Linear Programming and Combinatorial Optimization
Tutorial 2

January 18, 2019

1. Show how to compute the inverse of an $n \times n$ matrix. What is the time complexity?

2. Given a non-zero vector c, find a vector x such that $c^T x > 0$. A point in $S = \{x : Ax \leq b\}$ is said to be on the *boundary* if at least one of the inequalities is an equality i.e. $\exists i \quad A_i x = b_i$. Prove that in such a case, there is a direction vector y such that moving along the direction of y takes you outside S. That is, $x + \epsilon y \notin S$ for any $\epsilon > 0$.

3. Let x_0 be on the boundary. Let $A'x_0 = b'$ be the equalities and $A''x_0 < b''$ be the strict inequalities. Let rank of A' be r. Then the set $\{y \mid A'y = b', A''y < b''\}$ is called an $n-r$ *face* or $n-r$ *dimensional face* of S.

   What is the maximum number of k dimensional faces possible as a function of m, n, k?

4. During the execution of simplex algorithm, show how to detect if there is a degeneracy at a given extreme point.

5. In continuation to the above, show how a degeneracy can be eliminated without affecting the feasible region.