CBOT Assignment

Done by
18MCME01 Shruti Patchigolla
18MCME02 Amruta Jandhyala
18MCME03 Arun Kumar Dharavath
18MCME07 Kajjayam Preethi
18MCME14 Bingi Harshitha

Introduction:
It starts with a basis which is the Identity matrix. Then a column of this identity matrix is switched with a non-basic column from matrix A. Then the inverse of the new basis matrix is computed. A product form of inverse to compute the inverse of a matrix should be followed.

Input instructions : (same for both the methods)
1. Variables are the letters x, s, e, R followed by a digit with no spaces in between.
2. x1,x2,... are original variables in the constraints.
   s1,s2,... are slack variables.
   e1, e2,... are excess variables.
   R1, R2,... are artificial variables.
3. Coefficients of variables which are 0 or 1 must be explicitly stated. Eg: 1x1 -1 x2 + 0 s1 + 0 R1 = 10
4. The keywords "Maximise" and "Subjected to " must be given in input.
Objective function should start with "Maximise Z = "...
5. RHS must be nonnegative. Otherwise, absolute value will be taken.

Sample input-1:

```
var x1 >=0;
var x2 >=0;
maximize z: 5*x1+4*x2;
subject to
  c1: 6*x1+4*x2<=24;

  c2: x1+2*x2<=6;

  c3: -x1+x2<=1;

  c4: x2<=2;
solve;
display z,x1,x2;
```

Input in text file:
Maximise Z=5x1+4x2+0s1+0s2+0s3+0s4
Subjected to 6x1+4x2+1s1+0s2+0s3+0s4=24
1x1+2x2+0s1+1s2+0s3+0s4=6
-1x1+1x2+0s1+0s2+1s3+0s4=1

0x1+1x2+0s1+0s2+0s3+1s4=2

<u>Sample Output-1:</u>
x1=[3.]
x2=[1.5]
s3=[2.5]
s4=[0.5]
Maximum value is [21.]

<u>Sample Input-2:</u>

Max $Z = x_1 + 2x_2$
Subject to
$x_1 + x_2 \leq 3$
$x_1 + 2x_2 \leq 5$
$3x_1 + x_2 \leq 6$
and $x_1, x_2 \geq 0$

Input in text file:
Maximise Z = 1x1 + 2x2 + 0s1 + 0s2 + 0s3
Subjected to 1x1 + 1x2 + 1s1 + 0s2 + 0s3 = 3
1x1 + 2x2 + 0s1 + 1s2 + 0s3 = 5
3x1 + 1x2 + 0s1 + 0s2 + 1s3 = 6

<u>Sample Output-2:</u>
s1=[0.5]
x2=[2.5]
s3=[3.5]
Maximum value is [5.]

Here, x1=0, s2=0 since they are non-basic variables.

<u>Revised Simplex Method:</u>

- ***A*** is coefficient of LHS part when slack and artificial variables are present, ***b*** is RHS.
- ***C*** is coefficient of RHS of when slack and artificial variables are present. Example: It should be of the form Z=5x1+4x2+0s1+0s2+0s3+0s4
- Each column in ***A*** has an index.
- **B** is the matrix of coefficients of basic variables.
- In the first iteration we have to select columns from **A** for the basic variable matrix **B** such that it is an identity matrix.
- BVindices list contain the column indices for this.
- The other indices go to NBVindices list.

- **X** is the matrix containing values of the basic variables, while Non-basic variables are set to zero. It is computed by **X=(B inverse)*(b)**.
- Each column in **B** is replaced by a new column from **A**, and **B inverse** is calculated. This is done till an optimal and feasible solution is reached.
- **Z** is the final value of objective function, the maximum value.

Method-1: Gauss Jordan Reduction for inverse of a matrix.

Start with ***A*** inverse as an identity matrix.
a is the column from ***A*** inverse.
We need to do a tensor product of ***w*** and ***a***, then add previous ***A*** inverse to get new ***A*** inverse.
After, n iterations we get final ***A*** inverse.

$(\mathcal{A}_{GJR})$

- data: $A \in \mathbb{R}^{n \times n}$

Initialization of the matrix $(A^{-1})^{(1)} = Id$
for $k = 1$ to n do:
begin
compute: $v^{(k)} = (A^{-1})^{(k)} a_k$
compute: $w^{(k)}$

$$w^{(k)} = \begin{pmatrix} -\frac{v_1}{v_k} \\ \vdots \\ -\frac{v_{k-1}}{v_k} \\ \frac{1}{v_k} \\ -\frac{v_{k+1}}{v_k} \\ \vdots \\ -\frac{v_n}{v_k} \end{pmatrix} \leftarrow k \quad \text{pivot } v_k^{(k)} \neq 0$$

compute: $(A^{-1})^{(k+1)} = (\tilde{A}^{-1})^{(k)} + w^{(k)} \otimes \tilde{a}_k^{(k)}$
end{ k }

$(A^{-1})^{(n+1)}$: inverse matrix of A

Fig. 4. Gauss–Jordan algorithm to construct A^{-1}.

Method 2 : Elementary Matrix form or Product form of inverse

- In Iteration 0 $\boldsymbol{A_b}$ matrix is identity matrix then its inverse is also an identity matrix
- xi leaving column i.e., $\boldsymbol{A_b}$ is matrix having xi column in say l-th column
- xj entering column i.e., $\boldsymbol{A_b}'$ is matrix after iteration 0 having xj column in l-th column
- Now $\boldsymbol{A_b}'$ = A_b*E, d=(A_b^{-1})*xj, E is the identity matrix whose l-th column is d

- $\boldsymbol{A_{b'}^{-1}}$ = $(E^{-1})*(A_b^{-1})$
- E^{-1} will be an identity matrix whose l-th column is
  (-d1/dl -d2/dl.....-d(l-1)/dl 1/dl-d(l+1)/dl......-dm/dl)

<u>Time complexity:</u>
Method 1 : 15.6245 sc
Method 2 : 15.6002 sc

Using the elementary matrix method is comparatively better than Gauss Jordan inverse as it reduces time taken by approx 0.200s.