| |
|---|
| Name: |
| PIN: |

Applied Matrix Algebra
Solving Linear Systems of Equations

*Objective* The objective of this project is to illustrate how solutions to systems of linear equations can be computed using *Mathematica*.

*Narrative* If you have not already done so, read Section 1.3B of the text. In earlier sections we saw the connection between linear systems of equations and matrices. To solve a linear system of equations we use elementary row operations to reduce the system to an equivalent system of equations that directly give the solution. These elementary row operations can be performed on the corresponding augmented coefficient matrix instead of the system of equations. That is the solution of the system of equations does not depend on the names of the variables used in the equations. For example

$$\begin{aligned} 3y - z &= 1 \\ 2x - y + z &= 1 \\ x + y + z &= 6. \end{aligned}$$

This system of equations has an augmented coefficient matrix given by

$$\begin{bmatrix} 0 & 3 & -1 & 1 \\ 2 & -1 & 1 & 1 \\ 1 & 1 & 1 & 6 \end{bmatrix}$$

Using elementary row operations on this matrix we can arrive at an equivalent matrix in reduced row echelon form:

$$\begin{bmatrix} 0 & 3 & -1 & 1 \\ 2 & -1 & 1 & 1 \\ 1 & 1 & 1 & 6 \end{bmatrix} \longrightarrow \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 2 \\ 0 & 0 & 1 & 5 \end{bmatrix}.$$

Then we interpret this resultant matrix to arrive at the solution to the original consistent system of linear equations: $x = -1$, $y = 2$, and $z = 5$ or $(-1, 2, 5)$.

Now we will learn how *Mathematica* can be used to do elementary row operations to reduce matrices and solve systems of equations.

*Task*
Download and run the Linear Systems notebook posted on the course web site. In order to perform the individual row opperations in *Mathematica* we define three commands: **RowSwap**, **RowMult**, and **RowComb**. Evaluate the notebook to define these commands and then type and evaluate the command lines below into the notebook in the order in which they are listed; they illustrate how the above system of equations can be solved step by step using elementary row operations on the augmented coefficient matrix.

Note that the first and last commands suppress the output of the assignment and then displays the matrix in standard form. Also note that to the right of each command is a comment that explains the opperation.

```
A={{0,3,-1,1},{2,-1,1,1},{1,1,1,6}};  (* Enter matrix *)
MatrixForm[A]

A1=RowSwap[A,1,3]                     (* Interchange r1 and r3 *)

A2=RowComb[A1,2,1,-2]                 (* r2 = r2 - 2*r1 *)

A3=RowMult[A2,2,-1/3]                 (* r2 = (-1/3)*r2 *)

A4=RowComb[A3,3,2,-3]                 (* r3 = r3 - 3*r2 *)

A5=RowMult[A4,3,-1/2]                 (* r3 = (-1/2)*r3 *)

A6=RowComb[A5,2,3,-1/3]               (* r2 = r2 - r3 *)

A7=RowComb[A6,1,3,-1]                 (* r1 = r1 - r3 *)

A8=RowComb[A7,1,2,-1];                 (* r1 = r1 - r2 *)
MatrixForm[A8]
```

OK, what did we just do? In *Mathematica* the "(* *)" is used for comments. The curly brackets are used to enter a matrix in *Mathematica* and the **MatrixForm** command is used to display a matrix in standard form. We then use elementary row operations to perform the Gaussian Elimination and Back Substitution on the augmented matrix.

*Task*
Explore the help menu and experiment with the commands: **RowReduce** and **LinearSolve**. Then do the following problems.

(1) Use the **RowSwap**, **RowMult**, and **RowComb** commands in the provided *Mathematica* notebook on the augmented coefficient matrices to solve the following systems of equations.
(a)

$$\begin{aligned} 5y + 2z &= 9 \\ x - 4y - z &= 0 \\ 2x + 3y + z &= 10 \end{aligned}$$

(b)

$$\begin{aligned} x_1 + x_2 + x_3 &= 1 \\ 2x_1 - x_2 + 3x_3 &= 2 \\ x_1 + 3x_2 - x_3 &= 3 \end{aligned}$$

(c)

$$\begin{aligned} x_1 + x_2 - x_3 - x_4 &= -1 \\ 2x_1 + x_3 + x_4 &= 1 \\ 4x_1 + 2x_2 - x_3 - x_4 &= -1 \\ x_1 + 3x_2 - 2x_3 - 2x_4 &= -4 \end{aligned}$$

(2) For the above three systems of linear equations, use the **RowReduce** command to find the solutions sets.

(3) For the above three systems of linear equations, use the **LinearSolve** command to find the solution sets. Comment on any differences that you may notice using this method.