Advanced Algorithms (3501)

Spring Semester, 2023

Problem Set 3

Published on: 30/05/2023
Submit by: 14/06/2023, 23:59

1. A collective of farmers is growing avocados. They have two fields, located in the Galilee and in the Negev, and three stores, located in Tel Aviv, Jerusalem and Haifa. The monthly demands in each store, in tons, are 8, 6 and 5 (respectively). The field in the Galilee produces 10 tons per month, and the field in the Negev produces 9 tons per month.

   The transportation costs of 1 ton of avocados, in 1000's of shekels, between any field and any store are given in the following table:

	Tel Aviv	Haifa	Jerusalem
Galilee	5	3	7
Negev	6	10	4

   Our goal is to determine how many tons of avocado are to be transported from each field to each store, in order to meet the demands while minimizing the total shipping cost.

   Formulate this problem as a linear program. Then, use some linear programming solver (there are even online tools) in order to solve the problem. In your solution, you should only write what is the linear program you're solving and then give both the optimum and the optimal solution (that is, write what is the minimal total cost, and how many tons should be transported from each field to each city). You don't need to specify all the steps of the algorithm used to solve this optimization problem.

2. Use the second version of Farkas' lemma we showed in class in order to prove the following, original formulation:

   Let $\{Ax = b, x \geq 0\}$ be a linear system of inequalities. If it is infeasible, then there's $y \in \mathbb{R}^n$ such that $y^T A \geq 0$ and $y^T b < 0$.

3. Recall that *Bland's rule* is a pivoting rule that guarantees that the simplex algorithm runs in finite time even if we encounter degenerate solutions in which we can't improve the optimum by moving to a neighboring vertex. The rule is as follows:

   > Choose the entering basic variable x_j such that j is the smallest index with $(\tilde{c}_N)_j < 0$, where $\tilde{c}_N$ is the reduced cost vector as defined in class. As for the variable leaving the basis, in case there are multiple indices i minimizing $\min_{i:\bar{A}_{i,j}>0} \frac{\bar{b}_i}{\bar{A}_{i,j}}$, pick the one with the smallest index.

In this question, we will prove:

Theorem 1. *Bland's pivoting rule guarantees a finite termination of the simplex algorithm with an optimal solution.*

Assume towards contradiction that this doesn't happen, i.e., the algorithm doesn't terminate. Since the number of bases is finite, this means that there is a *cycle* of bases considered by the algorithm: $B_0, B_1, \ldots, B_k, B_0$. Recall that the value of the objective function we are minimizing remains the same throughout the cycle, since we are always in a degenerate solution. Further, observe that our solution itself x^* doesn't change throughout the cycle: in each step we compute $\varepsilon = \min_{i:\bar{A}_{i,j}>0} \frac{\bar{b}_i}{\bar{A}_{i,j}}$, and update x_B with $x_B - \varepsilon \bar{A} e_j$ and x_j with $x_j + \varepsilon$, but in our case $\varepsilon = 0$.

So far, we thought of our reduced cost vector $\tilde{c}_N^T$ as indexed only by variables in N, since these are the only values we care about. However, now we will be comparing different bases, so it is convenient to extend the reduced cost vector to an n-dimensional vector $\tilde{c}$ by extending it with 0's in all the coordinates corresponding to variables in B. Alternatively, for a base B and $N = \{1, \ldots, n\} \setminus B$, define $\tilde{c}^T = c^T - c_B^T (A_B)^{-1} A$. (You can verify that these definitions are equivalent but don't need to write it in your submitted solution).

Definition 2. *We say a variable j is* fickle *if it appears in some, but not all bases B_i in the cycle.*

Observe that if x_j^* is fickle then $x_j^* = 0$ throughout the cycle.

Let t be the fickle variable with largest index, and let B be a basis in the cycle such that $t \in B$ but not in the next basis in the cycle. Let s be the index of the variable that enters the basis instead of t in the next basis. Let $\tilde{c}$ be the reduced cost vector with respect to B.

(a) Show the following properties regarding s and t:

  i. $\bar{A}_{t,s} > 0$ where as usual $\bar{A} = A_B^{-1} A_N$.

  ii. $s < t$.

  iii. $\tilde{c}_s < 0$ and $\tilde{c}_t = 0$.

Let $d \in \mathbb{R}^n$ be the following vector: $d_B = -\bar{A}_s$, where $\bar{A}_s$ denote the s-th column of $\bar{A}$, $d_s = 1$ and $d_j = 0$ for all $j \in N \setminus s$. (The motivation for defining d is as follows: if we were indeed at a non-degenerate vertex, the algorithm would have moved in "direction" d by an "amount" ε, that is, the new vertex is obtained by adding εd to the current vertex. However, in our case the "amount" ε equals 0, so in fact the algorithm doesn't move at all).

(b) Prove the following properties regarding d:

i. $d_t < 0$ and $d_s = 1$.

ii. If $j \in B$, $j \neq t$ and j is fickle, $d_j \geq 0$. (this is a harder one! Here you actually need to use properties of Bland's pivoting rule)

iii. If $j \in N$ and $j \neq s$, $d_j = 0$.

iv. $Ad = 0$ (Hint: your life will be much easier if you write Ad as $A_B d_B + A_N d_N$).

Note that we didn't record any information on d_j if $j \in B$ isn't fickle.

There's also a basis B' in the cycle such that $t \notin B'$, but t is selected to be the entering variable for the next basis. We note by $\tilde{c}'$ the reduced cost vector with respect to B'.

(c) Prove the following properties regarding $\tilde{c}'$:

i. $\tilde{c}'_t < 0$.

ii. For all fickle $j \neq t$, $\tilde{c}'_j \geq 0$ (Hint: consider the cases $j \in B'$ and $j \notin B'$, and use properties of the pivoting rule).

Finally, define $\tilde{f} = \tilde{c}' - \tilde{c}$. To derive a contradiction, we'll compute $\tilde{f}^T \cdot d = \sum_{i=1}^n \tilde{f}_i d_i$ in two ways.

(d) Use the definition of $\tilde{f}^T$ to show that it equals $v^T A$ for some vector v, and use this and the previous items to deduce that $\tilde{f}^T \cdot d = 0$.

(e) Show the following properties about $\tilde{f}$: $\tilde{f}_t < 0$. $\tilde{f}_s > 0$. $\tilde{f}_j \geq 0$ if $j \in B$, $j \neq t$ and j is fickle. And $\tilde{f}_j = 0$ if $j \in B$ isn't fickle.

(f) Use the item above along with the properties of d to conclude that $\tilde{f}^T \cdot d > 0$, a contradiction. Guidance: partition the sum $\sum_{i=1}^n \tilde{f}_i d_i$ according to whether $i = s$, $i = t$, $i \in B$ is fickle, $i \in B$ isn't fickle, or $i \in N$ and $i \neq s$. For each group of summands, try to gather all the information we have about $\tilde{f}_j$ and d_j.

4. In this question we will investigate the properties of a class of matrices called *positive semi-definite matrices*. It will probably not be clear to you yet how is this related to linear programming — but trust me, it is. We will talk about these matrices later in class and then everything will make sense.

Recall that a real, symmetric matrix is diagonalizable, its eigenvalues are real and it has n orthonormal eigenvectors. These facts will come in handy.

An $n \times n$ symmetric real matrix A is called *positive semi-definite* (PSD) if for every $v \in \mathbb{R}^n$, $v^T A v \geq 0$.

(a) Prove that if A is PSD then all of its eigenvalues are non-negative.

(b) Prove that if $A \in \mathbb{R}^{n\times n}$ is a real symmetric matrix such that all of its eigenvalues are non-negative, then there exists $B \in \mathbb{R}^{n\times n}$ such that $A = B^T B$.

(c) Prove that if $A \in \mathbb{R}^{n\times n}$ is such that there exists $B \in \mathbb{R}^{n\times n}$ such that $A = B^T B$, then A is PSD.

From the 3 items above, it follows that the following are 3 equivalent definitions for an $n \times n$ real symmetric matrix to be PSD:

(a) For all $v \in \mathbb{R}^n$, $v^T A v \geq 0$

(b) All the eigenvalues of A are non-negative

(c) $A = B^T B$ for some $B \in \mathbb{R}^{n \times n}$.