example of a pair of prisoners and their subsequent interrogation. The game is as follows:

Something about the PD

The single game itself is very basic and is modeled in the following way:

$$give - a - model - here$$

The Iterated Prisoners Dilemma is the iterated version of the Prisoners Dilemma[1]. The iteration of the game is what makes the game an interesting concept, as now **learn the technical stuff and put it here!!!!!** we are able to create strategies[2] that look to gain an upper hand based on **Something here**

[1] reference this stuff dude, come on..

[2] When referring to ourselves, we will describe our moves as a strategy. When referring to an opponent we can use the term opponent and strategy interchangeably.

1.1.2 Machine Learning Concepts

This section will briefly provide a background to machine learning algorithms implemented in the axelrod-dojo[3]. This is by no means a comprehensive look into these subjects but will provide sufficient background on technical discussion later on.

Genetic Algorithms

Genetic Algorithms are a description of techniques for generating solutions complex problems such as searching and, in our case, optimization[4]. The basis of a genetic algorithm is focused on a cycle of evolution. Like nature, we create a survival of the fittest concept[5] to evaluate a population, kill off the weakest members and create offspring from the most successful population.

Put a figure of the cycle here.

Initially we create a heuristic function, say our fitness function, which is a measure of how successful a candidate in our population is. Then we run our whole population through this function, ranking each one by how successful their score is. At this point we can create a cut off[6] to decide which of the population not to put through to the next round.

Bayesian Optimization

1.2 Brief Overview

In this document I will be be looking at the creation of sequences to beat given players in The Iterated Prisoners Dilemma[7]. My research looked into just the single opponent use case, but the idea of designing a sequence for a given number of opponents is looked at in the further study of the report. This task is the Problem:
Given a certain opponent, O, (with a provided strategy, S) what is the best possible sequence of moves, in a game of n turns, made by my strategy to maximise my players score?

[3]referencing opportunity here
[4]Mitchell, Melanie (1996). An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press. ISBN 9780585030944. learn to reference soon
[5]need to reference Darwin?
[6]Can often be referred to as the bottleneck
[7]Reference this for some background

Chapter 2

Literature Review

2.1 Background

- book1
- book2

2.2 Strategies Of Interest

2.2.1 Tit for Tat

this is a sentence which has been changed[?]

2.2.2 Cycler

this is a type of opponent

2.2.3 Other

add at least two here

Chapter 3

Approach To Problems

3.1 In Depth Definition Of Task

3.2 Solution Form

The sequence archetype will use the Cycler() player for our strategy each time, only editing the input to improve our score. To this model we can apply an optimised input of length n (tbd) to the player, this sequence, as per the design of the strategy will then be repeated until the games end(if n = len(game) then we are just calculating the sequence for the whole game).

The input sequence itself will be created using a genetic optimisation. Starting with a set of randomly generated sequences, we will have each one play the opponent and return with a score. These sequences will be ranked and the lowest x% will be discarded, resulting in a fitter, but smaller, population than before. This smaller population will then create offspring using a —X TBD method X— pairing algorithm before mutating with —X TBD method X—. This new set of offspring will be included in the next scoring round and the process repeats for k number of rounds

This sequence of Play-Rank-Create-LOOP will be the basis of creating the optimal strategy for each other opponent.

3.3 initial research

local vs global maximum simple[1] vs non-simple
Here i will look at the algorthem in a general manner. i hope to answer questions like:

[1]ones we can figure out optimal

- what happens to the convergence rate to the optimal sequence for an oppenent when the initial starting poulation for our algorithm changes?
- what happens to the convergence rate to the optimal sequence for an oppenent when the mutation rate of each of our population is changed?
- what happens to the convergence rate to the optimal sequence for an oppenent when we change the crossover algorithm to be more random?
-

Chapter 4

Implementation Of Sequence Discovery

Chapter 5

Results and Discussion

Chapter 6

Practical Applications for Solution Sequences

Chapter 7

Summary and Future Research

Bibliography