A Study on Optimal Assignment problem

EE 694: Seminar Report, submitted as part of course project
under the guidance of Prof. Harish Pillai (IITB)
Author: Satwik (193079009)

11^{th} May, 2021.

Contents

1 Introduction **2**

2 Linear programming (LP) problem **2**

3 Dual of the linear programming problem **3**

4 Convex spaces **3**

5 Graph theory on an LP problem **4**

6 Birkhoff polytope **4**

7 Birkhoff and von Neumann Theorem **5**

8 Assignment problem **5**

9 Konig's theorem **6**

10 Birkhoff algorithm for decomposing a doubly stochastic matrix **8**

11 Conclusion **9**

1 Introduction

The report consists of a study on algorithms for finding a solution to some of the well defined optimization problems. Throughout the report we find recurring presence of doubly stochastic matrices in the optimization problems.

2 Linear programming (LP) problem

In linear optimization problems the goal is to maximize the value of a linear function of a variable $\mathbf{x} = (x_1, x_2, x_3, \ldots, x_n)^T$ i.e.

$$arg\ max_x\ c^T x$$

Such that a set of inequalities: $A\mathbf{x} <= b$ and $\mathbf{x} >= \mathbf{0}$
If there are m inequalities and n variables then A is m x n matrix and b is an m x 1 vector
Example:

$$\text{Maximize } 4x_1 + x_2 + x_3$$

Subject to:

$$x_1 + 4x_2 <= 1$$
$$3x_1 - x_2 + x_3 <= 3$$
$$x_1, x_2, x_3 >= 0$$

The problem can be written as:

$$\text{Maximize } \begin{bmatrix} 4 & 1 & 3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Subject to:

$$\begin{bmatrix} 1 & 4 & 0 \\ 3 & -1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \leq \begin{bmatrix} 1 \\ 3 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

This is a preferred form of expressing a linear programming problem, partially because most of the optimization libraries require input in this form.

3 Dual of the linear programming problem

For a optimal solution of a linear programming problem the cost function has to be either maximized or minimized. At the optimal solution of a linear programming problem the complementary slack conditions are equal to zero. This interesting property of the dual of an LP problem plays a significant role in the algorithms used for solving the optimal assignment problem.
An inequality can be converted into an equality by adding a non negative variable called a slack variable.

$$a_1 * x_1 + a_2 * x_2 + \ldots + a_n * x_n \leq b$$
$$a_1 * x_1 + a_2 * x_2 + \ldots + a_n * x_n + s = b$$

Here s is the slack variable and $s \geq 0$

4 Convex spaces

A set $C \subseteq \mathbb{R}^n$ is convex if and only if it contains all convex combinations of its points. In other words, if every point that lies on a line joining two points $x_1, x_2 \in C$ must also belong in C, hence $(1 - \lambda) * x_1 + \lambda * x_2 \in C$ for all $0 \leq \lambda \leq 1$ then the set C is a convex set.

The set formed by the solutions of $Ax \leq b$ is a convex set i.e. $\{x \in \mathbb{R}^n : Ax \leq b \text{ for } A \in \mathbb{R}^{mxn}, b \in \mathbb{R}^m\}$ is a convex set and to be more precise is a polyhedron.

In order to form the polyhedron from a set we need the concepts of a affienly independent set and convex hull.

In an affinely independent set of vectors, no vector can be written as linear combination of other points. i.e. $\sum_{i=1}^{n} \lambda_i x_i = 0$ and $\sum_{i=1}^{n} \lambda_i = 0 \Rightarrow \lambda_1 = \ldots = \lambda_t = 0$.

A convex hull formed from a convex set S is the convex combinations of all the points in set S and denoted by conv(S).

If a vector of a convex set cannot be written as a convex combination of other vectors of the set then, such a vector is known as an extreme point of the convex set.

If a convex set contains finite number of points, then conv(S) forms a bounded polyhedron also known as polytope. The polytope is a convex hull of its extreme points.

The solution to $Ax \leq b$ is a convex set i.e. $\{x \in \mathbb{R}^n : Ax \leq b \text{ for } A \in \mathbb{R}^{mxn}, b \in \mathbb{R}^m\}$ if it is bounded will form a polytope.

A convex hull of an affinely independent set will form a convex region in which every point has a unique representation as a convex combination of the vectors in affinely independent set.

In an LP problem, the optimal point is found at the boundary regions and hence the optimal point is found at the extreme points of the convex set.

Another way to describe an extreme point is a point where some objective function is uniquely minimised there, $x \in S$ is a vertex of S, then there exists a $c \ \in \mathbb{R}^n$ such that $c^T x < c^T y \quad \forall y \neq x;\ y \in S$

5 Graph theory on an LP problem

For understanding solution to the dual of the LP problem we need the concepts of maximum matching in a graph.

A graph contains nodes and edges which can be represented using an adjacency matrix. For problems involving assigning states of one feature to states of another (such as assigning jobs to people) we expect the solution to be a perfect matching graph.

We solve the dual problem of LP for maximum matching. Konig's theorem helps us to understand the connection between perfect matching in a graph and minimum covering required to cover the non zero elements with straight lines

6 Birkhoff polytope

An $n \times n$ matrix $A = (a_{ij})$ is double stochastic if,

(i) $a_{ij} \geq 0$ for $i, j = 1, \ldots, n$

(ii) $\sum_{i=1}^{n} a_{ij} = 1,\ j = 1, \ldots, n$

(iii) $\sum_{j=1}^{n} a_{ij} = 1,\ i = 1, \ldots, n$

The set of all doubly stochastic matrices is called a Birkhoff polytope. The Birkhoff polytope has following properties,

- Birkhoff polytope is convex, hence the set of all $n \times n$ doubly stochastic matrices is convex.
- For $A \in \Omega_n$, $\dim(A) = (n-1)^2$

$$\begin{bmatrix} 0.2 & 0.3 & 0.5 \\ 0.3 & 0.6 & 0.1 \\ 0.5 & 0.1 & 0.4 \end{bmatrix}$$

This is an example of a doubly stochastic matrix of dimension 4.

7 Birkhoff and von Neumann Theorem

A permutation matrix is a matrix which has exactly one entry of 1 in each row and each column and zeros elsewhere.

The permutation matrices constitute the extreme points of the set of doubly stochastic matrices.

The set of doubly stochastic matrices is the convex hull of the permutation matrices which means any doubly stochastic matrix can be uniquely represented as a convex combination of the permutation matrices.

This means any doubly stochastic matrix can be decomposed into convex combination of permutation matrices.

Example:

$$\begin{bmatrix} 0.2 & 0.3 & 0.5 \\ 0.3 & 0.6 & 0.1 \\ 0.5 & 0.1 & 0.4 \end{bmatrix} = 0.3 \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} + 0.1 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} + 0.5 \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} + 0.1 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

But the decomposition is not unique. For example,

$$\begin{bmatrix} 0.2 & 0.3 & 0.5 \\ 0.3 & 0.6 & 0.1 \\ 0.5 & 0.1 & 0.4 \end{bmatrix} = 0.2 \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} + 0.4 \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} + 0.1 \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix} + 0.2 \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} + 0.1 \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

In the above example, the dimension of the doubly stochastic matrix is 4. So, the doubly stochastic matrix can be represented uniquely using 4 extreme points or 4 permutation matrices and will have non-unique representations in higher dimensions of permutation matrices.

8 Assignment problem

An optimal assignment problem can be defined as identifying a perfect matching such that the associated cost function maximizes (or minimizes).

Consider an $n \times n$ matrix $A = (a_{ij})$. An optimal assignment problem can be defined as finding a permutation p of the integers $\{1, 2, \dots, n\}$ that maximizes (or minimizes):

$$\sum_{i=1}^{n} a_{ip(i)} \tag{1}$$

A classical representation of the problem is:

$$max \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij}x_{ij} \tag{2}$$

$$\text{such that: } \sum_{j=1}^{n} x_{ij} = 1(i = 1, 2, \ldots, n), \tag{3}$$

$$\sum_{i=1}^{n} x_{ij} = 1(j = 1, 2, \ldots, n), \tag{4}$$

$$x \in 0, 1(i, j = 1, 2, \ldots, n). \tag{5}$$

A classical problem the "The personnel assignment problem" where the problem consists of finding a way to assign N individuals to N jobs such that the average time for completion of a task of N jobs is minimum.

Though trying out all possible arrangements and finding the values is feasible for low values of N the problem becomes computationally time consuming as N increases to larger and larger values. For $N > 100$ it is almost impossible to solve within workable time using this method.

For solving the assignment problem (1) we would need a few results from graph theory.

9 Konig's theorem

The assignment problem (1) can be thought of as finding a perfect matching for a bipartite graph. Consider a bipartite graph G with matching vertices U, matched vertices V and edges E. For such a bipartite graph G = (U,V;E) a matching can be defined as the subset of the edges E denoted by M such that every and any vertex of G will coincide at most with one edge of M. If cardinality of matched and to be matched vertices are equal then such a matching is know as perfect matching. A cover C $\subseteq$ U $\bigcup$ V such that every and any edge of G coincides with at least one vertex of C.

The main construct of this section is that the maximum cardinality of a matching is equal to the minimum cardinality of covering for a bipartite graph. This result helps us in constructing a dual to the primal equation (1), and proving the fact that the dual and the primal solutions match.

Theorem 1 *In a bipartite graph the maximum cardinality of matching is equal to the minimum cardinality of cover.*

The proof for this proposition is provided in [1]

For a perfect matching to exist in a bipartite graph the following result is also needed:

Theorem 2 *A bipartite graph $G = (U, V;E)$ with $|U| = |V|$ admits a perfect matching if and only if*

$$|U'| \leq |N(U')| \tag{6}$$

holds $\forall\ U' \subseteq U$ where $N(U')$ denote the set of neighbours of U' i.e. $N(U') = \bigcup_{i \in U'} \{j \in V : [i, j] \in E\}$

A covering system can be defined as set of lines (rows and columns) required that connect the nonzero elements of a matrix. Let λ_i be the multiplicity of the line that covers i^{th} row and μ_j be the multiplicity of the line that covers j^{th} column of a matrix M and follows an additional constrain $\lambda_i + \mu_j \geq a_{ij}(i, j = 1, 2, \ldots, n)$

From optimization theory, one can recognize that the above construct is a dual to the original primal problem (1)

Theorem 3 *If a_{ij} is an $n \times n$ matrix of non-negative integers then,*

$$min \sum_{k=1}^{n} (\lambda_i + \mu_j) = max \sum_{i=1}^{n} a_{ip(i)} \tag{7}$$

With this theorem it is clear that if a bipartite graph has a perfect matching then the dual of the perfect matching problem will give the same result as the primal problem i.e. condition for strong duality has been established.

The proof of [Theorem 3] in [1] also provided us an algorithm for solving the optimization problem.

The following algorithm is known in the art as the Satellite-switched Time-Division Multiple Access (SS/TDMA). We shall later see the use of this algorithm for obtaining the decomposition of a doubly stochastic matrix using the Birkhoff Vonnuemann theorem.

Consider the $n \times n$ cost matrix $C = (c_{ij})$ where c_{ij} denotes the cost corresponding to the i^{th} and j^{th} states. The following is the algorithm:

1. Consider the maximum of the row sum of each row and column sum of each column c^*
2. Now make another correction matrix S using the following method
   - Calculate the slack for each row and column by subtracting the i^{th} row sum (or column sum j^{th}) from c^*
   - For i in 1 to n
   - For j in 1 to n, do the following
   - $s_{ij} = min(a_i, b_j)$

- $a_i = a_i - s_{ij}$
- $b_j = b_j - s_{ij}$

3. The above step creates a correction matrix S that when added to C creates a modified cost matrix whose row (and column) sum adds upto c^*

4. Dividing every element with $c*$ creates a doubly stochastic matrix

5. Let G = (U, V; E) be a bipartite graph with $|U| = |V|$ and $E = [i, j] : t_{ij} \geq 0$;

6. Find a perfect matching in G, and the corresponding permutation matrix P;

7. $\tau = \min\ t_{ij} : p_{ij} = 1, T = T - \tau P$

8. If T is a zero matrix then stop else go to 5

The above code also provides the algorithm for decomposing a doubly stochastic matrix into convex combination of permutation matrices.

10 Birkhoff algorithm for decomposing a doubly stochastic matrix

```
# Python code used for Birkhoff decomposition
import numpy as np
import itertools
from networkx import from_numpy_matrix
from networkx.algorithms.bipartite.matching import maximum_matching

if __name__=='__main__':
    D = np.matrix([[0.1,0.2,0.7],[0.3,0.4,0.3],[0.6,0.4,0]])
    print(D)
    print()
    D = D.astype('float')
    birk_coeff = []
    birk_permutations = []
    m, n = D.shape
    if m != n:
        raise ValueError('given matrix is not a square matrix'.format(m, n))
    indices = list(itertools.product(range(m), range(n)))
    while not np.all(D == 0):
        replica = np.zeros_like(D)
        replica[D.nonzero()] = 1

        bipartite = np.vstack((np.hstack((np.zeros((m, m)), replica)),
            np.hstack((replica.T, np.zeros((n, n))))))
        bipartite_graph = from_numpy_matrix(bipartite)
        left_nodes = range(n)
```

```
    perfect_matching_graph = maximum_matching(bipartite_graph, left_nodes)
    perfect_matching_nodes = {u: v % n for u, v in
        perfect_matching_graph.items() if u < n}

    num = len(perfect_matching_nodes)
    P = np.zeros((num, num))
    P[list(zip(*(perfect_matching_nodes.items())))] = 1
    c = min(D[i, j] for (i, j) in indices if P[i, j] == 1)
    birk_coeff.append(c)
    birk_permutations.append(P)
    D -= c * P
    D[np.abs(D) < np.finfo(np.float).eps * 10.] = 0.0

answer = list(zip(birk_coeff, birk_permutations))
for i in answer:
    print('coeff:',np.around(i[0],2))
    print(i[1])
```

```
[[0.1 0.2 0.7]
 [0.3 0.4 0.3]
 [0.6 0.4 0.  ]]

coeff: 0.4
[[0. 0. 1.]
 [0. 1. 0.]
 [1. 0. 0.]]
coeff: 0.1
[[1. 0. 0.]
 [0. 0. 1.]
 [0. 1. 0.]]
coeff: 0.2
[[0. 1. 0.]
 [0. 0. 1.]
 [1. 0. 0.]]
coeff: 0.3
[[0. 0. 1.]
 [1. 0. 0.]
 [0. 1. 0.]]
```

11 Conclusion

We have looked into the application of doubly stochastic matrices in optimization problems and further implemented an algorithm for solving such optimization problem. We have looked in the dual construction of the optimization problem and used the results from graph theory for understanding the algorithm.

References

[1] Bapat, Ravi B., Ravindra B. Bapat, and T. E. S. Raghavan. Nonnegative matrices and applications. No. 64. Cambridge university press, 1997.

[2] Mehlum, Maria. Doubly stochastic matrices and the assignment problem. MS thesis. 2012.

[3] Martello, Silvano. "Jenő Egerváry: from the origins of the Hungarian algorithm to satellite communication." Central European Journal of Operations Research 18.1 (2010): 47-58.

[4] Hillier, Frederick S., and Camille C. Price. "International Series in Operations Research Management Science." (2001).

[5] Parthasarathy, T., and Sujatha Babu. Stochastic games and related concepts. Hindustan Book Agency, 2020.