Problem Set 8

Due: Wednesday, November 2, 2005.

Problem 1. Markov chains. An $n \times n$ matrix P is **stochastic** if all entries are nonnegative and every row sums to 1, that is $\sum_j p_{ij} = 1$ (so each row can be thought of as taking a convex combination). Stochastic matrices are used to represent the **transition matrices** of **Markov chains**—random walks through a series of states. The term p_{ij} represents the probability, if you are in a current state i, that your next state will be j (thus the sum to one rule). If you have a probability distribution π over your current state, where π_i denotes the probability you are in state i, then after a transition with probability defined by P, your new probability distribution is πP.

Use duality to prove that for any stochastic matrix P, there is a **nonzero** vector $\pi \geq 0$ such that $\pi P = \pi$.

The vector π can be normalized to 1, in which case it represents a probability distribution that is **stationary** under the action of the transition matrix—that is, if π is the probability distribution on what state you are in before a transition, it is also the probability distribution after the transition. This proves that every Markov chain has a stationary probability distribution.

Hint: there is no objective function, so think in terms of feasibility/unboundedness. Also, you must somehow express the constraint $\pi > 0$ (a strict inequality). Consider the constraint $\sum \pi_i = 1$.

Problem 2. In a 0-*sum 2-player game*, Alice has a choice of n so-called *pure* strategies and Bob has a choice of m *pure* strategies. If Alice picks strategy i and Bob picks strategy j, then the *payoff* is a_{ij}, meaning a_{ij} dollars are transfered from Alice to Bob. So Bob makes money if a_{ij} is positive, but Alice makes money if a_{ij} is negative. Thus, Alice wants to pick a strategy that minimizes the payoff while Bob wants a strategy that maximizes the payoff. The matrix $A = (a_{ij})$ is called the *payoff matrix*.

It is well known that to play these games well, you need to use a *mixed* strategy—a random choice from among pure strategies. A mixed strategy is just a particular probability distribution over pure strategies: you flip coins and then play the selected pure strategy. If Alice has mixed strategy x, meaning he plays strategy i with probability x_i, and Bob has mixed strategy y, then it is easy to prove that the expected payoff in the resulting game is xAy. Alice wants to minimize this expected payoff while Bob wants to maximize it. Our goal is to understand what strategies each player should play.

We'll start by making the pessimal assumption for Alice that whichever strategy she picks, Bob will play best possible strategy against her. In other words, given Alice's strategy x, Bob will pick a strategy y that achieves $\max_y xAy$. Thus, Alice wants to find a distribution x that minimizes $\max_y xAy$. Similarly, Bob wants a y to maximize $\min_x xAy$. So we are interested in solving the following 2 problems:

$$\min_{\sum x_i=1} \max_{\sum y_j=1} xAy$$
$$\max_{\sum y_j=1} \min_{\sum x_i=1} xAy$$

Unfortunately, these are nonlinear programs!

(a) Show that if Alice's mixed strategy is known, then Bob has a pure strategy serving as his best response.

(b) Show how to convert each program above into a linear program, and thus find an optimal strategy for both players in polynomial time.

(c) Give a plausible explanation for the meaning of your linear program (why does it give the optimum?)

(d) Use strong duality (applied to the LP you built in the previous part) to argue that the above two quantities are *equal*.

The second statement shows that the strategies x and y, besides being optimal, are in *Nash Equilibrium*: even if each player knows the other's strategy, there is no point in changing strategies. This was proven by Von Neumann and was actually one of the ideas that led to the discovery of strong duality.

Problem 3. You are given a collection of n points in some metric space (i.e., the distances between the points satisfy the triangle inequality). Consider the problem of dividing the points into k clusters so as to minimize the maximum diameter of (distance between any two points in) a cluster.

(a) Suppose the optimum diameter d is known. Devise a greedy 2-approximation algorithm. **Hint:** consider any point and all points within distance d of it.

(b) Consider the algorithm that (k times) chooses as a "center" the point at maximum distance from all previously chosen centers, then assigns each point to the nearest center. By relating this algorithm to the previous algorithm, show that you get a 2-approximation.

Problem 4. Consider the problem of scheduling, on one machine, a collection of jobs with given *processing times* p_j, *due dates* d_j, and *lateness penalties* (*weights*) w_j paid for jobs that miss their due dates, so as to minimize the total lateness penalty. (If we let U_j denote the indicator variable for job j completing after its due date, we want to minimize $\sum w_j U_j$.)

(a) Argue that any *feasible* subset of jobs (that can all together be completed by their due dates) might as well be scheduled in order of increasing deadline (so it is sufficient to find a set without worrying about order). **Hint:** if two adjacent jobs in the sequence are out of order, swap them.

(b) Assuming the lateness penalties are polynomially bounded integers, give a polynomial-time dynamic program that finds the fastest-completing maximum-weight feasible subset.

(c) Give a fully polynomial-time approximation scheme for the original problem of minimizing lateness penalty with arbitrary lateness penalties.