CS 561, HW7

Prof. Jared Saia, University of New Mexico

Due: Dec. 3rd

1. Exercise 29.2-2 (Linear Program for the example in Figure 24.2(a))
2. Exercise 29.2-4 (Network Flow as an LP)
3. Rock, Paper, Scissors is a simple 2 person game. In a given round, both players simultaneously choose either Rock, Paper or Scissors. If they both choose the same object, it's a tie. Otherwise, Rock beats Scissors; Scissors beats Paper; and Paper beats Rock. Imagine you're playing the following betting variant of this game with a friend. When Scissors beats Paper, or Paper beats Rock, the loser gives the winner \$1. However, in the case when Rock beats Scissors, this is called a **smash**, and the loser must give the winner \$10.

   (a) Say you know that your friend will choose Rock, Scissors or Paper, each with probability 1/3. Write a linear program to calculate the probabilities you should use of choosing each object in order to maximize your expected winnings. Let p_1, p_2, p_3 be variables associated with the best way of choosing Rock, Scissors and Paper respectively. Note: If you want to check your work, there are several free linear program solvers on the Internets: check the Wikipedia page on linear programming.

   (b) Now say that your friend is smart and, also, clairvoyant: she will magically know the exact probabilities you are using and will respond optimally. Write another linear program to calculate the probabilities you should now use in order to maximize your expected winnings. Hint 1: If your opponent knows your strategy, her strategy will be to choose one of the three objects with probability 1. Hint 2: Review the LP you wrote for the shortest paths problem.

4. Exercise 34.5-2 (0-1 Integer Programming)

5. Exercise 34.5-1 (Subgraph Isomorphism)

6. Problem 4 from the 2018 final exam.

7. (Arora) Implement the portfolio management appearing in Gradient Descent slides in any programming environment and check its performance on S&P stock data (at `http://ocobook.cs.princeton.edu/links.htm`). Include your code as well as your final performance in the case where your initial portfolio is worth \$1. You will find the following code snippet useful for projecting back onto the convex space: `https://www.mathworks.com/matlabcentral/fileexchange/30332-projection-onto-si`